В. М. ГЛУШКОВ

ВВЕДЕНИЕ В АСУ

На современном этапе
технической революции
все большее значение
приобретают вопросы
научной организации труда
и управления,
экономики
промышленного производства,
автоматизации
технологических процессов
с применением
электронных вычислительных машин,
психологии труда и др.
Сегодня каждый инженер,
независимо
от его узкой специализации,
должен не только владеть
основами знаний в этих отраслях
но и быть информированным
о последних достижениях в них.
Именно эти вопросы
определяют тематику
«Библиотеки инженера».

Издательство «Техніка»
Киев
1974

В. М. ГЛУШКОВ

ВВЕДЕНИЕ В АСУ

Издание 2-е,
исправленное и дополненное

6Ф6.5
Г55
УДК 681.14

Введение в АСУ. Глушков В. М. Изд. 2-е, исправленное и дополненное. «Техніка», 1974, 320 стр.

Изложены основные принципы работы ЭВМ и математические методы, употребляющиеся в АСУ (линейное и динамическое программирование, теория массового обслуживания, системный анализ и др.). Даны общие сведения об АСУ и системах обработки данных, принципы построения справочно-информационных систем и комплексов задач планирования и управления, которые необходимо решать в АСУ. Особое внимание уделено управлению запасами и его связи с оперативно-календарным планированием. Описаны задачи управления в макроэкономике и принципы построения общегосударственной автоматизированной системы.
Предназначена для инженерно-технических работников и хозяйственных руководителей всех отраслей промышленности.
Табл. 17, илл. 27, библ. 122.

Рецензент *И. А. Данильченко,* докт. техн. наук

Редакция литературы по энергетике, электронике, кибернетике и связи

Заведующий редакцией инж. *З. В. Божко*

$\Gamma\frac{3314-132}{M202(04)-74}$ БЗ-36-8-74

© Издательство «Техніка», 1974 г., с изменениями.

ПРЕДИСЛОВИЕ

XXIV съезд КПСС уделил большое внимание проблемам совершенствования управления. Поставлены большие задачи в области развития прикладной математики и кибернетики, электронной вычислительной техники и автоматизированных систем управления. В нашей стране проводится комплекс мероприятий, направленный на подготовку и переподготовку кадров по широкому кругу проблем, которые необходимы для совершенствования управления. Важное место в этом комплексе занимают вопросы автоматизации управления на основе применения электронных вычислительных машин (ЭВМ).

Настоящая книга предназначается в первую очередь для того, чтобы помочь широкому кругу специалистов и хозяйственных руководителей, вовлекаемых в автоматизацию управления, получить начальную ориентировку в том широком комплексе вопросов, которые необходимо решать при разработке и внедрении автоматизированных систем управления (АСУ). Книга состоит из трех глав. В первой главе излагаются основные принципы построения электронных вычислительных машин и систем программирования. Во второй главе собраны необходимые сведения о математических методах, применяемых при решении различных задач управления. Третья глава, составляющая основную часть книги, посвящена собственно автоматизированным системам управления.

Учитывая назначение книги, при ее написании был выдержан ряд принципов. Во-первых, это четкое ограничение предмета изложения. Книга посвящена в основном автоматизированным системам организационного управления, предназначенным прежде всего для управления экономическими объектами. Вопросы автоматизированного управления технологическими процессами остаются почти полностью за ее пределами и затрагиваются лишь очень бегло. При выборе материала основной упор сделан на технические, математические и информационные аспекты создания АСУ. Из

широкого круга проблем экономического, социологического и психологического плана, с которыми приходится сталкиваться при разработке и внедрении АСУ, выбран лишь минимум, который наиболее органически связан с пониманием того нового, что вносит автоматизация в управление.

В связи с этим из рассмотрения опущены многие экономические процедуры традиционного характера (например, детали организации бухгалтерского учета), а также ряд более новых задач, хотя и очень важных для эффективного управления, но непосредственно не вытекающих из особенностей АСУ (например, многие задачи, связанные с развитием систем экономического стимулирования).

Во-вторых, книга не рассчитана на то, чтобы служить справочником по существующим ЭВМ и АСУ. Основной упор в ней делается на раскрытие принципов и методов, овладев которыми можно строить новые АСУ и организовать эффективное применение ЭВМ для решения экономических задач. Третья глава, посвященная собственно АСУ, в своей основной части нацелена не столько на настоящее, сколько на будущее автоматизированных систем управления.

Третий принцип касается уровня изложения материала. Имея в виду широкий круг читателей, не являющихся специалистами в области ЭВМ и АСУ, в книге по мере возможности выдерживается наиболее элементарный и доступный метод изложения. От читателя не требуется никаких специальных знаний в области электронной вычислительной техники и программирования. Вместе с тем книга не является просто популярным рассказом об ЭВМ и АСУ. Ее цель — вооружить читателя пусть не исчерпывающим, но доведенным до возможности реального использования комплексом идей и методов, необходимых для разработки автоматизированных систем управления на различных уровнях — от предприятия до проблем макроэкономического характера.

Большой объем материала обусловил необходимость следования еще двум принципам. Во-первых, математические результаты приводятся без доказательства. Основной упор сделан не на то, как эти результаты получены, а как ими пользоваться при решении практических задач. Во-вторых, при нескольких методах решения одной и той же задачи обычно приводится лишь один из них. Предполагается, что читатель, получивший первое представление о предмете из настоящей книги, сможет, в случае необходимости, получить более подробные сведения по заинтересовавшим его вопросам из специальной литературы, список которой приводится в конце книги.

При большом числе хороших книг, посвященных различным конкретным аспектам АСУ, и применяющихся в них математичес-

ких методов, до сих пор, по-видимому, не было книги, в которой весь этот круг вопросов освещался бы в комплексе. Поэтому настоящая книга может оказаться полезной не только для широкого круга лиц, впервые приобщающихся к вопросам автоматизации управления, но и для специалистов, имеющих глубокие знания по отдельным аспектам АСУ и желающих получить представление о всем предмете в целом. Для таких читателей соответствующие разделы книги будут, естественно, казаться элементарными. Вместе с тем можно надеяться, что специалисты по вычислительной технике и программированию найдут для себя немало интересного о математических методах, а специалистам по методам будет небезынтересно познакомиться с принципами построения операционных систем и тенденциями развития ЭВМ.

Кроме того, даже в первых двух главах, носящих в основном обзорный характер, часть материала (например, параграф 8 гл. I и параграф 7 гл. II) основана на новых результатах, не освещавшихся ранее.

Что же касается основной (третьей) главы, то она в своей большей части основана на новых идеях и результатах. Хотя часть этих результатов была получена автором уже относительно давно (в 1964—1968 гг.), но еще нигде не публиковалась, тем более в таком комплексном виде. Можно надеяться, что ознакомление с этими идеями и результатами окажется полезным для широкого круга разработчиков, пользователей и будущих заказчиков автоматизированных систем управления.

Отзывы и пожелания просим направлять по адресу: *252601, Киев, 1, ГСП, Пушкинская, 28, издательство «Техніка»*.

ВВЕДЕНИЕ

Во второй половине двадцатого века произошло резкое увеличение темпов роста науки и техники, получившее название научно-технической революции, которая оказала и продолжает оказывать глубокое влияние на развитие экономики. Одним из ее проявлений является быстрое увеличение сложности управления экономикой. Имеются четыре основные причины такого увеличения.

1. Стремительный рост номенклатуры изделий, выпускаемых народным хозяйством. За последние 20—25 лет номенклатура изделий выросла не менее, чем в десять раз и исчисляется ныне многими миллионами. А ведь за каждым наименованием изделий стоит целый комплекс задач управления (определение потребностей, планирование, материально-техническое снабжение и др.).

2. Резкое увеличение средней сложности выпускаемых изделий и сложности технологии их производства. Если первый фактор касается главным образом изделий новой техники, то второй охватывает все народное хозяйство. В самом деле, даже такой неизменный продукт как каменный уголь производится теперь с помощью гораздо более сложной технологии, чем 20—30 лет назад. В шахты пришли новые, более совершенные машины, металлическая крепь, электроника и автоматика. А это означает появление новых задач материально-технического снабжения, увеличивает количество связей между предприятиями самого различного профиля. А ведь именно связи и являются в первую очередь объектом управления.

3. Увеличение сменяемости выпускаемых изделий, в результате чего задачи управления необходимо решать чаще и быстрее.

4. Научно-техническая революция породила необходимость в решении новых задач управления, не известных или не имевших существенного значения до сих пор. Среди них следует назвать комплекс задач управления самим научно-техническим прогрессом. Если раньше план развития науки и техники можно было составлять до известной степени независимо от основного народно-хозяйственного плана, то теперь положение изменилось. Средний срок от науч-

ного открытия до его применения в промышленности, составлявший в прошлом десятки лет, ныне подходит к пяти годам, а в отдельных случаях составляет два-три года.

Поэтому пятилетние планы (а тем более планы на длительные периоды) должны строиться так, чтобы быть в любой момент готовыми воспринять и использовать результаты любого научного открытия. В связи с этим резко увеличилось значение методов прогнозирования научно-технического прогресса, появились новые эффективные методы решения задач долгосрочного планирования и т. д. В результате еще в большей степени возрастает сложность задач управления экономикой.

Возникает вопрос, а по какому закону изменяется эта сложность? Нетрудно понять, что сложность задач управления экономикой возрастает быстрее, чем растет число занятых в экономике людей. В самом деле, предположим сначала, что рост экономики происходит за счет простого увеличения числа элементарных несвязанных между собой экономических объектов типа средневековых мануфактур. Если n — число таких объектов, k — число работающих на каждом объекте людей, а p — сложность задач управления таким объектом, то общая сложность P всех задач управления экономикой выразилась бы формулой $P = pn$, а общее число N занятых в экономике людей — формулой $N = kn$. Тогда

$$\frac{P}{N} = \frac{p}{k} = \text{const}, \quad \text{или} \quad P = \frac{p}{k} N.$$

Поскольку $\frac{p}{k} = \text{const}$, то в этом простейшем случае имела бы место линейная зависимость между числом занятых в экономике людей и суммарной сложностью задач управления.

Однако хорошо известно, что экономика не развивается таким простейшим способом. В результате развития процессов специализации и кооперации, неизбежных при совершенствовании технологии и организации с целью увеличения производительности труда, между отдельными экономическими объектами возникают дополнительные связи, а значит, и новые задачи управления. Отсюда следует вывод, что *сложность задач управления экономикой растет быстрее числа занятых в экономике людей.*

Уточним, о каких задачах управления идет речь. Дело в том, что существуют задачи управления, целиком или частично обусловливаемые принятой формой организации управления. Примером такого рода задач может служить статистическая или бухгалтерская отчетность, т. е. заполнение некоторых определенных форм, которые могут совершенно измениться или вовсе исчезнуть при изменениях формы организации учета.

Есть и такие задачи управления, которые не зависят от существующих организационных форм, поскольку они определяются не

этими формами, а объективно существующими материальными потоками. Мы будем называть такие задачи объективно необходимыми. Примерами таких задач могут служить задачи согласования потока выпускаемой продукции с потоком материально-технического снабжения или задача распределения задания между отдельными единицами оборудования. Сложность такого рода задач зависит от точности их решения, а степень точности, в свою очередь, определяет размер потерь, которые несут предприятия, отрасли и вся экономика в целом. Если задаться максимально допустимой величиной таких потерь, например в размере не более одного процента от суммарной стоимости выпускаемой продукции, то понятия объективно необходимых задач управления и их сложности приобретают вполне определенный однозначный смысл.

Следует иметь в виду, что в отличие от действительно решаемых задач управления, объективно необходимые задачи могут в данный момент и не решаться или решаться грубо, «на глазок». Результатом этого является увеличение потерь выше принятого условного порога (в данном случае одного процента).

Совершенно очевидно, что для сложности P объективно необходимых задач управления справедлив сделанный выше вывод о ее более быстром росте, по сравнению с числом занятых в экономике людей N. Для точного выражения зависимости между N и P недостаточно одних лишь теоретических построений, необходимы экспериментальные данные. Такие данные показывают, что в современную эпоху сложность объективно необходимых задач управления растет быстрее, чем квадрат числа людей, занятых в экономике: $P > bN^2$.

Этот вывод подтверждается также и тем, что при увеличении общего числа предприятий, которое обозначим через n, относительная доля $\frac{1}{r}$ числа предприятий, с которым в среднем связано одно предприятие, не уменьшается. Если даже принять, что $\frac{1}{r} = \text{const}$, то общее число связей выразится формулой $n\frac{1}{r}n = \frac{n^2}{r}$. Поскольку установленная зависимость имеет место и внутри предприятий, то она в конечном счете оказывается применимой и к отдельным людям, число связей между которыми растет в среднем не медленнее, чем $\frac{N^2}{q}$, *где q — некоторая константа.*

Известно, что суммарная сложность P объективно необходимых задач управления растет быстрее общего числа связей, существующих в экономике. Тем самым снова приходим к выводу, что величина P растет быстрее, чем квадрат N. Если бы даже $P = CN^{1+\alpha}$, где α — любая сколь угодно малая положительная константа (напомним, что зависимость $P = f(N)$ выражается более быстро

растущей функцией, чем линейная функция), то и тогда (а тем более при $P > bN^2$) имеет место следующий вывод:

При достаточно больших значениях N суммарная сложность P всех объективно необходимых задач управления экономикой превзойдет любую константу и любую линейную функцию от N.

Этот результат имеет фундаментальное значение, поскольку из него непосредственно следует, что в развитии сложности задач управления экономикой имеется два порога, при переходе через которые технология управления должна претерпевать коренные изменения. Необходимо подчеркнуть, что речь идет именно о технологии управления, т. е. о методах сбора и обработки экономической информации, процедурах регулирования спроса и предложения, способах подготовки и принятия решений и т. п. Это замечание имеет существенное значение, поскольку проблемы управления экономикой не сводятся к одной лишь технологической стороне. Определяющее значение имеют прежде всего коренные политэкономические проблемы идеологического характера (формы собственности, способы распределения материальных благ и т. п.).

Технология управления представляет собой лишь инструмент или, точнее, набор инструментов. Отдельные составные части этого набора (бухгалтерский учет, рыночный механизм и др.) могут употребляться различными экономическими формациями. Поэтому пороги, о которых шла речь выше, не обязательно совпадают во времени с порогами, отделяющими одну экономическую формацию от другой, а определяются в основном количественными показателями.

На заре развития экономики, когда экономические связи полностью замыкались в рамках ограниченных коллективов (рода, племени, семьи), способностей одного человека хватало, чтобы держать в своей памяти все ресурсы и осуществлять эффективное управление подобным коллективом. Однако способности человека к восприятию, запоминанию и переработке информации ограничены, и по мере развития экономики сложность объективно необходимых задач управления рано или поздно превосходит эти способности.

Этот порог будем называть первым информационным барьером в развитии экономики. Он был достигнут многие тысячелетия тому назад и вызвал соответствующие изменения в технологии управления. Изменения состояли в изобретении двух механизмов, позволивших распараллелить решение задач управления на многих людей. Первый механизм относился к области организации и состоял в использовании вместо одного единственного руководителя иерархических систем управления. Второй механизм экономического характера основан на введении рыночных (товарно-денежных) отношений.

Не вдаваясь в роль рынка, как механизма для распределения товаров, заметим, что он является одновременно и регулятором производства. Каждый индивидуальный акт купли-продажи оказывает влияние на уровень цен, а через него — на увеличение или уменьшение производства тех или иных товаров, являясь тем самым (косвенным) актом управления производством. Рыночный механизм, используя в целях управления каждого покупателя, способен вовлечь (пусть не слишком прямым и эффективным образом) в решение задач управления все взрослое население.

Однако способности к переработке информации даже у всего населения ограничены. Если через N обозначить число всех людей, занятых в экономике (в качестве производителей или покупателей); через A — среднюю способность одного человека к переработке информации, то суммарная способность к переработке информации всех этих людей будет равна AN. Поскольку A можно считать практически постоянной величиной, то AN представляет собой линейную функцию от N. При развитии экономики суммарная сложность P объективно необходимых задач управления рано или поздно превысит AN. Этот момент будем называть вторым информационным барьером в развитии экономики.

После достижения второго информационного барьера никакие меры организационного или экономического характера, направленные на распараллеливание задач управления как путем создания соответствующих организационных структур, так и путем создания любых регуляционных механизмов экономического характера, не смогут обеспечить решения всех объективно необходимых задач управления в полном объеме. Неизбежным результатом будет нарастание трудностей в управлении экономикой и вызываемое им увеличение потерь (выше фиксированного минимального порога).

Единственным выходом из возникшей трудности является увеличение производительности труда в сфере управления, по сравнению с наилучшим образом организованными способами прямого или косвенного вовлечения в решение задач управления всего взрослого населения. Такую возможность сегодня дают только электронные вычислительные машины (ЭВМ) и основанные на их использовании автоматизированные системы управления (АСУ). Именно АСУ являются тем новым инструментом, который в тесном взаимодействии с развивающимися в течение многих тысячелетий инструментами организационного и экономического характера может помочь преодолеть второй информационный барьер.

Возникает естественный вопрос о том, перешла ли современная экономика второй информационный барьер или нет? Ответ на этот вопрос может дать только практика. Попытка выявления и фактического решения всех (или, по крайней мере, большей

части) объективно необходимых задач управления делается сегодня лишь при разработке и внедрении АСУ. Опыт этих работ в течение 60-х годов позволяет оценить общую сложность объективно необходимых задач управления в масштабе всей страны на конец этого периода в 10^{16} арифметических операций в год.

Разумеется, масштабы имеющегося опыта еще недостаточны для получения точных количественных результатов, однако соображения качественного характера* позволяют с большой степенью уверенности утверждать, что приведенная оценка является оценкой снизу и, следовательно, действительная сложность объективно необходимых задач управления больше, чем 10^{16} арифметических операций в год.

Еще более трудной задачей является определение средней производительности человеческого мозга в процессах переработки информации, необходимых для управления экономикой. Тем не менее можно с достаточной степенью уверенности предполагать, что на одну операцию, примерно равную по сложности арифметическим операциям (сложения, вычитания, умножения и деления) над многозначными числами человеку, не вооруженному ЭВМ и АСУ, требуется не менее 10 *сек*. Это подтверждается существующими нормами для операторов, работающих на настольных клавишных арифмометрах. Даже на операцию сравнения цен двух одинаковых товаров и выбора из них более дешевого человек вряд ли затратит менее 10 *сек*. А эта операция в ЭВМ имеет тот же порядок сложности, что и операция вычитания.

Учитывая, что в году около $3 \cdot 10^7$ *сек*, получаем, что при 8-часовой активной работе (без выходных дней и отпуска) человек способен выполнить не более $\left(\frac{1}{3} \cdot 3 \cdot 10^7\right) : 10 = 10^6$ операций в год. Отсюда следует, что для выполнения 10^{16} операций в год (без применения ЭВМ и АСУ) потребуется не менее 10^{10}, т. е. не менее десяти миллиардов человек. Это означает, что экономика страны уже перешла второй информационный барьер и, следовательно, без применения ЭВМ и АСУ все объективно необходимые задачи управления с требуемой полнотой решать сегодня невозможно. Как показывают проведенные оценки, решение всех объективно необходимых задач управления с требуемой ныне полнотой и точностью позволило бы (при всех прочих равных условиях) по меньшей мере в два раза повысить темпы развития экономики.

Таким образом, задача эффективного использования ЭВМ и АСУ для управления экономикой превращается сегодня в одну из

* Эти соображения основываются на том, что объекты, на которых созданы АСУ, являются типичными для народного хозяйства, а также на том, что на них выявлена и реализована только часть объективно необходимых задач управления.

наиболее актуальных задач национального масштаба. Ее решение позволяет в полной мере использовать огромные возможности, заложенные в социалистическом способе ведения хозяйства.

Масштаб предстоящей работы не определяется лишь большим числом арифметических операций, которые необходимо выполнять. Кстати сказать, 10^{16} операций в год можно выполнить при быстродействии порядка 350 млн. операций в секунду, а это ожидаемое быстродействие одной большой ЭВМ в ближайшем будущем или 100 современных больших машин.

Однако, даже располагая мощной ЭВМ, бессмысленно пытаться втиснуть в нее всю экономику. Речь идет о создании эффективной человеко-машинной системы, в которой с наибольшей полнотой был бы использован творческий потенциал людей и которая была бы органической составной частью всего сложного регуляционного механизма, каким является процесс управления экономикой. Для этой цели необходимо создавать АСУ разного уровня, использующих весь спектр ЭВМ, замыкать через них информационные потоки, закладывать в ЭВМ наиболее эффективные методы решения задач планирования и управления. Важно отметить, что наряду с известными экономическими регуляционными механизмами необходимо разрабатывать новые механизмы, наилучшим образом использующие возможности ЭВМ и АСУ.

Ведь каждый регуляционный механизм приспособлен к определенным условиям и лишь в этих условиях работает хорошо. Например, такой известный регулятор, как рынок, хорошо приспособлен для условий, в которых применим закон больших чисел (большое число индивидуальных актов купли-продажи). Рыночный механизм работает эффективно, когда среднее время сменяемости товаров существенно больше времени затухания переходных процессов, возникающих при появлении товара на рынке.

В условиях научно-технической революции все чаще и чаще приходится встречаться со случаями, когда описанные условия не соблюдаются и когда, следовательно, обычный рыночный механизм неприменим. То же самое относится и ко многим другим механизмам регулирования, успешно применявшимся ранее и начинающих отказывать в условиях научно-технической революции. Разработка новых механизмов (в которые после достижения второго информационного барьера обязательно должны быть включены ЭВМ) представляет собой стержневой вопрос построения АСУ. Для успешного решения этой задачи необходимо с одной стороны хорошо понимать принципы работы ЭВМ (включая организацию информационной базы и программирование) и возможности математических методов, а с другой — обладать глубоким знанием производства, реальных процессов, протекающих в экономике.

Глава 1

ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ

1. Принципы представления информации в ЭВМ

Современные электронные вычислительные машины (ЭВМ) представляют собой устройства для автоматического преобразования произвольной буквенно-цифровой информации. При реальном представлении букв и десятичных цифр используются двоичные элементарные сигналы и состояния. Иными словами, элементарный сигнал, переносящий информацию, может принимать лишь два различных значения. Например, высокое или низкое значение напряжения, отсутствие или наличие электрического импульса и т. п. Аналогично при запоминании информации используются два различных состояния какого-либо физического элемента, например намагниченность сердечника в одном или другом направлении. Такие сердечники изготавливаются для ЭВМ в виде маленьких колечек (диаметром 1 *мм* и даже меньше) из специальных магнитных материалов, называемых ферритами. Один обычный сигнал или состояние составляет самую мелкую единицу информации, получившую специальное название *бит*, или *двоичный разряд*.

Буквы, десятичные цифры и другие символы внутри ЭВМ представляются в виде групп двоичных сигналов. Операция их представления в таком виде называется кодированием, или, более точно, двоичным кодированием. С помощью двух двоичных сигналов (двух битов) можно закодировать $2 \times 2 = 4$ различных символа, с помощью трех — $2 \times 2 \times 2 = 8$ символов и т. д. Группа из n двоичных сигналов может использоваться для представления 2^n различных символов. Группа из n двоичных сигналов (разрядов), используемая для представления одного символа, называется *байтом*. Исторически вначале употреблялись 6- или 5-разрядные байты. Однако в последнее время наибольшее распространение начинает получать 8-разрядный байт, имеющий целый ряд преимуществ с точки зрения организации запоминания и поиска информации в ЭВМ. Будучи достаточным для представления $2^8 = 256$ различных символов, 8-разрядный байт дает возможность рассматривать в качестве одного символа пары десятичных цифр, оставляя при этом большой простор для кодирования не только русских

и латинских букв, но и различных других символов (математических, бухгалтерских, типографских и др.). Таким образом, фактически для представления одной десятичной цифры употребляется не 8, а лишь 4 двоичных разряда. Обычно в тексте для обозначения двоичных сигналов употребляются цифры 0 и 1. Тогда при обычном способе кодирования десятичная цифра 0 представляется двоичным кодом 0000, десятичная цифра 1 — кодом 0001, десятичная цифра 2 — кодом 0010, десятичная цифра 3 — кодом 0011 и т. д. Малая латинская буква *а* представляется кодом 1000000, восклицательный знак — кодом 01011010 и т. п.

Более крупной единицей информации является машинное слово При числовой информации машинное слово представляет собой одно число. Длина слова (т. е. количество составляющих его двоичных разрядов) определяет в этом случае точность проводимых ЭВМ расчетов. Длина машинного слова различна в разных машинах и может изменяться в зависимости от их назначения. В управляющих ЭВМ (предназначенных для управления технологическими процессами) ввиду малой точности исходных данных (получаемых от различных измерительных приборов — датчиков) нет необходимости в большой точности вычислений. Длина машинного слова в таких ЭВМ колеблется в пределах 16—24 двоичных разрядов. В ЭВМ, предназначенных для научных расчетов, длина машинного слова повышается до 32 разрядов, а в случае особо большой точности вычисления — до 64 разрядов и даже выше. Столь же высокая точность расчетов обычно бывает нужна при решении сложных задач планирования и управления экономикой.

В случае необходимости еще более точных вычислений современным ЭВМ придается возможность оперировать с так называемыми словами переменной длины. В отличие от слов фиксированной длины, рассматривавшихся выше, слова переменной длины могут изменять число составляющих их двоичных разрядов в довольно широких пределах (практически неограниченно). За такую возможность приходится, однако, платить уменьшением скорости вычислений. В случае слов фиксированной длины арифметические операции над представляемыми им числами выполняются параллельно. Иными словами, необходимые действия производятся со всеми составляющими число цифрами практически одновременно (с точностью до скорости распространения электрического сигнала). В случае слов переменной длины операции выполняются последовательно, цифра за цифрой подобно тому, как это делает человек, выполняя вычисления на листе бумаги.

В словах переменной длины, как правило, употребляется двоично-десятичное представление чисел, когда каждая составляющая число десятичная цифра представляется соответствующим двоичным кодом (обычно четверкой двоичных разрядов). В дробных

и смешанных числах запятая, отделяющая целую часть от дробной, обычно кодируется отдельным байтом и может находиться в любом месте. Введение подобной плавающей запятой при последовательном способе выполнения операций не вносит существенных усложнений в конструкцию машины. При параллельных вычислениях (со словами фиксированной длины) положение изменяется. В случае фиксированной запятой, когда количество разрядов как в целой, так и в дробной части числа (машинного слова) определено раз и навсегда, устройство машины существенно упрощается. Вместе с тем введение ограничения на положение запятой усложняет программирование, поскольку при постановке задач на машину пользователь должен принимать во внимание не только требуемую относительную точность вычислений, но и абсолютную величину чисел, с которыми предстоит оперировать машине.

Поэтому в последнее время все большее распространение получает модульный принцип построения ЭВМ, при котором пользователь может наращивать возможности своей машины, оснащая ее дополнительными блоками. В минимальном комплекте возможности выполнения арифметических операций ограничиваются числами фиксированной длины с фиксированным положением запятой. Один из дополнительных блоков обеспечивает выполнение операций над числами фиксированной длины с плавающей запятой, а другой — над числами переменной длины.

При параллельном способе выполнения операций над числами фиксированной длины более выгодным является употребление не десятичной, а двоичной системы счисления. В двоичной системе, в отличие от десятичной, каждый следующий разряд в два раза (а не в десять) больше предыдущего. Так, например, число 27 в двоичной системе счисления записывается в виде 11011 ($2^4 \cdot 1 + 2^3 \times 1 + 2^2 \cdot 0 + 2^1 \cdot 1 + 2^0 \cdot 1 = 1 + 2 + 0 + 8 + 16 = 27$). Для его представления требуется всего лишь пять двоичных разрядов. При двоичном кодировании каждой десятичной цифре (такая система представления чисел называется двоично-десятичной) потребовалось бы восемь двоичных разрядов (по четыре разряда на каждую десятичную цифру). Число 27 представляется при этом в виде 00100111.

Числа фиксированной длины с плавающей запятой внутри машины представляются обычно в виде пары чисел: целого числа m (со знаком или без знака), называемого характеристикой, и второго числа n (со знаком «+» или «—»), называемого мантиссой, которое является чисто целым или чисто дробным (без целой части). Этой парой представляется число $2^{k(m-l)} n$, где k и l — некоторые фиксированные целые положительные числа, зависящие от типа ЭВМ (в машинах старых выпусков обычно принималось $k = 1$ и $l = 0$).

Поскольку $2^{10} = 1024 \approx 1000 \approx 10^3$, то при записи числа в двоичной системе каждые 10 двоичных разрядов соответствуют примерно трем десятичным разрядам. Таким образом 64-битовые машинные слова с фиксированной запятой при использовании одного бита для изображения знака числа («+» или «—») позволяют кодировать 18—19-значные числа ($63 \cdot \frac{3}{10} = 18{,}9$). Такая точность оказывается заведомо достаточной для подавляющего большинства ЭВМ. Даже затратив несколько десятичных разрядов на представление характеристики, имеем вполне достаточную точность для представления мантиссы.

В ряде современных ЭВМ употребляется несколько фиксированных форматов слов, например 16,32 и 64 *бит*, что дает возможность варьировать точность вычислений.

Буквенная и смешанная (буквенно-цифровая) информация чаще всего организуется в слова переменной длины. Совокупность таких слов, органически связанных между собой, составляет еще более крупную единицу информации, которую принято называть записью. Примерами записей могут служить обычная кадровая анкета, паспорт станка или другой единицы оборудования. Для более эффективного использования ЭВМ желательно по возможности сокращать размеры слов, составляющих запись. Например, в кадровой анкете, в графе «квалификация» вместо слов «токарь 7-го разряда» желательно использовать какой-либо сокращенный символический (буквенно-цифровой, буквенный или чисто цифровой) код, рассматривая его как одно слово. Совокупность символических кодов, описывающих те или иные группы объектов (материалы, готовые изделия, специальность, предприятия, отрасли промышленности и т. п.) носит название *классификатора* соответствующей группы объектов. Создание и непрерывное обновление различного рода классификаторов является одним из условий эффективного применения ЭВМ для задач планирования и управления экономикой.

Записи, имеющие одну и ту же структуру и относящиеся к одноименным объектам, объединяются в (информационные) массивы, называемые также файлами; совокупность кадровых анкет (в машинном представлении) составляет кадровый массив, совокупность паспортов оборудования — массив оборудования и т. п. Меньшие массивы могут входить в более крупные массивы в качестве их подмассивов. Так, массивы оборудования по отдельным предприятиям могут быть объединены в более крупный массив, описывающий оборудование всех предприятий отрасли; массивы описаний станков, прессов, испытательных стендов и других видов оборудования — в массив описания всего оборудования завода и т. п.

2. Принципы организации запоминания и выборки информации в ЭВМ

Информация, с которой работает ЭВМ, помещается в запоминающие устройства (ЗУ), или устройства памяти. Обработка информации в современных ЭВМ производится пословно: из запоминающих устройств выбираются необходимые машинные слова, которые направляются в так называемое арифметическое устройство (АУ), где над ними выполняются те или иные операции (сложение, умножение и др.).

Полученные результаты направляются опять в ЗУ и запоминаются в нем. В современных арифметических устройствах выполняются не только арифметические, но и логические операции, обработка буквенной информации и др. Поэтому более правильно называть их операционными устройствами (ОУ).

Для того чтобы осуществить переработку информации, операционные устройства должны предварительно ее запомнить. Такое запоминание делается на особых устройствах, называемых регистрами, каждый из которых служит обычно для запоминания одного машинного слова. Регистры операционного устройства являются наиболее быстродействующим видом ЗУ. Они строятся из дорогих элементов, и потому общее число их не может быть большим (в лучшем случае один-два десятка). К каждому из регистров имеется отдельный доступ.

В быстродействующих машинах запись или считывание информации может выполняться несколько десятков миллионов раз в секунду, т. е. рабочая частота элементов современных ЭВМ достигает несколько десятков мегагерц. При 64-битовых (8-байтовых) регистрах скорость считывания и записи информации на регистры может составить несколько сот миллионов байт в секунду. Даже для самых медленных полупроводниковых элементов, уже отживающих свой век, рабочая частота исчисляется сотнями килогерц, что соответствует скоростям считывания и записи на 32-битовые регистры порядка миллиона байт в секунду.

Объем памяти на регистрах в операционном устройстве никогда не бывает большим и лишь в редких случаях превышает 100 *байт**, поэтому основная нагрузка по обеспечению операционного устройства информацией в процессе решения задач ложится на так называемое оперативное запоминающее устройство (ОЗУ).

В ОЗУ может храниться относительно большое число машинных слов (от нескольких десятков до многих сотен тысяч) при скорости считывания и записи в сотни тысяч и даже миллионы слов

---

* Это утверждение не касается объема сверхоперативных ЗУ, о которых речь пойдет ниже.

в секунду. Характерным признаком современных ОЗУ является прямой доступ к любому хранящемуся в них слову по так называемому адресному принципу.

Сущность этого принципа заключается в следующем. Все ОЗУ разбито на ячейки, в каждой из которых может храниться одно слово. Ячейки нумеруются последовательными целыми числами, называемыми адресами соответствующих ячеек. Нумерация обычно начинается с нуля, так что при общем числе ячеек N адресом последней ячейки будет служить $N - 1$. ОЗУ состоит из отдельных блоков, каждый из которых снабжается своим собственным устройством для записи и считывания информации. Это устройство состоит из коммутатора и двух регистров A и B (рис. 1).

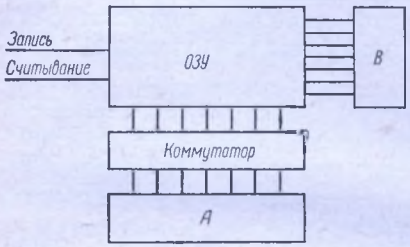

Рис. 1.
Структурная схема ОЗУ.

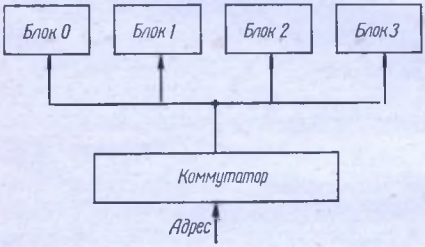

Рис. 2.
Структурная схема блочного ОЗУ.

На регистре A, называемом регистром адреса, устанавливается двоичный код адреса требуемой ячейки, после чего коммутатор открывает для обмена с регистром B ту ячейку, которая соответствует этому адресу. Подавая управляющий сигнал по каналу считывания, осуществляется передача слова из выбранной ячейки ОЗУ в регистр B. Сигнал по каналу записи вызывает запись в выбранную ячейку содержимого регистра B. Регистр B носит название регистра числа (машинного слова); он служит связующим звеном между ОЗУ и операционным устройством машины.

Если ОЗУ состоит из нескольких блоков, то код адреса направляется на регистр адреса соответствующего блока через специальный коммутатор, как показано на рис. 2.

Код адреса состоит из двух частей: кода номера блока и кода адреса ячейки в данном блоке. Первый код анализируется коммутатором, и в зависимости от результатов такого анализа вторая часть кода направляется в регистр адреса соответствующего блока.

Оказывается выгодным под номер блока отводить не старшие, а младшие разряды общего кода адреса. Например, двоичный код 1011011 будет адресом 22-й ячейки ($10110 = 16 \cdot 1 + 8 \cdot 0 + 4 \cdot 1 + 2 \cdot 1 + 0 = 22$) в 3-м блоке ($11 = 2 \cdot 1 + 1 = 3$).

Особенность подобной нумерации состоит в том, что ячейки, адресуемые соседними номерами, находятся в разных блоках. Так, код 1011011, трактуемый как одно число, равен $64 \cdot 1 + 32 \cdot 0 + 16 \cdot 1 + 8 \cdot 1 + 4 \cdot 0 + 2 \cdot 1 + 1 = 91$.

Соседние с ним адреса 90(1011010) и 92(1011100) будут адресовать ячейки во 2-м и в 0-м блоках. Поэтому выборку или запись по адресам 90, 91 и 92 можно производить одновременно. Поскольку в ЭВМ чаще всего после выборки или записи слова из той или иной ячейки ОЗУ приходится обращаться к следующей по порядку ячейке, то возможность описанной параллельной выборки (или записи) сильно увеличивает скорость обмена данными между ОЗУ и другими частями машины. Увеличить скорость обмена можно также за счет удлинения ячеек ОЗУ, чтобы хранить в них не по одному, а по несколько машинных слов.

Время выборки и записи длинных слов (8—9 *байт*) в лучших современных ОЗУ имеет порядок 1 *мксек*. При наличии одновременного доступа к четырем блокам ОЗУ скорость обмена достигает 30—40 млн. *байт/сек*. Чтобы наглядно оценить эту скорость, достаточно вспомнить, что страница печатного текста содержит приблизительно две тысячи символов, т. е. один миллион байт. Это том объемом 500 страниц.

Возможность чтения и записи информации со скоростью 30—40 томов в секунду выглядит достаточно внушительно. Однако в особо быстродействующих операционных устройствах даже эта скорость может оказаться недостаточной для обеспечения их информацией. Поэтому наряду с оперативными ЗУ, имеющими время цикла (записи или считывания) порядка 1 *мксек*, в особо быстродействующие ЭВМ включается еще сверхоперативное ЗУ (СОЗУ) со временем цикла, на порядок меньшим. Операционное устройство работает в основном с этим ЗУ. Что же касается ОЗУ, то его роль состоит в том, чтобы своевременно передавать в СОЗУ целые блоки слов, подлежащие обработке, и принимать на хранение обработанные блоки. При работе операционное устройство обращается к словам СОЗУ по нескольку раз, так что, несмотря на меньшую скорость обмена информацией между СОЗУ и ОЗУ, последнее обычно не задерживает сколько-нибудь значительно процесс обработки.

В современных ЭВМ для кодирования адресов используется двоичная система счисления, поэтому отдельные блоки памяти делают так, чтобы общее число ячеек в них составляло некоторую степень двойки. При этом наилучшим образом используются возможности регистров адреса и коммутаторов соответствующих блоков.

Для исчисления объема (числа ячеек) ЗУ принята специальная единица измерения, равная $2^{10} = 1024$ ячейки. Эта единица обозначается *К* и читается как «*кей*». Грубо можно считать, что 1*К* равняется тысяче ячеек. В настоящее время размеры блоков ОЗУ

колеблются от 1—4*К* (для малых управляющих машин) до 8—64*К* (для больших и средних машин).

При 8-байтовых словах в одном блоке может храниться до 256 тыс. *байт* информации. Максимальное число подключаемых к машине блоков ОЗУ лимитируется принятой в ней максимальной длиной кода адреса и возможностями коммутатора блоков. Наибольший потенциально возможный объем ОЗУ в современных ЭВМ доходит до 16 млн. *байт*. Объем сверхоперативных ЗУ значительно меньше. Он исчисляется обычно несколькими тысячами байт.

Возможность постепенного наращивания объема ОЗУ путем покупки и установления новых блоков наталкивается также и на ограничения чисто экономического характера. Несмотря на значительное снижение цен на ОЗУ, происшедшее за последние годы, эти устройства продолжают оставаться достаточно дорогими. Для грубой ориентировки можно считать, что цена одного байта объема ОЗУ на мировом рынке составляла в 1970 г. до 20 центов*. Таким образом, для комплектования ОЗУ емкостью в 16 млн. *байт* пришлось бы купить блоки стоимостью порядка трех миллионов рублей. Поэтому при комплектовании системы стремятся комплектовать ОЗУ в соответствии с имеющимися потребностями и оптимизировать его использование.

Современные ОЗУ изготавливают в основном из магнитных сердечников диаметром порядка 1 *мм* и менее. Намагниченность сердечника в одном или другом направлении соответствует запоминанию в нем 0 или 1. Чем меньше диаметр сердечников, тем большее быстродействие. Стремление получить быстродействующие малогабаритные ОЗУ приводит к замене сердечников тонкими магнитными пленками. В последнее время появились ОЗУ на малогабаритных (так называемых интегральных) полупроводниковых элементах, которые постепенно вытесняют ОЗУ на магнитных сердечниках.

Помимо указанного принципа непосредственной адресации, в современных ЭВМ обычно предусматривается возможность косвенной адресации по адресам второго и более высоких рангов. При обращении по адресу второго ранга слово *m*, которое расположено в ячейке с указанным адресом, должно рассматриваться как код адреса. По этому адресу производится новое обращение к соответствующей ячейке, и лишь результат этой второй выборки является окончательным. При адресации по адресу *n*-го ранга всего производится *n* обращений, и лишь результат последнего трактуется как искомое слово.

Большое значение имеет выбор из ЗУ слов не по адресам, а по

* Цена эта, разумеется, зависит от быстродействия и некоторых других характеристик ОЗУ.

тем или иным признакам, включенным в состав самих этих слов. В обычных ЗУ решение подобной задачи требует последовательного перебора всех или значительной части ячеек. В так называемых *ассоциативных* ЗУ соответствующее слово выбирается непосредственно. Принцип работы ассоциативного ЗУ можно пояснить следующим простым примером. Пусть имеется некоторое количество карточек, пронумерованных последовательными номерами от 1 до n. Для быстрого поиска карточек по их номерам можно употребить два приема. Первый состоит в том, чтобы разложить их по пронумерованным ячейкам так, чтобы в ячейке с номером m лежала карточка с номером m. Это — аналог адресного поиска.

Второй способ заключается в том, что на краях всех карточек пробивается n отверстий и все отверстия за исключением того, которое соответствует номеру данной карточки, прорезаются до краев. На рис. 3 показано, каким образом выполнены пробивки и разрезы для карточки № 4. Теперь карточки можно сложить в кипу в любом порядке. При необходимости найти карточку № 4, в 4-е от края отверстие продевается спица (сквозь все карточки). При подъеме спица вытащит лишь требуемую карточку № 4. Это пример ассоциативного поиска.

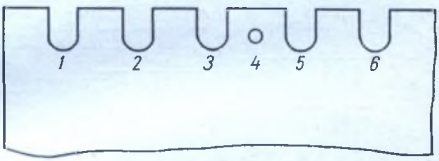

Рис. 3.
Карта с краевой перфорацией.

В отличие от указанного механического принципа в ЭВМ ассоциативная выборка должна основываться на принципах электроники, чтобы обеспечить высокую скорость работы. Это связано с определенными трудностями, из-за чего ассоциативные ЗУ строятся пока лишь для малых объемов и еще не получили широкого распространения.

Оперативные запоминающие устройства далеко не исчерпывают всех проблем хранения информации в ЭВМ. Главная проблема заключается в организации работы с большими массивами информации, объектами в сотни миллионов и миллиардов байтов. Решение этой проблемы дают так называемые внешние запоминающие устройства (ВЗУ), строящиеся на электромеханическом принципе. Наибольшее распространение сегодня получили три вида таких устройств: магнитные барабаны, магнитные диски и магнитные ленты.

Магнитный барабан (рис. 4) представляет собой покрытый слоем магнитного материала цилиндр, вращающийся с большой скоростью — 1500—3000 *об/мин* для больших барабанов, 6000 *об/мин* и выше — для малых. Вдоль образующих цилиндра располагаются

блоки магнитных головок, позволяющих записывать (на магнитном слое) и считывать дискретную информацию (последовательности сигналов 0 и 1). Каждой головке соответствует узкая кольцевая полоска на поверхности барабана, называемая рабочей дорожкой, на которой размещаются записываемые сигналы. Адресация информации на барабане осуществляется указанием номеров дорожек и места на дорожке.

В устройстве управления выборкой имеется коммутатор, анализирующий первую часть адреса в регистре адреса *А* и выбирающий необходимую головку (или группы головок). Распознавание места записи или считывания на дорожке (угла поворота барабана) происходит путем непрерывного счета специальных меток, записанных на особую дорожку. После каждого оборота счетчик останавливается на нуле, и счет начинается сначала. Содержимое счетчика непрерывно сравнивается с заданным адресом места и при их совпадении устройство управления выборкой открывает канал для передачи (слово за словом) необходимой информации через специальный регистр *В* из барабана в ОЗУ (режим чтения) или из ОЗУ на барабан (режим записи).

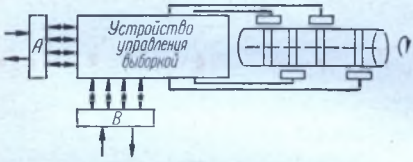

Рис. 4.
Схема магнитного барабана.

Обмен данными между ОЗУ и внешними ЗУ осуществляется не отдельными словами, а целыми группами слов, называемыми *блоками*. Размер блоков может быть фиксированным или переменным. Во втором случае одновременно с передачей в устройство управления выборкой адреса начала блока передается также размер блока. Выгодность передачи информации достаточно крупными блоками определяется тем, что внешние ЗУ тратят довольно много времени на поиск места, на котором записана требуемая информация. Это время, называемое *временем доступа*, для магнитных барабанов определяется скоростью их вращения. При скорости 3000 *об/мин* один оборот совершается за 0,02 *сек*. Таким образом, *среднее время доступа* (время на установку головки и поворот барабана) составляет 0,01 *сек*.

Скорость передачи данных после нахождения места записи можно достаточно увеличить. Это чрезвычайно важно для своевременного обеспечения ОЗУ необходимой информацией. Скорость передачи данных для магнитного барабана определяется его размерами, скоростью вращения и плотностью записи информации вдоль дорожки. В настоящее время имеются барабаны с плотностью записи 60 двоичных разрядов на один миллиметр длины дорожки. При диаметре барабана 60 *см* общая длина дорожки составляет

около 2 *м*, т. е. на одной дорожке может разместиться 120 тыс. *бит* информации. При скорости вращения 3000 *об/мин* (50 *об/сек*) этот объем информации может быть считан или записан за 1/50 *сек*, что увеличивает скорость обмена до 6 млн. *бит/сек* или около 800 тыс. *байт/сек*. Скорость можно увеличить при параллельной работе нескольких головок. Общее число головок на одном барабане может доходить до 800—1000 шт. Это означает, что на одном барабане можно разместить 100—120 млн. *бит*, т. е. 12—15 млн. *байт* информации. Разумеется, такими большими барабанами снабжаются, в основном, лишь высокопроизводительные машины. Для машин среднего класса показатели как по емкости, так и по скорости передачи данных являются значительно более скромными. Для увеличения общей емкости ЗУ на магнитных барабанах оно комплектуется из нескольких барабанов, объединенных общим устройством управления и составляющих общее информационное поле.

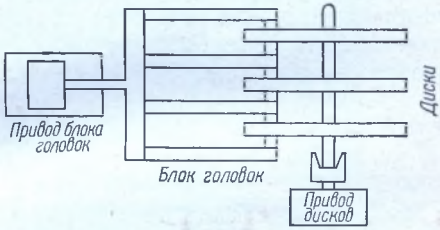

Рис. 5.
Схема магнитного диска.

Магнитные барабаны обладают целым рядом серьезных недостатков. Прежде всего, наличие большого числа головок и в соответствии с этим достаточно сложного коммутатора значительно удорожает установку. Далее, запись информации лишь на одной поверхности цилиндра не дает возможности эффективно использовать для хранения информации весь объем установки. Наконец, барабаны нельзя быстро заменять, как это делается, например, при замене бобин с магнитной лентой на обычном магнитофоне.

Все указанные причины приводят к тому, что барабаны все в большей и большей степени заменяются более эффективными и перспективными внешними ЗУ на магнитных дисках. Носителями информации в таких устройствах являются *кипы* (или пакеты) магнитных дисков. Они представляют собой несколько плоских круглых пластин, закрепленных на общем валу и покрытых с обеих сторон магнитным материалом. Различают устройства с постоянными (несменяемыми) и сменными дисками.

В первом случае кипа дисков жестко скрепляется с приводящим ее во вращение механизмом; во втором — кипу можно быстро снимать и вновь устанавливать на приводной механизм. Сменные пакеты дисков позволяют организовывать накопление и длительное хранение информации в специальных архивах. В то же время количественные показатели (по объему и скорости передачи данных) обычно бывают лучшими для устройств с несменяемыми дисками.

Коэффициент использования объема у дисковых ЗУ значительно лучше, чем у ЗУ на магнитных барабанах. Запись и считывание информации на дисковых ЗУ может осуществляться двумя различными путями. В устройствах с *неподвижными головками*, как и в барабанах, имеется по одной головке на каждую дорожку. Они представляют собой обычные магнитные барабаны с увеличенной поверхностью. В устройствах с *подвижными головками* имеется по одной головке на каждую поверхность дисков (по две головки на один диск). С помощью специального привода (управляемого определенными разрядами кода адреса) головка перемещается вдоль радиуса и устанавливается на ту или иную дорожку. Тем самым резко уменьшается общее число головок и соответственно упрощается коммутатор. Принципиальная схема подобного устройства изображена на рис. 5.

Количественные показатели современных ЗУ на дисках

Среднее время доступа	0,05—0,2 *сек*
Максимальный объем на одну установку:	
для постоянных дисков	630 млн. *байт*
для сменных дисков	7—30 млн. *байт*
Скорость передачи данных:	
для устройств небольшого объема	150—1000 *К байт/сек*
для самых больших устройств	100 *К байт/сек*

Внешние ЗУ на магнитных дисках комплектуют из нескольких устройств; они имеют общую емкость, измеряемую многими миллиардами байтов (многие тысячи томов информации). В этом отношении ЗУ на магнитных дисках намного превосходит ЗУ на магнитных барабанах, емкость которых сравнима с емкостью больших ОЗУ. Вместе с возможностью создания больших архивов на пакетах дисков это обстоятельство определяет решающее преимущество дисковых ЗУ. Стоимость ЗУ на магнитных дисках в пересчете на один байт хранимой информации на два-три порядка меньше, чем в современных ОЗУ. Еще более дешевыми являются ЗУ на магнитных лентах. Они набираются из отдельных устройств — лентопротяжек (рис. 6), которые представляют собой большие магнитофоны, управляемые автоматически с помощью ЭВМ.

Одно устройство управления объединяет обычно несколько (4—8) лентопротяжек. Оно получает от ЭВМ команды на начало перемотки ленты в том или другом направлении, остановку, поиск зоны, начало и окончание считывания или записи информации и выполняет эти команды.

В большинстве современных лентопротяжек употребляется лента шириной 1/2 дюйма (1,26 *см*) и длиной порядка 1 *км*. Дорожки, на которых записывается информация, располагаются вдоль ленты.

Обычно предусматривается восемь основных дорожек (для побайтовой записи информации) и 1—2 дополнительные дорожки, употребляемые для контроля правильности записи и считывания информации, для размещения специальных меток фиксации различных зон на ленте и т. п. Плотность записи составляет 30—60 двоичных разрядов на миллиметр дорожки. Скорость перемотки ленты 2—5 *м/сек*. На одной ленте может храниться 100—120 млн. *байт* информации. Скорость передачи информации 300 тыс. *байт/сек*. Даже для небольших машин (например, для машины Минск-32) емкость одной ленты достигает 12 млн. *байт*. В ближайшие годы ожидается дальнейшее улучшение этих показателей.

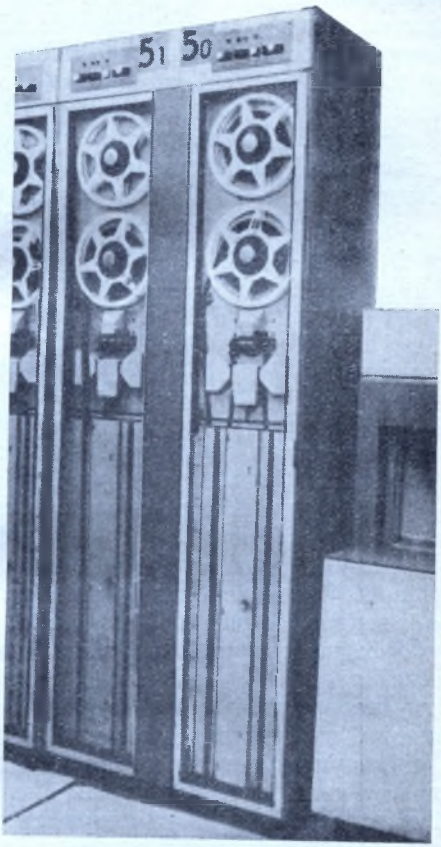

Рис. 6.
Лентопротяжка.

Основным недостатком ЗУ на магнитных лентах является большая величина времени доступа. При 2-километровой ленте ее полная перемотка даже при скорости 5 *м/сек* занимает 400 *сек*, т. е. около 7 *мин*. В этом отношении ЗУ на магнитных лентах резко отличаются от ЗУ на магнитных барабанах и дисках, в которых время доступа на 3—4 порядка меньше и которые называются внешними ЗУ с прямым доступом. В то же время магнитные ленты еще в большей мере, чем сменные диски, приспособлены для длительного хранения, создания архивов и пересылки на большие расстояния при организации межмашинного обмена информацией.

Все описанные ЗУ являются *двусторонними*, т. е. обладают способностью как считывания, так и записи информации. При

этом они строятся так, что при считывании информация остается в ЗУ неизменной. При записи новой информации на место, которое было уже занято, информация (старая) автоматически стирается.

В ряде случаев оказывается выгодным использовать наряду с двусторонними также односторонние, или *долговременные*, запоминающие устройства (ДЗУ). ЭВМ может только читать информацию из таких устройств, причем со скоростями, даже превосходящими скорость работы соответствующих двусторонних ЗУ. Что же касается записи информации, то она осуществляется в результате более медленных процессов и, как правило, вне машины. Поэтому ДЗУ применяются для хранения неизменной либо, по крайней мере, медленно меняющейся информации (программ, справочных таблиц, словарей и т. п.).

Существует несколько типов ДЗУ. ДЗУ на ферритовых сердечниках по принципу работы похоже на обычное ферритовое ОЗУ, о котором шла речь выше. Однако в отличие от ОЗУ информация запоминается не путем перемагничивания сердечников, а специальной прошивкой сердечников системой проводов: прохождению проводника сквозь сердечник соответствует единица, а обходу им сердечника — нуль. Подобное ДЗУ дешевле, чем ОЗУ соответствующего объема, надежнее его, поскольку потеря информации в нем практически исключена. Скорость работы такого ДЗУ также оказывается больше, чем у ОЗУ такого же объема.

Второй тип ДЗУ в качестве носителя информации употребляет фотопленку с обычным или голографическим изображением некоторой матрицы, состоящей из черных и светлых точек (нулей или единиц). Считывание осуществляется электронно-оптической системой, использующей либо лазерный луч с управляемым отклонением, либо фокусируемое на пластинку световое пятнышко, которое высвечивает на экране катодной трубки (напоминающей кинескоп телевизора) управляемый электронный луч ДЗУ. Этот тип ДЗУ может быть использован и для записи информации на пластинку с последующим ее проявлением. Таким образом, запись осуществляется лишь один раз, перезапись (на той же самой пластинке) исключается. Важным преимуществом подобного типа ДЗУ является возможность относительно простого изменения хранимой в нем информации путем замены пластинок. Особенно перспективными являются голографические ДЗУ, которые могут запоминать большой объем информации (многие десятки миллионов бит) при больших скоростях выборки.

Перспективно ДЗУ очень большой емкости, в котором электронно-оптический принцип считывания совмещается с возможностью механического перемещения носителя (обычной кинопленки) подобно тому, как это делается в магнитных лентах. Такие устройства,

как и магнитные ленты, имеют большое время доступа, но способны нести гораздо больший объем информации.

Имеются и другие типы ДЗУ на металлизированных картах, на диодных матрицах и др.

3. Принципы организации ввода и вывода информации в ЭВМ

Вывод и особенно ввод информации — наиболее узкое место современных ЭВМ. Поэтому при проектировании основанных на использовании ЭВМ автоматизированных систем управления и систем обработки данных вопросам ввода и вывода необходимо уделять особое внимание.

В современных ЭВМ в качестве устройств ввода и вывода часто используются электрифицированные пишущие машинки и буквопечатающие телеграфные аппараты — телетайпы. Они позволяют человеку осуществлять непосредственный ввод информации в ОЗУ набором соответствующих букв и цифр на клавиатуре. Скорость такого непосредственного ввода очень мала — всего несколько символов в секунду, что представляет собой ничтожно малую величину по сравнению со скоростями передачи обработки данных внутри ЭВМ. Поэтому непосредственный ввод целесообразно применять при одновременной работе ЭВМ с большим числом пишущих машинок и телетайпов и при условии продолжения решения задач машиной во время ввода данных. Такие возможности обеспечиваются современными ЭВМ за счет специальной организации их структуры и построения довольно сложной системы управляющих программ (см. ниже). ЭВМ более старых выпусков подобными свойствами не обладают.

При соблюдении указанного условия параллельной работы электрифицированные пишущие машинки и телетайпы могут использоваться и для вывода информации. Автоматическая (под управлением ЭВМ) печать в обычных (рычажных) машинках может производиться со скоростью 7—8 символов в секунду. В специальных машинках, в которых применяется для печати сменяемый поворачивающийся шарик с нанесенными на него литерами, достигается скорость автоматической печати до 20—25 символов в секунду. Управление со стороны ЭВМ (по специальной программе) обеспечивает необходимые перемещения каретки, оставление пробелов, автоматическую остановку в случае окончания запаса бумаги и др.

Непосредственный ввод, совмещенный с выводом, дает возможность организовать (с помощью специальных программ) *диалог* человека с машиной и создавать различного рода *человеко-машин-*

ные системы обработки информации. При необходимости ввода больших объемов информации непосредственный ввод оказывается слишком медленным даже при параллельной работе. На смену ему приходит метод *предварительной подготовки данных* для их последующего быстрого ввода в ЭВМ.

В процессе предварительной подготовки информация фиксируется на промежуточном *носителе*, приспособленном для быстрого автоматического ввода в ЭВМ через специальные вводимые устройства. Исторически первыми и до сих пор еще широко применяемыми промежуточными носителями явились так называемые перфокарты и перфоленты.

Перфокарта (рис. 7) представляет собой кусок плотной бумаги длиной 187,32 *мм*, шириной 82,5 *мм* и толщиной 0,175 *мм*. Все поле перфокарты разбито на 80 вертикальных колонок, в каждой из которых имеется 12 позиций: 10 — для десяти цифр — 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 и 2 — для дополнительных знаков. Пробивка в колонке одного прямоугольного отверстия в цифровой позиции означает запись в эту колонку соответствующей цифры. Буквы и другие символы кодируются отверстиями от 1 до 7. На перфокарте, изображенной на рис. 7, в 12-м столбце записана цифра 2, в 16-м — цифра 5, в 34-м — буква *А*, в 37-м — буква *Ц*, в 44-м столбце — символ !

Для пробивки (перфорирования) перфокарт употребляются специальные устройства — *карточные перфораторы*. На рис. 8 изображен стандартный перфоратор для ручной пробивки перфокарт. Оператор набирает необходимую информацию на клавиатуре, а перфоратор пробивает отверстия в соответствующих позициях. Для проверки правильности пробивки употребляется второе устройство, называемое *контрольником*. Оно работает точно так же, как и перфоратор, только вместо пробивки осуществляет проверку наличия уже готового отверстия на соответствующем месте. Оператор на контрольнике получает колоду пробитых перфокарт вместе с первичными документами, с которых информация переносилась на карты. Оператор контрольника снова, символ за символом набирает на клавиатуре информацию, содержащуюся в первичных документах. При обнаружении несоответствия набранной и пробитой информации соответствующая перфокарта поступает в карман брака. Для забракованных карт снова повторяется процесс пробивки, проверки и т. д., пока все карты не пройдут контроль.

Такая процедура очень медленная: за смену на одном комплекте аппаратуры (один перфоратор и один контрольник) подготавливается обычно 200 перфокарт, т. е. в общей сложности 200 × × 80 символов (букв или цифр). Немаловажным отрицательным фактором является также и то, что в указанном процессе исходная информация фиксируется дважды: сначала на первичном документе, а затем уже на перфокарте.

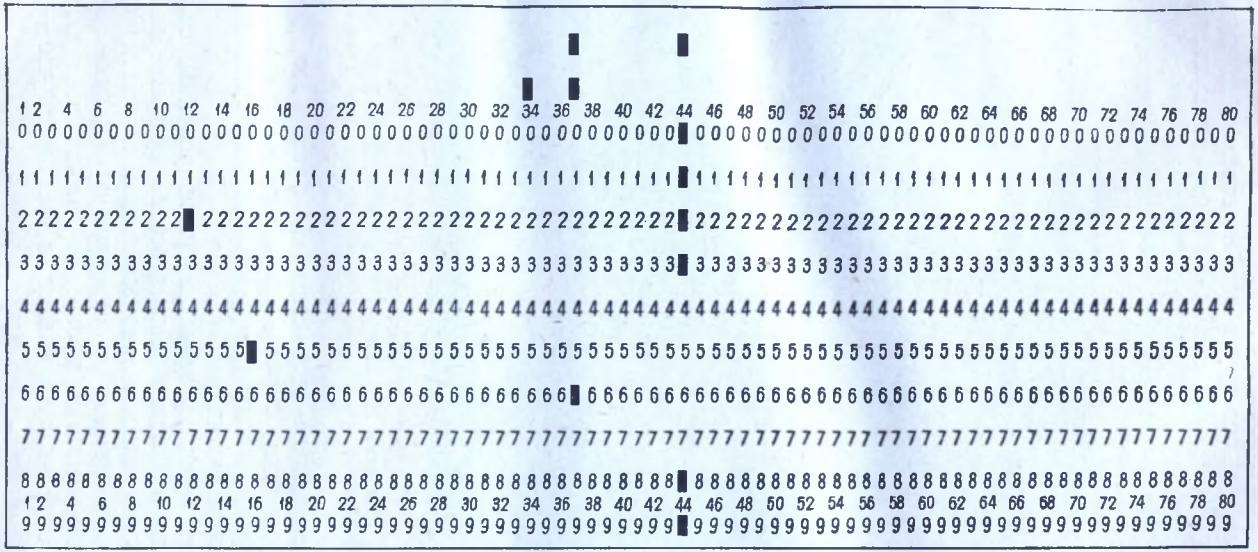

Рис. 7. Перфокарта.

Таким образом происходит удвоение (а с учетом контроля — даже утроение) ручного труда для подготовки исходных данных по сравнению с обычными системами ведения документации. Хотя возможности последующей быстрой машинной обработки обычно компенсируют указанные затраты, они, тем не менее, существенно

Рис. 8.
Внешний вид карточного перфоратора.

снижают эффективность автоматизации. Кроме того, при двойном переносе информации, сначала на первичный документ, а потом на перфокарту могут появиться дополнительные ошибки, что, разумеется, крайне нежелательно.

Поэтому при проектировании автоматизированных систем управления и систем обработки данных особое внимание должно быть уделено вопросам максимально возможной автоматизации подготовки исходных данных и совмещению процесса подготовки первичных документов с процессом изготовления их копий на машинных носителях информации (в данном случае — на перфокартах).

При использовании в качестве машинных носителей информации перфокарт наиболее радикальное решение вопроса состоит в том, чтобы сама перфокарта выступала как первичный документ. В этом случае на перфокартах заранее типографским способом печатаются формы различных первичных документов. Эти формы согласуются с принятым на перфокартах поколонковым способом записи информации. Так, например, одна из колонок в документе первичного учета для начисления зарплаты рабочему может быть отведена для указания его разряда, три другие — для указания часовой или дневной расценки и т. д. Соответствующие колонки оформляются при этом в виде табл. 1*.

Таблица 1

Разряд		1	2	3	4	5	6	[7]	8
Расценка (дневная)	руб.	1	2	3	4	[5]	6	7	8
	коп.	1	[2]	3	4	5	6	7	8
		1	2	3	[4]	5	6	7	8

Человек, осуществляющий первичный учет, должен пробить отверстия в соответствующих позициях. Например, в табл. 1 записана информация о том, что рабочий имеет 7-ой разряд, а дневная расценка для оплаты его труда равна 5 руб. 24 коп.

Таким образом, перфокарты, с одной стороны, являются исходным первичным документом, несущим информацию в форме, достаточно понятной для человека, и с другой,— служат для автоматического ввода информации в ЭВМ. Такую (размеченную) перфокарту называют иногда *дуаль-картой*.

Данные на дуаль-карты можно заносить несколькими различными способами. Наиболее простой из них состоит в том, что персонал, осуществляющий первичный учет, снабжается дешевыми ручными перфораторами-дыроколами (типа железнодорожных компостеров). Могут употребляться и более сложные цеховые или складские перфораторы с клавиатурой. При необходимости с перфоратора данные автоматически переносятся на дуаль-карту с другой перфокарты или специальных перфожетонов. В перфораторе имеются две щели: в одну вставляется заполняемая дуаль-карта, а во вторую — репродуцируемая перфокарта или перфожетон.

---

* Для удобства чтения колонки в табл. 1. расположены горизонтально.

Мастер может иметь, например, набор репродуцируемых перфокарт, содержащих информацию о сменных заданиях безотносительно к конкретному рабочему. При репродуцировании эта информация переносится на дуаль-карту, описывающую сменное задание для определенного рабочего. Необходимые данные о рабочем (учетный номер, квалификация и др.) могут быть перенесены на дуаль-карту автоматически с индивидуального перфожетона, который выдается каждому рабочему. Дополнительные сведения и комментарии могут быть занесены на дуаль-карту через клавиатуру.

Репродуцируемые карты обычно изготавливаются автоматически после расчета ЭВМ сменных заданий и других элементов плана текущей работы. Их пробивка осуществляется специальными выходными перфораторами, производительность которых составляет 100 перфокарт в минуту (133 символа в сек.). С помощью таких перфораторов можно заранее наносить постоянную информацию непосредственно на дуаль-карты, что позволяет избежать репродуцирования на цеховых перфораторах. Наконец, управляемые ЭВМ выходные перфораторы используются для быстрого приготовления дубликатов перфокарт, становящихся непригодными к дальнейшему употреблению в результате физического износа.

В тех случаях, когда хотят избежать применения перфорационного оборудования непосредственно на рабочих местах (в цехах, складах и т. п.), употребляют дуаль-карты с *карандашными отметками*. При использовании таких карт в качестве первичных документов, вместо пробивки отверстий, в соответствующих позициях карты проводят карандашные черточки. Специальное устройство, устанавливаемое в вычислительном центре, автоматически просматривает поступившие карты и пробивает отверстия в местах, где сделаны карандашные отметки. Такой метод максимально упрощает первичный учет. Однако для обеспечения надежности перфокарты должны быть высокого качества, а персонал, ведущий первичный учет,— снабжен унифицированными карандашами хорошего качества.

После того как перфокарты подготовлены (перфорированы), их вводят в ЭВМ с помощью специальных перфокартных устройств ввода. Колода карт, предназначенных для ввода, закладывается во входной карман и после включения устройства автоматически, карта за картой считывается, кодируется и передается в ОЗУ. Считанные перфокарты накапливаются в выходном кармане. После исчерпания карт во входном кармане устройство автоматически останавливается. Считывание в современных устройствах ввода осуществляется обычно фотооптическим способом. Производительность одного устройства 20—25 перфокарт в секунду, т. е. 1500—2000 символов в секунду. За счет одновременной работы нескольких устройств можно получить бо́льшие скорости ввода,

однако даже в этом случае скорость ввода значительно уступает скорости передачи информации с магнитных носителей и тем более скоростям передачи данных в ОЗУ и в операционных устройствах.

В последние годы наметилась тенденция к изменению установившихся стандартов в перфокартном оборудовании. Благодаря появлению новых материалов (органических полимеров) возникла возможность уменьшения размеров перфокарт, а следовательно, миниатюризации и соответственного уменьшения стоимости оборудования для подготовки ввода и вывода данных. Миниатюризированные перфокарты по своим размерам примерно соответствуют книжным формулярам, которыми снабжаются книги в библиотеках. Уменьшение размеров и использование новых полимерных материалов увеличивает скорость ввода данных с перфокарт.

Наряду с перфокартами для ввода данных в ЭВМ широко используется перфолента, т. е. бумажная лента с перфорируемыми в ней отверстиями. Существует два основных стандарта на перфоленты. В 8-дорожечных лентах в каждой строке (располагаемой поперек ленты) размещается восемь информационных двоичных позиций, т. е. мест, где могут быть пробиты отверстия. Пробивка отверстия означает запись в соответствующую позицию кода 1, отсутствие отверстия соответствует коду 0. Расположение строк на ленте отмечается отверстиями на специальной синхронизирующей дорожке. С их помощью устройства для записи на ленту и считывания с ленты определяют момент начала работы с каждой очередной строкой при движении ленты.

На 8-дорожечной ленте одной строкой кодируется один байт информации. При этом теряется возможность использования одного бита для контроля правильности считывания информации с ленты. Такой контроль обычно производится по правилу четности: общее количество единиц в коде всегда должно быть четным. Если код в основных разрядах содержит нечетное число единиц, то в контрольный разряд заносится 1; если число единиц в основном коде было четным, то в контрольный разряд записывается 0. Для осуществления контроля по четности необходимо добавить на перфоленту 9-ю дорожку, как это принято уже в ряде стран.

В телеграфной аппаратуре, в частности, в буквопечатающих аппаратах — телетайпах употребляется 5-дорожечная лента (с 6-й синхронизирующей дорожкой). Для представления одного байта информации в таких лентах используются две строки, поэтому для контроля и других целей остается не один, а два добавочных разряда.

Устройства ввода с перфоленты и вывода на перфоленту получаются более простыми и дешевыми, чем при использовании перфокарт. Скорость ввода информации в ЭВМ 1500 *байт/сек* и вывода — 80—150 *байт/сек*.

В отличие от перфокарты перфоленту не столь удобно использовать в качестве первичного документа. Поэтому применяется принцип совмещения подготовки исходного первичного документа и его копии на перфоленте. Для этого разработаны специальные пишущие машинки с перфоприставками (флексорайтеры). При ударе по клавише такой машинки она не только печатает соответствующий знак, но и пробивает на перфоленте его двоичный код.

Аналогичной способностью одновременной печати и записи на перфоленту обладает телетайп, который также может передавать и принимать данные по каналам связи. Передача может производиться как непосредственно в процессе набора знаков, так и с перфоленты после предварительного накопления на ней значительного обмена информации.

При передаче данных на небольшие расстояния (порядка сотен метров) используется *прямая* передача сигналов, не требующая их предварительного преобразования в специальный вид, принятый в каналах связи. С помощью прямой передачи сигналов можно связать много специальных пультов с наборной клавиатурой (так называемых регистраторов производства) со специальным пунктом концентрации информации, где осуществляется ее автоматический перенос на перфоленты. Таким образом строятся системы сбора информации из цехов завода (через цеховые регистраторы производства), из складов (через складские регистраторы) и т. п.

Среди устройств вывода наибольшее распространение получили *алфавитно-цифровые печатающие устройства* параллельного действия (АЦПУ). В отличие от обычной пишущей машинки, в которой последовательно печатается буква за буквой, в АЦПУ печатается сразу целая строка, состоящая более чем из 128—144 знаков (букв, цифр, знаков препинания, специальных символов). В механических АЦПУ такая печать выполняется вращающимся барабаном, на котором закреплено соответствующее число (128—144) алфавитных колец. Каждое кольцо содержит полный набор (до 96) символов, которые данное АЦПУ способно печатать. Скорость механических АЦПУ 20 строк в секунду (обычная скорость 10 строк в секунду), т. е. почти 3000 символов в секунду.

В электрохимических АЦПУ печать осуществляется бесконтактным способом подачей различных напряжений на систему игл, мимо которых протаскивается соответствующая электрохимическая бумага, изменяющаяся под влиянием зарядов на иглах. Скорость печати в АЦПУ подобного типа доходит до нескольких сот строк в секунду, т. е. приближается к скорости обмена информацией с магнитными лентами. Возможны и другие способы быстрой печати, например, с помощью управляемой струи краски и др.

Существуют устройства вывода, выводящие буквенно-цифровую информацию на экраны специальных катодных трубок, так

называемых *характронов*. Электронный луч в таких трубках направляется сначала на специальную маску, принимая в сечении форму вырезанного на этой маске символа, а затем уже отклоняется в требуемое место от экрана. Будучи соединены с установками быстрого фотографирования и проявления, устройства вывода на характроне способны обеспечить быстрое изготовление фотоклише для печати выведенной информации типографским способом.

Имеются системы, позволяющие проектировать изображения на специальных трубках (эйдофорах) на большие экраны. Существуют и другие возможности отображения выводимой из ЭВМ информации на больших экранах (люминесцентные, мозаичные экраны и др.).

В последние годы все большее распространение начинают получать ввод и вывод информации через магнитные носители: магнитные ленты и магнитные карты. Магнитные ленты широко употребляются для передачи информации от одной ЭВМ к другой. При накоплении данных, готовящихся вручную, магнитные ленты обладают одним существенным недостатком. Дело в том, что обычно запись на ленту производится во время ее движения. Скорость этого движения намного выше, чем скорость ввода информации человеком, сидящим за клавиатурой. Поэтому для эффективного использования ленты приходится ставить в этом случае буферное ОЗУ, которое накапливает информацию, вводимую человеком, и записывает ее на ленту целыми блоками. Устройство ввода превращается фактически в небольшую ЭВМ и стоит довольно дорого. Второй путь (не нашедший еще пока широкого применения) — разработка методов записи на неподвижную ленту с передвижением ее (на небольшую долю миллиметра) после записи каждого очередного символа.

Магнитная карта представляет собой прямоугольный кусок пластика, покрытого магнитным слоем, информация на который наносится таким же образом, как и на магнитную ленту. По внешнему виду они напоминают непробитые перфокарты. Преимуществом магнитной карты по сравнению с перфокартой является гораздо больший объем запоминаемой информации (8000 байтов, против 80 байтов для перфокарты) и гораздо бо́льшая скорость ввода, неудобством — отсутствие визуального контроля записанной информации и усложнения устройств подготовки данных.

В последние годы все больше и больше применяются устройства автоматического ввода данных с различных бланков, на которых проставляются простые стандартные метки в те или иные позиции (рис. 9) или печатаются (а иногда даже и вписываются) обычные буквы и цифры так, чтобы упростить их автоматическое считывание и распознавание специальным вводным устройством. Для записи подобных стилизованных букв и цифр (рис. 10) употребля-

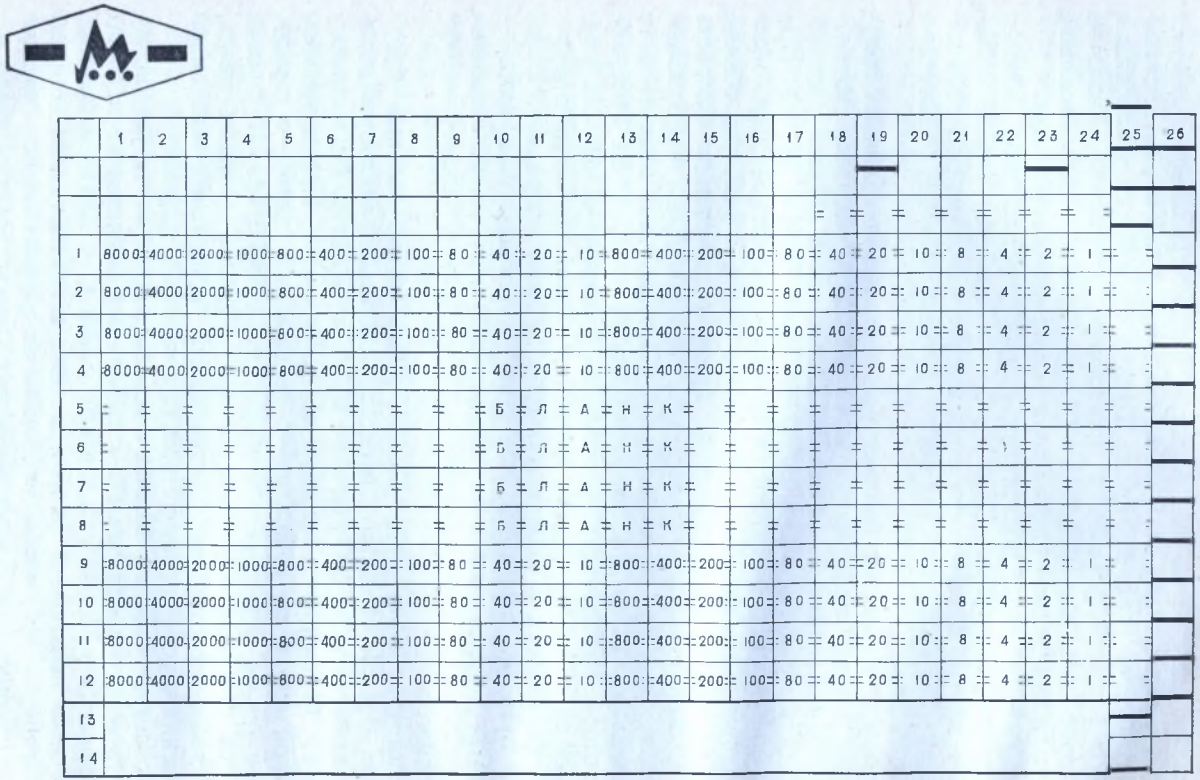

Рис. 9. Бланк дорожной ведомости.

ются специальные магнитные чернила, и информация считывается с помощью магнитных головок, как это делается в магнитных носителях информации (лентах, картах и дисках).

На рис. 11 изображено устройство, которое автоматически считывает и вводит в ЭВМ информацию с таких бланков со скоростью 2 бланка в секунду (до 2000 *байт/сек*).

Имеется возможность и прямого оптического ввода информации с обычных машинописных и даже (с заполненных разборчиво) рукописных документов. *Читающие автоматы*, способные выполнять такую работу, являются гораздо более сложными и дорогими по сравнению с устройствами, осуществляющими считывание стилизованных знаков и специальных меток. Скорость работы таких автоматов также невысока (200—300 символов в секунду).

Рис. 10.
Стилизированная запись цифр.

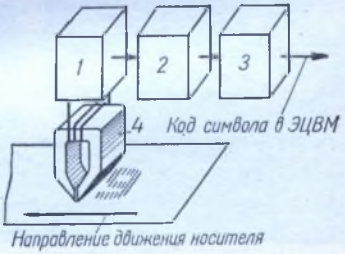

Рис. 11.
Схема считывающего устройства стилизированной записи: *1* — усилитель считывания; *2* — блок анализа временных интервалов и формирования кода символа; *3*— буферные ЗУ; *4* — считывающая головка.

Существуют и другие способы автоматизации ввода данных в ЭВМ. В торговле используются *перфоярлыки*, отрываемые в момент продажи товара или его поступления на склад (для организации системы учета). Специальные кассовые аппараты печатают на ленту не только цену, но и ее кодированное представление в виде системы черточек для последующего автоматического ввода в ЭВМ.

В автоматизированных системах управления производством все большее и большее значение приобретает непосредственный ввод данных в ЭВМ с различного рода датчиков через специальное *устройство связи с объектом* (УСО), которое опрашивает датчики и кодирует (превращает в цифровую форму) получаемую от них информацию.

Информация о дискретных процессах (например, о передвижении конвейера в следующую позицию) может передаваться в машину непосредственно. Существуют специальные штамп-датчики, регистрирующие каждый удар штампа (при простановке клейма

ОТК и др.). Причины простоя оборудования (будучи предварительно занумерованы) могут передаваться в ЭВМ с рабочих мест через специальные наборные датчики (типа телефонных вертушек).

Имеются специальные устройства для ввода в машину графической информации (чертежей и контурных рисунков) путем обвода всех линий специальным карандашом на особом планшете, а также управляемые ЭВМ графико-построители (чертежные автоматы), автоматически вычерчивающие сложные чертежи. Большие возможности для организации диалога человека с ЭВМ дают *экранные пульты (дисплеи)*. Дисплей (рис. 12) представляет собой экран типа телевизионного, на который ЭВМ может выводить любую буквенно-цифровую и графическую информацию. С помощью специального устройства, называемого *световым пером* (или *световым карандашом*), человек может рисовать на экране или стирать то, что было изображено раньше, перемещать отдельные знаки или элементы рисунка, вводить на экран новые знаки с пульта в отмеченные световым пером места и т. п.

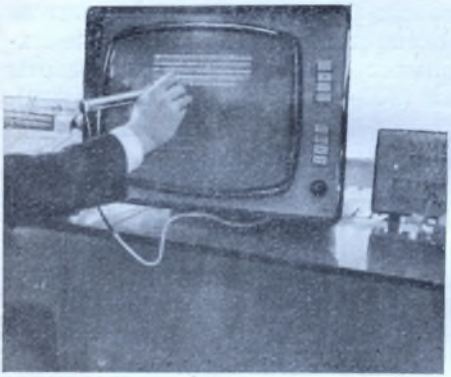

Рис. 12.
Дисплей.

4. Принципы обработки информации в центральном процессоре

Информация в центральных процессорах современных ЭВМ обрабатывается по *командно-адресному* принципу. Информация, которую необходимо обработать, и *программа* ее обработки помещаются в ОЗУ. Программа представляет собой последовательность команд (называемых также машинными инструкциями или приказами). В машинах старых выпусков команда обычно занимала одно машинное слово. В современных машинах употребляются команды различной длины (короткие и длинные), что увеличивает быстродействие и улучшает использование памяти машины.

Каждая команда состоит из двух частей: первая — код операции, которую предписывает выполнить данная команда, вторая — коды адресов машинных слов (постоянной или переменной длины), с которыми должна быть выполнена эта операция. Перед началом

работы в *устройство управления* (УУ) центрального процессора передается адрес первой команды программы, после чего УУ выполняет эту программу. Программа выполняется по отдельным частям, называемым *рабочими циклами* (тактами) центрального процессора. Каждый цикл начинается выборкой из памяти очередной команды, ее исполнения и формирования адреса следующей команды. При исполнении команды по ее адресной части находятся и передаются в операционное устройство слова, над которыми выполняются операции, фактически выполняются эти операции, а ее результаты направляются на хранение (или остаются в операционном устройстве) в соответствии с указаниями, имеющимися в адресной части команды.

Набор команд современных ЭВМ включает прежде всего арифметические операции: сложение, вычитание, умножение и деление. Обычно эти операции встречаются в нескольких модификациях: для чисел обычной и двойной длины, для чисел с фиксированной и с плавающей запятой. Часто вводится отдельно набор арифметических операций для десятичных чисел переменной длины (десятичная арифметика). В трехадресной системе каждая арифметическая команда имеет три адреса: два адреса чисел, над которыми производится операция, и адрес ячейки, куда надо направить результат операции. Покажем, как программируется в такой системе вычисление арифметического выражения, например,

$$A = \frac{a^2 - b}{c + d}.$$

Поместим: в ячейки с адресами 1, 2, 3, 4 величины a, b, c, d; ячейки 5, 6, 7, 8 отведем для хранения вычисленных величин a^2, $a^2 - b$, $c + d$, A соответственно. Предположим, что команда сложения имеет код 11, вычитания — 12, умножения — 13, деления — 14 (все операции с плавающей запятой). Тогда программа вычисления выражения записывается такой последовательностью команд: 13, 1, 1, 5; 12, 5, 2, 6; 11, 3, 4, 7; 14, 6, 7, 8. В конце этой последовательности добавляют еще специальную команду, вызывающую остановку машины (или вывод результата на печать).

Легко заметить, что, будучи записанной в указанном виде, программа теряет наглядность. Этот недостаток еще более усугубится при переводе всех чисел в двоичную систему, что является обязательным для *машинного представления* программы. Первая команда (13, 1, 1, 5) представится при этом в виде 1101000100010101, окончательно утеряв всякую наглядность.

Поэтому при записи программ обычно употребляют легко читаемые и проверяемые символические обозначения. В так называемых *мнемокодах* при сохранении *абсолютных* (десятичных) адресов вместо кода операции употребляется ее символическое

обозначение (обычно сокращенное название). При использовании мнемокода приведенная выше программа переписывается в упрощенном виде:

умножение 1, 1, 5; вычитание 5, 2, 6; сложение 3, 4, 7; деление 6, 7, 8. Более наглядно записывается программа при употреблении простого символического автокода, который вместе с символическими обозначениями для кодов операций использует символические обозначения для адресов. Еще больше запись упрощается, когда в качестве символа для обозначения адреса некоторой величины x выбирается сам символ x.

Если для выражения

$$A = \frac{a^2 - b}{c + d}$$

ввести обозначения $f = a^2$, $g = f - b$; $h = c + d$, то программа вычисления этого выражения в простом символическом автокоде представится в виде:

умножение a, a, f; вычитание f, b, g; сложение c, d, h; деление g, h, A. Современные ЭВМ снабжаются специальными *программами-трансляторами*, автоматически переводящими на *машинный язык* программы, записанные в мнемокоде или простом символическом автокоде. В последнем случае в задачу транслятора входит так называемое распределение памяти, т. е. приписывание тех или иных значений всем символическим адресам и размещение в ячейки команд самой программы. Следует иметь в виду, что в подавляющем большинстве ЭВМ команды программы должны располагаться в памяти рядом, одна за другой.

При распределении памяти возможно решение задачи *экономии памяти* за счет использования занятой ячейки памяти под новую величину, если с этого момента старая величина уже не используется. В нашем примере ячейка a может использоваться для хранения сначала величины $f = a^2$, а затем — величины $g = f - b$ и A. В ячейке c (или d) можно хранить также величину $h = c + d$. Операцию засылки в ячейку x величины y принято называть *присвоением величине x значения y* и обозначать $x := y$.

Используя это обозначение, операции по последовательному присвоению значений в рассматриваемом примере с учетом намеченной экономии памяти запишем в виде:

$$a := a \times a; \quad a := a - b; \quad c := c + d; \quad a := a : c.$$

Указанное символическое представление программы является дальнейшим расширением языка автокода в сторону так называемых *проблемно-ориентированных языков*, т. е. языков более высокого уровня. В этом представлении отсутствует явное указание такой особенности машины, как количество адресов в ее командах.

Следует отметить, что описанная выше трехадресная система команд, несмотря на свою естественность, имеет серьезные недостатки. Так, полученное в результате выполнения первой команды значение величины a^2 пересылается в 5-ю ячейку только для того, чтобы тут же быть извлеченной следующей командой. Аналогичный излишний обмен с ОЗУ происходит для величин g и h. Подобные лишние пересылки устраняются в *одноадресной системе команд*, при которой одно из чисел, с которым производится действие, и результат операции остаются в *суммирующем регистре (сумматоре)* операционного устройства. Единственный же адрес в команде определяет ячейку, из которой необходимо извлечь второе, участвующее в операции число. Помимо арифметических операций в этом случае необходимы еще операции *чтения* (передачи из ОЗУ в сумматор) и *записи* (передачи из сумматора в ОЗУ).

Программа вычисления значения величины $B = \frac{a^2 - b}{h}$ может быть записана в одноадресной системе команд:

чтение a; умножение a; вычитание b; деление h; запись B.

Если бы значение h требовалось вычислить (как это было выше), то сумматор пришлось бы освободить для этой операции, заслав его старое значение временно в ОЗУ на хранение.

Возможны и другие системы адресации, например двухадресные.

Вычисление значений арифметических выражений представляет собой простейший и наиболее легкий для программирования класс задач. Более сложными являются программы с циклами, где любая последовательность команд исполняется много раз, пока не будет выполнено некоторое условие. Для проверки условий служат специальные команды, выполняющие *операции сравнения*. В трехадресной системе такая операция сравнивает числа по 1-му и по 2-му адресу. Если первое число меньше второго, следующая команда выбирается по 3-му адресу, если — больше, то, как и обычно, следующей командой будет команда, непосредственно следующая за текущей. В одноадресной системе скачок к новой команде по этому единственному адресу совершается тогда, когда содержимое сумматора меньше или равно нулю. В противном случае сохраняется естественный порядок следования команд.

В результате выполнения команды сравнения могут возникнуть различные переходы (скачки) в программе (в зависимости от результатов сравнения), поэтому эти команды сравнения называют также командами *условного перехода*. Существуют различные вариации таких команд: переход совершается по равенству или неравенству чисел, равенству какому-либо определенному значению (чаще всего нулю), а также команды безусловного перехода, вызывающие переход к команде по указанному адресу без проверки условий.

Приведем простой пример программы, использующей команду перехода. Пусть требуется вычислить значение суммы

$$s = \sum_{k=1}^{N} \frac{1}{k} \quad (N \geqslant 1).$$

В ячейку с адресом a поместим число 1, ячейку с адресом b будем использовать для записи чисел $\frac{1}{k}$, а ячейку с адресом c — для записи чисел k; команды программы сопроводим метками вида k_1, k_2, ... В ячейку s первоначально запишем значение 0, а в ячейку N — значение числа N.

В трехадресной системе искомая программа запишется в виде: k_1: пересылка a, k,— ; k_2: деление a, k, b; k_3: сложение s, b s;, k_4: сложение k, a, k; k_5: сравнение k, N, k_2; k_6: конец.

Последняя программа демонстрирует важную особенность машинных программ, которая не имела места в ранее приведенных программах: работая по программе, ЭВМ выполняет обычно гораздо больше операций, чем число команд в программе. В рассматриваемой программе при шести командах число реально выполняемых машиной операций равняется, как нетрудно подсчитать, $4N + 2$. При бо́льших N это число много больше, чем шесть, так как в программе появляются циклы, образованные командами k_2, k_3, k_4, k_5, которые при больших N многократно повторяются.

Рассмотренный цикл используется при небольшом числе величин $\left(k, \frac{1}{k}, s_k = 1 + \frac{1}{k} + \ldots + \frac{1}{k}\right)$, значения которых все время меняются, пока одна из них (в данном случае k) не сравняется с заданной величиной (в данном случае N). Такой цикл носит название *итерационного*. Более сложными являются циклы с *переадресацией*, которые должны обрабатывать большие массивы величин, расположенных в различных (обычно смежных) ячейках. Подобный цикл возникает, например, при нахождении суммы квадратов n чисел a_1, a_2, ..., a_n (при произвольном N). Для его выполнения необходимо менять значение адресов в определенных командах программы. В принципе ничто не мешает рассматривать команды как машинные слова и производить с ними необходимые арифметические операции. Именно так и поступали в первых ЭВМ. Однако при этом программа в процессе работы изменяется, и для повторного применения ее восстанавливают. Для устранения указанного недостатка можно пользоваться адресами второго ранга или составными адресами, использующими так называемые индекс-регистры.

Индекс-регистры употребляются для хранения добавок к адресам, указываемым в командах. Соответствующий составной адрес состоит из обычного адреса и имени индекс-регистра, содержимое

которого должно быть прибавлено к указанному в команде адресу, чтобы получить истинный адрес для обращения к памяти. В состав системы команд вводятся команды пересылки информации из индекс-регистров в ОЗУ и обратно, операции прибавления (или вычитания) определенного числа (заданного в адресной части соответствующей команды) к содержимому индекс-регистра и операция условного перехода по содержимому индекс-регистров. При этом мы впервые сталкиваемся с понятием адреса *нулевого ранга*, когда в адресной части команды указывается не адрес какого-либо числа, а само это число. Имена регистров рассматриваются при этом как обычные адреса первого ранга.

Запишем программу нахождения суммы квадратов $n + 1$ чисел, расположенных в ячейках с адресами a, $a + 1$, ..., $a + n (n \geqslant 1)$. Фиксируем три рабочих ячейки b, c и s и некоторый индекс-регистр i_1, в котором, как и в ячейку s, в начале работы программы установим значение, равное нулю. В ячейку c поместим число n. Искомая программа запишется в виде:

k_1: умножение $(i_1)a$, $(i_1)a_1 b$; k_2: сложение b, s, s; k_3: сложение i_1, $1^{(0)}$, i_1; k_4: сравнение i_1, c, k_1; конец.

В команде k_3 через $1^{(0)}$ обозначена адресация числа 1 по нулевому адресу.

Помещение перед символическим адресом a имени индекс-регистра i_1 означает, что адрес модифицируется содержимым регистра i_1. Если это содержимое обозначить через $[i_1]$, то искомый истинный адрес, по которому производится обращение по команде k_1, будет $a + [i_1]$.

В современных ЭВМ употребляются еще более сложные составные адреса, истинное значение которых получается в результате сложения адреса, указанного в команде, с адресами не одного, а двух регистров.

Кроме перечисленных арифметических операций, операций управления (условных и безусловных переходов) и операций с адресами (индекс-регистрами), в набор операций ЭВМ включаются так называемые *логические операции*. Они имеют большое значение в специальных разделах программирования, позволяя, в частности, производить выделение различных частей машинных слов и компоновать слова из заданных частей. Такие операции часто применяются при обработке буквенной информации.

В машинах старых выпусков системы команд были плохо приспособлены для преобразований буквенной информации, поэтому такие преобразования представлялись в них очень сложными и малонаглядными программами. За последние годы этот недостаток начал исправляться. В американских ЭВМ серии ИБМ-360 появилась команда «переводить», по которой адресуемое слово p сравнивается по очереди со всеми словами, стоящими в левых частях

подстановок таблицы перевода: $p_1 \to q_1$, $p_2 \to q_2$, ..., $p_n \to q_n$ и, в случае совпадения ($p = p_k$), заменяется соответствующей правой частью подстановки (q_k).

В советской ЭВМ Мир-2, специально предназначенной для выполнения преобразований буквенной информации, набор операций ориентирован соответствующим образом. В частности, имеется операция «применить к формуле A систему тождеств α». Если, например,

$$A = (x - 1)(x + 1)\, 2 \sin 2x \cos 2x,$$

а система α состоит из тождеств

$$2 \sin n \cos n = \sin 2n \quad \text{и} \quad (a - b)(a + b) = a^2 - b^2,$$

то после выполнения указанной операции формула A преобразуется к виду $(x^2 - 1)\sin 4x$.

Начинают все шире применяться различного рода операции редактирования. Например, «отредактировать по шаблону разряд X, стаж XX лет». Если такая команда будет применена к ряду чисел 5, 6, 7, 12, то в результате получатся две фразы: «Разряд 5, стаж 06 лет. Разряд 7, стаж 12 лет».

Кроме перечисленных типов команд, в состав систем команд ЭВМ входят команды обмена между ОЗУ и внешней памятью, а также команды для ввода и вывода информации. В последние годы проявляется тенденция все больше и больше перекладывать задачу управления процессами обмена на специальные коммуникационные процессоры, освобождая центральный процессор для решения сложных задач обработки информации в режиме обмена с ОЗУ.

5. Стандартные подпрограммы и макрооператоры

Из простейших примеров, рассмотренных в предыдущем параграфе, видно, что программирование, даже при использовании простых символических автокодов, является достаточно кропотливым и трудоемким делом. Это объясняется тем, что при использовании обычно применяющихся систем команд программа решения задачи строится из достаточно мелких строительных элементов, т. е. отдельных машинных команд. Поэтому для облегчения программирования создаются системы более крупных программных блоков, дополняющих и расширяющих традиционные системы команд. Стандартные программы и макрооператоры как раз и представляют собой формы организации таких блоков.

Наиболее распространенные примеры стандартных программ — это программы вычисления значений простейших элементарных функций $\sin x$, $\cos x$, e^x, $\ln x$ и др. Для вычисления используются

различного рода приближенные формулы, выражающие с требуемой точностью, значения нужных функций на тех или иных интервалах. Формулы эти выбираются таким образом, чтобы в них входили лишь те операции, которые могут выполняться машиной. Поскольку обычные арифметические операции входят в системы команд всех ЭВМ, то в качестве таких формул выбирают отрезки разложений соответствующих функций в степенные ряды, например $\sin x \approx x - \frac{x^3}{6} + \frac{x^5}{120} - \ldots$ и т. п.

Полученная формула программируется и включается в *библиотеку стандартных подпрограмм* (БСП) соответствующей ЭВМ. Теперь всякий раз, когда программист встретится с необходимостью вычисления значения, например синуса, ему достаточно лишь включить соответствующую подпрограмму в составляемую им основную программу.

Такое включение может осуществляться различными способами. Простейший из них (рассчитанный на ручное, не автоматизированное программирование) состоит в том, что программист сам подставляет в команды стандартной подпрограммы необходимые адреса. Ведь составленная заранее стандартная подпрограмма не может знать, в каких ячейках программист, составляющий основную программу, решил разместить аргумент x и значение функции y, вычисляемое стандартной подпрограммой.

Этот способ очень неудобен, поскольку ручная адресация представляет собой один из самых трудоемких этапов программирования, несущих в себе потенциальные возможности ошибок. Существуют другие более удобные способы сопряжения подпрограммы с основной программой. Один из самых совершенных способов — это представление стандартных подпрограмм в виде *макрооператоров*, служащих естественным расширением системы машинных команд. При этом библиотека стандартных подпрограмм записывается в память ЭВМ (чаще всего на магнитную ленту), а каждая подпрограмма получает свое имя аналогично тому, как это делается при введении мнемонических обозначений для обычных машинных операций.

Описанный в предыдущем параграфе язык символического автокода расширяется введением в него символов новых операций — имен соответствующих стандартных подпрограмм, называемых макрооператорами.

Транслятор с такого расширенного автокода обладает способностью по имени макрооператора находить на ленте и автоматически включать в составляемую программу соответствующую стандартную программу, выполняя все необходимые сопряжения адресов. При использовании расширенного автокода, содержащего макроператоры «кв. кор.» (извлечение квадратного корня)

и «синус» (нахождение значения функции $y = \sin x$), символическая программа для вычисления значения выражения $v = a\sqrt{\sin x}$ в одноадресной машине может быть записана в виде:

чтение x; синус; кв. кор.; умножение a; запись v.

При выполнении макрооператоров «синус» и «кв. кор.» предполагается, что их аргументы перед началом вычислений находятся в сумматоре АУ и там же остаются вычисленные значения функций.

Обычно библиотека стандартных программ составляется уже для готовой ЭВМ и запоминается во внешнем ЗУ. Существуют, однако, и такие машины (например, отечественные машины серии «Мир»), где система макрооператоров проектируется вместе с ЭВМ и встраивается в быстродействующее одностороннее ЗУ. При этом удается значительно увеличить скорость выполнения сложных макрооператоров. С целью экономии памяти употребляется ступенчатая *многоуровневая* система представления операторов: операторы первого уровня — это простейшие машинные операции, второго уровня — макрооператоры, программы которых составляются из этих операций. В программах операторов третьего уровня разрешается использовать как простейшие машинные операции (операторы первого уровня), так и операторы второго уровня и т. д.

Например, если во второй уровень включены макрооператоры «синус» и «косинус», то для программирования операторов на третьем и более высоких уровнях можно использовать разложения в тригонометрические ряды (ряды Фурье).

При составлении стандартных программ необходимо определять степень их общности. Например, не имеет смысла составлять отдельно стандартные подпрограммы для (численного) решения уравнений пятой и седьмой степени. Их целесообразно заменить единой подпрограммой для решения уравнений n-й степени, где n может принимать любое значение. Лишь для $n = 2$ наличие простой формулы для решения квадратных уравнений позволяет составить значительно более эффективную программу, чем в общем случае, которую вместе с общей программой полезно включить в БСП*.

Разумеется, составить программу для численного решения уравнения произвольной степени n труднее, чем для любого фиксированного n. Однако труд на составление такой программы окупается сторицею, поскольку тем самым раз и навсегда снимается необходимость программирования любых задач, сводящихся к решению алгебраических уравнений. Достаточно обширная разумно составленная библиотека стандартных программ может

---

* Для случая $n = 1$ решение уравнения $ax = b$ сводится к машинной операции деления $x = \frac{b}{a}$ и не нуждается в специальной подпрограмме.

резко сократить время, требуемое для программирования, а иногда и совершенно устранить необходимость программирования.

Следует подчеркнуть одно существенное отличие стандартной подпрограммы для численного решения алгебраического уравнения n-й степени от стандартных подпрограмм вычисления простейших элементарных функций, рассмотренных выше, например, синуса. Эти последние подпрограммы имеют фиксированное число параметров: каждая из них получает от основной программы значение аргумента и передает обратно вычисленное значение функции. В этом смысле соответствующие макрооператоры аналогичны обычным двухадресным машинным командам. Для алгебраического уравнения n-й степени имеем $n + 2$ параметров ($n + 1$ коэффициентов и значение вычисляемого корня), т. е. число параметров переменно.

При программировании однородные наборы параметров (векторы, матрицы и т. п.) объединяются в массивы, записями в которых служат отдельные машинные слова.

Если такой массив находится в оперативном ЗУ, то при его обработке обычно приходится перебирать все его элементы один за другим. Для облегчения адресации элементов массивов в ЭВМ вводятся индекс-регистры. Указываемые в машинных командах адреса могут модифицироваться с помощью содержимого индекс-регистров. Так, если в адресе команды стоит номер какого-либо индекс-регистра, например i_1 и обычный адрес a, то обращение будет выполнено по модифицированному адресу a_1, получаемому прибавлением к адресу a содержимого индекс-регистра i_1. В систему команд вводятся операции прибавления и вычитания тех или иных величин из содержимого индекс-регистров, операции обмена между индекс-регистрами и ячейками ОЗУ, а также операции сравнения содержимого индекс-регистров с величинами, хранящимися в ОЗУ.

Используя эти операции, можно компактно записывать программы, нуждающиеся в переборе элементов различных массивов. Пусть, например, требуется вычислить скалярное произведение двух n-мерных векторов $\bar{a} = (a_1, a_2, \ldots, a_n)$ и $\bar{b} = (b_1, b_2, \ldots, b_n)$. Будем предполагать, что компоненты вектора $\bar{a}$ размещаются в последовательных ячейках ОЗУ с адресами от $A + 1$ до $A + n$, а компоненты вектора $\bar{b}$ — в ячейках с адресами от $B + 1$ до $B + n$. Через i_1 обозначим один из индекс-регистров. В ячейке с адресом S получим искомое скалярное произведение, а в ячейке с адресом P — частичные произведения $a_i \cdot b_i$ ($i = 1, 2, \ldots, n$). Наконец, в ячейку с адресом N запишем размерность векторов n. Используя мнемонические обозначения, программу, вычисляющую скалярное произведение, можно представить в виде:

засылка 0 в i_1; засылка 0 в s; z: прибавление 1 в i_1; умножение $A(i_1)$, $B(i_1)$; P; сложение s, P, s; сравнение i_1, N, z; стоп.

Предполагается, что в результате выполнения операции сравнения переход к команде с меткой z происходит всякий раз, когда содержимое регистра i_1 меньше содержимого ячейки N (т. е. меньше n). Когда эти величины сравняются, выполняется команда «стоп».

В состав библиотеки стандартных подпрограмм обычно включаются различные операции с простейшими числовыми массивами, в первую очередь с векторами и матрицами (сложение и умножение векторов и матриц, обращение матриц и др.). Для автоматизированных систем управления особое значение имеют стандартные программы для работы с большими массивами, составленными не из отдельных чисел, а из записей, имеющих более сложную форму. Рассмотрим для примера два весьма употребительных в АСУ массива, а именно плановый и нормативный массивы.

Запись в плановом массиве представляет собой совокупность двух слов — шифра изделия и количественного показателя y, определяющего выпуск этого изделия в течение заданного планового периода. Массив норм состоит из записей вида x', z_1, z_2, ..., z_m, где x' — шифр изделия; z_i — норма расхода i-го материала (или другого ресурса) на это изделие*.

Перечислим некоторые виды стандартных программ (операторов).

Оператор групповой выборки задается своим мнемокодом, например $C_{\text{в. г}}$; адресами (на магнитных лентах или дисках) начал трех массивов M_1, M_2 и M_3; названиями признаков u, v (т. е. некоторых слов в записях) в массивах M_1 и M_2 и некоторого условия $P(u, v)$, связывающего эти признаки. Оператор формирует новый массив M_3, включая в него те и только те записи Q из массива M_1, для которых в массиве M_2 найдутся записи R, так что признак u записи Q и признак v записи R удовлетворяют заданному условию $P(u, v)$.

Если взять в качестве M_1 массив норм, в качестве M_2 — плановый массив, в качестве признаков u и v — шифры изделий x' и x, а в качестве условия $P(u, v)$ — условие равенства этих шифров $x' = x$, то оператор $C_{\text{в. г}}(M_1, x'; M_2, x; x' = x; M_3)$ выделит из массива M_1 подмассив M_3 норм для всех тех и только тех изделий, которые заданы планом.

Оператор сокращения массива $C_{\text{в. с}}(M_1, x; M_2y; P(x, y); M_3)$ формирует новый массив M_3 из всех записей массива M_1, которые не удовлетворяют условию $P(x, y)$ (по признакам x и y) ни для каких записей массива M_2. Применение этого оператора к норма-

---

* На практике употребляются и более сложные формы записей в указанных массивах.

тивному и плановому массивам с условием $x' = x$ приведет к образованию нормативного массива для всех изделий, которые не включены в план. При перестановке местами массивов (M_1 — плановый, а M_2 — нормативный массив) в результате возникает перечень всех позиций плана, для которых отсутствуют нормативы.

Оператор выборки экстремальных записей $C_{\text{в. э}}$ $\left(M_1,\ x;\ \left(\frac{\max}{\min}\right);\right.$ $\left.M_p\right)$ производит выборку из массива M_1 всех тех записей, для которых признак x имеет экстремальное (максимальное или минимальное) значение. Если M_1 — массив норм, то оператор $C_{\text{в. э}}$ (M_1, z_1; max; M_p) составит массив M_p тех изделий, для которых норма расхода первого материала является наибольшей.

Оператор анализа массива $C_{\text{в. а}}$ (M_1, x_1, ..., x_n; $P(x_1, \ldots, x_n)$; M_p) формирует массив M_p из всех записей массива M_1, которые удовлетворяют условию P, и подсчитывает число таких записей. Если M_1 — массив норм, то оператор $C_{\text{в. а}}$ (M_1, z_1, z_2; $z_1 > z_2$; $M_{\text{р}}$) сформирует массив всех тех изделий, для которых норма расхода первого материала выше, чем второго.

Операторы сортировки массивов, позволяющие располагать составляющие массивы записи в порядке роста или убывания тех или иных признаков. Если M_1 — плановый массив, то оператор сортировки $C_{\text{с. у}}\{M_1,\ y,\ \rightarrow;\ M_{\text{р}}\}$ перепишет его в такой форме (массив M_p), что на первое место будет поставлено изделие с максимальным планом выпуска, а на последнее — с минимальным. Если сортировка производится не по числовому, а по алфавитно-цифровому признаку (например, шифру изделия, фамилии и т. п.), то упорядочение происходит в лексикографическом порядке; код АК17 будет предшествовать коду АК18, этот последний — коду АП19 и т. п.

Часто случается, что одно и то же значение некоторого признака (например, шифра специальности в кадровом массиве) принадлежит не одной, а многим записям. *Оператор группировки* $C_{\text{с. г}}$ $\{M_1,\ x;\ M_{\text{р}}\}$ переставляет записи таким образом, чтобы все записи с одинаковым значением такого группового признака (в данном случае x) стояли бы рядом.

Оператор слияния массивов $C_{\text{с. с}}$ (M_1, x; M_2, x; $M_{\text{р}}$) строит массив $M_{\text{р}}$, упорядоченный по признаку x из записей двух массивов M_1 и M_2, упорядоченных по тому же самому признаку.

Оператор таблиц $C_{\text{с. т}}$ (M_1, Q; $M_{\text{р}}$) осуществляет заданную перестановку Q признаков во всех записях массива M_1.

Вычислительные операторы применяют заданную программу преобразований признаков записей (включая формирование новых признаков) для всех записей по очереди.

Оператор внесения изменений Φ_n (M_1, x_k; x_1, x_2, ..., x_n; $M_2 y_k$; y_1, y_2, ..., y_n; $P(x, y)$; M_p) строит массив $M_{\text{р}}$ из записей массива

M_1, заменяя в тех записях, для которых удовлетворяется условие $P(x_k, y_k)$, признаки $x_1, x_2, \ldots, x_n$ на признаки $y_1, y_2, \ldots, y_n$. Этот оператор применяется всякий раз, когда требуется внести те или иные изменения в заранее составленный массив. Пусть, например, в плановом массиве M_1 с записями x, y нужно заменить несколько позиций: для изделия АК15 вместо ранее записанного показателя 29 шт. записать 32 шт., а для изделия БР-01 вместо 16 шт.— 15 шт. Для того чтобы произвести необходимые изменения, составляется *массив изменений* M_2 с записями вида x', y', состоящих из двух записей АК15, 32 и БР-01, 15. Оператор Φ_n $(M_1, x; y; M_2, x', y'; x = x', M_p)$ сформирует массив M_p, содержащий требуемые исправления в двух записях.

Большое значение при работе с большими массивами имеют операторы перезаписи массивов с одних машинных носителей на другие: с перфокарт на магнитные ленты, с магнитных дисков на перфоленту, с одной магнитной ленты на другую и т. п., а также операторы вывода на АЦПУ, электрифицированные пишущие машинки, телетайпы, дисплеи и другие выводные устройства. Имеются также операторы вывода с редактированием. В таких операторах для любого массива M_1 с записями вида $x_1, x_2, \ldots, x_n$ можно осуществить при выводе вставку различных служебных слов (общих для всех записей, или применяемых к отдельным записям по тому или иному условию). Например, вывод на печать планового массива M_1 с записями вида (x, y) можно осуществить специальным оператором. Вывод (N, M_1; шифр изделия x; годовой план y штук), где N — номер печатающего устройства; M_1 — имя (или адрес) массива, а остальная информация в скобках представляет собой так называемый шаблон, содержащий служебные слова «шифр изделия», «годовой план», «штук», общие для всех записей, и показатели (признаки) x и y, меняющиеся от записи к записи.

В случае необходимости требуемые служебные слова могут быть выведены в виде заголовка, под которым одна за другой будут печататься записи выводимого массива. Более сложные операторы вывода предусматривают дополнительные возможности для указания форм расположения отдельных записей и составляющих их признаков на выводимом документе, использования цветной печати, подчеркиваний тех или иных мест текста и т. п.

6. Алгоритмические языки

Описанные в предыдущих параграфах методы упрощения программирования, основанные на применении символических автокодов и библиотек стандартных подпрограмм, лишь частично решают задачу. Ведь даже самая обширная библиотека стандартных подпрограмм не может охватить всего разнообразия случаев, с ко-

торыми приходится встречаться на практике, и не может поэтому полностью устранить необходимость в программировании. С другой стороны символические автокоды остаются еще в сильной степени привязанными к конкретным особенностям ЭВМ (адресность, системы команд и т. п.). Они представляют собой пример *машинно-ориентированных языков программирования*, требующих знания этих особенностей, а следовательно, и специальной подготовки программистов.

Значительно больше упрощается программирование при употреблении *проблемно-ориентированных языков программирования*, называемых также *алгоритмическими языками*. Смысл таких языков состоит в том, чтобы описать алгоритмы (правила) решения различных задач в рамках того или иного круга проблем без использования специальных знаний об особенностях тех или иных ЭВМ. При этом стремятся по возможности использовать такие формы описания программ, которые были бы наиболее доступны и понятны человеку, знакомому с сутью решаемых проблем, а не с электронной вычислительной техникой. Такие языки называют иногда ориентированными на *пользователей* (а не на машины).

Программы, написанные на проблемно-ориентированных языках, не могут непосредственно исполняться ЭВМ и нуждаются поэтому в предварительном переводе на машинный язык. Такой перевод (транслирование) совершается автоматически с помощью специальной (как правило, очень сложной) программы, называемой *транслятором* или *компайлером*. Иногда вместо трансляции предпочитают использовать интерпретацию. При этом программа в процессе ее выполнения остается записанной в исходном (проблемно-ориентированном) языке. Специальная программа-интерпретатор просматривает кусок за куском исходную программу и формирует последовательности машинных команд, выполняющих работу.

Выгодность употребления трансляции и интерпретации зависит от различных обстоятельств, и, прежде всего, от характера использования написанной программы. Если эта программа будет использоваться в будущем многократно без всяких изменений, то, как правило, оказывается выгодным осуществить ее предварительную трансляцию. Дело в том, что интерпретация программ, написанных на проблемно-ориентированных языках происходит медленнее, чем выполнение хорошо составленных программ на машинном языке. При многократном счете получаемая за счет этого экономия вполне компенсирует затраты машинного времени, необходимые для транслирования, хотя для получения достаточно хороших машинных программ время транслирования оказывается сравнительно большим.

В случае, когда программа используется небольшое число раз или когда в нее часто вносятся различные изменения, интерпре-

тация может оказаться значительно выгоднее трансляции. Преимущества интерпретации проявляются особенно заметно, когда интерпретирующая система встраивается в конструкцию ЭВМ, а не прикладывается к ней в виде специальных программ. Такое решение принято в отечественных ЭВМ серии «Мир». В настоящее время в связи с задачей создания так называемого 4-го поколения ЭВМ это направление становится все более и более популярным. Его реализация фактически приводит к тому, что машинные языки подтягиваются на уровень проблемно-ориентированных языков, открывая тем самым новые возможности для развития последних.

Переходим к более подробной характеристике проблемно-ориентированных языков (ПО-языков). Каждый из таких языков характеризуется, прежде всего, набором *основных символов*, из которых составляются различные выражения соответствующего языка. Поскольку такие выражения приходится вводить и выводить из ЭВМ, возможности устройств ввода и вывода накладывают естественные ограничения на наборы основных символов ПО-языков. Обычно в число основных символов включаются десять десятичных цифр, буквы латинского и русского алфавитов, символы арифметических операций, знаки равенств и неравенств и различные разделительные символы (скобки, запятая, двоеточие, точка с запятой и др.). В зависимости от конкретной направленности языка в него могут включаться и другие основные символы (знак интеграла, краткие обозначения бухгалтерских понятий и т. п.).

При замене выражений в языке следует иметь в виду одну особенность машинного восприятия информации. Как известно, человеческий глаз способен к целостному восприятию двумерной информации. Благодаря этому он легко воспринимает различного рода «многоэтажные конструкции» вида e^{x^2}, y_{i_j} и др. В отличие от этого ввод информации в ЭВМ осуществляется последовательно вдоль каждой строки, символ за символом. Поэтому выражения в ПО-языках, предназначенные для ввода в ЭВМ, строятся по линейному принципу: вместо обычного a^n употребляется запись $a\uparrow n$ (Алгол-60) или $a{*}{*}n$ (Фортран-4), вместо y_{ij} пишут $y[i, j]$ (Алгол-60) и т. п.

В математике принято буквенное обозначение величин, могущих принимать различные значения. Программы для ЭВМ обычно оперируют с гораздо большим количеством таких переменных величин, чем это имело место в математических выкладках в домашинную эпоху. Букв для их обозначения, как правило, не хватает. Поэтому в ПО-языках для обозначения переменных приняты последовательности букв (обычно включая дефис) и цифр, начинающихся буквами. Такие последовательности, могущие иметь произвольную длину, обычно не превосходящую некоторую фиксированную величину, называются *идентификаторами*. Примерами идентификаторов

фикаторов могут служить последовательности {*x7*, *Pv* = 27, расчетный счет, АЛЬФА и др.}. Последний идентификатор представляет собой пример того, каким образом в язык могут быть введены величины, обозначенные первоначально символом, не включенным в язык (в данном случае греческой буквой α).

В современных ЭВМ приходится иметь дело не только с величинами, обозначенными буквенно-цифровыми последовательностями, но и с самими этими последовательностями, т. е. с величинами двух классов: обычными величинами, представляющими собой значения тех или иных идентификаторов, и так называемыми *литералами* буквенно-числовыми последовательностями, значения которых совпадают с ними самими. Как обычно принято в математике, литералами считаются все выражения, представляющие собой числа, например 2,721 или 1013. Для нечисловых последовательностей, например *Pv* = 27 или АЛЬФА, указанием того, что они должны рассматриваться как литералы, служит заключение их в кавычки. Таким образом, значением литерала «АЛЬФА» будет последовательность пяти букв АЛЬФА, а значением идентификатора АЛЬФА может служить любая величина, присвоенная ему в качестве значения (например, 3, 14).

Следующей основной характеристикой ПО-языков является набор *типов* величин, с которыми они могут оперировать. Одним из важнейших типов величин являются вещественные числа, представляемые при их реализации в конкретных ЭВМ с той или иной степенью точности. Эта точность (количество значащих цифр) определяется длиной машинного слова. Поскольку в большинстве современных ЭВМ возможны вычисления с двойной точностью, в более поздних ПО-языках (Алгол-68) вводится новый тип чисел— двойные вещественные. Отнесение тех или иных числовых величин к этому типу означает, что все вычисления с ними должны производиться с двойной точностью.

Существование наряду с параллельной двоичной более медленной последовательной десятичной арифметики порождает в ряде языков (например, в Коболе-61) новый тип величин — *десятичные вещественные*. Этот тип целесообразно присваивать тем вещественным числам, которые подвергаются небольшому числу арифметических действий. В таком случае их перевод в двоичную систему при вводе в ЭВМ и обратный перевод в десятичную систему при выводе занял бы больше времени, чем та экономия, которая могла бы быть получена за счет более быстрого выполнения операций в двоичной системе.

Во многих ПО-языках наряду с вещественными числами употребляются также целые числа, вычисления с которыми производятся без округления во всех случаях, когда в результате должны получаться также целые числа.

В ПО-языках часто применяют так называемые булевые величины, которые могут принимать лишь два значения: «истинно» или «ложно» и употребляются для обозначения тех или иных *высказываний* или *условий*. Наиболее употребительными в математике примерами условий являются равенства или неравенства. В результате проверки любого такого условия может быть получено лишь два ответа: «истинно» или «ложно».

Так называемые «строки» (Алгол-60 и другие языки) представляют собой любые отрезки текста, составленного из основных символов языка (в частности, из обычных букв). Этот тип величин является основным при обработке буквенной информации, и поэтому соответствующий тип величин часто называется просто *буквенным* (или символьным).

Многие современные ПО-языки (Алгол-68, Симула-67, *PL*-1 и др.) обладают так называемым свойством *расширяемости*. Это означает, что в языке имеются средства для определения в случае необходимости, новых типов величин (например, комплексные числа) и операций над ними.

Величины одного и того же типа (а иногда и различных типов) в ПО-языках могут организовываться в различного рода упорядоченные структуры. Простейшим видом таких структур являются *оперативные массивы* (англ. *arrays*), которые следует отличать от массивов на магнитных лентах и других внешних носителях (англ. *files*).

Оперативные массивы представляют собой совокупности величин одного и того же типа и записываются в виде $x[i_1, i_2, \ldots, i_n]$. Каждый из индексов $i_1, i_2, \ldots, i_n$ может меняться в определенных пределах (вообще говоря, различных для разных индексов). Если число различных значений индексов $i_1, i_2, \ldots, i_n$ обозначить соответственно $N_1, N_2, \ldots, N_n$, то общее число составляющих массив величин (элементов массива) будет равно произведению $N_1 \cdot N_2 \cdots N_n$. Число различных индексов (в данном случае n) называется *размерностью* массива. В частности, векторы могут рассматриваться как массивы размерности 1, матрицы — как массивы размерности 2. В памяти ЭВМ оперативные массивы располагаются в последовательных ячейках. За элементом $x[i_1, i_2, \ldots, i_{n-1}, i_n]$ обычно следует элемент $x[i_1, i_2, \ldots, i_{n-1}, i_n + 1]$. Если же i_n имеет максимально возможное значение (скажем N_n), то следующим будет элемент $x[i_1, i_2, \ldots, i_{n-1} + 1, a]$, где a — минимально возможное значение индекса i_n (обычно 1 или 0). Аналогичным образом, если i_{n-1} имеет максимальное значение, то в следующем элементе оно заменяется на минимальное, а значение индекса i_{n-2} увеличивается на 1 и т. д.

Наряду с описанным расположением массива (по строкам) может употребляться и двойственный ему порядок расположения

(по столбцам), когда вместо последнего индекса выступает первый, вместо индекса $n - 1$ — индекс 2 и т. д.

Более общим понятием, чем (оперативные) массивы, являются так называемые *списочные структуры* или попросту списки. В списочных структурах каждый элемент обычно дополняется *связкой*, указывающей месторасположение следующего за ним элемента.

Список, состоящий из простых величин, лишь их расположением в памяти отличается от обычного одномерного массива. Особенностью списочных структур является то, что часть или все элементы подобного одномерного списка могут быть, в свою очередь, заменены списками. В этих списках второго уровня некоторые элементы, в свою очередь, могут оказаться списками и т. д.

Информация, организованная в списочную структуру, аналогична книге, разбитой на главы, параграфы, пункты, подпункты и т. д. Список первого уровня составляют главы, каждая из глав, имеющая дальнейшее разбиение (на параграфы), в свою очередь, является списком. То же самое можно повторить и в отношении следующих структурных элементов книги, пока не дойдем до простых элементов, не разбиваемых далее на новые разделы и подразделы.

Обычный (оперативный) массив может рассматриваться как частный случай списочной структуры, каждая глава которой разбита на одинаковое число параграфов, каждый параграф — на одинаковое число пунктов и т. д.

Для каждого типа величин (как элементарных, так и составных) определяются операции над этими величинами. Так, для вещественных чисел в ПО-языках вводятся четыре арифметические операции. Они могут дополняться операцией возведения в степень (как правило, не только целую, но и дробную), операциями вычисления значений логарифма, показательных, тригонометрических и обратных тригонометрических функций и т. п.

Выражения, составленные из чисел и переменных (идентификаторов) с помощью этих операций, принято называть *арифметическими выражениями*. В ряде ПО-языков (Алгол-60 и др.) в состав арифметических выражений вводятся условия. Например, выражение, если $x \leqslant 10$, то $10x$ иначе x^2 задает функцию, равную $10x$ для $x \leqslant 10$ и равную x^2 для $x > 10$.

Обычно для арифметических операций употребляются принятые в математике сокращенные символические обозначения. В некоторых языках (Кобол-61) наряду с такими обозначениями употребляются и словесные выражения, например «вычесть x из y». Слова «вычесть» и «из» принадлежат к числу так называемых *служебных слов* соответствующего ПО-языка, которые, разумеется, должны быть известны транслятору (для перевода их в нужные куски

машинной программы). Эти слова не следует использовать в качестве идентификаторов).

Для чисел вводятся также элементарные *операции отношения*: $=$, $\neq$, $>$, $\geqslant$, $<$, $\leqslant$, позволяющие конструировать элементарные условия вида $A \geqslant B$, $A = B$ и т. д., где A и B — произвольные арифметические выражения.

В простейшем случае эти арифметические выражения могут сводиться к числам и переменным, приводя к условиям вида: $x \geqslant 10$ и т. д. Элементарные условия представляют собой простейший вид так называемых булевых выражений или условий. Значениями булевых выражений являются величины типа *булевы*. Для таких величин (а следовательно, и для булевых выражений) в большинстве ПО-языков разрешается пользоваться элементарными логическими операциями $\wedge$ *(И)*, $\vee$ *(ИЛИ)*, $\neg$ *(НЕ)* и др. Эти операции позволяют из простейших булевых выражений вида $A = B$, $C > D$ и др. строить сложные выражения. Например, выражение $(x \geqslant 10) \wedge \neg (x = 20)$ означает условие, истинное тогда и только тогда, когда одновременно удовлетворяются элементарные условия $x \geqslant 10$ и $x \neq 20$.

В современных ПО-языках допускается расширение системы операций для величин различных типов, включая массивы и списки. Такое расширение оказывается возможным как в результате определения новых операций средствами самого языка, так и с помощью использования библиотеки стандартных подпрограмм.

Одним из основных понятий ПО-языков является понятие *оператора*, представляющее собой обобщение понятия машинной команды. К числу наиболее употребительных операторов относится *оператор присваивания*, встречающийся в различных модификациях во всех ПО-языках. Смысл его заключается в том, что в результате его выполнения некоторая величина (задаваемая своим идентификатором) приобретает новое значение, получаемое в результате вычисления какого-нибудь выражения. В принципе оператор присваивания может быть задан знаком равенства: $x = A$, где x — идентификатор, а A — выражение, значение которого имеет тот же смысл, что и величина x. Однако знак равенства имеет в ПО-языках различный смысл. Возможны случаи, когда идентификатору x присваивается значение, скажем, $x + 1$, т. е., попросту говоря, значение переменной x увеличивается на единицу. Выражение $x = x + 1$, что называется, «режет глаз». Поэтому обычно для обозначения оператора присваивания предпочитают употреблять другой символ, например, символ: $:=$ (Алгол-60).

Запись $x := a x (b + c)$ означает, что величине x (имеющей тип вещественной или целой величины) присваивается значение, получаемое в результате вычисления арифметического выражения $a x (b + c)$. После трансляции величине x и величинам a, b, c сопо-

ставляется некоторая ячейка ОЗУ. Таким образом, рассматриваемый оператор как бы уподобляется 4-адресной машинной команде. Разумеется, при трансляции в машинный язык этот оператор должен быть представлен в виде последовательности реальных машинных команд соответствующей адресности.

Важное значение для ПО-языков имеют условные операторы, позволяющие применять те или иные операторы в зависимости от проверки условий. Одна из возможных форм условного оператора может иметь вид (Алгол-60): если B то A иначе C. Здесь B — условие (булево выражение), проверяемое в начале выполнения условного оператора. Если это условие окажется истинным, то выполняется оператор A, в противном же случае — оператор C.

Для осуществления переходов (скачков) в программе операторы, к которым возможны такие переходы, снабжаются специальными идентификаторами — так называемыми метками. Метка ставится обычно перед отмечаемым ею оператором, отделяясь от него каким-либо разделительным знаком, например двоеточием. Последовательные операторы в программе отделяются друг от друга другим знаком, например точкой с запятой (Алгол-60). Последовательность четырех операторов A, B, C, D, из которых третий отмечен меткой $x1$ в Алголе-60 запишется A; B; $x1$: C; D.

Переходы в программе, нарушающие естественный порядок следования операторов, совершаются обычно с помощью оператора безусловного перехода вида: «идти на Λ», где Λ — имя (идентификатор) соответствующей метки.

Для описания различного рода операций, выполняемых для элементов массивов, большое значение имеют операторы цикла. Существенной составной частью оператора цикла является описание совокупности значений индексов элементов массивов (или набора элементов списка), для которых должен быть выполнен тот или иной оператор или последовательность операторов. Если такое описание обозначить Г, а оператор, который необходимо выполнить над всеми элементами массива P, то общий вид оператора цикла может быть записан в виде: «для Г выполнить P». Если оператор P составной, т. е. представляет собою последовательность операторов A_1; A_2; ...; A_n, то его необходимо заключить в скобки. Для избежания двусмысленностей из-за употребления скобок в различных смыслах при выделении составных операторов используются специальные операторные скобки. В языке Алгол-60, например, для этой цели употребляется пара слов — «начало» и «конец». Эти слова, а также слова «если то», «иначе», «идти на», «для», «выполнить» относятся к числу служебных слов соответствующего языка, значение которых должно быть известно транслятору и которые не должны использоваться ни для каких иных целей.

Описание совокупности значений индексов (Г) в операторе цикла может быть выполнено несколькими различными способами. Простейший из них — простое перечисление всех требуемых значений индексов. При одном индексе i оператор цикла с подобным перечислением может, например, быть представлен в такой форме: «для $i = 1, 3, 6, 10$ выполнить начало A_1; A_2; ...; A_n конец».

Очень часто значения индексов изменяются в заданных пределах с определенным шагом. В этом случае перечисление Г удобно записывать в виде: «для $i = B$ шаг C до D выполнить P». В простейшем случае в качестве B, C, D берутся целые числа. Оператор «для $i = 0$, шаг 2 до 10 выполнить P» означает, например, что оператор P будет выполнен для всех элементов, у которых индекс i принимает значения 0, 2, 4, 6, 8, 10. В общем случае в качестве B, C и D могут выступать арифметические выражения (с целочисленными значениями), так что, например, шаг C может изменяться при каждом очередном прохождении через цикл (применении оператора P для очередного элемента массива).

Окончание оператора цикла может происходить не только тогда, когда соответствующий индекс i достигнет своей верхней границы D, но и в том случае, когда в первый раз станет истинным некоторое булево выражение β. В этом случае описание Г приобретает вид: «для $i = B$ шаг C до тех пор пока β». При этом само собой предполагается, что в выражение β входят величины, меняющиеся от цикла к циклу.

В развитых ПО-языках обычно бывает возможным комбинировать различные формы построения описания Г. Например, если Г имеет вид «для $i = 1$ шаг 2 до 7, 10, 14; $i = 16$ шаг 3 до 22», то описываемая совокупность значений индекса i есть 1, 3, 5, 7, 10, 14, 16, 19, 22.

В принципе возможно определение операторов цикла с произвольным количеством индексов. На практике, однако, чаще всего ограничиваются одним индексом, употребляя для нескольких индексов соответствующее количество вложенных друг в друга операторов цикла. Например, программа «для $i = 1$ шаг 1 до 5 выполнить начало для $j = 1$ шаг 1 до 6 выполнить P конец», состоящая из двух вложенных друг в друга операторов цикла, означает, что оператор P будет выполнен для всех элементов вида $x[i, j]$, где индекс i пробегает значения от 1 до 5, а индекс j — значения от 1 до 6.

Описанных операторов оказывается достаточно для построения произвольных программ. Их применение делает простым и естественным (по сравнению с машинно-ориентированными языками) описание всех вычислительных процессов. С помощью этого набора операторов можно осуществлять также любые преобразования буквенной информации. Однако в этом случае программы получа-

ются мелкоблочными и трудночитаемыми, поскольку отсутствует возможность использовать арифметические выражения.

Положение может быть исправлено путем введения достаточно большого набора стандартных подпрограмм для преобразования буквенной информации. В ПО-языках для использования стандартных подпрограмм вводится оператор процедуры. Каждая подпрограмма (процедура) получает свое собственное имя (идентификатор) и описывается с помощью этого имени и наборов своих входных и выходных параметров (в число которых могут входить массивы и списки). Оператор процедуры (или, точнее, оператор вызова процедуры) состоит из имени соответствующей процедуры и списка *фактических параметров*, которыми должны быть замещены формальные параметры, содержащиеся в описании процедуры.

Например, пусть G есть идентификатор процедуры, вычисляющий корень квадратный из суммы квадратов: $z = \sqrt{x^2 + y^2}$. В этой процедуре имеются три формальных параметра: два входных (x и y) и один выходной (z). Описание состоит из заголовка $G(x, y, z)$ и так называемого тела процедуры, т. е. программы в проблемно-ориентированном или машинном языке, задающей необходимую последовательность действий для получения z из x и y. В стандартных подпрограммах соответствующие описания находятся в библиотеке СП, если же программист хочет определить некоторую процедуру для использования только в своей программе, он должен приложить ее к основной программе и ввести в ЭВМ до начала работы программы.

В основной программе одна и та же процедура может использоваться многократно при различных значениях фактических параметров. Ее включение в программу на ПО-языке осуществляется с помощью оператора вызова процедуры. Например, в отрезке программы A; B; $G(a, b, c)$; C; $G(3, 4, d)$; D дважды используется описанная выше процедура G: один раз для вычисления по формуле $c = \sqrt{a^2 + b^2}$, а второй раз — $d = \sqrt{3^2 + 4^2}$. Значения параметров a и b и смысл идентификаторов c и d задаются основной программой.

Перечисленными операторами не исчерпываются все виды операторов, встречающиеся в различных ПО-языках. Однако они дают достаточно полное представление об основных идеях, закладываемых в конструкцию операторов. Поэтому мы ограничимся приведенным списком операторов и перейдем к вопросу о представлении программ в ПО-языках.

В практике современного программирования часто приходится иметь дело со сложными программами. Даже будучи записанными в ПО-языке такие программы могут содержать многие тысячи или даже десятки тысяч операторов. Для того чтобы уменьшить

число ошибок при написании программы и дать возможность работать над составлением программы нескольким программистам одновременно, в ПО-языках предусматриваются средства для расчленения программы на удобно обозримые части.

Одним из таких средств является уже описанный метод выделения из основной программы подпрограмм, которые могут составляться затем независимо от нее. В большинстве современных ПО-языков предусматривается возможность расчленения на отдельные части также и основной программы. Такие части называются блоками. Впервые достаточно четко понятие блока было определено в ПО-языке Алгол-60.

Каждый алгольский блок начинается операторной скобкой «начало» и заканчивается операторной скобкой «конец». Непосредственно за словом «начало» идет описание величин, с которыми оперирует данный блок. После описания следует собственно программа, т. е. последовательность операторов (с метками или без них), заключаемая словом «конец». Важно иметь в виду, что в число операторов программы могут входить не только описанные выше элементарные операторы, но и блоки, которые можно рассматривать как составные операторы. Таким образом, могут возникать сложные конструкции вложенных друг в друга блоков*. В число операторов программы могут входить пустые операторы, обычно снабжаемые метками, но не производящие никаких действий. Такой оператор удобно использовать, например, в конце блока, когда из середины блока возможен переход на его конец с помощью оператора «идти на».

Описания в Алголе-60 содержат сведения о типах величин, встречающихся в данном блоке (если они не описываются в охватывающих его блоках), а также о диапазонах изменения индексов в массивах. Например, описание «вещественные x, y, a; целые k, n; вещественный массив A [1 : p, 1 : q]» означает, что в блоке вводятся три вещественные величины x, y, a, две целые величины k и n и двумерный массив, состоящий из вещественных чисел A [i, j], где индекс i может принимать значения от 1 до p, а индекс j — от 1 до q.

Величины, описанные в данном блоке, локализованы в нем. Это означает, что вне блока любой описанный в нем идентификатор может употребляться для обозначения совсем других величин. Это обстоятельство дает возможность при программировании разных блоков пользоваться одними и теми же обозначениями без опасности возникновения ошибки. При работе (уже транслированной в машинный язык) программы в ОЗУ могут вызываться отдельные блоки, а ячейки памяти, необходимые для хранения локализован-

---

* В некоторых ПО-языках (Симула-67) блоки могут быть организованы в произвольные списочные структуры.

ных в этих блоках величин, могут отводиться динамически, т. е. лишь на время работы этих блоков. Тем самым создается возможность выполнения очень больших программ с большим числом данных в ЭВМ, имеющих относительно небольшие объемы ОЗУ.

Для возможности подобного динамического распределения памяти необходимо обеспечить возможность пользования описаниями. Поскольку они помещаются в начале блока, то входить в блок (в результате выполнения оператора перехода) разрешается только через его начало. Иными словами, метки всех операторов блока локализованы в этом блоке и закрыты для пользования извне блока. Вместе с тем выход из блока на любой помеченный оператор в охватывающих его блоках возможен не только из конца блока, но и из любого другого места, в которое помещен оператор «идти на».

Для примера опишем алгольский блок, вычисляющий скалярное произведение двух векторов $s = x_1y_1 + x_2y_2 + \ldots + x_ny_n$.

Начало вещественные s, p; вещественные массивы $x[1 : n]$, $y[1 : n]$; $s := 0$; для $i = 1$ шаг 1 до n выполнить начало $p := x[i] \times y[i]$; $s := s + p$ конец конец.

В ПО-языках, ориентированных на обработку экономической и учетной информации, употребляются более сложные и детализованные способы описания данных. Так, в языке Кобол-61 вводится 49 уровней данных, которые могут использоваться для описания сложных иерархических структур данных. Более мелким подразбиениям данных приписываются более высокие номера уровней. Например, структура записей списка адресов некоторого круга лиц может быть описана в Коболе-61 следующим образом:

```
02 порядковый номер
02 адрес
   04 город
   04 улица
   04 номер дома
   04 номер квартиры
   04 наименование адресата
      07 фамилия
      07 имя
      07 отчество
```

Это описание означает, что каждая запись состоит из двух частей, имеющих одинаковый уровень (в данном случае 02), а именно — из порядкового номера записи и адреса. Адрес подразбивается на наименование города, улицы, адресата, номера дома и квартиры, а наименование адресата разделяется, в свою очередь, на фамилию, имя и отчество.

Для каждого вида данных может быть описан их формат, т. е. максимальное количество составляющих их символов и тип этих символов (цифры или буквы).

Если какие-то данные (константы) не входят в более широкое поле данных, то для них употребляется особый дополнительный уровень — 77. Наконец, еще один дополнительный уровень (—88) употребляется для описания значений, которые могут принимать переменные качественного характера. Например, описание:

```
12 семейное положение
88 семейный
88 одинокий
```

означает, что переменная «семейное положение» (получившая уровень 12) может принимать два значения: «семейный» и «одинокий». Эти два литерала относятся к особому, 88-му уровню.

Во многих современных ПО-языках имеются еще большие возможности для описания сложных структур данных, в том числе и возможность введения кратких обозначений для часто употребляющихся структур.

7. Операционные системы

Алгоритмические языки и трансляторы не решают всех проблем, связанных с задачей эффективного использования ЭВМ и упрощения программирования. Очень сложными и специфическими вопросами, требующими решения, являются: организация первичного ввода (загрузки) программ и данных, осуществление передачи данных между ЗУ разных уровней, хранение и поиск стандартных программ и данных во внешней памяти и др. Все задачи решаются специальными системами программ, называемыми *операционными системами*.

Операционные системы подверглись быстрому изменению в связи с совершенствованием ЭВМ. Опишем основные идеи построения операционных систем для третьего поколения ЭВМ (отечественные машины серии «Ряд», американские *IBM* система 360, английские *ICL* система 4 и др.).

Для понимания работы указанных операционных систем необходимо иметь в виду одну особенность ЭВМ третьего поколения, а именно — возможность одновременной работы нескольких программ,— *мультипрограммирование*.

Разумеется, если машина имеет лишь один центральный процессор, то обработка данных в нем в каждый момент может проходить под управлением только одной программы. Однако исполнение программы, отнюдь, не сводится лишь к работе центрального процессора. Много времени уходит, например, на поиск и передачу данных из внешних ЗУ в оперативное ЗУ. Вначале ЭВМ строились таким образом, что все эти операции выполнялись последовательно, одна за другой. В результате центральный процессор

простаивал, ожидая, пока не будет закончена передача данных в ОЗУ или из ОЗУ.

В ЭВМ третьего поколения обмен данными между ОЗУ и внешней памятью, а также между ОЗУ и устройствами ввода-вывода проходит под управлением специальных *коммуникационных процессоров* — так называемых каналов. Получив команду от центрального процессора «начать передачу данных», канал выполняет все остальные операции самостоятельно. В это время программа, выдавшая эту команду, переходит в состояние ожидания, уступая центральный процессор другим программам. После окончания передачи данных канал посылает сигнал, уведомляющий центральный процессор о выполнении задания. Этот сигнал прерывает исполнение текущей программы и передает управление специальной программе операционной системы, называемой *диспетчером*. Диспетчер решает, какую из программ следует продолжать выполнять.

Описанный процесс позволяет запараллелить работу центрального процессора с работой периферийного оборудования и тем самым поднять производительность ЭВМ. Обычно ЭВМ снабжаются несколькими каналами, часть из которых *(селекторные каналы)* специализируются на обслуживании высокопроизводительных периферийных устройств (магнитных лент, барабанов и дисков). При работе такой канал полностью занимается одним устройством. Переключение его на другое устройство происходит лишь после окончания работы с первым устройством. Медленные устройства (в основном устройства ввода и вывода) подключаются к *мультиплексным каналам*, которые успевают работать одновременно с несколькими устройствами.

Все периферийные устройства (индивидуально или группами) снабжаются своими собственными устройствами управления, которые осуществляют сопряжение обслуживаемых ими устройств с каналами и освобождают каналы от управления выполнением некоторых операций. Например, каналу достаточно лишь передать начальную команду на обратную перемотку ленты или поиск нужной зоны на ленте, после чего он может отключиться для обслуживания других устройств, пока не получит сигнал (от устройства управления группой лент), что поданная команда выполнена.

Таким образом, особенностью ЭВМ третьего поколения является наличие специальной *системы прерывания*, позволяющей по сигналам, формируемым одними устройствами, останавливать и возобновлять работу других устройств. В систему прерывания поступают также сигналы от системы контроля, реагирующей на ошибки в передаче сигналов в различных частях машины, а также на ошибки в программе, приводящие к невыполнимым операциям (например, делению на нуль или попытке обращения по несуществующему в данном комплекте ЭВМ адресу).

Одновременное исполнение нескольких программ вызывает необходимость в защите памяти. Защита эта заключается в том, что при попытке одной программы вторгнуться в зоны памяти, отведенные другим программам, также формируется сигнал прерывания, и операционной системе предоставляется право решать, следует допустить такое вторжение или нет.

При мультипрограммировании естественным режимом использования ЭВМ является *пакетная обработка информации*, т. е. одновременная загрузка и выполнение не одной единственной программы, а целого пакета программ. Обеспечение эффективного управления такой обработкой представляет собой одну из основных задач операционной системы. Предполагается, что выполняемые программы могут использовать обширные массивы данных. Задача операционной системы состоит в том, чтобы организовать эффективное взаимодействие программ с этими массивами.

Современные операционные системы проводят четкое различие между задачами и заданиями. Задача состоит в работе некоторой программы с определенным комплектом данных. Та же программа, работающая с другим комплектом данных, представляет собою уже другую задачу. Задание есть краткое описание последовательности работы одной или нескольких программ с одним или многими комплектами данных. Для описания задания каждая программа и каждый набор данных получают свое имя. Понятие набора данных является несколько более общим, чем понятие массива. Помимо обычных массивов оно может включать в себя программы на разных стадиях обработки (до и после трансляции), библиотеку стандартных программ и т. п.

Задание начинается с общих сведений: название задания, время его поступления, имя программиста, приоритет задания и т. п. После заголовка следуют пункты задания, каждый из которых порождает одну задачу. В первом подпункте каждого пункта указывается имя программы, которую надо выполнить, и вид носителя (магнитная лента, перфокарты и др.), на которых хранится эта программа. Во втором подпункте перечисляются имена наборов данных, с которыми должна работать эта программа, и виды носителей, на которых они хранятся.

В очень упрощенной форме один из примеров задания выглядит следующим образом:

Задание: расчет зарплаты 16.III.1971 г., 15 *час* 30 *мин*
программист Иванов, приоритет 10
Пункт 1. Программа A, магнитный диск, данные B перфокарты, C магнитная лента
Пункт 2. Программа C, магнитная лента, данные P_1 перфолента
Пункт 3. Программа C, магнитная лента, данные P_2 перфокарты

Здесь через *A* обозначено имя транслятора с ПО-языка, на котором записана исходная программа *B*. Первый пункт задания требует, чтобы была проведена трансляция исходной программы *B*, вводимой с перфокарт, а результат трансляции (машинная программа *C*) был записан на магнитную ленту. Во втором пункте программа *C* обрабатывает массив данных P_1, вводимый с перфолент, а в третьем — массив P_2, записанный на перфокартах. P_1 и P_2 могут представлять данные о выработке работающих в двух разных цехах или двух разных предприятиях.

Любопытно отметить, что в первом пункте задания *C* выступает как имя набора данных, а в следующих пунктах эти данные рассматриваются как программа.

Сформировав пакет заданий на том или ином машинном носителе (например, на перфокартах), системный оператор (человек, ответственный за организацию работы ЭВМ) загружает этот пакет в соответствующее входное устройство, сообщая с пульта управления системой номер этого устройства специальной программе операционной системы так называемому планировщику.

Планировщик осуществляет ввод пакета заданий, определяет порядок их выполнения (в соответствии с их приоритетами и возможностями вычислительной системы) и инициирует выполнение пунктов заданий с учетом имеющихся ресурсов.

Ресурсы системы, о которых идет здесь речь — это, прежде всего, состав, работоспособность и занятость периферийного оборудования, с которого должен быть осуществлен ввод программ и данных. Планировщик ведет постоянно строгий учет интересующих его ресурсов (таблицу исправности и занятости периферийного оборудования). Через системный пульт (снабженный пишущей машинкой) оператор сообщает планировщику о том, что такое-то устройство отключается на ремонт или профилактику, такое-то устройство вновь включается в систему и т. п.

Расшифровкой и передачей данных, сообщаемых оператором системе и системой оператору, занимается специальная программа операционной системы, называемая коммуникатором. При достаточно мощном коммуникаторе разговор (через пишущую машинку) оператора с системой может происходить на языке, близком к естественным человеческим языкам.

Для инициирования очередного пункта задания планировщик осуществляет «прикрепление» программы и наборов данных к соответствующим периферийным устройствам, выбирая их из числа исправных и незанятых. Данные о таком прикреплении сообщаются оператору и главной программе операционной системы *супервизору*. Задача системного оператора состоит в том, чтобы загрузить соответствующие носители (перфокарты, перфоленты, магнитные

ленты, кипы дисков) в указанные планировщиком периферийные устройства, если они не были установлены там заранее*.

Если планировщик ответственен за управление заданиями, то супервизор несет ответственность за управление *процессом решения задач*. Основной ресурс, которым управляет супервизор — это оперативная память. Специальная подпрограмма, называемая *менеджером ресурсов* ведет постоянный учет занятости ОЗУ. Следить в отдельности за каждой ячейкой ОЗУ оказывается практически невозможным. Поэтому оперативная память разбивается на участки — *страницы*. Выделение памяти различным задачам и ее перераспределение производится целыми страницами. Обычный размер страниц — 512 машинных слов, хотя, разумеется, возможны и другие размеры.

Супервизор осуществляет загрузку программ в ОЗУ. Крупные программы разбиваются на отдельные части — *модули загрузки*, которые взаимодействуют между собой с помощью специальных макрооператоров операционной системы. Каждый модуль получает свое собственное имя и может фактически рассматриваться как отдельная программа.

Загрузка программ осуществляется помодульно. Поскольку на практике обычно бывает трудно выдержать стандартный размер модулей, каждому модулю предпосылается информация о его длине. Используя эту информацию, супервизор отводит в ОЗУ необходимое число страниц и загружает в них соответствующий модуль. В начало модуля выносится также информация об объеме оперативной памяти, необходимой для данных, с которыми оперирует этот модуль. Требуемое число страниц из наличных ресурсов также отводится супервизором.

Может случиться, что вызываемый на счет модуль не помещается на свободных страницах ОЗУ. В подобных конфликтных ситуациях супервизор производит вытеснение из ОЗУ во внешнюю память части старой информации в соответствии с приоритетами задач и историей использования этой информации**.

Поскольку значительная часть сведений о необходимом объеме ОЗУ должна готовиться в процессе трансляции машинных программ, трансляторы с различных ПО-языков должны быть сопряжены с операционной системой. Поэтому удобно считать, что трансляторы составляют просто одну из частей операционной системы. Следует заметить, что современные операционные системы с помощью специальных программ, называемых *редакторами*

---

* Трансляторы и другие служебные программы обычно постоянно находятся во внешней памяти ЭВМ.

** Обычно вытесняется в первую очередь информация, которая перед этим дольше всего не использовалась.

связей, обеспечивают «сшивание» в единое целое кусков программы, написанных на различных алгоритмических языках и транслированных различными трансляторами.

Как уже отмечалось выше, при исполнении программ связь между отдельными модулями загрузки осуществляется специальными макрокомандами. Эти макрокоманды по форме напоминают оператор «идти на» в ПО-языках. Вместо метки в них указывается имя следующего вызываемого модуля. Главная трудность при реализации подобных макрокоманд состоит в том, чтобы по имени модуля найти его фактическое расположение во внешней памяти. Методы такого поиска будут описаны ниже. Существует несколько разновидностей указанных макрокоманд перехода.

Кроме простого перехода часто необходимо иметь возможность после выполнения вызванного модуля снова вернуться на исходный модуль и продолжать работу, начиная с того места, на котором совершился переход. В этом случае предварительно выполняется макрокоманда «сохранить», осуществляющая запоминание состояния центрального процессора на момент перехода. Вызываемый модуль заканчивается макрокомандой «вернуться», по которой восстанавливается состояние центрального процессора, а управление передается исходному модулю.

Модули, которые необходимо сохранить в ОЗУ для повторного использования, вызываются специальной макрокомандой «загрузить *A*», где *A* — имя модуля. Модуль остается в ОЗУ, пока по ходу выполнения программы не встретится макрокоманда «выгрузить *A*», после чего менеджер ресурсов может занимать отведенные этому модулю страницы. Существуют и другие модификации макрокоманд перехода, обеспечивающие возможность параллельного выполнения вызванного и исходного модулей.

Одной из наиболее трудоемких задач, решаемых операционной системой, является поиск наборов данных (включая программные модули) по их именам. Для решения этой задачи в системе создается постоянно обновляемый каталог, в котором каждому набору данных, хранимом в системе, сопоставляется номер тома, содержащего этот набор или (для особо больших наборов) его начало. Под томом здесь понимается тот или иной машинный носитель: бобина магнитной ленты, сменная кипа дисков или пространство на несменяемых дисках, обслуживаемое одним и тем же механизмом выборки.

Для ускорения поиска в каталоге наборы данных снабжаются составными именами вида *A*. *B*. *C*. ... , где, например, *A* — наименование завода; *B* — номер цеха; *C* — название заказа и т. п. Поиск при этом организуется ступенчатым образом: по первому простому имени (в данном случае *A*) происходит выбор раздела каталога, в котором помещены все имена, начинающиеся с этого имени, по второму имени происходит выбор подраздела и т. д.

Тем самым исключается необходимость перебора всех позиций каталога и ускоряется поиск.

Поиск по каталогу организует супервизор по специальной макрокоманде «открыть набор». Этой макрокомандой, помещаемой в нужных местах программы, программист обеспечивает подготовку передачи необходимых ему данных в оперативную память. Операционная система ведет специальную таблицу установки томов, в которой для каждого периферийного устройства указывается номер установленного на нем тома. Получив команду «открыть набор», супервизор по каталогу находит номер тома, содержащего требуемый набор, и по таблице установки томов определяет установлен ли требуемый том на одном из устройств или нет.

Если данный том еще не установлен, то супервизор находит свободное периферийное устройство требуемого типа и через коммуникатор посылает сообщение системному оператору с указанием номера тома и номера устройства, на который его нужно установить. Предполагается, что свободные тома хранятся в упорядоченном архиве (фильмотеке и дискотеке) на строго закрепленных за ними местах, так что поиск оператором нужного тома происходит достаточно быстро. Если свободного периферийного устройства в данный момент нет, то задача, вызвавшая набор, переводится в режим ожидания до момента, пока требуемое устройство освободится.

После установки требуемого тома на устройство супервизор обращается *к таблице содержания тома* (помещаемой обычно в начале тома). В этой таблице указывается точное месторасположение требуемого набора в томе. Далее супервизор (через канал) выдает команду соответствующему устройству управления на поиск требуемого места. После установки тома в нужное положение выдается сигнал о готовности к передаче запрашиваемого набора данных в ОЗУ.

Если программисту известно точное расположение запрашиваемого набора, то операционная система представляет ему возможность сформировать самому требуемые команды и тем самым ускорить поиск.

Следующим важным моментом является выбор и передача в ОЗУ необходимой информации из заданного набора. При небольших наборах он может быть переписан в ОЗУ целиком. Однако такое положение является скорее исключением, чем правилом. Обычно приходится иметь дело с большими массивами и организовывать их обработку по отдельным записям. Записи могут иметь свои имена, либо различаться по номерам.

Операционная система предоставляет программисту специальную макрокоманду «получить запись». Эта макрокоманда по имени (или порядковому номеру) запрашиваемой записи находит ее

в предварительно открытом наборе и передает в заранее отведенное для нее место в ОЗУ, где она и подвергается обработке запросившей ее программой.

Программисту нет необходимости в этом случае описывать в своей программе весь массив, он ограничивается описанием элементов максимальной записи этого массива. По этому описанию транслятор формирует требование к числу ячеек ОЗУ, необходимых для размещения записи*, а супервизор при вызове программы выделяет эти ячейки. Таким образом, при наличии операционной системы программирование на ПО-языках может вестись так, как если бы все данные, необходимые для работы программы, содержались в ОЗУ. Все заботы о вводе и выводе информации из ОЗУ в периферийные устройства и обратно берет на себя операционная система.

Для осуществления такого обмена операционная система резервирует специальное пространство в ОЗУ для организации так называемых буферов. Буфер представляет собой совокупность последовательных ячеек ОЗУ, число которых выбирается таким образом, чтобы в буфере мог разместиться один блок информации, считанный с дисков или магнитной ленты.

Необходимость записи информации блоками обусловливается особенностями работы дисков и лент. Будучи механическими устройствами, они обладают определенной инерцией, так что при попытке прочитать одно слово все равно под считывающей головкой пройдет достаточно большая зона, на которой размещается заведомо больше одного слова. Если записывать и считывать лишь по одному слову, то прийдется либо оставлять между словами большие пробелы (достаточные для торможения и разгона носителя), либо после работы с очередным словом осуществлять обратное движение носителя для нахождения следующего слова. В обоих случаях снижается производительность внешних ЗУ (а в первом и их емкость). Поэтому, чтобы достигнуть высокой производительности внешних ЗУ, обмен данными между ними и ОЗУ организуется блоками. Рациональный размер блока определяется характеристиками ЗУ и обычно колеблется от 128 до 2048 машинных слов. Соответственно выбираются и размеры буферов.

При запросе первой записи из предварительно открытого набора супервизор организует передачу содержащего эту запись блока (или нескольких блоков) в какие-либо из свободных буферов, а затем передает требуемую запись в рабочую зону программы для обработки. При запросе следующих записей супервизор предварительно проверяет, не находится ли она уже в одном из буферов, и лишь

* Не исключен, разумеется, случай, когда программист может организовать вызов в ОЗУ нескольких записей одновременно с помощью повторения макрокоманды «получить запись».

при отрицательном ответе организует передачу необходимого блока в один из свободных буферов.

Как правило, в результате работы программы из обрабатываемых ею записей входных массивов формируются записи выходных массивов. С помощью специальной макрокоманды «отправить запись» программист сообщает операционной системе о необходимости переместить готовые для вывода записи из рабочей зоны в один из буферов, где из готовых записей формируется блок. По мере окончания формирования готовые блоки переписываются из ОЗУ во внешнюю память, где предварительно должен быть открыт соответствующий выходной массив. Операционная система ведет постоянный учет состояния буферов (какие из них заполнены, какие загружаются, какие разгружаются и какие свободны) и соответствующим образом оперирует имеющимися в ее распоряжении буферными фондами.

Программисту при желании предоставляется возможность самому организовать буферные фонды и работу с ними, включая выделение и передачу требуемых записей.

Обычно в заголовке массива указывается характер составляющих его записей. В случае записей постоянной длины выделение записей супервизором происходит в результате отсчета необходимого числа слов или байтов. В случае записей переменной длины длина каждой записи должна быть указана в ее начале, либо должны быть применены специальные метки для отделения одной записи от другой.

Когда набор не помещается в одном томе, в конце тома помещается ссылка на номер тома, в котором помещено продолжение набора. При попытке прочитать запись, выходящую за пределы текущего тома, операционная система прерывает выполнение текущей программы и организует (через коммуникатор и системный оператор) поиск и установку тома, содержащего продолжение открытого набора.

После окончания работы с открытым набором программист с помощью специальной макрокоманды «закрыть набор» сообщает операционной системе о том, что устройство, оперировавшее с этим набором, освободилось и может быть использовано для работы с другими наборами.

Если необходимо обеспечить секретность информации, содержащейся в тех или иных наборах, то эти наборы, кроме имени, снабжаются также *паролями*. Операционная система обеспечивает невозможность открытия таких массивов простым указанием их имен. Все обращения к таким наборам регистрируются. Если пароль набора, который пытается открыть программа, не указан в самой программе, то операционная система передает имя запрашиваемого массива на терминал пользователя (который должен

быть указан в программе). Открытие набора произойдет лишь в том случае, когда с этого терминала будет набран действующий в данное время пароль, которым этот набор защищен. Область памяти (обычно на магнитных дисках), где размещается таблица паролей, защищается *главным паролем*, известным лишь очень узкому кругу лиц.

Помимо обычного обращения к наборам данных по их именам, развитые операционные системы могут обеспечивать также образование *поколений данных*, которые используются в тех случаях, когда в архиве (фильмотеке или дискотеке) имеются однотипные наборы данных, отличающихся только временем их поступления. Например, можно говорить о поколениях ведомостей на зарплату по данному цеху, поколениях месячных или квартальных планов и т. д.

Для таких наборов устанавливается предельный срок хранения и их порядковые номера, начиная от самого старого набора, хранящегося в системе. При обращении к наборам указывается их имя и порядковый номер. Возможно также указание отрицательного номера: принимая номер последнего имеющегося набора за нуль, присваиваем относительный номер —1 предпоследнему набору и т. д. (—2, —3, —4, — ...). Операционная система следит за сроком давности хранения наборов и при достижении предельного срока устраняет соответствующие наборы из архива, для чего формируется соответствующее уведомление системному оператору.

Из сказанного ясно, что современные операционные системы рассчитываются на ведение сложного информационно-архивного хозяйства. Это обстоятельство требует наличия и постоянного использования различного рода служебных (сервисных) программ. К их числу относятся программы перезаписи наборов с одного носителя на другой, программы, обеспечивающие внесение необходимых изменений в каталог и таблицу паролей, программы, реализующие операторы для работы с массивами и т. п. Все это программное хозяйство вместе с трансляторами, редакторами связей и различными программными средствами для отладок рабочих программ обычно также включаются в понятие операционной системы.

Помимо пакетной обработки информации, операционные системы многих ЭВМ третьего поколения обеспечивают и другие режимы работы. В последние годы все большее значение начинает приобретать использование ЭВМ в *режиме разделения времени*. Этот режим предполагает наличие многих терминалов, т. е. устройств для двусторонней связи с ЭВМ*. Пользователи, за которыми закреплены эти терминалы, вводят через них в ЭВМ свои задания или

---

* В качестве терминалов могут использоваться телетайпы, электрифицированные пишущие машинки, экранные пульты и др.

задачи, а операционная система отпускает элементарные порции времени (обычно тысячные доли секунды) для работы с каждой задачей. Для этой цели используются встроенные в ЭВМ электронные часы, способные измерить очень малые интервалы времени.

Поскольку современные быстродействующие ЭВМ даже за эти малые промежутки времени выполняют десятки тысяч операций, то малые задачи (типа разных вычислений по формулам, занимающим несколько страниц текста) выполняются за один такой промежуток. Задачи средней и большой сложности требуют, разумеется, многих элементарных интервалов. Как правило, разделение времени оказывается эффективным именно для мелких задач. При достаточно большом быстродействии ЭВМ у каждого пользователя при решении мелких задач создается ощущение, что он один полностью занимает машину, хотя, на самом деле, число одновременно работающих абонентов может составлять несколько десятков.

В ряде случаев может оказаться целесообразным смешанный режим использования ЭВМ: пакетная обработка больших задач вместе с разделением времени для мелких задач.

Так называемые операционные системы *реального времени* отличаются той особенностью, что данные поступают в ЭВМ не через обычные устройства ввода, а от реальной аппаратуры через соответствующую систему датчиков и преобразователей. Они должны обрабатываться немедленно в момент поступления. Поэтому, в противоположность обычным операционным системам, здесь не программы организуют вызов необходимых им данных (хотя это и не исключено), а поступающие данные вызывают нужные для их обработки программы.

Операционные системы вместе с трансляторами, набором сервисных программ и библиотекой основных стандартных подпрограмм принято называть общим (или общесистемным) математическим обеспечением ЭВМ. Кроме общего существует и специальное математическое обеспечение, включающее в себя операционные системы, трансляторы и библиотеки стандартных подпрограмм, нацеленные на более эффективное решение некоторых специальных классов задач (автоматизация проектирования, автоматизированные системы управления и т. п.).

8. Перспективы развития ЭВМ

Историю ЭВМ в настоящее время принято делить на поколения. Пятидесятые годы были временем господства первого поколения ЭВМ. Эти машины строились на электронных лампах, имели небольшой объем оперативной памяти и малый ассортимент периферийных устройств. Возможности параллельной работы устройств, системы прерывания и мультипрограммирования не существовало.

Автоматизация программирования ограничивалась, в основном, мнемокодами и символическими автокодами. Имелись лишь отдельные идеи современных транслирующих и операционных систем.

Начиная с 1959—1960 гг. ламповые машины начинают быстро вытесняться машинами второго поколения, строившихся на полупроводниковой технике. Замена ламп полупроводниками увеличила скорость работы центральных процессоров, значительно повысила надежность их работы, уменьшила габариты и энергопотребление. Одновременно значительно улучшились и другие характеристики ЭВМ, прежде всего объем и скорость работы ОЗУ, увеличился ассортимент и характеристики периферийного оборудования. В мировой практике широкое распространение получают магнитные диски. Возникают различные ПО-языки, разрабатываются транслирующие системы.

В ЭВМ второго поколения появляются отдельные элементы организации параллельной работы устройств, производятся эксперименты по мультипрограммированию и разделению времени, создаются достаточно удовлетворительные, хотя еще и не развитые во всех деталях, операционные системы.

В 1965—1967 гг. ряд фирм (в частности, наиболее крупная американская фирма IBM) организует выпуск ЭВМ третьего поколения, постепенно вытесняющих в последующие годы машины второго поколения.

ЭВМ третьего поколения строятся на базе *интегральных схем*. До их появления каждый логический элемент ЭВМ, служащий для запоминания и преобразования одного бита информации, состоял из нескольких физических элементов (полупроводниковых диодов и триодов, сопротивлений, емкостей и др.).В противоположность этому, интегральная схема, являясь одним физическим элементом, заключает в себе несколько логических.

Особенностью ЭВМ третьего поколения является модульный принцип построения и автономность работы отдельных устройств. Благодаря этому оказалось возможным построить развитые операционные системы, обеспечивающие достаточно эффективное использование всех возможностей ЭВМ. В структуру ЭВМ были введены коммуникационные процессоры и стандартизация сопряжения (или, как обычно принято говорить, *интерфейса*) ЭВМ с периферийным оборудованием. Это открыло новые возможности для совершенствования периферийных устройств и расширения их ассортимента. Большое значение приобрела аппаратура сопряжения ЭВМ через коммуникационные процессоры с каналами связи, использующая обычные телеграфные и телефонные каналы для работы ЭВМ с удаленным периферийным оборудованием, например терминалы, обеспечивающие возможность «диалога» пользователя с находящейся от него на большом расстоянии ЭВМ. Фактически, лишь

в ЭВМ третьего поколения развернулись возможности мультипрограммирования и был сделан существенный шаг в развитии систем разделения времени.

ЭВМ первого и второго поколения специализировались либо на решении научных задач, либо на обработке экономической (учетной) информации. В первом случае использовались двоичные арифметические, во втором — последовательные двоично-десятичные (побайтовые) устройства. В ЭВМ третьего поколения впервые произошло объединение этих двух классов машин.

Это обстоятельство сказалось и на развитии ПО-языков. Языки, предназначавшиеся для использования в ЭВМ первого и второго поколений (Фортран, Алгол-60, Кобол-61 и др.), достаточно четко подразделялись на языки для научных задач и для описания обработки деловой информации. В новом поколении ПО-языков (Алгол-68, *PL*-1 и др.) фактически произошло слияние этих двух классов языков.

В ЭВМ третьего поколения резко улучшились технические характеристики: быстродействие центрального процессора, объем оперативной и внешней памяти, скорость обмена с периферийным оборудованием и ОЗУ. Уменьшились габариты электронного оборудования и увеличилась надежность его работы. Наконец, с введением автоматизации проектирования и изготовления интегральных схем резко снизилась стоимость центральных процессоров и другого электронного оборудования ЭВМ.

Степень интеграции (число логических элементов в одном физическом элементе) для интегральных схем, применяющихся в ЭВМ третьего поколения, относительно невелика (обычно в пределах одного десятка). Развитие электронной техники резко увеличивает степень интеграции (до нескольких сотен и даже тысяч). Подобные *большие интегральные схемы* (называемые кратко БИСами), или, точнее, схемы большой интеграции — это шаг к дальнейшей микроминиатюризации ЭВМ и улучшению их технических характеристик.

Большинство ученых сходятся в том, что БИСы будут представлять собой конструктивную основу для построения ЭВМ следующего, *четвертого поколения*.

В настоящее время прогнозируются и другие возможности ЭВМ четвертого поколения*. Прежде всего, эти ЭВМ будут *мультипроцессорными*. Используя микроминиатюризацию, в тех же габаритах, в которых ранее размещался один процессор, станет возможным разместить многие десятки и даже сотни процессоров.

Наряду с понятием мультипрограммной обработки информации возникает противоположное ему понятие *мультипроцессорной об-*

---

* Многие из этих черт уже проверены экспериментально и опубликованы.

аботки. В первом случае один процессор используется для реше-ия многих задач, во втором — одна задача (разные ее части) ешается несколькими процессорами. Возможности такого «рас-араллеливания» процесса решения имеются у многих математичес-их задач. Разумеется, подобный прием должен использоваться основном при решении особо сложных задач. В случае малых средних задач каждый процессор мультипроцессорной машины южет быть занят своими задачами.

Мультипроцессорная обработка накладывает новые требования а операционные системы и трансляторы. ПО-языки должны пре-оставлять программисту широкие возможности для распаралле-ивания задач, а трансляторы и операционные системы должны обладать способностью эффективно использовать эти возможности. Средства для описания параллельно текущих процессов имеются ряда уже существующих ПО-языков (Алгол-68, *PL*-1 и др.). Эти средства в будущем будут, по-видимому, значительно расширены.

Новые функции, возлагаемые на трансляторы и операционные системы, увеличат их сложности. В то же время уже сейчас их сложность такова, что доля общего математического обеспечения в общей стоимости ЭВМ превысила 50%. Поэтому в машинах чет-вертого поколения значительная часть задач, решаемых ныне с по-мощью математического обеспечения, должна быть возложена на оборудование.

Имеется три основных направления решения этой задачи. Во-первых, это изменение понятия о системе команд ЭВМ с целью подтягивания понятия команды к понятию оператора в развитых ПО-языках. Во-вторых, это расширение системы команд за счет макрооператоров, выполняющих функции, которые ныне выполня-ются многими стандартными подпрограммами. Наконец, третье направление состоит в разработке аппаратурных, схемных реше-ний для обработки прерываний и выполнении макрооператоров (или их составных частей) операционных систем.

Первые две задачи (еще на элементной базе машин второго по-коления) решены в отечественных ЭВМ серии «Мир». Их дальнейшее развитие в ЭВМ четвертого поколения может привести к специали-зации процессоров для выполнения тех или иных функций (сложные вычисления, обработка буквенной информации, справочно-инфор-мационные задачи и др.). Решение третьей задачи может привести к возникновению понятия *управляющего процессора*, который возьмет на себя многие функции управляющих программ нынешних операционных систем.

Приближение структуры ЭВМ к ПО-языкам и требованиям эффективного управления процессами переработки информации создает предпосылки для стандартизации сопряжения математи-ческого обеспечения с ЭВМ различных типов. Иными словами,

речь идет о том, чтобы математическое обеспечение, разработанное для одних машин, могло достаточно эффективно использоваться в других.

Это может быть достигнуто, например, путем стандартизации основного ядра систем команд и способов организации систем прерывания таким образом, чтобы особенности надстройки над этим ядром могли бы быть описаны средствами лишь этого ядра. В таком случае вместо разработки конкретных программ математического обеспечения могут разрабатываться *генераторы* таких программ, которые после сообщения им описания особенностей данной конкретной машины могли бы генерировать конкретные программы.

Надо думать, что дальнейшее развитие получат также коммуникационные процессоры (каналы). Помимо относительно простой функции организации обмена информацией между ОЗУ и периферийными устройствами, каналы в ЭВМ четвертого поколения возьмут на себя более сложные функции. В частности, на коммуникационные процессоры смогут замыкаться макрооператоры сортировки и группировки массивов, программы поиска (подборки) информации по тем или иным принципам и другие программы, работающие с внешней памятью без больших вычислений.

С появлением машин четвертого поколения войдет в практику понятие *периферийного процессора*. Дело в том, что режим разделения времени при работе с большим числом мелких задач может привести к неэффективной работе центральных процессоров в связи с необходимостью частных переключений и потерь времени на работу программ операционной системы. Лучшими возможностями в этом отношении обладают иерархические системы, в которых пульты пользователей замыкаются на периферийные процессоры относительно небольшой производительности. Рядовые запросы пользователей удовлетворяются этими процессорами, и лишь при необходимости решать сложные задачи периферийные процессоры пользуются помощью центрального процессора.

Особое место (в первую очередь в связи с задачами автоматизации проектирования) займут вопросы, связанные с обработкой графической (двумерной и трехмерной) информации. Для этой цели, по-видимому, будут разработаны специальные периферийные и центральные процессоры. В связи с увеличением общей доли задач, которые будут решаться в режиме диалога человек—машина, большое место займут вопросы упрощения общения человека с машиной. Помимо устройств для ввода и вывода графической информации получат распространение также устройства для ввода и вывода информации голосом*.

* В настоящее время имеются экспериментальные устройства для ввода в ЭВМ голосом относительно небольшого числа слов. Устройства вывода голосом употребляются пока в ограниченных масштабах.

Ожидается, что скорость работы ЭВМ четвертого поколения будет измеряться сотнями миллионов и даже миллиардами операций в секунду. Объем оперативной памяти превысит миллиард (10^9) байтов, а объем внешней памяти 10^{12}—10^{13} *байт*. Гораздо большее значение, чем сейчас, приобретут вопросы сопряжения ЭВМ с каналами связи. Если сейчас каналы связи употребляются в основном для дистанционного ввода и вывода информации из ЭВМ, то для ЭВМ четвертого поколения важнейшее значение приобретет вопрос использования широкополосных каналов связи для организации совместной работы сложных комплексов территориально удаленных друг от друга машин. Начнется процесс органического слияния систем ЭВМ с единой общегосударственной системой связи.

Глава II

МАТЕМАТИЧЕСКИЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ПЛАНИРОВАНИЯ И УПРАВЛЕНИЯ

1. Общие сведения о задачах оптимизации

При решении задач управления на электронных вычислительных машинах особое место занимают вопросы оптимизации. Процесс оптимизации предполагает выбор такого варианта управления, при котором достигается минимальное или максимальное значение некоторого критерия, характеризующего качество управления. Выбор надлежащего критерия является основной проблемой правильной постановки задачи оптимизации. Этот выбор зависит от общих целей, которые ставятся перед управлением. В одном случае целью может служить достижение максимального объема выпуска продукции или получение максимальной прибыли, в другом — минимизация себестоимости или выбор такого плана, при котором та или иная поставленная заранее цель достигается за минимальное время, и т. д.

В реально встречающихся на практике задачах управления имеют дело не с одним критерием, а с многими. Причем, зачастую эти критерии противоречивы: улучшение управления по одному критерию ухудшает значения других и наоборот. Так, стремление к увеличению объема выпуска продукции может привести к ухудшению ее качества, достижение цели в кратчайшие сроки — к дополнительным расходам и т. п. В то же время математически правильная постановка задачи оптимизации предполагает наличие лишь одного критерия. Выход из положения достигается тремя основными путями. Первый и наиболее часто применяемый способ состоит в том, что для оптимизации выбирается один из критериев, а для других устанавливаются некоторые пороговые значения. Например, ставится задача: получить максимальную прибыль, при объеме производства по определенной номенклатуре не меньше планового. В этом случае ставят и решают не одну, а несколько задач оптимизации. Так, если имеется n критериев A_1, A_2, ..., A_n, то вначале оптимизация производится по параметру A_1 (с заданием пороговых значений для остальных критериев), затем по параметру A_2 и т. д.

Второй способ состоит в построении смешанного критерия, представляющего собою некоторую функцию от первичных параметров. В большинстве встречающихся на практике случаев эта функция является линейной: $f(A_1, A_2, \ldots, A_n) = \alpha_1 A_1 + \alpha_2 A_2 + \ldots + \alpha_n A_n$. Выбор весовых коэффициентов $\alpha_1, \alpha_2, \ldots, \alpha_n$ (они могут быть как положительными, так и отрицательными) производится в соответствии с качественным представлением постановщика задачи об относительной важности различных критериев. Как и в предыдущем случае, при этом способе решают не одну, а несколько задач оптимизации при различных выборах функций f или в линейном случае при различных значениях весовых коэффициентов.

Третий способ заключается в изменении масштаба постановки задачи. Так при оптимизации управления относительно мелкой экономической системой, например, отдельным заводом, возникает необходимость рассмотрения многих противоречивых критериев. Если расширить масштаб постановки задачи, взяв в качестве объекта целую отрасль промышленности или даже все народное хозяйство, то эти частные критерии можно вывести из меньшего числа глобальных критериев. В масштабе всего народного хозяйства можно в конечном счете свести дело к одному глобальному критерию, а именно к времени, требуемому для достижения достаточно детализованной цели по объему и составу конечного продукта на определенный момент в будущем.

Классическая математическая постановка задачи оптимизации состоит в том, чтобы найти экстремальное (минимальное или максимальное) значение некоторой функции $f(x_1, x_2, \ldots, x_n)$ в определенной области S значений параметров $x_1, x_2, \ldots, x_n$. Область S обычно задается системой неравенств вида

$$g(x_1, x_2, \ldots, x_n) \geqslant 0 \text{ или } g(x_1, x_2, \ldots, x_n) \leqslant 0.$$

Точка $M(y_1, y_2, \ldots, y_n)$ тогда и только тогда принадлежит области S, когда ее координаты удовлетворяют всем неравенствам, задающим эту область. Если при этом имеют место строгие неравенства, то точка оказывается внутри области. В случае же, когда хотя бы одно из неравенств обращается в равенство, точка попадает на границу области. При этом предполагается, что задающие границы области функции g непрерывны.

Если функция f также непрерывна (что обычно бывает на практике), то, как доказывается в математическом анализе, в конечной области S найдутся точки M_1 и M_2, в которых эта функция принимает наибольшее и наименьшее значение*. Если экстремум (максимум или минимум) достигается во внутренней точке области, а функция f дифференцируема, то в точке экстремума все частные производные

* Таких точек может оказаться несколько и даже бесконечно много.

$\frac{\partial f}{\partial x_i}$ $(i = 1, 2, \ldots, n)$ обращаются в нуль*. На этом основан классический метод нахождения экстремумов: находятся точки M $(y_1, y_2, \ldots, y_n)$, координаты которых удовлетворяют системе уравнений $\frac{df}{dx_i} = 0$ $(i = 1, 2, \ldots, n)$ и которые лежат внутри области S.

Следует напомнить, что условие обращения в нуль частных производных является необходимым, но недостаточным условием экстремума. Поэтому все найденные точки подвергаются дополнительной проверке. На практике это чаще всего делается вычислением значений функции f во всех найденных точках и нахождения среди них наибольшего и наименьшего значения. В случае, когда заранее известно, что экстремальные значения достигаются функцией f внутри области, указанная процедура приводит к локальному решению задачи. К сожалению, однако, в реальных задачах управления такое положение является скорее исключением, чем правилом: наиболее часто экстремальные значения достигаются именно на границах области.

В таком случае возникает задача на условный экстремум: найти точки M $(x_1, x_2, \ldots, x_n)$ экстремума функции f при условии, что они удовлетворяют одному или нескольким граничным уравнениям

$$g_1(x_1, x_2, \ldots, x_n) = 0; \ \ldots; \ g_m(x_1, x_2, \ldots, x_n) = 0.$$

Одним из методов решения задач на условные экстремумы является метод множителей Лагранжа. Суть его состоит в том, что функция f заменяется функцией (Лагранжа)

$$F = f + \lambda_1 g_1 + \lambda_2 g_2 + \cdots + \lambda_m g_m,$$

где $\lambda_1, \lambda_2, \ldots, \lambda_m$ — некоторые неизвестные пока числовые множители Лагранжа.

Далее, как обычно, решается система уравнений $\frac{dF}{\partial x_i} = 0$ $(i = 1, 2, \ldots, n)$ совместно с системой граничных уравнений. Если при этом будет получен ряд решений $x = (x_1, \ldots, x_n)$, $\lambda = (\lambda_1, \ldots, \lambda_m)$, то снова можно вычислить значения функции f во всех найденных точках $x = (x_1, \ldots, x_n)$ и определить наибольшее или наименьшее значение.

В тех случаях, когда получается только одно решение, требуется дополнительная проверка (с использованием вторых производных): является ли это решение точкой локального минимума, максимума или точкой перегиба. В связи с тем, что перебор множества решений системы уравнений практически не реализуемая операция, метод множителей Лагранжа не получил

* Для граничных точек это, в общем случае, неверно.

широкого распространения при решении сложных практических задач.

Покажем применение описанного метода на простейших примерах.

Пример 1. Найти максимальное и минимальное значение функции $f = x + y$ в области S, заданной неравенствами $x^2 + y^2 \leqslant 1$ и $x + y \geqslant 1$. Границы области представляют окружность $x^2 + y^2 = 1$ и прямую $x + y = 1$ (рис. 13).

Нетрудно видеть, что если граница области задается уравнением $g(x_1, x_2, \ldots, x_n) = 0$, то точки, лежащие по одну сторону границы, будут удовлетворять неравенству $g > 0$, а по другую — неравенству $g < 0$. В нашем случае неравенство $x^2 + y^2 \leqslant 1$ задает точки внутри окружности и на самой окружности (горизонтальная штриховка). Неравенство $x + y \geqslant 1$ задает точки, расположенные вправо от прямой, включая и саму прямую (вертикальная штриховка).

Для проверки этих утверждений достаточно рассмотреть по одной точке в каждой из указанных областей, например, точку O (0, 0) внутри окружности и точку D (2, 0), расположенную вправо от прямой:

$$0^2 + 0^2 < 1; \quad 2 + 0 > 1.$$

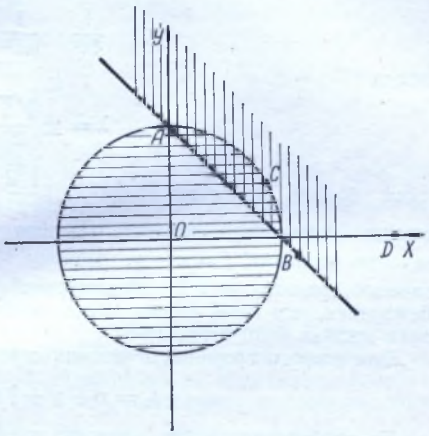

Рис. 13.
Область минимизации.

Ввиду непрерывности функций $x^2 + y^2$ и $x + y$, тем же неравенствам будут удовлетворять и все точки, в которые можно попасть из точек O и D без пересечения соответствующих границ.

Искомая область, в которой определяется экстремум, представляет собой сегмент ABC. Решение первой экстремальной задачи (для внутренних точек области) приводит к системе двух уравнений

$$\frac{\partial f}{\partial x} = 0 \quad \text{и} \quad \frac{\partial f}{\partial y} = 0.$$

Так как в нашем случае

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y} = 1,$$

то система приводится к виду $1 = 0$; $1 = 0$, указывающему на то, что данная система *несовместна*, т. е. не имеет никаких решений. Таким образом, точки экстремума внутри области отсутствуют.

Далее, решаем две экстремальные задачи с одним граничным условием. Для окружности по методу множителей Лагранжа находим

$$f_1 = x + y + \lambda (x^2 + y^2 - 1).$$

Система уравнений $\frac{\partial f_1}{\partial x} = 0$ и $\frac{\partial f_1}{\partial y} = 0$ после взятия частных производных запишется в виде:

$$1 + 2\lambda x = 0; \quad 1 + 2\lambda y = 0.$$

Ее решение представляется в виде точки с координатами $x = y = -\frac{1}{2\lambda}$. Подставляя эти координаты в уравнение окружности, получаем соотношение

$$\frac{1}{4\lambda^2} + \frac{1}{4\lambda^2} = 1.$$

Отсюда

$$\lambda = \pm \frac{\sqrt{2}}{2}.$$

Таким образом, имеем две точки $M_1\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$ и $M_2\left(-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right)$, подлежащие дальнейшей проверке. Первая точка соответствует максимальному значению функции $f = x + y$, равному $+\sqrt{2}$, а вторая — ее минимальному значению, равному $-\sqrt{2}$. Подстановкой во второе неравенство $x + y \geqslant 1$ убеждаемся, что лишь первая точка принадлежит заданной области S, вторая точка должна быть отброшена.

Для второго граничного условия строим функцию

$$f_2 = x + y + \lambda (x + y - 1).$$

Ее дифференцирование приводит к системе $1 + \lambda = 0$; $1 + \lambda = 0$, которая при выборе $\lambda = -1$ обращается в тождество. Полученный результат означает, что условия существования экстремума удовлетворяются для всех точек границы. В самом деле, поскольку на границе $x + y = 1$, то значение функции для всех граничных точек постоянно и равно 1. Для рассматриваемого граничного условия это будет как максимумом, так и минимумом заданной функции f.

Следующая экстремальная задача состоит в рассмотрении обоих граничных условий и, соответственно, функции

$$f_3 = x + y + \lambda (x^2 + y^2 - 1) + \mu (x + y - 1).$$

Нетрудно, однако, понять, что обоим граничным условиям удовлетворяют лишь две точки $A(0, 1)$ и $B(1, 0)$ пересечения окружности и прямой. Достаточно поэтому ограничиться вычислением значения функции f в обеих этих точках. К тому же нет необходимости и в этой последней проверке, поскольку обе точки A и B лежат на границе, для которой уже вычислено значение функции f во всех ее точках.

Окончательный вывод: функция f достигает максимума, равного $\sqrt{2}$ в точке $M\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$, лежащей на окружности $x^2 + y^2 = 1$, и минимума, равного 1, в любой точке отрезка AB прямой $x + y = 1$.

Пример 2. Для той же области S, что и в примере 1, найти максимальное и минимальное значения функции $h = x - y$.

Как и в предыдущем случае, внутренних точек экстремума не имеется *.

* Это имеет место для любой не равной тождественно константе линейной функции h в любой замкнутой области S.

Для отыскания условного экстремума на окружности дифференцируем функцию

$$h_1 = x - y + \lambda (x^2 + y^2 - 1),$$

в результате чего приходим к уравнениям

$$1 + 2\lambda x = 0 \quad \text{и} \quad -1 + 2\lambda y = 0,$$

откуда

$$x = -\frac{1}{2\lambda}, \quad y = \frac{1}{2\lambda}.$$

Как и в предыдущем примере находим, что

$$\lambda = \pm \frac{\sqrt{2}}{2}.$$

Это дает две точки $N_1\left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right)$ и $N_2\left(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$. Однако обе они не принадлежат области S, поскольку их координаты не удовлетворяют неравенству $x + y \leqslant 1$. Попытка найти условный экстремум на прямой $x + y = 1$ приводит к дифференцированию функции

$$h_2 = x - y + \mu (x + y - 1).$$

Получающиеся уравнения $1 + \mu = 0$ и $-1 + \mu = 0$ несовместны. Следовательно, условного экстремума на прямой не существует, и остается проверить лишь две граничные точки $A(0, 1)$ и $B(1, 0)$. Подставляя их координаты в функцию $h = x - y$, находим, что в точке B эта функция достигает максимума, равного 1, а в точке A — минимума, равного -1.

При решении реальных задач оптимизации, возникающих в практике управления, описанный классический метод применяется крайне редко. Основная причина — громоздкость метода: необходимость рассмотрения большого числа задач, каждая из которых требует решения систем уравнений, в большинстве случаев достаточно сложных. Кроме того, далеко не всегда функция f, для которой ищется экстремум, обладает свойством дифференцируемости. Зачастую она даже не может быть задана в виде формулы и задается лишь более или менее сложной программой. Подобные сложные экстремальные задачи решаются обычно другими методами, в основе которых лежит идея постепенного приближения к экстремуму.

Одним из наиболее известных методов является метод *наискорейшего (градиентного) спуска или подъема*. Его идею проще всего пояснить на примере, когда целевая функция f зависит лишь от двух переменных $f = f(x, y)$. В этом случае она описывает обычную двумерную поверхность, которую удобно представлять себе в виде какого-то рельефа (обычно сильно изрезанного) земной поверхности.

Предположим, что наша задача состоит в нахождении максимума функции f в некоторой области S значений x и y. Возьмем произвольную точку $A(x, y)$ этой области и вообразим, что в соответствующей ей точке поверхности $B[x, y, f(x, y)]$ стоит человек. Если рельеф

во все стороны от него понижается, т. е., если точка B — самая высокая, по сравнению со всеми достаточно близкими к ней точками, то, как принято говорить, она (и, соответственно, точка A) является точкой *локального максимума* функции f. Если же локальный максимум в этой точке не достигается, то среди всех направлений движения из нее можно выбрать такое направление, по которому подъем будет наиболее крутым. Это направление носит название *градиента* функции f или направления *наискорейшего подъема*. В математическом анализе доказывается, что, в случае дифференцируемости функции f, градиент определяется единственным образом для всех точек, исключая те, в которых все частные производные этой функции одновременно обращаются в нуль. Направление, противоположное градиенту, будет направлением *наискорейшего спуска*.

Все указанные рассмотрения легко обобщаются на случай функции n переменных $f = f(x_1, x_2, \ldots, x_n)$. Направление градиента (наискорейшего подъема) такой функции задается вектором

$$\bar{g} = \left(\frac{\partial f}{\partial x_1}, \quad \frac{\partial f}{\partial x_2}, \quad \ldots, \quad \frac{\partial f}{\partial x_n}\right),$$

противоположное направление (наискорейшего спуска) — вектором $-\bar{g}$. Если требуется найти экстремум функции f в области S, то процесс приближения к нему может быть начат из любой точки $A_0(x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$ этой области. Для этой цели вычисляется градиент $\bar{g}_0$ в этой точке и осуществляется сдвиг от точки A_0 на вектор $c_0\bar{g}_0$, где c_0 — некоторая константа (положительная при движении к максимуму и отрицательная при движении к минимуму). В результате получаем новую точку A_1 $(x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)})$, в которой снова повторяем описанную процедуру и т. д. и т. п., пока, наконец, не дойдем до точки условного экстремума*.

Указанная процедура и составляет метод наискорейшего спуска (подъема). Выбор значений констант c_0, c_1, ... для каждого шага может осуществляться несколькими различными способами. Простейший (но далеко не самый лучший) из них состоит в том, чтобы выбирать эти значения в соответствии с точностью α, с которой определяются значения переменных x_1, x_2, ..., x_n (например, 0,01, 0,001 и т. п.). Значение константы c_i принимается при этом равным $\frac{\alpha_i}{g_i}$, где через g_i обозначена длина вектора $\bar{g}_i$, равная

$$\sqrt{\left(\frac{\partial f}{\partial x_1}\right)^2 + \left(\frac{\partial f}{\partial x_2}\right)^2 + \cdots + \left(\frac{\partial f}{\partial x_n}\right)^2}$$

* При конечном числе шагов процесс, вообще говоря, приводит лишь в точку, сколь угодно близкую к экстремуму.

(предполагается, что значения производных $\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}$ вычисляются в точке A_i).

Более рациональным может оказаться подбор методом последовательных проб наибольшего (по абсолютной величине) значения c_i, при котором сдвиг вдоль вектора $c_i\bar{g}_i$ все еще приводит к увеличению (при нахождении максимума) или к уменьшению (при нахождении минимума) значения функции f. При этом, разумеется, надо все время следить, чтобы новая точка A_{i+1} оставалась в пределах заданной области S. Когда процесс наискорейшего спуска (подъема) на каком-то шаге приводит к границе области, дальнейшее движение должно производиться вдоль этой границы.

Градиентный метод (как, впрочем, и описанный выше классический) приводит первоначально к нахождению лишь локальных экстремумов. *Абсолютный экстремум* выбирается после нахождения всех локальных экстремумов методом простого сравнения значений целевой функции f во всех точках локальных экстремумов. Классический метод приводит, однако, к нахождению всех локальных экстремумов, а градиентный метод в описанном выше виде — лишь одного из них. Поэтому обычно градиентный метод применяется многократно — при различном (случайном) выборе начальных точек A_0. Если при достаточно большом числе повторений перестают отыскиваться новые локальные экстремумы, это может служить известной (но, разумеется, не абсолютной) гарантией того, что найдены все локальные экстремумы, и можно переходить к их сравнению и выбору абсолютного экстремума.

При практическом применении градиентного метода, как правило, вместо истинного направления градиента находится его приближенное значение. С этой целью каждой из переменных x_1, x_2, ..., x_n придается последовательно одно и то же достаточно малое приращение δ и вычисляются соответствующие приращения целевой функции Δ_1, Δ_2, ..., Δ_n. Вектор $\bar{p}=(\frac{\Delta_1}{\delta}, \frac{\Delta_2}{\delta}, \ldots, \frac{\Delta_n}{\delta})$ задает искомое приближение направления градиента. В дальнейших вычислениях вместо градиента используется этот вектор.

В большинстве практических задач оптимизации градиентный метод оказывается во много раз более экономным по сравнению с классическим. Его преимущество перед классическим методом становится особо ощутимым при росте числа переменных $x_1, x_2, \ldots, x_n$ или, как обычно принято говорить — при росте *размерности* задачи оптимизации.

Имеется и дальнейшая возможность уменьшения количества вычислений в методах последовательного приближения к экстремуму. Можно, в частности, вместо градиента $\bar{g}$ выбирать случайный вектор $\bar{g}$ и исследовать значение целевой функции f в точке,

получаемой из начальной точки достаточно малым сдвигом $\hat{cg}$. Если значение функции f увеличивается, то при поиске максимума этот сдвиг заменяет градиентный, а при поиске минимума надо изменить направление сдвига на противоположное. В случае, если при первоначальном сдвиге функция f уменьшается, то выбор нужного направления сдвига производится прямо противоположным образом.

Описанная процедура составляет суть так называемого метода *стохастического программирования*. Подобный процесс стохастического приближения сводится к локальному экстремуму при достаточно общих предположениях о целевой функции $f(x_1, x_2, \ldots, x_n)$.

2. Линейное программирование

На практике довольно часто встречается частный случай задачи оптимизации, когда при нахождении экстремума как целевая функция f, так и уравнения границ области S являются линейными функциями. Задача нахождения экстремума в этом случае носит специальное название задачи *линейного программирования*. Опишем одну из важных практических задач управления, сводящихся к линейному программированию.

Пусть имеем n производственных объектов $A_1, A_2, \ldots, A_n$ (цехов или предприятий), которые предназначены для производства m различных видов изделий $B_1, B_2, \ldots, B_m$. Каждый из объектов располагает некоторыми ресурсами (оборудованием различных типов, производственными площадями, рабочими различных профессий и квалификаций и т. п.). Предположим, что общее число типов ресурсов равно k: $C_1, C_2, \ldots, C_k$. Для каждого изделия B_i задан директивный план b_i, т. е. минимальное количество изделий вида B_i, которое необходимо произвести в совокупности на всех объектах.

Предполагается также, что для каждого вида изделий B_i заданы нормативы, т. е. количество единиц времени использования каждого из ресурсов $C_1, C_2, \ldots, C_k$, необходимых для производства одной единицы изделий данного вида. Норматив для i-го изделия по j-му ресурсу обозначим b_{ij}*.

При установленной продолжительности рабочего дня и рабочей недели, а также коэффициента сменности, каждый объект A_r будет располагать вполне определенным максимально возможным суммарным временем для использования каждого из ресурсов C_j в течение рассматриваемого планового периода (станкочасы, человекодни и т. п.). Обозначим соответствующие величины через a_{rj} ($r = 1, 2, \ldots, n$; $j = 1, 2, \ldots, k$).

* Предполагаем здесь, что нормативы для всех объектов одинаковы. В случае необходимости можно ввести дифференцированные нормативы b_{ij}, различные для разных объектов.

Поставим задачу нахождения таких планов x_{ri} выпуска изделий i-го вида на r-ом объекте, при которых достигается максимально возможная суммарная прибыль для всех рассматриваемых объектов, при условии, что суммарный выпуск изделий каждого вида будет не меньше соответствующего директивного плана.

Обозначим p_{ri} — прибыль, получаемую на r-ом объекте при производстве одной единицы изделия i-го вида*.

Целевая функция (суммарная прибыль) определяется формулой

$$P = \sum_{r=1}^{n} \sum_{i=1}^{m} p_{ri} x_{ri}. \qquad (1)$$

При нахождении максимума целевой функции на значения переменных x_{ri} накладываются ограничения трех различных типов. Во-первых, это простейшие ограничения, выражающие тот очевидный факт, что искомые планы x_{ri} не могут быть отрицательными:

$$x_{ri} \geqslant 0 \quad (r = 1, \ \ldots, \ n; \quad i = 1, \ \ldots, \ m). \qquad (2)$$

Второй тип ограничений связан с необходимостью выполнения директивных планов:

$$\sum_{r=1}^{n} x_{ri} \geqslant b_i \quad (i = 1, \ 2, \ \ldots, \ m). \qquad (3)$$

Третий тип ограничений возникает из условий ограниченности ресурсов на каждом из объектов $A_1, \ A_2, \ \ldots, \ A_n$:

$$\sum_{i=1}^{n} x_{ri} b_{ij} \leqslant a_{rj} \quad (r = 1, \ \ldots, \ n; \ j = 1, \ \ldots, \ k). \qquad (4)$$

Поскольку формулы (1)—(4) линейны относительно переменных x_{ri}, то имеем задачу линейного программирования. Общее количество N границ области S, в которой ищется максимум функции P, равняется $mn + m + kn = n(m + k) + m$. Именно таково общее число неравенств (2)—(4).

Фундаментальным фактом, устанавливаемым теорией для задач линейного программирования, является то, что экстремум целевой функции всегда достигается в вершинах многогранника, которым в таких задачах является область S поиска экстремума.

Если при этом окажется, что целевая функция принимает одно и то же экстремальное значение не в одной, а в нескольких вершинах, то она будет принимать то же самое (а следовательно, экстре-

---

* При необходимости можно рассматривать случай, когда прибыль от плановой и сверхплановой продукции различна.

мальное) значение во всех точках задаваемой этими вершинами грани. Поэтому в принципе задача линейного программирования может быть решена путем нахождения всех вершин многогранника S, вычисления значений целевой функции f во всех вершинах и выбора из них вершины с экстремальным значением функции f. Однако на практике этот способ оказывается весьма громоздким, поскольку каждая вершина находится в результате решения систем линейных уравнений, имеющих достаточно большую размерность, а число вершин, как правило, тоже велико*.

Имея в виду сказанное, на практике при решении задач линейного программирования используются другие, более экономные способы. Опишем один из известных общих методов решения задач линейного программирования, называемого *методом симплексных преобразований* или просто *симплексным методом*.

Первым шагом при применении этого метода является сведение системы ограничений от неравенств к равенствам. Это делается стандартным способом с помощью введения новых (вспомогательных) переменных: например, неравенство $2x + 3y - 1 \geqslant 0$ можно заменить равенством $2x + 3y - 1 = z$, где z — вспомогательная переменная, принимающая любые неотрицательные значения.

В задачах, обычно встречающихся на практике, *основные переменные* (x_{ri} в рассмотренном выше примере) не могут принимать отрицательные значения. Вспомогательные переменные, возникающие при сведении неравенств к равенствам, также могут всегда быть выбраны таким образом, чтобы удовлетворять этому условию. Поэтому *общая задача линейного программирования* может быть сформулирована следующим образом: найти неотрицательное решение заданной системы линейных уравнений, обращающее в максимум или в минимум заданную линейную функцию f.

Возможно и дальнейшее сужение задачи. Если целевая функция f достигает максимума в некоторой точке A области S, то функция $f_1 = -f$, очевидно, достигает в этой точке минимума. Если функция f линейна, то линейной будет и функция f_1. Поэтому, не нарушая общности, можно считать, что общая задача линейного программирования всегда состоит в нахождении минимума, а не максимума целевой функции.

Общее число переменных (как основных, так и вспомогательных) называется размерностью задачи линейного программирования. При этом следует иметь в виду, что ограничения типа $x \geqslant 0$ для переменных считаются тривиальными, так что соответствующие неравенства $x \geqslant 0$ не должны заменяться равенствами и, вообще, играть какую-либо явную роль в дальнейших построениях. Метод

* При N переменных и M ограничениях число подлежащих проверке точек равно C_M^N.

ɔешения задачи строится таким образом, что эти ограничения будут ıвтоматически выполняться.

Учитывая все сказанное, нетрудно свести к описанному стандартному виду рассмотренный выше пример.

Необходимо найти неотрицательное решение системы линейных уравнений

$$z_i = \sum_{r=1}^{n} x_{ri} - b_i \quad (i = 1, \ldots, m); \tag{5}$$

$$z_{rj} = a_{rj} - \sum_{i=1}^{m} b_{ij} x_{ri} \; (r = 1, \ldots, n; \; j = 1, \ldots, k), \tag{6}$$

обращающее в минимум линейную целевую функцию

$$P_1 = -\sum_{r=1}^{n} \sum_{i=1}^{m} p_{ri} x_{ri}. \tag{7}$$

Число основных переменных (x_{ri}) в этой задаче равно, очевидно, nm, а число вспомогательных переменных (z_i и z_{rj}) — $m + nk$. Таким образом, размерность задачи в рассматриваемом случае равна $n(m + k) + m$. Как видим, размерность совпадает с общим числом ограничений, имевшихся в первоначальной постановке задачи (см. выше). Нетрудно понять, что это будет иметь место всякий раз, когда все первоначальные ограничения заданы неравенствами*.

Переходя к описанию симплексного метода решения общей задачи линейного программирования, заметим прежде всего, что его основой являются так называемые симплексные преобразования заданной системы линейных уравнений.

Для возможности применения симплексных преобразований уравнения системы нужно переписать в правильной форме, которая может иметь один из следующих двух видов:

$$0 = b - (a_1 x_1 + a_2 x_2 + \cdots + a_n x_n), \quad \text{где } b \geqslant 0, \tag{8}$$

или

$$x_i = d - (c_1 x_1 + \cdots + c_{i-1} x_{i-1} + c_{i+1} x_{i+1} + \cdots + c_n x_n), \tag{9}$$

где

$$d \geqslant 0.$$

Уравнения первого вида будем называть 0-уравнениями, а второго — x-уравнениями или более точно, уравнениями, разрешенными относительно x_i. При этом всегда будем предполагать, что

---

* В общем случае не исключено, что некоторые ограничения будут заданы в виде равенств.

при наличии в системе x-уравнений во всех остальных уравнениях произведены замены неизвестных x_l, относительно которых разрешены эти уравнения, соответствующими правыми частями выражений (9).

Легко понять, что к выражению вида (8) может быть приведено всякое уравнение, а к виду (9) — лишь такое, у которого знак коэффициента при x_l противоположен знаку свободного члена.

Поясним сущность симплексных преобразований на простом примере, иллюстрируя попутно и другие ранее уже объясненные вопросы.

Пусть требуется найти минимум линейной функции $f = 6 - x - 3y$ в области S, заданной системой неравенств $y \leqslant 2$; $x + y \leqslant 3$; $x \geqslant 0$; $y \geqslant 0$. Для решения задачи прежде всего обращаем нетривиальные неравенства в равенства:

$$y + z = 2; \quad x + y + u = 3.$$

Перепишем полученные равенства в правильной форме:

$$0 = 2 - (y + z); \quad 0 = 3 - (x + y + u).$$

Для выполнения симплексного преобразования выберем произвольным образом одно из неизвестных. Пусть это будет y. Далее, выделяем те уравнения, в которых коэффициенты при выбранном неизвестном положительны. В данном случае этому условию удовлетворяют оба уравнения (знак минус перед скобками при этом во внимание не принимается). Делим свободные члены соответствующих уравнений на эти коэффициенты и выбираем из полученных отношений $\frac{2}{1}$ и $\frac{3}{1}$ наименьшее (первое). Соответствующее уравнение (первое) разрешаем относительно выделенного неизвестного:

$$y = 2 - (z),$$

а полученное значение y подставляем в остальные уравнения $0 = 3 - (x + 2 - (z) + u)$, что приводит благодаря минимальности отношения $\frac{2}{1}$ к уравнениям правильного вида: $0 = 1 - (x - z + u)$.

В результате получаем новую систему уравнений:

$$\left.\begin{aligned} y &= 2 - (z); \\ 0 &= 1 - (x - z + u), \end{aligned}\right\}$$

которая и является результатом выполненного симплексного преобразования.

Применение симплексного преобразования к переменной x приводит к выбору второго уравнения и, после его решения относительно x и подстановки (в данном случае тривиальной) результата в остальные уравнения,— к системе, состоящей только из x-уравнений:

$$\left.\begin{aligned} y &= 2 - (z); \\ x &= 1 - (-z + u). \end{aligned}\right\} \qquad (10)$$

Когда после выполнения надлежащего числа симплексных преобразований получаем систему, состоящую только из x-уравнений, можно переходить к следующему шагу. Для этого подставляем в целевую функцию f вместо y и x соответствующие правые части из уравнений (10):

$$f = 6 - (1 + z - u) - 3(2 - z) = -1 + 2z + u. \qquad (11)$$

Поскольку $z \geqslant 0$ и $u \geqslant 0$, то к -1 в правой части соотношения (11) прибавляется неотрицательное число. Наименьшее значение для f (равное минус 1) получится, очевидно, при $z = u = 0$. Соответствующие значения x и y находятся из уравнений (10):

$$y = 2 - 0 = 2; \quad x = 1 - 0 = 1.$$

Этот результат и дает решение рассматриваемой задачи линейного программирования.

Для другой целевой функции, например $g = 4 - x + y$, подстановка значений x и y из уравнений (10) приводит к результату:

$$g = 4 - (1 + z - u) + (2 - z) = 5 - 2z + u.$$

Поскольку коэффициент при z здесь отрицателен, то приведенные выше рассуждения неприменимы. В этом случае к системе (10) нужно еще раз применить симплексное преобразование, причем, именно по переменной z;

$$\begin{aligned} z &= 2 - (y); \\ x &= 3 - (y + u). \end{aligned}$$

После подстановки этих выражений в целевую функцию получаем

$$g = 5 - 2(2 - y) + u = 1 + y + u.$$

Поскольку снова коэффициенты при неизвестных положительны, то (учитывая, что $y \geqslant 0$ и $u \geqslant 0$) искомый минимум (равный 1) получается при $y = u = 0$. Точка, в которой достигается этот минимум, имеет координаты $y = 0$ и $x = 3 - (0 + 0) = 3$.

Если в каком-либо случае окажется, что применить требуемое симплексное преобразование невозможно, то соответствующая задача линейного программирования неразрешима. Это может случиться как на первом этапе (преобразования 0-уравнений в x-уравнения), так и на втором (ликвидации отрицательных коэффициентов при неизвестных в целевой функции). Если, например, область S задана неравенствами $x \geqslant 0, y \geqslant 0, x + y \leqslant -1$, соответствующая система равенств сводится к одному уравнению $x + y + z = -1$ или, в правильном виде, $0 = 1 - (-x - y - z)$. Так как все коэффициенты при неизвестных отрицательны, то 0-уравнение не может быть приведено к x-уравнению при симплексном преобразовании. Это означает, что наше уравнение не имеет неотрицательных решений (область пуста) и, следовательно, задача линейного программирования не имеет решения.

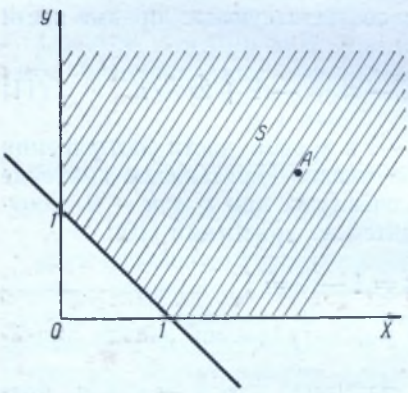

Рис. 14.
График значений целевой функции.

Другой пример. Для $x \geqslant 0$, $y \geqslant 0$, $x + y \geqslant 1$ найти минимум функции $f = 1 - x$. Система равенств сводится к одному уравнению:

$$x + y = 1 + z$$

или, в правильной форме

$$0 = 1 - (x + y - z).$$

Это уравнение может быть симплексно разрешено как относительно x, так и относительно y:

$$x = 1 - (y - z) \quad \text{и} \quad y = 1 - (x - z).$$

В обоих случаях преобразованная целевая функция не приводится к требуемому виду. В первом случае $f = y - z$, во втором $f = 1 - x$.

Поскольку при этом остаются отрицательные коэффициенты при неизвестных, задача линейного программирования не имеет решения. Однако причина отсутствия решения на этот раз иная, чем в предыдущем случае. Область S (заштрихована) бесконечна, а целевая функция убывает по мере роста x (рис. 14). Поэтому для любой точки A области S найдутся новые точки (расположенные правее точки A), в которых значение целевой функции f меньше, чем в точке A.

Имеется еще одна причина (правда, достаточно редко встречающаяся), которая может помешать решению задачи линейного

программирования симплексным методом даже в том случае, когда эта задача имеет решение. Это так называемое зацикливание. Зацикливание заключается в том, что после некоторого числа раз выполнения симплексных преобразований преобразуемая система уравнений возвращается к исходному виду, не исчерпав всех возможных базисов (т. е. совокупности неизвестных, относительно которых система может быть x — разрешена). Для устранения этого в симплексный метод вводятся специальные дополнения, которые рассматриваются в более подробных курсах (см., например, [11]). При наличии этих дополнений симплексный метод приводит к решению задачи линейного программирования во всех случаях, когда такое решение существует.

Если обозначить через M число уравнений в общей задаче линейного программирования, а через N — общее число неизвестных (как основных, так и вспомогательных), то число необходимых арифметических операций для выполнения одного симплексного преобразования в общем случае выражается приближенно формулой $2MN$. Для ориентировочных расчетов можно считать, что эта величина имеет порядок квадрата размерности задачи.

Возвращаясь к примеру в начале этого параграфа, предположим, что нам необходимо найти оптимальное распределение планов между 20 предприятиями ($n = 20$) по 50 видам изделий ($m = 50$) при учете 10 видов ресурсов ($k = 10$). Из соотношений (1)—(4) находим общее число переменных N

$$n(m + k) + m = 20(50 + 10) + 50 = 1250,$$

а общее число уравнений, равное числу нетривиальных неравенств (3), (4),

$$m + nk = 50 + 20 \cdot 10 = 250.$$

Отсюда

$$2MN = 2 \cdot 1250 \cdot 250 = 625\,000.$$

Учитывая, что в задачах такой размерности для нахождения решения приходится выполнять сотни и даже тысячи симплексных преобразований, получаем, что решение указанной задачи может привести к необходимости выполнения сотен миллионов и даже миллиардов арифметических операций.

Имеются частные случаи задач линейного программирования, для решения которых существуют более простые приемы, чем описанный выше симплексный метод. Наиболее важными являются так называемые *транспортные задачи*, имеющие целью нахождение плана перевозок по *критерию наименьшей стоимости*, или задача, имеющая целью минимизировать время перевозок. Рассмотрим

лишь первый случай, который наиболее часто встречается на практике.

Формальная постановка транспортной задачи состоит в том, что фиксируется m пунктов отправления однородного груза, в которых имеется соответственно $a_1, a_2, \ldots, a_m$ единиц этого груза, и n пунктов назначения с потребностями в $b_1, b_2, \ldots, b_n$ единиц груза соответственно. Кроме того, заданы стоимости c_{ij} перевозки одной единицы груза из каждого пункта отправления в каждый пункт назначения. Требуется определить количества x_{ij} единиц груза, которые нужно перевести из i-го пункта отправления в j-й пункт назначения ($i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$) так, чтобы суммарная стоимость перевозки $f = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$ была бы минимальной, и потребности всех пунктов назначения были бы удовлетворены.

Обычно предполагается, что суммарное количество груза, имеющегося во всех пунктах отправления, равно общей потребности всех пунктов назначения:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j.$$

При этих условиях система ограничений, при которых необходимо решить задачу, сводится к $m + n$ равенствам:

$$\sum_{j=1}^{n} x_{ij} = a_i \quad (i = 1, 2, \ldots, m);$$

$$\sum_{i=1}^{m} x_{ij} = b_j \quad (j = 1, 2, \ldots, n).$$

Первая система равенств выражает тот очевидный факт, что все грузы должны быть вывезены из пунктов отправления, а вторая система — то, что потребности всех пунктов назначения должны быть удовлетворены.

Для решения поставленной задачи рассмотрим матрицу $C = \| c_{ij} \|$ стоимостей перевозок. Будем называть элемент c_{ij} этой матрицы выбранным, если мы задали некоторый план перевозок x_{ij} из i-го пункта отправления в j-ый пункт назначения. В теории транспортной задачи доказывается, что ее решение может быть получено выбором не более $m + n - 1$ ненулевых планов, и что матрица $X = \| x_{ij} \|$ оптимального плана не может содержать циклов. Под циклом здесь понимается последовательность ненулевых элементов x_{ij}, начинающаяся и кончающаяся одним и тем же элементом, и такая, что любые два соседних элемента x_{ij}, x_{kl} последовательности расположены либо в одной строке ($i = k$), либо в одном столбце ($j = l$) матрицы X.

Так, например, в матрице

$$X = \begin{Vmatrix} 0 & a & 0 & b \\ c & d & e & 0 \\ 0 & 0 & f & g \end{Vmatrix}$$

последовательность элементов a, d, e, f, g, b, a составляет цикл.

Всякий цикл будем делить на два полуцикла: четный и нечетный, собирая в них элементы последовательности, образующей цикл, с четными и нечетными номерами. В только что рассмотренном примере нечетный полуцикл образуют элементы a, e, g, a, а четный — d, f, b.

Решение транспортной задачи начинается с построения первого варианта матрицы плана X, который должен строиться так, чтобы были выбраны ровно $m + n - 1$ элементарных планов $x_{ij} \geqslant 0$ и чтобы из этих планов нельзя было образовать ни одного цикла. Метод построения этого первого варианта лучше всего проиллюстрировать на примере.

Пусть имеется три пункта отправления с запасами 10, 15 и 20 единиц груза и четыре пункта назначения с потребностями 15, 15, 10 и 5 единиц груза. Выпишем матрицу стоимостей, обратив ее строки и столбцы запасами и потребностями по соответствующим пунктам:

	15	15	10	5
10	2	3	4	4
15	3	4	2	3
20	1	3	2	4

Начальный выбор может быть сделан последовательным (строка за строкой) распределением грузов, имеющихся в пунктах отправления, в пункты назначения с еще не удовлетворенными потребностями, стоимость перевозки в которые минимальна. Так, весь груз, имеющийся в первом пункте отправления, направим в первый пункт назначения. Из 15 единиц груза во втором пункте отправления 10 занарядим в третий пункт назначения, а оставшиеся 5 единиц — в первый пункт назначения. Из третьего пункта отправления 15 единиц груза направим во второй, а оставшиеся 5 — в четвертый пункт назначения.

В последнем случае окончание распределения груза в третьем пункте отправления явилось одновременно и окончательным удовлетворением потребности четвертого пункта назначения. Для того чтобы в начальном выборе оказалось нужное число планов ($m + n - 1 = 3 + 4 - 1 = 6$) в соответствующей строке (в данном случае в третьей) вводится еще один добавочный нулевой план.

Матрица начальных планов запишется в виде:

	15	15	10	5
10	10	—	—	—
15	5	—	10	—.
20	0	15	—	5

Для улучшения полученных начальных планов матрица C стоимостей перевозок подвергается *эквивалентным преобразованиям*. Элементарное эквивалентное преобразование состоит в том, что ко всем элементам одного и того же столбца, или одной и той же строки, одновременно прибавляется или вычитается одно и то же число. Цель эквивалентных преобразований состоит в том, чтобы сделать все выбранные (соответствующие выбранным планам) элементы матрицы стоимостей C равными нулю. Для достижения этой цели отметим выбранные элементы квадратиками:

$$C = \left\| \begin{matrix} \boxed{2} & 3 & 4 & 4 \\ \boxed{3} & 4 & \boxed{2} & 3 \\ \boxed{1} & \boxed{3} & 2 & \boxed{4} \end{matrix} \right\|.$$

Для обращения в нуль элемента c_{31} (равного 1) вычитаем из элементов первого столбца 1:

$$\left\| \begin{matrix} \boxed{1} & 3 & 4 & 4 \\ \boxed{2} & 4 & \boxed{2} & 3 \\ \boxed{0} & \boxed{3} & 2 & \boxed{4} \end{matrix} \right\|.$$

Далее из элементов первой строки вычитаем 1, а из элементов второй — 2:

$$\left\| \begin{matrix} \boxed{0} & 2 & 3 & 3 \\ \boxed{0} & 2 & \boxed{0} & 1 \\ \boxed{0} & \boxed{3} & 2 & \boxed{4} \end{matrix} \right\|.$$

В заключение вычитаем из элементов второго столбца 3, а из элементов четвертого столбца — 4:

$$\left\|\begin{array}{cccc} \boxed{0} & -1 & 3 & -1 \\ \boxed{0} & -1 & \boxed{0} & -3 \\ \boxed{0} & \boxed{0} & 2 & \boxed{0} \end{array}\right\|.$$

Если бы все элементы полученной матрицы были неотрицательными, то первоначальный выбор давал бы искомое оптимальное решение. Поскольку в нашем случае это не имеет места, применяется процедура исправления начального решения. Для исправления полученного решения выбираем наибольший по абсолютной величине отрицательный элемент преобразованной матрицы стоимостей (в данном случае $c_{24} = -3$) и присоединяем его к начальному выбору. Если начальный выбор содержал ровно $m + n - 1$ элементов, то, как доказывается в теории, при таком присоединении обязательно образуется цикл. В данном случае такой цикл образуется элементами

$$c_{24} = -3; \quad c_{34} = 0; \quad c_{31} = 0; \quad c_{21} = 0$$

(цикл начинается с вновь присоединенного элемента!). Передвигаем теперь из четного полуцикла в нечетный число единиц груза, являющееся наименьшим в этом полуцикле*. Получаем новый выбор планов:

	15	15	10	5
10	10	—	—	—
15	—	—	10	5
20	5	15	—	0

Матрицу стоимостей вновь преобразуем так, чтобы выбранные элементы обратились в нуль. Для этой цели к элементам второй строки прибавляем 3, а из элементов третьего столбца вычитаем 3:

$$\left\|\begin{array}{cccc} \boxed{0} & -1 & 0 & -1 \\ 3 & 2 & \boxed{0} & \boxed{0} \\ \boxed{0} & \boxed{0} & -1 & \boxed{0} \end{array}\right\|.$$

Поскольку снова в матрице имеются отрицательные элементы, используем примененное выше преобразование. Фиксируем один

* Продвижение делается по всем элементам цикла: из каждого элемента четного полуцикла в следующий за ним элемент нечетного полуцикла.

из наибольших по абсолютной величине отрицательных элементов, например, $c_{33} = -1$ и образуем цикл

$$c_{33} = -1; \quad c_{23} = 0; \quad c_{24} = 0; \quad c_{34} = 0.$$

Опять производим передвижение из четного полуцикла в нечетный наименьшее число единиц груза, встречающееся в полуцикле (в данном случае 0). Получаем новую матрицу планов:

	15	15	10	5
10	10	—	—	—
15	—	—	10	5
20	5	15	0	—

Для обращения в нуль вновь выбранного элемента $c_{33} = -1$ матрицы стоимостей

$$\begin{Vmatrix} \boxed{0} & -1 & 0 & -1 \\ 3 & 2 & \boxed{0} & \boxed{0} \\ \boxed{0} & \boxed{0} & \boxed{-1} & 0 \end{Vmatrix}$$

к третьему столбцу прибавляем 1, из второй строки вычитаем 1, а к четвертому столбцу прибавляем 1:

$$\begin{Vmatrix} \boxed{0} & -1 & 1 & 0 \\ 2 & 1 & \boxed{0} & \boxed{0} \\ \boxed{0} & \boxed{0} & \boxed{0} & 1 \end{Vmatrix}.$$

Снова присоединяем остающийся отрицательный элемент $c_{12} = -1$ и образуем цикл:

$$c_{12} = -1; \quad c_{32} = 0; \quad c_{31} = 0; \quad c_{11} = 0.$$

После передвижения 10 единиц груза приходим к матрице планов:

	15	15	10	5
10	—	10	—	—
15	—	—	10	5
20	15	5	0	—

Соответствующая матрица преобразованных стоимостей

$$\begin{Vmatrix} 0 & \boxed{-1} & 1 & 0 \\ 2 & 1 & \boxed{0} & \boxed{0} \\ \boxed{0} & \boxed{0} & \boxed{0} & 1 \end{Vmatrix}.$$

Прибавим 1 к первой строке

$$\begin{Vmatrix} 1 & \boxed{0} & 2 & 1 \\ 2 & 1 & \boxed{0} & \boxed{0} \\ \boxed{0} & \boxed{0} & \boxed{0} & 1 \end{Vmatrix}.$$

Поскольку отрицательные элементы в полученной матрице отсутствуют, последний из планов перевозок будет оптимальным. Согласно этому плану весь груз из первого пункта отправления (10 единиц) следует занарядить во второй пункт назначения. Из 15 единиц груза, имеющихся во втором пункте отправления, 10 следует направить в третий, а 5 — в четвертый пункты назначения. Остающиеся 20 единиц груза (в третьем пункте отправления) распределяются между первым (15 единиц) и вторым (5 единиц) пунктами назначения.

По первоначальной матрице стоимостей вычисляем общую стоимость перевозок по полученному плану:

$$C_{\text{опт}} = 10 \cdot 3 + 10 \cdot 2 + 5 \cdot 3 + 15 \cdot 1 + 5 \cdot 3 = \\ = 30 + 20 + 15 + 15 + 15 = 95.$$

Стоимость перевозок при первоначальном плане

$$C_{\text{ном}} = 10 \cdot 2 + 5 \cdot 3 + 10 \cdot 2 + 15 \cdot 3 + 5 \cdot 4 = \\ = 20 + 15 + 20 + 45 + 20 = 120.$$

Таким образом, оптимальный план более чем на 20% экономнее начального.

Читатель может поупражняться на этом примере, сравнивая с оптимальным планом различные планы, составляемые по интуиции. Ввиду простоты рассмотренного примера возможно достаточно хорошее интуитивное приближение к оптимальному плану, не исключено даже его точное угадывание*. Однако по мере роста

* Как и во всякой задаче линейного программирования, оптимальный план может не быть единственным.

размерности задачи вероятность угадывания оптимального плана и даже более или менее хорошего приближения к нему становится практически нулевой.

Рассмотренный пример демонстрирует еще одну особенность, свойственную всем задачам оптимального планирования. Дело в том, что любой оптимальный план характеризуется не только меньшей стоимостью, но и, как правило, меньшим числом материальных ресурсов, необходимых для его выполнения. В рассмотренном примере при оптимальном планировании перевозок уменьшается потребность в количестве транспортных средств (автомобилей; вагонов и др.). Таким образом руководитель за счет оптимального планирования как бы приобретает дополнительные (скрытые до того) ресурсы.

В заключение параграфа заметим, что существует немало случаев, когда при решении задач линейного программирования допускаются лишь целочисленные решения. С таким положением приходится встречаться всякий раз при определении относительно небольших количеств «штучных» (т. е. не дробящихся на части) объектов (станки, автомашины, суда и т. п.). Соответствующие задачи называются задачами целочисленного линейного программирования. Обычный симплексный метод оказывается, вообще говоря, непригодным для задач такого рода, и для их решения отыскиваются специальные методы. Теория этих методов сегодня еще далека от завершения, и решение задач целочисленного линейного программирования обычно сопряжено с большими вычислительными трудностями.

3. Динамическое программирование

На практике очень часто приходится встречаться со случаями, когда целью оптимального планирования является установление наилучшей последовательности тех или иных работ (производственных операций, этапов строительства различного рода сооружений и т. п.).

Одной из первых задач такого рода, привлекших внимание математиков, была так называемая задача о коммивояжере (бродячем торговце). Суть ее состоит в следующем: имеется $n+1$ городов A_0, A_1, ..., $A_n (n \geqslant 1)$ с заданными между ними расстояниями d_{ij} $(i, j = 0, 1, \ldots, n)$; требуется, отправляясь из A_0, выбрать такой маршрут передвижения A_0, A_{i1}, A_{i2}, ..., A_{in}, A_0, при котором коммивояжер, побывав в каждом городе по одному разу, вернулся бы в исходный пункт A_0, проделав минимально возможный суммарный путь.

Эта задача может быть в принципе решена методом простого перебора всех возможных маршрутов. Легко видеть, однако, что

общее число таких маршрутов равно $1 \cdot 2 \cdot 3 \ldots (n-1)n$, а эта величина уже при n, измеряемом несколькими десятками, столь огромна, что практически исключает возможность прямых вычислений даже при использовании ЭВМ. Проблема, следовательно, состоит в том, чтобы резко сократить перебор, отбрасывая заранее заведомо непригодные множества вариантов. Методы решения этой проблемы составляют суть так называемого *динамического программирования*.

Рассмотрим основные принципы динамического программирования на примере задачи о коммивояжере. Пусть имеется всего пять пунктов: A_0, A_1, A_2, A_3, A_4 со следующей таблицей расстояний*:

	A_0	A_1	A_2	A_3	A_4
A_0	0	300	250	200	400
A_1	300	0	500	350	600
A_2	250	500	0	250	200
A_3	200	350	250	0	250
A_4	400	600	200	250	0

Для определения кратчайшего пути коммивояжера будем развивать варианты его передвижения последовательно, пункт за пунктом. Начинаем с вариантов, состоящих из трех участков: $A_0A_{i1}A_{i2}A_{i3}$, группируя их по последнему пункту A_{i3}. Некоторые из полученных вариантов можно отбросить, не развивая их далее. Так, например, сравнивая варианты $A_0A_1A_2A_3$ и $A_0A_2A_1A_3$ получаем, что для первого варианта суммарный путь равен $300 + 500 + 250 = 1050$, а для второго — $250 + 500 + 350 = 1100$. Поскольку оба варианта относятся к одному и тому же множеству пунктов и оканчиваются в одном и том же пункте (A_3), то в дальнейшем оба варианта могут развиваться одинаково. Ясно, что проигрыш 50 *км*, полученный для второго варианта, делает его бесперспективным в сравнении с первым при любом его дальнейшем развитии. Следовательно, второй вариант может быть отброшен, как бесперспективный. Тем самым отбрасываются и все получающиеся из него варианты дальнейшего движения (число которых при больших n может быть огромным).

Сведем результаты рассмотрения вариантов в табл. 2.

Далее развиваются и сравниваются только перспективные варианты (в нашем случае 12). Развитие каждого из них однозначно (поскольку для каждого из них остается лишь один непосещенный пункт).

Результаты сравнения вариантов, оканчивающихся в одном и том же пункте, приведены в табл. 3.

---

* Расстояния определяются для конкретных путей сообщения не обязательно по прямой.

Из табл. 3 выбираем четыре перспективных варианта. Дальнейшее их развитие заключается в возвращении в исходный пункт A_0. Для этих четырех окончательных маршрутов вычисляем суммарные расстояния и записываем в табл. 4.

Таблица 2

Варианты	Расстояние, *км*	Перспективно или нет
$A_0A_2A_3A_1$	850	Да
$A_0A_3A_2A_1$	950	Нет
$A_0A_2A_4A_1$	1050	Да
$A_0A_4A_2A_1$	1100	Нет
$A_0A_3A_4A_1$	1050	Нет
$A_0A_4A_3A_1$	1000	Да
$A_0A_1A_3A_2$	900	Да
$A_0A_3A_1A_2$	1050	Нет
$A_0A_1A_4A_2$	1100	Да
$A_0A_4A_1A_2$	1500	Нет
$A_0A_2A_4A_2$	650	Да
$A_0A_4A_3A_2$	900	Нет
$A_0A_1A_2A_3$	1050	Да
$A_0A_2A_1A_3$	1100	Нет
$A_0A_1A_4A_3$	1150	Да
$A_0A_4A_1A_3$	1350	Нет
$A_0A_2A_4A_3$	700	Да
$A_0A_4A_2A_3$	850	Нет
$A_0A_1A_2A_4$	1000	Да
$A_0A_2A_1A_4$	1350	Нет
$A_0A_1A_3A_4$	900	Да
$A_0A_3A_1A_4$	1150	Нет
$A_0A_2A_3A_4$	750	Нет
$A_0A_3A_2A_4$	650	Да

Таблица 3

Варианты	Расстояние, *км*	Перспективно или нет
$A_0A_2A_3A_1A_4$	1450	Нет
$A_0A_2A_4A_1A_3$	1400	Нет
$A_0A_4A_3A_1A_2$	1500	Нет
$A_0A_1A_3A_2A_4$	1100	Да
$A_0A_1A_4A_2A_3$	1350	Нет
$A_0A_3A_4A_2A_1$	1150	Нет
$A_0A_1A_2A_3A_4$	1350	Нет
$A_0A_1A_4A_3A_2$	1400	Нет
$A_0A_2A_4A_3A_1$	1050	Да
$A_0A_1A_2A_4A_3$	1250	Да
$A_0A_1A_3A_4A_2$	1100	Да
$A_0A_3A_2A_4A_1$	1250	Нет

Таблица 4

Варианты	Расстояние, *км*
$A_0A_1A_2A_3A_4A_0$	1500
$A_0A_2A_4A_3A_1A_0$	1350
$A_0A_1A_2A_4A_3A_0$	1450
$A_0A_1A_3A_4A_2A_0$	1350

Таким образом, имеются два оптимальных маршрута следования коммивояжера $A_0A_1A_3A_4A_2A_0$ и $A_0A_2A_4A_3A_1A_0$, имеющие минимальную из всех возможных маршрутов длину, равную 1350 *км*.

В описанной процедуре в общем случае подвергается анализу

$$n(n-1)(n-2)+\frac{n(n-1)(n-2)(n-3)}{2}+$$
$$+\frac{n(n-1)(n-2)(n-3)(n-4)}{2\cdot 3}+\cdots$$

неполных вариантов (за исключением последних). Если бы процедура отсева начиналась с частичных вариантов, включающих

один участок (а не 3, как это имеет место в данном случае), то общее число исследуемых вариантов выразилось бы суммой

$$c_n^1 + c_n^2 + \cdots + c_n^{n-1} = 2^n - 2,$$

что при больших n значительно меньше, чем общее число всех вариантов, равное, как было уже отмечено выше, $n!$

К задаче коммивояжера сводятся многие важные задачи планирования. Пусть, например, речь идет о строительстве нескольких объектов $A_1, A_2, \ldots, A_n$ в условиях ограниченности ресурсов строительных организаций. Предположим, что строительство каждого объекта состоит из m последовательных стадий (земляные работы, закладка фундамента, кладка стен и т. д.), а мощность строительной организации не позволяет вести один и тот же вид работ одновременно на нескольких объектах. Продолжительности t_{ij} работ каждого вида для каждого объекта заданы ($i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, m$). Считая, что работа на каждом объекте должна продолжаться непрерывно с момента начала строительства до его окончания, требуется определить сроки начала строительства каждого объекта так, чтобы суммарный срок строительства всех объектов был бы минимальным.

Таблица 5

Объекты	Виды работ			
	1	2	3	4
A_1	1	3	4	2
A_2	3	2	3	1
A_3	2	2	1	4

Пусть, например, речь идет о строительстве трех объектов A_1, A_2, A_3 с четырьмя видами работ для каждого из них. Сроки (в месяцах) для каждой из работ задаются табл. 5.

Покажем, как оценивается суммарное время строительства для одного из вариантов, например $A_1A_2A_3$. Сроки окончания работ на первом объекте выражаются соответственно в следующих цифрах: для первой стадии — 1 мес., для второй — $1 + 3 = 4$ мес., для третьей — $4 + 4 = 8$ мес., для четвертой — $8 + 2 = 10$ мес. Время t_2 начала работ на втором объекте должно удовлетворять следующим очевидным неравенствам*:

$$t_2 \geqslant 1; \quad t_2 + 3 \geqslant 4; \quad t_2 + 5 \geqslant 8; \quad t_2 + 8 \geqslant 10.$$

Наименьшее значение t_2, удовлетворяющее этим неравенствам, равно 3. Поэтому самый ранний возможный срок начала второго

* Эти неравенства выражают требования, чтобы каждая из работ на объекте A_2, начиналась лишь после окончания работ соответствующих стадий на объекте A_1.

строительства есть 3 мес. после начала строительства объекта A_1, а сроки окончания соответствующих стадий строительства 6, 8, 11 и 12 мес. соответственно. Для срока t_3 начала строительства объекта A_3 имеем систему неравенств

$$t_3 \geqslant 6; \quad t_3 + 2 \geqslant 8; \quad t_3 + 4 \geqslant 11; \quad t_3 + 5 \geqslant 12,$$

что приводит к минимальному сроку $t_3 = 7$ мес. Сроки окончания отдельных стадий строительства объекта A_3 соответственно 9, 11, 12 и 16 мес.

Таким образом, для выбранной последовательности строительства объектов общее время строительства (совпадающее со сроком завершения работ на объекте A_3) равно 16 мес. Поскольку в рассмотренном примере имеется всего $3! = 6$ вариантов строительства, то выбор наилучшего варианта может быть осуществлен простым перебором всех вариантов. При большем количестве объектов выбор оптимальной последовательности может быть осуществлен точно таким же образом, как и в задаче о коммивояжере.

Рассмотренные примеры демонстрируют основной прием динамического программирования — нахождение *правил доминирования*, которые позволяют производить сравнение вариантов развития последовательностей и заблаговременное отсеивание бесперспективных вариантов. В ряде случаев в задачах динамического программирования удается получить столь сильные правила доминирования, что они определяют элементы оптимальной последовательности однозначно один за другим. В этом случае правила доминирования называют обычно *разрешающими правилами*.

Разрешающие правила (в случае их существования) обычно выводятся с помощью *принципа оптимальности Беллмана*. Суть принципа оптимальности состоит в следующем. Пусть критерий F (задаваемый формулой или алгоритмом), дающий числовую оценку качества варианта (последовательности) $A_n = a_{i_1} a_{i_2} \ldots a_{i_n}$, можно применять не только ко всей последовательности, но и к любому ее начальному отрезку $A_k = a_{i_1} a_{i_2} \ldots a_{i_k}$. Последовательность A_n, которой соответствует экстремальное значение критерия F, называется оптимальной. Если любой начальный отрезок оптимальной последовательности также оптимален (в классе всех последовательностей, составленных из тех же элементов и, быть может, еще имеющий те же начало и конец, что и данный отрезок), то говорят, что *для соответствующей задачи справедлив принцип оптимальности*.

Принцип оптимальности, как нетрудно понять, всегда выполняется для *аддитивных* критериев F, характеризующихся тем свойством, что для любой последовательности A, составленной из двух подпоследовательностей B и C, $(A = BC)$,

$$F(A) = F(BC) = F(B) + F(C).$$

Рассмотрим один из многочисленных примеров задач динамического программирования, для которых выполняется принцип оптимальности. Предположим, что нам необходимо обработать последовательно n различных деталей $d_1, d_2, \ldots, d_n$. Время, требуемое для обработки детали d_i обозначим через t_i, а стоимость хранения этой детали в единицу времени — через ρ_i $(i = 1, 2, \ldots, n)$. Предполагая, что обработку можно производить в любом порядке, и пренебрегая временем переналадки оборудования, можно найти такой порядок обработки, при котором общая стоимость F хранения деталей (до начала их обработки) минимальна.

Справедливость принципа оптимальности в данном случае очевидна: если бы какой-то начальный отрезок оптимальной последовательности $d_{i_1}d_{i_2}\ldots d_{i_k}$ был бы неоптимальным, то перестановкой его элементов, приводящей его к оптимальному виду, можно было бы, очевидно, уменьшить значение критерия F для всей последовательности.

Пусть теперь $D = d_{i_1}d_{i_2}\ldots d_{i_k}$ — любой начальный отрезок оптимальной последовательности $(k \geqslant 2)$, а D' — последовательность, полученная из D перестановкой двух его последних элементов. Тогда по принципу оптимальности

$$F(D) \leqslant F(D'). \tag{11а}$$

Обозначим T_j сумму $t_{i_1} + t_{i_2} + \ldots + t_{i_j}$. Тогда

$$F(D) = \rho_{i_2}T_1 + \rho_{i_3}T_2 + \cdots + \rho_{i_{k-1}}T_{k-2} + \rho_{i_k}(T_{k-2} + t_{i_{k-1}});$$
$$F(D') = \rho_{i_2}T_1 + \rho_{i_3}T_2 + \cdots + \rho_{i_k}T_{k-2} + \rho_{i_{k-1}}(T_{k-2} + t_{i_k}).$$

Подставляя эти значения в неравенство (11 а) и вычеркивая одинаковые члены в левой и правой частях, приходим к искомому разрешающему правилу:

$$\rho_{i_k}t_{i_{k-1}} \leqslant \rho_{i_{k-1}}t_{i_k}.$$

Или, что то же самое

$$\frac{t_{i_{k-1}}}{\rho_{i_{k-1}}} \leqslant \frac{t_{i_k}}{\rho_{i_k}}.$$

Из этого правила следует, что оптимальная последовательность предполагает обработку деталей в порядке роста (точнее, неубывания) отношения времени обработки t_i к стоимости ρ_i ее хранения в единицу времени: первой должна обрабатываться деталь с минимальной величиной отношения $\frac{t_i}{\rho_i}$, следующей обрабатывается деталь с минимальным отношением $\frac{t_j}{\rho_j}$ среди оставшихся деталей и т. п.

Любопытно, что при интуитивном подходе к решению данной задачи обычно предлагается обработка в порядке убывания величины удельной стоимости ρ_i хранения деталей, что, как видно из приведенного рассмотрения, не приводит, вообще говоря, к оптимальной последовательности.

Рассмотрим еще один пример задачи динамического программирования, возникающей при проектировании дорог, трубопроводов, линий электропередач и других протяженных объектов.

Пусть, например, нам требуется проложить кабель из пункта A в пункт B (рис. 15) так, чтобы он прошел через один из пунктов C_i, один из пунктов D_j и т. д. Между любой парой соседних пунктов (A и C_i, C_i и D_j и т. д.) возможен лишь один вариант прокладки (например, по прямой). Требуется определить вариант прокладки, имеющий наименьшую стоимость.

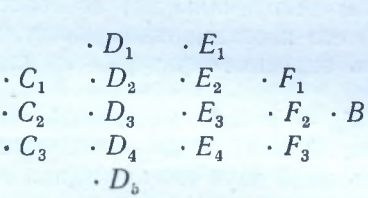

Рис. 15.
Схема трассировки.

Как и в предыдущем случае, легко понять, что в данной задаче имеет место принцип оптимальности. Для первого отрезка AC_i вычисляем стоимости всех вариантов: AC_1, AC_2, AC_3. Каждый из полученных отрезков продолжаем во все D_j и для всех D_j фиксируем варианты траекторий AC_iD_j, имеющие минимальную стоимость (если искать не все, а лишь одну оптимальную траекторию, то таких вариантов будет пять). Далее, продолжаем каждую из этих траекторий во все E_k и оставляем для каждого E_k вариант с наименьшей стоимостью (таких вариантов будет четыре). Повторяем ту же процедуру с точками F_r и т. д., пока не достигнем точки B. В нашем случае это приведет к анализу $3 + 3 \cdot 5 + 5 \cdot 4 + 4 \cdot 3 + 3 \cdot 1 = 53$ траекторий (из которых только три полные), тогда как общее число всех возможных полных траекторий равно, очевидно, $3 \cdot 5 \cdot 4 \cdot 3 = 180$.

Если число точек в каждом сечении (C_i, D_j и т. д.) одинаково и равно m, а число сечений (не считая A и B) есть n, то при применении описанного приема анализируется $m + (n - 1)m^2 + m$ траекторий, а их общее число равно, как нетрудно видеть, m^n. При больших m и n перебор значительно сокращается и, что не менее важно, при развитии траекторий на каждом шаге необходимо хранить ограниченное число траекторий (не более $2m$), что резко ограничивает требования к объему памяти при решении задачи на ЭВМ.

Необходимость же иметь большие m и n вытекает из того, что задачи описанного класса возникают обычно в результате дискрет-

ной аппроксимации непрерывных задач: в исходной задаче допустимы любые криволинейные траектории, соединяющие точки A и B, мы же заменяем (аппроксимируем) эти кривые ломаными, проходящими через вершины достаточно густой (для обеспечения хорошей аппроксимации) сетки.

Число подлежащих рассмотрению траекторий может быть еще более уменьшено, если к правилам доминирования, вытекающим из принципа оптимальности, предъявляются дополнительные ограничения, вытекающие из конкретного смысла задачи (например, отсутствие резких изломов траектории, ее обязательное прохождение через заданные промежуточные пункты и т. п.).

Количество элементарных арифметических операций, которые необходимо выполнить при решении задач динамического программирования, определяется помимо количества просматриваемых вариантов также сложностью вычисления критерия F для каждого из оцениваемых вариантов. Эта сложность может сильно варьироваться.

4. Сетевые графики

Сетевые графики представляют собой математический аппарат для представления, изучения и управления сложными комплексами взаимосвязанных работ, направленных к достижению относительно небольшого числа четко определенных целей. Наиболее простыми являются одноцелевые сетевые графики, с которых мы и начнем наше рассмотрение. Целью здесь может являться окончание строительства того или иного объекта, создание уникального изделия, выполнение научно-исследовательской или опытно-конструкторской работы, завершение реконструкции или ремонта и т. п.

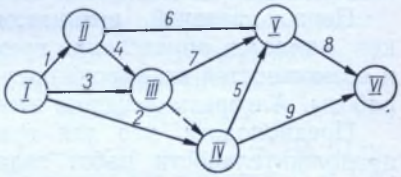

Рис. 16.
Одноцелевой сетевой график.

Простейший пример одноцелевого сетевого графика изображен на рис. 16. Сетевой график состоит из объектов двух видов: узлов (обозначенных кружками) и соединяющих их (направленных) ребер (обозначенных стрелками).

Каждому узлу соответствует некоторое событие, заключающееся в окончании того или иного этапа работ, например, «закладка фундамента окончена», «технический проект принят комиссией» и т. д. Каждой стрелке (ребру графика) соответствует та или иная работа, понимаемая как процесс, а не как результат, например, процесс сооружения стен, процесс оформления эскизного проекта

и т. д. Для каждой работы задается ее продолжительность, измеряемая в фиксированных для данного графика единицах (часы, дни, недели или месяцы).

Смысл графика состоит прежде всего в том, чтобы указать все технологические связи, определяющие возможные последовательности работ. Из рис. 16 видно, что работа *6* может быть начата лишь после окончания работы *1*, а работа *7* — лишь после окончания работ *3* и *4*. В ряде случаев для задания связей приходится пользоваться так называемыми фиктивными работами, имеющими нулевую продолжительность и обозначаемыми на графике пунктирными стрелками.

Роль фиктивных работ легко уяснить из рис. 16. Работа *7* может выполняться после окончания работ *3* и *4*, а работы *5* и *9* — после окончания работ *2*, *3* и *4*. Если бы мы объединили узлы *III* и *IV*, то работе *7* пришлось бы ждать окончания не только работ *3* и *4*, но и работы *2*. При наличии фиктивной работы необходимость такого ожидания исключается.

На любом одноцелевом сетевом графике выделяются два события — *начальное* и *конечное*. Начальное событие (в нашем случае событие *I*) характеризуется тем, что в него не входит ни одна стрелка, а конечное (в данном случае событие *VI*) тем, что из него не выходит ни одна стрелка. Все остальные события носят название *промежуточных*. Начальное событие соответствует началу работ (нулевой момент времени), а конечное — завершению всех работ (достижению поставленной цели).

Первой задачей, возникающей при анализе сетевых графиков, является определение времени наступления каждого события и возможностей варьирования времени начала и окончания каждой работы. Алгоритм решения этой задачи весьма прост.

Предположим, что для графика, изображенного на рис. 16, продолжительности работ заданы следующими числами: $t_1 = 3$; $t_2 = 7$; $t_3 = 5$; $t_4 = 3$; $t_5 = 5$; $t_6 = 6$; $t_7 = 3$; $t_8 = 2$; $t_9 = 6$. Событию *I* соответствует момент времени $T_1 = 0$.

Чтобы найти минимальный срок T_i наступления любого другого события ($i \geqslant 2$), необходимо, очевидно, просчитать суммарные затраты времени по всем путям, ведущим в соответствующий этому событию узел из начального узла, и выбрать из них максимальное значение. Всякий раз, когда для какого-то события срок его наступления установлен, можно строить дальнейшие пути, отправляясь от этого события, как от начального. Для события *II*, к которому ведет только один путь (работа *1*), получим

$$T_2 = T_1 + t_1 = 0 + 3 = 3.$$

Для события *III* имеем два пути. Учитывая лишь работу *3*,

получим первое (не окончательное) значение для времени T_3 наступления этого события:

$$T_3^{(3)} = T_1 + t_3 = 0 + 5 = 5.$$

Соответствующее вычисление по работе *4* приводит к значению:

$$T_3^{(4)} = T_2 + t_4 = 3 + 3 = 6.$$

Наибольшее из этих двух значений и дает как раз искомый срок наступления события *III*: $T_3 = 6$.

Для события *IV*

$$T_4^2 = T_1 + t_2 = 0 + 7 = 7; \quad T_4^{(\text{фикт})} = T_3 + 0 = 6 + 0 = 6.$$

Отсюда $T_4 = 7$. Для события *V*

$$T_5^{(6)} = T_2 + t_6 = 3 + 6 = 9; \quad T_5^{(7)} = T_3 + t_7 = 6 + 3 = 9;$$
$$T_5^{(5)} = T_4 + t_5 = 7 + 5 = 12.$$

Таким образом, $T_5 = 12$. Наконец,

$$T_6^{(8)} = T_5 + t_8 = 12 + 2 = 14; \quad T_6^{(9)} = T_4 + t_9 = 7 + 6 = 13.$$

Откуда $T_6 = 14$.

Получив минимальный срок T_6 наступления конечного события (а значит, и время окончания всех работ), мы можем, отправляясь от него и выполняя описанную процедуру в обратном порядке, вычислить максимальные возможные сроки $\overline{T}_i$ наступления всех событий графика, исходя из необходимости закончить все работы в вычисленный срок. Для конечного события полагаем $\overline{T}_6 = T_6 = 14$. Для события *V* имеем $\overline{T}_5 = T_6 - t_8 = 14 - 2 = 12$. Для события *IV* имеем две возможности (двигаясь от событий *V* и *VI* в направлении, обратном стрелкам):

$$\overline{T}_4^{(5)} = \overline{T}_5 - t_5 = 7 \text{ и } \overline{T}_4^{(9)} = \overline{T}_6 - t_9 = 14 - 6 = 8.$$

Из полученных чисел необходимо, очевидно, выбрать минимальное, т. е. $\overline{T}_4 = 7$. Для события *III* получаем:

$$\overline{T}_3^{(7)} = \overline{T}_5 - t_7 = 12 - 3 = 9; \quad \overline{T}_3^{(\text{фикт})} = 7.$$

Таким образом, $\overline{T}_3 = 7$. Для события *II*:

$$\overline{T}_2^{(6)} = \overline{T}_5 - t_6 = 12 - 6 = 6; \quad T_2^{(4)} = \overline{T}_3 - t_4 = 7 - 3 = 4.$$

Откуда $T_2 = 4$.

При правильном вычислении время наступления начального события всегда должно быть равно 0. Проверяем:

$$\overline{T}_1^{(1)} = \overline{T}_2 - t_1 = 1; \quad \overline{T}_1^{(2)} = \overline{T}_4 - t_2 = 7 - 7 = 0;$$
$$\overline{T}_1^{(3)} = \overline{T}_3 - t_3 = 2.$$

Минимальное значение равно 0. Таким образом, $\overline{T}_1 = 0$, что подтверждает (хотя и не с абсолютной гарантией) правильность проведенных нами вычислений.

Таблица 6

Сроки	События					
	I	*II*	*III*	*IV*	*V*	*VI*
T_i	0	3	6	7	12	14
$\overline{T}_i$	0	4	7	7	12	14

Сведем полученные значения в табл. 6. Из этой таблицы видно, что для событий *II* и *III* допустимо опоздание на единицу времени без изменения общего срока выполнения работ. Для событий *I*, *IV*, *V*, *VI* никакие опоздания недопустимы; любая задержка в их наступлении влечет за собой увеличение общего срока всех работ. Про события, обладающие таким свойством, принято говорить, что они находятся на *критическом пути*.

Для нахождения критического пути для каждой j-й работы вычислим минимально и максимально возможные сроки начала и окончания работы:

$$T_{j\,\text{нач}}^{\min} = T_p;$$
$$T_{j\,\text{нач}}^{\max} = \overline{T}_q - t_j;$$
$$T_{j\,\text{кон}}^{\min} = T_p + t_j;$$
$$T_{j\,\text{кон}}^{\max} = \overline{T}_q,$$

где p — событие, стоящее в начале работы j; q — в ее конце. Результаты расчетов сводим в табл. 7.

Разность $\tau_j = T_{j\,\text{нач}}^{\max} - T_{j\,\text{нач}}^{\min} = T_{j\,\text{кон}}^{\max} - T_{j\,\text{кон}}^{\min} = \overline{T}_q - T_p - t_j$ будем называть *резервом времени* для j-й работы. Работы, у которых резерв времени равен нулю, составляют *критический путь*. В нашем случае критический путь составляют работы *2*, *5* и *8*. Остальные работы имеют меньшие или большие *ненулевые* резервы времени. Разумеется, при использовании сетевого графика в качестве инструмента управления (а не только в качестве средства для начального анализа), необходимо осуществлять регулярное обновление информации (как об ожидаемой продолжительности работ, так и о фактическом их состоянии) и каждый раз

Таблица 7

Срок	Виды работ								
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
$T_{j\,\text{нач}}^{\min}$	0	0	0	3	7	3	6	12	7
$T_{j\,\text{нач}}^{\max}$	1	0	2	4	7	6	9	12	8
$T_{j\,\text{кон}}^{\min}$	3	7	5	6	12	9	9	14	13
$T_{j\,\text{кон}}^{\max}$	4	7	7	7	12	12	12	14	14
t_j	3	7	5	3	5	6	3	2	6

заново вычислять все перечисленные выше показатели. При этом может меняться не только ожидаемое время наступления тех или иных событий (в том числе и конечного), но и критический путь. Основное преимущество управления на основе сетевого (а не обычного) графика состоит в том, что внимание руководства может сосредоточиваться на тех работах, которые являются решающими с точки зрения сроков окончания всех работ.

На этапе предварительного планирования определение критического пути сопровождается анализом возможностей сокращения продолжительностей находящихся на этом пути работ. Учет этих возможностей с соответствующим пересчетом всего графика определяет новый критический путь, для которого вновь ищутся возможности сокращения сроков и так далее, пока не будут исчерпаны все возможности дальнейшего улучшения графика. Эта работа должна продолжаться постоянно и в процессе фактического выполнения работ, поскольку здесь могут открываться новые возможности улучшения графика, недоступные на этапе предварительного планирования, либо могут возникать новые задачи по улучшению графика в связи с изменениями критического пути.

При переходе от одноцелевых сетевых графиков к многоцелевым описанная выше методика временного анализа по существу не меняется. Разница состоит лишь в том, что максимально возможные сроки $\overline{T}_i$ наступления событий должны вычисляться для каждой из поставленных целей. Самый поздний из этих сроков определяет время решения задачи, т. е. достижение всех конечных целей.

Рассмотрим двухцелевой сетевой график, изображенный на рис. 17, для которого продолжительности работ заданы табл. 8.

Таблица 8

Виды работ	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*
Срок (t_i)	4	3	4	2	3	4	4	6	5	2	4

Действуя точно таким же образом, как и в предыдущем примере, определим минимальные сроки наступления всех событий (табл. 9).

Для определения максимально возможных сроков $\overline{T}_i$ наступления событий, полагаем

$$\overline{T}_7 = T_7 = 12 \quad \text{и} \quad \overline{T}_8 = T_8 = 14.$$

Обозначим через $\overline{T}_i^{(VII)}$ и $\overline{T}_i^{(VIII)}$ максимально возможные сроки наступления события i с точки зрения задач достижения отдельно взятых целей *VII* и *VIII* соответственно.

Таблица 9

События	*I*	*II*	*III*	*IV*	*V*	*VI*	*VII*	*VIII*
Срок (T_i)	0	4	3	8	6	7	12	14

Таблица 10

События	*I*	*II*	*III*	*IV*	*V*	*VI*	*VII*	*VIII*
Срок ($\overline{T}_i$)	0	4	4	8	7	10	12	14

Тогда $\overline{T}_6^{(VII)} = \infty$, поскольку достижение цели *VII* не требует наступления события *VI*. С другой стороны,

$$\overline{T}_6^{(VIII)} = \overline{T}_8 - t_{11} = 10.$$

Максимально возможный срок наступления события *VI* с точки зрения достижения обеих целей:

$$\overline{T}_6 = \min(\infty,\ 10) = 10.$$

Для события *V* получаем:

$$\overline{T}_5^{(VII)} = \overline{T}_7 - t_9 = 12 - 5 = 7; \qquad \overline{T}_5^{(VIII)} = \overline{T}_8 - t_{10} = 12.$$

Тогда $\overline{T}_5 = \min(7,\ 12) = 7$. Продолжая таким же образом, находим сроки $\overline{T}_i$ для всех событий (табл. 10).

Для вычисления минимально и максимально возможных сроков начала и окончания работ, можно пользоваться теми же формулами, что и раньше, если речь идет о достижении обеих целей. Заменяя в формулах $\overline{T}_i$ на $\overline{T}_i^{(VII)}$ и $\overline{T}_i^{(VIII)}$, находим соответствующие сроки с точки зрения задач достижения каждой из целей *VII* и *VIII* в отдельности:

Например,

$$T_{4нач}^{min} = T_2 = 4; \quad T_{4нач}^{max} = \overline{T}_5 - t_4 = 5; \quad T_{4нач}^{(VII)max} =$$
$$= \overline{T}_5^{(VII)} - t_4 = 7 - 2 = 5; \quad T_{4нач}^{(VIII)max} = \overline{T}_5^{(VIII)} - t_4 = 12 - 2 = 10.$$

Соответственно

$$T_{4кон}^{min} = T_2 + t_4 = 3 + 2 = 5; \quad T_{4кон}^{max} = \overline{T}_5 = 7;$$
$$T_{4кон}^{(VII)max} = \overline{T}_5^{(VII)} = 7; \quad T_{4кон}^{(VIII)max} = \overline{T}_5^{(VIII)} = 12.$$

Аналогичным образом вычисляем сроки для всех остальных работ.

До сих пор при рассмотрении сетевых графиков мы имели дело лишь с проблемами, касающимися лишь сроков выполнения и продолжительностей работ. Гораздо более трудными являются проблемы, связанные с распределением ресурсов при планировании и управлении на основе сетевых графиков. Такие проблемы сводятся к задачам нелинейного целочисленного программирования весьма большого объема, решение которых представляет большие трудности даже при использовании самых мощных ЭВМ. На практике поэтому ограничиваются упрощенными постановками задачи и частичной оптимизацией, без гарантии достижения абсолютного оптимума.

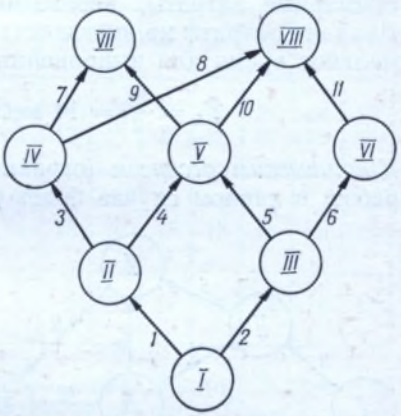

Рис. 17.
Двухцелевой сетевой график.

Для лучшего уяснения сущности проблем, встающих в связи с оптимизацией распределения ресурсов при планировании и управлении на основе сетевых графиков, рассмотрим следующий пример. Пусть совокупность подлежащих выполнению работ описывается сетевым графиком, изображенным на рис. 18. Предположим, что имеется всего два ресурса R_1 и R_2, которые подлежат распределению (рабочие двух специальностей или оборудование двух различных видов).

Таблица 11

Ресурсы	Виды работ												
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	Σ
R_1	2	4	6	2	0	2	2	2	0	2	2	4	28
R_2	3	3	0	3	3	0	3	3	3	6	3	6	36

На всем протяжении работ располагаем двумя единицами первого ресурса (например, двумя бригадами рабочих) и тремя единицами второго ресурса. Предположим, что время работ исчисляется месяцами, а затраты ресурсов (в ресурсо-месяцах) для работ *1*, *2*, ..., *12* приведены в табл. 11. В последнем столбце указаны суммарные затраты, необходимые для выполнения всех работ. Деля эти затраты на имеющиеся в наличии ресурсы, получим минимальное время для выполнения всех работ:

$$T_1 = \frac{28}{2} = 14 \text{ мес}; \quad T_2 = \frac{36}{3} = 12 \text{ мес}.$$

Критическим ресурсом (ограничивающим снизу срок выполнения работ) в данном случае будет ресурс R_1. Минимально возможное время выполнения всех работ $T \geqslant 14$ мес. Это время представляет собой теоретический предел (не всегда достижимый) для оптимизации плана выполнения работ.

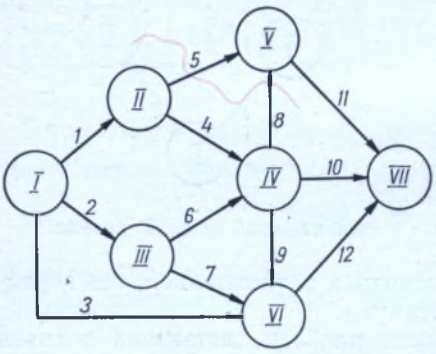

Рис. 18.
К пояснению оптимизации распределения ресурсов.

Составляем начальный план, при котором соблюдается технологическая последовательность выполнения работ, заданная рассматриваемым сетевым графиком. В нашем случае в качестве такого плана может быть, например, принято последовательное выполнение всех работ в порядке роста их номеров. При этом каждый раз все имеющиеся ресурсы полностью бросаются на выполнение одной работы и лишь после ее окончания перебрасываются на следующую.

Для первой работы как по первому, так и по второму ресурсу время t_1 ее выполнения одинаково и равно одному месяцу. Потерь (простоев) ресурсов при этом нет. При выполнении второй работы критическим является первый ресурс. Вычисленное по нему время исполнения работы t_2 равняется $\frac{4}{2} = 2$ мес. При работе в течение этого срока располагаем $3 \times 2 = 6$ ресурсо-месяцами по второму ресурсу. Поскольку фактически для выполнения второй работы требуется лишь 3 ресурсо-месяца, то по ресурсу R_2 образуется потеря, равная $6 - 3 = 3$ ресурсо-месяцам. Продолжая аналогичным образом, сведем время, а также суммарные затраты и потери ресурсов при выбранном порядке выполнения работ нарастающим итогом (т. е. в виде сумм по всем выполненным к рассматриваемому моменту работам) в табл. 12.

Таблица 12

Показатели	Виды работ											
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Затраты: R_1	2	6	12	14	14	16	18	20	20	22	24	28
R_2	3	6	6	9	12	12	15	18	21	27	30	36
Потери: R_1	0	0	0	0	2	2	2	2	4	6	6	6
R_2	0	3	12	12	12	15	15	15	15	15	15	15
Время	1	3	6	7	8	9	10	11	12	14	15	17

Очевидный недостаток построенного плана состоит в том, что происходят потери критического (первого) ресурса. Именно в результате этих потерь полученный срок выполнения работ (17 мес.) на 3 мес. больше, чем теоретически допустимый предел.

Предположим, что ресурсы R_1 и R_2 допускают независимое использование и при выполнении любой работы могут расходоваться в любом порядке*. Основная идея по улучшению начального плана при таком предположении состоит в том, чтобы начинать использование освободившегося ресурса для выполнения последующих работ, не дожидаясь полного высвобождения всех

* Взаимозависимые ресурсы, например каменщики и подъемные механизмы, обычно объединяются в единый комплексный ресурс. Если работа допускает использование ресурсов только в одном порядке, то ее следует свести к нескольким работам, введя новые события на исходном сетевом графике.

ресурсов. Очевидно, что такая тактика приведет к успеху лишь в том случае, когда работа, в которой начинается использование освободившегося ресурса, является критической в отношении именно этого ресурса. Это достигается изменением порядка выполнения работ, разумеется, при строгом соблюдении условий очередности работ, вытекающих из исходного сетевого графика.

Для облегчения выбора надлежащей очередности выполнения работ разделим их на три группы: критичные по первому, второму и по обоим ресурсам. В первую группу попадают работы *2*, *3*, *6*, во вторую — работы *5*, *9*, *10* и в третью—работы *1*, *4*, *7*, *8*, *11*, *12*. Работы последней группы могут выполняться в любое время, допускаемое исходным сетевым графиком, не создавая нежелательных потерь ресурсов. Проблема состоит в определении правильной последовательности работ первой и второй группы.

Поскольку при выполнении второй работы образуется резерв 3 ресурсо-месяца по ресурсу R_2, то желательно, не дожидаясь окончания этой работы, начать критичную по R_2 работу, в которой этот ресурс желательно полностью использовать. Такой является, очевидно, работа *5*.

Для работы *3* такими могут быть работы *5*, *9* и *10*, а для работы *6*—*9* и *10*. Одним из возможных решений, дающих требуемую очередность, служит последовательность выполнения работ *1*, *2*, *5*, *4*, *7*, *6*, *9*, *3*, *10*, *8*, *11*, *12*, для которой все показатели приведены в табл. 13.

Таблица 13

Показатели	Виды работ											
	1	*2*	*5*	*4*	*7*	*6*	*9*	*3*	*10*	*8*	*11*	*12*
Время начала использования ресурса R_1	0	1	3	3	4	5	6	6	9	10	11	12
Время окончания использования ресурса R_1	1	3	3	4	5	6	6	9	10	11	12	14
Время начала использования ресурса R_2	0	1	2	3	4	5	5	6	6	10	11	12
Время окончания использования ресурса R_2	1	2	3	4	5	5	6	6	8	11	12	14
Время начала работы	0	1	2	3	4	5	5	6	6	10	11	12
Время окончания работы	1	3	3	4	5	6	6	9	10	11	12	14
Использование: R_1	2	6	6	8	10	12	12	18	20	22	24	28
R_2	3	6	9	12	15	15	18	18	24	27	30	36
Потери: R_1	0	0	0	0	0	0	0	0	0	0	0	0
R_2	0	0	0	0	0	0	0	3	6	6	6	6

В рассматриваемом случае удалось достичь теоретического предела для времени выполнения работ и полностью избежать потерь критического ресурса. Нетрудно понять, что, поскольку речь идет о нахождении надлежащей перестановки исходного порядка выполнения работ, то, вместо примененного приема угадывания нужного порядка, можно применить общий метод динамического программирования, аналогично тому, который был применен в задаче о коммивояжере. Отличие состоит лишь в том, что рассмотрению подлежат не все возможные подстановки, а только те из них, в которых последовательность работ удовлетворяет требованиям исходного сетевого графика.

В общем случае при оптимизации использования ресурсов приходится рассматривать возможности параллельного выполнения работ, что еще более усложняет задачу.

5. Теория массового обслуживания

В теории массового обслуживания рассматриваются задачи планирования и управления работ по удовлетворению потока требований, возникающих случайно и требующих различного, заранее точно не предсказуемого времени для их удовлетворения. Типичными производственными задачами такого рода являются работы по ремонту и наладке оборудования. Другие примеры задач: обслуживание покупателей в магазине, медицинское обслуживание, продажа билетов, организация информационно-справочной службы, телефонная и телеграфная связь и т. п.

Хотя каждое индивидуальное требование в системе массового обслуживания возникает случайно, потоки таких требований удовлетворяют тем или иным закономерностям статистического характера. Для простейшего пуассоновского потока, вероятность $d\rho$ появления требования в любой бесконечно малый промежуток времени dt пропорциональна (с точностью до бесконечно малых высших порядков) длине этого промежутка: $d\rho = \alpha dt$ и не зависит от того, возникали или нет требования в предшествующие моменты времени.

Величина α, называемая интенсивностью (или плотностью) потока, может быть постоянной или меняться во времени по определенному закону: $\alpha = \alpha(t)$. В первом случае поток называется стационарным, во втором — нестационарным. Термин *простейший поток* обычно применяется к стационарному пуассоновскому потоку.

Пуассоновские потоки описывают реально возникающие потоки требований во всех перечисленных выше примерах. В ряде случаев, однако, приходится рассматривать потоки более общего вида, например, часто встречающиеся на практике *неординарные* (груп-

повые) пуассоновские потоки. Моменты возникновения требований в таких потоках распределены как и в обычном, *ординарном* потоке, но в каждый такой момент возникает не одно, а целая группа требований. Величина этой группы n также случайна и описывается некоторым законом распределения, задающим тем или иным способом вероятности p_n того, что в группе содержится ровно n требований ($n = 1, 2, \ldots$). Характерными примерами такого потока является поток коллективных заявок на билеты, обслуживание организованных групп туристов и т. п.

Имеются и другие обобщения, связанные с тем, что возникновение требования может изменять (обычно уменьшать) вероятность возникновения новых требований в последующие промежутки времени.

Важной характеристикой любого потока требований является время T между моментом возникновения одного и непосредственно следующего за ним требования. Эта величина случайна и характеризуется некоторым законом распределения. Для пуассоновского потока с интенсивностью α вероятность того, что время T заключено в пределах от x до $x + dx$ с точностью до бесконечно малых высших порядков равно $\alpha e^{-\alpha x}\, dx$. Таким образом, вероятность $P\{T \leqslant A\}$ того, что $T \leqslant A$, равна

$$\int_0^A \alpha e^{-\alpha x} dx = 1 - e^{-\alpha A}.$$

Следовательно, вероятность того, что $T > A$ есть $e^{-\alpha A}$. Это так называемый *показательный* закон распределения. В непуассоновских потоках это распределение отлично от показательного.

Другой важной характеристикой потока является количество N требований, возникающих в заданный интервал времени длины t. Эта величина также случайна и для простейшего (стационарного пуассоновского) потока задается распределением

$$P\{N = k\} = -\frac{(\alpha t)^k}{k} e^{-\alpha t *}.$$

Это — так называемое *распределение Пуассона*, по имени которого и назван соответствующий поток.

Кроме перечисленных характеристик, описывающих моменты возникновения требований, важное значение для расчета систем массового обслуживания имеет также рассмотрение закономерностей распределения длительности обслуживания каждого требования. Обычно считается, что эта закономерность является

---

* $P\{N = k\}$ есть вероятность того, что в течение времени t будет получено k требований.

общей для всех требований. Наиболее часто рассматриваются два частных случая: когда длительность обслуживания постоянна и когда она имеет *показательное распределение*. Во втором случае вероятность $P\{\tau > B\}$ того, что длительность обслуживания τ превышает B, равна $e^{-\lambda B}$, где λ — некоторый постоянный коэффициент. Величина τ, обратная этому коэффициенту $\left(\sigma = \frac{1}{\lambda}\right)$, равна как нетрудно подсчитать, *средней длительности* обслуживания.

Показательное распределение достаточно хорошо описывает распределение длительностей телефонных разоворов и ряд других примеров. Для описания многих других случаев требуются различные обобщения этого распределения. Одним из таких обобщений является распределение Эрланга, предельными случаями которого являются показательное распределение и постоянная длительность обслуживания.

Все законы распределения, характеризующие поток требований, могут задаваться либо на основе априорных соображений общего характера, либо в результате специального экспериментального исследования соответствующего потока. Примером первого закона может служить телефонная станция, обслуживающая большое количество индивидуальных абонентов. Априорным соображением здесь является естественное предположение о независимости телефонных звонков различных абонентов и полной случайности момента, когда такой звонок последует. Из этих предположений следует, как доказывается в теории, что рассматриваемый поток будет пуассоновским. На долю экспериментального исследования остается лишь определение интенсивности α этого потока. Для стационарного потока эта величина может быть получена в результате определения числа N звонков в течение достаточно большого момента времени T и деления первой величины на вторую: $\alpha \approx \frac{N}{T}$. Если априорные соображения, позволяющие определить вид закона распределения, отсутствуют, или имеются основания сомневаться в их справедливости, необходимо более детальное экспериментальное исследование рассматриваемого потока. Для этой цели в течение достаточно длительного промежутка времени фиксируются моменты поступления требований и потребовавшиеся фактически длительности обслуживания. На основании полученных данных строятся экспериментальные кривые распределения, которые затем обычно аппроксимируются формулой (желательно из числа хорошо изученных в теории законов распределения).

Кроме закономерностей, характеризующих поток требований, для расчета систем массового обслуживания необходимо задаться той или иной *дисциплиной обслуживания*. Дисциплина обслуживания определяет порядок, в котором данная система обрабатывает посту-

пающие требования. В системе с потерями всякое требование, поступившее тогда, когда все средства системы заняты обслуживанием ранее поступивших требований, теряется. Так обстоит дело, например, для обычных АТС. В системе с очередью все вновь поступающие требования ждут своей очереди для обслуживания. Встречается и смешанный случай, когда при достижении некоторой длины очереди вновь поступающие требования теряются. Возможен случай отпугивающей очереди, когда при увеличении длины очереди возрастает вероятность того, что вновь поступившее требование не встанет в очередь и потеряется.

В понятие дисциплины обслуживания входит также порядок обработки поступивших требований. На практике чаще всего встречается случай обработки в порядке поступления. Возможны также случаи, когда требования разделяются по определенным приоритетам, и обслуживание производится в порядке приоритетности требований*. Встречаются и такие системы, обработка в которых производится по принципу «последнее требование обслуживается первым».

Основным элементом дисциплины обслуживания является порядок, в котором система выделяет средства обслуживания. Этот вопрос возникает в кратных системах, когда любое из поступивших требований может быть обслужено одним из нескольких приборов, имеющих общее назначение (например, парикмахерская). В этом случае возможны три основные дисциплины обслуживания. Во-первых,— циклическое обслуживание, когда первое требование поступает на первый прибор, второе — на второй и т. д. Этот вариант наиболее прост для математического анализа, но относительно редко встречается на практике. Более употребительны два других варианта. В первом из них образуются различные очереди к разным приборам, и в момент поступления нового требования принимается решение, к какой из очередей его присоединить. Во втором варианте имеется одна общая очередь, из которой требования поступают на обслуживание к первому из освободившихся приборов.

Задача расчета системы массового обслуживания состоит в том, чтобы определить законы распределения и средние значения различных (случайных) величин, связанных с этой системой. К числу таких величин относятся длина очереди, время ожидания обслуживания, время занятости приборов и т. п.

Существуют два основных метода решения указанной задачи.

1. **Метод математического моделирования.** Сущность его состоит в том, что специальная программа, поставленная на ЭВМ, повто-

* Обычно при этом не происходит прерывания уже начатой обработки. Однако в так называемых системах с абсолютными приоритетами любое требование высшего приоритета прерывает обслуживание требований более низкого приоритета.

ряет шаг за шагом все действия рассматриваемой системы массового обслуживания: определяет случайным образом в соответствии с заданным законом распределения моменты появления требований и длительности их обработки, в соответствии с заданной дисциплиной обслуживания производит постановку на очередь, выделяет обслуживающие приборы и т. п. При этом специальная часть программы все время ведет подсчет интересующих величин (длин очередей, простоев обслуживающих приборов и т. п.) и строит экспериментальные кривые распределения вероятностей (частот) — гистограммы. По полученным гистограммам вычисляются средние значения, дисперсии, среднеквадратичные отклонения и другие характеристики экспериментальных распределений. Например, пусть было проделано сто вычислений длины очереди. Откладывая по горизонтали различные значения длины, а по вертикали частоту, т. е. количество раз, в течение которых длина очереди принимала заданные значения, получаем некоторую гистограмму (рис. 19). Из этой гистограммы видно, что длина очереди, равная нулю, встречалась 10 раз, а длина очереди, равная трем — 25 раз, что соответствует экспериментальным вероятностям

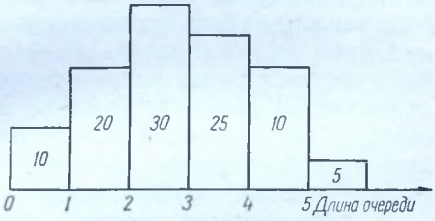

Рис. 19.
Гистограмма.

$$P_0 = \frac{10}{100} = 0{,}1 \quad \text{и} \quad P_3 = \frac{25}{100} = 0{,}25.$$

Среднее значение длины очереди

$$n_0 = 0P_0 + 1P_1 + 2P_2 + 3P_3 + 4P_4 + 5P_5 =$$
$$= 0 + 0{,}2 + 0{,}6 + 0{,}75 + 0{,}4 + 0{,}25 = 2{,}2.$$

Наиболее вероятная длина очереди равна, очевидно, 2. Дисперсия

$$d = \sum_{i=0}^{5} (i - n_0)^2 P_i = 2{,}2^2 \cdot 0{,}1 + 1{,}2^2 \cdot 0{,}2 +$$
$$+ 0{,}2^2 \cdot 0{,}3 + 0{,}8^2 \cdot 0{,}25 + 1{,}8^2 \cdot 0{,}1 + 2{,}8^2 \cdot 0{,}05 = 1{,}307.$$

А среднеквадратичное отклонение

$$\sigma = \sqrt{d} = 1{,}14.$$

Для облегчения построения моделирующих программ используются алгоритмические языки, ориентированные специально на

задачи моделирования (Симула, Симскрипт, Слэнг и др.). ЭВМ снабжается специальным устройством или программой, генерирующей случайные числа, обычно числа заданного отрезка (например, [0,1]), с равной вероятностью выбора любого такого числа. Специальные операторы языка моделирования могут переводить получаемое таким образом равномерное (равновероятное) распределение в распределения по любым другим употребляющимся на практике законам, например, в показательное распределение, распределение Пуассона и т. п. Языки моделирования обычно располагают также операторами для регистрации очередей, построения гистограмм и т. п.

Методом математического моделирования в принципе могут быть решены любые задачи по расчету систем массового обслуживания, в том числе и сложных систем с переменной (зависящей от состояния системы) дисциплиной обслуживания. Отрицательной чертой этого метода являются большие затраты машинного времени, поскольку для получения устойчивых статистических характеристик исследуемой системы приходится производить большое число (многие тысячи) экспериментов.

2. **Аналитические методы** позволяют представить интересующие нас характеристики в виде функций от исходных характеристик. К сожалению, простые аналитические (формальные) выражения для такого рода функций удается получать лишь в простейших случаях.

Рассмотрим пример аналитического расчета одной из простейших систем массового обслуживания. Пусть поток исходных требований является стационарным пуассоновским с интенсивностью α. Длительность обслуживания задана показательным законом:

$$P\{T > B\} = e^{-\lambda B}.$$

Дисциплина обслуживания предусматривает наличие очереди, обрабатываемой в порядке поступления требований. Искомые величины: длина очереди n (случайная величина) и коэффициент полезного действия η, т. е. среднее относительное время, в течение которого система занята обслуживанием.

Строго говоря, для ответа на указанные вопросы несбходимо, кроме уже приведенных данных, знать состояние системы (т. е. длину очереди и признак, занята ли она обслуживанием или нет) в начальный момент времени. Однако в теории для широкого класса систем массового обслуживания, заведомо включающего рассматриваемую систему, доказывается наличие у них так называемого *свойства эргодичности*. Сущность этого свойства состоит в том, что для моментов времени, достаточно удаленных от начального, влиянием начального состояния на состояние системы в рассматриваемый момент можно пренебречь. Для любой эргодичной системы

обычно ищется устойчивое (случайное) решение, к которому она стремится при $t \to \infty$. Расчет переходного процесса, в течение которого система переходит (с некоторой заданной точностью) в устойчивое состояние, представляет собой самостоятельную (и, как правило, гораздо более трудную) задачу.

Переходя к решению поставленной задачи, заметим прежде всего, что состояние системы можно охарактеризовать лишь одним числовым показателем — длиной очереди n, если считать, что случай $n = 0$ соответствует нулевой очереди при занятости системы, а случай $n = -1$ — нулевой очереди при свободной (не занятой обработкой) системе. Заметим далее, что, если система в течение времени t_0 была занята обработкой некоторого требования, то вероятность окончания обработки в промежуток $[t_0, t_0 + \Delta t]$ есть условная вероятность

$$\frac{P\{t_0 \leqslant t \leqslant t_0 + \Delta t\}}{P\{t > t_0\}}$$

Для показательного распределения $e^{-\lambda t}$ величина в числителе равна (с точностью до бесконечно малых более высокого, чем Δt, порядка) $\lambda e^{-\lambda t_0} \Delta t$, а в знаменателе — $e^{-\lambda t_0}$. Таким образом, вероятность окончания уже начатого обслуживания в любой бесконечно малый промежуток времени равна $\lambda \Delta t$ и не зависит от времени начала обслуживания.

Обозначим $p_i(t)$ вероятность того, что рассматриваемая система в момент времени t находится в i-ом состоянии ($i = -1, 0, 1, 2, \ldots$). Вычислим те же вероятности в близкий момент времени $t + \Delta t$. Для того чтобы система в момент времени $t + \Delta t$ оказалась в состоянии -1 имеется две возможности. Первая состоит в том, что система была в этом состоянии (-1) в момент времени t и в промежутке от t до $t + \Delta t$ не приняла ни одного нового требования. Вероятность такой возможности*

$$p_{-1}(t)(1 - \alpha \Delta t),$$

поскольку вероятность возникновения нового требования за элементарное время Δt равна для рассматриваемого потока $\alpha \Delta t$.

Вторая возможность состоит в том, что в момент времени t система находилась в состоянии 0, а за последующий отрезок времени Δt закончилось обслуживание обрабатывавшегося требования и не поступило нового требования. Вероятность такой возможности

$$p_0(t)\lambda \Delta t(1 - \alpha \Delta t) \approx p_0(t)\lambda \Delta t.$$

---

* Все вычисления производятся с точностью до бесконечно малых величин более высокого, чем Δt, порядка.

Объединяя обе эти (независимые) возможности, получаем соотношение:

$$p_{-1}(t+\Delta t) = p_{-1}(t)(1-\alpha\Delta t) + p_0(t)\lambda\Delta t,$$

или

$$\frac{p_{-1}(t+\Delta t) - p_{-1}(t)}{\Delta t} = \lambda p_0(t) - \alpha p_{-1}(t).$$

Переходя к пределу при $\Delta t \to 0$, находим дифференциальное уравнение:

$$\frac{dp_{-1}(t)}{dt} = \lambda p_0(t) - \alpha p_{-1}(t). \qquad (12)$$

Для состояния $i \neq -1$ рассмотрим три случая.

1. В момент времени t состояние системы было $i-1$, а за время Δt пришло новое требование и не закончилась обработка старого. Вероятность этой возможности

$$p_{i-1}(t)\,\alpha\Delta t\,(1-\lambda\Delta t) \approx p_{i-1}(t)\,\alpha\Delta t.$$

2. В момент времени t система находилась в состоянии $i+1$, за время Δt закончилась обработка старого требования, а новое требование не пришло. Вероятность

$$p_{i+1}(t)\,\lambda\Delta t\,(1-\alpha\Delta t) \approx p_{i+1}(t)\,\lambda\Delta t.$$

3. В момент времени t система была в состоянии i, а за время Δt не закончилась обработка старого требования и не пришло ни одного требования. Вероятность

$$p_i(t)(1-\alpha\Delta t)(1-\lambda\Delta t) \approx p_i(t) - (\alpha+\lambda)\,p_i(t)\,\Delta t.$$

Существует и четвертый случай, когда система сохраняет состояние i, закончив за время Δt обработку старого требования и получив за это время новое требование. Однако вероятность такой возможности

$$p_i(t)\,\alpha\Delta t\,\lambda\Delta t$$

есть бесконечно малая величина более высокого, чем Δt, порядка и должна быть поэтому отброшена.

Объединяя три случая, получаем дифференциальное уравнение:

$$\frac{dp_i(t)}{dt} = \alpha p_{i-1}(t) - (\alpha+\lambda)\,p_i(t) + \lambda p_{i+1}(t). \qquad (13)$$

Решение полученной (бесконечной) системы дифференциальных уравнений (12) и (13) полностью определяет (при задании начального состояния) дальнейшее поведение системы. Пользуясь свой-

ством эргодичности системы (заведомо справедливому при $\alpha < \lambda$), будем искать лишь установившееся решение, для которого, как нетрудно понять, $p_i = \text{const}$ и производные $\frac{dp_i(t)}{dt} (i = -1, 0, 1, 2, \ldots)$ должны обращаться в нуль. В результате имеем систему обычных алгебраических уравнений:

$$\lambda p_0 - \alpha p_{-1} = 0;$$
$$\alpha p_{i-1} - (\alpha + \lambda) p_i + \lambda p_{i+1} = 0 \quad (i = 0, 1, 2, \ldots). \tag{14}$$

Эта система позволяет по p_{-1} найти p_0, по этим двум величинам — p_1, по величинам p_0 и p_1 — величину p_2 и т. д. Если обозначить через r величину $\frac{\alpha}{\lambda}$, то решение системы (14) запишется в виде:

$$p_i = r^{i+1} p_{-1} \quad (i = 0, 1, 2, \ldots).$$

Для определения значения p_{-1} воспользуемся тем очевидным соображением, что сумма вероятностей всех состояний

$$\sum_{i=-1}^{\infty} p_i = p_{-1} \sum_{i=-1}^{\infty} r^{i+1} = p_{-1} \frac{1}{1-r}$$

должна равняться 1.
Отсюда

$$p_{-1} = 1 - r; \tag{15}$$
$$p_i = r^{i+1} (1 - r). \tag{16}$$

При выводе соотношений (15)—(16) предполагаем, что $r = \frac{\alpha}{\lambda} < 1$. В противном случае полученные для вероятностей значения приводят к абсурду. Этот результат легко понять, если вспомнить, что $\frac{1}{\lambda}$ есть средняя длительность обслуживания, а $\frac{1}{\alpha}$ — средняя величина интервалов времени между моментами поступления очередных требований. При $r > 1$ первая величина больше второй, и, как легко понять, система не будет справляться с обработкой поступающих требований, очередь будет неограниченно расти, и стремление к какому-либо устойчивому состоянию не будет иметь места.

Полученное распределение вероятностей различных состояний (15)—(16) представляет собой геометрическое распределение. Величина $1 - p_{-1} = r$ есть искомый коэффициент полезного действия системы:

$$\eta = r = \frac{\alpha}{\lambda}.$$

Вероятность того, что длина очереди будет не меньше k, равна:

$$\sum_{i=k}^{\infty} r^{i+1}(1-r) = \frac{r^k}{1-r}(1-r) = r^k.$$

Отсюда следует, что при стремлении «выжать» из системы большой к. п. д., неизбежно будет расти очередь. Так, при $\eta = 0{,}9$ вероятность того, что в очереди находится не менее пяти требований,

$$0{,}9^5 = 0{,}59.$$

Средняя длина очереди в геометрическом распределении

$$\sum_{i=0}^{\infty} ir^{i+1}(1-r) = \frac{r}{1-r}.$$

При $r = 0{,}9$ приводит к значению $l = 9$. Таким образом, стремление к максимальной загрузке обслуживающих приборов неизбежно приводит к росту очередей. При расчете систем массового обслуживания необходимо поэтому находить разумный компромисс между этими противоречивыми тенденциями.

Необходимость поиска такого компромисса — одна из особенностей систем со случайными потоками требований, отличающихся от регулярных систем, рассматривавшихся в предыдущих параграфах, где в принципе достижим сколь угодно высокий к. п. д. использования имеющихся ресурсов.

Для решения задачи на оптимум в случае систем массового обслуживания необходимо прежде всего задаться удельной (за единицу времени) ценой a потерь в результате пребывания требования в системе. Как показывается в теории, в рассмотренной выше простейшей задаче массового обслуживания вероятность того, что время t пребывания требования в системе заключено в бесконечно малом интервале $[t, t + dt]$, есть $(\lambda - \alpha)e^{-(\lambda-\alpha)t}\,dt$. Отсюда среднее время пребывания требования в системе

$$\int_0^{\infty} t(\lambda - \alpha)\, e^{-(\lambda-\alpha)t}\,dt = \frac{1}{\lambda - \alpha}.$$

За единицу времени в среднем возникает α требований. Общие потери в результате задержек в системе будут тогда в среднем равны

$$\frac{\alpha}{\lambda - \alpha}\, a = \frac{ra}{1-r}.$$

Уменьшить эти потери можно только за счет наращивания обслуживающих мощностей, суммарная пропускная способность

Q которых обратно пропорциональна $r = \frac{\alpha}{\lambda}$, т. е. к. п. д. их использования. Удельные расходы, отнесенные на единицу времени работы системы, которые необходимы для создания и эксплуатации этих мощностей, равны $\frac{c}{r}$, где c — некоторый постоянный коэффициент.

Суммарная цена обслуживания и происходящих в результате его ожидания потерь

$$f = \frac{ar}{1-r} + \frac{c}{r}.$$

Нетрудно проверить, что минимум этой функции достигается при $r = \frac{1}{1+\sqrt{\frac{a}{c}}}$. Это есть оптимальный к. п. д. рассматриваемой системы массового обслуживания.

6. Управление в динамических системах

Классическая динамическая система есть объект, который описывается системой обыкновенных дифференциальных уравнений. Такого рода объекты характерны прежде всего для различных механических систем: системами обыкновенных дифференциальных уравнений исчерпывающим образом описываются движения планет солнечной системы, полеты ракет и спутников, колебания маятника и т. п.

Второй большой областью приложения динамических систем является теория автоматического регулирования, основная задача которой состоит в обеспечении заданных режимов работы различного оборудования (двигателей, химических и атомных реакторов автоматических поточных линий и т. п.). В этом случае системами обыкновенных дифференциальных уравнений описываются как объекты управления, так и регуляторы, строящиеся на механических, электромеханических и электронных принципах.

Наконец, системами обыкновенных дифференциальных уравнений могут достаточно эффективно описываться и некоторые экономические процессы, прежде всего в так называемых макроэкономических моделях, описывающих экономику страны как одно целое.

Существует простой способ, позволяющий в системах обыкновенных дифференциальных уравнений ограничиться лишь первыми производными. Для этого достаточно все производные, кроме старших, принять за новые неизвестные. Пусть имеем систему

$$x'' + 2x' + xy = \sin t; \quad y'y = x + y.$$

Обозначая x' через u, получаем систему, содержащую лишь производные первого порядка:

$$u' + 2u + xy = \sin t; \quad y'y = x + y; \quad x' = u.$$

Обычно предполагают дополнительно, что система может быть решена относительно входящих в нее производных. В качестве независимой переменной в динамических системах, которые мы будем рассматривать, всегда выступает время t.

После всех этих предварительных замечаний запишем произвольную динамическую систему:

$$\left\{\begin{aligned}
&\frac{dy_1}{dt} = f_1(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; v_1, v_2, \ldots, v_k);\\
&\frac{dy_2}{dt} = f_2(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; v_1, v_2, \ldots, v_k);\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&\frac{dy_n}{dt} = f_n(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; v_1, v_2, \ldots, v_k).
\end{aligned}\right. \tag{17}$$

В систему входят переменные трех видов: *основные переменные* — $y_1, y_2, \ldots, y_n$ и параметры, разделяющиеся на два класса — *входные параметры* (входные функции) $u_1, u_2, \ldots, u_m$ и *управляющие параметры* $v_1, v_2, \ldots, v_k$. Входные функции, называемые также иногда возмущающими воздействиями, представляют собой некоторые заданные для каждого конкретного случая функции времени

$$u_i = u_i(t) \quad (i = 1, 2, \ldots, m).$$

С помощью управляющих параметров $v_1, v_2, \ldots, v_k$ можно осуществлять то или иное желательное для нас управление рассматриваемой системой. Управление в общем случае задается системой соотношений:

$$\left.\begin{aligned}
v_1 &= \varphi_1(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t);\\
v_2 &= \varphi_2(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t);\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
v_k &= \varphi_k(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t),
\end{aligned}\right\} \tag{18}$$

связывающих управляющие параметры с основными переменными и входными функциями системы. Систему уравнений (18) будем называть *законом управления* или просто *управлением*.

После задания управления система (17) преобразуется в новую, так называемую *свободную* форму, не содержащую управляющих параметров:

$$\left.\begin{aligned}
\frac{dy_1}{dt} &= g_1(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t);\\
\frac{dy_2}{dt} &= g_2(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t);\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
\frac{dy_n}{dt} &= g_n(y_1, y_2, \ldots, y_n; u_1, u_2, \ldots, u_m; t).
\end{aligned}\right\} \qquad (19)$$

Система (17), содержащая управляющие параметры, называется *управляемой* динамической системой.

Для иллюстрации введенных определений рассмотрим простой пример. Предположим, что у нас имеется запас y некоторого материала, потребляемого со скоростью u и производящегося (с целью пополнения запаса) со скоростью v. Скорость изменения запаса $\frac{dy}{dt}$ будет равна разности скоростей его пополнения и расходования:

$$\frac{dy}{dt} = v - u. \qquad (20)$$

Расход $u = u(t)$ представляет собой некоторую заданную функцию времени и не является объектом управления. Что же касается скорости производства v, то ею можно управлять в соответствии с тем или иным законом.

Естественным желанием является выбрать такой закон управления, который обеспечил бы поддержание некоторого запаса $a > 0$ при любом расходе $u(t)$. Этому условию удовлетворяет закон:

$$v = \begin{cases} u - k(y - a), & \text{при } u - k(y - a) \geqslant 0; \\ 0, & \text{при } u - k(y - a) < 0, \end{cases} \qquad (21)$$

где k — некоторая положительная константа. Предполагая, что управление ведется при $y \leqslant a$, и учитывая $u(t) \geqslant 0$, можно ограничиться для определения v только верхней строчкой соотношения (21). После внесения управления система (20) преобразуется в свободную динамическую систему, задаваемую уравнением:

$$\frac{dy}{dt} = -k(y - a). \qquad (22)$$

В теории управления динамическими системами представляют интерес два класса задач, называемых обычно задачами анализа и синтеза. Задача анализа применяется к свободным динамическим системам и состоит в изучении различных аспектов их поведения. Задача синтеза ставится применительно к системам, имеющим хотя

бы один управляющий параметр v_i. Смысл этой задачи состоит в нахождении такого закона управления, который обеспечит некоторое, наперед заданное поведение системы.

Простейшая задача анализа (задача Коши) состоит в том, чтобы при заданных начальных условиях $y_1(0) = a_1$, $y_2(0) = a_2$, ..., $y_n(0) = a_n$ определить траекторию системы, т. е. систему функций $y_1(t)$, $y_2(t)$, ..., $y_n(t)$, удовлетворяющую уравнениям, которыми задана рассматриваемая динамическая система, и имеющих заданные начальные значения.

Как доказывается в теории, задача Коши для системы обыкновенных дифференциальных уравнений с непрерывными правыми частями всегда имеет одно и притом единственное решение. Имеется большое число методов фактического численного решения задачи Коши, обеспечивающих нахождение последовательных точек искомой траектории с любой заданной точностью. В основе большинства этих методов лежит следующая идея: идя от начальной точки $M(0) = (y_1(0), y_2(0), \ldots, y_n(0))$ строится последовательно ряд точек $M(t_i)$ искомой траектории, где $t_i = t_{i-1} + \Delta t (i = 1, 2, \ldots, t_0 = 0)$. Причем, для вычисления координат $y_j(t_i)$ очередной точки используют координаты $y_j(t_{i-k})$ построенных ранее точек и значения производных $y_j'(t_{i-k})$ в этих точках, которые могут быть получены из уравнений, задающих систему.

Простейший (хотя и очень несовершенный) метод основывается на использовании приближенного соотношения $y_i(t + \Delta t) \approx \approx y_i(t) + y_i'(t)\Delta t$. Пусть, например, в уравнении (22) $k = 1$; $a = 1$. Построим несколько последовательных точек траектории с начальным условием $y(0) = 0$ и шагом $\Delta t = 0{,}1$.

Записывая уравнение $y' = 1 - y$, получаем

$$y'(0) = 1 - y(0) = 1,$$

откуда

$$y(0{,}1) \approx y(0) + y'(0)\,\Delta t = 0 + 1 \cdot 0{,}1 = 0{,}1;$$
$$y(0{,}2) \approx y(0{,}1) + y'(0{,}1)\,\Delta t = 0{,}1 + (1 - 0{,}1) \cdot 0{,}1 = 0{,}1 + 0{,}09 = 0{,}19;$$
$$y(0{,}3) \approx y(0{,}2) + y'(0{,}2)\,\Delta t = 0{,}19 + (1 - 0{,}19) \cdot 0{,}1 \approx 0{,}27.$$

Точное решение легко находится

$$y = 1 - e^{-t},$$

что дает

$$y(0{,}1) \approx 0{,}095; \quad y(0{,}2) \approx 0{,}181; \quad y(0{,}3) \approx 0{,}260.$$

Повышение точности при приближенном решении требует уменьшения шага Δt, или перехода к более точным методам (Рунге-Кутта, Адамса и др.).

Вторая важнейшая задача анализа свободных динамических систем — это анализ их устойчивости. Общий смысл понятия устойчивости состоит в том, что малые изменения начальных условий и входных функций не должны приводить к большим изменениям определяемых ими траекторий. В теории авторегулирования особое значение приобрела так называемая *асимптотическая устойчивость*.

Суть ее состоит в следующем. Пусть $M(t) = (y_1(t),\ y_2(t),\ \ldots,\ y_n(t))$ — траектория, заданная некоторыми начальными условиями $y_i(0) = a_i$, $(i = 1,\ 2,\ \ldots,\ n)$ или, другими словами, начальной точкой $M_0 = (a_1,\ \ldots,\ a_n)$. Давая малые изменения координатам $a_1, a_2, \ldots, a_n$, построим новую начальную точку $\tilde{M}_0$ и определяемую ею траекторию (переменную точку) $\tilde{M}(t)$. Говорят, что система локально асимптотически устойчива в точке M_0, если расстояние между точками $\tilde{M}(t)$ и $M(t)$ стремится к нулю при $t \to \infty$. Термин «локальный» применительно к устойчивости означает, что устойчивость имеет место лишь по отношению к малым изменениям начальных условий.

Если устойчивость сохраняется при любых изменениях начальных условий, то говорят об устойчивости системы в целом: в этом случае в пределе при $t \to \infty$ все траектории, независимо от своих начальных точек, сливаются в одну, система, как бы «забывает», из какого начального положения она начала развиваться.

Обе задачи анализа — определение траекторий и анализ на устойчивость — получают исчерпывающее решение для систем, определяемых линейными дифференциальными уравнениями с постоянными коэффициентами. В общем виде такая система может быть задана следующим образом:

$$\left.\begin{aligned}
\frac{dy_1}{dt} &= a_{11}y_1 + \ {}_{12}y_2 + \cdots + a_{1n}y_n + u_1(t);\\
\frac{dy_2}{dt} &= a_{21}y_1 + a_{22}y_2 + \cdots + a_{2n}y_n + u_2(t);\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
\frac{dy_n}{dt} &= a_{n1}y_1 + a_{n2}y_2 + \cdots + a_{nn}y_n + u_n(t).
\end{aligned}\right\} \qquad (23)$$

В теории линейных систем с постоянными коэффициентами особое значение имеет определитель:

$$D(s) = \begin{vmatrix} s - a_{11} & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ -a_{21} & s - a_{22} & -a_{23} & \cdots & -a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & s - a_{nn} \end{vmatrix}.$$

Этот определитель представляет собой полином степени n от s. Он называется обычно характеристическим полиномом рассматриваемой системы.

Общее решение системы (23) записывается в виде:

$$y_i(t) = c_{i1}\varphi_{i1}(t) + c_{i2}\varphi_{i2}(t) + \cdots + c_{in}\varphi_{in}(t) + \\ + \psi_{i1}(t) + \psi_{i2}(t) + \cdots + \psi_{in}(t) \qquad (i = 1, 2, \ldots, n).$$

Каждая из функций $\varphi_{ik}(t)$ соответствует определенному корню характеристического полинома $D(s)$, а каждая функция ψ_{il} — свободному члену $u_i(t)$. Вещественному корню $s = r$ соответствует функция $\varphi(t) = e^{rt}$, а паре комплексных корней $s = p \pm iq$ — пара функций $\varphi(t) : e^{pt}\cos qt$ и $e^{pt}\sin qt$. Функции $u_i(t) = e^{rt}$ соответствует функция $\psi_{il}(t) = be^{rt}$, функциям $\sin px \cos px$ или $a \sin px + b \cos px$ соответствует функция $A \sin px + B \cos px$, если

$$u_i(t) = a_0t^k + a_1t^{k-1} + \cdots + a_n,$$

то

$$\psi_{il}(t) = A_0t^k + A_1t^{k-1} + \cdots + A_n \text{ и т. д.}$$

Если функция $u_i(t)$ представляет собою сумму функций указанного вида, то и $\psi_{il}(t)$ будет представляться в виде соответствующей суммы.

Наконец, нужно иметь в виду, что все функции $\varphi_{ik}(t)$ должны быть различны как между собой, так и с любой из функций $\psi_{il}(t)$. Если это не выполняется, то различие достигается умножением соответствующих функций φ_{ik} в первом случае и функций $\psi_{il}(t)$ во втором, на множители вида t^m с минимально возможной степенью m, способной обеспечить требуемое различие функций.

Рассмотрим пример. Имеем систему

$$\begin{cases} x' = x + y + e^t; \\ y' = y - e^t. \end{cases}$$

Характеристический полином $D(s)$ системы

$$\begin{vmatrix} s-1 & -1 \\ 0 & s-1 \end{vmatrix} = (s-1)^2.$$

Он имеет двойной корень $s = 1$. Поскольку функции φ_{ik} должны строиться для каждого корня, то в рассматриваемом случае их должно быть две. Для того чтобы обеспечить их различие, одну умножаем на t. Получаем две функции e^t и te^t. Свободные члены e^t и $-e^t$ снова порождают функции ψ_{il} вида be^t. Между собой они могут совпадать, однако, необходимо обеспечить их различие по отношению к уже построенным функциям φ (постоянные коэффи-

циенты при установлении сходства и различия во внимание не принимаются). Для этой цели достаточно умножить ее на t^2. Поскольку коэффициент b неизвестен, обе функции ψ можно объединить в одну. Итак, решение нашей системы можно искать в виде:

$$x = ae^t + bte^t + ct^2e^t; \quad y = de^t + fte^t + gt^2e^t,$$

где a, b, c, d, f, g — некоторые постоянные коэффициенты.

Для определения значений этих коэффициентов подставляем найденные значения x и y в уравнения:

$$ae^t + be^t + bte^t + ct^2e^t + 2cte^t = ae^t + bte^t + \\ + ct^2e^t + de^t + fte^t + gt^2e^t + e^t;$$
$$de^t + fe^t + fte^t + gt^2e^t + 2gte^t = de^t + fte^t + gt^2e^t - e^t.$$

Приравнивая коэффициенты при одинаковых членах, получаем следующую систему уравнений:

$$a + b = a + d + 1;$$
$$b + 2c = b + f;$$
$$c = c + g;$$
$$d + f = d - 1;$$
$$f + 2g = f;$$
$$g = g.$$

Откуда

$$g = 0; \quad f = -1; \quad c = -\frac{1}{2}; \quad d = b - 1.$$

Коэффициенты a и b могут оставаться произвольными. Общее решение рассматриваемой системы представится в виде:

$$x = \left(a + bt - \frac{1}{2}t^2\right)e^t; \quad y = (b - 1 - t)e^t.$$

Значения коэффициентов a и b можно определить из начальных условий. Пусть $y|_{t=0} = y_0$ и $x|_{t=0} = x_0$. Тогда

$$a = x_0; \quad b - 1 = y_0; \quad b = 1 + y_0.$$

Таким образом, траектория с начальной точкой $\mu_0(x_0, y_0)$ задается уравнениями:

$$x = \left[x_0 + (1 + y_0)t - \frac{1}{2}t^2\right]e^t; \quad y = (y_0 - t)e^t.$$

Вопрос об асимптотической устойчивости линейных систем с постоянными коэффициентами получает исчерпывающее решение:

для асимптотической устойчивости такой системы в целом необходимо и достаточно, чтобы все корни ее характеристического полинома $D(s)$ имели отрицательные вещественные части.

В только что рассмотренном примере вещественные части обоих корней полинома $D(s)$ равны $+1$, и потому асимптотическая устойчивость не имеет места.

Для системы (22) характеристический полином $D(s)$ равен, как нетрудно проверить, $s + k$. Единственный корень $s = -k$ этого полинома отрицателен, и поэтому данная система обладает свойством устойчивости в целом. Ее решение, в соответствии с описанными выше правилами, следует искать в виде:

$$y = be^{-kt} + c.$$

После подстановки в уравнение (22) получаем:

$$-kbe^{-kt} = -k(be^{-kt} + c - a),$$

откуда $c = a$. Полагая $t = 0$, находим $y_0 = b + c = b + a$, откуда $y_0 - a = b$. Общее решение системы записывается в виде:

$$y = (y_0 - a)e^{-kt} + a.$$

При $t \to \infty$, независимо от начального значения y_0, $y \to a$, что наглядно демонстрирует свойство асимптотической устойчивости системы в целом. Попутно убеждаемся, что при $y_0 < a$ выбранный закон управления (21) действительно обеспечивает управление с наличием запаса y, который, начиная с некоторого момента времени, будет сколь угодно близок к установленной величине запаса a.

Переходя к задачам синтеза, остановимся прежде всего на классической задаче теории автоматического регулирования, а именно, на задаче построения такого закона управления, при котором рассматриваемая динамическая система оказывается локально асимптотически устойчивой. Один из наиболее часто встречающихся частных случаев этой задачи заключается в том, чтобы поддерживать с определенной степенью точности заданные значения основных переменных:

$$y_1(t) \approx a_1; \quad y_2(t) \approx a_2; \quad \ldots; \quad y_n(t) \approx a_n.$$

Как и задачи анализа, задачи синтеза наиболее полно изучены для линейных систем с постоянными коэффициентами. Удобным математическим аппаратом в этом случае является так называемое преобразование Лапласа. Суть этого преобразования состоит в том, что для любой непрерывной функции $y(t)$, определенной для зна-

чений $t > 0$ может быть определено ее изображение (по Лапласу) $Y(s)$, задаваемое формулой:

$$Y(s) = \int_0^\infty e^{-st} y(t)\, dt. \tag{24}$$

Преобразование, при котором функции заменяются их изображениями, как раз и называется преобразованием Лапласа. По изображению всегда можно восстановить исходную функцию, используя формулу:

$$y(t) = \frac{1}{2\pi i} \int_{a-i\infty}^{a+i\infty} e^{st} Y(s)\, ds, \tag{25}$$

где a — любая неотрицательная константа, превосходящая вещественные части всех особых точек (точек разрыва) функции $Y(s)$.

Применение преобразования Лапласа сильно упрощается тем, что может быть построен специальный словарь для наиболее употребительных функций и операций, позволяющий в большинстве встречающихся на практике случаев выполнять преобразования без применений формул (24) и (25). Выпишем для примера несколько строк из этого словаря:

$$y_1(t) + y_2(t) \longleftrightarrow Y_1(s) + Y_2(s);$$
$$ay(t) \longleftrightarrow aY(s) \quad (a = \text{const});$$
$$y'(t) \longleftrightarrow -y(0) + sY(s);$$
$$y''(t) \longleftrightarrow -y'(0) - sy(0) + s^2 Y(s);$$
$$a \longleftrightarrow \frac{a}{s} \quad (a = \text{const});$$
$$t^n \longleftrightarrow \frac{(n+1)!}{s^{n+1}};$$
$$e^{at} \longleftrightarrow \frac{1}{s-a} \quad (a = \text{const});$$
$$\sin at \longleftrightarrow \frac{a}{s^2 + a^2} \quad (a = \text{const});$$
$$\cos at \longleftrightarrow \frac{s}{s^2 + a^2} \quad (a = \text{const}).$$

Используя преобразование Лапласа и обратное ему преобразование, получаем еще один способ решения систем обыкновенных линейных дифференциальных уравнений с постоянными коэффициентами. Например, применяя преобразование Лапласа к уравнению (22).

$$\frac{dy}{dt} = -k(y - a),$$

находим

$$-y(0) + sY(s) = -k\left(Y(s) - \frac{a}{s}\right),$$

откуда

$$Y(s) = \frac{ak}{s(s+k)} + \frac{y(0)}{s+k}.$$

Поскольку

$$\frac{ak}{s(s+k)} = \frac{a}{s} - \frac{a}{s+k},$$

то, применяя обратное преобразование, получаем

$$y(t) = a - ae^{-kt} + y(0)e^{-kt} = a + [y(0) - a]e^{-kt},$$

что совпадает с решением, найденным выше.

Применим преобразование Лапласа к общей системе обыкновенных линейных дифференциальных уравнений с постоянными коэффициентами (23):

$$\begin{gathered}
(s - a_{11})Y_1(s) - a_{12}Y_2(s) - \cdots - a_{1n}Y_n(s) = U_1(s) + y_1(0);\\
-a_{21}Y_1(s) + (s - a_{22})Y_2(s) - \cdots - a_{2n}Y_n(s) = U_2(s) + y_2(0);\\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
-a_{n1}Y_1(s) - a_{n2}Y_2(s) - \cdots + (s - a_{nn})Y_n(s) = U_n(s) + y_n(0).
\end{gathered}$$

Решая полученную систему линейных алгебраических уравнений, придем к следующим выражениям:

$$\begin{gathered}
Y_i(s) = [U_1(s) + y_1(0)]\frac{N_{i1}(s)}{D(s)} + [U_2(s) + y_2(0)]\frac{N_{i2}(s)}{D(s)} +\\
+ \cdots + [U_n(s) + y_n(0)]\frac{N_{in}(s)}{D(s)} \qquad (i = 1,\ 2,\ \ldots,\ n),
\end{gathered}$$

где $D(s)$ — характеристический полином рассматриваемой системы; $N_{ij}(s)$ — полиномы от s степени на единицу меньше, чем $D(s)$ т. е. $(n-1)$, представляющие собой алгебраические дополнения элементов j-го столбца определителя.

Функция $\frac{N_{ij}(s)}{D(s)} = F_{ij}(s)$ называется передаточной функцией рассматриваемой системы по отношению к входному параметру u_j и выходной основной переменной y_i.

Эта функция имеет определенный физический смысл. Если принять

$$u_j(t) = \begin{cases} \infty & \text{при } t = 0 \\ 0 & \text{при } t \neq 0 \end{cases}; \qquad \int_0^\infty u_j(t)\,dt = 1,$$

то, как нетрудно видеть, $u_j(s) = 1$. Полагая $u_k(t) \equiv 0$ при $k \neq j$ и $y_1(0) = y_2(0) = \ldots = y_n(0) = 0$, получаем:

$$Y_i(s) = F_{ij}(s) \cdot 1 = F_{ij}(s).$$

Иными словами, передаточная функция $F_{ij}(s)$ может трактоваться как реакция системы по i-му выходу на единичный импульс, подаваемый на j-й вход при нулевых начальных условиях и отсутствии входных сигналов по всем остальным входам.

Если при нулевых начальных условиях положить $u_j(t) = a_j = \text{const}$, а $u_k(t) = 0$ при $k \neq j$, то система дает постоянные сигналы $y_i(t) \equiv b_i$ по каждому выходу. Отношение $\frac{b_i}{a_j} = p_{ij}$ называется коэффициентом усиления системы по паре j-й вход — i-й выход. Можно показать, что

$$\frac{b_i}{a_j} = F_{ij}(0).$$

Аналогично, если при нулевых начальных условиях принять $u_j(t) = a_j \sin \omega t$, а $u_k(t) \equiv 0$ при $k \neq j$, то на выходах появятся сигналы $y_i(t) = b_i \sin(\omega + \alpha_i) t$. Отношение

$$\frac{b_i(\cos \alpha_i + i \sin \alpha_i}{a_j} = p_{ij}(\omega)$$

называют (комплексным) коэффициентом усиления при частоте ω.

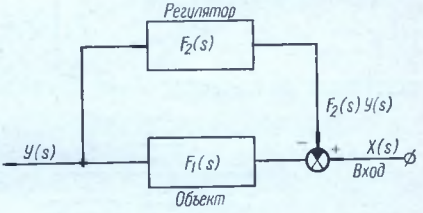

Рис. 20.
Структурная схема регулятора с обратной связью.

Можно показать, что $p_{ij}(\omega) = F_{ij}(\omega i)$. Функция $F_{ij}(\omega i)$ называется частотной характеристикой системы по паре j-й вход — i-й выход. Задание этой характеристики вместе с коэффициентом усиления p_{ij} определяет соответствующую передаточную функцию системы.

Приведенные результаты показывают, каким образом может быть экспериментальным путем определена передаточная функция линейной системы, в случае, когда соответствующая система уравнений (23) (т. е. коэффициенты a_{ij}) неизвестна.

Для решения задачи о синтезе линейной системы управления представляют с помощью передаточных функций как объект управления, так и регулятор. Предположим для простоты, что речь идет о системах с одним входом и одним выходом.

Пусть $F_1(s)$ — передаточная функция объекта; $X(s)$ и $Y(s)$ — Лапласовские изображения его входной и выходной функции (рис. 20). Вход регулятора, задаваемого передаточной функцией

$F_2(s)$, присоединяем к выходу объекта, а вырабатываемый им сигнал $F_2(s)Y(s)$ вычитается из входного сигнала $X(s)$.

Такое присоединение регулятора называется обратной связью (от выхода к входу). Уравнения (в Лапласовских изображениях) для полученной таким образом новой системы запишем в виде:

$$Y(s) = F_1(s)[X(s) - F_2(s)Y(s)],$$

откуда

$$\frac{Y(s)}{X(s)} = \frac{F_1(s)}{1 + F_1(s)F_2(s)} = F_3(s),$$

где $F_3(s)$ — передаточная функция новой системы.

Для обеспечения устойчивости достаточно подобрать функцию F_2 так, чтобы нули знаменателя вновь полученной передаточной функции F_3 имели бы отрицательные вещественные части. Например, если

$$F_1(s) = \frac{1}{s-a}; \quad a > 0,$$

то, выбирая

$$F_2(s) = \frac{c}{s+b},$$

получаем в качестве $F_3(s)$ функцию

$$\frac{\frac{1}{s-a}}{1 + \frac{1}{s-a}\,\frac{c}{s+b}} = \frac{s+b}{s^2 + (b-a)s + c - ab}.$$

Легко видеть, что при $b > a$ и $c > ab$ вещественные части обоих корней знаменателя будут отрицательны. Таким образом, обратная (отрицательная) связь с передаточной функцией $F_2(s) = \frac{c}{s+b}$ превращает первоначально неустойчивую систему в устойчивую.

Более сложным образом ставится и решается задача синтеза в том случае, когда требуется сконструировать регулятор, обеспечивающий не только устойчивое, но и целенаправленное и тем более оптимальное регулирование. Одна из наиболее известных постановок задачи целевого регулирования заключается в следующем. Пусть на множестве основных переменных $y_1, y_2, \ldots, y_n$ и входных параметров $u_1, u_2, \ldots, u_m$ задана некоторая целевая функция $f(y_1, y_2, \ldots, y_n, u_1, u_2, \ldots, u_m)$. Требуется сконструировать регулятор, который обеспечил бы поддержание при изменениях входных параметров постоянного (с некоторой точностью) значения целевой функции f. Пример подобной задачи дает рассмотренная выше за-

дача управления запасами. Управление, заданное соотношением (21), обеспечивает равенство a целевой функции $f(y, u) = y$.

Одна из известных постановок задач в теории оптимального управления состоит в том, чтобы обеспечить (при изменениях входных параметров u_i) минимальное или максимальное значение целевой функции. Например, пусть все в той же задаче управления запасами необходимо найти управление, обращающее в минимум функцию

$$f = by + \frac{cu}{y}.$$

Первый член by в целевой функции оценивает расходы на содержание запаса y, а второй оценивает потери, которые могут быть вызваны нехваткой запасов при внезапном росте спроса u. Минимум этой функции (при заданном u) находится из уравнения

$$b - \frac{cu}{y^2} = 0,$$

откуда

$$y = \sqrt{\frac{cu}{b}}.$$

Подстановка в уравнение (20) дает:

$$y' = \frac{\sqrt{\frac{c}{b}}}{2\sqrt{u}} u' = v - u,$$

или

$$v = u + \frac{cu'}{2\sqrt{bcu}}.$$

Для обеспечения устойчивости регулирования добавляем член ky. Требуемый закон управления запишется в виде:

$$v = \frac{cu'}{2\sqrt{bcu}} + u - ky.$$

Перепишем уравнение (20)

$$\frac{dy}{dt} = \frac{cu'}{2\sqrt{bcu}} - ky,$$

его общее решение

$$y = Ae^{-kt} + \sqrt{\frac{cu}{b}},$$

что и обеспечивает требуемое оптимальное регулирование.

Хотя рассмотренные примеры относятся к числу линейных систем с постоянными коэффициентами, многие из описанных выше приемов, в частности метод преобразования Лапласа и использование передаточных функций, могут быть распространены на более широкие классы динамических систем.

В заключение опишем еще один класс задач по синтезу законов управления, основанный на применении *принципа максимума* Л. С. Понтрягина. Наиболее просто этот принцип применяется для динамических систем вида:

$$\left.\begin{aligned}
&\frac{dy_1}{dt} = f_1(y_1, y_2, \ldots, y_n; \quad v_1, v_2, \ldots, v_m);\\
&\frac{dy_2}{dt} = f_2(y_1, y_2, \ldots, y_n; \quad v_1, v_2, \ldots, v_m);\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&\frac{dy_n}{dt} = f_n(y_1, y_2, \ldots, y_n; \quad v_1, v_2, \ldots, v_m)
\end{aligned}\right\} \qquad (26)$$

Системы такого вида, правые части которых не содержат входных параметров u_i и не зависят явно от времени t, принято называть *автономными*. Система (26) может быть переписана в *векторной форме*:

$$\frac{d\bar{y}}{dt} = \bar{f}(\bar{y}, \quad \bar{v}), \qquad (27)$$

где $\bar{y} = (y_1, \ldots, y_n)$; $\bar{v} = (v_1, \ldots, v_m)$.

Решение $\bar{y}(t)$ уравнения (27) принято называть траекторией. Управление $\bar{v}$ ищется для задач рассматриваемого типа в виде $\bar{v}(t)$, т. е. исключается зависимость управления от основных параметров (обратная связь).

На множестве управлений задается функционал вида

$$J = \int_{t_0}^{t_1} f_0(\bar{y}(t), \bar{v}(t))\, dt, \qquad (28)$$

оценивающий *качество* управления. Пределы интегрирования (t_0 и t_1) представляют собой моменты времени, когда точка $\bar{y}(t)$ находится в начале и в конце оцениваемого отрезка траектории. Простейшим случаем оценочного функционала J, получающемся при $f_0(\bar{y}, \bar{v}) \approx 1$, является время $\int_{t_0}^{t_1} dt = t_1 - t_0$ прохождения данного участка траектории.

Количество возможных вариантов $\bar{v}(t)$ ограничивается замкнутым множеством P допустимых значений. Задача оптимального управления в этом случае ставится следующим образом: в множестве P допустимых управлений найти такое, для которого функционал J принимает наименьшее возможное значение.

Для решения этой задачи исходная система (26) дополняется $n + 1$-м уравнением:

$$y_0' = f_0(y_1, y_2, \ldots, y_n; v_1, v_2, \ldots, v_m), \qquad (29)$$

правая часть которого совпадает с подынтегральным выражением оценочного функционала J.

Вводятся вспомогательные неизвестные $\psi_0, \psi_1, \psi_2, \ldots, \psi_n$, для которых выписываются дифференциальные уравнения:

$$\frac{d\psi_i}{dt} = -\sum_{k=0}^{n} \frac{\partial f_k(\bar{y}, \bar{v})}{\partial y_i} \psi_k \qquad (i = 1, 2, \ldots, n). \qquad (30)$$

После их решения в общем виде (с постоянными интегрирования) строится функция:

$$H(\bar{\psi}, \bar{y}, \bar{v}) = \sum_{k=0}^{n} \psi_k f_k(\bar{y}, \bar{v}).$$

Возможные оптимальные управления находятся на основе принципа максимума.

Для того чтобы управление $\bar{v}(t)$ было оптимальным, необходимо, чтобы для любого значения t на рассматриваемом интервале $t_0 \leqslant t \leqslant t_1$ функция $H(\bar{\psi}, \bar{y}, \bar{v})$ переменного вектора $\bar{v}$ достигала максимума в точке $\bar{v} = \bar{v}(t)$. При этом величина ψ_0 постоянна и $\psi_0 \leqslant 0$.

В приведенной формулировке принцип максимума дает только необходимые условия оптимальности управления. Для нахождения необходимых и достаточных условий требуются дальнейшие его уточнения. Однако и в приведенном виде принцип может с успехом применяться для решения практических задач.

Предположим, например, что нужно переместить на горизонтальном участке дороги с пренебрежимо малым трением из одной точки в другую транспортное средство, двигатель которого может сообщить ему любое ускорение v, по величине не превосходящее единицу.

Критерием оптимальности будет служить время, затраченное на перемещение. Иными словами, в качестве оценочного функционала выбираем

$$J = \int_{t_1}^{t_2} dt.$$

Уравнение движения объекта представится в следующем виде: $y_1'' = v$. Вводя обозначение $y_2 = y_1'$, запишем уравнения движения

в обычном виде: $y_1' = y_2$, $y_2' = v$. После дополнения уравнением вида (29) придем к системе:

$$\left.\begin{aligned} y_0' &= 1; \\ y_1' &= y_2; \\ y_2' &= v. \end{aligned}\right\}$$

Уравнения (30) в данном случае будут выглядеть следующим образом:

$$\frac{d\psi_0}{dt} = \frac{\partial(1)}{\partial y_0}\psi_0 + \frac{\partial y_2}{\partial y_0}\psi_1 + \frac{\partial v}{\partial y_0}\psi_2 = 0;$$

$$\frac{d\psi_1}{dt} = \frac{\partial(1)}{\partial y_1}\psi_0 + \frac{\partial y_2}{\partial y_1}\psi_1 + \frac{\partial v}{\partial y_1}\psi_2 = 0;$$

$$\frac{d\psi_2}{dt} = \frac{\partial(1)}{\partial y_2}\psi_0 + \frac{\partial y_2}{\partial y_2}\psi_1 + \frac{\partial v}{\partial y_2}\psi_2 = \psi_1.$$

Откуда $\psi_0 = \text{const}$; $\psi_1 = \text{const}$; $\psi_2 = \psi_1 t + c$ ($c = \text{const}$). Для функции H получим выражение:

$$H = \psi_0 + \psi_1 y_2 + \psi_2 v = \psi_0 + \psi_1 y_2 + (\psi_1 t + c)\, v.$$

Поскольку $-1 \leqslant v \leqslant 1$, то максимум функции H достигается при $v = -1$, если $\psi_1 t + c < 0$, и при $v = +1$, если $\psi_1 t + c > 0$. Будучи линейной, функция $\psi_1 t + c$ на заданном интервале может менять знак не более одного раза. Поэтому на этом интервале возможны лишь следующие последовательности оптимальных управлений: (-1), $(+1)$, $(-1, +1)$ или $(+1, -1)$.

Выбор нужной последовательности и момент переключения управления в двух последних случаях, происходит в зависимости от значений y_1 и y_2 на концах рассматриваемого интервала. Пусть, например,

$$t_1 = 0; \quad y_1(0) = 0; \quad y_2(0) = 1; \quad y_1(t_2) = 2; \quad y_2(t_2) = 0.$$

Закон движения на участке от 0 до τ, получаемый из уравнений движения, запишется в виде:

$$y_2(t) = vt + y_2(0); \quad y_1(t) = \frac{vt^2}{2} + ty_2(0) + y_1(0),$$

или

$$y_2 = vt + 1; \quad y_1 = \frac{vt^2}{2} + t.$$

Выбор нужной последовательности управлений можно осуществить методом перебора. На начальном участке $[0, \tau]$ выбираем управление $v = +1$ (режим разгона) и записываем уравнения движения в виде:

$$y_2 = t + 1; \quad y_1 = \frac{t^2}{2} + t.$$

На участке $[\tau, t_2]$ выбираем управление $v = -1$ (режим торможения). Уравнения движения представятся в виде:

$$y_2 = -t + c_1; \quad y_1 = -\frac{t^2}{2} + c_1 t + c_2.$$

Для точки t_2 получаем соотношения:

$$y_2(t_2) = -t_2 + c_1 = 0; \quad y_1(t_2) = -\frac{t_2^2}{2} + c_1 t_2 + c_2 = 2.$$

Откуда

$$c_1 = t_2; \quad c_2 = 2 - \frac{t_2^2}{2}.$$

Для точки τ из двух законов движения имеем:

$$y_2(\tau) = \tau + 1 = -\tau + c_1; \quad y_1(\tau) = \frac{\tau^2}{2} + \tau = -\frac{\tau^2}{2} + c_1\tau + c_2.$$

Откуда $2\tau = c_1 - 1 = t_2 - 1$ или $\tau = \frac{1}{2}(t_2 - 1)$. Подставляя это значение в последнее уравнение, найдем:

$$\left[\frac{1}{2}(t_2 - 1)\right]^2 + \frac{1}{2}(t_2 - 1) - t_2\left(\frac{t_2 - 1}{2}\right) - \left(2 - \frac{t_2^2}{2}\right) = 0$$

или

$$t_2^2 + 2t_2 - 9 = 0.$$

Следовательно,

$$t_2 = -1 \pm \sqrt{1 + 9} = \pm\sqrt{10} - 1.$$

Из двух корней реальный физический смысл имеет лишь положительный. Таким образом,

$$t_2 = \sqrt{10} - 1; \quad \tau = \frac{t_2 - 1}{2} = \sqrt{2{,}5} - 1.$$

Следовательно, разгоняясь на временном интервале $[0, \sqrt{2{,}5} - 1]$ с максимальным ускорением $v = 1$ и затем тормозя с таким же самым максимальным ускорением, мы достигнем требуемой точки $y_1 = 2$ за минимально возможное время, равное $\sqrt{10} - 1$.

Случай неавтономных динамических систем можно, вводя новое основное переменное $y_{n+1} = t$, свести к уже изученной задаче для уравнений, не содержащих явно t в своих правых частях.

7. Системный анализ

Под системным анализом принято понимать совокупность приемов и методов для изучения сложных объектов, которые будем называть *обобщенными динамическими системами*. Обобщенная динамическая система представляет собой совокупность взаимосвязанных объектов и процессов, изменяющихся во времени. Примерами таких систем могут служить солнечная система, человеческий организм, промышленное предприятие, экономика отдельной страны или группы стран, международное положение и т. п. Обычные (классические) динамические системы, рассматривавшиеся в предыдущем параграфе, представляют собой частный случай подобных обобщенных систем.

Исследование обобщенных динамических систем в системном анализе разбивается на несколько основных этапов. Первый этап — постановка задачи — состоит из определения объекта исследования, постановки целей, а также задания критериев для изучения этого объекта и управления им.

Этот этап плохо формализуется, так что успех здесь определяется прежде всего искусством и опытом системного аналитика, глубиной его понимания поставленной проблемы. Нетрудно понять огромную важность данного этапа: ведь неправильная или неполная постановка целей может свести на нет результаты всего последующего анализа.

В истории системного анализа было немало случаев неправильной постановки целей. Один из наиболее известных примеров такого рода представляет собой организация противовоздушной обороны английских торговых судов во время Второй мировой войны. Исследуя эффективность такой меры, как установка на торговые суда зенитных орудий, английские специалисты пришли сначала к выводу о необходимости отказа от нее. Причиной такого заключения было то, что в качестве цели первоначально была принята задача уничтожения зенитным огнем вражеских самолетов. Естественно, что зенитные орудия, стрелявшие с качающихся палуб и обслуживаемые недостаточно квалифицированными расчетами, почти не сбивали самолетов. Отсюда логическим образом следовало заключение о необходимости передачи зенитных орудий с судов на наземные батареи, где эффективность их использования была бы значительно выше.

Специалисты, однако, вовремя сообразили, что действительной целью установки зенитных орудий на торговые суда является не уничтожение вражеских самолетов, а защита самих судов. В этом смысле орудия оказались весьма эффективны: опасаясь зенитного огня немецкие летчики бомбили вооруженные пушками суда с больших высот и с гораздо меньшей точностью. Экономия от сокращения

потерь судов намного перекрывала затраты на установку и обслуживание орудий.

Еще более часто приходится встречаться с ошибкой, заключающейся в *неполной* постановке целей. Примером может служить ошибка, состоящая в том, что при рассмотрении целей общественного развития ограничиваются лишь материальными факторами, забывая о духовных.

Второй этап системного анализа состоит в очерчивании границ изучаемой системы и ее (первичной) *структуризации*. Этот этап, как и первый, в значительной мере основан на искусстве и опыте проводящих его специалистов. Смысл состоит прежде всего в том, что вся совокупность объектов и процессов, имеющих отношение к поставленной цели, разбивается на два класса — собственно изучаемую систему и *внешнюю среду*. Такое разделение происходит в результате последовательного перебора и включения в систему объектов и процессов, оказывающих заметное влияние на процесс достижения поставленных целей.

Окончание такого перебора может произойти прежде всего потому, что будут исчерпаны все существенные факторы. Система в этом случае может рассматриваться как замкнутая, т. е. с известной степенью приближения независимая от внешней среды. Пример такого рода может дать, скажем, изучение Солнечной системы с целью описания взаимного движения Солнца, Земли и Луны, рассматриваемых как материальные точки. Помимо названных трех объектов и процессов их взаимного притяжения, в зависимости от требуемой точности, может потребоваться включение в систему других планет. Однако ни при какой разумной в настоящее время точности не потребуется включать в систему, ставящую перед собой столь ограниченные цели, звезды, туманности или даже кометы, входящие в Солнечную систему. При изменении целей изучения границы системы могут быть расширены. Например, при включении в число целей изучения изменений вращения Земли вокруг своей оси необходимо принять во внимание наличие на ее поверхности больших водных масс.

Другая возможность разграничения системы от внешней среды основывается на том, что в ряде случаев можно при изучении системы ограничиться лишь влиянием среды на систему и пренебречь (с точки зрения поставленных целей) влиянием системы на среду. Если быть более точным, то это последнее влияние должно быть столь малым, чтобы оно не могло существенно изменить влияние среды на систему. При этом получаем *открытую систему*, поведение которой зависит от входных сигналов, поступающих из внешней среды. Таким образом, может изучаться, например, человеческий организм в различных природных условиях или экономика малой страны, зависящая от конъюнктуры на мировом рынке. Наоборот,

для страны, внешняя торговля которой оказывает большое влияние на мировой рынок, изучение национальной экономики может потребовать расширения рамок изучаемой системы до экономики группы стран или даже мировой экономики в целом.

Завершение процесса первичной структуризации состоит в том, что выделяются отдельные составные части — элементы изучаемой системы, а возможные внешние воздействия представляются в виде совокупности элементарных воздействий.

После этого наступает третий важнейший этап — составление *математической модели* изучаемой системы. Первым шагом в этом направлении является *параметризация*, т. е. описание выделенных элементов системы и элементарных воздействий на нее с помощью тех или иных *параметров*. В классических динамических системах употребляются лишь так называемые непрерывные параметры, т. е. переменные, принимающие любые вещественные значения на отрезке $[a, b]$ (где $-\infty \leqslant a < b \leqslant +\infty$). В обобщенных динамических системах наряду с непрерывными параметрами могут рассматриваться и *дискретные* параметры, например переменные, принимающие лишь целочисленные значения. Особую роль играют параметры, принимающие конечные множества значений. С их помощью можно описывать процессы и объекты, которые не могут быть охарактеризованы с помощью обычных числовых параметров, а различаются лишь *качественно*.

Например, для отношений между странами не подходит прямая количественная мера, но тем не менее их можно охарактеризовать какой-либо системой оценок: «хорошие», «плохие», «теплые», «нормальные» и т. п. В случае необходимости охарактеризовать нюансы отношений можно пользоваться не одним, а несколькими качественными параметрами.

Введение любого параметра (качественного или количественного) предполагает задание его *области определения*. В принципе область определения качественных параметров может быть составлена из любых объектов, например из понятий «хорошо», «плохо», «да», «нет» и т. п. Более удобно, однако, заменять их числовыми оценками в виде баллов, подобно тому как это делается в школьной 4- или 5-балльной системе оценок. Для различных качественных параметров число характеризующих их баллов может быть различным. Особое значение имеют качественные параметры с двухбалльной областью определения, которые мы будем называть *булевыми*. Булевыми параметрами характеризуются, например, различные разовые события. В качестве системы баллов в этом случае обычно употребляются 1 и 0, означающие соответственно наступление или ненаступление рассматриваемого события.

Параметризация изучаемой системы представляет собой лишь первый шаг в построении ее математической модели. Второй важ-

нейший шаг заключается в установлении различного рода зависимостей между введенными параметрами. Характер этих зависимостей может быть любым: для количественных (числовых) параметров зависимости обычно задаются в виде систем уравнений (обыкновенных алгебраических или дифференциальных). Для качественных параметров могут использоваться табличные способы задания зависимостей, основанные на перечислении всех возможных комбинаций значений параметров.

В общем случае могут встречаться комбинации зависимостей различных типов. Например, зависимость $y = f(x, \alpha)$, где y и x — непрерывные, а α — булев параметр, может задаваться в виде:

$$\begin{gathered}\text{при} \quad \alpha = 0 \quad y' = x + y + 1 \quad y|_{x=0} = 1, \\ \text{при} \quad \alpha = 1 \quad y = x^2 \ \text{для} \quad -\infty < x \leqslant 1 \\ \text{и} \ y = x^2 - x + 1 \quad \text{для} \quad x > 1.\end{gathered}$$

Зависимость $\beta = g(x, \alpha)$, где α — булев; x — непрерывный; β — качественный параметр, оцениваемый балами 1, 2, 3, может быть задана, например, в виде следующей таблицы:

	$x < 2$	$2 \leqslant x \leqslant 5$	$x > 5$
$\alpha = 0$	$\beta = 1$	$\beta = 2$	$\beta = 3$
$\alpha = 1$	$\beta = 2$	$\beta = 3$	$\beta = 1$

Таким образом, зависимости между параметрами в обобщенных динамических системах задаются в общем случае не простыми формулами, а произвольными алгоритмами с использованием любых средств как количественных, так и описательных.

Описывающие систему параметры, вообще говоря, изменяют свои значения во времени. Зависимости между параметрами могут в соответствии с этим использовать значения параметров в различные моменты времени. Например, значение параметра $z(t)$ может зависеть от значения параметров $y(t - a)$ и $x(t - b)$ в моменты времени, предшествующие моменту времени t. Наконец, наряду с вполне определенными функциональными зависимостями (задаваемыми однозначными функциями) в обобщенных динамических системах широко используются различного рода вероятностные соотношения.

Для рассмотренных параметров α, β, x вероятностная зависимость $\beta = G(x, \alpha)$ может быть, например, установлена следующим образом: при $x < 3$ и $\alpha = 0$ β с равной вероятностью может принять любое значение (1, 2 или 3); при $x < 3$ и $\alpha = 1$ β не может равняться 1, остальные два значения равновероятны; при $x \geqslant 3$,

независимо от значения α, β принимает значения 1, 2, 3 с вероятностями $\frac{1}{2}$, $\frac{1}{3}$ и $\frac{1}{6}$ соответственно.

Не исключается возможность взаимопротиворечивых зависимостей, которые в этом случае должны обязательно сопровождаться *весами*, т. е. оценками (выражаемыми в баллах) степени уверенности в справедливости таких зависимостей. С помощью этих весов противоречивые зависимости переводятся в вероятностные. Например, если имеются две зависимости $\beta|_{\alpha=0} = 1$ и $\beta|_{\alpha=0} = 2$ с весами 3 и 2 соответственно, то естественно считать, что при $\alpha = 0$ параметр β может принимать значение 1 с вероятностью $\frac{3}{3+2} = \frac{3}{5}$ и значение 2 с вероятностью $\frac{2}{3+2} = \frac{2}{5}$.

Современный системный анализ, как правило, имеет дело с системами, характеризующимися большим числом (от нескольких сотен до многих десятков тысяч) параметров различной природы. Зависимости между ними обычно являются разнообразными и сложными. Описание всех этих зависимостей (т. е. математическая модель системы) также весьма сложно и громоздко. Поэтому при построении математической модели стремятся по возможности сократить это описание. Одним из наиболее употребительных приемов является разбиение изучаемой системы на подсистемы, выделение *типовых* подсистем (имеющих одинаковые описания), установление иерархии подсистем и стандартизация связей подсистем на одних уровнях иерархии с однотипными системами на других уровнях.

В экономике таким приемом пользуются на различных уровнях, группируя однотипные отрасли или однотипные предприятия внутри одной и той же отрасли. При этом удается сократить суммарное описание системы, поскольку в группе однотипных подсистем достаточно привести описание лишь одной из них. В результате использования подобных методов иногда удается получить простое описание систем с огромным числом параметров. Один из наиболее разительных примеров такого рода дает статистическая теория газов, дающая простое и притом достаточно удовлетворительное описание поведения системы, состоящей из огромного числа молекул.

К сожалению, подобная ситуация в современном системном анализе является достаточно редкой: обычно, хотя в результате указанных выше примеров описание (модель) системы и сокращается, оно все-таки остается достаточно сложным и громоздким. Системы такого рода принято называть *большими* или *сложными*.

Выделение подсистем и установление их иерархии, помимо упрощения описания, преследует и другую цель: в процессе иссле-

дования производится уточнение первоначальной структуризации (разбиения на элементы) и параметризации системы, а также окончательная фиксация целей и критериев. В результате этого (третьего) этапа возникает законченная математическая модель системы, описанная на формальном (например, алгоритмическом) языке.

Задачи следующих этапов заключаются в исследовании построенной модели. Первая задача — это *прогноз развития* изучаемой системы. Для ее решения задаются различными предположениями о внешних воздействиях на систему в течение рассматриваемого периода и с помощью построенной математической модели определяют распределение вероятностей значений характеризующих систему параметров для любых фиксированных моментов времени. Термином *прогноз развития* мы подчеркиваем то обстоятельство, что, в отличие от классических динамических систем, для обобщенных динамических систем нельзя определить единственную траекторию развития, а лишь множество таких траекторий. Каждая траектория этого множества может реализоваться в действительности лишь с той или иной степенью вероятности.

Второе отличие от классического случая заключается в том, что, как правило, для сложных систем не удается найти аналитическое решение, позволяющее описать поведение системы в общем виде (для любого t). Поэтому обычно пользуются прямым (имитационным) моделированием изучаемой системы на ЭВМ. Методика моделирования напоминает рассмотренный в предыдущем параграфе численный метод решения дифференциальных уравнений. Отправляясь от начальных значений параметров (которые предполагаются известными) и задаваясь определенным шагом Δt по времени, последовательно шаг за шагом по заданным зависимостям между параметрами определяют значения параметров (или распределения этих значений) для моментов времени 0, Δt, $2\Delta t$, $3\Delta t$, ... и т. д. Часто оказывается полезным употреблять переменный шаг.

Получив прогноз развития изучаемой системы, производят анализ его результатов на соответствие заданным целям и критериям и, в случае необходимости, вырабатывают предложения по улучшению принятого ранее управления. Затем снова производится прогноз уже при новом управлении, снова вырабатываются предложения по улучшению управления и т. д., пока не получится удовлетворяющий результат.

По существу здесь описан метод решения задачи синтеза управления методом «проб и ошибок». Такой метод применим, разумеется, и к классическим динамическим системам. Однако, в отличие от классического случая, для обобщенных динамических систем этот метод является не только основным, но большей частью и единственно возможным, поскольку известные аналитические приемы

(подобные принципу максимума Понтрягина) для таких систем, как правило, непригодны.

Разумеется, последняя задача решается лишь тогда, когда целью системного анализа является выработка оптимального или, точнее, приближенно оптимального управления. В ряде случаев оказывается достаточным ограничиться лишь прогнозом развития системы. Возможно также привлечение и других методов анализа системы.

Рассмотрим более подробно два класса моделей обобщенных динамических систем, которые будем называть соответственно *булевыми моделями с односторонними и с двусторонними переходами*. Модели называются булевыми потому, что все характеризующие их параметры (включая элементарные внешние воздействия) являются булевыми. В первой модели сначала все параметры имеют значение 0 и в какие-то моменты времени (разные для разных параметров) могут изменить это значение на 1, после чего их обратный переход (к значению 0) оказывается уже невозможным.

Именно в этом и состоит свойство односторонности переходов, отмеченное в наименовании модели. В модели с двусторонними переходами параметры могут любое число раз менять свои значения с 0 на 1 и обратно.

Частным случаем булевой модели с односторонними переходами являются изученные ранее сетевые графики. Общая модель отличается от сетевого графика наличием альтернативных путей (которые не обязательно будут пройдены при реализации графика) и использованием вероятностных оценок для продолжительностей переходов от одних событий к другим.

Чтобы проще уяснить себе сущность модели, рассмотрим ее в конкретной интерпретации — как задачи прогнозирования научно-технического прогресса. Предположим, что требуется оценить вероятное время и пути решения некоторого числа нерешенных сегодня научно-технических проблем s_1, s_2, ..., s_m. Среди них могут быть как проблемы сугубо прикладного характера (например, проблема повышения производительности труда в угольной промышленности в 10 раз), так и абстрактные проблемы (например, построение единой теории поля). Предположим далее, что каждой из указанных проблем s_i приписан некоторый весовой коэффициент λ_i, определяющий относительную важность этой проблемы по сравнению с остальными ($i = 1, 2, \ldots, m$). Поставим в качестве цели решение всех указанных проблем, а в качестве критерия качества управления — минимизацию суммы $\Sigma\lambda_i\tau_i$, где τ_i — срок, потребовавшийся для решения проблемы s_i при заданных суммарных ресурсах R, отпущенных на решение всех этих проблем.

Первоначальная структуризация проблемы состоит в том, чтобы дополнить список поставленных проблем (называемых далее конеч-

ными или основными целями) $s_1, s_2, \ldots, s_m$ новыми проблемами (промежуточными целями) $s_{m+1}, \ldots, s_{m+n}$, решение которых может оказаться необходимым или полезным для достижения конечных целей. Параметризация состоит в приписывании каждому из выделенных элементов s_i двух булевых параметров $\alpha_i(t)$ и $\beta_i(t)$ ($i = 1, 2, \ldots, m + n$). Первый из них характеризует состояние элемента (проблема s_i решена или не решена), а второй — управляющее воздействие (проблема s_i поставлена в план и финансируется или нет). Введем также оценку вероятности $P_i(t)$ того, что к моменту времени t проблема s_i окажется решенной, иными словами, вероятность того, что $\alpha_i(t) = 1$ ($i = 1, 2, \ldots, m + n$).

С целью установления зависимости между параметрами для каждой из проблем s_i привлекается группа экспертов. Каждый из экспертов формулирует условие, состоящее в том, что некоторые из целей $s_{i1}, s_{i2}, \ldots, s_{ik}$ считаются уже достигнутыми и дают оценки времени t_i достижения цели s_i после выполнения поставленного условия. Эта оценка может производиться для нескольких различных величин ассигнований v_i, которые могут быть отведены для решения проблемы s_i, так что t_i являются в общем случае функцией от v_i. В результате возникают зависимости между параметрами α_i вида:

$$\alpha_i(t) = \alpha_{i1}(t - t_i)\,\alpha_{i2}(t - t_i) \ldots \alpha_{ik}(t - t_i). \qquad (31)$$

Для любого данного i будет несколько таких зависимостей в соответствии с числом привлеченных экспертов (некоторые зависимости могут, разумеется, совпадать, но нам все равно удобно считать их различными). Каждая из зависимостей получает весовой коэффициент r_{ij} (i — номер цели; j — номер эксперта в группе, оценивавшей эту цель)*. Из зависимостей вида (31), установленных различными экспертами, получаем следующую очевидную формулу для вычисления вероятности:

$$P_i(t) = \frac{\sum_{j=1}^{e_j} r_{ij} P_{ij1}(t - t_{ij})\, P_{ij2}(t - t_{ij}) \ldots P_{ijk_j}(t - t_{ij})}{\sum_{j=1}^{e_i} r_{ij}}, \qquad (32)$$

где $ij1, ij2, \ldots, ijk_j$ обозначены номера промежуточных целей, выдвинутых j-м экспертом в качестве условия достижения цели s_i; t_{ij} — его оценка времени достижения цели s_i после выполнения поставленного условия.

---

* Процедура подсчета r_{ij} включает в себя самооценку эксперта, его оценку другими экспертами и ряд дополнительных приемов.

Для того чтобы по уравнениям вида (32) можно было последовательно найти функции $P_i(t)$ для всех целей s_i ($i = 1, 2, \ldots, m + n$), нужно в процессе построения модели произвести некоторые дополнительные уточнения. Один из возможных путей заключается в том, что, вводя новые вспомогательные цели и дополнительно работая с экспертами, добиваются расслоения всех целей (как конечных, так и вспомогательных) на непересекающиеся множества M_0, M_1, ..., M_p. Множество M_0 составляется из целей, имеющих лишь безусловные оценки времени своего достижения. Для целей в любом из множеств M_i в качестве условий могут выступать лишь цели из множеств $M_0, M_1, \ldots, M_{i-1}$ ($i = 1, 2, \ldots, p$). При выполнении этого требования формул (32) оказывается достаточно для вычисления вероятностей $P_i(t)$ для всех целей при всех значениях t (с некоторым интервалом дискретности).

Чтобы лучше уяснить себе сущность таких вычислений, рассмотрим простейший пример. Пусть нам заданы две конечные цели s_1 и s_2 с весами 2 и 1 соответственно и две вспомогательные цели s_3 и s_4, каждая из которых оценивалась двумя экспертами. Данные оценок сведены в табл. 14 (для простоты пренебрегаем здесь зависимостью оценок времени t_{ij} от размера выделяемых ассигнований).

Таблица 14

Цели	s_1		s_2		s_3		s_4	
Эксперты	1	2	1	2	1	2	1	2
Условные	(s_2, s_4)	(s_3, s_4)	(s_3, s_4)	(s_3)	(s_4)	—	—	—
Оценка времени, t_{ij}	2	3	2	4	1	2	2	3
Вес r_{ij}	3	2	2	3	1	3	3	2

Из табл. 14 непосредственно видно, что цели распадаются на слои M_0 (цель s_4), M_1 (цель s_3), M_2 (цель s_2), M_3 (цель s_1). Функции $P_i(t)$ будем задавать векторами $(P_i(0); P_i(1); P_i(2); P_i(3); \ldots)$. Начиная с некоторого t все $P_i(t)$ в этих векторах равняются 1 и в этих местах векторы могут быть оборваны (ибо недостающие компоненты при необходимости легко могут быть выписаны). Если при оценке цели эксперт не выдвинул никаких условий, то произведение $r_{ij}P_{ij1}(t - t_{ij}) \ldots P_{ijk_j}(t - t_{ij})$ в уравнениях (32), оче-

видно, должно быть заменено на $r_{ij}Q(t-t_{ij})$, где $Q(t)=0$ при $t<0$ и $Q(t)=1$ при $t\geqslant 0$.

В таком случае будем иметь:

$$P_4(t)=\frac{3}{5}Q(t-2)+\frac{2}{5}Q(t-3)=\frac{3}{5}(0;\ 0;\ 1;\ 1)+$$
$$+\frac{2}{5}(0;\ 0;\ 0;\ 1)=(0;\ 0;\ 0,\ 6;\ 1);$$
$$P_3(t)=\frac{3}{4}Q(t-2)+\frac{1}{4}P_4(t-1)=\frac{3}{4}(0;\ 0;\ 1;\ 1;\ 1)+$$
$$+\frac{1}{4}(0;\ 0;\ 0;\ 0,\ 6;\ 1)=(0;\ 0;\ 0{,}75;\ 0{,}9;\ 1);$$
$$P_2(t)=\frac{2}{5}P_3(t-2)P_4(t-2)+\frac{3}{5}P_3(t-4)=$$
$$=(0;\ 0;\ 0;\ 0;\ 0{,}18;\ 0{,}36;\ 0{,}85;\ 0{,}94;\ 1);$$
$$P_1(t)=\frac{3}{5}P_2(t-2)P_4(t-2)+\frac{2}{5}P_3(t-2)P_4(t-3)=$$
$$=(0;\ 0;\ 0;\ 0;\ 0;\ 0{,}18;\ 0{,}47;\ 0{,}61;\ 0{,}91;\ 0{,}97;\ 1).$$

Полученные данные могут быть использованы для прогноза наиболее вероятного времени достижения каждой из рассматривающихся целей. Обычно в качестве такого времени принято считать *медиану* распределения, т. е. время t, для которого вероятность $P_i(t)$ равна 0,5. Если считать, что между найденными точками вероятность $P_i(t)$ меняется по линейному закону, так, как указано для случая $P_4(t)$ на рис. 21, то медианы N_i распределений $P_i(t)$ будут равны:

$$N_1=6\frac{3}{14}\approx 6{,}21;\quad N_2=5\frac{2}{7}\approx 5{,}29;$$
$$N_3=1\frac{2}{3}\approx 1{,}67;\quad N_4=1\frac{5}{6}\approx 1{,}83.$$

Степень неопределенности прогноза характеризуется обычно так называемыми *квартилями*: *нижний квартиль* Q_i' есть значение t, при котором $P_i(t)=0{,}25$, а для *верхнего квартиля* $Q_i''P(Q_i'')=0{,}75$. Нетрудно подсчитать, например, что $Q_4'=1\frac{5}{12}\approx 1{,}42$, $Q_4''=2\frac{3}{8}\approx 2{,}38$. В качестве единой *меры неопределенности* ΔQ_i может быть выбрана разность $Q_i''-Q_i'$. Для четвертой цели, например, эта разность равна $\Delta Q_4=\frac{23}{24}\approx 0{,}96$.

Для уточнения прогноза экспертиза делается непрерывной: всякий раз, когда тот или иной эксперт изменяет свое мнение, он посылает соответствующее сообщение в систему, где каждый раз

осуществляется пересчет функций распределения $P_i(t)$. Чтобы ускорить процесс улучшения прогноза, производится ранжировка целей (как основных, так и промежуточных) в соответствии с их *информационной значимостью*. Наибольшей информационной значимостью обладают те цели, уточнение прогноза для которых вызывает наибольшее уточнение прогноза для основных целей.

Количественной мерой информационной значимости i-й цели может служить специальный коэффициент I_i, вычисляемый следующим образом. Прежде всего распределение $P_i(t)$ заменяется дельта-распределением $P_i'(t)$ с той же самой медианой N_i ($P_i'(t) = 0$ при $t < N_i$ и $P_i'(t) = 1$ при $t \geqslant N_i$). Для основных целей s_1, s_2, ..., s_m вычисляются уменьшения d_j разностей ΔQ_j между соответствующими квартилями ($j = 1, 2, \ldots, m$). Тогда величину

$$\frac{\sum_{j=1}^{m} \lambda_j d_j}{\sum_{j=1}^{m} \lambda_j}$$

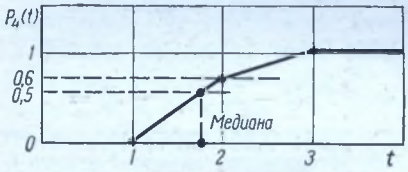

Рис. 21.
График распределения вероятности.

естественно принять за меру информационной важности i-й цели. Понятие информационной важности помогает концентрировать внимание на тех целях, уточнение прогноза по которым является наиболее важным. Сам процесс уточнения может использовать различные процедуры дополнительного обращения к экспертам, целенаправленное снабжение их информацией, проведение совещаний с соответствующими группами экспертов и т. п.

Близким к понятию информационной значимости является понятие *важности* (по срокам) той или иной цели. По определению *коэффициентом* важности i-й цели называется величина

$$z_i = \frac{\sum_{j=1}^{m} \lambda_j \Delta_i N_j}{\sum_{j=1}^{m} \lambda_j},$$

где $\Delta_i N_j$ — приращение медианы распределения $P_j(t)$ при условии сдвига вправо на один временной интервал распределения $P_i(t)$, т. е., иными словами, при замене функции $P_j(t)$ функцией $P_i(t-1)$.

В рассмотренном выше примере для вычисления коэффициента важности Z_3 необходимо, сохранив распределение $P_4(t)$, заменить

распределение $P_3(t) = (0;\ 0;\ 0{,}75;\ 0{,}9;\ 1)$ на $P_3'(t) = (0;\ 0;\ 0;\ 0{,}75;\ 0{,}9;\ 1)$. Тогда

$$P_2'(t) = \frac{2}{5} P_3'(t-2) P_4(t-2) + \frac{3}{5} P_3'(t-4) = \frac{2}{5}(0;\ 0;\ 0;\ 0;\ 0;\ 0{,}75;\ 0{,}9;\ 1)\ (0;\ 0;\ 0;\ 0;\ 0{,}6;\ 1) + \frac{3}{5}(0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0{,}75;\ 0{,}9;\ 1) = (0;\ 0;\ 0;\ 0;\ 0;\ 0{,}3;\ 0{,}36;\ 0{,}85;\ 0{,}94;\ 1);$$

$$P_1'(t) = \frac{3}{5} P_2'(t-2) P_4(t-2) + \frac{2}{5} P_3'(t-3) P_4(t-3) =$$

$$= \frac{3}{5}(0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0{,}3;\ 0{,}36;\ 0{,}85;\ 0{,}94;\ 1)\ (0;\ 0;\ 0;\ 0;\ 0{,}6;\ 1) +$$

$$+ \frac{2}{5}(0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0{,}75;\ 0{,}9;\ 1)\ (0;\ 0;\ 0;\ 0;\ 0;\ 0{,}6;\ 1) =$$

$$= (0;\ 0;\ 0;\ 0;\ 0;\ 0;\ 0{,}3;\ 0{,}54;\ 0{,}62;\ 0{,}91;\ 0{,}96;\ 1).$$

Отсюда $N_1' = 6\frac{5}{6} \approx 6{,}83$; $N_2' = 6\frac{2}{7} \approx 6{,}29$. Поскольку $\lambda_1 = 2$, $\lambda_2 = 1$,
то

$$Z_3 = \frac{2(N_1' - N_1) + 1(N_2' - N_2)}{2+1} = \frac{2(6{,}83 - 6{,}21) + (6{,}29 - 5{,}29)}{3} \approx 0{,}75.$$

Коэффициенты важности промежуточных целей могут употребляться для выбора начальных участков путей достижения конечных целей, когда прогноз еще не доведен до такой степени точности, чтобы был возможен однозначный выбор всего оптимального пути. В самом деле, нетрудно понять, что цели с большими коэффициентами важности являются наиболее полезными (а возможно, даже и необходимыми) для достижения конечных целей. Поэтому, когда неопределенность прогноза еще не позволяет выбрать оптимальные пути для достижения конечных целей, задача управления переформулируется и временной целью управления может становиться достижение наиболее важных промежуточных целей.

Еще в более непосредственном виде такая временная подмена целей может быть достигнута введением весов для всех промежуточных целей. Эти веса мы будем обозначать через $\mu(s_k)$. По определению,

$$\mu(s_k) = \sum_{i=1}^{m} \lambda_i p_{ik},$$

где λ_i — вес i-й основной цели; p_{ik} — вероятность того, что при ее достижении окажется необходимым предварительно достигнуть k-й промежуточной цели. Поскольку основные цели также могут выступать в качестве промежуточных (как, например, цель s_2

в рассмотренном примере), им могут быть приписаны кроме исходных базовых весов λ_i также относительные веса $\mu(s_i)$. Для примера, рассмотренного выше, легко получаем:

$$\mu(s_1) = \lambda_1;\ \mu(s_2) = 0{,}6\lambda_1 + \lambda_2;\ \mu(s_3) = \mu(s_4) = \lambda_1 + \lambda_2.$$

Управление в булевых моделях с односторонними переходами состоит в переводе тех или иных целей в реальный план. Планом можно, разумеется, считать и саму исходную модель, однако, благодаря наличию в ней альтернативных путей, при этом пришлось бы достигать многих промежуточных целей, которые впоследствии могут оказаться ненужными. Таким образом, сущность управления состоит в выборе альтернатив (одной или нескольких) для достижения каждой конечной цели. Эти альтернативы (варианты) характеризуются множествами всех целей, которые надо достичь в процессе достижения данной цели.

Основанием для выбора той или иной альтернативы служат три основные момента. Главным является степень уверенности, что данная альтернатива приведет к рассматриваемой конечной цели. Основанием для такой уверенности могут служить прежде всего достаточно высокая вероятность данной альтернативы (зависящая, в частности, от числа высказавшихся за нее экспертов) и степень согласованности мнений экспертов, высказавшихся за эту альтернативу. Мерой указанной согласованности может служить разность ΔQ_i между квартилями распределения времени достижения (конечной) цели s_i при учете только рассматриваемой альтернативы (с отбрасыванием всех остальных).

В рассмотренном выше примере для достижения конечной цели s_2 имеем две альтернативы (s_3) и (s_3, s_4). Вероятность первой из них равняется, очевидно, произведению вероятностей достижения s_2 без s_4 и s_3 без s_4, т. е.

$$\frac{3}{5} \cdot \frac{3}{4} = \frac{9}{20} = 0{,}45,$$

а вероятность второй — ее дополнению до 1, т. е. 0,55. Распределение по первой альтернативе выразится функциями

$$P_3'(t) = (0,\ 0,\ 1);\quad P_2'(t) = P_3'(t-4) = (0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 1).$$

Хотя степень неопределенности здесь равна нулю, это, разумеется, не может служить основанием для выбора первой альтернативы ввиду ее невысокой вероятности.

Вторым (после степени уверенности) основанием для выбора альтернативы служит обычно ожидаемое время (медиана распределения $P_i(t)$) достижения заданной конечной цели s_i при отбрасывании всех прочих альтернатив. Третим основанием служат оценка затрат

для достижения цели s_i при данной альтернативе и мера ее неопределенности (среднеквадратичное уклонение или разность между квартилями). Оценки затрат $R(s_i)$ производятся экспертами для каждой цели s_i (как основной, так и промежуточной) *без включения затрат* на достижение промежуточных целей, выдвинутых в качестве условий. В этом случае $R(s_i)$ можно считать независимыми случайными величинами и оценивать общие затраты по каждой альтернативе суммой затрат на все входящие в нее цели*.

Выделение и оценка альтернатив может производиться также для любых групп конечных или промежуточных целей. Так, как нетрудно видеть, в рассмотренном выше примере для группы (s_1, s_2) существует единственный вариант (s_1, s_2, s_3, s_4), требующий достижения всех рассматриваемых целей.

В случае, если не имеется достаточных оснований для выбора единственной альтернативы, для осуществления управления можно применять два основных приема. Один из них уже был описан выше: он заключается во включении в план ближайших целей с наибольшими коэффициентами важности (или с наибольшими относительными весами) без предопределения последующего развития плана. Второй прием заключается в объединении близких альтернатив (отличающихся небольшим числом элементов). Например, альтернативы (s_1, s_3, s_5, s_6) и (s_1, s_4, s_5, s_6) могут быть объединены в один вариант (s_1, s_3, s_4, s_5, s_6). Для возможности такого объединения существенно, чтобы цели, которыми различаются объединяемые варианты (в данном случае s_3 и s_4), имели сравнительно невысокие затраты для их реализации.

Перейдем теперь к описанию *булевой модели с двусторонними переходами*, которую удобно применять для описания различных социальных процессов. Снова, как и прежде, имеем некоторое число основных событий (булевых параметров) s_1, ..., s_m, дополненное вспомогательными событиями (параметрами) s_{m+1}, ... , s_n. С помощью экспертных оценок (или другим путем, описанным ниже) для каждого s_i ($i = 1, 2, \ldots, n$) устанавливаются зависимости вида:

$$s_i(t) = F_{ij}[s_{ij_1}(t - t_{ij_1}), \ \ldots \ , s_{ij_k}(t - t_{ij_k})],$$

где F_{ij} — произвольная булева функция параметров s_{ij_l} (строящаяся с помощью логических операций *И*, *ИЛИ*, *НЕ*); $t_{ij_l} \geqslant 0$**.

Например,

$$s_1(t) = [s_2(t-1) \wedge s_3(t)] \vee [s_3(t-2) \wedge \neg s_1(t-1)].$$

---

* О методах нахождения распределений для сумм независимых случайных величин см. [24].

** Если в правую часть входит $s_i(t - t_{ij_l})$, то $t_{ij_l} \neq 0$.

Для каждого параметра $s_i(t)$, как правило, устанавливается несколько таких зависимостей ($j = 1, 2, \ldots, l_i$), снабжаемых весами r_{ij}.

Обозначим $p_i(t)$ вероятность $p(s_i(t))$ того, что параметр s_i в момент времени t имел значение 1 (событие s_i имело место). Будем предполагать вероятности известными для всех моментов прошлого и настоящего времени ($t \leqslant 0$). Ограничиваясь определенной точностью рассмотрения событий во времени, можно считать, что все временные сдвиги t_{ij} кратны некоторому элементарному промежутку времени τ. Предполагая события в правых частях первичных оценок независимыми и используя известные формулы

$$p(\neg s_{i_1}) = 1 - p(s_{i_1}); \quad p(s_{i_1} \wedge s_{i_2}) = p(s_{i_1})\, p(s_{i_2});$$
$$p(s_{i_1} \vee s_{i_2}) = 1 - p(s_{i_1}) - p(s_{i_2}) + p(s_{i_1})\ p(s_{i_2}),$$

легко подсчитать вероятность $p_{ij}(\tau)$ наступления события s_i в момент времени τ в соответствии с оценкой F_{ij}.*

Тогда вероятность $p_i(\tau)$ наступления события s_i в момент времени τ (в соответствии со всеми имеющимися оценками)

$$p_i(\tau) = \frac{\sum_j r_{ij} p_{ij}(\tau)}{\sum_j r_{ij}}. \tag{33}$$

Последовательное повторение описанного процесса приведет к вычислению вероятностей $p_i(\tau)$, $p_i(2\tau)$, ..., $p_i(k\tau)$ для всех событий s_i. Тем самым мы получаем прогноз развития рассматриваемой модели.

Для перехода к управлению необходимо иметь группу событий, наступление или ненаступление которых не определяется зависимостями F_{ij}, а зависит исключительно от нас (например, посылка нашей ноты в модели отношений между государствами). Назовем эти события (параметры) *управляющими* и выделим их, вводя новые обозначения, скажем, $R_1, R_2, \ldots, R_k$. Задавая моменты наступления (или ненаступления) этих событий, мы вносим в модель определенное управление. Привлекая экспертов для задания тех или иных управлений и строя прогноз для каждого такого управления, обычно можно подобрать если не самое лучшее, то, во всяком случае, достаточно приемлемое управление. Работу экспертов, задающих управ-

* При использовании нулевых временных сдвигов t_{ij} приходится вводить добавочные ограничения на зависимости F_{ij}, расслаивающие события s_i, аналогично тому, как это имело место в предыдущей модели. Заметим также, что в ряде случаев, особенно при наличии нулевых временных сдвигов, предположение о независимости событий, входящих в правые части первичных оценок, оказывается слишком ограничивающим, и его приходится заменять условиями более общего характера.

ления, в ряде случаев целесообразно организовать в реальном масштабе времени: ЭВМ тем или иным образом выводит обстановку (значения параметров $p_i(t)$ в данный момент времени), а эксперты задают управляющие воздействия $R_1(t), \ldots, R_k(t)$. Далее ЭВМ проигрывают все варианты воздействий, после чего (с помощью экспертов или без нее) производится выбор наилучшего варианта, а значит, и обстановки в следующий момент времени $t + \tau$. Целью управления является достижение обстановки из некоторого класса, а критерием — чаще всего время.

Рассмотрим простейший пример булевой модели с двусторонними переходами. Пусть дано одно основное событие s_1, одно вспомогательное s_2 и одно управляющее $s_3 = R$.

Пусть для события s_1 имеется две зависимости с весами 1 и 2:

$$s_1(t) = [s_1(t-1) \wedge s_3(t)] \vee s_2(t); \quad s_1(t) = s_3(t) \wedge s_2(t).$$

Для события s_2 также предположим две зависимости с весами 2 и 1:

$$s_2(t) = s_1(t-2) \wedge s_3(t-1); \quad s_2(t) = \neg s_2(t-1) \wedge s_3(t).$$

В соответствии с приведенными выше формулами получаем:

$$\left.\begin{aligned} p_1(t) &= \frac{1 - p_1(t-1)\,p_3(t) - p_2(t) + p_1(t-1)\,p_3(t)\,p_2(t) + 2p_3(t)\,p_2(t)}{3}; \\ p_2(t) &= \frac{2p_1(t-2)\,p_3(t-1) + (1-p_2)(t-1)\,p_3(t)}{3}. \end{aligned}\right\} \tag{34}$$

Предположим также, что задана обстановка для $t = -1$ и $t = 0$:

$$p_1(-1) = 1; \quad p_2(-1) = 0; \quad p_1(0) = 0; \quad p_2(0) = 1.$$

Пусть также известно, что $p_3(-1) = 0$.

Пусть нашей целью является наступление события s_1. Рассмотрим два управления: $p_3(t) \equiv 0$ $(t \geqslant 0)$ и $p_3(t) \equiv 1 (t \geqslant 0)$. Пользуясь формулами (34), построим две таблицы изменения обстановки в соответствии с заданными управлениями:

t	-1	0	1	2	3
$p_1(t)$	1	0	1	1	1
$p_2(t)$	0	1	0	0	0
$p_3(t)$	0	0	0	0	0

t	-1	0	1	2
$p_1(t)$	1	0	$\frac{7}{9}$	$\frac{124}{243}$
$p_2(t)$	0	1	$\frac{2}{3}$	$\frac{1}{9}$
$p_3(t)$	0	1	1	1

Поскольку в первом случае требуемая обстановка достигается в минимальное время (один элементарный промежуток), то управление $R(t) = 0$ при $t \geqslant 0$ является наилучшим из всех возможных. Кроме того, оно, как нетрудно видеть, сохраняет наилучшую обстановку и во все последующие моменты времени. Не исключено, разумеется, что в будущем понятие наилучшей обстановки может измениться и потребуется менять управление.

Выше мы указали на один из возможных источников установления зависимостей $s_i(t) = F_{ij}$ — экспертные оценки. Имеется еще другой путь, который будем называть *ретроспективным анализом*. Предположим, что известна история изменения обстановки за достаточно длительный промежуток времени. Если некоторая формула $s_i(t) = f[s_{i_1}(t - t_1), \ \ldots, \ s_{i_p}(t - t_p)]$ удовлетворялась для подавляющего большинства моментов времени $t = 0, -1, -2, \ldots, -T$ из заданного промежутка $[-T, 0]$, то естественно предположить, что это будет иметь место и в ближайшем будущем и использовать эту формулу наряду с зависимостями, задаваемыми экспертами.

Ретроспективный анализ можно использовать также для уточнения весов r_{ij} экспертных оценок: те экспертные оценки, которые имеют много подтверждений в прошлом, увеличивают свой вес, и наоборот. Впрочем, далеко не всегда отсутствие достаточно убедительных подтверждений экспертной оценки в ретроспективном плане может служить основанием для умаления ее значения в прогнозном плане. Не исключено, что эксперт принял во внимание факторы, не включенные в модель, которые не имели места в прошлом, но будут иметь место в будущем. Кроме того, хотя обычно экспертные оценки зависят от сравнительно небольшого числа параметров, неявным образом каждый эксперт учитывает всю обстановку в целом. А эта обстановка (даже по параметрам, включенным в модель) может быть существенно иной в будущем, нежели в прошлом.

Техника работы с булевыми моделями может быть распространена на произвольные *конечные модели*, т. е. модели с конечным числом параметров, принимающих конечные множества значений (в общем случае различные для разных параметров). Сведение к булеву случаю осуществляется введением нескольких булевых параметров для каждого конечного параметра. Один из простейших (хотя и не самых экономных) способов сведения заключается в следующем. Пусть, например, конечный параметр x принимает четыре различных значения: 2 (плохо), 3 (удовлетворительно), 4 (хорошо) и 5 (отлично).

Параметр x может быть заменен системой событий (булевых параметров) s_1, s_2, s_3, s_4, принимающих значения 1 тогда, когда соответственно $x = 2$; $x = 3$; $x = 4$; $x = 5$. Зависимости устанав-

ливаются не для параметра x, а для заменяющих его событий s_1, s_2, s_3, s_4. Единственным отличием от случая обычной булевой модели является необходимость нормирования полученных вероятностей. В качестве истинных значений вероятностей p_i' наступления событий s_i принимаются не вероятности p_i, вычисленные по формулам (33), а величины

$$p_i' = \frac{p_i}{p_1 + p_2 + p_3 + p_4} \qquad (i = 1, 2, 3, 4).$$

В этом случае сумма $\Sigma p_i'$ равна 1, как и должно быть для вероятностей полной системы независимых событий.

8. Методы экстраполяции

При решении задач управления часто приходится встречаться с необходимостью предвидеть, какие значения будет принимать та или иная непрерывная величина $X = X(t)$ в будущем, если известен ряд ее значений в прошлые моменты времени $X_1 = X(t_1)$, ..., $X_n = X(t_n)$.

Задачи такого рода принято называть задачами экстраполяции. С экстраполяцией приходится встречаться при управлении сложными динамическими системами, при изучении будущего спроса в экономических системах и т. п.

Решение задачи экстраполяции сводится к нахождению зависимости $X = f(t)$, которая с достаточной точностью описывает поведение переменной X в прошлом и вместе с тем определена также и для некоторого интервала времени в будущем. Простейшим, хотя и не всегда достаточно эффективным, способом решения этой задачи является нахождение точной зависимости, так что $X_1 = f(t_1)$, ..., $X_n = f(t_n)$. Иными словами, речь идет о том, чтобы найти уравнение кривой $X = f(t)$, проходящей через все n заданных точек.

Задача эта имеет бесконечно много различных решений. Однако на практике по ряду соображений предпочитают искать решение в виде полинома возможно более малой степени. Как доказывается в курсах математического анализа, такое решение для n точек может быть всегда получено в виде полинома $P(t)$ $n-1$-й степени от t, причем этот полином определяется заданными точками однозначно*. Одной из простейших форм представления указанного полинома является следующая формула:

$$P(t) = \frac{(t-t_2)(t-t_3)\ldots(t-t_n)}{(t_1-t_2)(t_1-t_3)\ldots(t_1-t_n)} X_1 + \frac{(t-t_1)(t-t_3)\ldots(t-t_n)}{(t_2-t_1)(t_2-t_3)\ldots(t_2-t_n)} X_2 + \\ + \ldots + \frac{(t-t_1)(t-t_2)\ldots(t-t_{n-1})}{(t_n-t_1)(t_n-t_2)\ldots(t_n-t_{n-1})} X_n. \qquad (35)$$

* Не исключен случай вырождения, когда степень полинома оказывается меньшей, чем $n-1$.

Действительно, при $t = t_1$ первый член суммы обратится в X_1, в то время как остальные будут равны нулю, при $t = t_2$ второй член будет равен X_2, а остальные нулю и т. д. Полином, задаваемый формулой (35), называется обычно *интерполяционным полиномом Лагранжа*. Этим названием подчеркивается, что указанный полином может использоваться не только для экстраполяции, но и для *интерполяции*, т. е. для нахождения (приближенных) значений функции $X(t)$ для моментов времени, лежащих между заданными моментами $t_1, t_2, \ldots, t_n$.

Рассмотрим простейший пример применения интерполяционного полинома Лагранжа. Пусть при значениях $t = 0, 1, 2$, переменная X принимает значения соответственно 0, 1, 3. Требуется экстраполировать значение X для $t = 3$:

$$P(t) = 0 + \frac{(t-0)(t-2)}{(1-0)(1-2)} \cdot 1 + \frac{(t-0)(t-1)}{(2-0)(2-1)} \cdot 3 =$$
$$= -t(t-2) + \frac{3}{2} t(t-1) = \frac{t(t+1)}{2}.$$

Непосредственной проверкой убеждаемся, что полином действительно принимает в заданных точках значения 0, 1 и 3. Подставляя в него значение $t = 3$, получаем в качестве требуемого экстраполированного значения $X = 6$.

При большом числе точек степень интерполяционного полинома возрастает и его использование для целей экстраполяции и интерполяции приводит к очень громоздким вычислениям. Эти усложнения далеко не всегда необходимы с точки зрения требуемой точности интерполяции и экстраполяции. Поэтому на практике зачастую используют не все заданные точки, а только некоторую их часть. В случае экстраполяции берут некоторое число k последних точек. Интерполяционный полином строится только по этим точкам. При $k = 2$ дело сводится к проведению прямой через две точки, и экстраполяция (и, соответственно, интерполяция) называется *линейной*. При $k = 3$ полином имеет вид квадратного трехчлена и экстраполяция (интерполяция) носит название *квадратичной*.

Полиномиальная экстраполяция является достаточно удобной, однако ничто не мешает использовать для экстраполяции функции любого другого вида. Так, если известно, что экстраполируемая функция является периодической с периодом T, то экстраполяцию лучше всего выполнять с помощью отрезка ряда Фурье:

$$f(t) = a_0 + a_1 \cos pt + b_1 \sin pt + \cdots + a_n \cos npt + b_n \sin npt,$$

где $p = \frac{2\pi}{T}$. Неизвестные коэффициенты a_i и b_i могут быть опре-

делены из уравнений, получающихся подстановками $t = t_j$; $f(t_j) = X_j$ для всех заданных точек.

В этом случае, как и при полиномиальной экстраполяции, получается система линейных алгебраических уравнений. В других случаях могут возникать системы нелинейных и даже неалгебраических (так называемых *трансцендентных*) уравнений. Например, для функции $f(t)$ вида ae^{bt} и двух точек ($t = 1$, $X = 2$); ($t = 2$, $X = 5$) возникает система двух трансцендентных уравнений

$$2 = ae^b; \quad 5 = ae^{2b}.$$

Полученную систему нетрудно решить аналитически (например, поделив левые и правые части уравнений друг на друга, откуда $e^b = 2{,}5$ и $b = \ln 2{,}5$, $a = 2 : 2{,}5 = 0{,}8$). В более общих случаях применяют различные численные методы. С помощью современных ЭВМ такого рода задачи решаются сравнительно просто.

Использованный нами до сих пор принцип *точного* проведения экстраполирующей кривой через заданные точки не учитывает одного важного обстоятельства. Дело в том, что в сложных случаях заданные точки (t_1, X_1), (t_2, X_2), ..., (t_n, X_n) не отражают зависимость, которую требуется экстраполировать, абсолютно точно. Имеются две основные причины такого положения. Во-первых, в измерения величин t_i и X_i могут вкрасться случайные ошибки. Во-вторых, даже при точных измерениях изучаемая кривая $X(t)$ (на которой лежат заданные точки) может испытывать различного рода случайные отклонения, не связанные непосредственно с зависимостью, которую надо выявить и использовать для экстраполяции.

Поэтому, как это на первый взгляд ни парадоксально, результаты экстраполяции могут оказаться даже лучшими, если экстраполирующая кривая пройдет вблизи заданных точек, а не точно через них. На принципе приближенного проведения экстраполяционной кривой основан один из наиболее употребительных на практике методов экстраполяции — так называемый *метод наименьших квадратов*.

Смысл этого метода состоит в следующем. Прежде всего строится класс экстраполирующих функций $f(t, a_1, a_2, \ldots, a_m)$, зависящих от параметров $a_1, a_2, \ldots, a_m$. Выбор значений этих параметров, задающих требуемую экстраполирующую функцию, производится из условия минимальности суммы квадратов разностей значений функций f и X при заданных значениях $t = t_1, t_2, \ldots, t_n$. Указанная сумма имеет вид:

$$S = [f(t_1, a_1, \ldots, a_m) - X(t_1)]^2 + [f(t_2, a_1, \ldots, a_m) - X(t_2)]^2 + \cdots + [f(t_n, a_1, \ldots, a_m) - X(t_n)]^2.$$

Поиск значений параметров $a_1, a_2, \ldots, a_m$, обращающих величину S в минимум, производится любым из способов нахождения экстремумов, описанных в параграфе 1 настоящей главы. Для ряда простых классов функций $f(t, a_1, \ldots, a_m)$ требуемый минимум может быть найден классическим путем в результате решения системы уравнений $\frac{\partial S}{\partial a_i} = 0$ $(i = 1, 2, \ldots, m)$.

В случае, когда имеются основания для введения различия между заданными точками (t_i, X_i) с точки зрения желательной близости прохождения к ним экстраполяционной кривой, каждый член суммы S может быть снабжен специальным *весовым коэффициентом*, задаваемым постановщиком задачи. При необходимости точного прохождения кривой через те или иные точки, соответствующие члены исключаются из суммы S, и для них пишутся точные уравнения вида $f(t_i, a_1, \ldots, a_m) = X_i$, позволяющие исключить из выражения соответствующее число параметров.

Рассмотрим простейший пример экстраполяции с использованием метода наименьших квадратов. Предположим, что речь идет о линейной экстраполяции по трем точкам $A_1(t_1 = 0, X_1 = 2)$; $A_2(t_2 = 1, X_2 = 3)$; $A_3(t_3 = 2, X_3 = 5)$, и требуется экстраполировать значение X для $t = 3$.

Экстраполирующая функция f будет иметь вид $at + b$, а сумма S запишется в виде:

$$S = (b - 2)^2 + (a + b - 3)^2 + (2a + b - 5)^2.$$

Далее $\frac{\partial S}{\partial a} = 2(a + b - 3) + 4(2a + b - 5) = 10a + 6b - 26;$

$$\frac{\partial S}{db} = 2(b - 2) + 2(a + b - 3) + 2(2a + b - 5) = 6a + 6b - 20.$$

Для определения a и b получаем систему уравнений:

$$5a + 3b = 13;$$
$$3a + 3b = 10.$$

Отсюда

$$a = \frac{3}{2}; \quad b = \frac{11}{6}.$$

$$X|_{t=3} = a \cdot 3 + b = \frac{9}{2} + \frac{11}{6} = \frac{19}{3} = 6\frac{1}{3}.$$

Полученные результаты наглядно отображены на рис. 22. Легко видеть, что экстраполяционная прямая не проходит точно ни через одну из заданных точек A_1, A_2, A_3.

Метод наименьших квадратов удобен тем, что позволяет применять аналитические приемы нахождения минимума. При использовании численных методов можно применять и другие меры откло-

нения экстраполирующей кривой от заданных точек, например, сумму абсолютных величин разностей $f(t_i, a_1, \ldots, a_m) - X_i$.

Важнейшее значение имеет вопрос о точности экстраполяции. К сожалению, в общем случае, не имея никаких гипотез о свойствах экстраполируемой кривой $X = X(t)$, произвести оценку точности экстраполяции невозможно. Предположение о достаточно высокой степени гладкости кривой $X = X(t)$ (непрерывность не только самой функции $X(t)$, но и ее производных) приводит к тому, что полиномиальная экстраполяция (и, вообще, экстраполяция с помощью гладких функций) дает хорошую точность в непосредственной близости отрезка $A = [t_1, t_n]$, на котором расположен заданный участок экстраполируемой кривой.

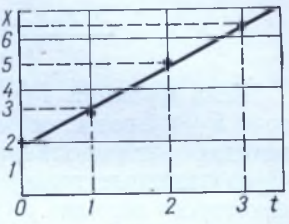

Рис. 22.
График линейной экстраполяции.

Предположение о существовании и ограниченности всех производных функции $X(t)$, на некотором отрезке B, объемлющем отрезок A, позволяет осуществить ее полиномиальную экстраполяцию на этот отрезок с любой заданной степенью точности*.

Центральным вопросом для получения хорошей экстраполяции является правильный выбор вида экстраполирующей функции $f(t, a_1, \ldots, a_m)$. Решение этого вопроса, достаточно сложное само по себе, еще более усложняется при наложении на основную закономерность изменения экстраполируемой функции различного рода случайных помех. Предположим сначала, что помеха мала по абсолютной величине и порядок ее величины приблизительно известен. Кроме того, будем считать, что помеха действует с равной вероятностью как в сторону увеличения изучаемой величины $X(t)$, так и в сторону ее уменьшения, так что среднее значение помехи равно нулю.

Приведем теперь в описанных условиях один достаточный признак правомерности полиномиальной экстраполяции. Этот признак основан на использовании так называемых разделенных разностей для любой функции $X = F(t)$, заданной в конечном множестве точек $t_1, t_2, \ldots, t_n$.

Обозначим через $X_1, X_2, \ldots, X_n$ значения функции $F(t)$ в соответствующих точках. Тогда первые разделенные разности вычисляются по формуле

$$\Delta X_i = \frac{X_{i+1} - X_i}{t_{i+1} - t_i} \qquad (i = 1, 2, \ldots, n-1),$$

* При этом предполагается возможным использовать любое число значений функции $X(t)$ на отрезке A и вычислять их с любой степенью точности.

вторые разделенные разности

$$\Delta^2 X_i = \frac{\Delta X_{i+1} - \Delta X_i}{t_{i+2} - t_i} \qquad (i = 1,\ 2,\ \ldots,\ n-2),$$

третьи разделенные разности

$$\Delta^3 X_i = \frac{\Delta^2 X_{i+1} - \Delta^2 X_i}{t_{i+3} - t_i} \qquad (i = 1,\ 2,\ \ldots,\ n-3).$$

Разделенные разности порядка $k \leqslant n-1$

$$\Delta^k X_i = \frac{\Delta^{k-1} X_{i+1} - \Delta^{k-1} X_i}{t_{i+k} - t_i} \qquad (i = 1,\ 2,\ \ldots,\ n-k).$$

Если функция $F(t)$ представляет собой полином k-й степени, то ее $k+1$-е разделенные разности обращаются в нуль. При точных методах экстраполяции (проведении экстраполирующей кривой через заданные точки) обращение в нуль разделенных разностей некоторого порядка p (разделенные разности вычисляются в заданных точках) означает возможность экстраполяции полиномом степени $p-1$. При малой помехе с нулевым средним значением условие обращения разностей p-го порядка в нуль должно быть заменено совокупностью двух условий: 1) абсолютные величины разделенных разностей p-го порядка малы (сравнимы с величиной помехи), 2) среднее значение этих разностей имеет еще меньший порядок величины. При этих условиях в методе наименьших квадратов в качестве экстраполирующей функции следует выбрать полином $(p-1)$-й степени.

Пусть, например, функция $X(t)$ задана множеством точек (1, 1); (2, 3); (3, 9); (4, 15); (5, 26); (6, 37); (7, 48); (8, 65). Нетрудно видеть, что значения функции X_i с точностью до ± 1 совпадают с t_i^2. В данном случае на квадратичную зависимость наложена помеха порядка величины, равной 1. Первые разделенные разности для заданных значений функции равны соответственно 2, 6, 6, 11, 11, 11, 17. Для вторых разностей получим: 2; 0; $\frac{5}{2}$; 0; 0; 3; а для третьих — $\pm\frac{2}{3}$; $\frac{5}{6}$; $-\frac{5}{6}$; 0; 1. Среднее значение для вторых разностей равно

$$\frac{2 + \frac{5}{2} + 3}{6} = 1{,}25,$$

а для третьих —

$$\frac{-4 + 5 - 5 + 6}{6 \cdot 5} = \frac{1}{15}.$$

Порядок величин третьих разностей сравним с величиной помехи, а их среднее — на порядок меньше. Таким образом, приходим

к выводу о необходимости применить квадратичную экстраполяцию. Этот выход вполне согласуется с тем, что заданные значения функции $X(t)$ могут быть вычислены по формуле $X = t^2 + \varepsilon(t)$, где $\varepsilon(t)$ — случайная помеха.

Описанный признак может быть распространен и на некоторые неполиномиальные зависимости. С этой целью используется подходящая замена переменных, сводящая исходную зависимость к полиномиальной.

Например, зависимость вида

$$X = a_0 + a_1 e^t + a_2 e^{2t} + \cdots + a_n e^{nt}$$

подстановкой $\tau = \ln t$ сводится к полиномиальной зависимости

$$X = a_0 + a_1\tau + a_2\tau^2 + \cdots + a_n\tau^n.$$

Подстановка $y = \ln X$, $\tau = \ln t$ превращает зависимость вида $X = \frac{a}{t}$ в линейную зависимость $y = \ln a - \tau$ и т. д.

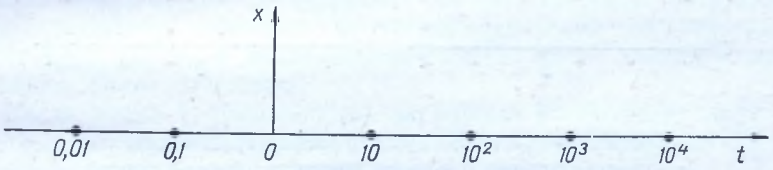

Рис. 23.
Логарифмическая шкала.

Свойства человеческого зрения позволяют особенно быстро и просто улавливать линейные зависимости, когда заданные точки (t_i, X_i) группируются на плоскости около некоторой прямой. Учитывая это свойство, иногда прибегают к нанесению заданных точек на график, причем, если линейная зависимость получается лишь после некоторой подстановки, на графике могут употребляться так называемые *функциональные шкалы*. Суть их заключается в том, что на шкале от начала координат откладываются отрезки, длина которых равна не значениям самой величины, а есть некоторая заданная функция от этих значений. На рис. 23 в качестве примера на оси абсцисс построена *логарифмическая шкала*. Использование логарифмической шкалы по оси ординат превращает в прямую линию график любой функции вида $y = ae^{bx}$. Логарифмические шкалы по обеим осям линеаризируют зависимости вида $y = at^b$ и т. д.

Иногда вид экстраполирующей функции может быть подсказан самой природой изучаемого процесса. Так, например, динамику

розничного спроса на тот или иной потребительский товар естественно характеризовать произведением $f(t)\,\varphi(t)\,\psi(t)$, где $\varphi(t)$ и $\psi(t)$ периодические функции с периодами, равными соответственно неделе и полумесяцу (выплата зарплат). Для характеристики товаров (одежды, обуви и т. п.) может добавиться третий периодический множитель с периодом в один год, характеризующий сезонные изменения спроса. Этот множитель $f(t)$ определяет динамику изменения среднегодового спроса. Он может экстраполироваться с помощью полиномиальных или экспоненциальных приближений. Для периодических множителей, как уже отмечалось выше, можно использовать приближения в виде начальных отрезков соответствующих рядов Фурье.

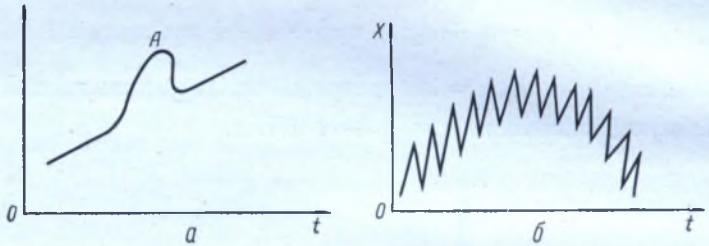

Рис. 24.
График, поясняющий принцип отфильтрации помех:
а — помеха единичная; *б* — помеха изменяется с большой частотой.

Метод наименьших квадратов и другие аналогичные ему методы приближенной экстраполяции дают хорошие результаты лишь тогда, когда случайная помеха имеет сравнительно небольшую величину. Если это условие не выполняется, то для успешной экстраполяции небходимо предварительно отфильтровать помеху. В ряде случаев такая фильтрация может быть сделана на основании содержательного представления об изучаемом процессе. Рассмотрим в качестве примера динамику ремонтно-восстановительных работ жилого фонда некоторого города за ряд лет (рис. 24, *а*). Допустим, что выброс A, имеющийся на кривой, явился результатом стихийного бедствия, которое случается в среднем раз в тысячу лет. Размер этой «помехи» настолько велик, что ее учет при экстраполяции полностью исказил бы реальную картину. Отфильтровав помеху (т. е. удалив из графика расходы, вызванные стихийным бедствием), мы можем значительно улучшить прогноз. Другой случай, когда задача фильтрации помех получает достаточно удовлетворительное решение, имеет место при условии, что основной процесс выражается сравнительно медленно меняющейся функцией, в то время как помеха (не обязательно малая по величине), сохраняя

нулевое среднее значение, меняет знак с большой частотой. Характерная картина такого процесса изображена на рис. 24, *б*.

Один из методов, позволяющих (хотя бы частично) отфильтровывать помеху, является метод *сглаживания* на основе *простого усреднения*. Пусть X_0, X_1, X_2, ..., X_n — значения функции (рис. 24, *б*). Предположим, что точки t_0, t_1, t_2, ..., t_n, в которых вычислены эти значения, расположены на оси абсцисс достаточно густо так, чтобы уловить все «взлеты» и «падения» быстропериодической составляющей изучаемой функции. Выберем так временный интервал τ, чтобы он был существенно больше среднего интервала τ_0, на котором быстропериодическая составляющая процесса меняет знак, и в то же время меньше интервала τ_1, на котором медленноменяющаяся составляющая процесса изменится существенно с точки зрения принятой точности экстраполяции*.

Двигая τ вдоль интервала $[t_1, t_n]$, заменяем для всех попавших в него точек t_p, t_{p+1}, t_{p+2}, ..., t_{p+k} соответствующие значения X_p, X_{p+1}, ..., X_{p+k} их средними значениями $\frac{1}{k+1}(X_p + \ldots + X_{p+k})$, относя эти значения к середине соответствующего интервала. При усреднении значения быстропериодической составляющей гасят друг друга, и медленноменяющаяся часть процесса проявляется более отчетливо.

Когда точки t_i расположены на равных расстояниях друг от друга, дело сводится к выбору числа q и замене значения X_q на

$$X_q = \frac{1}{2q+1}(X_0 + X_1 + \cdots + X_{2q}),$$

значения X_{q+1} на

$$X_{q+1} = \frac{1}{2q+1}(X_1 + X_2 + \cdots + X_{2q+1})$$

и т. д. Наилучший выбор интервала τ (или числа q) определяется предположениями о свойствах быстропериодической и медленноменяющейся составляющих процесса $X(t)$ и требует специального дополнительного исследования.

Иногда при сглаживании «загрубляют» приближение, используя средние значения для неперекрывающих друг друга интервалов. Так, вместо ежедневных значений спроса (365 точек за год) изучают поведение среднего спроса за каждую неделю (52 точки за год).

Второй способ фильтрации быстропериодических составляющих основан на использовании преобразования Фурье; интегрирование при этом производится от $-\infty$ до настоящего момента. Изображение

---

* Если величины интервалов τ_0 и τ_1 имеют один и тот же порядок, то неизбежно падение точности экстраполяции.

$\overline{X}(s)$ функции $X(t)$, называемое в этом случае ее спектром, имеет вид, показанный на рис. 25. Правая часть спектра соответствует быстропериодической части процесса. Отсекая ее (т. е. полагая $\overline{X} = 0$ при $s > s_0$), получаем (с некоторым приближением) спектр $\overline{X}(s)$ медленноменяющейся составляющей процесса. Сама эта составляющая $\overline{X}(t)$ может быть теперь найдена с помощью обратного преобразования Фурье.

В заключение отметим одно важное различие, которое имеется при экстраполяции детерминированных процессов (быть может, со случайными помехами) и чисто случайных процессов. Смысл экстраполяции детерминированного процесса состоит в угадывании закономерности, которая имеет место *для всего* изучаемого участка процесса. В случайных процессах встречаемся с противоположным явлением, поскольку поиск такой закономерности противоречит самой их сущности. Правда, свойство непрерывности случайного процесса приводит к тому, что его значения в близких друг к другу точках взаимозависимы (хотя и не в строго функциональном, а лишь в вероятностном плане).

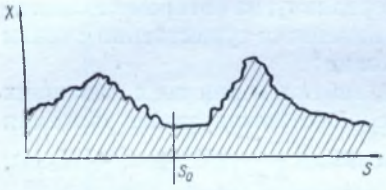

Рис. 25.
График изображения функции.

По мере удаления точек друг от друга эта зависимость делается все более слабой. Методы экстраполяции случайных процессов должны учитывать это обстоятельство, отдавая большее предпочтение точкам, близким к настоящему моменту, по сравнению с более далекими точками.

При применении метода наименьших квадратов отмеченный факт может быть учтен введением специальных весовых коэффициентов $\lambda_1, \lambda_2, \ldots, \lambda_n$, убывающих по мере удаления в прошлое. При равномерном расположении точек $t_1, t_2, \ldots, t_n$ (в которых задан процесс) вдоль оси абсцисс лучше всего воспользоваться экспоненциальным законом убывания коэффициентов λ_i (справа налево). Иными словами, выбрав некоторое положительное число $r < 1$, полагаем

$$\lambda_1 = r^{n-1}, \quad \lambda_2 = r^{(n-2)}, \ \ldots\ , \quad \lambda_{n-1} = r, \ \lambda_n = 1.$$

При экстраполирующей функции $f(t, a_1, \ldots, a_m)$, взвешенная сумма квадратов S, которую требуется минимизировать, запишется в виде:

$$S = \sum_{i=1}^{n} r^{n-i} (f(t_i, a_1, \ldots, a_m) - X_i)^2.$$

Результат экстраполяции не изменится, если вместо $r < 1$ взять

число $R = \frac{1}{r} > 1$ и записать взвешенную сумму квадратов

$$S = \sum_{i=1}^{n} R^{i-1} [f(t_i, a_1, \ldots, a_m) - X_i]^2.$$

Выбор величины r (или, соответственно, R) зависит от степени гладкости процесса. Для ее определения необходимо более глубокое исследование рассматриваемого процесса и, прежде всего, изучение корреляционной функции этого процесса. Для так называемых стационарных случайных процессов, с которыми обычно приходится иметь дело на практике, корреляционная функция $r(s)$ процесса $X(t)$ вычисляется по формуле

$$r(s) = \int_{-\infty}^{\infty} X(t+s) X(t)\, dt.$$

9. Методы статистического контроля

Статистический контроль представляет собой метод контроля качества изделий в случае, когда по тем или иным причинам такой контроль не может быть применен к каждому изделию, и о качестве *всех* изделий необходимо судить, контролируя лишь некоторую их *часть*. Совокупность всех изделий называется *генеральной совокупностью*, а часть, подвергаемая контролю,— *выборочной совокупностью* или просто *выборкой*.

Предположим, что объектом контроля является некоторая непрерывная величина x (размер, вес и т. д.).* В результате измерений величины x для выборочной совокупности получаем ряд значений этой величины $x_1, x_2, \ldots, x_n$. Если выборка была случайной и достаточно представительной, то *выборочное среднее*

$$\tilde{x} = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

можно принять в качестве приближенного значения *генеральной средней* $\bar{x}$ (т. е. среднего значения параметра для всех изделий генеральной совокупности).

Кроме выборочной средней вычисляется также выборочная дисперсия σ^2, равная квадрату средней квадратичной ошибки σ:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \tilde{x})^2.$$

* Таких контролируемых величин может быть несколько.

Выборочная дисперсия представляет собой приближение дисперсии (т. е. меры разброса значений) генеральной совокупности (хотя ее математическое ожидание несколько меньше). Она позволяет также оценить дисперсию $\sigma_{\bar{x}}^2$ и среднеквадратичное отклонение $\sigma_{\bar{x}}$ выборочной средней от генеральной средней. Для так называемой *бесповторной выборки* при числе элементов генеральной совокупности, равном N, используется формула

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n-1}} \sqrt{1 - \frac{n}{N}}.$$

При малой величине отношения $\frac{n}{N}$ применяется приближенная формула:

$$\sigma_{\bar{x}} \approx \frac{\sigma}{\sqrt{n-1}}. \tag{36}$$

Бесповторная выборка производится следующим образом. Из генеральной совокупности *случайным образом* выбирается первое изделие, затем из *оставшегося* множества также случайным образом — второе и т. д. Другой тип выборки (называемой повторной) получается, если после выборки и измерения очередного изделия его снова возвращают в генеральную совокупность, сохраняя таким образом за ним шанс быть выбранным повторно. Для повторной выборки

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n-1}} \sqrt{1 - \frac{1}{N}} \approx \frac{\sigma}{\sqrt{n-1}}. \tag{37}$$

Формулы (35)—(37) позволяют оценить степень достоверности определения генеральной средней $\bar{x}$. При достаточно больших выборках (на практике для $n > 20$) вероятность P_Δ того, что выборочная средняя будет отличаться (в ту или другую сторону) меньше, чем на $\Delta(\Delta > 0)$ определяется по формуле:

$$P = \frac{1}{2\pi} \int_{-\frac{\Delta}{\sigma_{\bar{x}}}}^{\frac{\Delta}{\sigma_{\bar{x}}}} e^{-\frac{t^2}{2}} dt. \tag{38}$$

Выражение, стоящее в правой части формулы (38), являющееся функцией от $\frac{\Delta}{\sigma_{\bar{x}}}$, принято обозначать $\Phi\left(\frac{\Delta}{\sigma_{\bar{x}}}\right)$. В учебниках по теории вероятностей обычно приводятся таблицы значений функции $\Phi(X)$, так что для подсчета вероятности P_Δ можно не вычислять интеграл, а воспользоваться этими таблицами. Для малых выборок

в курсах по математической статистике приводятся специальные таблицы, показывающие зависимость P_Δ от $\frac{\Delta}{\sigma_{\bar{x}}}$ и от числа n элементов выборки.

Например, при $n = 10$:

$$\Delta = 1{,}83\,\sigma_{\bar{x}} \qquad p_\Delta = 0{,}9;$$
$$\Delta = 3{,}25\,\sigma_{\bar{x}} \qquad p_\Delta = 0{,}99;$$
$$\Delta = 4{,}78\,\sigma_{\bar{x}} \qquad p_\Delta = 0{,}999$$

Используем полученные результаты для следующего примера. Пусть в результате бесповторной случайной выборки 10 изделий из 1000 получены следующие значения контролируемого параметра x:

$$x_1 = 1{,}01; \quad x_2 = 0{,}98; \quad x_3 = 1{,}02; \quad x_4 = 1{,}00;$$
$$x_5 = 0{,}99; \quad x_6 = 0{,}97; \quad x_7 = 1{,}02; \quad x_8 = 1{,}01; \quad x_9 = 0{,}98; \quad x_{10} = 1{,}02.$$

Требуется на основании произведенной выборки установить интервал, в котором с вероятностью 0,99 заключено среднее значение параметра x для 1000 изделий. Выборочное среднее

$$\bar{x} = \frac{1}{10}\sum_{i=1}^{10} x_i = 1,$$

выборочная дисперсия

$$\sigma^2 = \frac{1}{10}(0{,}01^2 + 0{,}02^2 + 0{,}02^2 + 0^2 + 0{,}01^2 + 0{,}03^2 +$$
$$+ 0{,}02^2 + 0{,}01^2 + 0{,}02^2 + 0{,}02^2) = \frac{1}{10^5}(1 + 4 + 4 + 1 + 9 + 4 +$$
$$+ 1 + 4 + 4) = 3{,}2 \cdot 10^{-4}.$$

$\sigma = 0{,}018$. Среднее квадратичное отклонение выборочной средней

$$\sigma_{\bar{x}} \approx \frac{\sigma}{\sqrt{n-1}} = \frac{0{,}018}{\sqrt{9}} = 0{,}006.$$

На основании приведенного выше табличного значения заключаем, что с вероятностью 0,99 генеральная средняя отличается от единицы (т. е. от выборочной средней) не более, чем на

$$3{,}25 \cdot \sigma_{\bar{x}} = 3{,}25 \cdot 0{,}006 \approx 0{,}02.$$

Больший интерес с точки зрения контроля качества представляет оценка ошибки (отклонения от заданного значения) параметра x для генеральной совокупности по результатам выборки. Пусть заданное значение параметра x есть 1. Тогда для приведенной выше выборки ошибки (взятые по абсолютной величине) равны:

$$\delta_1 = 0{,}01; \quad \delta_2 = 0{,}02; \quad \delta_3 = 0{,}02; \quad \delta_4 = 0; \quad \delta_5 = 0{,}01;$$
$$\delta_6 = 0{,}03; \quad \delta_7 = 0{,}02; \quad \delta_8 = 0{,}01; \quad \delta_9 = 0{,}02; \quad \delta_{10} = 0{,}02.$$

Выборочное среднее ошибки

$$\tilde{\delta} = \frac{1}{10}\sum_{i=1}^{10}\delta_i = 0{,}016.$$

Выборочная дисперсия ошибки

$$\tilde{\sigma}_\delta^2 = \frac{1}{10}(0{,}006^2 + 0{,}004^2 + 0{,}004^2 + 0{,}016^2 + 0{,}006^2 + 0{,}014^2 + 0{,}004^2 + 0{,}006^2 + 0{,}004^2 + 0{,}004^2) = 10^{-7}(36 + 16 + 16 + 256 + 36 + 196 + 16 + 36 + 16 + 16) = 64 \cdot 10^{-6}.$$

Выборочное среднее квадратичное отклонение $\tilde{\sigma}_\delta = 0{,}008$. Среднее квадратичное отклонение выборочной средней

$$\sigma_\delta = \frac{\sigma_\delta}{\sqrt{9}} \approx 0{,}003.$$

При $\Delta = 4{,}78\,\sigma_\delta = 0{,}013$ на основании приведенного выше табличного значения $P_\Delta = 0{,}999$. Таким образом, с вероятностью не менее, чем 0,999 можно утверждать, что средняя ошибка в генеральной совокупности не превысит $\tilde{\delta} + \Delta = 0{,}016 + 0{,}013 = 0{,}029$.

Третий пример связан с оценкой доли брака P в генеральной совокупности. Предположим, что по техническим условиям изделие бракуется, если отклонение параметра x от 1 больше или равно 0,03. В нашем примере при таком условии бракуется лишь одно (шестое) изделие сделанной выборки. Доля r брака в выборке составит 0,1 (10%). Величину r можно трактовать как среднее величин r_i, равных 0, если i-е изделие годно, и 1, если i-е изделие бракуется. Дисперсия

$$\sigma^2 = \frac{1}{10}(r^2 + r^2 + r^2 + r^2 + r^2 + (1-r)^2 + r^2 + r^2 + r^2 + r^2).$$

Легко понять, что при любой выборке эта дисперсия выражается формулой:

$$\sigma^2 = (0-r)^2(1-r) + (1-r)^2 r = r(1-r).$$

Итак, при подсчете доли брака в выборке

$$\sigma^2 = r(1-r). \tag{39}$$

В нашем случае $\sigma^2 = 0{,}1 \cdot 0{,}9 = 0{,}09$, откуда $\sigma = 0{,}3$. Среднее квадратичное отклонение

$$\sigma_p = \frac{\sigma}{\sqrt{n-1}} = 0{,}1.$$

При $\Delta = 3{,}25\sigma_p \approx 0{,}3P_\Delta = 0{,}99$. Таким образом, с вероятностью не меньшей чем 0,99 можно утверждать, что доля брака в генеральной совокупности не превысит $r + \Delta = 0{,}1 + 0{,}3 = 0{,}4$. Чтобы улучшить оценку доли брака в генеральной совокупности, нужно увеличить размер выборки.

В качестве четвертого примера определим размер выборки, при которой с заданной вероятностью 0,99 можно утверждать, что доля брака в выборке отличается от доли брака в генеральной совокупности не более чем на 0,01 (т. е. на 1%).

Трудность задачи состоит в том, что заранее неизвестна величина дисперсии выборки σ^2. Для решения задачи нужно оценить величину σ^2 сверху. Если нет никакой другой оценки, то из формулы (39) легко видеть, что максимум $\sigma^2 = 0{,}25$ при $r = 0{,}5$. В ряде случаев на практике эту оценку можно улучшить. Предположим, например, что доля брака r не может превысить 0,1. Тогда из формулы (39) имеем $\sigma^2 \leqslant 0{,}1 \cdot 0{,}9 = 0{,}09$ и $\sigma \leqslant 0{,}3$.

При бесповторной выборке n элементов из N среднее квадратичное отклонение σ_p доли брака в выборке от доли брака в генеральной совокупности найдем по формуле (35):

$$\sigma_p = \frac{\sigma}{\sqrt{n-1}} \sqrt{1 - \frac{n}{N}}.$$

Поскольку выборка должна быть, очевидно, большой, для оценки вероятности P_Δ можно воспользоваться формулой (38). По таблице значений $\Phi(x)$ находим, что для $P_\Delta = 0{,}99$, $\Delta = 2{,}6 \cdot \sigma_p$. Но нам задано, что $\Delta \leqslant 0{,}01$. Отсюда получаем:

$$2{,}6 \frac{\sigma}{\sqrt{n-1}} \sqrt{1 - \frac{n}{N}} \leqslant 0{,}01,$$

или

$$\frac{n-1}{1 - \frac{n}{N}} \geqslant 6{,}76 \cdot 10^4 \sigma^2.$$

Так как $\sigma \leqslant 0{,}3$, то наше неравенство будет выполнено, если

$$\frac{n-1}{1 - \frac{n}{N}} \geqslant 6{,}76 \cdot 10^4 \cdot 0{,}09 \approx 6 \cdot 10^3.$$

Тогда

$$n\left(1 + \frac{6000}{N}\right) \geqslant 6000 + 1.$$

Единицей в правой части можно пренебречь, так что окончательно имеем

$$n \geqslant \frac{6000}{1 + \frac{6000}{N}}.$$

При $N = 1000$ $n \geqslant \frac{6000}{7} \approx 860$, т. е. иными словами, выборка должна практически охватить всю генеральную совокупность. Однако, как показывает полученная формула, при очень больших n, например, $N = 600\,000$, для достижения требуемой точности оценки оказывается достаточной выборка 6 тыс. элементов, что может составлять сколь угодно малую долю генеральной совокупности.

При проведении выборок для статистического контроля чрезвычайно важным является соблюдение условия *случайности* выборки. Дело в том, что любая регулярность выборки, например, выборка каждого десятого изделия, сходящего со станка, может дать совпадение с тем или иным производственным ритмом и исказить результат. Так получится, например, если каждое десятое изделие выпускается в начале смены или после смены режущего инструмента и т. п.

Для достижения полной случайности выборки могут употребляться специальные датчики случайных чисел. В простейшем случае такой датчик генерирует случайную последовательность нулей и единиц, в которой единицы встречаются с заданной (регулируемой в случае надобности) частотой. Смена показаний датчика управляется потоком изделий. Если в момент прохода какого-то изделия мимо датчика его показание равнялось единице, изделие идет на контроль, при нулевом показании датчика изделие в выборку не попадает и не контролируется.

Другой метод — использование специальных таблиц случайных чисел. Если, например, имеется таблица 4-значных случайных чисел, и необходимо устроить случайную выборку, скажем 60 изделий из 10 000, то достаточно отобрать изделия, порядковые номера которых будут совпадать с первыми 60 числами таблицы.

В ряде случаев генеральная совокупность естественно делится на части: например, из общего числа N изделий N_1 изделий выпущено на первом станке, N_2 — на втором и т. д. Тогда средние $\tilde{x}_i$ и дисперсии $\tilde{\sigma}_i^2$ берутся по выборкам из каждой части, а общая средняя $\tilde{x}$ и дисперсия средней $\tilde{\sigma}^2$ находятся по формулам:

$$\tilde{x} = \frac{\tilde{x}_1 N_1 + \tilde{x}_2 N_2 + \cdots + \tilde{x}_k N_k}{N_1 + N_2 + \cdots + N_k},$$

$$\tilde{\sigma}^2 = \frac{\tilde{\sigma}_1^2 N_1^2 + \tilde{\sigma}_2^2 N_2^2 + \cdots + \tilde{\sigma}_k^2 N_k^2}{(N_1 + N_2 + \cdots + N_k)^2}.$$

10. Игровые методы

Переходим к рассмотрению методов так называемой *теории игр*. Эти методы применяются для анализа и выбора решений в *конфликтных ситуациях*, когда имеются две стороны, преследующие противоположные цели. Наиболее типичный пример конфликтных ситуаций в экономических системах — конкурентная борьба, могущая найти конкретное выражение в борьбе за международный рынок. В учебных примерах оказывается полезным рассматривать обычные игры (шахматы, карты и т. п.).

Результат или *исход* игры, даже в том случае, когда он не имеет прямой количественной оценки, обычно характеризуется некоторым числом, например, выигрыш $+1$, проигрыш -1, ничья — 0. Игра может быть парной или множественной (со многими участниками). Наиболее полно разработана теория *парных игр с нулевой суммой*, т. е. таких игр, при которых одна сторона выигрывает то, что проигрывает другая.

Развитие игры происходит в результате последовательного выполнения тех или иных *ходов*. Ход называется *личным*, если он предпринимается одним из игроков в результате сознательного анализа ситуаций. В противоположность личным, *случайные ходы* возникают не в результате сознательного решения, а в результате того или иного случайного процесса (например, сдача карт).

Стратегией игрока называется совокупность правил, по которым он анализирует ситуацию и делает ходы от начала игры до ее завершения.

Задание пары стратегий (A, B) (своей и противника) в парной игре полностью определяет ее исход, т. е. выигрыш одного и проигрыш другого*. Игра называется *конечной*, если у каждого игрока имеется лишь конечное число стратегий. Результаты конечной парной с нулевой суммой игры можно задавать матрицей, строки и столбцы которой соответствуют различным стратегиям, а ее элементы — соответствующие выигрыши одной стороны (равные проигрышам другой). Эта матрица называется *платежной матрицей* или *матрицей игры*. При этом удобно проигрыш первой стороны рассматривать как ее отрицательный выигрыш, а выигрыш второй — как ее отрицательный проигрыш. Если первая сторона имеет m стратегий, а вторая — n, то говорят, что мы имеем дело с *игрой* $m \times n$.

Рассмотрим игру $m \times n$ со следующей матрицей.

---

* Если в игре есть случайные ходы, то, строго говоря, определяются не выигрыш и проигрыш, а их математические ожидания.

	$B_1 B_2 \ldots\ldots B_n$
A_1	$a_{11} a_{12} \ldots\ldots a_{1n}$
A_2	$a_{21} a_{22} \ldots\ldots a_{2n}$
$\ldots$	$\ldots\ldots\ldots\ldots$
$\ldots$	$\ldots\ldots\ldots\ldots$
A_m	$a_{m1} a_{m2} \ldots\ldots a_{mn}$

Если первый игрок применяет стратегию A_i, то второй игрок будет стремиться к тому, чтобы выбором соответствующей стратегии свести выигрыш первого игрока к минимуму. Величина этого минимума, которую мы обозначим через α_i, равна, очевидно, $\min\limits_j a_{ij}$.

С точки зрения первого игрока (при любых ответах противника) целесообразно стремиться найти такую стратегию, при которой α_i обращается в максимум. Этот максимум, который мы обозначим α, называется *нижней ценой игры*. Поскольку значение α вычисляется по формуле

$$\alpha = \max_i \min_j a_{ij},$$

то его называют также *максимином*. Ему соответствует *максиминная стратегия* (их может быть и несколько), придерживаясь которой первый игрок при любых стратегиях противника обеспечит себе выигрыш, не меньший чем α (в зависимости от знака α это может быть и проигрыш, который в этом случае окажется минимальным).

Аналогичным образом определяется минимальный проигрыш (который может быть в действительности и выигрышем) для второго игрока:

$$\beta = \min_j \max_i a_{ij}.$$

Величина β называется *верхней ценой игры* или *минимаксом*. Ей соответствует минимаксная стратегия второго игрока.

Имеет место неравенство $\alpha \leqslant \beta$. При $\alpha < \beta$ первый игрок может существенно увеличить свой средний выигрыш по сравнению с α, если он будет пользоваться не *чистой* (одной единственной) стратегией, а так называемой *смешанной* стратегией. Смешанная стратегия C состоит в том, что при повторении игры происходит случайный выбор стратегии из некоторого множества смешиваемых стратегий и для каждой смешиваемой стратегии указывается вероятность ее выбора.

Доказано, что для любой конечной парной игры с нулевой суммой существует пара *оптимальных* стратегий (вообще говоря, смешанных). Свойство оптимальности означает, что любое отступление одного из игроков от оптимальной стратегии (при условии, что вто-

рой игрок продолжает придерживаться своей оптимальной стратегии) при многократном повторении игры может только уменьшить его средний выигрыш (увеличить средний проигрыш).

Величина выигрыша (быть может, отрицательного) первого игрока при пользовании парой оптимальных стратегий называется *ценой игры* и обозначается γ. Цена игры заключена между нижней и верхней ценой игры: $\alpha \leqslant \gamma \leqslant \beta$. Стратегии, которые смешиваются для получения оптимальной стратегии, будем называть *полезными*.

Решить игру — значит найти пару оптимальных стратегий и цену игры. Решение игры обладает одним важным свойством: если один из игроков использует свою оптимальную стратегию, а другой смешивает свои полезные стратегии в *любых пропорциях* (не обязательно оптимальных), то средний выигрыш продолжает оставаться равным цене игры.

При этом, правда, как и при любых отступлениях от оптимальной стратегии, соответствующее изменение стратегии противником может привести к увеличению его среднего выигрыша. Доказано, что у игры $m \times n$ число полезных стратегий с каждой стороны не превосходит минимального из чисел m и n.

Решение всякой парной конечной игры с нулевой суммой может быть получено методами линейного программирования. Однако, перед тем, как использовать эти методы, на практике обычно стараются провести предварительное упрощение задачи, исключая заведомо бесполезные стратегии.

Пусть, например, имеется игра 3×3 со следующей матрицей:

	B_1	B_2	B_3
A_1	1	0	2
A_2	0	2	1
A_3	−1	2	−2

Все элементы третьей строки матрицы не больше, чем соответствующие элементы второй строки. Поэтому, применяя вторую стратегию, первый игрок получит при любых стратегиях противника не меньший выигрыш, чем при применении второй стратегии.

Значит третья стратегия заведомо не может быть полезной, и ее следует исключить из рассмотрения. В оставшейся матрице 2×3 все элементы третьего столбца больше, чем соответствующие элементы первого столбца. Следовательно, если второй игрок применяет третью стратегию, то выигрыш его противника (т. е. его собственный проигрыш) больше, чем при применении первой стратегии. Поэтому при поиске решения стратегия B_3 должна быть заведомо исключена из рассмотрения.

Таким образом, в результате исключения заведомо бесполезных стратегий игра 3×3 свелась к игре 2×2 с матрицей:

	B_1	B_2
A_1	1	0
A_2	0	2

Следующим шагом в поиске решения находим нижнюю и верхнюю цену игры:

$$\alpha = \max_i \min_j a_{ij} = \max_i (0,0) = 0; \quad \beta = \min_j \max_i a_{ij} = \min_j (1,2) = 1.$$

Поскольку максимин (нулевой элемент) встречается как в первой, так и во второй строке, то обе стратегии первого игрока являются максиминными. Минимаксной же стратегией второго игрока является лишь стратегия B_1.

В нашем случае $\alpha \neq \beta$, и, следовательно, решение нужно искать среди смешанных стратегий. Рассмотрим другой пример. Пусть игра 2×3 задана матрицей:

	B_1	B_2	B_3
A_1	0	3	-1
A_2	3	2	1

Находим нижнюю и верхнюю цену игры:

$$\alpha = \max_i \min_j a_{ij} = \max_i (-1,1) = 1; \quad \beta = \min_j \max_i a_{ij} = \min_j (3,3,1) = 1.$$

В этом случае $\alpha = \beta = 1$. Поскольку $\alpha \leqslant \gamma \leqslant \beta$, то цена игры также равна 1. Решение же игры задается парой чистых стратегий (A_2, B_3), ибо максимин (он же минимакс) расположен во второй строке и третьем столбце.

В области чистых стратегий решение может быть получено непосредственно. Если же решение нужно искать в области смешанных стратегий (при $\alpha \neq \beta$), то в общем случае $m \times n$ матрицы $\|a_{ij}\|$ применяется следующий прием. Считая все m стратегией первого игрока полезными, определим вероятности их применения в смешанной *оптимальной* стратегии (если какая-то стратегия в действительности бесполезна, то соответствующая вероятность обратится в нуль). Обозначим эти вероятности $p_1, p_2, \ldots, p_m$, а цену игры (пока неизвестную) — γ.

Поскольку при оптимальной стратегии средний выигрыш первого игрока не меньше γ при *любой* стратегии противника, то запишем n неравенств:

$$\begin{aligned} p_1 a_{11} + p_2 a_{21} + \cdots + p_m a_{m1} &\geqslant \gamma; \\ p_1 a_{12} + p_2 a_{22} + \cdots + p_m a_{m2} &\geqslant \gamma; \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ p_1 a_{1n} + p_2 a_{2n} + \cdots + p_m a_{mn} &\geqslant \gamma. \end{aligned} \tag{40}$$

Введем новые неизвестные

$$x_1 = \frac{p_1}{\gamma}; \quad x_2 = \frac{p_2}{\gamma}; \ldots; x_m = \frac{p_m}{\gamma}.$$

Чтобы исключить возможность деления на нуль, можно всегда добиться $\gamma > 0$. Для этой цели достаточно ко всем элементам матрицы $\|a_{ij}\|$ прибавить одно и то же положительное число C и все ее элементы сделать положительными. Легко понять, что эта операция увеличивает цену игры на C, но не меняет искомых оптимальных стратегий.

Поскольку $p_1 + p_2 + \ldots + p_m = 1$, то $x_1 + x_2 + \ldots + x_m = \frac{1}{\gamma}$. Разделив левые и правые части неравенств (40) на γ, получим новую систему неравенств:

$$\begin{aligned} &a_{11}x_1 + a_{21}x_2 + \cdots + a_{m1}x_m \geqslant 1; \\ &a_{12}x_1 + a_{22}x_2 + \cdots + a_{m2}x_m \geqslant 1; \\ &\cdots\cdots\cdots\cdots\cdots\cdots \\ &a_{1n}x_1 + a_{2n}x_2 + \cdots + a_{mn}x_m \geqslant 1. \end{aligned} \tag{41}$$

При сделанных предположениях все x_i неотрицательны. Поскольку цель оптимальной стратегии — максимизация выигрыша, то при ее достижении линейная функция

$$f = x_1 + x_2 + \cdots + x_m = \frac{1}{\gamma}$$

должна обратиться в минимум. Итак, оптимальная стратегия первого игрока (т. е. набор вероятностей $p_i = \gamma x_i$) находится в результате минимизации функции

$$f = x_1 + x_2 + \cdots + x_m$$

при неотрицательных x_i, удовлетворяющих системе неравенств(41). Это есть обычная задача линейного программирования, которая может быть решена методами, описанными в параграфе 2 настоящей главы. В результате ее решения находится не только оптимальная стратегия первого игрока, но и цена игры γ.

Зная цену игры, оптимальную стратегию $(a_1, a_2, \ldots, a_n)$ второго игрока обычно можно находить уже без решения задачи линейного программирования.*. Для этого выбирается $n - 1$ полезных стратегий первого игрока (имея возможность менять местами игроков, можно считать, что $m \geqslant n$) и для каждой из них записывается средний выигрыш, который при этом должен быть обяза-

* Чтобы найти оптимальную стратегию второго игрока методом линейного программирования, достаточно поменять игроков местами и применить тот же прием, что и выше.

тельно равен цене игры γ. Например, если для первого игрока полезна стратегия A_i, то ей соответствует уравнение:

$$a_{i1}q_1 + a_{i2}q_2 + \cdots + a_{in}q_n = \gamma.$$

Кроме того, имеется еще одно уравнение

$$q_1 + q_2 + \cdots + q_n = 1.$$

Всего имеем n уравнений для n величин $q_1, q_2, \ldots; q_n$.

Для игр 2×2 оптимальная стратегия обоих игроков может быть получена без решения задачи линейного программирования. При $\alpha = \beta$ это очевидно, поскольку решением являются чистые стратегии. При $\alpha \neq \beta$ обе стратегии первого игрока и обе стратегии второго обязательно будут полезными (иначе существовало бы решение в чистых стратегиях). Поэтому неравенства (40) превращаются в этом случае в равенства:

$$\begin{aligned} a_{11}p_1 + a_{21}p_2 &= \gamma; \\ a_{11}p_1 + a_{22}p_2 &= \gamma. \end{aligned} \tag{42}$$

Поскольку $p_1 + p_2 = 1$, то имеется достаточное число уравнений для определения значений всех трех неизвестных p_1, p_2, γ. Применим этот прием для решения первого из рассматривавшихся выше примеров с матрицей $\left\|\begin{matrix} 1 & 0 \\ 0 & 2 \end{matrix}\right\|$. Уравнение (42) запишем в виде: $p_1 = \gamma$; $2p_2 = \gamma$. Присоединяя к ним уравнение $p_1 + p_2 = 1$, получаем решение:

$$p_2 = \frac{1}{3}; \quad p_1 = \frac{2}{3}; \quad \gamma = \frac{2}{3}.$$

Следовательно, при случайном чередовании первой и второй стратегии с относительными частотами $\frac{2}{3}$ и $\frac{1}{3}$ соответственно первому игроку обеспечен средний выигрыш в размере $\frac{2}{3}$. Заметим, что этот вывод идет вразрез с интуицией, которая подсказывает более часто использовать вторую стратегию, поскольку она имеет многообещающую строку выигрышей: (0,2) по сравнению с (1,0) в первой стратегии.

Для нахождения оптимальной стратегии второго игрока достаточно (поскольку уже известна цена игры) составить уравнение для его среднего проигрыша при любой из двух полезных стратегий первого игрока, например, стратегии A_1. Величина среднего проигрыша (равная цене игры) представится при этом в виде $1 \cdot q_1 +$

$+ 0 \cdot q_2$. Приравнивая ее цене игры $\gamma = \frac{2}{3}$, получаем $q_1 = \frac{2}{3}$ и, следовательно, $q_2 = \frac{1}{3}$. Итак, второй игрок должен смешивать стратегии B_1 и B_2 с частотами $\frac{2}{3}$ и $\frac{1}{3}$ соответственно.

Описанный метод простого решения игры может быть обобщен также на игры $2 \times n$.

Теоретико-игровые методы могут применяться для изучения ситуаций, которые не являются в строгом смысле слова конфликтными. Речь идет о ситуациях, где вторым игроком является природа.

Рассмотрим один из простейших примеров. Имеется участок земли и две стратегии его использования. Первая стратегия A_1 состоит в том, чтобы засевать участок яровой пшеницей, а вторая A_2 — озимой (с возможным пересевом весной). У природы также имеется две стратегии B_1 и B_2: первая, когда сильные морозы наступают после выпадания снега, а вторая — до выпадания (что приводит к гибели озимых). В этой игре ценой будет денежный доход, полученный от реализации урожая (в тыс. рублей).

Зададим некоторую матрицу игры (разумеется, на практике она должна быть определена из опыта):

	B_1	B_2
A_1	5	6
A_2	10	2

Определим нижнюю и верхнюю цены игры:

$$\alpha = \max_i \min_j a_{ij} = \max_i (5,2) = 5; \quad \beta = \min_j \max_i a_{ij} = \min_j (10,6) = 6.$$

Так как $\alpha \neq \beta$, то оптимальной будет смешанная стратегия. Уравнения (42) для нашего случая дают:

$$5p_1 + 10p_2 = \gamma;$$
$$6p_1 + 2p_2 = \gamma.$$

Вместе с уравнением $p_1 + p_2 = 1$ это приводит к решению:

$$p_1 = \frac{8}{9}; \quad p_2 = \frac{1}{9}; \quad \gamma = \frac{50}{9} = 5\frac{5}{9}.$$

Оптимальная стратегия второй стороны

$$5q_1 + 6q_2 = \frac{50}{9}; \qquad q_1 + q_2 = 1.$$

Откуда

$$q_1 = \frac{4}{9}; \quad q_2 = \frac{5}{9}.$$

Наиболее коварной «стратегией» со стороны природы в данном случае будет такая, в которой в среднем пять зим из девяти приводят к вымерзанию озимых. Если бы природа была сознательным противником и придерживалась бы именно этой стратегии, то лучшей стратегией для нас было бы высевание озимых лишь один раз в течение девяти лет. При этом имели бы средний годовой доход в размере $5\frac{5}{9}$ тыс. руб.

Если мы не располагаем никакой статистикой зим, то, по-видимому, разумнее всего предполагать худший случай, т. е. наиболее коварную стратегию со стороны природы. Однако в том-то как раз и состоит отличие рассматриваемой ситуации от подлинно конфликтной, что природа не является сознательным противником, и ее стратегия не обязательно оптимальна.

Предположим, что эта реальная (смешанная) стратегия задается вероятностями $q_1 = 0{,}9$; $q_2 = 0{,}1$ (которые могут быть определены из долголетних наблюдений), причем в чередовании «плохих» и «хороших» зим нет никакой другой закономерности, кроме этих вероятностей. Задача состоит в том, чтобы выработать оптимальную стратегию нашего поведения (вероятности p_1 и p_2) для обеспечения максимального среднего выигрыша в этих условиях.

Математическое ожидание выигрыша выразится суммой

$$\delta = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} p_i q_j$$

или

$$\delta = (5q_1 + 6q_2)p_1 + (10q_1 + 2q_2)\, p_2 = 5{,}1p_1 + 9{,}2p_2.$$

Необходимо найти максимум этой величины при условиях $p_1 \geqslant 0$; $p_2 \geqslant 0$; $p_1 + p_2 = 1$. Из последнего соотношения $p_2 = 1 - p_1$, откуда $\delta = 9{,}2 - 41p_1$. Ясно, что максимум этой величины, равный 9,2, может быть получен при $p_1 = 0$; $p_2 = 1$. В этих условиях правильной стратегией является высевание озимых в каждом году.

Более тщательный анализ ситуации показывает, что оптимальная стратегия противника (в данном случае природы) является своего рода перевальной точкой: если вероятность плохой зимы превышает вычисленную выше оптимальную вероятность $q_2 = \frac{5}{9}$, то наиболее благоприятной стратегией будет ежегодное высевание яровых. Если эта вероятность меньше чем $\frac{5}{9}$, то рационально высевать

каждый год озимые. Выигрыш в обоих этих случаях будет больше, чем цена игры равная $5\frac{5}{9}$.

На идее поиска оптимального ответа на любую (не обязательно оптимальную) стратегию другой стороны основываются приближенные методы решения игр, употребляющиеся при больших платежных матрицах и не близких друг другу α и β.* Для получения приближения употребляется следующий итерационный метод. Игрок A выбирает любую из своих стратегий, например A_{i1}. Игрок B выбирает в качестве ответа ту стратегию B_{j1}, которая наиболее невыгодна для A при стратегии A_{i1}. Игрок A отвечает на это лучшей (против стратегии B_{j1}) стратегией A_{i2}. Игрок B ищет стратегию B_{j2}, наилучшую против смеси в равных пропорциях стратегий A_{i1} и A_{i2}. Стратегия A_{i3} должна быть наилучшей против равновероятной смеси стратегий B_{j1} и B_{j2}, стратегия B_{j3} — наилучшей против равновероятной смеси стратегий A_{i1}, A_{i2}, A_{i3} и т. д.

Если продолжить этот процесс достаточно долго, то частоты, с которыми встречаются различные стратегии, стремятся к вероятностям их применения в паре оптимальных стратегий, а средний выигрыш — к цене игры.

---

* Если α и β мало отличаются друг от друга, то достаточно хорошими приближениями к решению будут минимаксные стратегии.

Глава III

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ

1. Общие сведения об автоматизированных системах управления и системах обработки данных

Всякая система управления с точки зрения технологии ее функционирования решает три основные задачи: сбор и передача информации об управляемом объекте, переработка информации и, наконец, выдача управляющих воздействий на объект управления. Автоматизированная система управления (АСУ) автоматизирует все эти этапы. Этим прежде всего АСУ отличается от простого использования ЭВМ в управлении. При простом использовании ЭВМ на ней решаются отдельные задачи управления, т. е. осуществляется переработка информации, этапы же сбора информации и формирования управляющих воздействий при этом обычно не автоматизируются. В этой главе будут рассмотрены принципы построения АСУ, включающие в свой состав ЭВМ, но, разумеется, не сводящиеся к ним.

Различают два основных типа автоматизированных систем управления: системы управления *технологическими процессами* (АСУТП) и системы *организационного* или *административного* управления (АСОУ). Первые имеют дело с технологическими процессами в широком смысле слова (управление ракетой, станком, домной и т. п.), вторые — с объектами экономической и социальной природы.

Главное отличие этих двух типов систем заключается прежде всего в характере объекта управления. Для первых — это различные машины, приборы, устройства, а для вторых — прежде всего люди, человеческие коллективы. Другое отличие — форма передачи информации. В системах управления технологическими процессами основной формой передачи информации являются различные сигналы (электрические, оптические, механические и др.) В системах же организационного управления основная форма передачи информации — документ.

В последнее время наметилась тенденция слияния АСУТП и АСОУ в единые *интегрированные* системы управления. При таком слиянии все большая часть циркулирующей в системе информации

передается в виде сигналов и специальных типов документов на машинных носителях (магнитных лентах для станков с программным управлением). Тем самым грани между АСУТП и АСОУ до известной степени стираются.

Различают также *автоматизированные* и *автоматические* системы управления. Автоматизированная система включает человеческое звено (операторов или административный аппарат) в качестве своей органической составной части. Автоматическая же система после монтажа и наладки в принципе может функционировать и без участия человека (исключая, быть может, лишь профилактический контроль и ремонты).

Автоматические системы применяются лишь для управления технологией, хотя и здесь зачастую по ряду причин предпочтение может быть отдано автоматизированным системам. Что же касается систем организационного управления, то они, по своей специфике, не могут быть полностью автоматическими. Люди в таких системах решают следующие задачи. Во-первых,— это постановка и корректировка целей и критериев управления (они могут меняться при изменении условий). Во-вторых,— это внесение творческого элемента в поиск наилучших путей достижения поставленных целей (решительное изменение применяемой технологии или организации и т. п.). Третья задача — это окончательный отбор вырабатываемых системой решений и придание им *юридической силы*. Наконец, четвертая задача — это снабжение системы первичной информацией, сбор которой невозможно или нерационально полностью автоматизировать (данные кадрового учета, пожелания об изменении места работы и т. п.).

Автоматизация управления технологическими процессами имеет уже достаточно длинную историю. На первых порах для целей автоматизации употреблялись лишь простые устройства — регуляторы (механические, электромеханические или электрические). Задача автоматизации сводилась в основном к обеспечению устойчивости регулируемых процессов и поддержанию относительного постоянства значений регулируемых параметров (уровня жидкости, давления, числа оборотов и т. п.).

В более позднее время появились *оптимальные* регуляторы, позволяющие при изменении внешних условий так изменять значения регулируемых параметров, чтобы поддерживать наиболее выгодный режим ведения регулируемого процесса (самый производительный, экономичный и т. п.). Как и во всех специализированных регуляторах прошлого, в оптимальных регуляторах регулировалось небольшое число параметров.

Внедрение в АСУТП электронных вычислительных машин неограниченно расширяет возможности автоматического управления. Необходимо отметить четыре принципиально новых момента,

связанных с использованием ЭВМ: 1) резкое увеличение числа регулируемых параметров (от 2—5 параметров в обычных регуляторах до многих сотен, а в отдельных случаях даже тысяч); 2) возможность осуществления сложных алгоритмов управления; 3) универсальность, способность быстро перестраиваться (за счет изменения программ) на новые алгоритмы управления; 4) наличие памяти, позволяющее учитывать при управлении не только нынешнее состояние регулируемого объекта, но и его предысторию.

В АСУТП обычно употребляются специально создаваемые для этой цели *управляющие* ЭВМ. Имеется два основных типа таких ЭВМ. *Цифровые дифференциальные анализаторы* — специализированные вычислительные устройства, предназначенные для решения систем обыкновенных дифференциальных уравнений. Они используются для замены обычных автоматических регуляторов; одна такая ЭВМ способна заменить много регуляторов.

Второй тип управляющих машин строится по тем же принципам, что и ЭВМ общего назначения, описанные в гл. I. Отличие состоит в способах ввода и вывода данных. Обычное перфокарточное и другое универсальное оборудование играет здесь вспомогательную роль (употребляется в основном для ввода программ), число таких устройств сводится к минимуму (одно-два устройства). Значительно менее развиты и устройства внешней памяти, да и оперативное ЗУ имеет обычно значительно меньший объем.

Зато управляющие ЭВМ имеют специальное *устройство связи с объектом* управления (УСО). УСО снабжается большим числом каналов (от нескольких десятков до нескольких тысяч), по которым могут передаваться не только цифровые, но и аналоговые сигналы, представленные электрическим напряжением различной величины (от микровольт до сотен вольт), силой тока, частотой, фазой и т. п.

В УСО встроены специальные преобразователи типа аналог-цифра, позволяющие переводить сигналы в цифровые коды ЭВМ, и типа цифра-аналог, осуществляющие обратные преобразования. К УСО могут придаваться также другие преобразователи, устанавливаемые непосредственно на объекте управления, например, устройства для преобразования в цифровой код угла поворота вала, датчики давлений, специальные реле для включения различных механизмов и т. п.

Управляющая ЭВМ через УСО может опрашивать датчики, присоединенные к входным каналам, и выдавать управляющие воздействия на объект через выходные каналы УСО. Управляющие ЭВМ строятся в малогабаритном, транспортабельном исполнении и обладают повышенной надежностью. В Советском Союзе выпускаются управляющие ЭВМ Днепр-1, Днепр-2, УМ-1, М-6000 и др.

Главная задача АСУТП, использующих управляющие ЭВМ,

заключается в оптимизации сложных технологических процессов, характеризующихся большим числом параметров и сложностью алгоритмов управления. При этом достигается увеличение производительности, экономичности процессов, уменьшение брака и т. п.

В первый период развития АСУТП, когда управляющие ЭВМ не отличались еще высокой надежностью, система управления строилась по двухуровневому принципу. На нижнем уровне использовались обычные регуляторы, которые обеспечивали устойчивость процесса и невозможность его выхода на аварийный режим. На верхнем уровне управляющая ЭВМ обеспечивала расчет наиболее эффективных режимов управления по тем или иным критериям экономического характера сразу для многих регуляторов. Преимущество этого способа состоит в том, что при выходе из строя ЭВМ или случайной ошибке в ее работе (так называемом сбое) произойдет лишь ухудшение экономических показателей процесса при сохранении его устойчивости и гарантии от аварии.

В процессе работы ЭВМ выдает *уставки* «подведомственным» ей регуляторам. Под уставками понимаются те значения регулируемых параметров, которые должны поддерживать соответствующие регуляторы для достижения рассчитанного эффекта. Уставки эти могут передаваться на регуляторы либо автоматически, либо через оператора. Во втором случае ЭВМ работает в *режиме советчика*, печатая оператору указания о том, как следует вести процесс, сообщая ему своевременно, какие из регулируемых параметров и насколько нужно изменить. Поскольку в большинстве случаев режимы изменяются не слишком часто, оператор может вполне справляться с выполнением всех этих указаний и не быть узким местом системы управления.

В настоящее время увеличение надежности управляющих ЭВМ приводит к тому, что они все в большей и большей мере берут на себя и задачи регуляторов нижнего уровня. В простых системах одна ЭВМ может справляться одновременно с работой на обоих уровнях. В сложных системах возможна иерархия управляющих ЭВМ, когда функции первичных регуляторов выполняют простые и надежные ЭВМ, а управление на высоких уровнях поручается более сложным машинам.

Вторая важнейшая задача, для которой строятся АСУТП — это так называемое *программное управление*. Как известно, в промышленности все большее и большее применение получают станки с программным управлением. Работа таких станков (процесс взаимного перемещения режущего инструмента и обрабатываемой детали) управляется специальными командами, записанными на перфоленте или магнитной ленте. Преимущество станка с программным управлением перед ручным трудом — это прежде всего скорость работы, отсутствие пауз для обдумывания предстоящих операций

при сложных работах. По сравнению же со специализированными станками-автоматами преимущество программного управления — его универсальность: простой заменой ленты станок с программным управлением перестраивается на любую другую работу.

Слабым местом станков с программным управлением является необходимость ручного труда при смене режущего инструмента, установке заготовки и съеме готовой детали. Поэтому в настоящее время в различных странах ведутся работы по созданию универсального программно-управляемого подъемно-транспортного оборудования — роботов. Робот представляет собой тележку, на которой установлен универсальный захват типа механических манипуляторов с 5—7 степенями свободы, напоминающий человеческую руку. Движения этой руки могут контролироваться специальными телевизионными камерами — «глазами» робота.

Наличие программно-управляемого обрабатывающего оборудования (станков, прессов и т. п.) в совокупности с программно-управляемыми роботами позволяет строить цехи и заводы-автоматы, принципиально отличные от заводов-автоматов недавнего прошлого (как, например, завод по выпуску автомобильных поршней). Последние были настроены на выпуск строго определенной продукции. Смена продукции фактически означала необходимость полной замены оборудования. В противоположность этому программно-управляемые заводы-автоматы обладают предельной гибкостью: практически мгновенно они способны перестроиться на выпуск любой продукции, удовлетворяющей лишь естественным ограничениям по габаритам, весам и возможным типам обработки деталей. На таком заводе координацией действий обрабатывающего оборудования и роботов занимается специальная управляющая ЭВМ. Через эту же машину осуществляется изменение программ для всех программно-управляемых единиц оборудования.

Принципы программного управления применяются также в приборостроении, например, при изготовлении интегральных схем для ЭВМ, при управлении гальваническими цехами (рациональное распределение обрабатываемых деталей между ваннами, управление режимами обработки), в малой химии и во всех других случаях, когда необходимо обеспечить возможность быстрой перестройки производства на выпуск новой продукции. В последнее время предприняты успешные попытки использования программно-управляемых роботов на сборочных конвейерах, в автоматизированных складах и т. п.

Важной составной частью программно-управляемой технологии являются программно-управляемые испытательные стенды, осуществляющие автоматические испытания как готовых изделий, так и их составных частей на разных стадиях обработки. В сложных изделиях, характеризующихся многими десятками и сотнями пара-

метров, для автоматизации процесса испытаний применяются управляющие ЭВМ. Применение автоматизации резко уменьшает сроки испытаний, поскольку ЭВМ успевает одновременно контролировать и анализировать большое число параметров, которые раньше приходилось анализировать последовательно, один за другим.

Выигрыш во времени для сложных испытаний может достигать многих тысяч раз (минуты и даже секунды вместо многих часов и дней). При этом очень важным обстоятельством является универсальность стенда: необходимость строить новые специализированные стенды для каждого нового изделия оказывается, как правило, очень накладной.

Для того чтобы в полной мере реализовать преимущества программно-управляемой технологии, необходимо автоматизировать подготовку программ, как для отдельных единиц оборудования, так и для всей технологии в целом. В настоящее время разработаны специальные проблемно-ориентированные языки для описания сложных геометрических поверхностей и отдельных элементов их обработки. Любому технологу, совершенно не знакомому с ЭВМ, сравнительно нетрудно освоить такие языки, после чего он может писать программы обработки достаточно сложных деталей за время, исчисляемое минутами или, в худшем случае, десятками минут. Разумеется, написанные программы нуждаются в трансляции, которая выполняется на обычных универсальных ЭВМ.

Следующий шаг — это соединение программно-управляемой технологии с *автоматизированными системами проектирования*. До последнего времени такие системы ориентировались в основном на традиционные формы проектно-конструкторской и технологической документации: чертежи, описания, спецификации, технологические карты и т. п. Однако сейчас все в большей и большей степени в выходную продукцию автоматизированных систем проектирования включаются программы для различного программно-управляемого оборудования. Таким образом, например, строятся системы автоматического проектирования деталей корпуса судов, выходом которых являются данные, уже записанные на магнитные ленты для газорезательных станков с программным управлением. Станки с установленными на них лентами автоматически вырезают из листа проката спроектированные детали.

Автоматизированные системы проектирования для сложных изделий строятся как человеко-машинные системы. Составляющая основу системы мощная ЭВМ (или даже система из нескольких ЭВМ) снабжается специальными дисплеями (экранными пультами), представляющими собой рабочие места конструкторов. Для системы разрабатывается специальное математическое обеспечение, осуществляющее диалог конструктора с ЭВМ.

В состав этого математического обеспечения входят обычно специальные макрооператоры для работы с чертежами: «повернуть на заданный угол», «изменить масштаб», «сделать разрез», «перевести из прямоугольной проекции в изометрическую» и т. п. Пользуясь световым карандашом, конструктор может вносить в чертежи различные правки, вводить в ЭВМ эскизы, которые, после ввода, (со специальной клавиатуры на пульте) необходимых размеров ЭВМ автоматически превращает в чертеж, могущий по желанию конструктора быть отображенным на экране.

Чертежи и их отдельные части, хранимые в памяти ЭВМ, получают специальные имена, пользуясь которыми конструктор может вызывать необходимую информацию на экран и давать задание на автоматическое вычерчивание чертежей, печатание спецификаций и т. п.

В распоряжении конструктора находится также система макрооператоров для работ расчетного характера: «определить центр тяжести или вес детали», «подсчитать механические усилия в конструкции», «определить токи и напряжения в электрической схеме» и т. п. Набор таких макрооператоров (задаваемых стандартными программами) зависит от специфики проектно-конструкторских работ, на которые ориентирована данная система. Он должен быть, однако, достаточно полным, чтобы обеспечить выполнение всех расчетных работ, необходимых для проектирования, а также для оформления чертежной и другой документации. Часто приходится объединять в одной системе не только конструкторскую, но и технологическую проработку. В этом случае также необходимо соответствующее расширение состава макрооператоров.

Большую пользу в автоматизированных системах проектирования могут принести архивы (на машинных носителях) прежних разработок в соединении со специальной справочно-информационной системой. Смысл подобной системы заключается в том, что конструктор может, описав свойства нужной ему детали, вызвать на экран чертежи всех имеющихся в архиве деталей, которые удовлетворяют этому описанию. Если среди них он найдет удовлетворяющую его конструкцию, то он не только просто сэкономит время на проектировании, самое главное состоит в том, что заводу, обычно, гораздо проще возобновить (или увеличить) выпуск деталей, с которыми он уже имел дело, чем организовать производство совершенно новой детали. Выгодность использования накопленного задела еще более возрастает, если речь идет не об отдельных деталях, а о целых узлах и блоках.

Переходим к общему описанию автоматизированных систем организационного управления. Основная задача таких систем состоит в том, чтобы обеспечить оптимальное функционирование объекта управления (завода, объединения, отрасли и т. п.), как

единого целого, за счет: правильного выбора целей и путей их достижения, наилучшего распределения заданий между отдельными частями, из которых состоит объект, и обеспечения их четкого взаимодействия. Попутно АСОУ должна решать задачи учета, отчетности, оплаты труда и другие задачи, которые мы будем называть *рутинными*.

Важное значение имеет также справочно-информационная функция АСОУ, т. е. выдача в соответствии с поступающими запросами различного рода справок о состоянии объекта управления (выполнение плана по участкам, наличие запасов на складах и т. п.). Целесообразно иногда создавать специальные автоматизированные справочно-информационные системы.

Основной эффект, который дают АСОУ, возникает за счет полноты, своевременности и оптимальности принимаемых решений и, как следствие этого,— ликвидации различного рода организационных неполадок. Второй важной, но обычно гораздо менее весомой долей эффекта, даваемого АСОУ, является экономия управленческого труда, позволяющая уменьшать управленческий аппарат (или затормозить его рост) без ущерба для качества управления.

Правильно спроектированная АСОУ должна полностью ликвидировать, или, по крайней мере, свести к минимуму, потери, возникающие вследствие причин организационного характера. А эти потери могут быть очень большими, если не выполнено хотя бы одно из трех сформулированных выше непременных условий, которым должно удовлетворять каждое решение, а именно полнота, своевременность и оптимальность.

Приведем простейшие примеры потерь, возникающих при неудовлетворении этих условий. Пример 3. Предположим, что литейному и механическому цехам машиностроительного завода заданы месячные планы производства, увязанные по объемам и номенклатуре, но не увязанные по точным срокам выпуска деталей каждого типа, как между собой, так и со сборочным цехом. Это — пример неполного решения, практически неизбежным результатом которого являются периодически возникающие на сборке дефициты изделий. Это влечет за собой штурмовщину в механическом и литейном цехах, частые переналадки оборудования, потери рабочего времени, увеличение брака.

Пример 4. Завод выпускает какой-либо предмет народного потребления (например, швейные машины), на который все время имелся хороший спрос. Решение задачи прогнозирования спроса запоздало, и в какой-то момент произошло затоваривание. Приходит решение прекратить производство данного предмета, но продукция на многие миллионы рублей уже выпущена и продолжает мертвым грузом лежать на складах. Это — пример несвоевременного решения.

Пример 5. Предположим, что министерство спустило заводам подробный план, полностью обеспеченный ресурсами, идеальным образом наладило кооперацию и материально-техническое снабжение. Отрасль работает с точностью часового механизма, сделанные рутинные расчеты показывают, что план увеличить больше невозможно. Оказывается, что использование современных

математических методов даже в этих условиях за счет оптимального перераспределения планов между заводами могло бы увеличить выпуск продукции на 10%, а то и больше. Это пример полного, своевременного, но неоптимального решения.

Из приведенных примеров ясно, что усилия разработчиков должны быть направлены прежде всего на то, чтобы АСОУ помогали в выработке полных, своевременных и оптимальных решений. Полная оптимизация далеко не всегда достижима даже с помощью ЭВМ. Однако следует прилагать усилия для достижения хотя бы приближенной оптимизации.

Следует подчеркнуть, что задачи организационного управления (особенно для постановки целей) отличаются большой комплексностью и должны решаться не изолированно (как это можно до известной степени делать в АСУТП), а в общенациональном масштабе.

Общегосударственная автоматизированная система управления экономикой (ОГАС) никоим образом не может быть сведена к простой сумме автоматизированных систем управления предприятиями (АСУП) и отраслевых автоматизированных систем управления (ОАСУ). В частности, постановка целей перед системами низших уровней, как правило, может быть выполнена лишь в результате рассмотрения целей систем более высоких уровней.

Автоматизированные системы организационного управления строятся на базе универсальных ЭВМ общего назначения, снабженных внешней памятью (на магнитных лентах и дисках) большого объема.

Важную роль в правильном функционировании АСОУ имеют *системы сбора и передачи данных*, в которых широко используются периферийные устройства и машинные носители информации. С каждым годом все большее и большее значение приобретает передача данных по каналам связи и непосредственный обмен данными между ЭВМ, входящими в различные АСОУ.

Непременным элементом каждой АСОУ является *информационная база*, состоящая из массивов, которые описывают объект управления (оборудование, кадры, нормы, планы, запасы данных и т. п.). Для постоянного обновления базы и работы с информационными массивами (с целью подготовки данных для решения различных задач управления) создается специальное математическое обеспечение.

Следующим элементом является совокупность рабочих программ, решающих различные задачи планирования и управления (включая подготовку соответствующих документов). Наконец, для успешной работы АСОУ оказывается необходимым проведение целого комплекса мероприятий организационного и экономического характера (изменение организационной структуры, систем ответственности и стимулирования и т. п.). Таким образом, создание АСОУ отнюдь не сводится просто к покупке и установке ЭВМ.

2. Основные принципы построения автоматизированных систем организационного управления

Автоматизированные системы организационного управления сначала применялись для решения рутинных задач управления: бухгалтерского учета, составления различного рода сводных ведомостей и других отчетных документов. Объясняется это тем, что до появления первых ЭВМ для этой цели употреблялись *счетно-аналитические* (перфорационные) машины, не способные производить сложные оптимизационные расчеты. Когда настала эра ЭВМ, они постепенно вытеснили старую перфорационную технику, но естественным образом восприняли разработанные для нее задачи и технологию обработки данных*.

Однако автоматизация только рутинных задач управления не дает коренного эффекта в улучшении организационного управления. Эффект же, получаемый от экономии управленческого труда, далеко не всегда оправдывает установку сложной и дорогостоящей электронной вычислительной техники. Поэтому один из первых принципов, которым нужно руководствоваться при создании АСОУ,— это так называемый *принцип новых задач.*

Суть этого принципа состоит в том, чтобы не просто перекладывать на ЭВМ традиционно сложившиеся методы и приемы управления, а перестраивать эти методы в соответствии с новыми огромными возможностями, которые дают ЭВМ. На практике это означает, что при анализе объекта управления выявляются потери, происходящие от недостатков организационного управления (простои, нерациональная загрузка оборудования и т. п.). В соответствии с результатами анализа намечается список задач, которые, ввиду их большого объема, сейчас не решаются или решаются неполно, но которые можно решать с помощью ЭВМ. Как уже отмечалось в предыдущем параграфе, эти задачи должны быть направлены на то, чтобы обеспечить полноту, своевременность и оптимальность (хотя бы приближенную) принимаемых решений.

Эти задачи, разумеется, различны для объектов различных типов. Для машиностроительных и приборостроительных предприятий наиболее важными оказываются задачи оперативно-календарного и объемно-календарного планирования. Решающий эффект получается в том случае, когда на уровне сменных заданий точно согласуются по времени все производственные и обеспечивающие (материально-техническое снабжение) операции, определяются оптимальные объемы партий и производится оптимизация

---

* В современных АСОУ иногда продолжают употреблять счетно-аналитические машины для предварительной обработки информации на входе ЭВМ.

загрузки оборудования. В строительстве возникают близкие к этим задачи. В ряде случаев на первый план выходят задачи технической подготовки производства, управление проектно-конструкторскими работами.

Для химических предприятий важнейшей задачей является оптимизация режимов в масштабе всего предприятия (в зависимости от текущих характеристик сырья, состояния оборудования и степени заполнения буферных емкостей). На транспорте центральное значение приобретает оптимизация маршрутов и расписаний, а также четкая организация погрузочно-разгрузочных работ. В торговле наиболее критическим вопросом является детальное прогнозирование спроса (на основе полного учета текущей продажи по всей номенклатуре) и выдача обоснованного заказа. В сельском хозяйстве — это могут быть задачи оптимизации рационов кормов, задачи наиболее рационального распределения имеющихся удобрений (в зависимости от почв, культур и прогнозируемых погодных условий) и др.

В отраслевых системах важнейшее значение приобретает задача оптимизации распределения плана между предприятиями, точное согласование и управление сроками взаимных поставок, а также задачи прогнозирования и перспективного развития отрасли.

Тесно связан с принципом новых задач второй основной принцип — *принцип комплексного* или, более точно, *системного подхода*, к проектированию АСОУ. Проектирование АСОУ должно основываться на системном анализе как объекта, так и системы управления им. Это означает, в частности, что должны быть определены цели и критерии для функционирования объекта (вместе с системой управления) и проведена структуризация, вскрывающая *весь комплекс вопросов*, которые необходимо решить для того, чтобы проектируемая система наилучшим образом соответствовала установленным целям и критериям.

В этот комплекс попадают вопросы не только технического, но также *экономического* и *организационного* порядка. Внедрение АСОУ открывает принципиально новые возможности для коренного усовершенствования системы экономических показателей и экономического стимулирования за счет детального учета и использования ЭВМ. И без такого усовершенствования возможности АСОУ, как правило, полностью использованы быть не могут.

Приведем несколько характерных примеров. Оптимизация планов транспортных перевозок методами, рассмотренными в предыдущей главе намного увеличивает эффективность использования транспорта. Однако, если основным показателем плана и системы стимулирования остаются тонно-километры, такая оптимизация не сможет успешно применяться.

Второй пример относится к применению сетевых графиков

в строительстве. Являясь мощным инструментом контроля и управления, сетевой график может в то же время быть оторванным от системы финансирования строительных организаций. Иными словами, строители получают деньги не тогда, когда достигается то или иное событие на сетевом графике, а по выполнению этапов, которые прямо с сетевым графиком не связаны. Поскольку система стимулирования не вытекает непосредственно из сетевого графика, его эффективность снижается, а в ряде случаев и вовсе сводится к нулю.

Третий пример. На одном из приборостроительных заводов в АСОУ заложена возможность: по результатам испытаний готовой продукции определять конкретных виновников брака, допущенного на любом производственном участке много дней тому назад. Ясно, что если не перестроить соответствующим образом систему стимулирования, эта возможность АСОУ не будет должным образом влиять на снижение процента брака.

Внедрение АСОУ должно сопровождаться крупными мероприятиями организационного характера: изменением привычных (а иногда даже узаконенных) форм документов, изменением структуры органов управления, функциональных обязанностей управленческого аппарата, форм контроля и ответственности не только собственного аппарата, но также и аппарата смежников. Не всегда все вопросы могут быть решены в рамках самой разрабатываемой системы. Принцип системного подхода требует, чтобы все такие вопросы были выявлены и поставлены в соответствующих инстанциях.

Приведем только один пример, касающийся форм документов. Эффективность создающихся и уже созданных АСОУ в автомобильном транспорте могла бы быть значительно увеличена при объединении всех документов, касающихся перевозимых грузов, и путевого листа в один документ типа дуаль-карты. Такие карты могли бы готовиться с помощью ЭВМ, снабжаться соответствующими отметками в пунктах погрузки и разгрузки грузов и снова поступать в ЭВМ в качестве единых учетных документов. При необходимости ЭВМ могла бы готовить дубликаты документов для грузоотправителей и грузополучателей после возвращения заполненной карты. Эффективность такого мероприятия выросла бы еще больше, если бы промышленность выпускала грузовики с весами (встроенными в рессоры) и счетчиком километража, способными автоматически (при вставлении карты в специальную щель) пробивать на ней свои показания.

Из приведенных примеров видно, что успешная реализация двух сформулированных принципов делает *совершенно необходимым*, чтобы *заказ* на АСОУ, а также ее *разработка* и *внедрение* производились под *непосредственным руководством первого руководителя* соответствующего объекта (завода, объединения, мини-

стерства и др.). В этом состоит *принцип первого руководителя*, без соблюдения которого успешная разработка АСОУ становится невозможной. Отечественная и зарубежная практика с достаточной убедительностью свидетельствует, что всякая попытка передоверить дело создания АСОУ второстепенным лицам неизбежно приводит к тому, что система ориентируется на рутинные задачи управления и не дает ожидаемого эффекта.

При создании АСОУ, как и вообще при реализации крупных проектов, необходимо иметь четкую систему распределения обязанностей и взаимодействия *заказчика* и *исполнителя*. Функция заказчика (завода, министерства и любого другого объекта автоматизации) состоит прежде всего в том, чтобы самостоятельно или с помощью специальной консультационной группы, выделяемой исполнителем, сформулировать *цели*, *критерии* и *общую концепцию* проектируемой системы. Заказчик (на основе консультаций с исполнителем) определяет приоритеты и очередность ввода различных задач управления, которые должна решать система. Заказчик должен также принимать *непосредственное* участие в создании *информационной базы системы* (нормативы, кадры, оборудование и т. п.) и осуществлять соответствующие организационные мероприятия (изменения структуры, функциональных обязанностей управленческого аппарата и его обучение), чтобы к моменту ввода отдельных частей системы работники аппарата полностью владели бы машинными массивами информации и методами машинного решения задач управления.

Что же касается разработки проекта и создания технической базы системы (включая систему сбора и передачи информации), системного математического обеспечения, состава и форм информационных массивов, а также программ решения конкретных задач управления, то все эти вопросы должны решаться исполнителем.

Такая организация работы помогает реализовать еще один важный принцип разработки АСОУ — принцип *максимальной разумной типизации проектных решений*. Суть его состоит в том, что, разрабатывая столь трудоемкие и дорогостоящие вещи, как технический комплекс, системное математическое обеспечение, рабочие программы и связанные с ними формы и состав информационных массивов, исполнитель обязан стремиться к тому, чтобы предлагаемые им решения подходили бы возможно более широкому кругу заказчиков. Однако хорошо известно, что попытка создать одну типовую универсальную программу решения задач оперативно-календарного планирования приведет к тому, что в каждом конкретном случае она будет работать значительно медленнее, чем программа, специально приспособленная к особенностям данного предприятия.

Разумность типизации означает разумный процент замедления

работы типовой программы по сравнению со специализированной. Величина этого процента, а значит, и количество типовых решений зависят от количества разработчиков АСОУ в стране. При увеличении этих сил можно позволить себе роскошь иметь более высокий уровень специализации*. Типизация решений способствует концентрации сил, что необходимо для создания действительно комплексных АСОУ. Следует заметить также, что зачастую типовая программа, составленная опытными программистами, оказывается эффективнее специализированных программ, создаваемых небольшими группами специалистов, не имеющих столь высокой программистской квалификации.

Правильно проведенная типизация проектных решений в АСОУ не только не препятствует, а наоборот, способствует увеличению возможностей учета индивидуальных особенностей систем стимулирования или любых других элементов систем управления. Чтобы понять это парадоксальное, на первый взгляд, утверждение, лучше всего обратиться к примеру, уже разобранному в гл. I. Имея одну типовую программу для решения алгебраических уравнений любых степеней, пользователь обладает гораздо большими возможностями приспособиться к индивидуальным особенностям интересующих его уравнений, чем в случае, когда он располагает десятком специализированных программ для решения уравнений различных конкретных степеней.

Типизация решений имеет и другую сторону: для ускорения разработки исполнитель должен максимально использовать решения, полученные при разработке других систем. Важность этого замечания легко понять, если вспомнить, что разработка действительно комплексной АСОУ имеет трудоемкость в несколько тысяч человеко-лет. При наличии же достаточного запаса типовых решений, которые остается лишь привязать к данным конкретным условиям, эта трудоемкость снижается в десятки раз.

Различные составные элементы системы имеют различный уровень типовости — наиболее высокий у технических комплексов и системного матобеспечения, наиболее низкий — у рабочих программ. Поэтому типовой проект, охватывающий все машиностроительные предприятия, компонуется по принципу *функциональной избыточности*: несколько типовых рабочих программ на одну и ту же задачу. При привязке системы достаточно выбрать из этого набора программ наиболее подходящую.

При проектировании АСОУ важно соблюдать еще один принцип — *принцип непрерывного развития системы*. По мере развития

* Стремление к максимальной типизации характерно для любой области техники в начале ее развития (достаточно вспомнить тракторы, автомобили и т. п.). По мере же накопления сил и достаточного рынка сбыта проявляются тенденции к большей специализации.

как экономики в целом, так и отдельных предприятий, возникают новые задачи управления, совершенствуются и видоизменяются старые. Для того чтобы АСОУ могли быстро реагировать на эти изменения, она должна снабжаться возможностями автоматизации программирования и перекомпоновки информационных массивов. Необходима также специальная служба, осуществляющая сбор и распространение новых программ и других усовершенствований, вносимых в систему пользователями.

Комплексы рабочих программ должны строиться таким образом, чтобы при необходимости можно было легко менять не только отдельные программы, но и критерии, по которым ведется управление. Полная реализация подобной гибкости АСОУ возможна на основе динамических моделей объектов управления, описанных в следующих разделах этой главы.

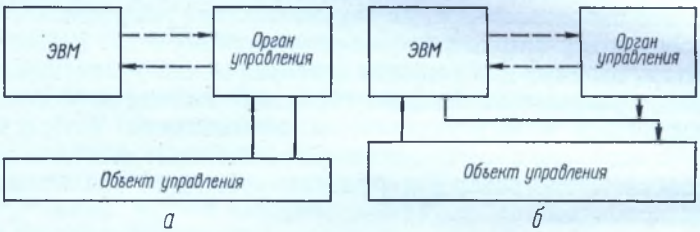

Рис. 26.
Схемы, поясняющие принцип автоматизации документооборота:
а — простой режим ЭВМ; *б* — системный режим

Принцип комплексного (системного) подхода распространяется не только на перечисленные выше вопросы, но и на саму организацию переработки информации в АСОУ. Здесь также можно сформулировать ряд принципов, характеризующих системный подход к процессу переработки информации.

Прежде всего следует отметить так называемый *принцип автоматизации документооборота*. Суть этого принципа легко уяснить из рис. 26, на котором сплошными линиями показаны пути движения документов без автоматизации документооборота и при наличии такой автоматизации; пунктирными линиями — запросы, задания и ответы ЭВМ.

Схема рис. 26, *а* соответствует простому использованию ЭВМ в интересах управления. Поток документов, как и до установки ЭВМ, идет непосредственно через орган управления. ЭВМ применяется для решения отдельных сложных задач управления, полученные результаты используются при подготовке выходных документов. В системном режиме (рис. 26, *б*) основной поток документов идет через ЭВМ, необходимые сведения (справки) об объекте

управления управленческий аппарат получает из ЭВМ по запросам. Результаты решения задач представляются машиной в виде окончательных выходных документов (печатаемых на АЦПУ), которые, разумеется, подвергаются контролю (и, в случае необходимости, отбору) со стороны управляющего органа. Управленческий аппарат сосредоточивает свои усилия на постановке целей и внесении творческого элемента в поиски путей их достижения (например, в результате построения динамических моделей методами системного анализа).

В реальных условиях часть документов (письма личного характера, жалобы и предложения, различного рода неформализованные сообщения и т. п.) продолжают рассматриваться органом управления непосредственно, хотя содержащиеся в них факты, а иногда и сами документы могут накапливаться ЭВМ в архивах на машинных носителях.

Следующим важным принципом, характеризующим системный подход к обработке информации, является *принцип единой информационной базы.* Смысл его состоит прежде всего в том, что на машинных носителях накапливается (и постоянно обновляется) информация, необходимая для решения не какой-то одной или нескольких задач, а всех задач управления. При этом в основных массивах исключается неоправданное дублирование информации, которое неизбежно возникает, если первичные информационные массивы создаются для каждой задачи отдельно. Действительно, возьмем, например, данные о запасах материалов на складах. Они нужны и для задач планирования, и для бухгалтерского учета, и для приготовления различного рода справок, отчетов и т. д. В случае их дублирования во многих массивах возникает опасность увеличения числа первичных документов (особенно если задачи и массивы для них строятся разными людьми без должной координации). Вторая опасность — это возможность разночтений, когда одна и та же величина (например, размер запаса какого-либо материала) в одних массивах имеет одно значение, а в других — другое. Основной массив (или, как его еще иначе называют, *мастер-файл*) служит своеобразным эталоном, по которому можно сверять и, при необходимости, восстанавливать или создавать заново любые рабочие массивы.

Основные массивы образуют *информационную модель* объекта управления. Их состав и форма определяются принятой концепцией управления. На уровне предприятий основные массивы должны содержать самую подробную информацию: кадровый массив на всех работающих (включающий в себя подмассивы руководящих и управленческих кадров), массивы основных фондов (земля, помещение, оборудование со всеми характеристиками, необходимыми для решений о его использовании, перераспределении

и т. п.), массивы запасов (оборотных фондов в материальном выражении) по самой подробной номенклатуре, включая запасы на промежуточных складах и незавершенное производство, массивы текущего состояния оборудования (включая режимы работы и причины остановок), массивы нормативов (трудовых и материальных) и технологических маршрутов (последовательностей производственных операций, необходимых для изготовления материалов, деталей, узлов и готовых изделий, увязанных с нормативами), массивы планов (включая заявки на материально-техническое снабжение), массивы цен и расценок, массив текущего состояния банковских счетов предприятия и др.

Система первичных документов, а также система различных автоматических датчиков должна быть построена таким образом, чтобы любое изменение, происходящее на предприятии в минимально короткий срок, вводилось в ЭВМ в специальный текущий *массив изменений*. Операционная система АСОУ автоматически или по указанию оператора периодически разносит эти изменения по основным массивам, поддерживая их постоянно в состоянии готовности для выдачи любой информации о состоянии объекта.

Для уменьшения времени на эти операции разрабатываются рациональные формы упорядоченности основных массивов, согласованные с системой первичных документов: если документы, характеризующие оборудование (паспорта) привязаны к цехам (т. е. снабжаются классификационным признаком цеха), то массив оборудования можно также рационально упорядочить по цехам, разбив его на соответствующие подмассивы. Другой путь — это такая организация основных массивов, при которой все изменения просто приписываются до поры до времени в конец соответствующих массивов, а фактическое занесение изменений на соответствующие места делается сравнительно редко.

При недостаточной мощности ЭВМ при использовании в качестве первичных входных документов перфокарт применяется их предварительная сортировка (в соответствии с составом основных массивов), с помощью механических или электронных сортировок, входящих в комплекты счетно-аналитических машин. В соответствии с признаками, пробитыми в определенных колонках перфокарт, сортировальная машина направляет их в различные карманы, разделяя поток перфокарт на определенные группы, более удобные для последующего использования.

Как уже отмечалось выше, основные массивы играют роль эталонов для всего остального информационного хозяйства АСОУ. Необходимо поэтому принимать специальные меры для обеспечения их идеального состояния: дублирование на случай порчи или утери, периодические проверки их содержания (инвентаризации) и т. п. Должна быть организована специальная служба

массивов, выделены ответственные «хранители» массивов. Наличие их особенно важно в том случае, когда на вход системы поступает противоречивая информация (например, о конъюнктуре на мировом рынке). Ответственный за соответствующий массив должен решать, какую информацию вводить: выбрать ли ее однозначно или ввести противоречивые сведения с указанием источников, из которых они поступили, дополнив их, быть может, собственной оценкой достоверности. Этот пример показывает, что хранитель массива — не просто технический работник. В ряде случаев им может и должен быть достаточно ответственный сотрудник управленческого аппарата.

Важнейший принцип организации информационной базы — ее *гибкость*. Это означает, что помимо основных массивов необходимо иметь инструмент, позволяющий на их основе создавать и поддерживать любое количество постоянных или временных *рабочих массивов*. Таким инструментом является специальное *системное математическое обеспечение* (операционная система и библиотека стандартных программ), позволяющее производить различные преобразования массивов (слияния, подборки, сортировки, сокращения, обеспечение нужной формы вывода и т. п.).

Необходимость рабочих массивов обусловливается тем, что основные массивы, будучи эталонами для воспроизведения любой нужной информации в системе, сами не очень удобны для непосредственного использования при решении задач управления. Массив нормативов включает нормативы на все детали или изделия, производящиеся (недавно производившиеся) на данном предприятии.

При составлении плана производства каких-то определенных изделий весь этот массив не нужен (при его непосредственном использовании в задаче оптимизации плана ушло бы много времени на непроизводительную перекрутку лент). Поэтому из основного массива вырезается необходимая часть, которая и составляет необходимый рабочий массив для решения данной конкретной задачи. Роль основного массива сводится к порождению временного рабочего массива.

Другой пример: в основном кадровом массиве содержится много сведений, необходимых по положению о кадровом учете. Для целей же планирования достаточен минимум сведений: учетный номер, принадлежность к определенному цеху, бригаде или смене, профессия, квалификация и, быть может, — стаж работы. Ясно, что массив с такими данными будет употребляться столь часто, что его целесообразно перевести в разряд постоянных и обновлять наряду с основными массивами. Роль основного массива, помимо первоначального порождения*, состоит в том, что с ним

---

* Не исключено, что подобные постоянные рабочие массивы рождаются независимо и даже раньше соответствующих основных массивов.

осуществляется периодическая сверка и устранение возможных расхождений.

Соблюдение принципов единства и гибкости информационной базы АСОУ сильно облегчает задачу их дальнейшего совершенствования и развития. Особенно следует подчеркнуть следующее обстоятельство: наличие полной информации об объекте управления, представленной в какой-нибудь (пусть даже случайно выбранной) форме, вместе с матобеспечением, способным ее любым сбразом перестраивать, неизмеримо облегчает задачу унификации информации при последующем объединении ранее созданных локальных АСОУ в общегосударственную систему.

Такой подход к организации информационной базы облегчает также задачу замены технической базы. Для этой цели необходимо потребовать, чтобы новые ЭВМ снабжались соответствующим системным матобеспечением и были бы способны работать с магнитными лентами и кипами магнитных дисков старых машин. В то время как старая ЭВМ продолжает обеспечивать работу АСОУ, новая машина, работая с основными массивами АСОУ, автоматически воссоздает для себя информационную базу системы в нужной для нее форме. Рабочие же программы системы должны быть написаны на ПО-языке с использованием стандартных макрооператоров системного математического обеспечения, в результате чего они также могут быть перетранслированы на новую ЭВМ без затрат ручного труда. В этом состоит смысл принципа стандартизации систем программирования, который также необходимо соблюдать при грамотном проектировании АСОУ.

Создание информационной базы — одна из самых трудоемких частей разработки АСОУ. Возникает естественный вопрос: каким путем строить эту базу? Существуют два различных подхода к решению этого вопроса: *аналитический* и *синтетический*. При аналитическом сначала создается системное матобеспечение (включая систему автоматизации программирования) и все основные информационные массивы (вместе с системой их обновления), а затем строится система рабочих программ и порождаются необходимые для них рабочие массивы. Преимущество этого метода заключается в том, что он избавляет от работы по ручному составлению рабочих массивов и, вообще, с точки зрения создания системы *в полном объеме* является наиболее экономным. Недостаток же его состоит в том, что при этом методе система долго (пока создаются основные массивы) не дает отдачи: система вводится не этапами, а сразу, целиком.

При синтетическом сначала создаются рабочие массивы для отдельных задач, но с таким расчетом, чтобы система матобеспечения позволила впоследствии скомпоновать из них основные массивы. Недостаток в том, что полностью избежать дублирования и, сле-

довательно, лишней работы при этом не удается. Очевидным же достоинством является возможность ввода системы по этапам и, следовательно, получения более быстрой (хотя и не полной) отдачи.

Возможен и комбинированный путь, когда параллельно идет создание основных и некоторых рабочих массивов. Однако при всех обстоятельствах работа должна быть организована таким образом, чтобы в конце была построена единая информационная база системы, подводящая прочный фундамент под ее будущее развитие и совершенствование. Непременное условие этого пути — опережающее развитие работ по системному математическому обеспечению и автоматизации программирования.

Следующий системный принцип — это *принцип комплексности задач и рабочих программ*. Суть его состоит в том, что большинство задач управления являются комплексными и не могут быть поэтому сведены к простой арифметической сумме мелких задач. Например, задачи материально-технического снабжения органически связаны со всем комплексом задач оперативно-календарного и объемно-календарного планирования: во-первых, само задание на материально-техническое снабжение возникает из задач планирования производства, а во-вторых, при невозможности точно (по срокам и по номенклатуре) выполнить это задание возникает необходимость пересмотра планов. Между задачами планирования производства и материально-технического снабжения происходит *постоянный взаимный обмен информацией*, что превращает их в единый *комплекс задач*. Раздельное их решение может сильно уменьшить эффект АСОУ.

Принцип комплексности задач и рабочих программ является характерным не только для АСОУ, но и для других автоматизированных систем обработки данных (автоматизация проектирования, испытаний и др.). Благодаря наличию программных комплексов информационные потоки в системах организованы более упорядоченным образом, чем в вычислительных центрах общего назначения, в которых потоки задач, а следовательно, и потоки данных случайны. В АСОУ, кроме того, большинство задач управления строго привязано ко времени, т. е. выполняется по определенному расписанию.

Упорядоченность потоков задач и данных позволяет использовать *принцип специализации* (системной ориентации) *операционных систем* и, прежде всего,— программ-диспетчеров. Операционные системы общего назначения начинают готовить данные для задачи, когда программа уже запущена в работу (по макрооператору «открыть массив»). Программа при этом переводится в режим ожидания, в результате чего теряется время (компенсируемое частично лишь за счет мультипрограммирования).

При специализации операционных систем для упорядоченных потоков задач возможна заблаговременная подготовка данных, так что очередная задача, будучи вызвана на счет, уже имеет необходимые для ее работы информационные массивы в открытом состоянии. Опыт показывает, что такая организация вычислительного процесса может значительно (в несколько раз) повысить эффективность работы системы. Особенно большим этот эффект становится при отсутствии в используемой ЭВМ режима мультипрограммирования.

Следует особо подчеркнуть, что принцип специализации операционных систем нисколько не противоречит принципу типизации проектных решений. Под специализацией здесь понимается не приспособление операционной системы к особенностям каждого конкретного объекта, а ее *общесистемная ориентировка*. При этом можно иметь относительно небольшое число типов операционных систем, рассчитанных на широкие классы применений. Конкретные особенности индивидуальных объектов учитываются сменяемыми таблицами, в которых операционной системе сообщается необходимая информация о структуре программных комплексов и расписание решения различных задач управления.

При разработке комплексов задач необходимо исходить из целей и критериев, поставленных перед системой в целом. Само выделение комплексов задач происходит в результате структурирования (на основе методов системного анализа) всей системы управления рассматриваемого объекта. При программировании комплексов задач необходимо иметь в виду, что отдельные составляющие их задачи порождают выходные массивы, являющиеся входными массивами для других задач. Эти промежуточные массивы также получают статут временных или постоянных рабочих массивов и входят в общую информационную базу системы.

Таким образом, при системном подходе, в отличие от решения отдельных задач поодиночке, соблюдается *принцип минимизации ввода и вывода информации*, что имеет большое значение для увеличения эффективности использования ЭВМ: ведь ввод и вывод являются для ЭВМ самым узким местом. Очень важно распространить этот принцип и на обмен информации между различными системами. Для этой цели обычный документооборот, рассчитанный на людей, должен все в большей и большей мере уступать место документообороту на машинных носителях, обеспечивающих быстрый ввод и вывод, т. е. на магнитных лентах и сменных магнитных дисках. Иначе, документ, напечатанный одной ЭВМ и посланный в другую организацию, приходится заново вводить в ЭВМ, что приводит к большим затратам ручного труда и неэффективному использованию машин.

Еще большие возможности открываются при прямом обмене

информацией между ЭВМ по каналам связи, особенно широкополосным. Преимущество этого метода — возможность системного подхода для организации решения комплексных задач междуведомственного характера при совместной работе различных АСОУ.

Принципы комплексности задач и единства информационной базы не исключают возможности существования изолированных подсистем, опирающихся на свои собственные информационные массивы и мало взаимодействующие с основными системными комплексами задач. Такой изолированной подсистемой может оказаться, например, справочно-информационная система по научно-технической информации в отраслевой АСУ или подсистема статистического контроля качества продукции в АСУ предприятий. Ведь ни комплексность задач, ни единство информационной базы не являются самоцелью. Эти принципы должны применяться там, где необходимость комплексности и единства информационной базы вытекает из существа проблемы. А для современных АСУ это имеет место в большинстве случаев. Искусственное раздробление таких комплексов на невзаимодействующие (или слабо взаимодействующие) между собой подсистемы, как правило, значительно снижает эффективность автоматизированных систем управления.

Принцип минимизации ввода и вывода находит свое воплощение не только внутри комплексов задач, но и при поступлении в систему новой информации. Как уже говорилось выше, при системном подходе основная масса информации (информационная модель объекта) постоянно находится внутри системы (на магнитных носителях). Обновление информации в процессе регулярной работы вводится по *принципу ввода изменений* (а не всей информации для решаемых задач целиком). Это сильно уменьшает нагрузку на вводные устройства, но зато в значительной степени увеличивает требования к достоверности вводимой информации: ведь одна ошибка на вводе приведет к ошибке в основных массивах, которая может влиять на решаемые задачи в течение длительного времени (до очередной «инвентаризации» массивов).

Поэтому системный подход накладывает большие требования на систему сбора и передачи первичной информации. Помимо обычных технических проблем повышения надежности, возникает задача устранения возможности субъективных ошибок людей, снабжающих систему первичной информацией. Субъективными ошибками могут быть простые описки, забывчивость или сознательные искажения информации. Для устранения подобных ошибок следует, по возможности, увеличивать долю информации, поступающей от различных автоматических датчиков, применять меры контроля и т. п.

Большую роль в уменьшении возможности ошибок субъективного характера играет *принцип совмещения подготовки документов*

первичного материального учета и первичных финансовых документов с приготовлением соответствующих машинных документов (предназначенных для *автоматического* ввода в ЭВМ). Этого можно добиться различными путями. 1. Объединение обычного и машинного документа в единый документ (дуаль-карты, документы, заполняемые магнитными чернилами, бланки или карты с карандашными отметками и т. п.). 2. Использование устройств, одновременно готовящих обычный документ и его машинную копию (флексорайтеры). 3. Непосредственное подключение к ЭВМ устройств, на которых готовятся первичные документы (телетайпы). Помимо уменьшения возможности ошибок, использование этого принципа устраняет необходимость в дополнительном труде по переносу первичных документов на машинные носители.

Еще один существенный принцип, который надо иметь в виду при проектировании АСОУ,— это *принцип согласованности пропускных способностей отдельных частей системы*. Не имеет смысла, например, увеличивать скорость выполнения арифметических операций ЭВМ, если при решении конкретных задач АСОУ узким местом в системе являются ввод или скорость обмена информации между внешней памятью и центральным процессором. При согласовании пропускных способностей отдельных устройств необходимо принимать во внимание не только номинальную скорость их работы, но и надежность.

В заключение отметим, что использование в управлении ЭВМ и АСОУ приводит к большей централизации обработки информации. Во многих странах уже созданы и продолжают создаваться основанные на использовании ЭВМ банки данных национальных масштабов. В качестве примера можно привести вычислительный центр (ВЦ) налогового управления в США или ВЦ по учету трудовых ресурсов в Японии. При наличии достаточно мощных ЭВМ во многих случаях оказывается целесообразным накапливать огромные массивы данных в одном ВЦ, обращаясь к нему из различных мест с помощью специальных терминалов по каналам связи. Такое решение может оказаться гораздо более эффективным по сравнению с тем, если бы эти данные были разбросаны по отдельным мелким ВЦ.

Однако абсолютно недопустимо смешивать две совершенно различные вещи: *централизацию обработки информации* и *централизацию управления*. Централизация обработки информации есть техническая необходимость такого же порядка, как, например, централизация производства электроэнергии. Что же касается централизации или децентрализации управления — то это прежде всего вопрос экономической и организационной целесообразности, непосредственно с методами обработки информации не связанный.

Широкое использование ЭВМ и АСОУ создает большие возможности для централизации управления, поскольку вышестоящие органы снабжаются при этом полной информацией и возможностью ее быстрой обработки. Но верно также и другое: централизация обработки информации увеличивает возможности и для *децентрализации* управления. В самом деле, предположим, что создана централизованная справочно-информационная система национального масштаба, которая за несколько минут может выдать справку о средней зарплате (в настоящий момент времени!) по любой должностной номенклатуре в любом районе страны или по всей стране в целом*. В таком случае можно децентрализовать функцию установления размеров индивидуальных зарплат, без опасности возникновения инфляции и других нежелательных с общегосударственной точки зрения явлений. Ведь, располагая *оперативной* информацией о *средних* зарплатах, центральные органы управления могут, при необходимости, принять соответствующие меры по устранению причин подобных нежелательных явлений. Если же средние зарплаты остаются в разумных пределах, то необходимости в таких мерах и не возникает.

3. Автоматизированные справочно-информационные системы

Справочно-информационная функция в том или ином виде присуща всем автоматизированным системам организационного управления. По-существу именно эту функцию выполняют специальные системы математического обеспечения, строя из общей информационной базы временные информационные массивы для различных программ управления. Заказчиком на информацию здесь выступает сама автоматизированная система.

Обычно автоматизированными справочно-информационными системами (АСИС) принято называть такие (основанные на ЭВМ) системы, которые выдают ответы по тому или иному кругу вопросов на поступающие в систему индивидуальные запросы людей (например, работников управленческого аппарата). Наделение АСОУ справочно-информационными функциями весьма желательно и даже необходимо всякий раз, когда в системе управления имеются задачи, для решения которых еще не разработаны полностью формализованные машинные процедуры и которые опираются на большой объем информации. В качестве примера такой задачи можно назвать задачу подбора и расстановки кадров, которую приходится решать в каждой системе организационного управления. Необходимость автоматизации справочно-информационной функции при работе

* Такая система имеется в Японии.

с кадрами (как, впрочем, и в любом другом случае) определяется в первую очередь объемом информации, с которой приходится иметь дело, а также частотой и требуемой скоростью решения задач.

Если речь идет о работе с несколькими десятками и даже с сотнями человек, то оказывается достаточным иметь дело с обычными способами представления информации (в виде бумажных документов) или с простыми формами механизации справочно-информационной работы, например, используя *карты с краевой перфорацией*, описанные в параграфе 2 гл. I. Если же необходимо работать с кадровыми массивами на многие тысячи человек, то для увеличения эффективности этой работы приходится прибегать к автоматизации.

Автоматизированные справочно-информационные системы разделяются на два класса: *фактографические* и *документальные**. В простейшем случае фактографическая система имеет дело с одним массивом информации, разделенным на записи *одной и той же длины* и *одного и того же формата*. Каждая запись характеризует один объект, а весь массив — сколь-угодно большое число таких объектов. Объекты характеризуются определенным (однотипным для всех объектов) *набором признаков*, т. е. совокупностью характеризующих объект фактов, представленных в *формализованном* виде.

Признаки могут быть количественные и качественные. В первом случае признак характеризуется числом (например, год рождения в кадровой анкете). Во втором случае значением признака может служить произвольная (но строго определенная) конечная совокупность буквенно-цифровых выражений (например, название месяца и дня рождения, специальность и номер разряда и т. п.). Формализация как раз и означает, что для характеристики того или иного признака надо пользоваться лишь строго определенными терминами, а не произвольными словесными описаниями.

Таким образом, фактографические системы нуждаются в специальных *классификаторах* признаков (в первую очередь качественных), содержащих строго фиксированные (обычно закодированные короткими кодами) значения соответствующих признаков. Для подробных кадровых анкет нужны классификаторы высших и средних специальных учебных заведений, классификаторы должностей, предприятий и организаций (для графы «работа в прошлом») и т. п. Для удобства автоматизированного поиска в классификаторы может вводиться внутренняя структура, когда тот или иной признак разбивается на несколько более мелких.

* В целях простоты мы исключаем из рассмотрения комбинированные системы, например фактодокументальные, где информация хранится в виде неформализованных документов, а задачей системы является извлечение из них различного рода фактов.

Например, признак «место работы» в кадровой анкете может быть составлен из группового признака (министерство, завод, колхоз и др.) и название объекта внутри группы (имя или номер завода, название министерства и т. д.).

Трудность создания фактографических систем определяется сложностью классификаторов. Для обычных кадровых систем разработка классификатора относительно нетрудна. Если же в анкету вводятся характеристики личных качеств человека (что необходимо для правильного подбора кадров), то задача создания классификаторов усложняется: для их разработки требуется помощь психологов и социологов. Очень трудные задачи приходится решать при разработке системы признаков и соответствующих классификаторов для характеристики конъюнктуры на мировом рынке и для других задач.

Следующим важным элементом фактографической системы является специальное математическое обеспечение (СМО), которое решает две задачи. Первая — это обеспечение ввода первичных документов, как для первоначального создания информационных массивов, так и для их последующего дополнения и обновления. Первичные документы, предназначенные для ввода в ЭВМ, в простейших фактографических системах должны быть формализованными. В простейшем случае первичный документ есть точная копия записи в массиве, которую надо ввести или обновить. Она состоит из имени (или номера) объекта и полной совокупности его признаков.

При обновлении длинных записей такая система неудобна, если обновлению подлежит лишь небольшая часть признаков. Тогда строится простой входной язык, использующий макрооператор «заменить» с указанием записи и параметров, подлежащих замене. Математическое обеспечение производит фактическое выполнение этого макрооператора.

Документооборот в фактографических системах может быть организован двумя способами. 1. Первичные документы поступают в ВЦ уже в формализованном виде (формализованные сообщения) и даже на машинных носителях. Формализация сообщений производится теми, кто их посылает. 2. Допускаются обычные неформализованные сообщения, которые анализируются специальной группой в ВЦ, составляющей на их основе формализованные документы первичного ввода. Второй способ требует больше ручного труда, чем первый, и поэтому применяется в основном при не очень интенсивных потоках входных документов.

Вторая часть специального математического обеспечения фактографических систем организует обслуживание запросов и выдачу необходимых справок. Для этой цели прежде всего необходим специальный формализованный язык — *язык запросов*. Простейший вид запроса, который мы будем называть *прямым*, состоит

в требовании выдать на тот или иной терминал (АЦПУ, телетайп или экранный пульт) определенную запись из определенного массива. Запись и массив называются своими именами (которыми могут служить и просто порядковые номера). Иными словами, в языке запросов должны быть макрооператоры вида «выдать на терминал <имя> записи <имя>, <имя>, ..., <имя> из массива <имя>». При наличии в системе лишь одного массива или при выдаче справки на тот же терминал, с которого поступил запрос, соответствующие имена могут быть исключены из запроса. Примером такого прямого запроса может быть следующий текст: «выдать на печать анкеты Иванов А. Д., Петров С. П.*». Подразумевается печать на любое свободное (или единственное имеющееся) АЦПУ.

Более сложную форму имеют *косвенные запросы* по признакам. В развитых языках запросов могут употребляться отношения $\geqslant$, $=$, $\leqslant$, $>$, $<$, $\neq$ и произвольные логические функции (булевы выражения) от отношений между признаками. Например, может быть построен запрос: «выдать на печать анкеты возраст >30 и возраст <50 и не холост и (токарь и разряд 5 или слесарь и разряд 4)». Поскольку таких анкет может оказаться много, с тем, чтобы не перегружать вывод, в язык могут вводиться макрооператоры счета, например, «сколько возраст 30 и токарь и разряд 5». После вывода результата запрашивающий решает, выводить ли все анкеты, или уточнить запрос и тем самым уменьшить число выводимых анкет. Полезными оказываются также различные макрооператоры *редактирования вывода*, позволяющие, например, выводить не все анкеты, а лишь одни фамилии или фамилии с небольшим числом дополнительных признаков.

Задача матобеспечения системы — анализировать поступающие запросы и организовывать их выполнение — вплоть до вывода ответа в требуемой форме.

Технический комплекс фактографических систем (как, впрочем, и любых АСИС) включает ЭВМ (или комплекс ЭВМ) с сильно развитой внешней памятью (желательно на магнитных дисках). Важным моментом для увеличения эффективности системы является наличие у ЭВМ возможностей мультипрограммирования и разделения времени. В АСИС большого объема, называемых *банками данных* обычно предполагается возможность запросов с многих десятков и даже сотен терминалов, в том числе и установленных в других местах, удаленных от ЭВМ.

---

* Формализованные запросы не всегда построены с соблюдением грамматических правил, поскольку это может затруднить их правильное понимание ЭВМ. В нашем примере согласование фамилии по падежу могло бы привести к тому, что машина приняла бы их за женские фамилии.

Любая АСИС, в том числе и фактографическая, рассчитывается, как система массового обслуживания. Основные характеристики системы — *среднее время ожидания ответа* и *пропускная способность* (производительность). Помимо технических характеристик ЭВМ (включая внешнюю память и вводно-выводные устройства), на величину указанных двух параметров влияет также *дисциплина обслуживания*. Чтобы понять механизм этого влияния и проиллюстрировать некоторые моменты ориентировочного расчета АСИС, рассмотрим простейший пример.

Пусть в нашем распоряжении имеется лишь небольшая ЭВМ с быстродействием 30 тыс. операций над 4-байтовыми словами в секунду, снабженная шестью лентопротяжками и не имеющая дисков.

Предположим, что ЭВМ не обладает возможностями мультипрограммной работы. Характеристики лент: емкость 10 млн. *байт* на одной ленте, скорость считывания и записи на ленту 30 тыс. *байт/сек*, полное время перемотки 6 *мин*, возможно считывание при обратном движении ленты; работа с лентой не занимает время центрального процессора. Скорость работы единственного АЦПУ 1000 символов в секунду.

Предположим, что на этой технической базе необходимо построить фактографическую систему, которая имеет один массив, состоящий из 600 тыс. записей (например, сокращенных анкет) размером в 100 *байт* каждая. Три первых байта заняты номерами записей (от 1 до 600 000), которые можно (в случае кадровой системы) отождествить с учетными номерами соответствующих лиц. Предположим, что задача системы — лишь печатать анкеты, запрашиваемые по учетным номерам.

Если расположить анкеты по лентам в порядке роста номеров анкет (на первой ленте от 1 до 100 000, на второй — от 100 001 до 200 000 и т. д.), то при поступлении запроса программа поиска сначала быстро определяет номер ленты и включает ее перемотку в нужном направлении. Записи считываются с ленты со скоростью $30\,000 : 100 = 300$ записей в секунду. На сравнение номера каждой записи с требуемым номером уходит 1—2 машинные команды (сравнения и, возможно, переадресации). Иными словами, при скорости 30 тыс. операций в секунду, процессор будет загружен на 1—2% своей мощности. Среднее время поиска нужной анкеты на ленте $6 : 2 = 3$ *мин*. Время печати найденной анкеты (в одну строку) $100 : 1000 = 0{,}1$ *сек*. Поскольку время работы центрального процессора по анализу первоначального запроса заведомо меньше этой величины, то все определяется временем поиска на ленте: среднее время ожидания ответа 3 *мин*, а максимальная пропускная способность $60 : 3 = 20$ запросов в час.

Как видим, система получилась весьма неэффективной, поскольку время ожидания имеет такой же порядок, который требуется для нахождения нужной анкеты (по порядковому номеру) в хорошо организованном архиве ручным способом. Очень малой является и пропускная способность системы. Последний недостаток можно исправить, изменив дисциплину обслуживания. Предположим для простоты, что запросы поступают регулярно через равные промежутки времени.

В результате предварительной обработки процессором запросы выстраиваются в очереди к соответствующим лентам. Как только в какой-нибудь из этих очередей набирается достаточное количество

запросов, начинается просмотр соответствующей ленты с целью удовлетворить все запросы очереди за один проход ленты. Если в группе N запросов, то на сравнение всех запросов с номером очередной просматриваемой анкеты уйдет уже не 1—2 команды, как раньше, а N—$2N$ команд. При $2N$ командах для полной загрузки процессора в группе должно быть 50 запросов. Это обеспечивает просмотр центральным процессором $\frac{30000}{2N} = \frac{30000}{100} = 300$ анкет в секунду, что соответствует скорости их считывания с магнитной ленты.

Вся группа запросов обрабатывается за время полной прокрутки ленты, т. е. 6 минут плюс $0{,}1 \cdot 50 = 5$ *сек* — на печатание ответов. Таким образом, пропускная способность системы при выбранной дисциплине обслуживания вырастет до величины 50 запросов за 6 *мин* 5 *сек*, что составляет более 490 запросов в час (рост почти в 25 раз). Вместе с тем среднее время ожидания вырастет на величину, равную среднему времени накопления очереди из 50 запросов, что составляет при полной загрузке системы не менее 6 *мин**. Разумеется, при меньших потоках запросов можно ограничиваться меньшими размерами групп, однако, среднее время ожидания все равно будет большим, чем при немедленной обработке редких одиночных запросов.

При групповой дисциплине обслуживания пропускная способность системы существенно превышает способности человека и поэтому может оказаться выгодной в тех случаях, когда решающим фактором является производительность, а задержка ответа на 5—10 *мин* не представляет особых неудобств.

Большое преимущество система получает в том случае, когда поиск анкет ведется не по номерам, а по любым другим признакам. Примером такого непрямого запроса может служить требование подобрать анкеты всех лиц, родившихся в 1945 г. на Дальнем Востоке, имеющих высшее техническое образование и опыт конструкторской работы, например, в самолетостроении — не менее 4 лет.

Чтобы удовлетворить такой запрос, человеку нужно просмотреть все 600 тыс. анкет. Если на просмотр каждой анкеты тратить всего 30 *сек*, то при 8-часовом рабочем дне на удовлетворение подобного запроса человеку потребовалось бы около двух лет. В нашей же системе для этого потребуется время, равное времени полной прокрутки всех шести лент, т. е. $6 \times 6 = 36$ *мин***. Таким обра-

---

* Точный расчет величины этого времени производится методами теории массового обслуживания.

** Анализ каждой анкеты на совпадение с приведенным запросом потребует менее 100 команд. Поэтому в данном случае ограничивающим фактором будет только скорость поступления информации с ленты.

зом, даже такая очень примитивная по современным понятиям техническая база системы повышает производительность труда по сравнению с ручным трудом не менее чем в 20 тыс. раз. При мультипрограммной работе и соответствующей производительности процессора можно организовать *одновременный просмотр* всех лент и тем самым еще в шесть раз поднять производительность.

В случае прямых запросов резкое улучшение характеристик системы происходит при замене лент магнитными дисками. Поскольку время поиска нужной анкеты (по ее номеру, легко преобразуемому в адрес) имеет порядок 0,1 *сек*, то даже в наиболее неблагоприятном случае, без мультипрограммирования и обслуживании запросов поодиночке, время ожидания (до окончания печати) составит $0{,}1 + 0{,}1 = 0{,}2$ *сек*, а пропускная способность системы $3600 : 0{,}2 = 18000$ запросов в час (время считывания с диска и преобразования номера анкеты в адрес много меньше, чем 0,1 *сек* и им можно пренебречь).

Таким образом, при удовлетворении прямых запросов диски имеют решающее преимущество перед лентами. При косвенных запросах (поиске по признакам) это преимущество не столь велико. Однако оно может тоже оказаться значительным, особенно когда анкеты нумеруются таким образом, чтобы анкеты с одинаковыми значениями тех признаков, по которым *часто* производится поиск, нумеровались *близкими* друг к другу (лучше всего последовательными) номерами.

Для убыстрения поиска информации по признакам в фактографических системах можно иметь также несколько экземпляров одного и того же массива только различным образом упорядоченных. Например, один экземпляр массива анкет упорядочивается по фамилиям и инициалам (в алфавитном порядке), а второй — по годам рождения и т. д.

Поиск в таких упорядоченных массивах обычно может быть организован более экономным образом, чем последовательный просмотр всех записей массива.

Если массив размещается на одной магнитной ленте, то так или иначе приходится просматривать все записи. Для массива на нескольких лентах можно запомнить в ОЗУ значения признака, которым кончаются части массива, записанные на разных лентах. В результате при заданном значении признака можно сразу определить, на какой ленте находится требуемая запись и просматривать только одну эту ленту, а не весь массив. Аналогичный прием можно применить для отдельных дорожек ЗУ на магнитных дисках или барабанах*.

---

* Описанный прием аналогичен тому, как мы ищем том энциклопедии, содержащий требуемое слово.

Для упорядоченного в ОЗУ массива часто употребляется *дихотомический* прием поиска по упорядочивающему признаку. С этой целью массив делится пополам, т. е. происходит обращение к *средней* записи массива (одной из двух при четном числе записей в массиве или единственной — при нечетном). Узнав значение признака для этой записи, определяют, в какой из двух половин массива лежит требуемая запись (при точном попадании в требуемую запись остается проверить лишь ее непосредственных соседей, которые могут иметь то же самое значение рассматриваемого признака). К найденной половине массива применяется тот же самый прием деления пополам, и так далее — до «точного попадания», т. е. нахождения записи с требуемым значением признака.

Если массив содержит N записей, то при дихотомическом поиске до попадания на нужную запись должно быть проанализировано не более $\log_2 N$ записей, что при больших значениях N значительно меньше, чем N.

Сложные фактографические системы могут иметь много разных массивов, переменную длину записей, сложную структуру самих массивов (иерархию подмассивов) и др. Основная задача проектирования таких систем состоит в том, чтобы максимально упростить и ускорить выборку необходимой информации как по прямым, так и по любым косвенным запросам. Это достигается тремя основными путями: выбором подходящей технической базы, организацией информационной базы (состава и структуры массивов) и, наконец, составом и качеством специального математического обеспечения — прежде всего программ, организующих поиск информации и надлежащую дисциплину обслуживания.

В отличие от фактографических систем, информационная база *документальных систем* строится в виде массива *неформализованных документов*, т. е. сообщений, писем, справок, рефератов статей, патентов и других материалов, написанных на обычном человеческом языке (русском, английском и др.). Будем называть такие языки *естественными* в отличие от *искусственных* (формализованных) языков, специально построенных для работы в различных областях (например, проблемно-ориентированных алгоритмических языков).

Основная идея любой документальной системы заключается в сопровождении документов специальными *поисковыми образами* этих документов, по которым и ведется поиск. Операция составления поискового образа документа носит название *индексирования*. Запросы, которые в документальных системах делаются на естественных языках, в результате специальной процедуры их *лингвистического анализа* также индексируются. Процедура поиска состоит в выборе тех документов, поисковые образы которых в некотором смысле близки к поисковому образу запроса.

Опишем один из простейших и наиболее употребительных путей реализаций этой идеи. Прежде всего производится выбор области или нескольких областей (науки, техники, экономики, политики и т. п.), к которым будут принадлежать накапливаемые в системе документы. Для этих областей составляется первичный словарь *базовых понятий*, называемых иногда *ключевыми словами*, которые наибольшим образом характеризуют данные области. Например, для документов о внешней торговле такими базовыми понятиями могут служить: фирма, договор, продажа, платежи, цена и т. п. В число ключевых слов могут попасть также названия конкретных фирм, стран, товаров и т. п.

Все выбранные слова нумеруются. Их полная совокупность образует *пространство понятий*, в котором и строятся соответствующие поисковые образы. Каждому базовому понятию соответствует координата этого пространства. Поисковые образы представляют векторы в пространстве понятий. Если тот или иной документ имеет дело с i-ым базовым понятием, то i-координата вектора этого документа отлична от нуля, в противном случае она полагается равной нулю. Конкретное числовое значение ненулевой координаты отождествляется с весом, который соответствующее базовое понятие имеет в рассматриваемом документе.

В простейшем случае вес понятия в документе выбирается равным числу различных упоминаний этого понятия в тексте документа. Так обычно поступают при автоматическом индексировании, когда поисковый образ документа составляет ЭВМ. Но даже и в этом случае могут быть использованы различные процедуры изменения веса, например увеличения веса тех базовых понятий, которые в оригинальном тексте были подчеркнуты, набраны жирным шрифтом, курсивом и т. п. При ручном индексировании вес определяется по содержательному смыслу документа и зависит в основном от компетентности и опыта человека, производящего индексирование.

Совершенно аналогичным образом, в виде вектора в пространстве понятий строится поисковый образ запроса. Степень близости двух поисковых образов A и B определяется обычно как косинус угла между соответствующими векторами

$$r(A, B) = \frac{A \cdot B}{|A| \cdot |B|},$$

где $A \cdot B$ — скалярное произведение; $|A|$ и $|B|$ — длины векторов A и B.

Величина $r(A, B)$ называется иногда также *коэффициентом* корреляции поисковых образов A и B. Близкими считаются поисковые образы, у которых коэффициент корреляции близок к единице. Поскольку все координаты векторов поисковых образов

неотрицательны, то минимально возможное значение коэффициента корреляции есть нуль. Ему соответствует случай полного различия понятий, характеризующих документ и запрос.

Простейший алгоритм поиска заключается в вычислении коэффициента корреляции поискового образа запроса с поисковыми образами всех документов, хранящихся в системе, и выдача тех документов, у которых этот коэффициент выше некоторого установленного порога (например, 0,5 или 0,75). Векторы поисковых образов задаются номерами и величиной ненулевых координат, нулевые координаты опускаются. Совокупность некоторых двух векторов A и B может быть представлена в виде: A : 16(1), 27(3), 195(4), 327(1), 592(3), B : 16(2), 82(3), 195(2), 327(2), 984(2). Коэффициент корреляции этих двух образов

$$r(A,\ B) = \frac{1\cdot 2+3\cdot 0+0\cdot 3+4\cdot 2+1\cdot 2+3\cdot 0+0\cdot 2}{\sqrt{1^2+3^2+4^2+1^2+3^2}\cdot\sqrt{2^2+3^2+2^2+2^2+2^2}} = \\ = \frac{2+6+3}{\sqrt{36}\cdot\sqrt{25}} = \frac{11}{30} \approx 0{,}37.$$

При реальном построении документальных систем описанные простейшие приемы являются обычно недостаточными. Прежде всего ключевые слова могут в различных документах изменяться по числам, падежам, родам. Для их опознания необходимо поэтому проводить простейший грамматический анализ для отделения корня от окончаний, суффиксов, а иногда и приставок. Поэтому в словарь для опознания ключевых слов вносятся лишь их корни. Другие же возможные элементы слов (в частности, окончания) задаются отдельно специальными таблицами.

Далее, в естественных языках возможны различные синтаксические формы для выражения одной и той же мысли, например: «фирма A продала фирме B ...» и «состоялась продажа фирмой A фирме B ...». Алгоритм анализа текста (при автоматической индексации) должен разобраться в том, что этими формами выражается одна и та же мысль. Аналогичная трудность возникает при употреблении различных синонимов, т. е. разных слов с одним и тем же значением. Например, одно и то же понятие в различных документах может получить название «договор», «контракт», «соглашение» и даже «сделка».

Для преодоления указанных трудностей в развитых документальных системах предпочитают различать ключевые слова, которые надо обнаруживать в тексте по их корням с помощью специальных алгоритмов грамматического анализа, и базовые понятия, которые могут выражаться различными ключевыми словами, и даже иметь более общий смысл (родовые понятия). Для установления связи между ключевыми словами и базовыми понятиями строятся

словари особого вида, называемые *тезаурусами*. Задача тезауруса — установить, каким базовым понятиям соответствует то или иное слово. При этом базовое понятие может быть более широким, чем понятие, выражаемое самим ключевым словом. Например, слово «база» может соответствовать базовым понятиям «полупроводниковый прибор», «торговля», «геодезия», «военные приготовления» и др. Слово «поле» характеризует базовые понятия «сельское хозяйство», «электромагнетизм», «гравитация» и т. п.

При наличии тезауруса обнаружение в тексте одного такого «многозначного» слова в простейшем случае приводит к приписыванию ненулевых значений координат, соответствующих всем связанным с ним базовым понятиям. Для устранения этого недостатка применяются более сложные алгоритмы *семантического анализа*, которые определяют, в каком именно смысле употреблялось данное слово в рассматриваемом документе. Такие алгоритмы пока еще далеки от совершенства, но исследования в этом направлении продолжаются.

В зависимости от совершенства алгоритмов анализа текста и его индексации меняются два важных показателя, характеризующих эффективность работы документальных систем, а именно,— полнота и избыточность удовлетворения запросов. Предположим, что на некоторый запрос A рассматриваемая система выдала n документов. Среди них только какое-то количество m ($m \leqslant n$) документов оказываются *релевантными* (т. е. имеющими отношение) к запросу A. Остальные $n - m$ документов являются лишними. Они, как иногда говорят, представляют собой «шум» данной системы. В то же время какое-то количество $k \geqslant 0$ релевантных к запросу A документов могут не быть выбраны системой.

Если обозначить общее число релевантных к A документов (равное $m + k$) N, то отношение $\frac{m}{N} = \frac{m}{m+k}$ будет характеризовать *полноту* рассматриваемого ответа, а отношение

$$\frac{n-m}{N} = \frac{n-m}{m+k}$$

его *избыточность*. Средние величины этих отношений при большом числе случайных запросов принимаются за характеристики полноты и избыточности системы.

При описанном выше способе построения поискового образа, когда документ характеризуется простой суммой базовых понятий, имеющих отношение к его тексту, шум является по существу неизбежным. Дело в том, что смысл документа характеризуется не только простым набором упомянутых в нем понятий, но и отношениями между этими понятиями. Действительно, если нам необходимо найти все документы, касающиеся продаж, сделанных фирмой

A фирме *B*, то набор базовых понятий «продажа», «фирма *A*», «фирма *B*» в поисковом образе запроса приведет к выборке документов не только о продажах фирмы *A* фирме *B*, но и о продажах фирмы *B* фирме *A*.

В настоящее время делаются попытки построения документальных систем с более сложными поисковыми образами, в которых учитываются различные отношения, существующие между базовыми понятиями. По мере прогресса в таких системах их эффективность возрастает.

Проблема автоматического индексирования тесно связана с проблемой *автоматического реферирования*. Один из простейших (хотя и не очень эффективных) методов автоматизации составления рефератов состоит в следующем. При первом просмотре текста ЭВМ выделяет встречающиеся в нем ключевые слова и определяет их веса. При втором просмотре выделяются все те предложения, которые включают наиболее весомые ключевые слова. Они и составляют требуемый реферат. Имеются различные возможности для улучшения описанного алгоритма, позволяющие получать рефераты более высокого качества. Однако исчерпывающего решения проблема автоматического реферирования еще не получила.

Кроме полноты и избыточности удовлетворения запросов, важнейшей характеристикой, определяющей эффективность работы документальных систем (как, впрочем, и любых других), является скорость ее работы. Поскольку вычисление скалярных произведений требует большего числа машинных операций, чем простые сравнения, документальные системы нуждаются при прочих равных условиях в более быстродействующих процессорах, чем системы фактографические. Однако не эта проблема является главной.

Трудность в другом. Дело в том, что при большом числе документов их полный просмотр при каждом запросе неизбежно занимает много времени. Возможности организации группового поиска в документальных системах относительно невелики, поскольку, по сравнению с фактографическими системами резерв производительности процессора у них значительно меньше. Для уменьшения времени на ввод из внешней памяти в документальных системах целесообразно организовать раздельное хранение самих документов и их поисковых образов. При этом поисковые образы снабжаются адресами документов, а процесс поиска идет лишь в массиве поисковых образов*.

Поисковые образы многих документов могут оказаться достаточно громоздкими и общее время полного их просмотра получа-

---

* Сами документы при этом могут находиться и вне ЭВМ, например, в виде микрофильмового архива. Знание адресов позволяет быстро подобрать требуемые документы.

ется недопустимо большим. Для увеличения скорости работы системы применяют специальный прием— группирование документов.

Основная идея этого приема состоит в объединении документов в группы однотипных документов и нахождении наиболее характерного представителя каждой группы, называемого обычно *центроидом* группы. Возможно образование групп (и соответствующих центроидов) разных уровней. Группы верхнего уровня (самые большие) подразбиваются на более мелкие, те еще на более мелкие и т. д.

При поиске поисковый образ запроса сравнивается сначала с поисковыми образами центроидов групп верхнего уровня, в результате чего находятся группы, подлежащие дальнейшему исследованию. Остальные группы, центроиды которых имеют малую корреляцию с запросом, отбрасываются.

В выделенных группах снова производится сравнение с центроидами содержащихся в них групп следующего уровня, и так далее, пока не дойдем до групп, состоящих из одного элемента, т. е. отдельных документов.

В настоящее время разработано несколько алгоритмов для автоматического группирования документов. Идея одного из наиболее естественных и простых алгоритмов заключается в следующем. Задаются ориентировочным размером групп данного уровня и последовательным коррелированием каждого документа со всеми остальными находят те документы, вокруг которых объединяются группы (приблизительно) требуемого размера с наибольшими минимальными значениями коэффициента корреляции всех документов группы с объединяющим их документом. Эти документы рассматриваются в дальнейшем как центроиды соответствующих групп.

Возможно и ручное (предварительное) группирование документов (в соответствии с их содержательным смыслом) с последующим автоматическим нахождением центроидов, как документов, имеющих наибольшую минимальную корреляцию со всеми документами внутри соответствующих групп.

Следует иметь в виду, что при группировании (как автоматическом, так и ручном) не исключается возможность попадания одного и того же документа в несколько различных групп. Даже, если это приведет к увеличению массива поисковых образов, преимущество такого «перекрестного» группирования, заключающееся в увеличении полноты системы, полностью компенсирует возникающие неудобства. Впрочем, применение специальных методов организации массивов — так называемых *списочных структур*, позволяет иметь каждый поисковый образ лишь в одном экземпляре. В центроиде при этом указываются все адреса поисковых образов соответствующих ему документов или центроидов следующего уровня.

Как показывает опыт, все описанные выше методы улучшения качества работы документальных АСИС еще не дают гарантии получения хороших показателей полноты и избыточности ответов. Поэтому в последнее время все большую популярность получают различные методы, использующие обратную связь для повышения качества функционирования документальных систем.

Имеется два основных пути использования обратной связи. При первом система из обычной справочно-информационной превращается в *диалоговую*. Получив ответ на запрос, пользователь может внести изменения в запрос (без полного его повторения), например, добавить или исключить какие-либо ключевые слова, изменить вес отдельных ключевых слов и т. д. В результате такого диалога быстро удается получить хорошие показатели полноты и избыточности ответа на сделанный запрос.

Второй путь состоит в том, что получив после анализа ответа указания пользователя о лишних документах, система начинает менять их поисковые образы (обычно за счет изменения ненулевых координат векторов) таким образом, чтобы уменьшить значения их коэффициентов корреляции к рассматриваемому запросу. В этом случае не запросы приспосабливаются к системе, а система к запросам. Возможна и комбинация обоих рассмотренных методов.

В последние годы стали развиваться методы автоматизации процессов *логического вывода*. Тем самым создается возможность автоматического извлечения из документов не только прямо содержащихся в них фактов, но и *логических следствий* из них. Важность этого обстоятельства обусловливается тем фактом, что уже сейчас возникает необходимость строить смешанные *документально-фактографические* системы, в которых пополнение и обновление фактографической части происходило бы автоматически в результате анализа информации, содержащейся в документальной части системы. Такую задачу сегодня удается решить только для самых простых случаев.

Гораздо более просто строятся смешанные системы несколько иного вида, которые мы будем называть *фактографическо-документальными*. От обычной фактографической системы, обновляемой в результате ручного анализа поступающих неформализованных документов, эта система отличается лишь в одном. А именно, каждый показатель во всех записях фактографических массивов сопровождается ссылкой на адреса документов, на основании анализа которых ему присвоено текущее (а возможно, и ряд прошлых) значение. Сам массив документов может при этом храниться как во внешней памяти ЭВМ, так и вне ее*. Возможен и промежуточный

* Одной из удобных форм хранения документов являются так называемые микрофиши — микрофильмы сверхминиатюрных размеров, снабженные устройствами для выборки, вывода на экран и снятия копий необходимых документов.

вариант, когда в ЭВМ хранятся краткие рефераты документов, а сами документы находятся вне машины.

Автоматизированные справочно-информационные системы могут оказать и фактически оказывают большую помощь в процессах управления и принятия решений в качестве важного вспомогательного инструмента. Существует путь превращения АСИС в еще более эффективный прямой инструмент управления. Идея такого превращения состоит в объединении АСИС с динамическими моделями (обобщенными динамическими системами), рассматриваемыми в системном анализе. С этой целью строится специальное математическое обеспечение, решающее три основные задачи. Первая задача — это обеспечение диалога человека с ЭВМ для построения моделей, подобных тем, которые описаны в параграфе 7 гл. II. Вторая задача — обеспечение работы с этими моделями (задание управления, прогноз, отображение результатов, корректировка управления, новый прогноз и т. д.). Для обеспечения достаточной наглядности отображения и скорости работы систему при решении таких задач желательно снабжать экранными пультами, особенно если они могут работать с ЭВМ в режиме разделения времени, что дает возможность коллективного пользования системой.

Третья задача — это автоматическое (опирающееся на информационную базу АСИС) внесение в построенные динамические модели информации о нынешнем состоянии моделируемого объекта или процесса. При этом специалисты, строящие модели, могут не иметь доступа к конкретным документам, на основании которых производится заполнение модели информацией.

4. Основные типы задач управления экономическими объектами

Рассмотрим основные понятия, необходимые для описания комплексов задач управления. *Экономическим объектом* будем называть любой объект, который может получить задание на производство или распределение тех или иных продуктов или выступить в качестве их потребителя. Под продуктами понимаются не только готовые товары, но и полуфабрикаты, предназначенные для последующей обработки, а также проекты, решение задач и различного рода услуги. Таким образом, к числу экономических объектов может быть отнесен не только завод, но и проектный институт, сбытовая контора, зрелищное предприятие и т. п.

Имеется естественная иерархия экономических объектов. На верхнем уровне этой иерархии находится все народное хозяйство. Относящиеся к этому уровню задачи управления принято называть *макроэкономическими*. Следующий уровень составляют отдельные

отрасли народного хозяйства. На третьем уровне находятся отдельные предприятия (или объединения) и организации, непосредственно производящие товары или услуги. Их в свою очередь можно представить в виде объединения еще более мелких объектов: цехов, участков и отдельных единиц оборудования и рабочих мест, не входящих в жесткие технологические линии. Эти последние объекты, не разделяющиеся на еще более мелкие экономические объекты, будем называть элементарными.

Примерами элементарных экономических объектов могут служить домна, металлорежущий станок, рабочее место продавца в магазине и т. п. Элементарным объектом можно считать также химический агрегат, хотя и включенный в линию, но работающий на промежуточную буферную емкость, что позволяет планировать его продукцию до известной степени независимо от всей линии.

Всякий экономический объект, кроме элементарного, характеризуется прежде всего своей структурой, т. е. теми экономическими объектами более низкого уровня, из которых он состоит. Мы будем называть их *подобъектами*. Далее, всякий действующий экономический объект для выполнения своих функций должен обладать определенными ресурсами. Ресурсы разделяются на трудовые и материальные.

Трудовые ресурсы составляют рабочие, инженерно-технические работники, работники управленческого и вспомогательного персонала различных специальностей, которыми располагает рассматриваемый экономический объект. Материальные ресурсы состоят из основных и оборотных фондов. К числу основных фондов относятся средства труда длительного пользования (земля, производственные помещения, оборудование). Оборотные фонды составляются из предметов труда, потребляющихся в процессе производства. Это — запасы материалов и покупных изделий, продукты, находящиеся в различных стадиях готовности (незавершенное производство); а также денежные средства, необходимые для нормального функционирования объекта.

Полный перечень ресурсов должен содержаться в основных информационных массивах, которые будем называть *информационной моделью* рассматриваемого экономического объекта. Для уровня предприятий, как уже отмечалось выше, информационная модель должна содержать перечень всех ресурсов в самой детальной номенклатуре и обновляться в том же темпе, в каком происходит их изменение (за счет ведения автоматизированного учета по всем складам и цехам).

На уровне отрасли производится агрегирование информации. Информационная модель отрасли не должна содержать сведения о каждом рабочем, а лишь количества рабочих одной и той же

специальности и квалификации по каждому предприятию. Аналогично сводится в группы и характеризуется общим количеством оборудование одного и того же типа и (примерно) одной и той же степенью изношенности, измеряемой качественно по 3- или 5-балльной системе. Из оборотных фондов учитываются запасы материалов и комплектующих изделий, поступающие на предприятие извне, а также готовая продукция, почему-либо не отправленная потребителям. Что же касается незавершенного производства, то его учитывают суммарной стоимостью. Исключение делается лишь для мелкосерийного и индивидуального производства с длительным циклом (например, строительство судов). В этом случае в информационной модели запоминается текущее состояние укрупненного сетевого графика с циклом обновления порядка недели.

Вторым исходным моментом для решения задач управления является технологическая информация о заданиях, которые может выполнять рассматриваемый экономический объект и составляющие его подобъекты. Для каждого задания эта информация представляется в виде сетевого графика, показывающего последовательность операций (производственных, транспортных, снабженческих, строительных и др.), которые необходимы для выполнения данного задания. В машиностроении и приборостроении такой график принято называть технологическим маршрутом. Каждая операция характеризуется специальной технологической картой, оценивающей время выполнения операции и затраты ресурсов, необходимые для ее выполнения.

Для решения задач оптимизации управления технологическая информация должна быть *многовариантной*. Многовариантность может иметь место как в технологическом маршруте, так и в технологических картах. В последнем случае различают разные способы и разные режимы выполнения операции. Разные способы отличаются использованием различных ресурсов (одну и ту же деталь можно изготовить точным литьем, штамповкой или обработкой на станке). Различие режимов предполагает различное количество или различную интенсивность использования одних и тех же ресурсов (перевозка грузов одним автомобилем или десятью, увеличение скорости резания на станке и т. п.).

Степень подробности технологической информации на уровне предприятий и отрасли различна: на предприятиях самая детальная, а в отрасли — укрупненная. Укрупнение производится как по линии агрегирования ресурсов (о чем уже было сказано выше), так и по линии уменьшения многовариантности, например рассмотрение только одного или двух режимов (среднего и наиболее интенсивного) вместо нескольких.

Технологическая информация и информация о ресурсах должна готовиться для вышестоящих уровней автоматически на основании

исходной (самой подробной). Соответствующие программы должны быть заложены в АСОУ предприятий и министерств.

О технологической информации должно быть сделано несколько важных замечаний. Во-первых, для эффективного решения задач перспективного планирования и управления научно-техническим прогрессом эта информация (пусть в очень ориентировочной укрупненной форме) должна начинать поступать в систему на самых ранних стадиях проектирования (начиная с технического задания на проектируемый объект) и непрерывно расширяться и уточняться.

Во-вторых, желательно, чтобы нормативы расходования трудовых и материальных ресурсов в технологических картах были двойными: наряду с обычными (завышенными), по которым производится планирование; в карту должны помещаться предельно допустимые (при данной технологии) нормативы, т. е. такие, при которых исключаются все потери (рабочего времени, материалов и т. п.), не вытекающие из существа технологии.

Третье замечание касается того, что такие предельно допустимые и вместе с тем все еще выполнимые нормативы не могут быть выработаны, когда разрабатывающие их конструкторы и технологи стимулируются (вместе с предприятием) за выполнение и перевыполнение планов. Они должны стимулироваться за максимальное ужесточение этих нормативов, а чтобы нормативы тем не менее остались реальными, процедура их утверждения должна базироваться на борьбе проектантов, которым выгодно максимально занизить нормативы, и работников предприятия, которым выгодно их завысить.

Наконец, еще одно замечание касается способа установления правильных нормативов. Для индивидуального и мелкосерийного производства разработка подробных подетальных норм на основании точных технологических расчетов не всегда возможна и оправдана. В этом случае, а также на ранних стадиях проектирования изделий для крупной серии можно пользоваться качественными оценками, основанными на аналогиях с ранее изготовлявшимися изделиями, узлами или деталями. Важно лишь, чтобы при таких оценках в равной мере проявлялись как тенденции к завышению, так и тенденции к занижению норм. Превалирование одной из этих тенденций выявляется по результатам работы объекта, и это должно являться сигналом для коррекции системы стимулирования.

Третья составная часть информации, необходимая для решения задач управления, это — поток предлагаемых заданий.

Существуют три основных источника формирования потока. 1. *Директивные задания*, определяемые решениями вышестоящих органов (например, задания от руководства объекта составляющим его подобъектам). 2. *Прямые связи* между различными экономичес-

кими объектами, не подчиненными друг другу административно. Такие задания принято называть заказами. 3. *Ожидаемый спрос*, т. е. будущие заказы на производимые объектом продукты. В первых двух случаях задания поступают извне рассматриваемого объекта и служат для АСУ объекта лишь входной информацией. В третьем случае перед АСУ возникает специальная задача — *прогнозирования спроса*.

Четвертый вид информации, являющийся отправной точкой для решения задач управления — это ограничения, задаваемые объектам с более высоких уровней управления. Одним типом ограничений являются директивные задания. Другой тип ограничений — верхние границы для использования тех или иных ресурсов, например ограничения по фонду зарплаты, по расходу электроэнергии и т. п. По мере широкого развития АСОУ, объединенных в единую систему, число ограничений будет снижаться.

В самом деле, подобные ограничения имеют своей целью упорядочить расходование ресурсов, дефицитных в общегосударственном масштабе. Выдача заранее обусловленных лимитов — есть один из методов решения этой задачи, далеко не лучший, но неизбежный при отсутствии достаточно совершенной обратной связи. Если система обратной связи построена так, что для любого заданного момента может быть быстро подсчитан средний темп расходования определенного ресурса (например, фонда зарплаты) в целом по стране, а также в любых разрезах отраслевого или территориального плана, то вместо заранее выдаваемых лимитов ограничения могут применяться по мере необходимости лишь к тем объектам, которые имеют наибольшие отклонения от нормы. Если же в целом по стране средняя норма выдерживается, то даже при наличии заметных отклонений на отдельных объектах нет нужды в применении ограничений. Третий тип ограничений — это директивные задания по различным экономическим показателям работы объекта (производительность труда, себестоимость продукции и т. п.).

По мере развития АСОУ и соответствующих экономических мероприятий разработка таких заданий все в большей и большей мере должна производиться на самом объекте, являясь не исходной информацией для планирования, а его результатом.

Полный вид информации, на котором зиждется управление,— это обратная связь о ходе выполнения заданий и о фактических расходах ресурсов, т. е. о текущем состоянии объекта.

Все перечисленные виды информации оказываются достаточными для управления работой объекта при неизменных основных ресурсах и неизменной технологической информации (меняются лишь задания, ограничения и текущее состояние объекта, включая вспомогательные ресурсы). Такая ситуация характерна для комп-

лекса задач *текущего (оперативного) управления* и *краткосрочного планирования*.

Развитие объекта приводит к количественному и качественному изменению основных ресурсов и появлению новой технологической информации (проекты новых изделий). В результате изменяется круг заданий, который может выполнять объект. *Задания на развитие объекта*, которые надо отличать от рассмотренных ранее основных заданий, возникают прежде всего в результате прогноза потоков основных заданий на сроки более длительные, чем при краткосрочном планировании. Основа такого прогноза и формирование заданий на развитие отраслей делается на макроэкономическом уровне. Их детализация и привязка к конкретным объектам производится на уровне отраслей и предприятий, обладающих своей собственной проектно-конструкторской базой.

Для выработки заданий на развитие объекта используется также технико-экономический анализ текущей работы объекта, позволяющий выявлять узкие места в существующей структуре ресурсов (как в количественном, так и в качественном плане), а также недостатки существующей технологической информации (несовершенство выпускаемых продуктов).

Для управления процессом выполнения заданий на развитие, так же как и в случае основных заданий, необходима информация о выделяемых для этой цели ресурсах, технологическая информация о необходимых последовательностях операций для проектно-конструкторских, строительных, монтажных и пусконаладочных работ. Наконец, для заданий также могут выдаваться различные ограничения (например, директивные требования об уровне очистки промышленных сбросов в реки и атмосферу).

Планы развития объекта составляют основу системы долгосрочного планирования, позволяющей, в частности (на основе увеличенных ресурсов), спланировать на более длительный период рост возможностей объекта по выполнению его основных заданий. В результате комплексы задач краткосрочного и долговременного планирования взаимно переплетаются.

Что нового вносит АСУ в организацию системы планирования?

1. *Непрерывность* — это означает, что планирование ведется не на какой-то фиксированный календарный период (год, квартал, месяц), а все время на определенный срок T, называемый *интервалом планирования*, вперед от настоящего момента. Если T равно одному году, то в феврале мы должны иметь план до февраля следующего года, в марте — до марта следующего года и т. д.

2. *Динамичность* плана — это означает, что хорошо выполненная система автоматизации планирования должна обеспечить возможность проведения любых корректировок плана без нарушения его целостности и взаимоувязанности. Иными словами,

при внесении в план какого-то изменения ЭВМ автоматически должна осуществить всю цепь вытекающих из него следствий (например, при изменении срока изготовления партии деталей изменить соответственно сроки заказа материалов, сроки изготовления необходимой оснастки и др.). Следует заметить, что при ручном планировании трудно добиться полной взаимоувязанности планов даже в их первоначальном (долготовящемся) варианте. После же внесения изменений эта взаимоувязанность зачастую полностью теряется. Осуществление полной взаимосвязанности возможно лишь на основе гораздо большей, чем при ручном планировании детализации планов.

3. *Многовариантность* планов и возможность их оптимизации по критериям, которые можно легко и быстро менять.

4. *Тесная взаимоувязанность* различных систем планирования (перспективного, долгосрочного, краткосрочного и текущего) между собой и с другими комплексами задач,— в первую очередь с задачами технико-экономического анализа. Хорошо спроектированная автоматизированная система планирования должна строиться как человеко-машинная система. В систему должны быть включены процедуры, позволяющие человеку вести разговор с машиной. Это означает, в частности, возможность получения ответов на различные вопросы экономического характера, не предусмотренные в отчетных документах. Например, вопрос о том, какие производственные операции дают наибольший вклад в себестоимость какого-либо определенного продукта или показатели по производительности труда на отдельных производственных участках. Используя такие процедуры экономического анализа, человек может определить желательные коррективы к плану. Иными словами, система должна обеспечить возможность непрерывной аналитической работы с планом (и с результатами его выполнения) для своевременного определения заданий для возможности целенаправленной корректировки плана.

АСУ осуществляет параллельное ведение плановых расчетов по двойной системе нормативов, о которой уже упоминалось выше. Такие расчеты позволяют получить ответ на вопрос об очередности решения задач совершенствования организации и технологии производства, о путях научно-технического прогресса.

Основой для технико-экономического анализа должна быть правильная система ценообразования. При этом речь идет не о способе установления отпускных цен на выпускаемую продукцию, а об объективном измерении стоимостей, фактически создаваемых на данном экономическом объекте. Не вдаваясь в дискуссию о преимуществах различных видов ценообразования, остановимся на одном из них. Как известно, стоимость любого продукта складывается из стоимости затраченных на его производство предметов

труда, определенной части стоимости использованных при его производстве средств труда — так называемая *амортизация*, стоимости затраченного *живого труда* (в виде соответствующей части зарплаты) и *прибыли*, которая исчисляется в определенном (одинаковом для всего народного хозяйства) проценте от затрат живого труда*.

Поскольку предметы труда и средства труда были произведены на каких-то экономических объектах раньше, чем они используются в рассматриваемом процессе, для определения их стоимости в прошлом мог бы быть применен тот же самый процесс. Если даже в какой-то начальный момент их стоимости были установлены случайно, то после многократного применения описанного ниже процесса это начальное «возмущение» можно будет уже не принимать во внимание.

Итак, стоимости предметов труда и средств труда будем считать заданными. Первый вопрос, который нужно решить дальше — как исчислять амортизацию? Для того чтобы в полной мере учитывать требования научно-технического прогресса, улучшения использования оборудования и повышения требований к его качеству, представляется целесообразным несколько изменить традиционный подход к исчислению амортизации. Прежде всего необходимо различать физическое и моральное старение (износ) оборудования.

Под физическим износом оборудования понимается такой, который делает невозможным его дальнейшее использование. Моральный износ оборудования наступает тогда, когда появляется новое, заменяющее его оборудование, при использовании которого стоимость производимых продуктов становится *меньше*, чем в случае продолжения использования старого оборудования. При выпуске оборудования в его паспорт вносится информация о сроке физического износа, т. е. указывается суммарное рабочее время (без учета простоев), на которое рассчитано данное оборудование, число рабочих ходов, оборотов, пройденных километров или иных рабочих операций, приводящих к физическому износу. Поскольку износ зависит от вида операций и режима, в котором используется оборудование, в общем случае информация о сроке физического износа может быть представлена в виде формулы или алгоритма, по которым может быть определен этот срок. Например, N — число рабочих ходов, приводящих к физическому износу оборудования, определяется по формуле $N = n_1 + 2n_2$, где n_1 — число операций в легком режиме; n_2 — в тяжелом. Иными словами, N представляет собой число приведенных рабочих операций, когда операция в тяжелом режиме засчитывается за две операции в легком ре-

* Величина этого процента определяется на макроэкономическом уровне.

жиме. В паспорт вносится также информация о сроках и ориентировочной стоимости текущих ремонтов и профилактики оборудования.

Для определения срока морального износа оборудования должна быть создана специальная служба непрерывного технологического прогнозирования (о ней будет более подробно сказано в параграфе, посвященном макроэкономическим задачам). Для каждого вида оборудования эта служба выдает и постоянно корректирует дату начала выпуска нового заменяющего его оборудования, при котором снижается стоимость производимых с его помощью продуктов с сохранением их качества или с таким улучшением качества, которое вполне компенсирует некоторое увеличение стоимости*. Может оказаться полезным одновременное прогнозирование двух таких дат — для мирового и национального уровня. Поскольку прогнозирование ведется *непрерывно*, в конце-концов (в момент свершения события или незадолго до его свершения) требуемые даты будут указаны точно.

Предположим, что какой-то комплект оборудования используется для выполнения элементарного задания, т. е. такого задания, которое выполняется элементарным экономическим объектом и состоит в производстве партии определенного продукта (изделий, узлов, деталей, материалов, полуфабрикатов и т. п.). В технологической информации для выполнения этого задания должно быть указано технологически необходимое (минимальное) время $t_{\text{мин}}$ или количество рабочих операций $n_{\text{мин}}$.

Время $t_{\text{мин}}$ складывается из технологически необходимого времени $t'_{\text{мин}}$ подготовки оборудования к выполнению задания и технологически необходимого (предельно допустимого) времени $t''_{\text{мин}}$ собственно на выполнение задания, которое будем называть технологически необходимым или предельно допустимым *рабочим временем* выполнения данного задания.

Фактически затрачиваемое на выполнение задания (включая подготовку) время $t_{\text{факт}}$ больше $t_{\text{мин}}$ за счет потерь. То же самое касается и фактически израсходованного числа рабочих операций $n_{\text{факт}}$.

Пусть стоимость оборудования в момент установки равна P. На момент τ_1 своей замены в силу физического или морального износа амортизация должна возместить величину P', большую, чем P на величину процента на ссуду P, полученную для приобретения оборудования (величину этого процента целесообразно устанавливать на макроэкономическом уровне в соответствии

---

* Для этой цели, как нетрудно понять, нужно прогнозировать срок морального старения не только нынешнего оборудования, но и того, которое его заменит (разумеется, еще более ориентировочно).

с темпами роста стоимости основных фондов в масштабе всего народного хозяйства). Момент τ' морального износа поступает из службы прогнозирования, момент τ'' физического износа прогнозируется непосредственно на рассматриваемом экономическом объекте, учитывая данные технического паспорта и ожидаемый коэффициент использования данного оборудования. Момент замены τ_1 есть минимальное из двух сроков τ' и τ''. На этот момент по формуле сложных процентов и подсчитывается величина P'.

Для определения доли стоимости, приходящейся на амортизацию, будем вычислять не одну, а две различные стоимости: *минимальную* (технологически допустимую) и *фактическую*. Доля амортизации в каждой из них состоит из двух частей: прямой и косвенной. Сумма p амортизации, начисленная на данную единицу оборудования, меняется по мере выполнения заказов и поступления сведений об изменении срока τ_1 предполагаемого износа (физического или морального).

Если эти суммы *перед началом* выполнения задания были равны $p_{\text{мин}}$ и $p_{\text{факт}}$, то после выполнения задания они увеличатся соответственно на величины

$$(P' - p_{\text{мин}})\max\left(\frac{t_{\text{мин}}}{t}, \frac{n_{\text{мин}}}{n}\right);$$

$$(P' - p_{\text{факт}})\max\left(\frac{t_{\text{факт}}}{t}, \frac{n_{\text{факт}}}{n}\right),$$

где t — отрезок времени от начала выполнения задания до момента τ_1 предполагаемого износа оборудования; n — разность числа операций, вызывающих физический износ оборудования, и выполненных на нем к началу выполнения заказа*.

Следует различать *технологически необходимые* и *чистые* простои оборудования. К технологически небходимых относятся простои, предусмотренные в паспорте оборудования для минимально необходимых сроков текущих ремонтов и профилактики, или связанные с сезонным характером работы (оборудование сахарных заводов, зерноуборочные комбайны и т. п.). Если обозначить $t_{\text{пр}}$ время простоя, начавшегося за время t до срока износа оборудования, то амортизационные начисления за время простоя определяются по формуле

$$(P' - p)\frac{t_{\text{пр}}}{t},$$

причем, для $p = p_{\text{мин}}$ начисления производятся только при техно-

* Число n может прогнозироваться, меняя свое значение в процессе наблюдения за работой оборудования.

логически необходимых простоях, а для $p = p_{\text{факт}}$ — для простоев всех видов.

Амортизация в случае простоев относится к разряду косвенной, т. е. такой, которая не связана непосредственно с выполнением какого-то определенного задания. Другой пример косвенной амортизации имеет место, когда оборудование применяется одновременно для выполнения многих заданий (освещение, вентиляция, производственные помещения и т. п.). В отличие от прямой амортизации, ложащейся полностью на стоимость выполнения определенного задания, косвенная амортизация распределяется между всеми заданиями, выполнявшимися в течение некоторого установленного промежутка времени (календарного года, квартала и т. п.). Распределение это проводится пропорционально прямой (минимальной или фактической) стоимости выполнения заданий. Под прямой стоимостью здесь понимается доля полной стоимости, не учитывающая косвенной амортизации и других накладных расходов.

Накладные расходы связаны с необходимостью расхода ресурсов (материалов, зарплаты и др.), обеспечивающих выполнение не какого-то определенного задания, а целого ряда заданий, и для работы рассматриваемого объекта в целом (расход электроэнергии на освещение, оплата административно-управленческого персонала и т. п.). Для этих расходов также желательно иметь технологически необходимые нормативы*. В минимальную стоимость включается только та часть расходов, которая укладывается в эти нормативы, а в фактическую стоимость— все эти расходы. Распределение стоимостей между заданиями производится тем же способом, что и при косвенной амортизации.

Что касается доли стоимости от затрат предметов труда и живого труда, то исходная информация о них должна быть заключена в технологической документации (в технологических картах). Для каждой операции должны быть указаны (как и для оборудования) технологически необходимое время и квалификация труда, который необходимо затратить на ее выполнение. Для определения доли минимальной стоимости от затрат живого труда необходимо иметь еще единую стандартную сетку почасовых зарплат по всем специальностям и квалификациям. Умножая величины этих зарплат на технологически необходимое время и складывая (если на операции занято несколько человек), получаем долю живого труда в минимальной стоимости.

Для подсчетов фактической стоимости берутся фактические затраты рабочего времени (без учета простоев) и фактические повре-

---

* Нормативы на административно-управленческий персонал должны разрабатываться при строительстве и реконструкции соответствующих объектов, особенно — в связи с внедрением АСУ.

менные зарплаты или, при сдельной оплате — фактическая стоимость выполнения операции. Фактическая зарплата может при этом сильно отличаться от норм по утвержденной сетке, которые строятся на макроэкономическом уровне лишь в качестве необязательных для каждого индивидуального случая средних нормативов. Важно лишь, чтобы средние уровни фактических зарплат в целом по стране (или по отдельным большим районам) мало отличались от нормативных средних.

Простои при повременной зарплате учитываются в качестве накладных расходов. В минимальную стоимость попадает лишь та часть этих расходов, которая связана с технологически необходимыми простоями, например, простои в системах массового обслуживания, неизбежно возникающие (см. параграф 5 гл. II), если стремятся уменьшить очереди на обслуживание. В фактическую стоимость при повременной зарплате попадают не только технологически необходимые, но и чистые простои. При сдельной оплате оплата простоев в фактическую стоимость естественно не входит.

Как технологически необходимое, так и фактическое время, приходящееся на одну операцию, зависят от размера партии. В расчете фактической стоимости это не имеет значения, поскольку в нее попадают все расходы. В минимальную стоимость включается лишь то время, которое затрачивается при изготовлении оптимальных партий, даже если по тем или иным соображениям истинная партия отличалась от оптимальной. Например, если время на подготовку к выпуску партии 20 *мин*, а технологически необходимое время для изготовления одной единицы продукта 5 *мин*, то при партии в две единицы на каждую единицу придется

$$\frac{20+2\times 5}{2}=15 \text{ мин},$$

а при партии в 20 единиц

$$\frac{20+20\times 5}{20}=6 \text{ мин}.$$

Величины оптимальных партий подсчитываются АСУ способами, описанными в следующем параграфе, а также могут указываться ориентировочно в паспортах соответствующего оборудования (все сказанное имеет место не только для живого труда, но и для времени работы оборудования).

Затраты предметов труда в натуральном выражении должны приводиться в технологических картах (так же как затраты живого труда и основных фондов) в двух видах: минимальный технологически необходимый и фактически планируемый. Они употребляются соответственно для подсчета минимальной и фактически

планируемой стоимости. Фактическая же стоимость подсчитывается на основании текущей учетной информации о фактическом выполнении плана. Для предметов труда (полуфабрикатов), изготовляемых на рассматриваемом объекте, стоимости (минимальная и фактическая) рассчитываются в АСУ в соответствии с описанными правилами, для покупных изделий и материалов их соответствующие данные должны поступать от поставщиков.

Описанные процедуры составляют основу комплекса задач *технико-экономического анализа*. На базе информации о раскладке минимальной и фактической стоимости (с учетом прибыли) и себестоимости (стоимость минус прибыль) каждого задания по всем видам затрат и аналогичной раскладки по затратам ресурсов в натуральном выражении строится справочно-информационная система, позволяющая анализировать узкие места и разрабатывать задания на развитие объекта. Из этой системы непосредственно видно, когда следует планировать замену оборудования, выявлять, какие виды потерь или какие виды технологических операций дают наиболее весомый вклад в себестоимость продукции и т. п. Творчески осмысливая эту информацию, руководство объекта сможет правильно решать вопросы о реконструкции объекта, заявках на новое оборудование, заданиях на совершенствование технологии. Конструкторам и технологам при разработке новых видов продукции даются рекомендации, как обходить те узкие места, которые выявились в результате технико-экономического анализа и которые не удастся устранить к моменту окончания разработок. Выявляются резервы в составляющих объект подобъектах и устанавливаются объективно обусловленные задания по росту производительности труда, снижению себестоимости и т. д. и т. п.

Следует подчеркнуть, что главное назначение описанного комплекса — не бухгалтерский учет, а именно технико-экономический анализ. Поэтому, хотя на его базе можно перестроить методы решения многих чисто бухгалтерских задач, возможен и такой вариант, когда он сочетается с традиционными бухгалтерскими методиками расчета амортизации и других составных элементов ценообразования. Такой подход может иногда оказаться единственно возможным, особенно когда АСУ вводится локально на одном объекте. Другая трудность при локальном вводе систем — организация прогноза морального старения оборудования. При наличии на объекте квалифицированных специалистов, своевременно снабжаемых патентной, рекламной и другой необходимой информацией, можно добиться значительного улучшения всего комплекса работ, связанных с улучшением использования и своевременной заменой оборудования.

Не останавливаясь подробно на комплексе рутинных задач, бухгалтерском учете и финансовой деятельности объекта, сделаем

лишь несколько замечаний в связи с внедрением АСУ. Основное заключается в том, что бухгалтерские задачи должны опираться на общую информационную базу системы.

1. В АСУ, например, имеется информационный массив, описывающий состояние запасов на складе (для предприятий — по самой подробной номенклатуре). Он необходим для решения задач планирования и оперативного управления и должен корректироваться с помощью потока документов об элементарных операциях выдачи и приема на склад тех или иных количеств хранящихся там продуктов в натуральном выражении. Имея массив цен на соответствующие продукты, легко построить программы, ведущие не только натуральный, но и денежный учет имеющихся запасов. Проводимые в рамках бухгалтерского учета инвентаризации, могут служить основой для периодической глобальной проверки правильности ведения соответствующих информационных массивов и их общей разовой коррекции. Аналогичное положение имеет место при начислении зарплаты, поскольку документы о фактических затратах рабочего времени, изготовлении тех или иных партий изделий поступают в систему для решения задач технико-экономического анализа.

2. Необходимость специального информационного массива, который ведет непрерывный учет поступлений и выплат на расчетном счете объекта. Новыми здесь являются два обстоятельства. Во-первых,— такая тесная связь этого массива с комплексами задач планирования и оперативного управления, ибо деньги это тоже ресурс, причем очень важный. Поэтому, когда при описании задач планирования (прежде всего оперативно-календарного планирования) говорим о необходимости наличия ресурсов для выполнения плана, нужно постоянно иметь в виду, что речь идет и об этом виде ресурса. Например, заказ, для которого имеются все необходимые материальные и трудовые ресурсы, тем не менее может оказаться невыполнимым для объекта, если предусмотренная система оплаты заказа приводит к исчерпанию денежных ресурсов до получения денег от заказчика.

Во-вторых, поскольку в АСУ ведутся массивы материально-технического снабжения (ожидаемые сроки поступления заказанных материалов или комплектующих изделий) и плановые массивы (ожидаемые сроки окончания выполнения заказов), то в финансовом массиве возможен не только повседневный учет, но и прогноз финансового состояния. Это дает возможность своевременно корректировать планы, если объект оказывается перед угрозой исчерпания денежных ресурсов.

3. Включение в систему средств, позволяющих осуществлять гибкое экономическое и моральное стимулирование. Под гибкостью понимается возможность быстро менять систему оплаты (прежде

всего премирования) в зависимости от результатов технико-экономического анализа (вскрывающего факты плохого отношения к труду) и от данных, получаемых в результате продуманной системы *социологических опросов* (позволяющих наметить меры для устранения подобных фактов). Выработка самих принципов стимулирования при этом остается за человеком. Задача же АСУ состоит в том, чтобы иметь средства быстрой перестройки программ начисления зарплаты и премий в соответствии с предложенными принципами. Должны быть обеспечены также средства автоматизации решения задач системного анализа (моделирования) при сравнении различных вариантов стимулирования и учета их далеких последствий.

Следует иметь в виду, что среди средств материального стимулирования могут выступать не только прямые, но и косвенные меры. Например, предприятие может вести личные счета работающих, осуществляя безналичные расчеты на льготных условиях при покупках предметов длительного пользования (автомобили, мотоциклы, мебель, телевизоры и т. п.).

Льготные условия могут состоять в установлении очередности продаж, взятия части расходов на счет предприятия, предоставление кредита в зависимости от длительности стажа работы на данном предприятии и т. п.

Переходя к описанию комплексов задач плановых расчетов, заметим, что все виды планирования в АСУ оказываются тесно переплетенными. Планы прежде всего различаются по их *направленности*. План основной деятельности объекта заключается в том, чтобы фиксировать задания и способы их выполнения по производству продуктов, для которых данный объект предназначен. План развития объекта представляет собой комплекс заданий и мероприятий, направленных на создание предпосылок для выполнения плана основной деятельности. Этот план разделяется на планы строительства новых объектов, реконструкции старых, поставок оборудования и технической подготовки производства. Последний план предусматривает задания и мероприятия по их выполнению например, при наличии соответствующих подразделений на объекте, а также задания на подготовку проектно-технологической документации для новых или модернизированных изделий, предназначенных для выпуска на данном объекте.

По срокам, на которые составляются планы, различают перспективное, долгосрочное, краткосрочное и оперативное (текущее) планирование. Срок перспективного планирования определяется возможностью более или менее точного прогноза коренных изменений в технологии и составления программ развития, которые должны служить основой при составлении планов на более короткие периоды. Ориентировочный срок такого планирования 15—

20 лет вперед от настоящего момента*. Срок долгосрочного планирования определяется сроками окончания мероприятий по развитию объекта, которые можно достаточно точно спланировать по определенным срокам. Ориентировочно этот срок составляет 5 лет вперед от настоящего момента.

Срок краткосрочного планирования определяется прежде всего возможностью более или менее точно фиксировать поток заданий, привязанных к конкретным потребителям и согласованных по объемам, номенклатуре и срокам поставок. Для краткосрочного планирования основной деятельности объекта важно также иметь достоверную информацию о сроках завершения различных этапов планов развития объекта, приходящихся на интервал краткосрочного планирования. Реальные сроки краткосрочного *непрерывного* планирования зависят от размеров объекта. Для завода тяжелого машиностроения срок выполнения заказа может составить несколько лет, а швейная фабрика должна гибко менять номенклатуру в соответствии с изменением спроса. Ориентировочно средний срок краткосрочного планирования можно принять равным одному году.

Задача оперативного (текущего) планирования — выработать конкретные *сменные задания* по цехам, участкам, бригадам, звеньям. Срок такого планирования определяется непредвиденными изменениями в ресурсах, условиями работы (погода на сельскохозяйственных работах). В потоке заданий вряд ли целесообразно рассчитывать сменные задания на месяц вперед, если нет уверенности, что изменение ситуации не заставит производить все расчеты заново. Для массового производства с постоянной номенклатурой и устойчивым спросом срок оперативного планирования может исчисляться месяцами, в других же случаях (например, для организаций, осуществляющих экстренные ремонтные работы) может уменьшаться до одного дня или даже до одной смены.

Для министерств задача оперативного планирования сливается с задачей *оперативного управления*, заключающейся в оперативной корректировке краткосрочных планов на основании информации о их выполнении.

По методам решения задач и способам представления результатов плановые расчеты в АСУ делятся на объемные, объемно-календарные и собственно календарные, разделяющиеся на оперативно-календарные и сетевые. При объемных расчетах ресурсы, материально-техническое снабжение и другие элементы плана рассчитываются на весь плановый период без разбивки на более мелкие. Объемно-календарное планирование предполагает грубую разбивку планового периода (например, год разбивается на кварталы и ме-

---

* Как уже было упомянуто выше, при внедрении АСУ планирование должно стать непрерывным.

сяцы). Оперативно-календарное планирование осуществляет точную привязку (в пределах разрешаемых допусков) проводимых мероприятий к календарю. Степень точности привязки определяется прежде всего видом плана и степенью удаленности мероприятия от настоящего момента: для долгосрочного планирования хорошей точностью может быть неделя, а для оперативного планирования иногда необходимо указать не только часы, но и минуты.

Различие между оперативно-календарным и сетевым планированием заключается в том, что в первом указывается определенный календарный срок для выполнения любой операции (время начала и конца операции), а во втором — задается плавающий срок с точным указанием допустимых пределов для сроков начала и конца операции (см. параграф 4 гл. II).

Планы на различные сроки в АСУ должны составлять единую систему и вытекать один из другого: долгосрочный план должен быть детализацией начального периода перспективного плана, краткосрочный план — детализацией начального периода долгосрочного плана и т. д. Изменения в более долгосрочных планах должны *немедленно* приводить к соответствующим изменениям в планах на более короткий срок и наоборот. В этом состоит одно из требований принципов непрерывности и *динамичности* планирования.

Всякое планирование начинается с постановки целей, задания критериев и фиксации предварительного потока заданий для рассматриваемого объекта. В результате выполнения комплекса всех плановых расчетов должна произойти фиксация окончательного потока заданий для данного объекта, а также выдача целей, критериев и потока заданий для отдельных составных частей объекта (план деятельности самого объекта) и других объектов (план материально-технического снабжения, задания строительным, монтажным и проектно-конструкторским организациям). Окончательный план должен быть представлен в виде *календарного* плана (оперативно-календарного или сетевого).

Обычно планы на длительные сроки представляются в виде сетевых графиков, а начальные участки графиков в пределах действия краткосрочных и оперативных планов переводятся в оперативно-календарную форму. Такой перевод (всего графика целиком) был продемонстрирован в параграфе 4 гл. II). Как следует из этого примера, для требуемого преобразования графика необходимы сведения о наличных ресурсах и задание критерия оптимизации (в рассмотренном примере таким критерием служила минимизация потерь критического ресурса).

Отметим еще одну особенность автоматизированных плановых расчетов. Если на каком-либо объекте (производственном объединении) есть АСУ, которая решает все задачи планирования и управ-

ления для входящих в ее состав подобъектов, то выработанные в АСУ плановые задания передаются в подобъекты в качестве директивных (обязательных для выполнения) документов. При иерархическом построении АСУ положение меняется. Например, *отраслевая автоматизированная система управления* (ОАСУ) состоит не только из *главного вычислительного центра* (ГВЦ) министерства, но и из совокупности *автоматизированных систем управления предприятиями* (АСУП), куда включаются и АСУ производственных объединений. Наиболее эффективным является *итерационный метод* составления плана, когда в результате обобщенных расчетов ГВЦ министерства направляет в АСУПы *предварительные* потоки заданий (среди которых определенная часть может иметь и директивный характер). Эти потоки служат основой более детальных плановых расчетов в АСУП, после чего *принятые* потоки заданий возвращаются в ГВЦ для второго цикла расчетов и рассылки нового потока заданий, в которых доля директивных заданий увеличена по сравнению с первым циклом, и так далее, пока не будут получены окончательные потоки заданий. Проведение аналогичных итераций необходимо также со смежниками, чтобы планируемые сроки выполнения заданий у поставщиков совпадали со сроками их ожидания (по планам материально-технического снабжения основной деятельности и планами развития).

В соответствии с сформулированным принципом *непрерывности* планирования подобные полные плановые расчеты делаются только один раз с последующей корректировкой, заключающейся в добавлениях и изменениях небольшого числа заданий, в частности,— добавление новых заданий на непрерывно сдвигающийся конечный участок соответствующего интервала планирования. Особенно важным является соблюдение и другого принципа — *динамического представления взаимоувязанных* планов в АСУ различных ведомств и уровней. Это означает необходимость передачи в АСУ смежников всех затрагивающих их изменений планов в момент, когда эти изменения производятся, а не тогда, когда произойдет срыв поставок. Поступающие изменения должны автоматически или в режиме диалога человек-машина за минимальное время обеспечить соответствующую коррекцию своих собственных планов и *немедленную передачу* в АСУ своих заказчиков затрагивающих их изменений (если, конечно, таких изменений не удастся избежать).

Следует иметь в виду, что для увеличения эффективности (быстроты работы) подобных динамических моделей приходится употреблять сокращенные нестандартные системы обозначений ресурсов, материалов, наименований заказов и т. п. В то же время для взаимных обменов требуется пользоваться унифицированными

кодами, установленными в государственных или отраслевых классификаторах. Поэтому в комплекс задач планирования вводятся вспомогательные процедуры для перевода информации из местной системы кодирования в унифицированную и наоборот.

Основой планирования являются планы основной деятельности экономических объектов. Планы развития объектов должны вытекать из основных планов, обеспечивая выполнение увеличивающегося потока заданий. Поэтому общая технологическая схема планирования выглядит следующим образом: дальний прогноз (на срок перспективного планирования) видов заданий и технологии их выполнения*, разработка длительных программ экономического развития (например, программы жилищного строительства и комплексного развития городов и сел или программы комплексного развития энергетики). Разработка потоков заданий для отраслей народного хозяйства на период долгосрочного планирования с более подробной детализацией его на (усредненный) срок краткосрочного планирования. Все эти задачи решаются в основном на макроэкономическом уровне, роль ОАСУ и АСУП состоит в снабжении верхнего уровня обобщенной технологической информацией, участии в системном анализе возникающих проблем и в решении балансовых задач по заданиям *общегосударственной автоматизированной системы управления* (ОГАС).

В результате этого этапа плановых расчетов отрасли получают ориентировку по перспективе и *предварительный поток заданий по основной деятельности*. При этом на макроэкономическом уровне устанавливается поток заданий для отраслей, обслуживающих непроизводственное потребление (жилищное строительство, производство товаров народного потребления, просвещение, культура, здравоохранение, оборона и т. п.). Остальные отрасли формируют потоки своих заданий по мере выработки заказов в отраслях, являющихся потребителями их продукции.

Как уже отмечалось выше, одним из источников формирования потоков заданий является прогноз спроса, который может выполняться как в ОАСУ, так и в АСУП всякий раз, когда дело идет о *неорганизованном* спросе. Для прогноза спроса на срок краткосрочного планирования основным инструментом служат методы экстраполяции (см. параграф 8 гл. II). Для более длительных интервалов, а также в случаях, когда ожидается резкое изменение спроса на помощь экстраполяции приходят методы системного анализа и экспертных оценок (см. параграф 7 гл. II). Прогноз спроса обычно сопровождается оценкой возможных ошибок. При экстраполяционном методе для оценок ошибок служит *ретроспективный анализ*,

* Речь идет не только о заданиях на производство продукции, но и на предоставление различных услуг, например технологии учебного процесса.

т. е. применение используемых при прогнозе методов для оценок настоящего и прошлого спроса.

Соответствующие данные о прошлом и настоящем спросе включаются в информационную базу системы. Эти данные вместе с процедурами их анализа и экстраполяции можно рассматривать как отдельную подсистему, называемую *подсистемой сбыта*. Часто в подсистему сбыта включается специальная информационно-справочная система, содержащая различные сведения о фактических и потенциально возможных потребителях продукции рассматриваемого экономического объекта. Через нее замыкают поток поступающих заказов, учитывая, что перед передачей заказов в систему планирования нужно проводить необходимую статистическую обработку для улучшения процедур прогнозирования спроса и ретроспективного анализа.

При большом объеме экспортных поставок в подсистему сбыта могут включаться средства для системного анализа конъюнктуры на мировом рынке. Особенно эффективным такой анализ оказывается в случае, когда он организован в национальном масштабе в виде постоянно действующей системы, а отдельные экономические объекты имеют доступ к соответствующим разделам этой системы.

Формирование потока заданий для долгосрочного планирования обычно ограничивается объемными показателями по годам или кварталам. Для краткосрочного планирования задания (заказы) должны иметь конкретные адреса и конкретные сроки их выполнения.

Для организации эффективной системы оптимального планирования в национальном масштабе недостаточно, однако, просто сформировать потоки заданий от одних экономических объектов к другим. Ведь с точки зрения потребителей, скажем, комбайнов, наиболее выгодно выполнение заказов на поставку новых комбайнов отнести к началу уборки урожая. Для промышленности же, производящей комбайны, целесообразно распределять выполнение заказов равномерно на весь год.

Поэтому при формировании заданий (в виде директивных планов или заказов от других объектов) каждое задание (заказ) должно сопровождаться оценкой потерь, которые возникнут у потребителя при несвоевременном или неполном выполнении задания. Следует иметь в виду, что потери могут иметь место не только при опоздании, но и при досрочном выполнении заказа (расходы на хранение, омертвление средств, которые могли бы быть использованы в другом месте, риск выхода заказанных товаров из употребления). Иногда досрочное выполнение заказа приносит заказчику выгоду, а не потери. В этом случае удобно исчислять получаемую выгоду как отрицательные потери.

Оценки потерь, связанные с несвоевременными выполнениями заказов, бывают двух видов: на стадии планирования и оперативного управления. На стадии планирования, когда у заказчика есть время для перестройки планов в связи с выяснившейся невозможностью своевременного выполнения заказа, эти потери обычно значительно меньше, чем в случае выяснения этого обстоятельства лишь в тот день, когда заказ должен был бы быть выполненным.

Строго говоря, для оптимальных расчетов нужны оценки типа формул или алгоритмов, позволяющих точно рассчитать потери при любых отклонениях срока выполнения заказа. Однако получение таких универсальных оценок у заказчика является довольно затруднительным. Причины, затрудняющие получение точных оценок (тем более универсальных) следующие: 1. Потери у прямого заказчика могут вызывать потери у косвенных заказчиков, т. е. у заказчиков этого заказчика и т. д. Для точных расчетов необходимо учитывать все эти опосредствованные потери, а они могут замыкаться в циклы, выводя в конце концов вновь на исходный объект. 2. Наличие *интегрального эффекта*: дело в том, что суммарные потери заказчика от несвоевременного выполнения нескольких заказов обычно существенно больше простой суммы потерь от задержек выполнения единичных заказов. Для оптимальных же расчетов нужен именно этот интегральный эффект.

Поэтому оценки, о которых идет речь, носят, как правило, достаточно грубый характер и применяются не столько для точных расчетов, сколько для ориентировки в отношении перемещения сроков выполнения различных заказов при разработке планов реализации поступивших заданий. Проведение более точных расчетов по взаимной увязке планов требует организации совместной работы АСУ различных экономических объектов, при которой скорость обмена данными между ними не являлась бы тормозом при проведении расчетов. Обычные формы согласования планов путем переписки здесь не подходят — нужен прямой контакт между ЭВМ через соответствующие каналы связи.

Грубые оценки, о которых идет речь, могут иметь следующий вид. Заказ: поставить к 15/IV партию из 125 штук изделий ДС835. Допуск с нулевыми потерями (—10, +1) дней, допуск с 10% потерями (от стоимости заказа) — (—50, +5) дней, допуск с 25% потерями (—100, +10) дней. Возможны, разумеется, и другие формы представления оценок.

Кроме оценок потерь для осуществления дальнейших этапов планирования необходимо иметь еще систему ориентировочных *приоритетов* заказов. В директивном плане такие приоритеты должны быть заданы вышестоящими органами, а при формировании плана — на основе заказов (имеющихся или прогнозируемых).

Основой для шкалы приоритетов может служить относительная выгодность заказов. Для определения выгодности можно воспользоваться точными расчетами по каждому из заказов. Однако на данном этапе обычно достаточно ограничиться грубыми экспертными оценками.

После формирования предварительных потоков заданий на период долгосрочного планирования начинается этап объемных и объемно-календарных расчетов. Методика их проведения по существу одинакова, различаясь лишь тем, что при объемном планировании расчет производится на весь плановый период, а в объемно-календарном планировании плановый период разбивается на несколько участков.

Основой таких расчетов служат данные о необходимых затратах ресурсов, получаемых из технологической информации*. Представляется целесообразным применять при этом минимальные (технологически необходимые) нормативы, а их отличие от фактических нормативов учитывать с помощью коэффициентов *рабочего* и *полного* использования ресурсов. Чтобы пояснить смысл этих коэффициентов, предположим, что технологически необходимая норма затраты времени станка на изготовление некоторой детали равна 6 *мин*. В действительности же на ее изготовление тратится 10 *мин*, хотя станок при этом все время находится в работе. Коэффициент рабочего использования станка будет равняться $6 : 10 = 0{,}6$. Если к тому же станок работает лишь 8 часов в сутки, а остальное время простаивает, то полное время (с учетом простоев), отнесенное на одну деталь, будет составлять уже не 10, а 30 *мин*. Коэффициент полного использования станка будет составлять $6 : 30 = 0{,}2$.

Фактический коэффициент полного использования оборудования полезно дополнить технологически необходимым, который учитывает лишь минимальные нормы расхода времени на рабочие операции и технологически необходимые простои (в первую очередь на текущие ремонты и профилактику, а также на подготовительные операции из расчета выпуска оптимальных партий). Различие между этими двумя коэффициентами представляет собой резерв, рассматриваемого экономического объекта.

Как известно, резервы ресурсов необходимы, без них невозможно было бы реальное управление и выполнение намеченных планов. При обычных методах управления эти резервы складываются стихийно и в большей своей части служат для оперативного восполнения различного рода «огрехов» в планировании.

---

* Напомним, что такие данные пусть в грубо ориентировочном виде должны иметься и для продукции, проектно-технологическая документация на которую только начинает разрабатываться.

При внедрении АСУ появляется возможность перехода к расчетам необходимых резервов и употреблению *расчетных коэффициентов полного использования ресурсов.* В расчетном коэффициенте учитываются не только технологически необходимые простои, но и простои, связанные с различного рода директивными ограничениями (например, когда требуется, чтобы работа велась в одну или две смены), дополнительные расходы времени на подготовку при обоснованной необходимости производства неоптимальных партий (например, срочное изготовление одной детали для ремонта уникального оборудования), простои, обусловливаемые ненадежностью (несмотря на правильное обслуживание) самого оборудования или факторов, обеспечивающих его нормальную работу (например, вероятные потери от внезапных перерывов в энергоснабжении), и, наконец, обоснованные резервы, необходимые для выполнения срочных внеплановых заданий (исходя из вероятности получения таких заданий).

Произведя расчет коэффициентов полной загрузки всех имеющихся видов оборудования и рабочей силы, вычисляют общее количество ресурсо-дней (или ресурсо-часов), имеющихся на расчетный период. Сюда включаются все новые ресурсы, которые должны появиться в течение рассматриваемого периода в соответствии с планами развития объекта (если таковые уже имеются). Далее по технологической документации определяют общее количество ресурсов, необходимых для выполнения всех имеющихся (предварительных) заданий. При этом производится и качественная проверка возможности выполнения заданий (например, наличие стапелей и сухих доков необходимых размеров или кранов необходимой грузоподъемности).

В соответствии с приоритетами набирается максимально возможный набор заданий, выполнимый с помощью наличных и уже запланированных ресурсов. Остальные задания временно исключаются из плана. Одновременно выявляются один или несколько критических ресурсов, которые не позволяют принять дополнительно ни одно из временно исключенных заданий.

До сих пор расчеты носили рутинный характер. Теперь выступает одно из главных преимуществ АСУ — возможность *оптимизации* планов. На отраслевом уровне объемно-календарные расчеты делают обычно, распределив поток заданий по предприятиям. При ручном распределении зачастую оказывается, что критические ресурсы на разных предприятиях различны. Поэтому за счет перераспределения потока заданий между предприятиями можно значительно увеличить возможности отрасли по выполнению заданий, без дополнительных капитальных вложений. По существу речь идет об установлении и поддержании правильной специализации предприятий. Такая задача (см. параграф 2 гл. II) может быть

решена методами линейного программирования (критерий максимизации прибыли можно легко заменить любым другим линейным критерием, например максимизацией суммарной стоимости выполненных заданий).

На уровне предприятия аналогичные задачи возникают при учете возможностей выполнения отдельных операций различными способами или с помощью различных режимов.

Таким образом, теоретически задача полной оптимизации объемно-календарного плана всей отрасли или отдельного предприятия решается методами линейного программирования. Однако на практике формальное применение методов линейного программирования ко всему плану может приводить к задачам слишком большой размерности, трудным для решения на имеющейся технической базе. Кроме того, далеко не всегда (особенно при индивидуальном и мелкосерийном производстве) удается осуществлять дробление отдельных заказов, что приводит к необходимости решения еще более трудоемких задач целочисленного линейного программирования, а при более сложных критериях и ограничениях — к задачам нелинейного программирования.

Поэтому на практике наиболее употребительными являются методы *последовательной частичной оптимизации*. Суть их состоит в том, чтобы в полной задаче выделять *варьируемые части* и известные математические методы (в том числе методы линейного программирования) применять лишь к этим частям. На отраслевом уровне варьирования можно произвести для части предприятий и части заданий (в том числе и временно исключенных). Критерием для отбора предприятий служит при этом различие критических ресурсов, а для выбора заданий — возможность (качественная) их выполнения на данных предприятиях. На уровне предприятий варьирование производится отдельными операциями, в которых возможна замена критических ресурсов некритическими. Для транспортных предприятий оптимизация плана осуществляется с помощью решения транспортных задач и за счет правильного распределения перевозок между различными видами транспорта.

После выполнения одного цикла такой оптимизации может проводиться второй цикл — с новыми варьируемыми частями, и так далее, пока новые циклы оптимизации не станут давать лишь пренебрежимо малые улучшения плана*.

В результате выполнения полного цикла объемно-календарных расчетов, даже при условии оптимизации планов основной деятельности и наличии уже принятого плана развития объекта, может выявиться недостаток тех или иных ресурсов. Отсюда возникают

---

* Возможно прекращение дальнейшей оптимизации и по другим причинам, например по причине нехватки машинного времени.

задания на капитальное строительство и реконструкцию в рамках рассматриваемого объекта. При этом в первую очередь просматриваются варианты реконструкции и наращивания ресурсов в уже существующих объектах. Причина заключается в том, что даже при наличии оптимизации не удается достигнуть полной равномерности использования ресурсов. Пусть, например, из 50 видов основных ресурсов на каком-либо предприятии только один оказывается загруженным на 100% (с учетом расчетного коэффициента полной загрузки), а остальные — на 50%. Чтобы увеличить общий выпуск продукции вдвое за счет нового строительства, нужно создать заново 100% имеющегося критического ресурса и по 50% всех остальных ресурсов. Тот же результат может быть получен за счет дополнительной поставки на имеющееся предприятие лишь 100% критического ресурса.

Таким образом, неравномерность загрузки основных ресурсов в долгосрочном плане делает соответствующий объект предметом рассмотрения для включения в план реконструкции и поставок нового оборудования. Окончательный план строительства, реконструкции, поставок оборудования и подготовки кадров выбирается на основании сравнительного анализа многих вариантов в человеко-машинном режиме использования АСУ. Он представляется в виде иерархически построенной системы сетевых графиков и служит основой окончательного (календарного) долгосрочного плана.

План же основной деятельности для всех участков периода долгосрочного планирования, кроме первого, остается до поры до времени на объемно-календарном уровне. В зависимости от объема утвержденного плана наращивания основных ресурсов происходит изменение предварительного потока заданий (исключение части предварительных заданий а, возможно, и добавление новых).

Одновременно при расчете ресурсов возникает объемно-календарный план материально-технического снабжения. Поскольку планы снабжения для одних объектов служат планами производства для других, на макроэкономическом уровне организуется итерационный процесс их взаимной увязки по грубым календарным срокам. Лишь в результате этого процесса предварительные потоки заданий превращаются в директивные, хотя при этом и не исключается полностью возможность их последующей корректировки.

Наряду с планами материально-технического снабжения из долгосрочного плана основной деятельности объекта возникает поток заданий на подготовку проектно-технологической документации. Эти задания и задания на строительство и реконструкцию либо через макроэкономический уровень, либо на основании прямых связей вливается в общий поток заданий по основной деятельности для соответствующих проектно-конструкторских, строительных

монтажных и пуско-наладочных организаций. Все эти планы также представляются в виде сетевых графиков.

При работе по согласованию потоков заданий соответствующие организации используют методы оптимизации использования ресурсов не только на основании объемно-календарных расчетов, но и на основании соответствующих методов для сетевых графиков. (см. параграф 4 гл. II).

При планировании проектно-конструкторских работ принимаются во внимание результаты расчета ресурсов при объемно-календарном планировании основной деятельности заказывающих эти работы объектов. Выявляющиеся неравномерности загрузки ресурсов служат основой для выработки рекомендаций конструкторам и технологам: при своих разработках стремиться, чтобы технология изготовления новой продукции по возможности не использовала бы критических ресурсов и способствовала тем самым выравниванию использования ресурсов и уменьшению капитальных вложений. Большое значение имеет также создание специальной справочно-информационной системы по имеющейся проектно-технологической документации. Пользуясь этой системой, конструктор вместо новой разработки какой-нибудь детали или узла может воспользоваться уже имеющимися решениями. Тем самым не только сокращается время самой разработки, но и упрощается освоение нового изделия в производстве.

При организации планирования и управления на основе сетевых графиков (не только в проектно-конструкторских работах, но также в строительстве и др.) необходимо иметь в виду два обстоятельства. Во-первых, при составлении сетевых графиков и особенно при оценке продолжительностей работ нужно, как и при определении нормативов, иметь двойную систему стимулирования (исполнители стимулируются за досрочное выполнение заданий, а их руководители и составители графиков — за точность планирования). Во-вторых, необходимо обеспечить регулярную корректировку графиков (продолжительности работ) и выдачу информации для других задач планирования и управления. Это касается прежде всего прогнозов сроков окончания различных этапов работ, а для проектно-конструкторских работ также уточняющейся с каждым разом информации о предполагаемых нормативах при производстве проектируемых изделий.

При строительстве новых объектов приходится решать еще один класс оптимизационных задач, а именно, задач *оптимального размещения* строящихся объектов. Здесь прежде всего приходится решать задачу о *минимизации транспортных расходов* по доставке сырья и готовой продукции. Даже в простейшем предположении, когда транспортировка от предполагаемых поставщиков и к вероятным потребителям производится по кратчайшим геометрическим

(прямым) путям, возникает общая задача оптимизации (нелинейного программирования). Разумеется, ее можно решать методами, приведенными в параграфе 1 гл. II, однако, несмотря на значительные вычислительные трудности, результат решения будет иметь лишь весьма условную ценность.

При решении реальных задач размещения приходится учитывать много дополнительных факторов (структуру уже существующей транспортной сети, необходимость нового жилищного строительства и т. п.). На практике обычно ограничиваются либо простыми *многовариантными* (при наличии АСУ) расчетами, либо сочетают их с идеями градиентного метода. В первом случае просто просчитывается несколько (при наличии АСУ несколько десятков) возможных мест размещения и суммарные стоимости этих вариантов. При этом наиболее важное значение имеет не столько сумма необходимых капитальных вложений, сколько сравнительная оценка предлагаемой стоимости выпускаемой продукции с учетом транспортных расходов. Важным моментом является необходимость включения в эту стоимость амортизации не только самого объекта, но и всех вспомогательных сооружений (дорог, трубопроводов, жилья и т. п.), сбеспечивающих работы именно этого объекта.

Добавление градиентного метода состоит обычно в том, что производятся сдвиги предполагаемого места строительства от выбранного пункта, в направлении от подходящих к нему железнодорожных, водных, автомобильных путей, а также не полностью загруженных трубопроводов требуемого типа. При обнаружении направления максимального убывания транспортных расходов в качестве следующего расчетного варианта выбирается ближайший подходящий (по качественно проверяемым условиям) пункт, лежащий на выбранной транспортной линии.

Оптимизация краткосрочного плана объемными методами должна быть выполнена для соответствующего интервала t, как части интервала T долгосрочного плана. Краткосрочный план должен быть представлен не только в объемно-календарном, но и в чисто *календарном* виде (сетевом или оперативно-календарном). Поэтому возникают дополнительные возможности оптимизации за счет правильного распределения заданий и отдельных составляющих их операций во времени, а также за счет правильной группировки как исходных заказов, так и порождаемых ими заданий в партии, близкие к оптимальным. Благодаря этим мерам удается поднять значения коэффициентов полного использования ресурсов.

Одной из главных причин малого значения этих коэффициентов при отсутствии АСУ является неритмичность работы объекта, приводящая к тому, что в одни периоды определенные ресурсы оказываются перегруженными, а в другие — простаивают. Иногда это может быть вызвано технологической необходимостью (сахарные

заводы, уборка урожая и т. п.). Однако в большинстве случаев основная причина заключается в недостатках планирования.

Точные методы решения задач календарного планирования основаны на применении методов динамического программирования (см. параграф 3 гл. II). Но над этими методами еще в большей степени, чем для задач линейного программирования довлеет «проклятие размерности». Поэтому практические методы основываются на последовательных частичных приемах оптимизации. При таком подходе динамическое программирование и другие точные методы используются локально, например, при определении наилучшей последовательности выполнения некоторого множества отобранных операций. Другим приемом, позволяющим резко уменьшить число рассматриваемых вариантов, является применение различных эмпирических *правил предпочтения*. Такие правила применяются для того, чтобы из технологически возможных вариантов указывать тот выбор, который в большинстве встречающихся случаев приводит к лучшему плану. Наконец, широкое применение находят различные эмпирические методы, не допускающие никаких потерь критических ресурсов (см. параграф 4 гл. II).

В процессе оптимизации производится сдвиг сроков выполнения отдельных заданий при условии, что потери у заказчика (определяемые на основе заданных оценок) компенсируются дополнительными прибылями на рассматриваемом объекте так, чтобы при необходимости он мог бы полностью возместить эти потери и остаться еще в выгоде.

Очень важной задачей, решаемой на этапе краткосрочного планирования, является расчет необходимых запасов оборотных фондов, включая запасы деталей и полуфабрикатов на различных стадиях их изготовления. Эти запасы — своеобразные буферы между различными производственными участками, позволяющие согласовывать различные ритмы на участках и обеспечивать продолжение нормальной работы последующих участков при временных срывах и неполадках на предыдущих участках.

Методы расчета запасов, применяемые в АСУ, значительно увеличивают ритмичность работы экономических объектов и коэффициенты использования ресурсов при одновременном существенном снижении незавершенного производства и общего объема оборотных фондов. Являясь основой для краткосрочного и оперативного планирования, эти методы заслуживают более подробного рассмотрения.

Комплекс задач оперативного планирования и управления получает непрерывно информацию о состояниях запасов и осуществляет учет и прогноз выполнения планов всех видов, в частности, прогноз сроков поступления заказов по материально-техническому снабжению. Ведется постоянный учет качества поставляемых мате-

риалов и комплектующих изделий, необходимый для прогноза процента брака и установления нужных режимов работы оборудования (последнее особенно для химии, металлургии и других непрерывных и непрерывно-дискретных производств). Для объектов, не обладающих возможностями создания запасов готовой продукции (электроэнергетика, магазины, торгующие скоропортящимися продуктами), важное место в оперативном управлении занимает оперативный прогноз спроса на эту продукцию.

Из системы оперативного управления идут задания на корректировку планов, на принятие мер по ускорению запланированных поставок, на приведение в действие резервов основных ресурсов при авариях и крупных срывах, которые не могут быть компенсированы за счет буферных запасов незавершенного производства, а также при поступлении незапланированных срочных заданий. Планирование сменных заданий и оперативных заявок на снабжение (например, заявка на порожняк железнодорожному транспорту) тесно переплетено с управлением запасами и будет рассмотрено особо.

Кроме описанных комплексов задач АСУ может включать в себя и другие комплексы, в том числе и относительно изолированные подсистемы. Примером такой изолированной подсистемы могут служить задачи статистического контроля качества продукции, методы решения которых описаны в параграфе 9 гл. II. Другой пример дает локальная справочно-информационная система по контролю выполнения приказов, распоряжений и отдельных поручений руководства.

Большую специфику имеют задачи расчета оптимальных режимов работы энергосистем, комплексов оборудования химических заводов, методы оптимального планирования изысканных и эксплуатационных работ в добывающих отраслях и т. д. Эти и другие специфические задачи имеют много общего с задачами управления технологическими процессами, которые настолько разнообразны, что полное их перечисление в рамках одной книги, тем более книги такого плана как данная, вряд ли возможно и целесообразно. Для их решения можно использовать многие из методов, описанных в предыдущей главе.

5. Управление запасами и оперативно-календарное планирование

Как уже отмечалось выше, процесс функционирования любого экономического объекта складывается из взаимосвязанных цепочек. Задержка в одном из мест этой цепочки может приводить к лавинообразному нарастанию задержек в последующих местах и к большим потерям. Для того чтобы получить высокую степень

надежности всего процесса при не слишком большой надежности отдельных составляющих его частей, прибегают к использованию резервов. Резервы могут быть либо в виде незадействованного оборудования, либо в виде запасных частей, при помощи которых осуществляется быстрый ремонт оборудования, либо, наконец, в виде запасов материалов, полуфабрикатов и готовых изделий для снабжения цепи при временных выходах из строя отдельных звеньев.

Установление правильных уровней запасов различных видов, их постоянный контроль, пополнение и использование — центральный вопрос комплекса задач оперативного планирования и управления. Многие системы оперативно-календарного планирования могут строиться как *системы управления запасами*.

Задачи управления запасами делятся на *статические* и *динамические*. В статических задачах вопрос о создании того или иного запаса выступает как *единичный акт*, для задач динамических расходование и периодическое пополнение запасов рассматривается как *процесс, развертывающийся во времени*.

Характерным примером статической задачи является проблема закупки запасных частей вместе с каким-либо уникальным дорогостоящим оборудованием (например, с прокатным станом, турбогенератором большой мощности и т. п.). При изготовлении заказанного оборудования поставщик может изготовить запасные части по относительно низкой цене. Если же закупать их после прошествия определенного времени, когда в них возникнет потребность, то их цена может оказаться значительно большей (в силу необходимости заново осуществить подготовку производства у поставщика). Кроме того, могут иметь место значительные потери от простоя оборудования в ожидании изготовления необходимой запасной части (вместо вышедшей из строя). Вопрос состоит в том, какое количество запасных частей покупать вместе с оборудованием?

Чтобы ответить на этот вопрос, предположим, что цена некоторой запасной части в момент поставки оборудования равна 1000 руб., убытки же от ее отсутствия в момент аварии равны 20000 руб. (включая закупку ее по более дорогой цене). Пусть далее вероятность p_0 того, что за время эксплуатации данная запасная часть не потребуется, равна 0,90; вероятность p_1 того, что она потребуется один раз равна 0,08, а два раза — $p_2 = 0{,}02$. Если не покупать с оборудованием ни одной запасной части, то математическое ожидание убытка

$$S_0 = 0{,}08 \cdot 20\,000 + 0{,}02 \cdot 2 \cdot 20\,000 = 2400 \text{ руб.}$$

При покупке одного экземпляра

$$S_1 = 1000 + 0{,}02 \cdot 20000 = 1400 \text{ руб}$$

а при покупке двух экземпляров

$$S_2 = 2 \cdot 1000 = 2000 \text{ руб.}$$

Таким образом, в данном случае минимальные потери будут во втором случае, и, следовательно, правильная политика состоит в закупке запасной части в одном экземпляре.

В качестве второго примера рассмотрим случай разового заказа магазином какого-либо скоропортящегося продукта. Пусть $p(n)$ есть вероятность того, что спрос за время реализации будет не менее, чем n единиц; A — прибыль, получаемая при продаже одной единицы продукта, а B — убыток на каждую единицу непроданного продукта. Тогда математическое ожидание прибыли от n-й единицы продукта выразится формулой:

$$E(n) = Ap(n) - B[1 - p(n)].$$

Легко понять, что закупка n-й единицы будет целесообразной, если $E(n) > 0$. Из неравенства $Ap(n) - B[1 - p(n)] > 0$ получаем

$$p(n) > \frac{B}{A+B}.$$

Следовательно, правильной политикой будет закупка наибольшего числа n единиц продуктов, при котором еще имеет место неравенство

$$p(n) > \frac{B}{A+B}.$$

В динамических моделях управления запасами заказы на пополнение запасов даются многократно. Важнейшим здесь является определение *точки заказа*, т. е. момента времени, в который должен быть произведен заказ, и *размера партии*, т. е. количество одновременно заказываемого продукта.

Предположим, что скорость v расходования запаса постоянна (простейший, так называемый линейный случай), а в момент его исчерпания поступает новая партия, пополняющая запас, снова происходит его расходование и т. д. Мы будем называть такую политику заказов политикой нулевых страховых запасов. Пусть A — расход на *подготовку* выполнения заказа у поставщика. В первом приближении можно считать, что эта величина не зависит от размера N заказываемой партии. В таком случае прибавка стоимости на каждую единицу заказываемого продукта выразится величиной $\frac{A}{N}$. При скорости расходования запаса, равной v, запас величиной в N исчерпается за время $T = \frac{N}{v}$. Среднее время $T_{ср}$ хранения одной единицы продукта

$$\frac{1}{2}T = \frac{N}{2v}.$$

При затратах на хранение одной единицы продукта в течение одной единицы времени, равных B, общие затраты на хранение, отнесенные на единицу продукта, представляются в виде $\frac{BN}{2v}$. Общая сумма S дополнительных расходов на единицу заказываемого продукта

$$S = \frac{A}{N} + \frac{BN}{2v}. \tag{43}$$

При сделанных предположениях (постоянство величин A, B, v)

$$\frac{dS}{dN} = -\frac{A}{N^2} + \frac{B}{2v}.$$

Оптимальный размер партии N_0 (дающий минимум дополнительных расходов) достигается при $\frac{dS}{dN} = 0$, откуда

$$N_0 = \sqrt{\frac{2vA}{B}}. \tag{44}$$

Величина дополнительных расходов S_0, приходящаяся на единицу заказываемого продукта, при оптимальной партии выразится формулой, получающейся подстановкой значения N_0 в выражение (43),

$$S_0 = \sqrt{\frac{2AB}{v}}.$$

Время T_0, через которое должно проходить периодическое обновление запаса (поступление нового заказа), равно $\frac{N_0}{v}$, т. е.

$$T_0 = \sqrt{\frac{2A}{vB}}. \tag{45}$$

Назовем это время *периодом оптимального обновления запаса.* Если скорость v потребления запаса переменна, но меняется относительно медленно, то для расчета объемов оптимальных партий и периодов оптимального обновления можно пользоваться теми же формулами (44) и (45), принимая в качестве соответствующего значения v среднюю скорость $v_{ср}$ расходования запаса в течение времени T_0.

Величины $v_{ср}$ и T_0 находятся последовательным приближением. Пусть

$$v = ce^{kt},$$

где k — достаточно малая величина.

Принимая сначала $v_{ср} = v|_{t=0} = c$, найдем

$$T_0' = \sqrt{\frac{2A}{cB}}; \quad v_1 = v|_{t=T_0} = ve^{k\sqrt{\frac{2A}{cB}}}; \quad v_{ср}' = v\left(1 + \frac{1}{2}e^{k\sqrt{\frac{2A}{cB}}}\right).$$

По этой новой средней скорости найдем новое значение

$$T_0'' = \sqrt{\frac{2A}{v_{ср}'B}}.$$

Если величина $k\sqrt{\frac{2A}{cB}}$ мала по сравнению с единицей, то указанный процесс быстро приведет к нахождению удовлетворительных значений для объема оптимальных партий и периода оптимального обновления запаса. Разумеется, указанные значения будут меняться с течением времени.

Малость величины $k\sqrt{\frac{2A}{cB}}$ означает, что скорость расходования запасов за период его оптимального пополнения меняется мало. При быстрых изменениях скорости расходования запаса выведенные формулы уже неприменимы и должны быть заменены более сложными приемами, основанными, как и прежде, на минимизации величины добавочных расходов, но уже без предположения о постоянстве величины v.

Может быть, например, применен следующий метод последовательной приближенной оптимизации. Вначале, на основании какой-то средней скорости (выбираемой без особых претензий на точность), находят первое приближение для очередной точки заказа T_0. Затем начинают изменять T_0 в направлении, в котором сумма дополнительных расходов уменьшается до тех пор, пока она не станет увеличиваться. Иными словами, применяется общий (градиентный) метод оптимизации, описанный в параграфе 1 гл. II. Таким образом, последовательно, одну за другой определяют очередные точки заказа и соответствующие размеры оптимальных партий.

При использовании подобных общих методов могут применяться более сложные алгоритмы для определения величины дополнительных расходов, связанных с выполнением заказа и хранением заказанного продукта. Например, можно учесть непостоянство расходов на подготовку выполнения заказа, возникающих в том случае, когда при больших размерах заказа нужно вести параллельную подготовку нескольких параллельных технологических линий или менять оснастку в процессе выполнения заказа. Появляется возможность учета нелинейного характера расходов на хранение, если при больших размерах запасов возникает необходимость в оборудовании дополнительных складских помещений. При хранении жидких и газообразных продуктов могут возникнуть и

учитываться ограничения, связанные с наличным объемом буферных емкостей. Такие же ограничения на размер запасов могут быть наложены в результате тех или иных директив вышестоящих органов.

Нелинейность зависимости расходов на хранение от времени хранения имеет место также в том случае, когда при хранении возникают потери. Эти потери могут быть как физической природы (например, уменьшение сахаристости свеклы при ее длительном хранении), так и технико-экономической (например, опасность морального устаревания хранимого продукта в результате возможного изменения технологии, при которой этот продукт сделается ненужным). В стоимость хранения должны быть включены также те дополнительные проценты роста омертвляемых в запасах денежных средств, которые могли бы быть получены при включении этих средств в более активные формы оборота*.

Как было указано в предыдущем параграфе, при формировании заказов желательно указывать размеры потерь при изменении сроков их выполнения. При досрочном выполнении заказа возникают дополнительные расходы на хранение (включая проценты с омертвленных денежных сумм при досрочной оплате заказов). В простейшем (линейном) случае, описанном выше, подобные расчеты не вызывают особых трудностей: прямые дополнительные расходы на хранение в расчете на одну единицу продукта составят $B\Delta t$, где Δt — время, на которое данный заказ опередил назначенный срок. Дополнительные потери от замораживания денежных средств на одну единицу продукта

$$P(1+r)\frac{\Delta t}{\tau},$$

где P — стоимость единицы продукта; r — процент роста средств в обороте; τ — период времени, на который начисляется этот процент.

В случае задержки поставки заказа на время Δt при политике нулевого страхового запаса это время представляет собой чистый простой для рассматриваемого объекта. Потери будут исчисляться суммой зарплат и амортизационными отчислениями за это время плюс потери от сдвига на соответствующий период заказа, который выполняет данный объект. Размер этих последних потерь предполагается определенным у заказчика в момент формирования заказа.

Политика нулевого страхового запаса приводит к тому, что задержки на самом объекте и в системе его материально-технического снабжения оборачиваются значительными потерями и рас-

* Эти потери обычно учитываются лишь при превышении рассчитанных описанными методами оптимальных размеров запасов.

пространением задержек по всем последующим цепочкам технологических связей. Поэтому на практике стремятся планировать снабжение таким образом, чтобы поступление очередной партии заказанного продукта происходило не в момент полного исчерпания имевшегося запаса, а при сохранении некоторой *страховой величины запаса* $N_{\text{стр}}$.

В линейном случае (при постоянной скорости v расходования запаса) обычно величину $N_{\text{стр}}$ выбирают постоянной. При этом, как легко проверить, ни размер N_0 оптимальной партии, ни величина T_0 периода оптимального обновления запаса не меняются, т. е. могут по-прежнему вычисляться по формулам (44) и (45). Дополнительные же расходы увеличатся на величину $\frac{BN_{\text{стр}}}{v}$ и составят

$$S_0 = \sqrt{\frac{2AB}{v}} + \frac{BN_{\text{стр}}}{v}.$$

Средний объем запаса в течение периода обновления T_0

$$\frac{1}{2}(N_{\text{стр}} + N_{\text{стр}} + N_0) = N_{\text{стр}} + \frac{N_0}{2}.$$

Расходы на хранение в расчете на одну единицу заказа

$$\frac{B\left(N_{\text{стр}} + \frac{N_0}{2}\right)T_0}{N_0} = \frac{B\left(N_{\text{стр}} + \frac{N_0}{2}\right)\cdot\frac{N_0}{v}}{N_0} = \frac{BN_{\text{стр}}}{v} + \frac{BN_0}{2v},$$

что лишь *постоянным* слагаемым $\frac{BN_{\text{стр}}}{v}$ отличается от расходов на хранение в схеме с нулевым страховым запасом.

При расчетах объемов страховых запасов также возможны постановки задач на оптимум. Для этой цели необходимы те или иные гипотезы о вероятностях задержек поставок заказанных партий продуктов и о потерях заказчика от этих задержек. Рассмотрим один из простейших примеров. Предположим, что для каждой заказанной партии возможна задержка, независимая от задержек, имевших место при поставках других заказанных партий, причем, вероятность $p(t)$ того, что эта задержка превзойдет время t, выражается экспоненциальной зависимостью $p(t) = e^{-kt}$. Пусть далее потери у поставщика за каждую единицу времени простоя постоянны и равны P. Обозначим время $\frac{N_{\text{стр}}}{v}$, на которое хватит страхового запаса при работе с постоянным расходом v, через $t_{\text{стр}}$. Вероятность того, что задержка выполнения заказа лежит в пределах t, $t + dt$, равна

$$-dp(t) = ke^{-kt}dt.$$

Если $t > t_{стр}$, то потери при такой задержке будут равны $P(t - t_{стр})$. Математическое ожидание этих потерь представится формулой

$$\int_{t_{стр}}^{\infty} P(t - t_{стр})\, ke^{-kt} dt = \frac{P}{k} e^{-kt_{стр}} = \frac{P}{k} e^{\frac{-kN_{стр}}{v}}.$$

Или, в расчете на единицу продукта в заказываемой партии

$$\frac{P}{kN_0} e^{-\frac{kN_{стр}}{v}}.$$

Дополнительные же расходы от хранения страхового запаса в расчете на единицу заказываемого продукта равны, как уже было отмечено выше, $\frac{BN_{стр}}{v}$. Минимизация суммы этих расходов (приравниванием нулю производной по $N_{стр}$) приводит к определению оптимального размера страхового запаса:

$$N_{стр} = \frac{v}{k} \ln\left(\frac{P}{BN_0}\right). \qquad (45а)$$

Разумеется, размер потерь от простоя объекта в единицу времени должен превышать расходы на хранение заказываемой партии продукта в единицу времени (BN_0). Иначе эксплуатация объекта стала бы делом не выгодным, а по формуле (45а) для величины страхового запаса получили бы отрицательное выражение.

При более сложных законах изменения вероятности $p(t)$, характеризующих *степень надежности* поставщика, равно как и при других отступлениях от принятых выше простейших предположений, задача минимизации дополнительных расходов (точнее, их математических ожиданий) за счет выбора оптимальных размеров страховых запасов усложняется и требует применения общих приемов нахождения экстремумов (см. параграф 1 гл. II).

Представляет значительный интерес вопрос о месте, в котором следует иметь страховые запасы. Далеко не всегда выгодно создавать запасы у потребителей. Действительно, нетрудно понять, что в случае, когда число поставщиков гораздо меньше, чем число потребителей (это как раз и имеет чаще всего место), а работа транспорта абсолютно надежна (что бывает далеко не всегда), то значительно выгоднее создавать страховые запасы не у потребителей, а у поставщиков. Само собой разумеется, что при этом необходимо обеспечить соответствующие формы ответственности поставщиков за безусловное соблюдение сроков поставок. Если принимать во внимание ненадежность транспорта, то страховые запасы необхо-

димы и у потребителей, однако их размер будет меньшим, поскольку ненадежность одного транспорта меньше суммарной ненадежности транспорта и поставщика.

Рассмотренные примеры иллюстрируют лишь простейшие случаи, возникающие при решении проблем управления запасами. Обобщения и усовершенствования могут идти в различных направлениях. Одно из обобщений возникает в том случае, когда доставка заказа представляет собой не разовый акт, а равномерное поступление составляющей заказ продукции в течение некоторого времени t, составляющего часть периода T_0 оптимального обновления запаса:

$$t = rT_0 \quad (0 \leqslant r \leqslant 1).$$

При этом скорость пополнения запаса и скорость v его расходования связаны соотношением

$$v = ru.$$

Случай $r = 0$ соответствует рассмотренному ранее случаю разовой поставки всего запаса целиком. Размеры оптимальных партий, величины дополнительных расходов (на единицу продукта) и период оптимального обновления запаса выразятся в общем случае формулами:

$$N_0 = \sqrt{\frac{2vA}{B(1-r)}};$$

$$S_0 = \sqrt{\frac{2AB(1-r)}{v}};$$

$$T_0 = \sqrt{\frac{2A}{Bv(1-r)}}.$$

Другие обобщения связаны с введением случайных колебаний в скорость v расходования запаса, что приводит к необходимости создания страховых запасов.

Большой интерес представляет собой *многопродуктовая* модель, когда запасы создаются по нескольким различным видам продуктов. В этом случае помимо задачи определения оптимальных размеров заказов по различным продуктам возникает задача о взаимном расположении точек заказа. Дело в том, что одновременный заказ различной продукции может привести к необходимости иметь значительный запас денежных средств для оплаты всех этих заказов. Более равномерное распределение точек заказа во времени при сохранении средней суммы оборотных средств, вложенных в запасы, приводит к уменьшению максимальных стоимостей всех имеющихся на данный момент запасов. Иными словами, в многопродуктовой модели обычно стремятся к более равномерному использованию денежных средств, предназначенных для оплаты заказов.

Решение этой задачи может быть получено общими методами поиска экстремума для оценочной функции $S(t_1, t_2, \ldots, t_k)$, где переменные $t_1, t_2, \ldots, t_k$ представляют собой искомые точки заказов соответствующих продуктов. Функция задается алгоритмом, описывающим скорости пополнения и расходования запасов.

В реальных системах управления запасами учитывается также задержка между временем заказа и временем получения заказанного продукта. Если это время обозначить через τ, размер страхового запаса — через $N_{стр}$, а скорость (постоянную) расходования запаса через v, то точка заказа определяется моментом, когда уровень запаса S станет равным

$$S = N_{стр} + v\tau.$$

Этой формулой обычно и пользуются в АСУ, ведущих непрерывный учет состояния запасов для определения точек заказов. Если учет запасов ведется с интервалом $t_{уч}$, то размер запаса, определяющий точку заказа, вычисляется по формуле

$$S = N_{стр} + v\left(\tau + \frac{1}{2} t_{уч}\right).$$

В противоположность описанной системе, в которой определяется оптимальный размер заказа и которую при $v = \text{const}$ можно назвать системой с *постоянным размером заказа*, иногда употребляются и другие системы управления запасами. Хотя они и не минимизируют дополнительных расходов, но могут быть удобными в других отношениях, прежде всего для упрощения вычислений.

Так, в системе с *постоянным уровнем запасов* фиксируется некоторый максимальный уровень запасов и через *равные* промежутки времени проверяется уровень запасов. В результате проверки формируется заказ размером

$$N = S_{max} - S_{тек} - S_{непол},$$

где S_{max} — принятый максимальный объем запаса; $S_{тек}$ — фактический запас на момент проверки; $S_{непол}$ — сделанные, но еще не полученные заказы (при $\tau > t_{уч}$). Максимальный уровень запаса определяется по формуле

$$S_{max} = N_{стр} + v\,(\tau + t_{уч}),$$

где $t_{уч}$ — величина промежутка времени между очередными проверками уровня запасов.

При $v = \text{const}$ средний уровень запасов

$$S_{сред} = N_{стр} + \frac{1}{2} v t_{уч}.$$

Иногда применяется система с *двумя уровнями*. В ней, как и в предыдущей системе, размер заказа определяется формулой

$$N = S_{\max} - S_{\text{тек}} - S_{\text{непол}}.$$

Однако *решение* о производстве заказа принимается не при каждой проверке (через время $t_{\text{уч}} = \text{const}$), а лишь в том случае, когда

$$S_{\text{тек}} + S_{\text{непол}} < S_{\text{заказ}} = N_{\text{стр}} + v\left(\tau + \frac{t_{\text{уч}}}{2}\right),$$

где $S_{\text{заказ}} < S_{\max}$ и представляет как раз второй уровень запаса, по которому производится регулирование и который дал название данной системе.

При переменной скорости v расходования запаса вместо членов $v(\tau + t_{\text{уч}})$, $v\left(\tau + \frac{t_{\text{уч}}}{2}\right)$ применяются *прогнозные оценки* для величины расхода за соответствующие промежутки времени.

Имеются вариации методов управления запасами, использующие *несколько точек заказа*. Например, кроме максимального размера запаса $S_{\max}$ может существовать не одна точка заказа, а две S_1 и S_2 ($S_1 > S_2$). При снижении уровня запаса S ниже S_1 (но выше S_2!) при очередной проверке дается заказ размером $N = S_{\max} - S_{\text{тек}} - S_{\text{непол}}$, а при $S < S_2$ размер заказа определяется без учета неполученных заказов: $N = S_{\max} - S_{\text{тек}}$. Такая политика применяется в том случае, когда между очередными проверками возможны быстрые изменения скорости расхода запаса, так что ожидание неполученных заказов может привести к возникновению дефицита.

Применение теории управления запасами к оперативно-календарному планированию основано на расчленении технологических маршрутов на отдельные участки, выступающие по отношению друг к другу как поставщики и потребители. Для возможности отделения одного участка от другого необходимо наличие буферных емкостей (складов, кладовых, резервуаров и т. п.) для хранения запасов продуктов, выпускаемых одним участком и потребляемых на другом участке. При отсутствии таких буферов отдельные участки соединяются в один участок, называемый поточной линией, на котором производительности отдельных составляющих его подучастков должны быть строго согласованы между собой. Примером поточной линии является обычный конвейер, хотя, разумеется, поточные линии могут строиться и без него.

Для каждого участка формируется поток заказов. В заключительных участках (например, сборочных цехах) такие задания формируются из заданий для всего рассматриваемого экономического объекта для других участков — из заказов участков, расположенных

впереди по соответствующим технологическим маршрутам. Результатом расчета календарного плана должны быть сменные задания на выполнение необходимых производственных, подготовительных и ремонтных операций, а также формирование и привязка к точным календарным срокам заказов другим участкам и заданий на материально-техническое снабжение всего объекта.

Для иллюстрации возникающих при оперативно-календарном планировании задач рассмотрим пример. Рассчитать календарный план работы производственного участка, работающего в две смены по 8 *час* с двумя выходными днями в неделю (на июнь), выходные дни приходятся на числа 1, 7, 8, 14, 15, 21, 22, 28, 29. Участок выпускает два вида деталей I и II. На изготовление детали типа I тратится 10 *мин*, а на изготовление детали типа II — 50 *мин*. Подготовительные операции (переналадка оборудования) для выпуска партии деталей типа I занимают 1 *час*, а деталей типа II — 2 *час*.

Производственное задание участку формируется в следующем виде: 9 июня к началу первой смены поставить партию из 400 шт. деталей типа I, 12 июня к началу второй смены — 60 шт. деталей типа II, 18 июня к началу второй смены — 400 шт. деталей типа I, 26 июня к началу второй смены 120 шт. деталей типа II и 1 июля к началу первой смены — 200 шт. деталей типа I. Заданные сроки поставок будем называть контрольными сроками. Предполагается, что описанное задание сформировано в результате предварительного решения двух типов оптимизационных задач. Во-первых, при объемно-календарных расчетах была произведена оптимизация распределения месячного задания между отдельными участками. Во-вторых, при определении точек заказа и размеров партий использовались описанные выше методы теории управления запасами. К сожалению, на практике при согласовании работы большого числа участков не удается полностью выдержать оптимальные размеры партий, а лишь приближаться к ним, как желательному ориентиру. Это обстоятельство нашло место и в приведенном выше задании.

Отправным пунктом календарных расчетов является общее количество часоресурсов (или минуто-ресурсов), которыми мы располагаем до моментов, когда необходимо осуществлять поставку заказанных партий продукции. В соответствии с приведенным заданием имеется пять таких моментов (9, 12, 18, 26 июня и 1 июля). Поскольку мы не расчленяем участок на более мелкие составные части*, то все его ресурсы могут рассматриваться как единый ресурс и исчисляться просто в минутах работы всего участка. Нетрудно подсчитать, что до первого момента располагаем 10 сменами, или 4800 *мин*, между первым и вторым моментами — 7 сменами (3360 *мин*) и т. д. Все данные расчета приведены в табл. 15.

Определяем общий резерв времени, т. е. разность между имеющимся и необходимым для выполнения всех заданий временем: $20160 - 19000 = 1160$ *мин* $=$ $= 19$ *час* 20 *мин*. Допустим, что в предстоящем месяце необходимо осуществить плановый текущий ремонт оборудования, что должно занять по нормам 12 *час*. Остальное время (7 *час* 20 *мин*) может быть использовано на переналадки и создание оперативного резерва рассматриваемого участка. Пусть к началу планового периода (т. е. 2 июня) участок был настроен на выпуск детали I. Тогда для подготовки выпуска всех пяти заказанных партий потребуется четыре переналадки общей продолжительностью $2 + 1 + 2 + 1 = 6$ *час*. Оперативный резерв времени составит лишь 1 *час* 20 *мин*, что означает очень плотную загрузку участка.

В практике планирования может встретиться и такой случай, когда этот резерв отрицателен. Тогда уменьшают общее число переналадок, объединяя не-

* Весь участок может сводиться к одному станку и одному рабочему месту.

Таблица 15

Контрольные сроки	Тип деталей	Размер партии, шт.	Время для выполнения задания, *мин*	
			необходимое	имеющееся
9 июня начало 1-й смены	I	400	4000	4800
12 июня начало 2-й смены	II	60	3000	3360
18 июня начало 2-й смены	I	400	4000	3840
26 июня начало 2-й смены	II	120	6000	5760
1 июля начало 1-й смены	I	200	2000	2400
Итого			19000 *мин*	20160 *мин* = =336 *час*

сколько партий, или плановые ремонты осуществляют не в рабочие смены и т. п. Разумеется, объединение заказов означает отступление от оптимальных размеров партий и, как следствие этого, дополнительные финансовые потери. Может возникнуть необходимость и в сдвиге сроков поставок, что, как уже отмечалось выше, должно производиться лишь на основании расчета объемов потерь, возникающих при этом у заказчика.

При наличии резервов времени возникает вопрос об их распределении между различными подынтервалами планового периода. Для этой цели составляется баланс времени (и ресурсов) на весь плановый период нарастающим итогом с учетом необходимых переналадок (табл. 16).

Таблица 16

Контрольные сроки	Номер партии	Время переналадки, *мин*	Рабочее время, *мин*	Время от начала планового периода, *мин*	
				расход	запас
9 июня	1	0	4000	4000	4800
12 июня	2	120	3000	7120	8160
18 июня	3	60	4000	11180	12000
26 июня	4	120	6000	17300	17760
1 июля	5	60	2000	19360	20160

Поскольку необходимый суммарный расход времени на 26 июня лишь на 360 *мин* (6 час) меньше наличного запаса времени, то 12-часовой плановый ремонт может быть отнесен лишь на последний интервал (между 26 июня и 1 июля). Остающийся резерв времени (1 *час* 20 *мин*) настолько мал, что вопрос о его распределении по плановому периоду не имеет существенного значения. При большей величине этого резерва обычно стремятся распределить его равномерно

вдоль всего планового периода, чтобы обеспечить равномерную загрузку участка и возможность маневрирования при возникновении непредвиденных обстоятельств. Наличие резервов позволяет производить загрузку оборудования в соответствии с экономическими критериями, сохранять оптимальные размеры партий и минимально необходимые размеры запасов.

Когда имеется значительная вероятность появления новых заданий, более целесообразно придерживаться политики сохранения резервов до момента появления этих новых заданий с тем, чтобы облегчить возможность быстрой коррекции плана.

В рассматриваемом случае естественно принять стратегию, основанную на заданной последовательности заданий, поскольку сделанная в табл. 16 раскладка времени ни к одному из контрольных сроков не обнаружила дефицита. Плановый ремонт, как уже было отмечено выше, относится на конец периода. Сменные задания можно планировать непосредственно с учетом данных табл. 16:*

8 смен 2, 3, 4, 5 июня — выпуск 384 деталей типа I (по 48 деталей в смену)
1-я смена 6 июня: $0 — 2^{40}$ — выпуск 16 деталей типа I (постановка 400 деталей типа I)

$2^{40} — 4^{40}$ — переналадка

$4^{40} — 8^{00}$ — выпуск 4 деталей типа II

2-я смена 6 июня — выпуск 48 деталей типа II
4 смены 9 и 10 июня
1-я смена 11 июня: $0 — 6^{40}$ — выпуск 8 деталей типа II (постановка 60 деталей типа II)

$6^{40} — 7^{40}$ — переналадка

$7^{40} — 8^{00}$ — резерв

2-я смена 11 июня — выпуск 48 деталей типа I
6 смен 12, 13, 16 июня — выпуск 288 деталей типа I
1-я смена 16 июня — выпуск 48 деталей типа I
2-я смена 16 июня: $0 — 2^{40}$ — выпуск 16 деталей типа I (постановка 400 деталей типа I)

$2^{40} — 4^{40}$ — переналадка

$4^{40} — 8^{00}$ — выпуск 4 деталей типа II

10 смен 17, 18, 19, 20, 23 июня — выпуск 96 деталей типа II
2 смены 24 июня — выпуск 19 деталей типа II

10 *мин* — резерв

1-я смена 25 июня: $0 — 0^{50}$ — выпуск одной детали типа II (постановка 120 деталей типа II)

$0^{50} — 8^{00}$ — ремонт

2-я смена 26 июня: $0 — 4^{50}$ — ремонт

$4^{50} — 5^{50}$ — переналадка

$5^{50} — 8^{00}$ — выпуск 13 деталей типа I

---

* Время исчисляется от начала каждой смены.

4 смены 27 и 30 июня — выпуск 144 деталей типа I
2-я смена 30 июня: $0 — 7^{10}$ — выпуск 43 деталей типа I (постановка деталей типа I)
$7^{10} — 8^{00}$ — резерв.

После составления сменных заданий проводится расчет материально-технического обеспечения участка. Описанными выше методами определяются объемы страховых запасов, размеры партий и точки заказа для заготовок, материалов, инструмента и т. п. Аналогичным образом заказываются вагоны и другие транспортные средства для вывозки готовой продукции. Определяется ожидаемое время поступления платежей за поставленную продукцию и на каждый день принятого планового периода рассчитывается величина сумм по различным статьям, которая будет находиться на текущем счете предприятия.

Составленный план непрерывно пополняется и уточняется. Для этой цели он постоянно держится на машинных носителях (лучше всего на дисках) в состоянии «боевой готовности». Для облегчения работы с планом создается специальное матобеспечение (специализированная операционная система), позволяющая пользователю быстро вносить необходимые изменения, сохраняя полную сбалансированность плана. Например, при изменении срока выполнения задания должны автоматически изменяться сроки поставок материалов, заготовок, инструмента и т. п. В систему должны быть включены специальные программы-макрооператоры, позволяющие быстро подсчитывать различные временные и технико-экономические характеристики плана.

Иными словами, нужно иметь возможность непрерывной работы с планом и принятия решений по его улучшению на основе машинного моделирования возникающих ситуаций и предлагаемых путей устранения возникающих трудностей. Динамичность плана должна проявляться, в частности, в том, что в нем постоянно корректируются запасы времени до соответствующих контрольных точек. Например, в табл. 15 на начало рабочего дня 9 июня в первой строке последнего столбца указано 4800 рабочих минут до первой контрольной точки. На начало следующего рабочего дня эта цифра должна быть заменена на 3840 *мин* и т. д.

Существенную роль играет связь системы планово-календарных расчетов с другими системами. Например, если размер страховых запасов по заготовкам деталей типа I был определен в 100 шт., а из системы технической подготовки производства поступает информация о том, что к концу месяца ожидается изменение технологии, устраняющей необходимость использования детали типа I, то система должна обеспечить постепенную выработку накопленного страхового запаса к моменту выхода приказа об изменении технологии.

Описанные задачи и приемы их решения являются общими для всех производств дискретного характера. Особенность производств, имеющих дело со сложными изделиями с длительными циклами изготовления (корабли, прокатные станы и т. п.), состоит в том, что основу оперативно-календарного планирования составляют сетевые графики (хотя и здесь управление заказами на сравнительно большие партии деталей могут строиться на описанных в этом параграфе принципах).

Свои особенности имеет также *массовое производство* (производство холодильников, телевизоров и т. п.). Одна из важных задач, которую приходится решать в этом случае — это задача нахождения *оптимального ритма* производства. Смысл ее состоит в том, чтобы через промежуток времени T, называемый циклом или ритмом работы объекта, повторялись все сменные задания, которые имели место на предшествующем цикле. Обычно стремятся найти минимальный цикл $T_{\text{мин}}$, выражаемый целым числом смен или дней.

Для примера рассмотрим работу заготовительного участка, обеспечивающего сборочный цех деталями n различных типов. Пусть p_i — время, затрачиваемое на изготовление одной детали i-го типа; τ_i —время на подготовку выпуска деталей i-го типа, а S_i — скорость расходования этих деталей на сборке ($i = 1, 2, \ldots, n$). Чтобы обеспечить сборку, заготовительный участок за время T должен произвести S_1T деталей первого типа; S_2T — деталей второго типа и т. п. Время, необходимое для изготовления S_iT изделий,

$$\tau_i + \frac{S_iT}{p_i},$$

а сумма всех этих времен не должна превосходить T:

$$T \geqslant \sum_{i=1}^{n} \left(\tau_i + \frac{S_iT}{p_i}\right),$$

откуда

$$T \geqslant \frac{\sum_{i=1}^{n} \tau_i}{1 - \sum_{i=1}^{n} \frac{S_i}{p_i}}. \qquad (46)$$

Минимальное (целое) значение T, удовлетворяющее неравенству (46), и представляет собой искомый минимальный ритм работы рассматриваемого объекта.

При $\tau_1 = \tau_2 = \tau_3 = 0{,}5$; $S_1 = 10$; $S_2 = 15$; $S_3 = 20$; $p_1 = 40$; $p_2 = 30$; $p_3 = 100$:

$$T \geqslant \frac{1{,}5}{1 - \left(\frac{1}{4} + \frac{1}{2} + \frac{2}{10}\right)} = \frac{1{,}5}{1 - 0{,}95} = 30.$$

Это означает, что возможен минимальный ритм работы с периодом в 30 смен.

6. Задачи управления в макроэкономике

Как и во всех случаях управления большими системами, в макроэкономических задачах необходимо прежде всего четко различать цели и средства их достижения. Естественно, что задачи управления большими экономическими системами вообще и задачи управления макроэкономикой в особенности в большой степени зависят от социального строя. В капиталистическом обществе цели развития экономики могут формулироваться в чисто экономических категориях (например, достижения максимальной прибыли). Одной из отличительных особенностей социалистического общества, ставящего своей целью максимальное удовлетворение материальных и духовных потребностей своих членов, является то, что цели формируются в основном за пределами собственно экономики, а экономика рассматривается прежде всего как средство достижения поставленных целей.

Цели развития социалистической экономики могут быть сформулированы в терминах *конечного продукта*. Под конечным продуктом мы понимаем все те производимые обществом товары и услуги, которые предназначаются для внеэкономического или, иными словами,— *непроизводственного потребления*. Можно выделить следующие составные части конечного продукта: 1) личное потребление граждан (предметы питания, одежда, жилье, мебель, радиоприемники, газеты, книги и т. п.); 2) общественное потребление, т. е. различные услуги, которые общество представляет своим членам (общее образование, детские учреждения, здравоохранение, общественный транспорт, розничная торговля, общественное питание, зрелищные предприятия, телевидение, радио, связь, коммунально-бытовое обслуживание и т. п.); 3) общегосударственные внеэкономические расходы (оборона, охрана общественного порядка, помощь другим странам и народам, фундаментальная наука, охрана природы и другие расходы, связанные с различными общественными функциями внеэкономического характера, например, выборные кампании, содержание посольств и т. п.).

В предложенном перечне проведено четкое различие между *общим образованием*, которое рассматривается как цель, и *специальным образованием*, которое должно рассматриваться как *средство* достижения цели. Аналогичное различие между фундаментальной наукой, как целью, и прикладной наукой, представляющей собой средство достижения поставленных целей; помощь другим странам можно рассматривать как цель, а внешнюю торговлю — как средство; общественный транспорт и связи — как цель, когда они обслуживают внеэкономические потребности, а перевозки грузов между промышленными предприятиями — как средства.

Некоторых дополнительных разъяснений требует, по-видимому, также отнесение к целям розничной торговли, которую, на первый взгляд, естественно рассматривать лишь как средство распределения предметов личного потребления. Дело, однако, заключается в том, что целью розничной торговли должно служить не простое выполнение функции распределения, а создание определенных удобств для населения. Если рассматривать розничную торговлю лишь как средство, то она (как и всякое средство) должна попасть под процедуру минимизации расходов, ибо смысл управления как раз и состоит в том, чтобы достигать поставленных целей минимальными средствами.

Целью же развития общества должно быть создание не только *материального*, но и *психологического* комфорта для своих членов. Поэтому при формировании и сравнительной оценке ставящихся целей могут и должны действовать прежде всего внеэкономические механизмы: цель не обязательно должна быть «экономически выгодной», в то время как для средств это требование является необходимым. Следует четко понимать при этом, что понятия психологического и материального комфорта тесно связаны между собой. Например, увеличивая психологический комфорт покупателей в розничной торговле мы неизбежно должны увеличить расходы на нее, уменьшить соответственно расходы в других местах (например, в жилищном строительстве), и снизить тем самым темпы роста материального комфорта.

Перечисленный выше примерный перечень типов целей не является, разумеется, догмой.

В отдельные периоды общественного развития в список целей может попасть, например, производство стали, не являющейся конечным продуктом. Обычно к такой перестановке средств в цели приходится прибегать в том случае, когда точные методы определения наилучших средств достижения действительно *конечных* целей не разработаны или чрезмерно трудоемки, а качественно полезность тех или иных средств для достижения конечных целей ясна. При таких условиях естественно переводить подобные от-

дельные средства в цели, которые, в отличие от конечных целей, мы будем называть *промежуточными*.

Следует также подчеркнуть, что термин «конечные» применительно к целям не означает, что они ставятся на вечные времена раз и навсегда. Речь идет о непрерывно действующей системе постановки, обновления и уточнения целей на достаточно длительный период (порядка 15—20 лет). При этом, по мере движения вперед, ранее выставленные цели конкретизируются в долгосрочных и краткосрочных планах и достигаются, а новые цели все время возникают (первоначально в менее четкой, а затем во все более уточняющейся форме) в указанной временной перспективе, т. е. через 15—20 лет от *настоящего* момента. Как уже отмечалось в параграфе 4, величина этого периода определяется возможностью более или менее точного научного предвидения будущего и длительностью реализации крупных программ национального и межнационального масштаба. При этом остается возможность появления (при неожиданных научных открытиях, резких поворотах процессов общественного развития и других непредвиденных ситуациях) новых целей, отдаленных от настоящего момента более короткими промежутками времени.

Непрерывно действующая система постановки целей может быть построена на использовании идей системного анализа и состоять из следующих основных элементов. Первый элемент системы — это так называемый *банк идей*, объединяющий широкий круг ученых, конструкторов и практических работников. Задачей участников этого банка является непрерывное его пополнение различными идеями о создании новых видов конечного продукта или коренном усовершенствовании старых. В отличие от существующих систем регистрации изобретений и открытий в этом банке регистрируются не только технические идеи (например, идея создания стереотелевизора), но и различные идеи по созданию новых форм услуг, дальнейшему развитию форм общественного самоуправления и социалистической демократии и т. д. и т. п.

При этом желательно иметь гибкую (учитывающую изменение областей интересов экспертов) систему распределения вопросов, за которыми следят участники банка, для того, чтобы ни одно новое открытие не осталось бы непроанализированным с точки зрения возможности создания на его основе нового конечного продукта. В идеале речь идет о системе, которая могла бы *сразу* после любого фундаментального открытия (типа открытия электромагнитных волн) сформулировать различные идеи его применения для создания нового типа конечных продуктов (радиоприемник, телевизор и т. п.), хотя, быть может, никаких реальных технических путей реализации этих идей еще и нет.

Второй элемент системы постановки целей — *системный анализ* и *системная группировка* целей. Речь идет прежде всего о посто-

яннодействующей системе коллективной экспертизы и прогнозирования сроков реализации идей, имеющихся в банке. Если такую экспертизу вести методами, изложенными в параграфе 7 гл. II, то попутно с прогнозом сроков осуществляется прогнозирование *возможных путей* достижения этих целей. Задача *системной группировки* состоит в том, что отдельные товары и услуги, которые имеют большую вероятность появления к концу периода перспективного планирования и общее функциональное назначение, комбинируются в *группы* или *системы целей*. Примером такой групповой цели может служить жилищно-мебельный комплекс, включающий в себя много вариантов квартир и их оборудования или комплекс «город», описывающий варианты различных решений комплексного развития различных городов.

Для эффективного функционирования такой системы необходимо, чтобы она включала в себя все проектно-конструкторские организации соответствующего профиля и центральный системно-прогнозный аппарат, объединяющий их воедино.

Третьим элементом системы является постоянно действующая система сравнительной оценки построенных групп целей или отдельных составляющих их частей. Оценка целей (ставящихся перед экономикой) может быть произведена на основе моделирования с использованием целей более высокого уровня. Для оценок же целей в области личного и общественного потребления целесообразно опираться на постоянно действующую систему опросов общественного мнения. Задача такой системы состоит не только в определении того, какие комбинации конечного продукта будут являться наиболее предпочтительными для различных групп населения, а и в том, чтобы примерно определить уровень цен на различные составные части конечного продукта, при котором реальное потребление (при прогнозируемом размере зарплаты) действительно соответствовало бы наиболее предпочтительным комбинациям.

Задача эта чрезвычайно трудна для точного решения. Однако в данном случае речь идет об ориентировочных оценках, которые могут быть получены на основании комбинирования данных в результате опросов общественного мнения с методами системного анализа и коллективных экспертных оценок. Данные о желательных ценах отдельных составных частей конечного продукта должны передаваться во вторую из описанных подсистем для уточнения сроков и путей достижения целей *с учетом найденного (желательного) уровня цен*.

После получения этих данных центральная группа системного анализа (во второй подсистеме) формирует *предварительную* систему целей с учетом возможных альтернатив (например, ориентировка на квартиры с встроенной мебелью или увеличение произ-

водства обычной мебели). Эти альтернативы составляются для целей последующего отбора или на основании внеэкономических соображений (на этапе утверждения системы целей) или в результате экономического анализа (на этапе перспективного планирования).

Утверждение системы целей для развития экономики является прерогативой высших органов *политического управления*. Что же касается перспективного планирования, осуществляющего отбор наиболее эффективных путей и средств для достижения поставленных целей, то оно организуется высшим плановым органом (Госпланом). Его результаты используются (на правительственном уровне) для уточнения системы целей, после чего вновь уточняется перспективный план и т. д.

Система постановки и уточнения целей и система перспективного планирования должна быть непрерывно действующей. Корректировка системы целей должна производиться по мере надобности (по-видимому, не реже, чем один раз в год, а быть может, и значительно чаще). Как система целей, так и перспективный план должен постоянно храниться на машинных носителях в *динамическом* (т. е. готовом к любым корректировкам) виде.

Из системы постановки целей в систему перспективного планирования цели поступают в виде относительных долей отдельных составных частей конечного продукта в общей массе конечного продукта (предназначенного для непроизводственного потребления). Это означает, в частности, что на правительственном уровне принимается решение о распределении расходов на отдельные виды конечного продукта, перечисленные выше. При определении желательных пропорций расходов могут применяться дополнительные средства в виде непрерывно действующих систем прогнозирования процессов расстановки сил на мировой арене или систем прогнозирования различных разделов фундаментальных наук. Системный анализ на современном уровне его развития может обеспечить лишь получение известных материалов, полезных при принятии решения, но не однозначное указание самого решения.

На правительственном уровне принимаются также решения о политике в отношении пропорции деления национального дохода между потреблением и накоплением. Определение конечного продукта, принятое выше, пригодно для планирования на сравнительно длительные сроки. При краткосрочном планировании многие задачи и, прежде всего, задачи капитального строительства и прироста основных фондов в народном хозяйстве переходят в самостоятельные цели. Этот прирост обеспечивается направлением в капитальное строительство и изготовление оборудования части совокупного продукта народного хозяйства, называемого *фондом накопления*. Вместе с *фондом потребления* (непроизводственного) фонд накопле-

ния за тот или иной промежуток времени составляет *национальный доход* (за это время).

Национальный доход при краткосрочном планировании может рассматриваться как конечный продукт, попадая тем самым в разряд целей. При перспективном же планировании накопление представляет собой тот инструмент, то средство, с помощью которого достигаются цели, поставленные в области потребления. Нетрудно понять, что увеличение доли накопления в данный момент (при прочих равных условиях) увеличивает скорость развития экономики и способствует ускорению достижения общественных целей в будущем. Вместе с тем в настоящем увеличение доли накопления означает неизбежное уменьшение доли потребления.

Поэтому возникает понятие политики деления национального дохода в течение рассматриваемого планового периода (перспективного планирования). Смысл этой политики состоит в задании закона изменения доли накопления β в национальном доходе в зависимости от времени:

$$\beta = \beta(t) \qquad (0 \leqslant t \leqslant T),$$

где T — рассматриваемый плановый период.

Выбор политики деления национального дохода является прерогативой высших органов политического управления. Системный анализ и математические методы могут дать лишь материал для выбора правильного решения, но не определить однозначно само решение. Материал, о котором идет речь, представляет собой результат опроса общественного мнения, прогноз развития международных отношений и вероятности угрозы военного нападения и др. Важнейшим инструментом, позволяющим принять правильное решение, являются также результаты расчетов *различных вариантов развития экономики* при различных вариантах политики деления национального дохода.

Аналогичные проблемы возникают при определении политики изменения во времени пропорций деления фонда потребления. Если относительная доля i-й составной части фонда потребления есть α_i, то речь идет о выборе функций

$$\alpha_i(t) \quad (0 \leqslant t \leqslant T, \quad i = 1, 2, \ldots, m).$$

Выбор функций $\alpha_i(t)$ является, однако, в большей мере результатом многовариантных плановых расчетов, чем их предпосылкой. Предплановое задание состоит из функции $\beta(t)(0 \leqslant t \leqslant T)$ (или указаний о ее характере) и системы целей $\alpha_i = \alpha_i(T)$ $(i = 1, 2, \ldots, m)$ на конец планового периода. Возможны также

указания о желательном характере функций $\alpha_i(t)$*. Сам период T определяется лишь ориентировочно, точная его величина, и окончательный вид функций определяются в результате плановых расчетов для каждого из возможных вариантов плана и составляют предмет последующего выбора на правительственном уровне.

Важным элементом предпланового задания является также политика изменения величины рабочей недели, пенсионного возраста и размеров пенсий, что необходимо для последующих расчетов трудовых ресурсов. Решения о выборе такой политики также являются прерогативой высших правительственных органов. На этом же уровне должны приниматься решения о политике в области создания государственных запасов.

Получив предплановое задание, плановые органы совместно с отраслевыми автоматизированными системами и системой управления научно-техническим прогрессом осуществляют многовариантные расчеты перспективных планов, используя *макроэкономические модели*.

Опишем одну из возможных моделей, в основу которой положена известная модель Леонтьева. Основой модели является разложение *совокупного продукта*, производимого экономикой на составные части $x_1, x_2, \ldots, x_n$. Эти продукты потребляются как вне экономики (непроизводственное потребление), так и в рамках самой экономики (производственное потребление), например потребление стали при производстве тракторов. Задача модели — найти соотношения, связывающие производство и потребление. Эти соотношения называют также *уравнениями материальных балансов*.

Для i-й составной части совокупного продукта обозначается величина его производства за какой-то относительно небольшой промежуток времени τ. Ее можно трактовать так же, как скорость производства i-го продукта. Обозначим φ_i — непроизводственное потребление i-го продукта; z_i — суммарный объем продаж i-го продукта на внешнем рынке (экспорт); y_i — суммарный объем его покупок (импорт); s_i — прирост государственных запасов i-го продукта (он может быть и отрицательным при деблокировании резерва). Все перечисленные величины отнесены к тому же самому промежутку времени τ.

Производственное потребление i-го продукта в сфере *текущего производства* выражается суммой $\sum_{j=1}^{n} a_{ij}x_j$, где a_{ij} представляет собой *технологический коэффициент*, называемый обычно коэффициентом прямых затрат. Он равняется величине затрат i-го

* Например, указание о том, что полный переход на всеобщее среднее образование желательно завершить в первой половине планового периода.

продукта при производстве одной единицы j-го продукта. Производственное потребление i-го продукта для наращивания производственных мощностей (т. е. создания новых основных и оборотных фондов) Леонтьев выражает суммой

$$\sum_{j=1}^{n} b_{ij} \frac{dx_j}{dt},$$

где b_{ij} — технологический коэффициент, показывающий величину необходимых затрат i-го продукта для создания новых мощностей, способных увеличить производство j-го продукта на одну единицу. Недостатком такого выражения является то, что оно не учитывает возможности блокирования и деблокирования производственных мощностей*, поскольку использует в качестве второго множителя $\frac{dx_j}{dt}$, т. е. скорость *увеличения производства* j-го продукта, вместо скорости $\frac{dM_j}{dt}$ *прироста мощностей* M_j, необходимых для такого увеличения.

При принятии предположения о *немедленной и полной* загрузке вновь создаваемых мощностей сделанное возражение снимается и уравнение материального баланса по i-му продукту выражается в виде:

$$x_i + y_i = \sum_{j=1}^{n} a_{ij}x_j + \sum_{j=1}^{n} b_{ij} \frac{dx_j}{dt} + z_i + s_i + \varphi_i. \qquad (47)$$

Совокупность таких уравнений для всех продуктов ($i = 1, 2, \ldots, n$) составляет то, что обычно называют *динамической моделью Леонтьева*. В левых частях уравнений стоят приходы, а в правых — расходы соответствующих продуктов. К этой модели добавляют обычно *уравнение баланса внешней торговли*:

$$\sum_i (p_i' z_i - p_i'' y_i) = \Delta M, \qquad (48)$$

где p_i' и p_i'' — цены одной единицы i-го продукта при продажах и покупках на мировом рынке; ΔM — прирост национальных валютных фондов (за время τ).

Решения о политике прироста (положительного или отрицательного) валютных фондов и соответствующие решения о политике наращивания или расходования государственных резервов также

---

* Отнесение подобных блокированных мощностей в разряд государственных резервов является слишком искусственным приемом и сильно усложняет анализ модели.

принимаются на правительственном уровне с использованием материалов соответствующих плановых прикидок.

Для исследования и решения системы (47) при предположении о постоянстве технологических коэффициентов a_{ij} и b_{ij} могут быть использованы методы, изложенные в параграфе 6 предыдущей главы. Ценность столь абстрактной модели, какой является исходная модель Леонтьева, для решения *практических задач* перспективного планирования невелика. Помимо уже отмеченного недостатка, эта модель имеет еще целый ряд существенных недостатков. Во-первых, это отсутствие в модели в явном виде наличных трудовых и материальных ресурсов, а значит, и соответствующих ограничений по ресурсам: достаточно заметить, что исходная модель Леонтьева строится при неявном предположении о неограниченности трудовых ресурсов. Второй (и, по-видимому, главный) недостаток модели — отсутствие возможности целенаправленного варьирования технологических коэффициентов для выбора оптимальных наборов технологий, обеспечивающих минимизацию сроков достижения поставленных целей и максимальные темпы роста производительности труда. Этот недостаток лишает модель ее основной ценности: она перестает быть инструментом активного управления и лишь регистрирует сложившиеся тенденции развития*.

Кроме того, даже если развитие действительно происходит в условиях постоянства технологических коэффициентов и полной загрузки вновь вводимых мощностей, то это обстоятельство может быть фактически использовано лишь в том случае, когда $\frac{dx_j}{dt} \geqslant 0$. При отрицательных $\frac{dx_j}{dt}$ использование тех же самых коэффициентов b_{ij} теряет всякий смысл, ибо сокращение мощностей требует совершенно иной структуры затрат, чем их наращивание.

Учитывая все указанные недостатки, опишем другую макроэкономическую модель, использующую в явном виде ресурсы и множество варьируемых технологий. Обозначим R_1, R_2, ..., R_l величины различных материальных ресурсов, которыми располагает экономика в целом, а через P_1, P_2, ..., P_l — величины трудовых ресурсов по различным категориям труда (различным специальностям). Обозначение P (без индекса) оставим для суммарной величины всех трудовых ресурсов (независимо от специальностей).

Имея постоянно действующую систему *демографического* прогнозирования (т. е. прогнозирование количества населения, его возрастного состава и т. п.) и опираясь на заданную политику изменения продолжительности рабочей недели и величины пенсионного возраста, плановые органы могут иметь прогноз величины $P =$

---

* Первоначально предложенная оригинальная модель Леонтьева дает лишь один неизменный вариант развития экономики.

$= P(t)$ на всю величину планового периода T. Важность специального рассмотрения величины P определяется тем, что этот ресурс при перспективном планировании является основным ограничивающим ресурсом. Иными словами, если при достаточно длинном периоде T можно построить необходимое число новых заводов, изготовить нужное количество оборудования и даже успеть изменить как угодно в пределах общей численности соотношение количеств имеющихся работников разных специальностей, для общего количества P наличных трудовых ресурсов такого рода соображения уже не применимы. Поэтому даже в перспективе экономику придется развивать с учетом этого ограничения.

При перспективном планировании могут существовать и другие типы ограничивающих ресурсов, например, земля и другие природные ресурсы. По всем ограничивающим ресурсам необходимо иметь систему постоянного прогнозирования их ожидаемой величины на весь плановый период. При краткосрочных планах в разряд ограничивающих попадает все большее число ресурсов, а именно — все те ресурсы, которые не можем существенно изменить за планируемый период. В то же время даже для таких ресурсов, как, например, запасы минерального сырья, возникающие ограничения могут быть полностью или частично обойдены рациональным использованием внешней торговли. Этот рычаг, однако, совершенно не применим для обхода ограничений в трудовых ресурсах. Именно поэтому ограничение в трудовых ресурсах является главным и определяющим в перспективном планировании, обусловливая объективную необходимость строить перспективные планы из расчета максимально возможного роста производительности труда.

Следующей существенной операцией в рассматриваемой модели есть *разделение ресурсов* на ресурсы, обеспечивающие текущее производство, и на ресурсы, обеспечивающие его развитие (т. е. приросты мощностей). Это разделение должно исходить из заданной политики деления национального дохода и в текущий момент времени оно задано. Разделение может быть выполнено различными способами при разных пропорциях отдельных ресурсов. Может оказаться например, что для работы на накопление (развитие экономики) расходуется лишь 0,1 всех трудовых ресурсов и 0,5 всех основных фондов. Важно лишь, чтобы с помощью этих ресурсов создавалась заданная часть (скажем, 0,3) национального дохода. Система коэффициентов, задающих пропорции использования ресурсов, является варьируемыми параметрами, с помощью которых можно осуществлять управление, создавая различные варианты планов. Способы объективного подсчета стоимостей создаваемых продуктов были объяснены ранее (см. параграф 4). Ряд пунктов макроэкономического ценообразования будет пояснен ниже.

Возможно и упрощение задачи, основанное на том, что в предплановом задании вместо функции $\beta(t)$ задается функция $\gamma(t)$, определяющая относительную долю трудовых ресурсов, работающих на накопление.

Производя тем или иным способом разделение каждого ресурса R_i (или P_i) на две части $R_i^{\text{тек}}$ и $R_i^{\text{разв}}$, можно составить две различные системы уравнений, которые называются соответственно *статической* и *динамической* частями модели. В отличие от исходной модели Леонтьева будем рассматривать несколько различных технологий для производства одного и того же продукта, обозначая через x_{iq_i} количество i-го продукта, произведенное за данное время τ, с помощью q_i-й технологии. Например, x_{11} может обозначать количество электроэнергии, произведенное обычными тепловыми электростанциями, x_{12} — атомными, а x_{13} — гидроэлектростанциями. Соответственно коэффициенты прямых затрат (технологические коэффициенты) a_{ij} и b_{ij} получат третий индекс q_j, указывающий, с помощью какой технологии получается рассматриваемая единица j-го продукта или единица прироста j-го ресурса.

Верхними индексами обозначим те части соответствующих величин, которые используются для нужд текущего производства, потребления экспорта и создания резервов и, соответственно, для нужд развития производства. При сделанных предположениях статическая часть модели выражается следующей системой уравнений и неравенств:

$$\sum_{q_i} x_{iq_i}^{\text{тек}} + u_i^{\text{тек}} y_i^{\text{тек}} = \sum_j \sum_{q_j} a_{ijq_j} x_{jq_j}^{\text{тек}} + z_i + v_i + \varphi_i \quad i = (1, 2, \ldots, n); \tag{49}$$

$$P^{\text{тек}} = \sum_i \sum_{q_i} A_{iq_i} x_{iq_i}; \tag{50}$$

$$P^{\text{тек}} = \sum_l \sum_{q_l} A_{jlq_l} x_{lq_l}^{\text{тек}} \quad (j = 1, 2, \ldots, l); \tag{51}$$

$$R_l^{\text{тек}} \geqslant \sum_i \sum_{q_i} C_{jlq_i} x_{iq_i}, \tag{52}$$

где $y_i^{\text{тек}}$ — часть импорта, используемая для текущего производства и потребления; $u_i^{\text{тек}}$ — употребленная для этих же целей часть деблокированных резервов; z_i — экспорт; φ_i — непроизводственное потребление; v_i — вновь созданные резервы i-го продукта.

Уравнение (50) будем называть *уравнением трудоемкости*. Коэффициенты A_{iq_i} представляют собой суммарные трудозатраты для производства одной единицы i-го продукта по q_i-й технологии за фиксированное время τ, принимаемое за единицу. Уравнение

(51) имеет тот же смысл, но трудозатраты в них разложены по выбранной номенклатуре специальностей. Следование традиционной для социалистической экономики *политике полной занятости* приводит к тому, что в соотношениях (50) и (51) стоят знаки $=$, а не $\geqslant$.

Неравенство (52) представляет соотношение, описывающее *фондоемкость* по соответствующим ресурсам (основным и оборотным фондам) текущего производства; коэффициенты c_{jiq_i} — необходимые количества j-го ресурса (с учетом принятого коэффициента сменности и других факторов, влияющих на коэффициенты использования фондов) для производтва по q_i-й технологии одной единицы i-го продукта в принятую единицу времени τ.

Динамическая часть модели описывается соотношениями:

$$\sum_{q_i} x_{iq_i}^{\text{разв}} + y_i^{\text{разв}} + u_i^{\text{разв}} = \sum_j \sum_{q_j} b_{ijq_i} \Delta R_{jq_j} \quad (i = 1, 2, \ldots, n); \tag{53}$$

$$P^{\text{разв}} = \sum_i \sum_{q_i} B_{iq_i} \Delta R_{jq_j}; \tag{54}$$

$$P_j^{\text{разв}} = \sum_i \sum_{q_i} B_{jiq_i} \Delta R_{jq_i} (j = 1, 2, \ldots, l); \tag{55}$$

$$R_j^{\text{разв}} \geqslant \sum_i \sum_{q_i} D_{jiq_i} \Delta R_{jq_j} \quad (j = 1, 2, \ldots, k), \tag{56}$$

где ΔR_{jq_j} — прирост i-го ресурса, полученный за счет q_j-й технологии производства этого ресурса за принятую единицу времени τ.

Соотношения (53)—(56) представляют собой основную часть модели. Чтобы не усложнять систему обозначений, которая уже и без того громоздка, в приведенном описании модели упрощены или опущены многие элементы. Перечислим важнейшие из них, чтобы при необходимости читатель мог выполнить недостающие элементы. Прежде всего систему необходимо дополнить соотношениями для резервов:

$$\Delta x_i = v_i - u_i; \quad x_i \geqslant 0 \quad (i = 1, 2, \ldots, n),$$

где x_i обозначен государственный резерв i-го продукта, Δx_i — его приращение за принятую единицу времени. Далее необходимо включить в систему рассмотренное выше уравнение (48) баланса внешней торговли, дополнив его заданием желательной политики $M(t)$ изменения валютных запасов (то же самое имеет место и для других запасов). Коэффициенты a_{ijq_j} в уравнение (49) должны учитывать не только затраты предметов труда, но и частичное восполнение средств труда, необходимое для поддержания постоянства используемых ресурсов, а также затраты на управление текущим производством и распределением. В соотношениях (49), (51) и (52) необходимо учитывать также затраты на профессионально-техни-

ческое и высшее специальное образование, идущие на восполнение естественной убыли работников соответствующих специальностей. Эти затраты можно учитывать либо добавлением в правые части соотношений (49), (51), (52) соответствующих членов (аналогичных уже имеющимся в них суммам), либо соответствующим увеличением коэффициентов a_{jiq_j}, A_{jiq_i} и C_{jiq_i}.

Затраты на специальное образование с целью изменения специальностей (в интересах совершенствования технологии) или увеличения числа работающих специалистов относятся с помощью таких же приемов на соотношения (53), (55) и (56). На эти же соотношения относятся расходы на прикладную науку, проектно-конструкторские работы и на управление развитием экономики. В тех случаях, когда расходы на поддержание экономики в текущем состоянии и на ее развитие невозможно разделить точно (как это может иметь место в части расходов на управление), то можно делить эти расходы в соответствии с принятыми пропорциями деления национального дохода.

Методы решения задач перспективного планирования с использованием описанной модели лучше всего объяснить на простом примере, поскольку в общем случае за громоздкими обозначениями может быть утеряна суть дела.

Пусть имеем всего два типа продуктов (две отрасли) с двумя технологиями в каждой, причем первый продукт используется только для непроизводственного, а второй — только для производственного потребления. Предположим, что расчет ведется при отсутствии внешней торговли и с неизменными запасами. Трудовые ресурсы, отведенные для статической и динамической части модели в течение всего планового периода постоянны и равны соответственно 12 и 6. Все материальные ресурсы сведем в один ресурс R_1, представляющий собой продукт исключительно лишь второй отрасли. Первоначальная пропорция деления ресурса R_1 между статикой и динамикой 1 : 1.

Вначале рассмотрим статическую часть модели, временно опуская для простоты обозначений индекс «тек» над соответствующими величинами. Задавая некоторые числовые значения технологических коэффициентов, выпишем статическую часть модели:

$$x_{11} + x_{12} = \varphi_1; \tag{57}$$

$$x_{21} + x_{22} = 0{,}1x_{11} + 0{,}3x_{12} + 0{,}2x_{21} + 0{,}5x_{22}; \tag{58}$$

$$P - 12 = 4x_{11} + x_{12} + 2x_{21} + x_{22}; \tag{59}$$

$$R_1 \geqslant 2x_{11} + 5x_{12} + 4x_{21} + 10x_{22}. \tag{60}$$

Предположим, что в начале планового периода $x_{12} = x_{22} = 0$. Иными словами, как первый, так и второй продукт выпускался

лишь по первым технологиям. Из соотношений (59) и (60) видно, что эти технологии обладают большей трудоемкостью (в четыре и в два раза), но меньшей фондоемкостью (в 2,5 раза) по сравнению с конкурирующими с ними технологиями. Из уравнений (58) и (59) легко найти, что при начальном распределении технологий ($x_{12} = x_{22} = 0$) $x_{11} = 2\frac{14}{17}$; $x_{21} = \frac{6}{17}$. Потребление $\varphi_1 = 2\frac{14}{17}$, а суммарные производственные фонды в обеих отраслях при условии их полной загрузки вычисляются из соотношения (60), превращающемся теперь в равенство: $R_1 = 7\frac{1}{7}$. Эти фонды между первой и второй отраслями распределяются в пропорции 4 : 1, а трудовые ресурсы — в пропорции 32 : 1.

Таблица 17

Номер комбинации	x_{11}	x_{12}	x_{21}	x_{22}	φ_1	R_1
I	$2\frac{14}{17}$	0	$\frac{6}{17}$	0	$2\frac{14}{17}$	$7\frac{1}{17}$
II	0	$6\frac{6}{7}$	$2\frac{4}{7}$	0	$6\frac{6}{7}$	$44\frac{4}{7}$
III	$2\frac{6}{7}$	0	0	$\frac{4}{7}$	$2\frac{6}{7}$	$11\frac{3}{7}$
IV	0	$7\frac{1}{2}$	0	$4\frac{1}{2}$	$7\frac{1}{2}$	$82\frac{1}{2}$

Первая задача, решаемая с помощью модели — нахождение общего направления технической политики, т. е. такого соотношения технологий, при котором получается максимальное значение для величины потребления φ_i.

В общем случае решение этой задачи может быть получено методами линейного программирования. Поскольку вектор потребления (φ_1, φ_2, ..., φ_n) представляется в виде (α_1 φ, $\alpha_2\varphi$, ..., $\alpha_n\varphi$, где α_i — заданные константы, то φ легко исключить из всех уравнений, кроме одного, как это имеет место в рассматриваемом примере. Решая это оставшееся уравнение относительно φ, получаем линейную функцию от x_{iq_i}, подлежащую максимизации. Остальные уравнения и неравенства тоже линейны. Тем самым задача оказывается сведенной к линейному программированию.

Как известно из теории линейного программирования (см. параграф 2 гл. II), оптимум следует искать в угловых точках многогранника, ограничивающего область, в которой ищется оптимум.

В нашем случае это соответствует *чистым технологиям*, т. е. таким, когда весь i-й продукт производится полностью лишь одной из наличных технологий. В рассматриваемом примере это приводит к перебору четырех комбинаций, которые сведены в табл. 17.

Хотя, на первый взгляд, наиболее выгодна IV комбинация технологий, однако легко понять, что это не так, поскольку переход от I $\rightarrow$ IV потребует гораздо большего увеличения производственных фондов $\left(\text{с } 7\frac{1}{17} \text{ до } 82\frac{1}{2}\right)$, чем при переходе I $\rightarrow$ II $\left(\text{с } 7\frac{1}{17} \text{ до } 44\frac{4}{7}\right)$, приводящем почти к тому же значению величины потребления φ_1, что и переход I $\rightarrow$ IV. Время перехода к новой комбинации технологий рассчитывается с помощью динамической части модели. Пусть для $\tau = 1$ *год* уравнения динамической части модели

$$x_{21}^{\text{разв}} + x_{22}^{\text{разв}} = 0{,}5\Delta R_{11} + 0{,}4\Delta R_{12}; \tag{61}$$

$$P^{\text{разв}} = 6 = 3\Delta R_{11} + 3\Delta R_{12}; \tag{62}$$

$$R_1^{\text{разв}} \geqslant 2\Delta R_{11} + 3\Delta R_{12}. \tag{63}$$

Поскольку ресурс R_1 производится целиком во второй отрасли, а при переходе I $\rightarrow$ II технология в этой отрасли остается неизменной, то $x_{22}^{\text{разв}} = 0$, $\Delta R_{12} = 0$. Из уравнения (62) находим, что $R_{11} = 2$ и, следовательно, $x_{21}^{\text{разв}} = 1$; $R_1^{\text{разв}} \geqslant 4$. Так как скорость роста постоянна, а фонды второй отрасли, работающие на развитие, остаются неизменными, то нетрудно рассчитать время перехода I $\rightarrow$ II.

В начале планового периода производственные фонды первой отрасли (при условии полной загрузки) составляли

$$2 \cdot x_{11}^{\text{тек}} = 2 \cdot 2\frac{14}{17} = \frac{96}{17} = 5\frac{11}{17},$$

а во второй

$$R_1^{\text{разв}} + 4x_{21}^{\text{тек}} = 4 + \frac{24}{17} = 5\frac{7}{17}.$$

В конце планового периода соответствующие цифры равны

$$5x_{12}^{\text{тек}} = 5 \cdot 6\frac{6}{7} = \frac{240}{7} = 34\frac{2}{7}$$

и

$$4x_{21}^{\text{тек}} + 4 = 4 \cdot 2\frac{4}{7} + 4 = 16\frac{2}{7}.$$

Следовательно, учитывая, что в первой отрасли технология меняется, а во второй сохраняется, необходимо создать новые фонды в первой отрасли $34\frac{2}{7}$ единицы, а во второй $16\frac{2}{7} - 5\frac{7}{17} =$

$= 10\frac{104}{109} = 10{,}9$. Суммарное количество вновь создаваемых фондов равно $34\frac{2}{7} + 109 \approx 45{,}2$. При постоянной скорости $\Delta R_1 = 2$ в год переход I → II будет завершен за 23,6 года. Нетрудно подсчитать ожидаемое состояние экономики при принятой технической политике через 20 или 15 лет, равно как и в любой другой момент планового периода.

В рассмотренном примере вычисления упростились за счет постоянной скорости роста. В противном случае пришлось бы рассчитывать очередные состояния модели последовательно год за годом. Выбор направления перехода (технической политики на плановый период) можно осуществить следующим путем. В начале, как и в рассмотренном примере, находятся варианты, дающие максимальный или близкий к нему рост суммарной величины потребления φ. Затем для найденных вариантов подсчитываются величины вновь создаваемых производственных фондов и оцениваются *средние* скорости соответствующих переходов (например, сравнением начальной и конечной скоростей, которые можно вычислить). Из всех вариантов предпочтение отдается тем, которые обеспечивают максимальную скорость роста φ, что означает и максимальные темпы роста *полезной* (т. е. направленной на достижение поставленной цели) *производительности труда.*

Если ни один из отобранных вариантов не обеспечивает достижение поставленной цели в течение планового периода, то возможны два варианта последующих действий. Простейшим является обращение к системе постановки целей с просьбой уменьшить задание. Возможно, однако, также обращение к *системе управления техническим прогрессом* для нахождения таких комбинаций технологий, которые обеспечили бы решение поставленной задачи *с учетом времени и ресурсов, необходимых на разработку соответствующих технических решений.* Действительно, до сих пор мы выбирали оптимальный набор из уже существующих технологий. Не исключено, что при соответствующем перераспределении ресурсов (в пользу прикладной науки за счет прямых капитальных вложений в развитие народного хозяйства) и *целенаправленном управлении разработками в междуведомственном масштабе* путь к цели окажется более коротким.

Например, в рассмотренном примере создание за 5—8 лет технологии x_{23} с коэффициентами $a_{223} = 0{,}1$; $A_{23} = 0{,}5$; $C_{123} = 3$; $b_{223} = 0{,}3$; $B_{23} = 1$; $D_{223} = 1$ даже при условии, если бы ее разработка почти полностью поглощала в этот период весь фонд накопления, могло бы привести к более быстрому достижению поставленных перед экономикой целей, чем при использовании уже известной технологии. Не разбирая другие возможности дальнейшей оптими-

зации перспективных планов, например, за счет постоянного прогнозирования конъюнктуры на мировом рынке и целенаправленного управления внешней торговлей, остановимся более подробно на взаимодействии описанной системы с системой управления техническим прогрессом.

Прежде всего необходимо заметить, что описанная модель перспективного планирования должна быть гибкой. При ее построении родственные продукты объединяются в один групповой продукт, а технологические коэффициенты усредняются в соответствии с ожидаемыми пропорциями производства отдельных видов этого группового продукта (например, в одну группу объединяются все полимерные материалы или все металлорежущие станки). Без такого объединения модель получилась бы чрезмерно громоздкой. С другой стороны, нельзя заранее зафиксировать все указанные группы, поскольку в процессе отбора вариантов необходимо в зависимости от обстоятельств разукрупнять одни группировки и укрупнять другие. Необходимость в этом обусловливается, во-первых, необходимостью быстро реагировать на изменения предложений по составу технологий: если новая технология, подлежащая проверке, касается только части группового продукта, то эту часть приходится выделять (на время проверки указанной технологии) в отдельный продукт. Во-вторых, без процессов последовательного разукрупнения и укрупнения групп в большинстве случаев невозможно точно определить ожидаемые пропорции производства отдельных продуктов, объединяемых в группы, а, значит, и осуществить правильное усреднение технологических коэффициентов.

Гибкость системы планирования состоит в том, что центральная ее часть (например, описанная выше модель) должна опираться на систему непрерывного прогнозирования технологических нормативов в *самой подробной* номенклатуре, в которой это оказывается возможным. Для этого в систему должны быть вовлечены все проектно-конструкторские и технологические силы, рассредоточенные по отраслям и ведомствам. Соответствующие отраслевые автоматизированные системы должны иметь возможность по требованиям центральной плановой модели быстро подсчитывать технологические коэффициенты по любым заданным группировкам*.

В системе должен существовать непрерывно обновляемый и пополняемый *банк новых технологий*. Дело в том, чтобы из потока заявок и предложений на отдельные *частные* усовершенствования формировать *комплексные технологические решения* (пусть еще не

* Это требование относится не только к перспективному плану, но и к другим видам планирования. Различие будет состоять лишь в том, что для краткосрочных планов можно в большей степени опираться на действующие нормативы.

проработанные в деталях и не опробованные), которые могли бы проверяться в плановой модели, как возможные *средства* решения задач, ставящихся перед экономикой. Технологические комплексы, оказавшиеся наиболее полезными в этом плане, переводятся в системе управления научно-техническим прогрессом в разряд *целей*, которых необходимо достигнуть.

Каждое предложение о новом технологическом комплексе включается в систему непрерывного прогнозирования технического прогресса, построенную на основании методов параграфа 7 гл. II, и охватывающую в перспективе все научно-технические силы страны. Из такой системы можно получать сведения об ожидаемых сроках реализации того или иного технологического комплекса и составляющих его элементов, а также оценки стоимости их разработок. Величины технологических коэффициентов должны быть включены в описание комплексов и все время учитываться в процессе прогнозирования. Если из прогноза следует, что многие элементы комплекса можно реализовать гораздо раньше, чем весь комплекс в целом, система управления банком новых технологий организует системный анализ и формирование из этих (а может, и некоторых других) элементов новых технологических комплексов.

По тем же принципам организуется система непрерывного прогнозирования времени и стоимости разработок проектов новых видов продуктов для внепроизводственного потребления. При этом из системы управления ценами поступают предложения о *желательном* ориентировочном уровне будущих *стоимостей* этих продуктов. Эти предложения учитываются экспертами в системе прогнозирования научно-технического прогресса, поскольку от них существенно зависят сроки реализации и затраты на разработку проектов и технологии производства таких продуктов.

Система непрерывного прогнозирования и управления научно-техническим прогрессом должна опираться на единый банк данных, абонентами которого были бы все научно-исследовательские, проектно-конструкторские и технологические организации. Необходим также определенный уровень централизации управления комплексными разработками, носящими, как правило, ярко выраженный межведомственный характер.

Централизация управления подобными разработками предполагает также организацию системы *государственного страхования риска* при разработках и внедрении крупных комплексных проектов. Это означает, что потери, которые может нести предприятие или отрасль при начальном внедрении результатов разработки, должны компенсироваться из специального государственного фонда с тем, чтобы последующее использование полностью отработанной и опробованной технологии компенсировалось предприятиями и отраслями соответствующими взносами в этот фонд.

Одним из важнейших элементов управления комплексными разработками и внедрением (ведущихся на основе сетевых графиков) является расчет финансирования на все время разработки с соответствующей разбивкой по годам и соисполнителям. В этом состоит главная идея так называемого *программного управления разработками*. При обычных методах планирования чаще всего идет равномерное наращивание финансирования по годам, в результате чего обычно происходит чередование периодов избыточного и недостаточного финансирования разработок. А исполнитель, получив деньги, стремится их истратить, чтобы не столкнуться с фактом уменьшения финансирования на следующий год, когда ожидается резкое увеличение затрат. Годовые бюджеты отраслей и отдельных организаций, выделяемые на новые разработки, складываются из соответствующих долей их участия в различных комплексных проектах.

Помимо указанного источника целевого финансирования должно существовать и обычное отраслевое финансирование, призванное обеспечить выполнение разработок более мелкого масштаба. Четкое взаимодействие и координация таких разработок могли бы осуществляться с помощью создания специального банка данных о взаимных требованиях, возникающих в процессе таких разработок. Если, например, разработчик нового технологического процесса или устройства сталкивается с необходимостью решения любой математической проблемы или получения нового материала с заданными свойствами, то он направляет соответствующее требование в центральный банк данных. Если ответ на поставленный вопрос известен и в банке имеются соответствующие документы (описания, патенты, проекты, отчеты, рефераты научных статей), то они направляются заказчику. При отсутствии ответа на вопрос он формируется в виде *задания на исследование*, которое рассылается во все организации, способные его выполнить, в качестве пожеланий при составлении планов работ, а иногда и в виде прямых заданий.

Существенную роль в задачах макроэкономического уровня играет система управления ценами и заработной платой. Одна из важнейших задач этой системы — обеспечить действительное приобретение населением предназначенной для этой цели части конечного продукта (как по объему, так и по номенклатуре) и вместе с тем избежать превышения спроса над предложением.

При планировании размеров зарплат, помимо обычно принимаемых во внимание факторов (квалифицированность и сложность труда, необходимость специальной подготовки, выявившиеся дефициты и перетоки рабочей силы), необходимо иметь специальную систему для непрерывного учета и анализа всех *пожеланий* об изменении места работы (а не только *фактов* такого изменения).

Для этого может служить специальная сеть территориальных вычислительных центров (возможно, объединенная с территориальными ВЦ системы опроса общественного мнения), которые регистрировали бы такие пожелания вместе с мотивами их возникновения (размер зарплаты, престижность профессии и т. п.). Система могла бы выполнять и текущую работу по установлению контактов между людьми, желающими изменить место работы и организациями, где возникает нужда в соответствующих кадрах. Однако ее основной задачей должно стать прогнозирование будущих возможных дефицитов и перетоков рабочей силы (по специальностям, отраслям и территориям), а также их наиболее вероятных причин.

Эта информация используется не только для планирования желательных уровней зарплаты, но и для других мероприятий, направленных на предотвращение нежелательных перетоков рабочей силы (задания на автоматизацию и механизацию, улучшение труда и быта, благоустройство городов и сел, повышение престижности тех или иных профессий и т. д.). Следует подчеркнуть, что организация единой сети АСУ в масштабах всей страны может дать возможность планировать именно желательные *средние* уровни зарплат, предоставив право фактического их установления руководителями предприятий и организаций. При этом необходимо иметь систему постоянного контроля за средними уровнями зарплат для различных профессий, квалификаций, территорий и т. д. Необходимые меры экономического или директивного характера должны применяться лишь при заметных отклонениях от заданных средних, причем только к тем, у кого это отклонение наиболее значительно.

Для прогнозирования и планирования желательного уровня розничных цен следует, опираясь на систему опросов общественного мнения и на локальные (в пределах одного города или района) эксперименты по временному изменению цен, строить кривые зависимости спроса от цены (по разным группам населения и разным районам страны). На этой основе и на основе экспертных оценок создается постоянно действующая система оценок желательных уровней цен на тот или иной период с тем, чтобы структура спроса соответствовала планируемой (в качестве цели) структуре конечного продукта. Абсолютные же уровни цен при известных относительных уровнях можно рассчитать из условия равенства сумм цен на все планируемые для сбыта продукты суммам всех зарплат и других денежных выплат (пенсий, стипендий и т. п.) за соответствующий период времени τ. Не исключено, что в соответствии с принятой политикой накопления денежных ресурсов у населения (прежде всего в виде вкладов в сберкассы) могут планироваться. отклонения этих двух сумм друг от друга в ту или другую сторону.

Для предотвращения создания временных локальных дефицитов может оказаться целесообразным, чтобы товарная масса, планируемая для производства, превышала бы на определенную величину товарную массу, планируемую для сбыта.

При установлении розничных цен на потребительские товары или услуги необходимо, чтобы эта цена не превышала предела, при котором индивидуальное частное производство могло бы оказаться конкурентноспособным с индустриальным. Уровень цен, по которым производятся безналичные расчеты между предприятиями, устанавливается на основе расчетов, проводимых на предприятиях (под контролем представителей специального государственного органа) по методике, описанной в параграфе 4 настоящей главы. При этом цена должна усредняться по предприятиям, выпускающим одинаковую (в том числе и одинаково качественную) продукцию. При превышении расчетной цены над средней соответствующий излишек вычитается из прибыли, а в противном случае — прибавляется к ней.

Если в процессе прогнозирования желательного уровня цен на отдельные потребительские товары и передачи их для оценки сроков достижения в систему прогнозирования научно-технического прогресса и перспективного планирования выяснится, что срок достижения такого уровня для фактической стоимости их производства выходит за пределы планового периода, то возможны четыре типа решений.

1. Планировать розничную цену на этот продукт ниже его фактической стоимости, компенсируя убытки предприятий из госбюджета (в виде отрицательного подоходного налога). Разумеется, поскольку сумма розничных цен постоянна (равна сумме зарплат), это неизбежно повысит цены на другие товары, так что общая сумма налога с оборота не изменится.

2. Усилить исследования в области путей уменьшения фактической стоимости производства.

3. Организовать заблаговременную целенаправленную рекламу соответствующего товара с тем, чтобы обеспечить возможность планируемого сбыта по цене, не меньшей фактической стоимости (результаты такой рекламы должны непрерывно контролироваться описанной выше системой оценок желательного уровня цен).

4. Скорректировать цель, уменьшив долю соответствующего продукта в векторе (цели или даже исключить его совсем, если речь идет о принципиально новом виде изделий). Все эти решения должны базироваться на постоянно проводимом системном анализе.

Система перспективного планирования является основой, с помощью которой определяется направление совершенствования

технологии, изменение пропорций между отдельными отраслями народного хозяйства, общие задания перед капитальным строительством. При планировании на менее длительные периоды (5 лет или 1 год) происходит дальнейшая детализация планов, в том числе конкретизация строящихся объектов, формирование конкретных заданий на научно-исследовательские и опытно-конструкторские работы. В принципе здесь применимы многие методы, описанные в системе перспективного планирования. Однако возникают и новые задачи, например, задачи, связанные с территориальным размещением вновь строящихся объектов. Их решение тесно увязывается с планами комплексного развития отдельных территорий и оптимизируется прежде всего по принципу уменьшения транспортных и строительных расходов и расходов на передачу электроэнергии. Ограничениями служат прогнозные данные о территориальном распределении трудовых ресурсов, а также различные требования, вытекающие из целей внеэкономического характера (примером таких целей может служить развитие передовых отраслей промышленности в некогда отсталых районах).

При конкретных плановых расчетах должны широко использоваться методы математического моделирования, особенно в режиме диалога человека с машиной. Весьма существенным требованием является динамичная форма представления планов и режим взаимодействия плановых систем разных уровней (в отраслевом и территориальном разрезе). При этом необходимы процедуры укрупнения и разукрупнения планов и нормативов при обменах данными между разными уровнями (от предприятия до Госплана). Укрупнение планов заключается в группировке продукции и уменьшении деталировки сроков. Задачи текущего управления могут трактоваться как задачи непрерывного слежения и корректировки планов.

Приведенный перечень задач далеко не исчерпывает всех проблем планирования и управления, возникающих на макроэкономическом уровне. Здесь и вопросы оптимального планирования внешней торговли, текущее управление финансами, система безналичных расчетов, система государственной стандартизации и многое другое. Однако и уже приведенные примеры дают достаточное представление о специфике задач управления на макроэкономическом уровне.

7. Общегосударственная автоматизированная система сбора и обработки экономической информации

Рассмотренные в предыдущем параграфе задачи управления макроэкономического уровня можно успешно решать лишь при наличии специальной общегосударственной *автоматизированной сис-*

темы сбора и обработки экономической информации (ОГАС). Задача построения такой системы поставлена директивами XXIV съезда КПСС по девятому пятилетнему плану. Основные требования, которым должна удовлетворять эта система, вытекают из существа задач планирования и управления, описанных в предыдущих параграфах. Это прежде всего унифицированность, полнота, объективность и иерархическая организация информации, возможность быстрого (автоматического) обмена информацией между любыми предприятиями и органами управления любого уровня (в ведомственном и территориальном разрезе), наличие специальных автоматических процедур для укрупнения и разукрупнения информации, динамическая форма представления планов и возможность быстрой автоматической коррекции планов не только в рамках ВЦ одного объекта, но и в ВЦ всех тех объектов, которые связаны с первым объектом таким образом, что коррекция его планов делает необходимым также и коррекцию их собственных планов.

Удовлетворять этим требованиям можно лишь в том случае, если все ВЦ страны, занятые обработкой экономической информации независимо от их ведомственной принадлежности, будут объединены в единое целое с помощью каналов связи, унификации форм представления и состава информации и специального математического обеспечения для управления процессами обмена информации между отдельными ВЦ. Поскольку ЭВМ предъявляют высокие требования к каналам связи как по надежности, так и по пропускной способности и поскольку стоимость средств связи высока, одним из основных принципов при проектировании ОГАС должен стать принцип наиболее экономного и эффективного использования связи. Этот принцип исключает подход, основанный на закреплении каналов для соединения «каждого с каждым», требуя организации высокоэффективной автоматической коммутации каналов и сообщений.

Кроме того, для того чтобы ведомственные ЭВМ в такой системе не уподоблялись человеку, которого то и дело отрывают от работы телефонными звонками, требуется четкая диспетчеризация процессов взаимных обращений и обменов информацией между различными ВЦ. Поэтому в ОГАС должно существовать центральное (междуведомственное) звено, выполняющее функции диспетчеризации и коммутации сообщений. Его техническую базу составляет система общегосударственных (междуведомственных) информационно-вычислительных центров (ОГИВЦ), являющихся одновременно и центрами коммутации сообщений. Расположение этих центров определяется конфигурацией системы связи, осуществляющей наиболее экономичным способом решение следующих двух основных задач. Во-первых,— это возможность непосредственного соединения всех ОГИВЦ между собой с помощью

широкополосных (сотни килогерц) каналов связи, организованных по радиально-кольцевому принципу. Во-вторых,—это охват сетью ОГИВЦ всей территории страны так, чтобы все ведомственные ВЦ могли быть подсоединены к соответствующим (территориально близким к ним) ОГИВЦ с помощью надежных узкополосных каналов связи (тысячи или даже сотни герц) в качестве *абонентов ОГАС**.

Выбор пропускной способности каналов связи в ОГАС желательно согласовать с характеристиками ЭВМ, установленных в ОГИВЦ и у абонентов. Широкополосные каналы связи должны допускать возможность *прямого обращения ЭВМ*, установленных в любом ОГИВЦ, к внешней памяти (магнитным лентам и дискам) ЭВМ в любом другом ОГИВЦ, осуществляя передачу данных в *темпе работы этих устройств*. Разумеется, для возможности такого обращения недостаточно одних каналов связи: необходимы еще устройства для сопряжения ЭВМ с этими каналами и специальное математическое обеспечение — *операционная система сети ОГИВЦ*.

Пропускная способность узкополосных каналов должна быть согласована со скоростью работы наиболее быстрых механических устройств ввода—вывода, имеющихся у соединяемых между собой ЭВМ. При этом должна быть обеспечена возможность непосредственной работы любой из двух соединенных между собой ЭВМ с любым устройством ввода—вывода ЭВМ на другом конце линии связи. В этом темпе должен быть возможен и прямой обмен информацией между двумя ЭВМ. Обеспечение такой возможности также требует создания специальных сопрягающих устройств и специального математического обеспечения как в сети ОГИВЦ (центральном звене ОГАС), так и у всех ее абонентов.

Помимо обычных территориальных ОГИВЦ в системе должен существовать *головной ОГИВЦ*, расположенный в непосредственной близости от места сосредоточения центральных органов управления (от уровня министерства или ведомства и выше) и соединственный с ВЦ этих органов широкополосными каналами связи. В зависимости от структуры управления подобные центры могут быть построены не в одном, а в нескольких пунктах, например в столицах крупных союзных республик. Отдельные министерства или ведомства, расположенные вне мест сосредоточения центральных органов, могут замыкаться (с помощью широкополосных каналов) на ближайшие территориальные ОГИВЦ. Поскольку все центры сети ОГИВЦ представляют собой фактически единое целое, такое подсоединение обеспечивает те же самые возможности, что и прямое подсоединение к головному ОГИВЦ.

---

* Для отдельных абонентов могут потребоваться широкополосные каналы.

Абонентам первого уровня, подсоединенным к сети ОГИВЦ широкополосными каналами связи, должна быть обеспечена возможность прямого обмена информацией между установленными у них ЭВМ в темпе работы устройств внешней памяти. В отдельных случаях такая возможность должна представляться и абонентам второго уровня, подключенным к сети ОГИВЦ с помощью узкополосных каналов. Наиболее же распространенным режимом связи для абонентов второго уровня явится коммутация сообщений. Абоненты (точнее их ЭВМ) посылают в сеть ОГИВЦ сообщение с условным кодом адресата. Эти сообщения запоминаются в ЭВМ

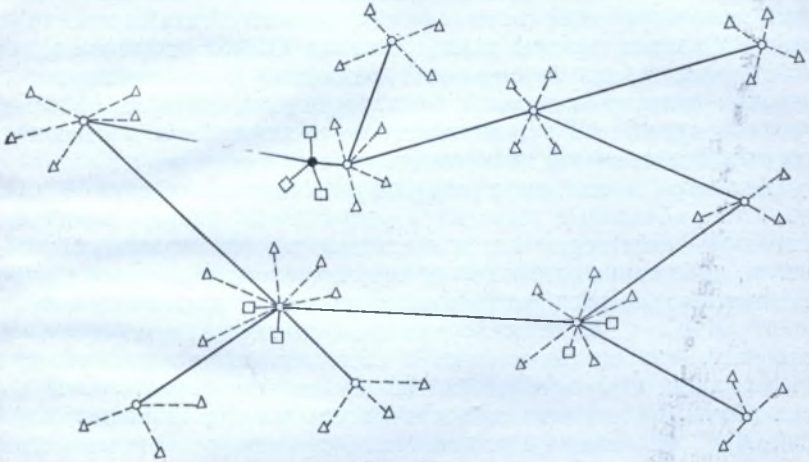

Рис. 27.
Структура ОГАС.

ОГИВЦ, сортируются по приоритетам и срочностям и в соответствующем порядке направляются адресатам.

Для передачи больших объемов информации возможна и прямая доставка абонентом на ближайший ОГИВЦ соответствующих бобин с лентами или кип дисков. Содержащаяся в них информация передается в ОГИВЦ, обслуживающий адресата, и копируется на ленте или дисках, которые направляются адресату. При этом не исключается и прямой обмен лентами и пакетами дисков между отдельными абонентами, не обращаясь к услугам сети ОГИВЦ.

На рис. 27 показана схематическая структура ОГАС. Кружками на ней изображены ОГИВЦ (головной ОГИВЦ зачернен), квадратами — абоненты первого уровня, треугольниками — абоненты второго уровня. Узкополосные каналы изображены пунктирными линиями, а широкополосные — сплошными.

Преимуществом такой структуры, помимо минимизации затрат на линии связи, является еще большая гибкость и легкая

приспособляемость к любым изменениям организации системы управления и административного деления. При образовании новых министерств или передаче предприятия из одного министерства в другое, равно как и при образовании новых областей, структура ОГАС остается прежней, хотя ведомственная принадлежность абонентов при этом, разумеется, изменится.

Межведомственное центральное звено ОГАС (сеть ОГИВЦ) должно иметь свою собственную человеко-машинную диспетчерскую службу, организующую удовлетворение поступающих от различных абонентов заявок на информацию. Для возможности диспетчирования всей системы (а не только его центрального звена) вычислительные центры всех абонентов ОГАС, оставаясь полной собственностью соответствующих ведомств, в оперативно-диспетчерском отношении должны быть подчинены центральной диспетчерской службе ОГАС, выполняя ее распоряжения в отношении состава передаваемой информации и сроках ее передачи.

Аналогом здесь могут служить энергетические службы предприятий, обязанные выполнять оперативные распоряжения диспетчеров энергосистемы, или ведомственные железнодорожные ветки, подсоединенные к соответствующим магистральным линиям. Ведомственные ВЦ в методическом отношении должны быть подчинены ОГАС, т. е. ее информационно-техническим службам. Это означает, что все изменения в составе и формах представления информации, к которой возможны обращения через ОГАС от других абонентов, должны согласовываться с информационной службой ОГАС. Изменения в устройствах сопряжения с каналами связи, обеспечивающие обмены программ, могут производиться лишь при согласовании с технической службой ОГАС. Аналогом здесь могут служить бухгалтерии предприятий и организаций, не имеющие права производить произвольные изменения в формах финансовых документов и порядке их заполнения.

Для пояснения принципов организации информационной *базы ОГАС* необходимо прежде всего перечислить основные типы абонентов ОГАС.

Об абонентах, представляющих высшие звенья управления, уже было сказано выше. Остановимся более подробно на абонентах, представляющих низовое звено, где собственно и рождается основная масса экономической информации. Это прежде всего автоматизированные системы управления крупными предприятиями и объединениями. Речь идет не только о промышленных предприятиях, а и о строительных (ВЦ строительных трестов и комбинатов), транспортных (ВЦ управлений дорог, крупных железнодорожных узлов, портов, пароходств, аэропортов, крупных автохозяйств), торговых (ВЦ крупных универмагов), сельскохозяйственных (ВЦ отдельных крупнейших совхозов) и др.

Абонентами ОГАС должны являться также ВЦ, решающие задачи управления крупными научно-исследовательскими и проектно-конструкторскими организациями, ВЦ крупных больничных и учебных комплексов (для оперативного решения задач материально-технического снабжения, финансовых расчетов и других), ВЦ крупных предприятий связи и др. Большой класс абонентов ОГАС составят территориальные ВЦ общесоюзных организаций, таких как Госснаб, ЦСУ, Госбанк, Стройбанк, а также сеть ВЦ, предназначенных для опросов общественного мнения.

Для мелких предприятий, принадлежащих союзным и республиканским министерствам и ведомствам, предприятий местной промышленности, колхозов, мелких магазинов, предприятий общественного питания и бытового обслуживания, зрелищных предприятий, мелких и средних организаций системы здравоохранения, просвещения, высшего, среднего специального и профессионально-технического образования, заготовительных контор и других должны создаваться *кустовые ВЦ*, также входящие в число абонентов ОГАС.

Эти ВЦ целесообразнее всего строить по территориально-отраслевому признаку, замыкая на один кустовой ВЦ группу предприятий или организаций одинакового профиля, расположенных на какой-то достаточно компактной территории.

Особо следует выделить ВЦ облисполкомов и горисполкомов крупных городов, предназначенные прежде всего для *комплексного* планирования и управления городским и областным хозяйством (планы комплексного развития городов, сел и поселков, развитие местной транспортной сети и т. п.).

Наконец, помимо абонентов, входящих в ОГАС через свои собственные или кустовые ВЦ, целесообразно иметь еще один класс абонентов, подключаемых к ОГАС через относительно простые и дешевые *информационно-диспетчерские пункты* (ИДИ). Этот способ подключения должен быть обеспечен в первую очередь для местных партийных и советских органов, не имеющих своих собственных ВЦ. В простейшем случае такой пункт может быть оборудован одним телетайпом, подключенным через коммутируемые или закрепленные каналы к ближайшему ОГИВЦ.

Что же касается высших звеньев управления республиканского или союзного уровня (начиная от министерств и ведомств), то создание у них ИДП может служить первым этапом на пути создания собственных ВЦ.

Переходим теперь к принципам построения *информационной базы ОГАС*. Поскольку основная масса экономической информации рождается в низовом звене, основная тяжесть создания и ведения (т. е. постоянного пополнения и обновления) соответствующих информационных массивов ложится на ВЦ и АСУ низового звена.

Общее требование к составу информации на низовом звене заключается в том, что она должна быть *полной*, *объективной* и *своевременной*. Требование полноты означает, что ВЦ, обслуживающий объект низового звена (предприятие или организацию), должен иметь об этом объекте самую подробную исчерпывающую информацию. Иными словами, он должен знать полный состав и состояние всех единиц оборудования (вплоть до самых мелких) и других основных фондов, которыми обладает объект, полный перечень запасов продукции и материалов в самой подробной номенклатуре на всех складах, в том числе и промежуточных, хранящих продукцию на разных стадиях ее обработки (незавершенное производство). Для изделий с длительным циклом производства, а также проектно-конструкторских, строительных, ремонтных, монтажных и пусконаладочных работ должны храниться данные о состоянии работ в соответствии с заранее составленными и также хранящимися в ВЦ сетевыми графиками.

ВЦ должен иметь информацию о всех работающих на обслуживаемых им объектах (пол, возраст, образование, профессия, квалификация, опыт работы, размер зарплаты, включая премии и другие виды доплат). Должны храниться данные о состоянии счетов в банке, а также вся планово-экономическая информация. Помимо системы материальных и трудовых нормативов (в самой подробной детализации) в памяти ЭВМ (на магнитных лентах и дисках) должна храниться в динамической форме (т. е. в постоянной готовности к любым возможным изменениям) *взаимоувязанная* система планов.

Требование объективности информации означает, прежде всего то, что она должна формироваться и пополняться не в результате простых устных сообщений, а на основе продуманной системы первичной документации, совмещенной с системой первичного материального и финансового учета, а также (по мере возможности) на данных, поступающих от системы автоматических датчиков.

Рассмотрим один из возможных примеров организации учета для системы мелких и средних магазинов. Прежде всего необходимо, чтобы промышленность, производящая товары народного потребления, и крупные торговые базы при поставке этих товаров в розничную торговую сеть снабжали каждую единицу такого товара тремя идентичными ярлыками, на которых в *удобной для автоматического считывания форме* (перфорация, магнитные чернила и т. п.) были бы записаны все необходимые данные о нем, включая фасон, размер и т. п. (сорт, цена, дата выпуска, наименование предприятия-изготовителя). В момент приемки товара в магазине один экземпляр ярлыка возвращается лицу, доставившему груз, один бросается в специальный запечатанный ящик — кассу, а третий, на котором предварительно пробивается дата поставки,

остается в магазине до момента продажи товара, после чего он бросается в другую кассу*. Каждый вечер после закрытия магазинов специальный инкассатор ярлыков доставляет ярлыки в ВЦ, где немедленно производится учет всех поступлений, продаж и вычисление остатков по всей номенклатуре товаров. Здесь же производится учет выручки и ее сравнение с объемом продаж. При условии автоматизации ввода ярлыков в ЭВМ такая работа за ночь легко может быть выполнена даже маломощной ЭВМ для многих десятков магазинов. При этом весь учет и ведение соответствующих отчетных документов перемещается в ВЦ.

На заводах данные о выпуске продукции в ВЦ также должны поступать в виде машинных копий соответствующих документов первичного учета (приемо-сдаточных ведомостей и др.). Объективность информации должна обеспечиваться целым комплексом мер. Среди этих мер следует отметить разумное дублирование и систему логических проверок, что имеет место, например, и в рассмотренном выше случае за счет раздельного поступления первичных данных материального и финансового учета и возможности их немедленного сравнения. Необходимо осуществлять периодические проверки — инвентаризации, без чего в системе могут накапливаться различного рода случайные ошибки.

Важнейшей мерой является перемещение (с уменьшением общей численности) учетных работников в ВЦ и строгое закрепление за ними персональной ответственности за правильность ведения всех основных информационных массивов и, прежде всего, массивов данных материального и денежного учета.

Своевременность информации означает своевременное пополнение и обновление информационных массивов в ВЦ при любых изменениях, происходящих на обслуживаемых этим ВЦ объектах. Данные об изменениях (в виде соответствующих документов или сигналов датчиков) должны передаваться в ВЦ не тогда, когда в них возникла потребность (срочный запрос из ОГАС!), а сразу же после того, как соответствующее изменение совершилось, *с минимальной возможной задержкой*. Величина этой задержки различна, но в любом случае она не должна превышать одни сутки. В ряде же случаев счет должен идти на часы и минуты (а в исключительных случаях даже на секунды).

Очень существенным является вопрос об унификации информации. Для того чтобы различные абоненты ОГАС могли понимать друг друга, необходима унификация основной массы передаваемой информации. Это означает, что должны применяться *единые*

---

* Для нештучного нерасфасованного товара ярлык бросается в кассу после окончания продажи более крупной единицы измерения товара, например ящика сахара, мясной туши и т. п.

системы кодов для обозначения различных материалов и изделий, различных предприятий и организаций, единые формы паспортов оборудования и т. д. и т. п. Вместе с тем при выполнении большинства расчетов на низовом уровне единая всесоюзная система классификации может оказаться чересчур громоздкой.

Выход из этого положения может быть найден в двух направлениях. 1. В низовом звене можно применять всесоюзную систему классификации только в основных информационных массивах (т. е. в информационной модели объекта). В рабочих же массивах, с которыми работают программы планирования и управления, не требующие выхода на ОГАС, могут использоваться любые местные (ведомственные) системы кодирования.

2. Использование специальных *перекодирующих программ*, которые при обращении в ОГАС или запросах из ОГАС осуществляют *быстрый перевод* запрашиваемой информации из местных кодов в общесоюзные и наоборот. Разумеется, для того чтобы программы перевода были бы действительно быстрыми и эффективными, необходимо при выборе локальных систем кодирования стремиться к максимальному упрощению процедуры перевода. Важность сделанного замечания обусловливается тем, что нельзя останавливать разработку ведомственных АСУ до полного окончания разработки всесоюзных классификаторов. Однако нельзя допустить и того, чтобы эти работы велись в полном отрыве друг от друга без хорошо налаженной системы взаимной информации.

Требования, которые были перечислены применительно к организации информационной базы низового звена, в полной мере относятся к информационной базе ВЦ облисполкомов и горисполкомов. Подробность их информационных массивов должна быть такова, чтобы доходить до каждого отдельного строения, отдельной квартиры, отдельного земельного участка, подробных схем коммуникаций, дорог, транспортных маршрутов, т. е. всех данных, которые необходимы для комплексного планирования развития городов и сел, отвода земельных участков, вопросов, связанных с отселением и переселением и т. д. и т. п. В специальных вопросах текущего планирования и управления строительством, торговлей, сельским хозяйством, здравоохранением, транспортом и других эти ВЦ должны работать в тесном контакте (в зависимости от взаимного расположения, непосредственно или через ближайший ОГИВЦ) с соответствующими специализированными ВЦ.

Информационная база ВЦ органов управления сильно зависит от их уровня и характера. Укажем для примера лишь информационную базу ВЦ промышленных министерств. Она должна включать в себя полный перечень основных фондов всех предприятий соответствующей отрасли, данные о числе работающих на каждом предприятии с подробной разбивкой по профессиям и квалифика-

циям, данные о состоянии банковских счетов предприятий, данные о запасах материалов и комплектующих изделий, поставляемых на предприятие извне, данные о выпуске и отгрузке всех видов продукции, суммарные (не подетальные) нормативы (трудовые и материальные) на каждое изделие, планы взаимных поставок продукции, укрупненные сетевые графики на производство изделий с длительным циклом изготовления, проектно-конструкторские, опытные, строительно-монтажные, ремонтные и пусконаладочные работы с недельным (а иногда и ежесуточным) циклом обновления.

В информационную базу должен быть включен перспективный план-прогноз развития отрасли в динамическом представлении, а также планы на более короткие периоды (5 лет и 1 год). Специальный массив должен быть отведен для различных постановлений, приказов и распоряжений.

Перечень этот можно было бы продолжить, однако нашей задачей является дать не полный перечень, а представление о степени подробности информации на рассматриваемом уровне управления. Основой ее так или иначе является информация низового уровня.

Поэтому, не продолжая дальше перечисления необходимого состава информационной базы у всех абонентов, перейдем к вопросу о *функциях ОГАС.*

Первой и одной из важнейших является справочно-информационная функция. Имея в низовых звеньях всю полноту экономической информации, а также располагая быстрыми средствами доступа к ней и ее обработки, ОГАС обладает неизмеримо большими возможностями по удовлетворению запросов органов управления любого уровня, чем любая существующая система. При этом важно подчеркнуть, что речь идет не о простом сборе сведений, а о любой их группировке, анализе, сравнении и т. п. ОГАС должна быстро (с задержкой не более суток, а при срочных запросах с высших уровней управления за часы и даже минуты) удовлетворять любые сложные заявки на сбор и обработку экономической информации.

Приведем примеры тех заявок, которые должна удовлетворять ОГАС. Какие суммарные запасы проката черных металлов имеются в данный момент на заводах и стройках Сибири? На каком заводе данного министерства были наихудшие показатели по использованию оборудования или по росту производительности труда? Какова средняя зарплата станочников на машиностроительных заводах Поволжья? Сколько потребуется денег и материальных ресурсов для того, чтобы довести среднюю норму обеспеченности жилплощадью по всем городам данной области до 15 $м^2$ на человека? Каков средний возраст всего станочного парка страны? Какое количество работающих на всех предприятиях и учреждениях

данного города планируется через 5 лет и каков прогноз на общее число трудоспособного населения города к этому сроку?

ОГАС должна обеспечивать выполнение заявок на информацию не только со стороны органов управления, но и со стороны абонентов низового звена. Примерами таких заявок могут служить следующие заявки. Имеется ли и если да, то где разработанный проект прибора с такими-то характеристиками? На каком заводе и в какой срок планируется освоить выпуск такого-то материала? Нет ли и если есть, то где среди излишков подлежащего реализации оборудования станок такого-то типа?

Построение подобной системы полностью ликвидирует необходимость в нескончаемых бумажных потоках. Вся необходимая отчетная и учетно-статистическая информация будет по строгому графику собираться через ОГАС из всех ВЦ низового звена и передаваться (в обработанном виде) в соответствующие органы управления и контроля. То же самое касается планов, заявок на материально-техническое снабжение, усредненных нормативов для планирующих органов верхнего уровня и т. д. и т. п. В традиционной бумажной форме будет поступать лишь различного рода неформализованная информация: предложения, жалобы, письма личного характера и т. п.

Возникает естественный вопрос о нагрузках на каналы связи. Прежде всего необходимо заметить, что способ сбора информации в ОГАС гораздо экономнее с точки зрения загрузки линий связи, чем существующие традиционные методы. Рассмотрим первый из приведенных выше запросов о запасах проката черных металлов на заводах и стройках Сибири. Сегодня для быстрого удовлетворения такого запроса необходимо связываться по телефону со всеми заводами и стройками по очереди, занимая каждый раз не только местные линии связи, но и магистральную линию Москва-Сибирь. В ОГАС запрос пройдет в соответствующие ОГИВЦ, используя магистраль лишь один раз. При связях ОГИВЦ с заводами, правда, будут использоваться все те же местные каналы связи, что и раньше. Зато, сложив полученные данные, ОГИВЦ передадут в Москву лишь одни суммы, снова минимизируя загрузку магистрального канала. Кроме того (и это, по-видимому, самое важное), обмен сжатыми подпрограммами между ЭВМ использует все каналы (в том числе и местные) гораздо экономнее по сравнению с телефонным разговором людей. Большая экономия будет даже в том случае, когда с целью *повышения надежности* передачи информации будут использоваться *избыточные коды*, повышающие надежность передачи информации по *ненадежным каналам* до любой заданной величины.

Таким образом, при том же, что и сейчас, объеме циркулирующей по каналам связи информации, загрузка этих каналов при

условии создания ОГАС не только не возросла бы, но даже уменьшилась. Однако для решения новых задач управления, способных дать решающий экономический эффект, потребуется существенное увеличение информационных потоков. Поэтому при проектировании ОГАС необходимо принять специальные меры для наиболее экономного использования каналов связи. В первую очередь это касается местных каналов связи, доля которых в общей стоимости системы без принятия этих мер может оказаться весьма значительной.

Одна из возможных мер экономии уже была указана выше. Это *прямая транспортировка* больших массивов несрочной информации на машинных носителях (магнитных лентах и дисках). Второй мерой является заблаговременное накопление на соответствующих ОГИВЦ дубликатов (в ряде случаев, возможно неполных и сокращенных) основных информационных массивов всех *абонентов низового звена*. При этом в первую очередь накапливаются часто запрашиваемые и относительно медленно меняющиеся данные. Накопленные массивы необходимо периодически обновлять либо путем передачи накопленных у абонентов *массивов изменений*, либо путем прямой транспортировки вновь изготовленных дубликатов. Разумеется, период обновления массивов будет в этом случае значительно большим, чем у абонентов и информация в них может не быть столь свежей. Наличие дубликатов массивов абонентов непосредственно в сети ОГИВЦ позволит удовлетворять большинство запросов, не нуждающихся в самой свежей информации, не обращаясь непосредственно в ВЦ абонентов. При этом основная тяжесть нагрузки ложится не на местные, а на магистральные каналы связи, которые так или иначе необходимо развивать для создания системы многопрограммного телевизионного вещания в масштабах всей страны (эти каналы будут мало загружены в ночные и ранние утренние часы).

Имея дубликаты основных информационных массивов всех своих абонентов, естественно накапливать в ОГИВЦ также дубликаты математического обеспечения всех абонентов ОГАС. С этой целью машинные комплексы абонентов должны быть программно совместимыми с машинными комплексами ОГИВЦ. Если эти последние комплексы сделать достаточно мощными и мультипрограммными, то они смогут выполнять еще две дополнительные функции: *аварийного резервирования* и *снятия пиковых нагрузок*.

Смысл первой функции заключается в том, что в случае выхода из строя машинного комплекса у любого абонента его задачи могут временно продолжать решаться на ОГИВЦ. У отдаленных от ОГИВЦ абонентов с этой целью массивы изменений должны накапливаться на перфолентах, чтобы немедленно в случае длительной аварийной остановки работы ВЦ эти изменения могли быть

переданы в ОГИВЦ по соответствующему каналу связи. Аналогичным образом, если мощности ВЦ абонента не хватает для решения отдельных особо сложных задач, они передаются для решения на ОГИВЦ. В этом состоит смысл функции снятия пиковых нагрузок.

Пользуясь правом оперативного диспетчерского управления работой ВЦ своих абонентов, сеть ОГИВЦ может при необходимости перепоручать функции аварийного резервирования и снятия пиковых нагрузок тем абонентам, ВЦ которых в данный момент мало загружены.

Указанные меры резко повышают надежность работы и живучесть всей системы в целом, а также уменьшают необходимую суммарную мощность парка ЭВМ, работающих во всех звеньях ОГАС (как в ОГИВЦ, так и у абонентов). Кроме того, накапливая дубликаты математического обеспечения (а, значит, и его изменений) своих абонентов, сеть ОГИВЦ может взять на себя функцию распространения по соответствующим ВЦ всех вновь разработанных программ независимо от ведомственной принадлежности разработчиков.

Как бы важны и значительны ни были перечисленные до сих пор функции сети ОГИВЦ, самой главной и решающей ее функцией *является организация и диспетчирование решения межведомственных задач планирования и текущего управления*. Одними из важнейших задач есть задачи составления и корректирования государственных планов. Главным абонентом, организующим работу, является ВЦ Госплана. По его сигналам ОГАС организует работу с союзными и республиканскими министерствами (через республиканские госпланы или непосредственно) по укрупнению и разукрупнению групп продукции и подсчету соответствующих нормативов, используя методы, подобные тем, которые описаны в предыдущем параграфе. Вообще через ОГАС замыкается вся система моделей задач управления макроэкономического уровня, описанная в параграфе 6 настоящей главы.

Важнейшей задачей, которую должна решать ОГАС, является *полная взаимоувязка* всех детализированных планов, хранящихся в ВЦ низовых уровней. Краткосрочные планы при этом согласуются по датам (а иногда и часам) взаимных поставок, размеров партий и т. д. В этом случае сеть ОГИВЦ должна решить многие задачи, которые призваны решать территориальные органы госматтехснаба. Учитывая объем и важность этих задач, территориальные ВЦ госматтехснаба целесообразно максимально приблизить к ОГИВЦ, а, может быть, даже сделать их частью ОГИВЦ.

Система *взаимоувязанных детализированных* планов, хранящихся в памяти *всей совокупности* ЭВМ ОГАС в *динамическом виде*, т. е. вместе с программами, которые *восстанавливают* взаимоувя-

занность *при любых изменениях* или случайных отклонениях его выполнения на отдельных объектах. Это позволяет реализовать широкую демократизацию системы планирования. Суть демократизации состоит в том, что любое предложение по улучшению плана, исходящее от любого абонента ОГАС, вплоть до низового звена, может быть проверено с точки зрения его влияния на сроки достижения конечных целей, стоящих перед обществом. И если предложение сокращает эти сроки, *имеется инструмент оперативного изменения плана*, при котором *не теряется его взаимоувязанность*.

В полном объеме ОГАС должна включать в себя порядка 100 ОГИВЦ, 300—400 абонентов первого уровня (подключенных к сети ОГИВЦ широкополосными каналами), 15000 абонентов второго уровня (ВЦ, подключенных к сети ОГИВЦ узкополосными каналами) и 5000 абонентов, не имеющих собственных ВЦ и подключенных к сети ОГИВЦ узкополосными каналами через информационно-диспетчерские пункты. Общая мощность всех ЭВМ, объединяемых ОГАС, должна составлять многие миллиарды операций в секунду.

Грандиозность задачи создания ОГАС обусловливается не только масштабами проблем чисто технического характера (ВЦ и каналы связи). Не меньший (а, пожалуй, даже значительно больший) объем работ предстоит выполнить в области создания специального математического обеспечения и, что особенно трудно,— в создании информационной базы системы, а также комплекса мероприятий экономического и организационного характера.

Поэтому особенно важное значение приобретает проблема концентрации сил и четкого управления разработками. Сеть ОГИВЦ (центральное звено ОГАС) и абонентов ОГАС должны создаваться параллельно, но в тесной взаимоувязке между собой. Такая взаимоувязка возможна лишь в том случае, когда помимо главного конструктора сети ОГИВЦ и генерального конструктора всей системы (ОГАС) имеется относительно небольшое количество (порядка 30) главных конструкторов, ответственных за создание типовых проектов абонентских ВЦ разных профилей, включая проектирование не только технических комплексов, но и типовых схем информационной базы, систем ее обновления и пополнения типового математического обеспечения и соответствующей системы мероприятий экономического и организационного характера.

Задачу создания столь большой системы, как ОГАС, целесообразно разбить на этапы с тем, чтобы каждый этап достигал определенной цели, позволяющей многократно окупить произведенные на этом этапе затраты. Главная цель первого этапа создания ОГАС — улучшение загрузки существующих ведомственных ВЦ за счет простейших средств межведомственной диспетчеризации их работы. Эта диспетчеризация должна осуществ-

ляться специальной службой территориальных диспетчеров, находящейся под общегосударственным управлением. Ее основу составляют первичные информационно-диспетчерские пункты (ИДП), создаваемые в областях, краях и республиках без областного деления. На первом этапе каждый такой ИДП может быть размещен в одной-двух комнатах, имеющих устойчивую телефонно-телеграфную связь со всеми ВЦ предприятий и организаций (независимо от их ведомственной принадлежности), расположенными на территории данной области (соответственно края или республики). С момента создания ИДП все ведомственные ВЦ становятся его *абонентами*.

ИДП наделяется правом получать от своих абонентов оперативную, исчерпывающую информацию о состоянии технических средств (ЭВМ и периферийного оборудования) и о планах их загрузки на определенный период (на неделю или месяц) вперед. Если тот или иной абонент имеет излишки машинного времени, они поступают в распоряжение дежурного диспетчера ИДП для распределения между теми, кто в соответствующий момент имеет недостаток машинного времени. Разумеется, такое распределение не должно нарушать планы абонентов относительно своих собственных задач.

Осуществление подобной диспетчеризации (на этом этапе еще достаточно примитивной) позволит прежде всего увеличить среднюю загрузку ЭВМ и тем самым сэкономить многие сотни миллионов рублей капитальных вложений на создание новых ВЦ и расширение старых. Организации и предприятия, не имеющие пока собственной вычислительной техники и даже не собирающиеся ее иметь вообще, получают возможность начать создание собственных АСУ. Наконец, детальный учет состояния оборудования дает возможность анализировать причины его плохого использования и оперативно вырабатывать необходимые меры для устранения этих причин.

На этом же этапе ИДП должны начать составление (а в последующем,— постоянное обновление) каталога алгоритмов, программ и информационных массивов на машинных носителях, имеющихся у всех их абонентов. Тем самым подводится организационная основа для систематической работы по унификации систем математического обеспечения и форм представления информации в АСУ различных ведомств, что является необходимой предпосылкой для автоматизации решения межведомственных задач.

Помимо первичных ИДП, оперативно координирующих работу ВЦ предприятий и организаций неправительственного уровня, должны быть созданы республиканские ИДП и головной (общесоюзный) ИДП, в задачу которых входит координация работы

главных вычислительных центров министерств и ведомств, а также управление всей сетью ИДП.

Головной (общесоюзный) ИДП и республиканские ИДП должны быть обеспечены устойчивой связью с подведомственными им первичными ИДП. Такая связь позволяет расширить возможности диспетчеризации до масштабов республик и даже всей страны. По мере унификации форм представления информации, выдаваемой АСУ различных ведомств по запросам ИДП на машинных носителях, появляется возможность оперативного решения сначала простейших справочно-информационных, а затем и более сложных межведомственных задач. Даже при отсутствии разветвленной сети передачи данных по линиям связи сеть ИДП может организовать достаточно оперативно сбор необходимой информации (на машинных носителях) от всех своих абонентов и пересылку этой информации на специально назначаемые каждый раз для этой цели ВЦ (концентраторы) для ее окончательной обработки в требуемом виде.

При увеличении числа абонентов и усложнения задач, решаемых сетью ИДП, появляется необходимость в снабжении ИДП собственными ЭВМ, способными автоматизировать решение задач диспетчеризации, ведение информационных каталогов и концентрацию информации. Кроме того, ИДП являются естественными заказчиками и будущими владельцами мощных ВЦ коллективного пользования, способных с помощью специальных удаленных терминалов обслуживать абонентов, не имеющих собственных вычислительных средств, либо имеющих их в недостаточном количестве.

По мере расширения собственной вычислительной базы и решения указанных выше задач сеть ИДП превращается в сеть ОГИВЦ, а вся система ВЦ страны — в Единую Государственную Систему вычислительных центров (ЕГСВЦ). Одновременно должно происходить развертывание Общегосударственной Системы передачи данных (ОГСПД), обеспечивающей автоматический обмен информации между любыми вычислительными центрами ЕГСВЦ как непосредственно, так и транзитом через соответствующие ОГИВЦ. Само собой разумеется, что хотя ОГИВЦ и могут выполнять роль центров коммутации сообщений, ОГСПД может иметь и другие (свои собственные) центры коммутации сообщений.

В полностью законченном виде ЕГСВЦ должна работать под управлением единой операционной системы, позволяющей (в соответствии с пропускными возможностями каналов связи) рассматривать все вычислительные мощности страны, как единую систему коллективного пользования с распределенными банками данных. В системе должны быть средства, позволяющие быстро

(на основе первичных информационных массивов предприятий и организаций) создавать любые вторичные массивы, необходимые для работы любых звеньев управления. Все это требует весьма сложного специального математического обеспечения.

Для превращения ЕГСВЦ в ОГАС необходимо к этому математическому обеспечению добавить не менее сложное математическое обеспечение для решения межведомственных задач на различных уровнях управления, позволяющих автоматизировать решение задач управления *горизонтальными связями* (предприятие — предприятие, Министерство — Министерство и т. п.). Для того чтобы представить себе сложность этих задач, достаточно сказать, что по имеющимся оценкам, только задача окончательного согласования (по самой подробной номенклатуре и срокам) планов производства и материально-технического снабжения на уровне отдельных предприятий требует у нас сегодня выполнения не менее 10^{15} арифметических операций в год. В то же время хорошо известно, что неполное или несвоевременное решение указанной задачи *синхронизации* работы предприятий (независимо от их ведомственной принадлежности) приводит к большим потерям, устранение которых может значительно (в 1,5—2 раза) повысить темпы экономического развития.

ОСНОВНАЯ ЛИТЕРАТУРА

1. Автоматизированные системы управления предприятиями. Киев, «Наукова думка», 1966.
2. Алгоритмический язык АЛГОЛ-60. Пересмотренное сообщение. М., «Мир», 1965.
3. Аллен Р. Математическая экономия. М., Изд. иностр. лит. 1963.
4. Акофф Р., Сасиени М. Основы исследования операций. М., «Мир», 1971.
5. Букан Дж., Кенигберг Э. Научное управление запасами. М., «Наука», 1967.
6. Бусленко Н. П. Математическое моделирование производственных процессов. М., «Наука», 1969.
7. Бусленко Н. П. Моделирование сложных систем. М., «Наука», 1968.
8. Вычислительная система *IBM*/360. Принципы работы. М., «Советское радио», 1969.
9. Глушков В. М. Введение в кибернетику. Киев, Изд-во АН УССР, 1964.
10. Глушков В. М. и др. Обработка информационных массивов в автоматизированных системах управления. Киев, «Наукова думка», 1970.
11. Данциг Дж. Линейное программирование, его обобщения и применения. М., «Прогресс», 1966.
12. Джермейн К. Программирование на *IBM*/360. М., «Мир», 1971.
13. Длин А. М. Математическая статистика в технике. М., «Наука», 1958.
14. Китов А. И. Программирование экономических и управленческих задач. М., «Советское радио», 1971.
15. Кокс Д. Р., Смит У. Л. Теория очередей. М., «Мир», 1966.
16. Кофман А., Дебазей Г. Сетевые методы планирования и их применение. М., «Прогресс», 1968.
17. Ледли Р. С. Программирование и использование цифровых вычислительных машин. М., «Мир», 1966.
18. Льюс Р. Л., Райфа Х. Игры и решения. М., Изд. иностр. лит. 1961.
19. Мидоу Ч. Анализ информационно-поисковых систем. Введение для программистов. М., «Мир», 1970.
20. Опыт разработки, внедрения и функционирования системы «Львов».— «Механизация и автоматизация управления», 1969, № 3.
21. Пшеничный Б. Н. Необходимые условия экстремума. М., «Наука», 1969.
22. Сэксон Дж. Кобол. М., «Статистика», 1970.
23. Уайлд Д. Дж. Методы поиска экстремума. М., «Наука», 1967.

24. Феллер В. Введение в теорию вероятностей и ее приложения. М., «Мир», 1964.
25. Фишер Ф. П., Суиндл Дж. Ф. Система программирования. М. «Статистика», 1971.
26. Хедли Дж. Нелинейное и динамическое программирование. М., «Мир», 1967.
27. Шкурба В. В. и др. Задачи календарного планирования и методы их решения, Киев, «Наукова думка», 1966.

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

28. Автоматизированные системы управления производством. Труды ИИЭР (Тематический выпуск) Т. 58, М., «Мир», 1970, № 1.
29. Аганбегян А. Г., Гранберг А. Г. Экономико-математический анализ межотраслевого баланса СССР. М., «Мысль», 1968.
30. Алгоритмический язык АЛГОЛ-68. — «Кибернетика», 1969, № 6, 1970, № 1.
31. Арис Р. Дискретное динамическое программирование. М., «Мир», 1969.
32. Бартлетт М. С. Введение в теорию случайных процессов. М., изд. иностр. лит. 1958.
33. Беллман Р. Динамическое программирование. М., Изд. иностр. лит. 1960.
34. Беллман Р., Калаба Р. Динамическое программирование и современная теория управления. М., «Наука», 1969.
35. Бертэн Ж. и др. Работа ЭВМ с разделением времени. М., «Наука», 1970.
36. Важоньи А. Научное программирование в промышленности и торговле. М., Изд. иностр. лит., 1963.
37. Вычислительные машины. Труды ИИЭР. (Тематический выпуск) Т. 54, М., «Мир», 1966, № 12.
38. Гвишиани Д. М. Организация и управление. М., «Наука», 1971.
39. Гейл Д. Теория линейных экономических моделей. М., Изд. иностр. лит., 1963.
40. Глушков В. М. Автоматизированные системы управления. — В сб.: «Труды Всесоюзной конференции по программированию». Вып. 1. Новосибирск, «Наука». 1970.
41. Глушков В. М. О прогнозировании на основе экспертных оценок. Науковедение. Прогнозирование. Информатика. Киев, «Наукова думка», 1970.
42. Глушков В. М. Основные принципы построения автоматизированных систем управления.— В сб.: «Кибернетика и вычислительная техника», Вып. 12, Киев, «Наукова думка», 1971.
43. Глушков В. М. и др. АНАЛИТИК (алгоритмический язык для описания вычислительных процессов с использованием аналитических преобразований).— «Кибернетика», 1971, № 3.
44. Глушков В. М. и др. Вопросы развития структуры ЦВМ в связи с системами их математического обеспечения.— «Кибернетика», 1967, № 5.
45. Глушков В. М. и др. Входной язык вычислительной машины для инженерных расчетов.— «Кибернетика», 1965, № 1.
46. Глушков В. М. и др. СЛЭНГ — система программирования для моделирования дискретных систем. Киев, Изд-во АН УССР, 1969.
47. Глушков В. М. и др. Человек и вычислительная техника. Киев, «Наукова думка», 1971.

48. Гнеденко Б. В. Курс теории вероятностей. М., «Наука», 1965.
49. Гнеденко Б. В., Коваленко И. Н. Введение в теорию массового обслуживания. М., «Наука», 1966.
50. Гуд Г. Х., Макол Р. Э. Системотехника: введение в проектирование больших систем. М., «Советское радио», 1962.
51. Дал У. и др. СИМУЛА-67 универсальный язык программирования М., «Мир», 1969.
52. Данскин Дж. М. Теория максимина. М., «Советское радио», 1970.
53. Джонсон Р. и др. Системы и руководство. М., «Советское радио», 1971.
54. Диксон Дж. Проектирование систем: изобретательство, анализ и принятие решений. М., «Мир», 1969.
55. Ермольев Ю. М., Мельник И. М. Экстремальные задачи на графах. Киев, «Наукова думка, 1968.
56. Зойтендейк Г. Методы возможных направлений. М., Изд. иностр. лит., 1963.
57. Информация. М., «Мир», 1968.
58. Информационно-поисковая система БИТ. Под ред. А. А. Стогния. Киев, «Наукова думка», 1968.
59. Карлин С. и др. Математические методы в теории игр. Программировании в экономике. М., «Мир», 1964.
60. Квейд Э. Анализ сложных систем. М., «Советское радио», 1969.
61. Корбут А. А., Финкельштейн Ю. Ю. Дискретное программирование. М., «Наука», 1969.
62. Кофман А. Методы и модели исследования операций. М., «Мир», 1966.
63. Кофман А., Крюон Р. Массовое обслуживание: теория и приложения. М., «Мир», 1965.
64. Кофман А., Фор Р. Займемся исследованием операций. М., «Мир», 1966.
65. Лавров С. С., Гончарова Л. И. Автоматическая обработка данных. «Наука», 1971.
66. Макроэкономические модели планирования и прогнозирования. М., «Статистика», 1970.
67. Мили Г. и др. Функциональная структура *OS*/360. М., «Советское радио», 1971.
68. Михалевич В. С. Последовательные алгоритмы оптимизации и их применение.— «Кибернетики», 1965, № 1, 2.
69. Моделирование процессов производства и управления. Под ред. А. Г. Аганбегяна. Новосибирск, «Наука», 1966.
70. Моисеев Н. Н. Численные методы в теории оптимальных систем. М., «Наука», 1971.
71. Моррис У. Т. Наука об управлении. Байесовский подход М., «Мир», 1971.
72. Мот Ж. Статистические предвидения и решения на предприятии М., «Прогресс», 1966.
73. Научно-техническое прогнозирование для промышленности и правительственных учреждений. Под ред. Дж. Брайта. М., «Прогресс», 1972.
74. Нейман Дж., Моргенштерн О. Теория игр и экономическое поведение. М., «Наука», 1970.
75. Новое в теории и практике управления в США. М., «Прогресс», 1971.
76. Оптнер С. Л. Системный анализ для решения деловых и промышленных проблем. М., «Советское радио», 1969.
77. Организационные принципы и технический комплекс АСУП.— В сб.: «Кибернетика и вычислительная техника». Вып. 12. Киев, «Наукова думка», 1971.
78. Оуэн Г. Теория игр. М., «Мир», 1971.

79. Поль де Брюйн. Подготовка кадров для управления предприятиями (цели, программы, методы). М., «Прогресс», 1968.
80. Понтрягин Л. С. и др. Математическая теория оптимальных процессов. М., Физматгиз, 1961.
81. Поспелов Д. А. Введение в теорию вычислительных систем. М., «Советское радио», 1972.
82. Поспелов Г. С. Ленинские принципы управления в эпоху научно-технической революции.— «Известия АН СССР. Техническая кибернетика», 1970, № 2.
83. Райветт П., Акофф Р. Л. Исследование операций. Пособие для административно-управленческих работников. М., «Мир». 1966.
84. Сергієнко І. В. Операційний метод дослідження ЭОМ і систем.— «Доповіді АН УРСР. Серія А», 1969, № 4.
85. Скурихин В. И. Научные основы автоматизированных систем управления. Управляющие машины и системы. Киев, Изд-во АН УССР, 1970.
86. Современные методы внутрифирменного управления в капиталистических странах. М., «Прогресс», 1971.
87. Современные методы управления. Под ред. Б. З. Мильнера. М., «Наука», 1971.
88. Современное программирование. Мультипрограммирование и разделение времени. М., «Мир», 1970.
89. Справочник по системотехнике. Под ред. Р. Макола. М., «Советское радио», 1970.
90. Старр М. Управление производством. М., «Прогресс», 1968.
91. Стогний А. А., Зайцев Н. Г. Автоматизированные информационно-справочные системы, их назначение, характеристики и основные требования к ним.— «Кибернетика», 1969, № 4.
92. Уилсон А., Уилсон М. Информация, вычислительные машины и проектирование систем. М., «Мир», 1968.
93. Универсальный язык программирования *PL/I*. М., «Мир», 1968.
94. Фельс Э., Тинтнер Г. Методы экономических исследований. М., «Прогресс», 1971.
95. Флорес А. Программное обеспечение. М., «Мир», 1971.
96. Форрестер Дж. Основы кибернетики предприятия (индустриальная динамика). М., «Прогресс», 1971.
97. Ханика Ф. Д. Новые идеи в области управления. Руководство для управляющих. М., «Прогресс», 1969.
98. Хемминг Р. В. Численные методы для научных работников и инженеров. М. «Наука» 1968.
99. Хитч У. Руководство обороной. Основы принятия решений. М., «Советское радио», 1968.
100. Цянь Сюэ Сэнь. Техническая кибернетика. М., Изд. иностр. лит. 1956.
101. Черчмен У. и др. Введение в исследование операций. М., «Наука», 1968.
102. Янч Эрих. Прогнозирование научно-технического прогресса. М., «Прогресс», 1970.
103. Ланкастер К. Математическая экономика. М., «Советское радио», 1972.
104. Ackley G. Macroeconomic theory. New York, Macmillan, 1961.
105. Ackoff R. L. Aconcept of corporate planning. New York, Willy, 1970.
106. Beveriolge G., Scheshter R. S. Optimisation: theory and partice. New York, McGraw-Hill, 1970.
107. Davis M. D. Game theory. A nontechnical introduction. New York, Basic Books, 1970.

108. McDonough A. M. Information Economics and Management Systems. Englewood cliffs, N. 1. Prentice — Hall, 1963.
109. Executive programs and operating systems. Ed. By G. Cuttle, P. B. Robinson. London, Macdonald, 1970.
110. Flores I. Computer organization. Englewood Cliffs, Prentice — Hall, 1969.
111. Hall A. D. A Methodology for system engineering. Princeton, N. Y., Van Nostrand, 1962.
112. Hopgood F. R. A. Compiling technigues. London, Macdonald, 1969.
113. Lancaster F. M. Information retrieval systems: characteristics, tisting and evaluation. New York, Wiley, 1968.
114. Kaufmann A. Des points et des fleches... le theorie des graphes. Paris, Dunod, 1968.
115. Kaufmann A. Gaphs, dynamic programming and finite games. New York — London, Academic Press, 1967.
116. Kaufmann A. The science of decision-making. An introduction to praxedogy. Weideufeld and Nicolson, London, 1968.
117. Knuth D. E. The art of computer programming. Vol. 1. Fundamental algorithmes. Menlo Cark, lalif., Addison-Wesley, 1968.
118. Mecarovic M. D, Macko D, Takanara. Theory of uievarcnical multilevel systems. New York, Academic Press, 1970.
119. Pierre D. A. Optimization theory with applications. New York, Wiley, 1969.
120. Salton G. Automatic information organization and retrieval. New York, McGraw-Hill, 1968.
121. Stabley H. D. Logical programming with SYSTEM/360. New York, Wiley, 1970.
122. Wagner H. M. Principles of operations research with applications to managerical decisions. Englewood cliffs, Prentice-Hall, 1969.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Абонент 294, 295, 302, 304, 305
— второго уровня 293, 303
— главный 302
— низового звена 301
— первого уровня 293, 303
Аварийное резервирование 301
Автокод символический 42, 45—48, 53, 75
Автоматизация проектирования 74, 76, 78
— процессов логического вывода 224
Автоматизированная система документальная 212, 218, 220, 222
— — организационного управления (АСОУ) 188, 194—196, 198—202, 206, 207, 210, 211, 228, 229, 304, 305
— — проектирования 193, 194
— — справочно-информационная 194, 209, 211, 224, 244, 250, 307
— — управления предприятием (АСУП) 12—14, 32, 188, 196, 236—243, 245, 247, 249, 304, 306, 307
— — — технологическим процессом (АСУТП) 188—191
— — фактографическая 212—215, 217, 218, 222
Автоматическая система управления 189
Автоматическое группирование документов 223
— реферирование 222
Автомат читающий 39
Адресация 22
— косвенная 22
— непосредственная 22
— элементов массива 49
Адрес абсолютный 41
— нулевого ранга 45
— первого ранга 45
— составной 44
Алфавитно-цифровое печатающее устройство (АЦПУ) 36, 203, 214, 215
Альтернатива 158, 159, 272
Амортизация 232, 234, 235, 237
— косвенная 235
— прямая 235
Арифметическое выражение 57—61
— устройство (АУ) 19

Байт 15, 37
Банк данных 214, 286, 305
— идей 271
— новых технологий 285, 286
Библиотека стандартных программ (БСП) 47—48, 50, 61, 205
Бит 15, 35, 75
Блок головок 25
— памяти 24
— программы 62
Большие интегральные схемы (БИС) 76
Буфер 71, 72
Буферное оперативное запоминающее устройство (буферное ОЗУ) 37
Буферный фонд 72

Важность цели 156
Весовой коэффициент 166
Внешняя память ЭВМ 78
— среда 147
Выборка бесповторная 174, 175, 177

— повторная 174
Выборочная дисперсия 173—175
— совокупность 173
Вычислительный центр (ВЦ) 291, 292, 296, 297, 303
— — абонента 301, 302
— — ведомственный 294, 303, 304
— — главный 242
— — горисполкома 295
— — кустовой 295
— — облисполкома 295
— — общесоюзной организации 295
— — предприятия связи 295

Генеральная совокупность 173—175, 178
Генеральное среднее 173
Генератор программ 78
Гистограмма 123

Децентрализация управления 210, 211
Диалог человек—машина 29, 75, 78, 242
Динамическое программирование 102, 103, 106—108, 119, 252, 306
— распределение памяти 63
Диспетчер 65, 207
Диспетчеризация сообщений 291, 303—305
Дисплей 193
Дисциплина обслуживания 121, 122, 124, 215, 216
Документооборот 202, 208, 213
Доход национальный 273, 274, 278, 281
Дуаль-карта 34, 199, 210

Загрузка 68
Задание 66, 67, 73
Задача 66, 74
— анализа 131—133
— о коммивояжере 102—106, 119
— оптимизации 81
— оптимального планирования 88 96, 102
— синтеза 131, 136, 140
— транспортная 95, 96
— экстраполяции 163
Задачи управления объективно необходимые 10, 13
— — рутинные 195, 197, 237
Запас 258, 260, 262, 263
Запись 18, 50, 52
Запоминающее устройство (ЗУ) 19, 39
— — ассоциативное 23
— — внешнее 23, 25, 71
— — голографическое 28
— — долговременное (ДЗУ) 28
— — оперативное (ОЗУ) 20—24, 26, 29, 34, 35, 40, 62, 63, 68, 69, 71, 190
— — сверхоперативное (СОЗУ) 21
Зарплата 211, 235, 236, 238, 258, 287—289
Защита памяти 66

Игра конечная 179
— парная с нулевой суммой 179, 181
Идентификатор 54, 55, 58, 59, 62
Износ моральный 232—234
— физический 232—234
Импорт 275, 279
Индексирование 218, 219
Индекс-регистр 44, 45, 49
Интервал планирования 230
Интерполяционный полином Лагранжа 164
Интерполяция 164
Информационная база 196, 200, 206—209, 218, 226, 244, 298, 299
Информационно-диспетчерский пункт (ИДП) 295, 304, 305
Информационный барьер в развитии экономики 10
— — второй 12
— — первый 11

Канал 65
— мультиплексный 65
— связи 209, 291, 292, 305
— селекторный 65
Карта с краевой перфорацией 212
Каталог 69, 70
Качество управления 142
Квартиль 155, 158
— верхний 155
— нижний 155
Кей 21
Классификатор признаков 212, 213, 242, 298
Ключевое слово 219, 220
Команда ЭВМ 40
Коммуникатор 67, 72
Коммутатор 20
Коммутация сообщений 291
Компайлер 53
Комфорт материальный 270
— психологический 270
Конечный продукт 269
Контроль качества 175
Контрольник 30

Конфликтная ситуация 179, 185
Коэффициент важности цели 156, 157
— корреляции поисковых образов 219, 220
— полного использования ресурсов 246
— рабочего использования ресурсов 246
— технологический 275—277, 286
Критерий аддитивный 106
— наименьшей стоимости 95
— оптимальности управления 143 161
Критический путь 112
— ресурс 116, 117, 119

Лингвистический анализ 218
Линейное программирование 88—95, 181—184, 248, 252, 282, 304
Литерал 55

Магнитный барабан 23, 24, 65
— диск 25, 26, 52, 65, 66, 71, 75, 196, 215, 217, 293
— карта 37
— лента 37, 50, 52, 65—67, 71, 196, 215, 217, 293

Макрооператор 46, 47, 194, 206, 207, 214
Максимин 180, 182
Мастер-файл 203
Массив 49—51, 196, 203, 214
— изменений 52, 204, 301
— кадровый 205
— нормативов 204, 205
— нормативный 50, 51
— плановый 50
— рабочий 205—208
Математическое обеспечение общее 74, 77, 304
— — общесистемное 74
— — системное 205, 206
— — специальное 74, 193, 194, 267, 303, 306
Материально-техническое снабжение 195, 207, 258, 264, 267
Матрица игры 179
— планов перевозок 96—100
— стоимостей перевозок 96, 98, 99, 101
Медиана распределения 155, 156
Менеджер ресурсов 68, 69
Мера неопределенности прогноза 155
Метка 59
Метод градиентного спуска 85—87
— множителей Лагранжа 82, 83
— наименьших квадратов 165, 166, 170, 172
— наискорейшего спуска 85—87
— сглаживания 171
— симплексных преобразований 90—95, 102
— стохастического программирования 88
Микрофильм 224
Микрофиша 224
Минимакс 180, 182
Моделирование математическое 122—124, 290
Модель динамическая 202, 203, 242
— информационная 203, 226, 298
— Леонтьева 275—279
— макроэкономическая 129, 275, 277
Модуль загрузки 68
Мультипрограммирование 64, 66, 74, 75
Мультипроцессорная обработка 76, 77
— ЭВМ 76

Набор данных 66, 67, 69, 72, 73
— команд ЭВМ 41
— признаков 212
Нормативы 228, 231, 235, 299

Обратная связь 140, 229
Общегосударственная автоматизированная система управления экономикой (ОГАС) 196, 243, 290—306
Общегосударственный информационно-вычислительный центр (ОГИВЦ) 291—293, 298—303, 305
Объект управления 188, 194—197, 202, 206
Оперативное изменение плана 303
Оператор безусловного перехода 59
— внесения изменений 51
— выборки экстремальных записей 51
— вывода 52
— вызова процедуры 61
— вычислительный 51
— группировки 51
— групповой выборки 50
— перезаписи массивов 52
— присваивания 58
— системный 67, 72
— слияния массивов 51
— сортировки массива 51
— составной 59, 62
— таблиц 51

— условный 59
— цикла 59, 60
Операции арифметические 19
— логические 19, 45, 58
— отношения 58
— сравнения 43
— условного перехода 59
— чтения и записи 43
Операционная система 64, 65, 70—77 204—208, 267, 292, 305
— — реального времени 74
Операционное устройство 19, 41
Опрос социологический 239
Оптимизация плана 247, 248
— распределения ресурсов 115—117
Оптимальный регулятор 189
— ритм производства 268

Пакетная обработка информации 66. 73
Параметр 189, 192, 193
— входной 130, 138
— выходной 138
— регулируемый 189—191
— управляющий 130, 131, 160
— фактический 61
— формальный 61
Параметризация 148, 153
Параметры обобщенной динамической системы 162
— булевы 148, 149, 152, 159, 162
— дискретные 148, 149
— качественные 148, 149
— непрерывные 148, 149
Пароль 72
Первичный документ 32, 33
— словарь 219
— учет 33
Передаточная функция 138, 139
Период оптимального обновления запаса 261
Перо световое 40
Перфокарта 31—35, 52
Перфолента 35, 36, 52
Перфоярлык 39
План 267, 302
— директивный 89, 244
— долгосрочный 249
— календарный 241, 264
— краткосрочный 241, 251
— материально-технического снабжения 249
— объемно-календарный 248—251, 264
— оперативный 241, 239, 240, 241
— оперативно-календарный 241, 250
— оптимальный 95, 101, 102
— развития объекта 239
Планирование 241, 243, 245, 291
— долгосрочное 231, 239—244
— краткосрочное 213, 231, 239, 240, 243, 244, 273, 274, 278
— на основе сетевого графика 115
— непрерывное 240
— объемно-календарное 197, 207, 240, 250
— оперативное 231, 239, 240, 241
— оперативно-календарное 197, 200, 207, 241, 264, 268
— оптимальное 244, 253
— перспективное 231, 239, 243, 273, 275, 277, 278, 285, 289, 290
— предварительное 113
Планировщик заданий 67, 78
Подпрограмма 61, 62
Поиск адресный 21, 22, 24
— ассоциативный 23
— дихотомический 218
— наборов данных 69
Поисковый образ 218, 224
Поколение ЭВМ 74
— второе 75
— первое 74
— третье 64, 65, 73, 75, 76
— четвертое 54, 76—79
Поколения данных 73
Последовательная частичная оптимизация 248
Поток требований 120
— — простейший 119
— — пуассоновский 119, 124
— — стационарный 119, 124
Потребление непроизводственное 269, 275
— производственное 275
Правила доминирования 106
— предпочтения 252
— разрешающие 106
Предварительная подготовка данных 30
— система целей 272
Преобразование Лапласа 136—138
— Фурье 171—172
Прибыль 89, 232
Признак 212, 213, 217
— качественный 212
— количественный 212
Принцип автоматизации документооборота 202
— ввода изменений 209
— динамического представления взаимосвязанных планов 242

— единой информационной базы 203, 209
— комплексного подхода 198, 202
— комплексности задач и рабочих программ 207, 209
— максимальной разумной типизации проектных решений 200
— максимума Понтрягина 142, 143, 152
— минимизации ввода и вывода информации 208, 209
— непрерывного развития системы 201
— непрерывности и динамичности планирования 241
— новых задач 197
— оптимальности Беллмана 106—107
— совмещения подготовки документов первичного материального учета и первичных финансовых документов с приготовлением соответствующих машинных документов 210
— согласованности пропускных способностей отдельных частей системы 210
— специализации операционных систем 207
— функциональной избыточности 201
— экономного и эффективного использования связи 291
Производительность внешних ЗУ 71
— ЭВМ 65
Прогнозирование 233, 273, 278, 288
— возможных путей достижения цели 272
— демографическое 277
— на основе экспертных оценок 306
— научно-технического прогресса 9, 152, 286, 289, 307
— спроса 195, 229, 243, 244, 253
— сроков достижения цели 272
— технологических нормативов 285
— уровня цен 289
Программа 40—46, 53, 58—68, 71, 73, 192, 193, 196
— перекодирующая 298
Программирование 17, 41—45, 46—49, 52, 53, 75, 76, 306, 307
Программное управление 191, 193, 287
Программно-управляемая технология 192
Программно-управляемый испытательный стенд 192
Пространство понятий 219
Процессор коммуникационный 65, 75, 78
— периферийный 78
— управляющий 77
— центральный 75, 76, 78, 210, 215, 216
Пульт системный 67
— экранный 40, 214

Разделение времени 73, 76, 306
Размерность задачи линейного программирования 90
— — оптимизации 87
Регистр 20, 21, 44, 45, 49
— адреса 20, 21
— числа 20
Регулятор 189—191
Редактор связей 68
Резерв 254, 279, 280
— времени 112
Результат игры 179
Ресурс 241, 278, 283, 284

Световой карандаш 194
Связка 57
Сглаживание помех 171
Секретность информации 72
Семантический анализ 221
Сетевой график 109, 113, 115, 116, 119, 152, 198, 241, 250, 268, 299
— — двухцелевой 114, 115
— — многоцелевой 113
— — одноцелевой 109, 113
Система автоматизации программирования 206
— автономная 142
— взаимоувязанных и детализированных планов 302
— государственного страхования риска 286
— документально-фактографическая 224
— информационно-справочная 244
— кодов единая 297, 298
— команд одноадресная 43
— — трехадресная 44
— массового обслуживания 119, 122—124, 129, 215
— неавтономная 145
— открытая 147
— прерываний 65, 74
— сбора и передачи данных 196
— справочно-информационная 194, 209, 211, 224, 250
— стимулирования 109, 238, 239

— управления запасами 254
— интегрированная 188
— — техническим прогрессом 284
— ценообразования 231
Системный анализ 146, 150, 152, 208, 239, 243, 286
Слово машинное 16
— — переменной длины 16
— — фиксированной длины 16, 17
Сменные задания 240
Снятие пиковых нагрузок 301
Событие конечное 110, 111
— начальное 110, 112
— промежуточное 110
Списочные структуры 57, 223
Стандартизация систем команд 78
— — прерывания 78
Стандартная подпрограмма 46, 48, 49, 52
Степень неопределенности прогноза 155, 158
Стоимость минимальная 234, 236, 237
— фактическая 234, 236, 237
Стратегия 179, 180, 185, 187
— максиминная 180, 182
— минимаксная 180, 182
— оптимальная 180—184, 186, 187
— полезная 181, 184
— смешанная 180, 182, 185—187
— чистая 180, 182, 184, 186
Страховой запас 258—260
Структуризация систем 147, 148, 152
Сумматор 43
Супервизор 67, 70—71
Счетно-аналитические машины 197
Считывающее устройство стилизированной записи 39

Таблица содержания тома 70
Тезаурус 221
Телетайп 29, 35, 36, 210, 214
Теория игр 179
Терминал 73, 210, 214, 305
Технологический маршрут 227, 264
Типы величин 55, 56
Том 69—72
Транслятор 42, 53, 64, 68, 71, 75, 77

Управление 130
— в булевой модели 158, 160
— горизонтальными связями 306
— запасами 131, 141, 304
— на основе сетевого графика 115
Уравнение баланса внешней торговли 276, 280
— трудоемкости 279
Уравнения материальных балансов 275, 276
Условие оптимальности 143
Устойчивость динамической системы 133, 140, 141
— — — асимптотическая 133—136
— — — локально-асимптотическая 133
Устройство ввода голосом в ЭВМ 78
— связи с объектом (УСО) 39, 190
— управления (УУ) 41

Файл 19
Флексорайтер 36, 210
Фондоемкость 280, 282
Фонд накопления 273, 274
— потребления 273, 274
Форматы слов ЭВМ 18

Характеристический полином 134, 136
Характрон 37
Ход личный 179
— случайный 179

Цели 269, 270, 286
— конечные 270
— промежуточные 271
— развития экономики 273
Целочисленное программирование 248
Цена 272, 287—289
— игры 181, 183, 184
— — верхняя 180—182, 185
— — нижняя 180—182, 185
Ценообразование макроэкономическое 278, 287—289
Централизация обработки информации 210
— управления 210, 211
Центр коммутации сообщений 305
Цикл итерационный 44
— с переадресацией 44

Часы электронные 74
Числа с плавающей запятой 17
— фиксированной длины 17

Шкала логарифмическая 169
— функциональная 169

Электронно-вычислительная машина (ЭВМ) 12—16, 20—27, 29—30, 124, 188, 190, 193, 197—199, 208, 222, 292, 296
— — — Днепр-1 190
— — — Днепр-2 190
— — — ИБМ-360 45

— — — М-6000 190
— — — МИР, 48, 54, 77
— — — МИР-2 46
— — — УМ-1 190
— — — управляющая 16, 190—193, 307
Эйдофор 37
Эквивалентное преобразование матрицы планов 98
Экономический объект 225, 226, 231, 244
Экспертная оценка 153—154, 158—160, 243, 246
Экспорт 275, 279
Экстраполяция 163, 164, 166—168, 171, 243
— квадратичная 164
— линейная 164, 166
— полиномиальная 164—167
Экстремум 81, 82, 88
— глобальный 87
— локальный 86—87, 89
Язык алгоритмический 52
— запросов 213
— машинно-ориентированный 53
— машинный 42
— моделирования 123, 124
— ориентированный на пользователя 53
— расширяемый 56
— АЛГОЛ-60 56—59, 62, 76, 304
— АЛГОЛ-68 55—56, 76, 77
— КОБОЛ-61 55, 57, 63, 76
— СЛЭНГ 124, 306
— СИМСКРИПТ 124
— СИМУЛА-67 56, 124, 306
— ФОРТРАН 54, 76
— *PL*-1 56, 76, 77
Ярлык 296, 297

СОДЕРЖАНИЕ

Стр.

Предисловие 5
Введение 8

Глава I

Электронные вычислительные машины

1. Принципы представления информации в ЭВМ 15
2. Принципы организации запоминания и выборки информации в ЭВМ 19
3. Принципы организации ввода и вывода информации в ЭВМ . . . 29
4. Принципы обработки информации в центральном процессоре . . 40
5. Стандартные подпрограммы и макрооператоры 46
6. Алгоритмические языки 52
7. Операционные системы 64
8. Перспективы развития ЭВМ 74

Глава II

Математические методы решения задач планирования и управления

1. Общие сведения о задачах оптимизации 80
2. Линейное программирование 88
3. Динамическое программирование 102
4. Сетевые графики 109
5. Теория массового обслуживания 119
6. Управление в динамических системах 129
7. Системный анализ 146
8. Методы экстраполяции 163
9. Методы статистического контроля 173
10. Игровые методы 179

Глава III

Автоматизированные системы управления

1. Общие сведения об автоматизированных системах управления и системах обработки данных 188
2. Основные принципы построения автоматизированных систем организационного управления 197
3. Автоматизированные справочно-информационные системы 211
4. Основные типы задач управления экономическими объектами . . 225
5. Управление запасами и оперативно-календарное планирование . 253
6. Задачи управления в макроэкономике 269
7. Общегосударственная автоматизированная система сбора и обработки экономической информации 290

Литература 307
Предметный указатель 312

Редактор издательства
инж. *Н. М. Корнильева*
Оформление художника *Е. В. Попова*
Художественный редактор
В. С. Шапошников
Технический редактор
Н. И. Старченкова
Корректор *М. И. Козиненко*

Подписано к печати 1.II. 1974 г. Формат бумаги 60×84¹/₁₆. Бумага типографская № 1. Объем: 20 физ. л.; 18,6 усл. л.; 20,6 уч.-изд. л. Тираж 40 000. Зак. № 4-96. БФ 34329. Цена 1 руб. 29 коп.

Издательство «Техніка», 252601, Киев, 1, ГСП, Пушкинская, 28.
Книжная фабрика им. М. В. Фрунзе республиканского производственного объединения «Полиграфкнига» Госкомиздата УССР, Харьков, Донец-Захаржевская, 6/8.