

А.В. Меньков
В.А. Острейковский

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

Учебник

*Рекомендовано
УМО вузов по университетскому
политехническому образованию
в качестве учебника для студентов
высших учебных заведений*

Москва
ОНИКС
2005

УДК 004.3
ББК 32.965.7
М51

Р е ц е н з е н т ы:

кафедра «Автоматизированные системы управления» Обнинского государственного технического университета атомной энергетики (зав. кафедрой д-р техн. наук, проф. *А.Н. Анохин*); д-р техн. наук, проф. *Л.И. Григорьев* (Московский государственный университет нефти и газа им. И.М. Губкина).

Меньков А.В.

М51 Теоретические основы автоматизированного управления/ А.В. Меньков, В.А. Острейковский. — Учебник для вузов. — М.: Издательство Оникс, 2005. — 640 с.: ил.

ISBN 5-488-00129-8

В учебнике согласно требованиям Государственных образовательных стандартов рассмотрены методология построения и основные понятия автоматизированных систем управления (АСУ), модели анализа и синтеза структуры АСУ, модели принятия решения. Приведены виды управления в АСУ и общие вопросы системного анализа. Большое внимание уделено различным типам и особенностям классификации АСУ, составу и функциям основных подсистем АСУ. Описано содержание математического, информационного, программного и других видов обеспечения функционирования АСУ.

Для студентов, обучающихся по специальности «Автоматизированные системы управления» и направлению «Информатика и вычислительная техника», а также студентов вузов кибернетического направления.

**УДК 004.3
ББК 32.965.7**

ISBN 5-488-00129-8

© А.В. Меньков, 2005
© В.А. Острейковский, 2005
© ООО «Издательство Оникс», 2005

ПРЕДИСЛОВИЕ

Вторая половина XX века характерна появлением машин и систем высокой конструктивной сложности, способных выполнять ответственные задачи. К началу 50-х годов прошлого века практически закончилась механизация и автоматизация физического труда человека. Более 99% объема работ выполнялось машинами. В это время человечество столкнулось с новой проблемой: физический труд человека стал выполняться машинами, а автоматизация умственной деятельности оператора (управленца) продолжала оставаться такой же, какой она была в течение последних 2000—2500 лет. Техника постоянно усложнялась, она выполняла все больше и больше функций, процессы управления становились все более сложными, а качество функционирования систем управления либо оставалось прежним, либо ухудшалось из-за усложнения производства. Решением этого противоречия стало естественное, объективное появление и становление во второй половине XX века новой научной дисциплины — теории автоматизированного управления как одного из разделов науки о закономерностях управления в технических, биологических, социальных и других классах систем.

Целями данного учебника являются:

1. Систематическое изложение теоретических вопросов автоматизированного управления.
2. Рассмотрение классификации и значительного числа принципов построения автоматизированных систем управления.
3. Ориентация на конкретные программы обучения студентов кибернетического профиля университетского политехнического образования.

В предисловии излагаются основные принципы, положенные в основу учебника, и методические особенности дисциплины «Теоретические основы автоматизированного управления».

Введение — вступление к основному тексту учебника. Его цель — познакомить студента с существом проблемы автоматизированного управления и краткой историей этой теории.

Первый раздел учебника — «Математические методы теории автоматизированного управления» — состоит из шести глав. Первая глава содержит основные понятия и определения автоматизированного управления. Автоматизированное управление рассматривается с позиций теории систем и системного анализа. Показана роль информации в процессе управления.

Во второй главе детально излагаются методология и принципы, закладываемые при проектировании автоматизированных систем управления. Приводится классификация систем автоматизированного управления и характеристика этапов (последовательности) их разработки.

Третья глава учебника посвящена применению системного анализа как основного научного метода построения автоматизированного управления.

Четвертая и пятая главы учебника содержат математический аппарат соответственно анализа и синтеза структуры автоматизированных систем управления (АСУ). При структурном анализе АСУ в четвертой главе подчеркивается универсальность математических методов и их необычайная важность для обеспечения функционирования структуры. В пятой главе рассмотрены математические модели формализации общей и частных задач синтеза оптимальной структуры АСУ.

Чрезвычайная роль человека-оператора при автоматизированном управлении определила содержание шестой главы учебника, в которой анализируется формализация поведения человека, на которого в человеко-машинных системах возлагается функция принятия решения. В этой главе изложены математический аппарат теории принятия решений.

Второй раздел учебника — «Прикладные вопросы автоматизированного управления» — состоит из трех глав и содержит описание видов управления и основных классов АСУ, обеспечивающих функционирование АСУ подсистем. Материал этой части учебника с достаточной степенью обобщенности позволяет студентам понять сущность АСУ различных классов. С этой целью в седьмой главе детально рассмотрены существующие в настоящее время виды управления, проанализированы их достоинства и недостатки. Основное внимание уделено содержанию управления иерархического типа.

В восьмой главе приведено описание основных классов АСУ. Это сделано с целью того, чтобы у студентов в процессе изучения теории автоматизированного управления сложилось целостное представление о связи изучаемой теории с практикой.

Заключительная, девятая глава учебника посвящена рассмотрению обеспечивающих подсистем автоматизированного управления. Здесь изложены методы создания математического, информационного, программного, технического, лингвистического и других видов подсистем автоматизированного управления.

В Заключении приведены сведения о направлениях дальнейшего развития теории и практики автоматизированного управле-

ния. Показана связь теории с вопросами автоматизированного управления, особенностями функционирования АСУ в условиях широкого применения информационной и вычислительных сетей, расширения возможностей вычислительной техники пятого поколения, использования экспертных и информационных интеллектуальных систем. Подчеркнуто исключительное влияние систем «советчик оператора» при принятии ответственных решений в условиях ограниченного времени, неопределенности информации и риска.

Каждая глава учебника дополнена контрольными вопросами.

Учебник содержит приложения: глоссарий автоматизированного управления, тематику курсовых работ и примеры их выполнения, содержание ГОСТов по АСУ.

В списке литературы указаны основные и дополнительные источники.

Имеющаяся в настоящее время литература по теории автоматизированного управления, которая может быть рекомендована студентам в качестве учебных пособий, к сожалению, либо устарела, либо отражает лишь отдельные вопросы дисциплины. Практически единственным учебником по теории автоматизированного управления является книга А.Г. Мамиконова «Основы построения АСУ», выпущенная издательством «Высшая школа» еще в 1981 г. и давно ставшая библиографической редкостью. Поэтому в библиотеках вузов в настоящее время учебная литература по теории автоматизированного управления практически отсутствует. Мы взяли на себя смелость восполнить этот пробел.

Авторы в работе над учебником поставили себе условие, чтобы содержание книги соответствовало требованиям Государственного образовательного стандарта и Примерной программе одноименной дисциплины — «Теоретические основы автоматизированного управления». В основу данного учебника положен более чем тридцатилетний опыт чтения курса лекций студентам в Московском государственном техническом университете им. Н.Э. Баумана, Московском государственном университете печати, Обнинском государственном техническом университете атомной энергетики и Сургутском государственном университете.

Работа над учебником распределена следующим образом: предисловие, введение, главы 1—3, 7—9, заключение написаны проф. В.А. Острейковским; главы 4—6, параграфы 2.2 — 2.4, 8.1 и 9.1 и приложения 1 и 2 — доц. А.В. Меньковым.

Методология и содержание учебника подготовлены благодаря многолетней работе авторов в УМО вузов по университетскому политехническому образованию под руководством выдаю-

щихся ученых в области автоматизации управления профессоров В.Н. Четверикова и Б.Я. Советова. Неоценимую помощь оказали авторам члены УМО проф. Ю.Г. Дреус, А.Г. Мамиконов, А.В. Петров, О.М. Петров, Н.И. Федунец, Я.Ф. Хетагуров, Г.А. Ходжаев, К.Э. Шахов, С.А. Яковлев.

Авторы выражают искреннюю благодарность рецензентам: кафедре «Автоматизированные системы управления» Обнинского государственного технического университета атомной энергетики (зав. кафедрой — доктор техн. наук, проф. А.Н. Анохин) и доктору техн. наук, проф. Л.И. Григорьеву (Московский государственный университет нефти и газа им. И.М. Губкина) за ценные замечания при рецензировании учебника. Авторы сердечно благодарят Н.А. Жукову за помощь при подготовке рукописи к печати, а также С.П. Саакяна за изготовление оригинал-макета учебника.

Авторы

Список основных сокращений

АИС	— автоматизированная информационная система
АП	— автоматизированное проектирование
АРМ	— автоматизированное рабочее место
АС	— автоматизированная система
АСНИ	— автоматизированная система научных исследований
АСОД	— автоматизированная система обработки данных
АСОИУ	— автоматизированная система обработки информации и управления
АСУ	— автоматизированная система управления
АСУП	— автоматизированная система управления предприятием
АСУТП	— автоматизированная система управления технологическими процессами
АТК	— автоматизированный технологический комплекс
АУ	— автоматизированное управление
БД	— база данных
БЗ	— база знаний
БнД	— банк данных
БИП	— блочно-иерархический подход
БИС	— большая интегральная схема
БС	— большая система
БУ	— бухгалтерский учет
ВП	— восходящее проектирование
ВТ	— вычислительная техника
ГАП	— гибкое автоматизированное производство
ГПМ	— гибкий производственный модуль
ГПС	— гибкая производственная система
ИАСУ	— интегрированная автоматизированная система управления
ИВС	— информационно-вычислительная система
ИО	— информационное обеспечение
ИПС	— информационно-поисковая система
ИТ	— информационная технология
КСАП	— комплекс средств автоматизации проектирования
ЛО	— лингвистическое обеспечение
ММ	— математическая модель
ММет	— математический метод
МО	— математическое обеспечение
НИР	— научно-исследовательская работа
НП	— нисходящее проектирование
ОАСУ	— отраслевая автоматизированная система управления
ОО	— организационное обеспечение
ОС	— операционная система
ОУП	— оперативное управление производством
ПО	— программное обеспечение

ПС	– программные средства
РП	– рабочий проект
РЭА	– радиоэлектронная аппаратура
САПР	– система автоматизированного проектирования
СРП	– сбыт и реализация продукции
САУ	– система автоматического управления
СИИ	– система искусственного интеллекта
СУ	– система управления
СЧМ	– система «человек– машина»
ТАУ	– теория автоматического управления
ТЗ	– техническое задание
ТО	– техническое (технологическое) обеспечение
ТОУ	– технологический объект управления
ТП	– технический проект
ТПП	– техническая подготовка производства
ТЭП	– технико-экономическое планирование
УВМ	– управляющая вычислительная машина
УКП	– управление качеством продукции
ЧПУ	– числовое программное управление
ЭВМ	– электронная вычислительная машина
ЭВТ	– электронно-вычислительная техника
ЭС	– экспертная система

Введение в теорию автоматизированного управления

В.1. Объективная необходимость автоматизации обработки информации и управления

Во второй половине XX века человечество вступило в новый этап своего развития. В этот период начался переход от индустриального общества к информационному. Процесс, обеспечивающий этот переход, получил название информатизации. *Информатизация* — это процесс создания, развития и всеобщего применения информационных средств и технологий, обеспечивающих достижение и поддержание уровня информированности всех членов общества, необходимого и достаточного для *кардинального улучшения качества труда и условий жизни общества*. При этом информация становится важнейшим стратегическим ресурсом общества и занимает ключевое место в экономике, образовании и культуре.

Неизбежность информатизации общества обусловлена резким возрастанием роли и значения информации.

В истории развития человеческого общества известны несколько информационных барьеров. В эпоху первобытно-общинного строя глава рода или племени один удерживал в голове все экономические вопросы своего сообщества. Однако по мере роста общественного производства, роста производительности труда возможности одного человеческого мозга оказались исчерпанными. Это *первый информационный барьер*. Выход оказался чрезвычайно прост: чем сложнее задача управления производством, войсками, образованием и т. д., тем большее число людей вовлекается в процесс управления. Но наступает такой момент, когда дальнейшее увеличение количества людей, занятых в обработке информации при принятии ответственных решений и управлении, уже не улучшает качества и эффективности управляющих воздействий на объект управления. Это *второй информационный барьер*.

В конце 50-х — начале 60-х годов XX века ведущие специалисты Франции предсказывали, что если экономика ведущих капиталистических стран мира будет развиваться высокими темпами под влиянием научно-технической революции второй половины прошлого века, то к середине 70-х годов (т. е. через 15 — 20 лет) для поддержания нормального функционирования экономики потребуется задействовать в сфере управления все взрослое население страны.

Аналогичные расчеты были выполнены и в СССР. Так, академик В.М. Глушков приводит следующие простейшие расчеты, ха-

рактизирующие состояние проблемы управления экономикой СССР на середину 60-х годов. Было подсчитано, что для более или менее удовлетворительного управления экономикой необходимо выполнять 10^{16} арифметических операций в год. Таким образом, если один человек с калькулятором способен за смену 8 часов выполнить тысячу арифметических операций, то работая без выходных, праздников и отпуска, он выполнит около 10^6 операций. Следовательно, для удовлетворительного управления экономикой необходимое число управленцев составит 10 миллиардов человек!

Средством преодоления второго информационного барьера являются автоматизированные информационные системы на базе ЭВМ.

В.2. Краткая историческая справка о развитии теории автоматизированного управления

Теория автоматизированного управления — это сравнительно молодая научная дисциплина, возникшая из потребностей практики в связи с бурным научно-техническим прогрессом и, в первую очередь, из-за появления сложных систем управления с большим числом элементов электроники и автоматики. Теория автоматизированного управления является ветвью кибернетики как науки о наиболее общих закономерностях управления в технических, биологических и социальных системах. Т. е. говорить о теории автоматизированного управления стало возможным только после появления классических работ в области теоретической кибернетики Н. Винера, У.-Р. Эшби, Дж. фон Неймана. Именно в монографиях этих ученых в 1948—1956 гг. были заложены современные основы науки об управлении. В сочетании с развитием теории информации (К. Шеннон, 1948 г.) наука об управлении получила строгое и стройное развитие. Большую роль в разработке теоретических основ кибернетики сыграли разработки советских и российских ученых: А.И. Берга, В.М. Глушкова, В.А. Трапезникова, Г.С. Поспелова, А.Г. Мамиконова, В.И. Скурихина, И.П. Норенкова, А.А. Тимченко и других.

На развитие теории автоматизированного управления оказала сильное влияние теория автоматического управления (ТАУ), которая к середине 50-х годов XX века была развита довольно хорошо. ТАУ — это наука о методах определения законов управления объектами, допускающих реализацию с помощью технических средств автоматики. Исторически сложилось так, что методы

ТАУ получили свое первое развитие применительно к процессам, встречающимся главным образом в технике. Три причины лежат в основе этого развития:

1) многие объекты управления были идентифицированы выдающимися учеными — английским физиком Дж.-К. Максвеллом (1831—1879), русским математиком А.И. Вышнеградским (1832—1895), словацким теплотехником А. Стодолой (1859—1942) и русским математиком А.А. Ляпуновым (1857—1918), т. е. были получены их математические модели, описываемые обыкновенными дифференциальными уравнениями (векторными)

$$\dot{x} = f(x, u, t) \quad (\text{B.1})$$

и неравенствами

$$N(x, u, t) \geq 0, \quad (\text{B.2})$$

где $x(x_1, x_2, \dots, x_n)$ — вектор состояния объекта, $u(u_1, u_2, \dots, u_m)$ — вектор управления, t — время. Уравнение (B.1) — математическая запись законов, которым подчиняется объект управления; неравенство (B.2) — область его определения;

2) еще задолго до развития ТАУ, благодаря установлению ряда фундаментальных законов природы, существовал хорошо развитый математический аппарат теории устойчивости движения;

3) инженеры открыли закон обратной связи и нашли средства его реализации.

Также в ТАУ внесли большой вклад такие выдающиеся отечественные и зарубежные ученые, как А. Гурвиц, Х. Найквист, А.В. Михайлов, А. Пуанкаре, А.А. Андронов, Н.Н. Красовский, Л.С. Понтрягин, В.И. Зубов, В.А. Бесекерский, Е.П. Попов, В.С. Емельянов, В.С. Пугачев и др.

Наряду с технической кибернетикой в 60-е годы XX века широкое развитие получила экономическая кибернетика. Экономическая кибернетика изучает процессы сбора, накопления, хранения и переработки информации об экономических объектах или явлениях. Предметом экономической кибернетики являются процессы управления экономикой страны, отрасли, района, промышленного комплекса, производства, отдельного предприятия, участка, нескольких станков или группы людей и т. д.

Методы анализа, применяемые в экономической кибернетике, помогают находить оптимальные режимы управления и строить рациональные системы обработки экономических данных, основанные на широком использовании вычислительной техники (ВТ). Исследования по экономической кибернетике направлены на:

- 1) выбор показателей, необходимых для управления экономическими объектами;
- 2) выбор способов получения, передачи и обработки информации об этих показателях;
- 3) выбор технических средств обработки экономической информации на различных уровнях управления;
- 4) изучение и рекомендации по алгоритмам и программам обработки информации.

Первоначально становление экономической кибернетики связано с разработкой математических моделей экономических систем и явлений, использованием электронных вычислительных машин (ЭВМ) для исследования этих моделей и для решения задач управления. Применение математических структур и методов во многом шло по линии применения математической символики и ликвидации терминологических различий. Это направление развивается и сейчас в рамках математических методов и моделей экономики. Математические модели экономических систем и явлений позволили лучше осмыслить динамику изучаемых систем, выработать действенные рекомендации по рационализации их структуры и методов экономического прогнозирования и управления. Особое значение имело изучение регулирующих факторов в таких моделях, вопросов устойчивости, равновесия, роста, регулирующего характера цен, выявление и подчеркивание обратных связей в экономике, исследование конфликтных ситуаций (в рамках теории игр), соотношений между оптимальным функционированием и общей мобильностью экономических систем. Экономическая кибернетика занимается вопросами принятия решений в управлении экономическими системами, применения математических моделей и получения на их основе машинно-формальных выводов для принятия решений в реальных ситуациях и постановках, решает вопросы границ и рецептов возможного использования формальных постановок и выводов. Большое внимание уделяется методам эвристического решения задач, экспертного прогнозирования и, наконец, построению человеко-машинных систем для решения экономических задач. Применение такого рода человеко-машинных систем, моделирование ситуаций и принятие решений для обучения и выработки наиболее рациональных форм и методов управления (деловые игры) способствуют тому, что математическое моделирование становится всё более универсальным средством совершенствования экономических систем, пробным камнем проверки экономических положений и доктрин.

Определяющим направлением в экономической кибернетике является разработка теории и построение автоматизированных

систем управления (АСУ) в народном хозяйстве, которое стало возможным только в связи с созданием современных технических средств обработки данных, особенно систем с разделением времени. Построение АСУ в экономике является составной частью процессов автоматизации, определяющих современную научно-техническую революцию. Необходимость создания таких систем обуславливается совершенствованием экономических систем, ростом производительности управленческого труда, эффективным использованием технических средств обработки данных и организации информационных процессов в целом, многими социальными требованиями. Быстрый рост производства, углубление специализации, расширение кооперирования и рыночных отношений, обновляемость продукции, эффективное использование ресурсов возможны только при создании АСУ. Экономическая кибернетика разрабатывает общие вопросы структуры, построения и функционирования АСУ в народном хозяйстве. Особое внимание уделяется вопросам эффективного сбора информации, ее представления, наименования, интерпретации, использования и циркуляции в АСУ. Информация в экономических системах становится предметом глубокого изучения. Разработка информационных технологий, методов моделирования, информационно-поисковых систем и информационно-справочных систем ориентируется на экономические объекты. В то же время экономическая кибернетика сама оказала во многом определяющее влияние и на развитие некоторых новых математических направлений — стохастического и динамического программирования, дискретной оптимизации, теории расписаний и др., на теоретические и практические разработки в области обработки данных, математического обеспечения ЭВМ. Развитие техники обработки данных, теории исследования операций и теории систем также связано с направлением развития и интересами экономической кибернетики.

В.3. Цель и задачи дисциплины «Теоретические основы автоматизированного управления»

Дисциплина «Теоретические основы автоматизированного управления» относится к циклу специальных дисциплин направления 654600 «Информатика и вычислительная техника», она является одной из основополагающих дисциплин в подготовке системного аналитика и дипломированных специалистов по специальности 220200 «Автоматизированные системы обработки информации и управления».

Цель дисциплины: изучение современного состояния теории автоматизированного управления на всех этапах проектирования, создания, отладки и эксплуатации АСУ.

В результате изучения дисциплины студенты должны ЗНАТЬ:

- основные понятия и определения теории автоматизированного управления;
- математические методы, используемые в теории автоматизированного управления;
- методологию и принципы построения автоматизированных систем (АС);
- методы и модели анализа и синтеза структуры АС;
- виды автоматизированного управления;
- методы принятия решений при автоматизированном управлении;
- классификацию и состав АСУ и обеспечивающих подсистем.

УМЕТЬ ИСПОЛЬЗОВАТЬ:

- принципы построения АС при решении практических вопросов исследования АС;
- математические методы при анализе и синтезе структуры АС;
- формальные методы при принятии решений в АСУ.

ИМЕТЬ ПРЕДСТАВЛЕНИЕ О:

- причинах появления теории автоматизированного управления;
- этапах и последовательности разработки АС;
- существующих и перспективных методах системного анализа;
- перспективах развития теории автоматизированного управления.

Раздел 1

МАТЕМАТИЧЕСКИЕ МЕТОДЫ ТЕОРИИ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

Глава 1

ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

В этой главе изложены основные понятия теории автоматизированного управления, подробно раскрыто одно из основных понятий – управление, показано, что любой процесс управления представляет собой информационный процесс, а система управления – это органическая совокупность объекта управления и управляющего органа. При этом выделена роль понятия цели управления, без формулировки которой управление системой теряет всякий смысл. Детально описано определение «информация». Хотя авторы понимают, что это понятие раскрывается студентам в дисциплинах «Информатика», «Теория информации», однако ввиду особой важности информации в процессах управления, и особенно при принятии решения, в этой главе студентам еще раз напоминает роль понятия «информация» в дисциплинах кибернетического направления. Дается краткое объяснение понятиям «система» и «большая система». В этой главе обращается особое внимание на отличие автоматизированного управления от автоматического.

1.1. Понятия «управление» и «система управления»

Центральным понятием кибернетики и ее раздела – теории автоматизированного управления – является понятие «управление».

Управление как процесс. **Управление** — в широком, кибернетическом смысле — это обобщение приемов и методов, накопленных разными науками об управлении искусственными объектами и живыми организмами. Язык управления — это использование понятий «объект», «среда», «обратная связь», «алгоритм» и т. д.

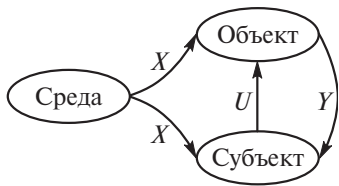


Рис. 1.1. Кибернетический подход к процессу управления

Под управлением будем понимать процесс организации такого целенаправленного воздействия на некоторую часть среды, называемую объектом управления, в результате которого удовлетворяются потребности субъекта, взаимодействующего с этим объектом. Анализ управления позволяет выделить триаду — среду, объект и субъект, — внутри которой

разыгрывается процесс управления (рис. 1.1). В данном случае субъект ощущает на себе воздействие среды X и объекта Y . Если состояние среды X он изменить не может, то состоянием объекта Y он может управлять с помощью специально организованного воздействия U . Это и есть управление.

Состояние объекта Y влияет на состояние потребностей субъекта. Потребности субъекта $A = (\alpha_1, \dots, \alpha_k)$, где α_i — состояние i -й потребности субъекта, которая выражается неотрицательным числом, характеризующим насущность, актуальность этой потребности. Свое поведение субъект строит так, чтобы минимизировать насущность своих потребностей, т. е. решает задачу многокритериальной оптимизации:

$$\alpha_i(X, U) \rightarrow \min_{r \in R} (i = 1, k), \quad (1.1)$$

где R — ресурсы субъекта. Эта зависимость выражает неизвестную, но существующую связь потребностей с состоянием среды X и поведением U субъекта.

Пусть U_x^* — решение задачи (1.1), т. е. оптимальное поведение субъекта, минимизирующее его потребности A . Способ решения задачи (1.1), позволяющий определить U_x^* , называется алгоритмом управления

$$U_x^* = \varphi(A_p, X), \quad (1.2)$$

где φ — алгоритм, позволяющий синтезировать управление по состоянию среды X и потребностей A_t . Потребности субъекта изменяются не только под влиянием среды или объекта, но и самостоятельно, отражая жизнедеятельность субъекта во времени, что отмечается индексом t .

Алгоритм управления φ , которым располагает субъект, и определяет эффективность его функционирования в данной среде. Алгоритм имеет рекуррентный характер:

$$U_{N+1} = \varphi(U_N, A_t, X),$$

т. е. позволяет на каждом шаге улучшать управление. Например, в смысле

$$A_t(X, U_{N+1}) < A_t(X, U_N),$$

т. е. уменьшения уровня своих потребностей.

Процесс управления как организация целенаправленного воздействия на объект может реализовываться как на интуитивном, так и на осознанном уровне. Первый используют животные, второй — человек. Осознанное удовлетворение потребностей заставляет декомпозировать алгоритм управления и вводить промежуточную стадию — формулировку цели управления, т. е. действовать по двухэтапной схеме:

$$\underbrace{A_t \rightarrow Z^*}_1 \rightarrow \underbrace{U^*}_2.$$

На первом этапе определяется цель управления Z^* , причем задача решается на интуитивном уровне:

$$Z^* = \varphi_1(X, A_t),$$

где φ_1 — алгоритм синтеза цели Z^* по потребностям A_t и состоянию среды X . На втором этапе определяется управление U_x^* , реализация которого обеспечивает достижение цели Z^* , сформулированной на первой стадии, что и приводит к удовлетворению потребностей субъекта. Именно на этой стадии может быть использована вся мощь формального аппарата, с помощью которого по цели Z^* синтезируется управление

$$U_x^* = \varphi_2(Z^*, X),$$

где φ_2 — алгоритм управления. Этот алгоритм и есть предмет изучения кибернетики как науки.

Таким образом, разделение процесса управления на два этапа отражает известные стороны науки — неформальный, интуитивный, экспертный и формальный, алгоритмизуемый. Если первый этап пока полностью принадлежит человеку, то второй является объектом приложения формальных подходов. Естественно, что эти различные функции выполняются разными структурными элементами. Первую функцию φ_1 выполняет субъект, а вторую φ_2 — управляющее устройство (УУ). На рис. 1.2 показано взаимодействие этих элементов. Штриховой линией выделена система

управления (СУ), выполняющая функцию реализации целей управления Z^* , формируемых субъектом.

Системы управления и сложный объект управления. Структурная схема СУ приведена на рис. 1.3. Здесь D_x и D_y — датчики, измеряющие состояние среды и объекта соответственно. Результаты измерений $X' = D_x(X)$ и $Y' = D_y(Y)$ образуют исходную информацию $J = \{X', Y'\}$ для УУ, которое на этой основе вырабатывает команду управления U , являющуюся лишь информацией о том, в какое положение должны быть приведены управляемые входы объекта. Следовательно, управление U есть результат работы алгоритма

$$U = \varphi_2(J, Z^*).$$

Как видно, управление в широком смысле образуется четверкой

$$\{Z^*, J, U, \varphi_2\}.$$

В качестве примера рассмотрим основные понятия управления в технических и организационных системах.

Управление — целенаправленная организация того или иного процесса, протекающего в системе. В общем случае процесс управления состоит из следующих четырех этапов:

- получение информации о задачах управления (Z^*);
- получение информации о результатах управления (т. е. о поведении объекта управления Y');

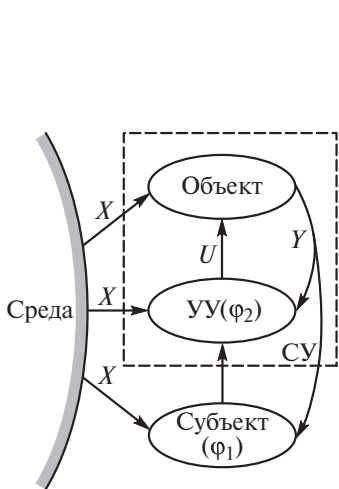


Рис. 1.2. Взаимодействие элементов системы управления

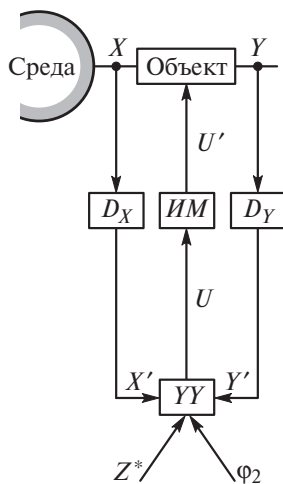


Рис. 1.3. Структурная схема системы управления

- анализ полученной информации и выработка решения ($J = \{X', Y'\}$);
- исполнение решения (т. е. осуществление управляющих воздействий U').

Процесс управления — это информационный процесс (рис. 1.4), заключающийся в сборе информации о ходе процесса, передаче ее в пункты накопления и переработки, анализе поступающей, накопленной и справочной информации, принятии решения на основе выполненного анализа, выработке соответствующего управляющего воздействия и доведении его до объекта управления. Каждая фаза процесса управления протекает во взаимодействии с окружающей средой при воздействии различного рода помех. Цели, принципы и границы управления зависят от сущности решаемой задачи.

Система управления — совокупность взаимодействующих между собой объекта управления и органа управления, деятельность которых направлена на достижение заданной цели управления (рис. 1.5).

Задачи, решаемые системой управления. В СУ решаются четыре основные задачи управления: стабилизация, выполнение программы, слежение, оптимизация.

Задачами стабилизации системы являются задачи поддержания ее выходных величин вблизи некоторых неизменных заданных значений, несмотря на действие помех. Например, стабилизация напряжения U и частоты f тока в сети вне зависимости от изменения потребления энергии.

Задача выполнения программы возникает в случаях, когда заданные значения управляемых величин изменяются во времени заранее известным образом. Например, полет ракеты, выполнение работ по заранее намеченному графику.

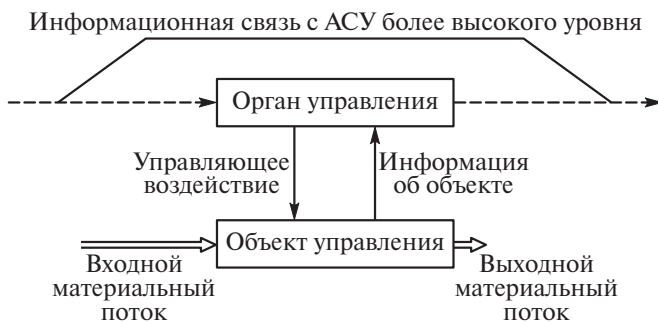


Рис. 1.4. Процесс управления как информационный процесс

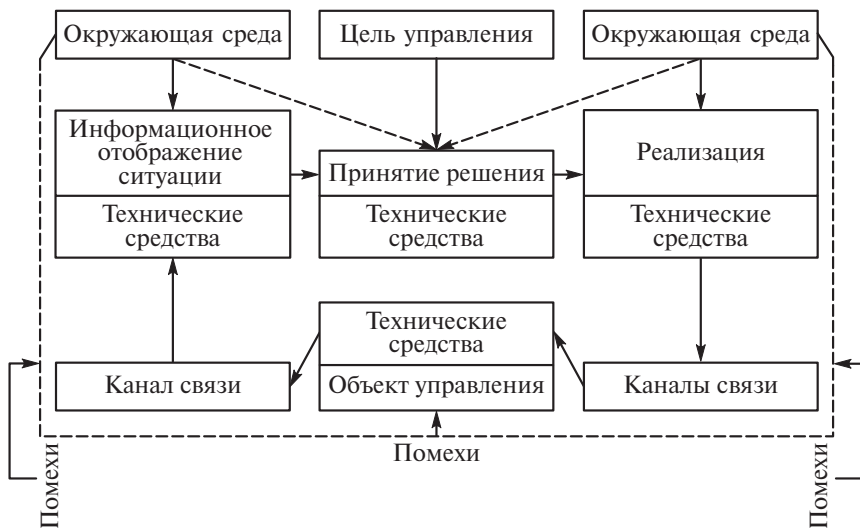


Рис. 1.5. Система управления как совокупность объектов

В тех случаях, когда изменение заданных значений управляемых величин заранее неизвестно и когда эти величины должны изменяться в зависимости от значений других величин, возникает задача *слежения*, т. е. как можно более точного соблюдения соответствия между текущим состоянием данной системы и состоянием другой системы. Например, управление производством в условиях изменения спроса, слежение за целью (например, самолетом, кораблем, космическим объектом).

В системах оптимального управления требуется наилучшим образом выполнить поставленную перед системой задачу при заданных реальных условиях и ограничениях. Понятие оптимальности должно быть конкретизировано для каждого отдельного случая.

Прежде чем принимать решение о создании СУ, необходимо рассмотреть все его этапы, независимо от того, с помощью каких технических средств они будут реализованы. Такой алгоритмический анализ управления является основой для принятия решения о создании СУ и степени ее автоматизации. При этом анализе следует обязательно учитывать фактор сложности объекта управления:

- отсутствие математического описания системы;
- стохастичность поведения;
- негативность к управлению;

- нестационарность, дрейф характеристик;
- невоспроизводимость экспериментов (развивающаяся система все время как бы перестает быть сама собой, что предъявляет специальные требования к синтезу и коррекции модели объекта управления).

Особенности сложной системы часто приводят к тому, что цель управления таким объектом в полной мере никогда не достигается, как бы совершенно ни было управление.

Автоматическое и автоматизированное управление. Системы управления делятся на два больших класса: системы автоматического управления (САУ) и автоматизированные системы управления (АСУ). В САУ управление объектом или системой осуществляется без непосредственного участия человека автоматическими устройствами. Это замкнутые системы. Основные функции САУ: автоматический контроль и измерения, автоматическая сигнализация, автоматическая защита, автоматические пуск и остановка различных двигателей и приводов, автоматическое поддержание заданных режимов работы оборудования, автоматическое регулирование. В отличие от САУ в АСУ в сферу управления включен человек, на которого возлагаются функции принятия наиболее важных решений и ответственности за принятые решения. Под АСУ обычно понимают человеко-машинные системы, использующие современные экономико-математические методы, средства электронно-вычислительной техники (ЭВТ) и связи, а также новые организационные принципы для отыскания и реализации на практике наиболее эффективного управления объектом (системой).

В соответствии с общепринятым представлением под автоматизированной системой понимается «человеко-машинная система, обеспечивающая автоматизированный сбор и обработку информации, необходимой для оптимизации управления в любых сферах человеческой деятельности». В определении особо следует выделить понятие «человеко-машинная система». В автоматических системах функции человека сводятся к разработке, отладке и контролю работы системы. Само же управление осуществляется без участия человека. В автоматизированных системах наличие человека (коллектива людей) в контуре управления является принципиальным. Человек (коллектив людей) является главным определяющим звеном системы управления, поскольку он принимает решения и несет за них всю ответственность — в этом принципиальная разница между автоматическими и автоматизированными системами.

1.2. Терминология теории автоматизированного управления

Определение понятия «система». В настоящее время нет единства в определении понятия «система». В первых определениях в той или иной форме говорилось о том, что система — это элементы и связи (отношения) между ними. Например, основоположник теории систем *Людвиг фон Берталанфи* определял систему как комплекс взаимодействующих элементов или как совокупность элементов, находящихся в определенных отношениях друг с другом и со средой. *А. Холл* определяет систему как множество предметов вместе со связями между предметами и между их признаками. Ведутся дискуссии, какой термин — «отношение» или «связь» — лучше употреблять.

Позднее в определениях системы появляется понятие цели. Так, в «Философском словаре» система определяется как «совокупность элементов, находящихся в отношениях и связях между собой определенным образом и образующих некоторое целостное единство».

В последнее время в определение понятия системы наряду с элементами, связями и их свойствами и целями начинают включать наблюдателя, хотя впервые на необходимость учета взаимодействия между исследователем и изучаемой системой указал один из основоположников кибернетики *У.-Р. Эшби*.

М. Масарович и *Я. Такаха* в книге «Общая теория систем» считают, что система — «формальная взаимосвязь между наблюдаемыми признаками и свойствами».

В качестве «рабочего» определения понятия системы в литературе по теории систем часто рассматривается следующее: система — *множество элементов, находящихся в отношениях и связях друг с другом, которое образует определенную целостность, единство.*

Рассмотрим основные понятия, характеризующие строение и функционирование систем.

Элемент. Под элементом принято понимать простейшую неделимую часть системы. Ответ на вопрос, что является такой частью, может быть неоднозначным и зависит от цели рассмотрения объекта как системы, от точки зрения на него или от аспекта его изучения. Таким образом, элемент — это предел членения системы с точки зрения решения конкретной задачи и поставленной цели. Систему можно расчленить на элементы различными способами в зависимости от формулировки цели и ее уточнения в процессе исследования.

Подсистема. Система может быть разделена на элементы не сразу, а последовательным расчленением на подсистемы, которые представляют собой компоненты более крупные, чем элементы, и в то же время более детальные, чем система в целом. Возможность деления системы на подсистемы связана с вычленением совокупностей взаимосвязанных элементов, способных выполнять относительно независимые функции, подцели, направленные на достижение общей цели системы. Названием «подсистема» подчеркивается, что такая часть должна обладать свойствами системы (в частности, свойством целостности). Этим подсистема отличается от простой группы элементов, для которой не сформулирована подцель и не выполняются свойства целостности (для такой группы используется название «компоненты»). Например, подсистемы АСУ, подсистемы пассажирского транспорта крупного города.

Структура. Это понятие происходит от латинского слова *structure*, означающего строение, расположение, порядок. Структура отражает наиболее существенные взаимоотношения между элементами и их группами (компонентами, подсистемами), которые мало меняются при изменениях в системе и обеспечивают существование системы и ее основных свойств. Структура — это совокупность элементов и связей между ними. Структура может быть представлена графически, в виде теоретико-множественных описаний, матриц, графов и других языков моделирования структур.

Структуру часто представляют в виде иерархии. Иерархия — это упорядоченность компонентов по степени важности (многоступенчатость, служебная лестница). Между уровнями иерархической структуры могут существовать взаимоотношения строгого подчинения компонентов (узлов) нижележащего уровня одному из компонентов вышележащего уровня, т. е. отношения так называемого древовидного порядка. Такие иерархии называют сильными или иерархиями типа «дерево». Они имеют ряд особенностей, делающих их удобным средством представления систем управления. Однако могут быть связи и в пределах одного уровня иерархии. Один и тот же узел нижележащего уровня может быть одновременно подчинен нескольким узлам вышележащего уровня. Такие структуры называют иерархическими структурами со слабыми связями. Между уровнями иерархической структуры могут существовать и более сложные взаимоотношения, например, типа «страт», «слои», «эшелоны». Примеры иерархических структур: энергетические системы, АСУ, государственный аппарат.

Связь. Понятие «связь» входит в любое определение системы наряду с понятием «элемент» и обеспечивает возникновение и сохранение структуры и целостных свойств системы. Это понятие характеризует одновременно и строение (статiku), и функционирование (динамику) системы.

Связь характеризуется направлением, силой и характером (или видом). По первым двум признакам связи можно разделить на направленные и ненаправленные, сильные и слабые, а по характеру — на связи подчинения, генетические, равноправные (или безразличные), связи управления. Связи можно разделить также по месту приложения (внутренние и внешние), по направленности процессов в системе в целом или в отдельных ее подсистемах (прямые и обратные). Связи в конкретных системах могут быть одновременно охарактеризованы несколькими из названных признаков.

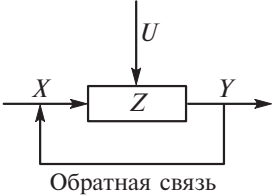
Важную роль в системах играет понятие «обратной связи». Это понятие, легко иллюстрируемое на примерах технических устройств, не всегда можно применить в организационных системах. Исследованию этого понятия большое внимание уделяется в кибернетике, в которой изучается возможность перенесения механизмов обратной связи, характерных для объектов одной физической природы, на объекты другой природы. Обратная связь является основой саморегулирования и развития систем, приспособления их к изменяющимся условиям существования.

Состояние. Понятием «состояние» обычно характеризуют мгновенную фотографию, «срез» системы, остановку в ее развитии. Его определяют либо через входные воздействия и выходные сигналы (результаты), либо через макропараметры, макросвойства системы (например, давление, скорость, ускорение — для физических систем; производительность, себестоимость продукции, прибыль — для экономических систем).

Более полно состояние можно определить, если рассмотреть элементы (или компоненты, функциональные блоки), определяющие состояние, учесть, что «входы» можно разделить на управляющие и возмущающие x (неконтролируемые) и что «выходы» (выходные результаты, сигналы) зависят от ε , u и x , т. е. $z_t = f(\varepsilon_t, u_t, x_t)$. Тогда в зависимости от задачи состояние может быть определено как $\{\varepsilon, u\}$, $\{\varepsilon, u, z\}$ или $\{\varepsilon, x, u, z\}$.

Таким образом, состояние — это множество существенных свойств, которыми система обладает в данный момент времени (табл. 1.1).

Система как «черный ящик»

Представление системы	Обозначения
	<p> X — вектор входных сигналов Y — вектор выходных сигналов Z — вектор состояния системы U — вектор управляющих сигналов </p>

Поведение. Если система способна переходить из одного состояния в другое (например, $z_1 \rightarrow z_2 \rightarrow z_3$), то говорят, что она обладает поведением. Этим понятием пользуются, когда неизвестны закономерности переходов из одного состояния в другое. Тогда говорят, что система обладает каким-то поведением, и выясняют его закономерности. С учетом введенных выше обозначений поведение можно представить как функцию $z_t = f(z_{t-1}, u_t, x_t)$.

Внешняя среда. Под внешней средой понимается множество элементов, которые не входят в систему, но изменение их состояния вызывает изменение поведения системы.

Модель. Под моделью системы понимается описание системы, отображающее определенную группу ее свойств. Углубление описания — детализация модели. Создание модели системы позволяет предсказывать ее поведение в определенном диапазоне условий.

Модель функционирования (поведения) системы — это модель, предсказывающая изменение состояния системы во времени, например: натурные, электрические, машинные и другие модели.

Равновесие — это способность системы в отсутствие внешних возмущающих воздействий (или при постоянных воздействиях) сохранить свое состояние сколь угодно долго.

Устойчивость. Под устойчивостью понимается способность системы возвращаться в состояние равновесия после того, как она была из этого состояния выведена под влиянием внешних возмущающих воздействий. Эта способность обычно присуща системам при постоянном u_t , если только отклонения не превышают некоторого предела.

Состояние равновесия, в которое система способна возвращаться, по аналогии с техническими устройствами называют *устойчивым состоянием равновесия*. Равновесие и устойчивость в экономических и организационных системах — гораздо более сложные понятия, чем в технике, и до недавнего времени ими

пользовались только для некоторого предварительного описательного представления о системе. В последнее время появились попытки формализованного отображения этих процессов и в сложных организационных системах, помогающие выявлять параметры, влияющие на их протекание и взаимосвязь.

Развитие. Исследованию процесса развития, соотношения процессов развития и устойчивости, изучению механизмов, лежащих в их основе, уделяют в кибернетике и теории систем большое внимание. Понятие развития помогает объяснить сложные термодинамические и информационные процессы в природе и обществе.

Цель. Применение понятия «цель» и связанных с ним понятий целенаправленности, целеустремленности, целесообразности сдерживается трудностью их однозначного толкования в конкретных условиях. Это связано с тем, что процесс целеобразования и соответствующий ему процесс обоснования целей в организационных системах весьма сложен и не до конца изучен. Большое внимание его исследованию уделяется в психологии, философии, кибернетике. В Большой Советской Энциклопедии цель определяется как «заранее мыслимый результат сознательной деятельности человека». В практических применениях *цель* — это идеальное устремление, которое позволяет коллективу увидеть перспективы или реальные возможности, обеспечивающие своевременность завершения очередного этапа на пути к идеальным устремлениям. В настоящее время в связи с усилением программно-целевых принципов в планировании исследованию закономерностей целеобразования и представления целей в конкретных условиях уделяется все больше внимания. Например: энергетическая программа, продовольственная программа, жилищная программа, программа перехода к рыночной экономике.

Определение большой системы. Существует ряд подходов к разделению систем по сложности. В частности, *Г. Н. Поваров* в зависимости от числа элементов, входящих в систему, выделяет четыре класса систем: малые системы ($10 \dots 10^3$ элементов), сложные ($10^4 \dots 10^7$ элементов), ультрасложные ($10^7 \dots 10^{30}$ элементов), суперсистемы ($10^{30} \dots 10^{200}$ элементов). Так как понятие элемента возникает относительно задачи и цели исследования системы, то и данное определение сложности является относительным, а не абсолютным.

Английский кибернетик *С. Бир* классифицирует все кибернетические системы на простые и сложные в зависимости от способа описания: детерминированного или теоретико-вероятностного. *Л. И. Берг* определяет сложную систему как систему,

которую можно описать не менее чем на двух различных математических языках (например, с помощью теории дифференциальных уравнений и алгебры Буля).

Очень часто сложными системами называют системы, которые нельзя корректно описать математически либо потому, что в системе имеется очень большое число элементов, неизвестным образом связанных друг с другом, либо потому, что неизвестна природа явлений, протекающих в системе. Все это свидетельствует об отсутствии единого определения сложности системы.

При разработке сложных систем возникают проблемы, относящиеся не только к свойствам их составляющих элементов и подсистем, но также к закономерностям функционирования системы в целом. При этом появляется широкий круг специфических задач, таких, как определение общей структуры системы, организация взаимодействия между элементами и подсистемами, учет влияния внешней среды, выбор оптимальных режимов функционирования системы, оптимальное управление системой и др.

Чем сложнее система, тем большее внимание уделяется этим вопросам. Математической базой исследования сложных систем является теория систем. В теории систем *большой системой (сложной, системой большого масштаба, Large Scale Systems) называют систему, если она состоит из большого числа взаимосвязанных и взаимодействующих между собой элементов и способна выполнять сложную функцию.*

Четкой границы, отделяющей простые системы от больших, нет. Деление это условное и возникло из-за появления систем, имеющих в своем составе совокупность подсистем с наличием функциональной избыточности. Простая система может находиться только в двух состояниях: состоянии работоспособности (исправном) и состоянии отказа (неисправном). При отказе элемента простая система либо полностью прекращает выполнение своей функции, либо продолжает ее выполнение в полном объеме, если отказавший элемент резервирован. Большая система при отказе отдельных элементов и даже целых подсистем не всегда теряет работоспособность, зачастую только снижаются характеристики ее эффективности. Это свойство больших систем обусловлено их функциональной избыточностью и, в свою очередь, затрудняет формулировку понятия «отказ» системы.

Под *большой системой* понимается совокупность материальных ресурсов, средств сбора, передачи и обработки информации, людей-операторов, занятых на обслуживании этих средств, и людей-руководителей, облеченных надлежащими правами и ответс-

твенностью за принятие решений. Материальные ресурсы — это сырье, материалы, полуфабрикаты, денежные средства, различные виды энергии, станки, оборудование, люди, занятые на выпуске продукции, и т. д. Все указанные элементы ресурсов объединены с помощью некоторой системы связей, которые по заданным правилам определяют процесс взаимодействия между элементами для достижения общей цели или группы целей.

Примеры больших систем: информационная система, пассажирский транспорт крупного города, производственный процесс, система управления полетом крупного аэродрома, энергетическая система и др.

Характерные особенности больших систем. К ним относятся:

- большое число элементов в системе (сложность системы);
- взаимосвязь и взаимодействие между элементами;
- иерархичность структуры управления;
- обязательное наличие человека в контуре управления, на которого возлагается часть наиболее ответственных функций управления.

1.3. Этапы управления

Управление сложной системой состоит из этапов, представленных на рис. 1.6.

1. *Формирование целей.* Множество целей управления, которое должно реализовываться СУ, определяется как внешними по отношению к системе, так и внутренними факторами и, в частности, потребностями субъекта A . Сложность формализации учета влияния этих факторов на цели очевидна. Различают три вида целей: стабилизация — заключается в требовании поддерживать выходы объекта на заданном уровне; ограничение — требует нахождения в заданных границах целевых переменных Z_i^* , $i = 1, k$; экстремальная цель — сводится к поддержанию в экстремальном состоянии целевых переменных Z_i^* .

2. *Определение объекта управления.* Этот этап связан с выделением той части среды субъекта, состояние которой он может изменить и тем самым воздействовать на свои потребности. В ряде случаев, когда границы объекта очевидны, проблемы выделения объекта из среды не возникает. Это бывает, когда объект достаточно автономен (самолет, телефонная станция и т. д.). Однако в других случаях связи объекта со средой настолько сильны и разнообразны, что порой очень трудно понять, где кончается объект и начинается среда. Именно это и заставляет вводить специальный этап — определение объекта управления.

Объект должен быть в определенном смысле минимальным, т. е. иметь наименьший объем. Это необходимо с целью минимизации трудоемкости его изучения при синтезе модели. При этом существенным ограничением выступает достижимость множества целей управления $\{Z^*\}$ в рамках выделенного для этого ресурса R . Это означает, что для любого состояния среды X должно найтись управление $U^* \in R$, с помощью которого можно добиться любой допустимой цели $Z^* \in \{Z^*\}$.

3. *Структурный синтез модели.* Последующие три этапа управления сложными системами связаны с решением задачи создания ее модели, которая нужна для синтеза управления U . Только с помощью модели объекта можно построить управление U^* , переводящее объект в требуемое (целевое) состояние Z^* .

Модель F , связывающая входы X и U с выходом Y , определяется структурой ST и параметрами $C = \{c_1, \dots, c_k\}$, т. е. представима в виде двойки $F = \{ST, C\}$. На этом этапе определяется структура ST , т. е. модель объекта с точностью до значений ее параметров C . Этап структурного синтеза включает определение внешней структуры модели, декомпозицию модели, определение внутренней структуры элементов модели.

Синтез внешней структуры сводится к содержательному определению входов X и U , выхода Y без учета внутренней структуры объекта, т. е. объект рассматривается как некий «черный ящик» с $n + q$ входами и m выходами. Декомпозиция модели заключается в том, чтобы, воспользовавшись априорными сведениями о структуре объекта, упростить задачу синтеза структуры модели. Синтез структуры модели сводится к определению вида оператора F модели объекта с точностью до параметров C . Это значит, что параметры становятся переменными модели, т. е.

$$Y = F(X, U, C), \tag{1.3}$$

где F — оператор преобразования структуры ST , параметры которого для удобства внесены в переменные C . Представление опе-



Рис. 1.6. Этапы управления сложной системой

ратора преобразования модели в виде (1.3) можно назвать параметризацией модели, что эквивалентно заданию его структуры. При синтезе структуры моделей объектов управления могут применяться различные подходы — от классических методов ТАУ до современных методов имитационного моделирования (методы случайного поиска, статистических испытаний и др.), семиотического моделирования с применением языка бинарных отношений и других методов современной математики, использующих сочетание дополняющих друг друга возможностей аналитических и статистических, семиотических и графических и других формализованных представлений системы.

4. *Идентификация параметров модели объекта.* Этот этап связан с определением числовых значений параметров C в режиме нормального функционирования объекта. Делается это стандартными приемами идентификации. Для выяснения зависимости выхода объекта от управляемых входов U необходимо преднамеренно их изменять, т. е. экспериментировать с объектом. Однако сложная система «не любит» эксперименты, нарушающие режим ее нормального функционирования. Поэтому эксперимент, которого нельзя избежать, следует проводить, минимально возмущая объект, но так, чтобы получить при этом максимальную информацию о влиянии варьируемых параметров на выход объекта.

5. *Планирование эксперимента.* На данном этапе главным является синтез плана эксперимента, позволяющего с максимальной эффективностью определить искомые параметры модели объекта управления. Для статического объекта этот план U представляет собой набор состояний управляемого выхода объекта $\tilde{U} = \{U_1, \dots, U_N\}$, а для динамического — план-функцию $\tilde{U} = U(t)$, $0 \leq t \leq T$, т. е. программу изменения во времени входа объекта. Эксперимент на объекте дает возможность определить реакцию объекта на это воздействие. В статическом случае эта реакция имеет вид $Y = \{y_1, \dots, y_N\}$, где $y_i = F^\circ(V_i)$, $i = 1 \dots N$, а в динамическом — $\tilde{Y}(t) = F^\circ[U(t)]$. Полученная информация и является исходной для определения параметров модели $F: Y = F(U, C)$, что осуществляется методами идентификации.

План эксперимента \tilde{U} определяется:

- структурой ST модели F ,
- ресурсом планирования R , который образуется выделяемыми на эксперимент средствами, областью планирования, определяющей пределы изменения входа U ;
- критерием планирования, который определяет эффективность плана \tilde{U} .

6. *Синтез управления.* На этом этапе принимается решение о том, каково должно быть управление U , чтобы достигнуть заданной цели управления Z^* в объекте. Это решение опирается на имеющуюся модель объекта F , заданную цель Z^* , полученную информацию о состоянии среды X и выделенный ресурс управления R , который представляет собой ограничения, накладываемые на управление U в связи со спецификой объекта и возможностями СУ. Достижение цели Z^* возможно соответствующим выбором управления U (состояние среды X изменяется независимо от нас). Это приводит к экстремальной задаче

$$Q(X, Y) \Rightarrow \min \Rightarrow U^*, \quad (1.4)$$

$$V \in \Omega$$

решение которой U^* является оптимальным управлением. Способы решения задачи (1.4) существенно зависят от структуры модели объекта F . Если объект статический, т. е. F — функция, то получаем задачу математического программирования, если же динамический, т. е. F — оператор, то решают вариационную задачу.

7. *Реализация управления или отработка в объекте оптимального решения U^* , полученного на предыдущем этапе.* Реализовав управление и убедившись, что цель управления не достигнута, возвращаются к одному из предыдущих этапов. Даже в лучшем случае, когда поставленная цель достигнута, необходимость обращения к предыдущему этапу вызывается изменением состояния среды X или сменой цели управления Z^* .

Таким образом, при благоприятном стечении обстоятельств обращаются к этапу синтеза управления (стрелка a на рис. 1.6), где определяется новое состояние, которое отражает новую ситуацию, сложившуюся в среде. Так функционирует стандартный контур управления простым объектом.

8. *Адаптация.* Специфика управления сложной системой состоит в том, что благодаря зашумленности и нестационарности информация, полученная на предыдущих этапах, приближенно отражает состояние системы лишь в предыдущие моменты времени. Это и вызывает необходимость коррекции. Коррекция может затрагивать различные этапы.

Простейшая коррекция связана с подстройкой параметров модели S (стрелка c , рис. 1.6). Такого рода коррекцию называют адаптацией модели, а управление — *адаптивным управлением*. Если управление U не обеспечивает необходимого разнообразия входа объекта для эффективной коррекции параметров модели, то приходится принимать специальные меры планирования эксперимента путем добавления специальных тестовых сигналов

(стрелка *b*, рис. 1.6). Такое управление называют дуальным. Однако одной коррекции параметров модели может оказаться недостаточно, если изменилась ее структура. Поэтому время от времени необходима коррекция структуры модели, т. е. приведение ее в соответствие с новой информацией (стрелка *d*, рис. 1.6). Далее коррекция может коснуться самого объекта, точнее, границы разделения объекта и среды. Это бывает необходимо при значительном изменении (эволюции) объекта и окружающей ее среды (стрелка *e*, рис. 1.6). И наконец, созданная СУ по ряду причин может не реализовать все множество целей управления, в результате необходима адаптация целей (стрелка *g*, рис. 1.6).

Очевидно, что не все из описанных выше восьми этапов управления присутствуют при синтезе СУ. В ряде случаев некоторые из них выпадают. Например, объект управления может быть выделен из среды и тогда нет необходимости в этапе планирования эксперимента, так как модель объекта проста и все ее параметры можно определить без специально организованного эксперимента.

1.4. Объект и предмет теории автоматизированного управления

Объектом любой теории является то, на что она направлена, т. е. что является ее содержанием. В этом смысле *объектом теории автоматизированного управления* является процесс управления в организационно-экономических и технических системах. Поэтому все содержание данного учебника представляет собой подробное изложение современного подхода к сущности процесса управления в соответствии с требованиями Государственного образовательного стандарта специальности 220200 – Автоматизированные системы обработки информации и управления.

Предметом теории является аппарат, с помощью которого производятся исследования объекта данной теории. Предметом теории могут быть математические методы и модели, виды моделирования, технические средства. Математическим аппаратом теории автоматизированного управления являются три раздела теоретической кибернетики: вероятностные методы, методы оптимизации и численные методы и методы дискретной математики.

Так как в основе исследования большинства сложных систем лежат *вероятностные методы*, то первые три раздела вероятностных методов посвящаются *теории вероятностей, математической статистике* и *теории марковских процессов*. В разделе математической статистики в основном излагаются вопросы опре-

деления числовых значений характеристик случайных величин и законов распределения (рис 1.7). Уже на примере этого раздела можно убедиться в том, что различные разделы кибернетики взаимно проникают друг в друга и резкого разделения провести не удастся. Часто изложение теории вероятностей производится с привлечением общей алгебры и математической логики, которые рассматриваются в разделе дискретной математики. Изложение математической статистики заканчивается теорией статистических решений, предлагающей математические методы для принятия решений в условиях неопределенности. Эти методы в значительной степени перекрываются методами *теории игр*.

Четвертый раздел вероятностных методов посвящен *теории информации и кодирования*. Наряду с рассмотрением вероятностного понятия информации, ее свойств и особенностей преобразования, развитых в теории связи, в этом разделе большое внимание уделяется другому подходу к информационным явлениям, который сформировался в связи машинной обработкой информации и внедрением АСУ и в котором используется понятие тезауруса. Тезаурус — словарь, который очищен от неоднозначности, т. е. в нем каждому слову может соответствовать лишь единственное понятие, хотя в обычном словаре одному слову может соответствовать несколько понятий. Между тезаурусом и обычным словарем имеются принципиальные различия.

Оптимальные методы обработки, передачи, преобразования и защиты информации существенным образом зависят от способов ее кодирования, которым уделяется большое внимание в этом разделе. Для построения оптимальных систем управления необходимо иметь в наличии математический аппарат для отыскания оптимальных законов управления, который составляет основное содержание второго цикла математических основ кибернетики. В зависимости от специфики системы управления могут применяться различные методы оптимизации: от классических методов Эйлера — Лагранжа, динамического программирования и при-



Рис.1.7. *Вероятностные методы в кибернетике*

нципа максимума Понтрягина до методов математического программирования. При этом рассматриваются непрерывные и дискретные, детерминированные и вероятностные варианты этих методов. Очень часто для отыскания оптимального управления приходится решать численными методами дифференциальные, интегральные или разностные уравнения, которые составляют в настоящее время самостоятельный большой раздел прикладной математики. Учитывая использование ЭВМ, желательно окончательно численные процедуры отыскания решения записывать с помощью какого-нибудь проблемно-ориентированного языка.

Особого внимания заслуживает третий раздел математических основ кибернетики — дискретная математика (рис.1.8). Как уже указывалось, большинство процессов управления, особенно в АСУ, дискретные. Массивы информации и программы, записанные на машинных носителях, дискретны по своей структуре. В редких случаях их можно описать с помощью аппарата непрерывной математики (дифференциальных уравнений). Поэтому для специалиста по управлению требуется более глубокая математическая подготовка по сравнению с той, которая дается в настоящее время в вузах. Ему необходимо хорошо знать дискретную математику, которая называется так потому, что в ней нет понятия непрерывности, дифференцируемости. Дискретная математика включает следующие разделы: теорию множеств и общую алгебру, математическую логику, теорию алгоритмов, теорию автоматов, теорию графов, комбинаторное исчисление, математическую лингвистику.

Все кибернетические модели, рассматриваемые в теории автоматизированного управления, разделены на три группы. Здесь первую группу составляют модели, в основе которых лежит вероятностная природа (рис. 1.9). Это модели теории массового обслуживания, теории надежности, теории игр, распознавания образов.

Теория массового обслуживания (теория очередей) — это раздел прикладной математики, изучающий процессы, связанные с удовлетворением массового спроса на обслуживание какого-либо вида с учетом случайного характера спроса и обслуживания (телефонные задачи, запасы продуктов, транспортные перевозки, дороги). *Теория надежности* изучает закономерности возникновения и устранения отказов. *Теория игр и статистических решений* — это математическая теория оптимальных решений в конфликтных ситуациях. *Теория распознавания образов* изучает процессы принятия решений о наиболее существенных свойствах некоторого объекта на основании косвенных данных, т. е. на основании наблюдения других свойств — признаков, зависящих от упомянутых существенных свойств.

Вторая группа объединяет кибернетические модели объектов, поведение которых описывается дифференциальными или разностными уравнениями. Большинство методов исследования таких систем излагается в работах по системам автоматического регулирования и управления. Для этих методов характерно рассмотрение процессов во времени, поэтому такие модели могут быть названы динамическими системами.

Третью группу кибернетических моделей составляют дискретные кибернетические модели. Эти модели применяются и для исследования процессов управления, протекающих во времени, но в основном в них время не используется. Например, требуется с помощью вычислительных машин раскроить листовое железо для обшивки корабля наилучшим образом с точки зрения расхода материала, причем время, в течение которого производится раскрой, не имеет особого значения. Здесь с успехом применяются как детерминированные, так и вероятностные методы расчета.

В общей структурной схеме математических основ кибернетики рассматриваются еще специальные прикладные вопросы.

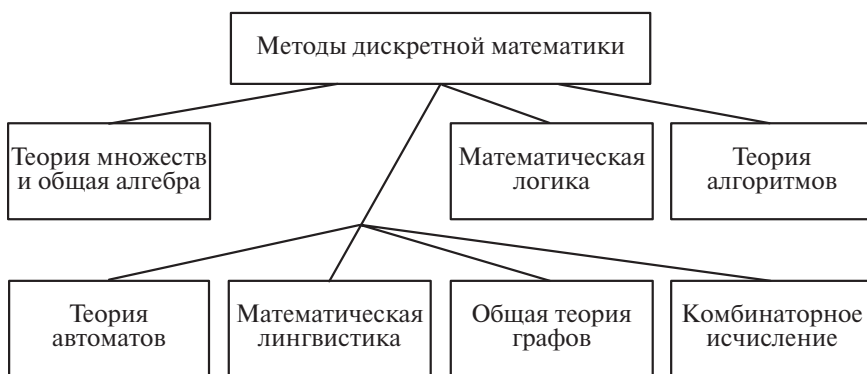


Рис 1.8. Методы дискретной математики в кибернетике

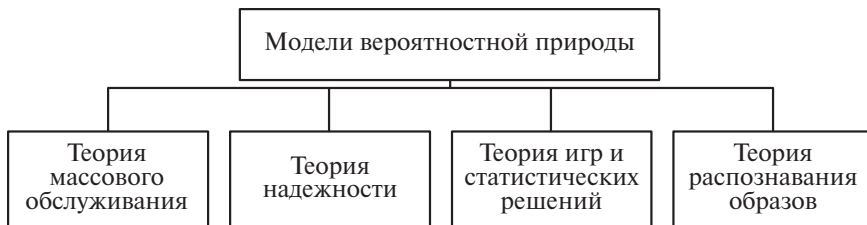


Рис.1.9. Классификация моделей вероятностной природы

Содержание этой части выглядит наиболее неопределенным, так как ее название допускает включение самых разнообразных разделов. Так, вопросы, связанные с проектированием АСУ, выделены в самостоятельный раздел. Хотя они и представляют модификацию больших систем, для них характерен человеко-машинный способ управления. В этом разделе в основном рассматриваются общие вопросы проектирования АСУ и особенности проектирования функциональных подсистем.

Раздел «Теория искусственного интеллекта» включает различные аспекты теории принятия решений в больших системах, а также вопросы создания информационных и программных комплексов, моделирующих искусственный разум.

Мощным аппаратом теории автоматизированного управления является моделирование. В основе моделирования лежит теория подобия, которая утверждает, что абсолютное подобие может иметь место лишь при замене одного объекта другим, точно таким же. При моделировании абсолютное подобие не имеет места, и стремятся к тому, чтобы модель достаточно хорошо отображала исследуемую сторону функционирования объекта.

Классификация видов моделирования приведена на рис. 1.10.

По степени полноты модели они делятся на полные, неполные и приближенные. Полные модели идентичны объекту во времени и пространстве. Для неполного моделирования эта идентичность не сохраняется. В основе приближенного моделирования лежит подобие, при котором некоторые стороны функционирования реального объекта не моделируются совсем.

В зависимости от характера изучаемых процессов в системе виды моделирования подразделяются на детерминированные и стохастические, статические, динамические, дискретные, непрерывные и дискретно-непрерывные. Детерминированное моделирование отображает процессы, в которых предполагается отсутствие случайных воздействий. Стохастическое моделирование учитывает вероятностные процессы и события. Статическое моделирование служит для описания поведения объекта в фиксированный момент времени, а динамическое — для исследования объекта во времени. Дискретное, непрерывное и дискретно-непрерывное моделирования используются для описаний процессов, имеющих изменение во времени. При этом оперируют аналоговыми, цифровыми и аналого-цифровыми моделями.

В зависимости от формы представления объекта моделирование классифицируется на мысленное и реальное. Мысленное моделирование применяется тогда, когда модели не реализуемы в заданном интервале времени либо отсутствуют условия для их физического создания (например, ситуации микромира).

Мысленное моделирование реализуется в виде наглядного, символического и математического. При *наглядном моделировании* на базе представлений человека о реальных объектах создаются наглядные модели, отображающие явления и процессы, протекающие в объекте. В основу *гипотетического моделирования* закладывается гипотеза о закономерностях протекания процесса в реаль-

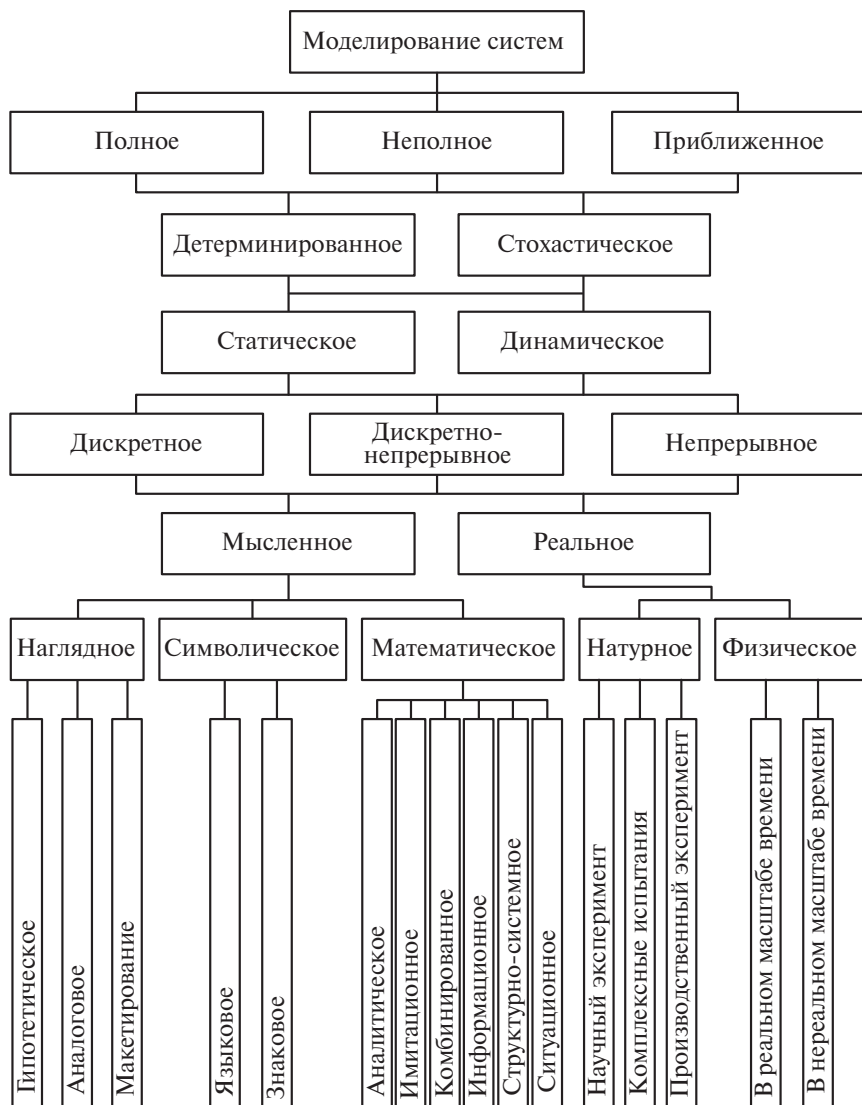


Рис. 1.10. *Виды моделирования систем*

ном объекте, которая отражает уровень знаний исследователя об объекте и базируется на причинно-следственных связях между входом и выходом изучаемого объекта. Этот вид моделирования используется, когда знаний об объекте недостаточно для построения формальных моделей.

Аналоговое моделирование основывается на применении аналогий различных уровней. Для достаточно простых объектов наивысшим уровнем является полная аналогия. С усложнением системы используются аналогии последующих уровней, когда аналоговая модель отображает несколько либо только одну сторону функционирования объекта. *Макетирование* применяется, когда протекающие в реальном объекте процессы не поддаются физическому моделированию либо могут предшествовать проведению других видов моделирования. В основе построения мысленных макетов также лежат аналогии, обычно базирующиеся на причинно-следственных связях между явлениями и процессами в объекте.

Символическое моделирование представляет собой искусственный процесс создания логического объекта, который замещает реальный и выражает основные свойства его отношений с помощью определенной системы знаков и символов. В основе языкового моделирования лежит некоторый тезаурус, который образуется из набора входящих понятий, причем этот набор должен быть фиксированным. Если ввести условное обозначение отдельных понятий, т. е. знаки, а также определенные операции между этими знаками, то можно реализовать знаковое моделирование и с помощью знаков отображать набор понятий — составлять отдельные цепочки из слов и предложений. Используя операции объединения, пересечения и дополнения теории множеств, можно в отдельных символах дать описание какого-то реального объекта.

Математическое моделирование — это процесс установления соответствия данному реальному объекту некоторого математического объекта, называемого математической моделью. В принципе, для исследования характеристик процесса функционирования любой системы математическими методами, включая и машинные, должна быть обязательно проведена формализация этого процесса, т. е. построена математическая модель. Исследование математической модели позволяет получать характеристики рассматриваемого реального объекта. Вид математической модели зависит как от природы реального объекта, так и от задач исследования объекта, требуемой достоверности к точности решения задачи. Любая математическая модель, как и всякая другая, описывает реальный объект с некоторой степенью приближения. Для аналитического моделирования характерно то, что

процессы функционирования элементов системы записываются в виде некоторых функциональных соотношений (алгебраических, интегродифференциальных, конечно-разностных и т. д.) или логических условий. Аналитическая модель исследуется следующими методами: аналитическим, когда стремятся получить в общем виде явные зависимости, связывающие искомые характеристики с начальными условиями, параметрами и переменными системы; численным, когда, не умея решать уравнений в общем виде, стремятся получить числовые результаты при конкретных начальных данных; качественным, когда, не имея решения в явном виде, можно найти некоторые свойства решения (например, оценить устойчивость решения).

В настоящее время распространены методы машинной реализации исследования характеристик процесса функционирования БС. Для реализации математической модели на ЭВМ необходимо построить соответствующий моделирующий алгоритм.

При *имитационном моделировании* реализующий модель алгоритм воспроизводит процесс функционирования системы во времени, причем имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры и последовательности протекания во времени, что позволяет по исходным данным получить сведения о состояниях процесса в определенные моменты времени, дающие возможность оценить характеристики системы. Основным преимуществом имитационного моделирования по сравнению с аналитическим является возможность решения более сложных задач. Имитационные модели позволяют достаточно просто учитывать такие факторы, как наличие дискретных и непрерывных элементов, нелинейные характеристики элементов системы, многочисленные случайные воздействия и др., которые часто создают трудности при аналитических исследованиях. В настоящее время имитационное моделирование — наиболее эффективный метод исследования БС, а часто и единственный практически доступный метод получения информации о поведении системы, особенно на этапе ее проектирования.

В имитационном моделировании различают метод статистического моделирования и метод статистических испытаний (Монте-Карло). Если результаты, полученные при воспроизведении на имитационной модели, являются реализациями случайных величин и функций, тогда для нахождения характеристик процесса требуется его многократное воспроизведение с последующей обработкой информации. Поэтому целесообразно в качестве метода машинной реализации имитационной модели использовать метод статистического моделирования. Первоначально

был разработан метод статистических испытаний, представляющий собой численный метод, который применялся для моделирования случайных величин и функций, вероятностные характеристики которых совпадали с решениями аналитических задач (такая процедура получила название метода Монте-Карло). Затем этот прием стали применять и для машинной имитации с целью исследования характеристик процессов функционирования систем, подверженных случайным воздействиям, т. е. появился метод статистического моделирования.

Метод имитационного моделирования применяется для оценки вариантов структуры системы, эффективности различных алгоритмов управления системой, влияния изменения различных параметров системы. Имитационное моделирование может быть положено в основу структурного, алгоритмического и параметрического синтеза ЕС, когда требуется создать систему с заданными характеристиками при определенных ограничениях. Система должна быть оптимальной по некоторым критериям эффективности.

Комбинированное (аналитико-имитационное) моделирование позволяет объединить достоинства аналитического и имитационного моделирования. При построении комбинированных моделей производится предварительная декомпозиция процесса функционирования объекта на составляющие подпроцессы и для тех из них, где это возможно, используются аналитические модели, а для остальных подпроцессов строятся имитационные модели. Такой подход позволяет охватить качественно новые классы систем, которые не могут быть исследованы с использованием только аналитического или имитационного моделирования в отдельности.

Информационное моделирование (часто называемое кибернетическим) связано с исследованием моделей, в которых отсутствует непосредственное подобие физических процессов, происходящих в моделях, реальным процессам. В этом случае стремятся отобразить лишь некоторую функцию и рассматривают реальный объект как «черный ящик», имеющий ряд входов и выходов, и моделируются некоторые связи между выходами и входами. Таким образом, в основе информационных (кибернетических) моделей лежит отражение некоторых информационных процессов управления, что позволяет оценить поведение реального объекта. Для построения модели в этом случае необходимо выделить исследуемую функцию реального объекта, попытаться формализовать эту функцию в виде некоторых операторов связи между входом и выходом и воспроизвести данную функцию на имитационной модели, причем на совершенно другом математическом языке и, естественно, иной физической реализации процесса.

Структурно-системное моделирование базируется на некоторых специфических особенностях структур определенного вида, используя их как средство исследования систем или разрабатывая на их основе с применением других методов формализованного представления систем (теоретико-множественных, лингвистических и т. п.) специфические подходы к моделированию.

Структурно-системное моделирование включает:

- методы сетевого моделирования;
- сочетание методов структуризации с лингвистическими (языковыми);
- структурный подход в направлении формализации построения и исследования структур разного типа (иерархических, произвольных графов) на основе теоретико-множественных представлений, понятия номинальной шкалы и теории измерений.

Ситуационное моделирование основано на теории мышления, в рамках которой можно описать механизмы регулирования процессов принятия решений. В модельной теории мышления лежит представление о формировании в структурах мозга информационной модели объекта и внешнего мира. Эта информация воспринимается человеком на базе уже имеющихся у него знаний. Целесообразное поведение человека строится путем формирования целевой ситуации и мысленного преобразования исходной ситуации в целевую. Основой построения модели является описание объекта в виде совокупности элементов, связанных между собой определенными отношениями, отображающими семантику предметной области. Модель объекта имеет многоуровневую структуру и представляет собой тот информационный контекст, на фоне которого протекают процессы управления. Чем богаче информационная модель объекта и выше возможности ее манипулирования, тем лучше и многообразнее качество принимаемых решений при управлении.

При реальном моделировании используется возможность исследования характеристик либо на реальном объекте целиком, либо на его части. Такие исследования на объектах, работающих как в нормальных режимах, так и при организации специальных режимов для оценки интересующих исследователя характеристик (при других значениях переменных параметров, в другом масштабе времени и т. д.). Реальное моделирование является наиболее адекватным, но его возможности ограничены. Например, проведение реального моделирования АСУП требует проведения экспериментов с управляемым объектом, т. е. предприятием, что в большинстве случаев невозможно.

Натурным моделированием называют проведение исследования на реальном объекте с последующей обработкой результатов эксперимента на основе теории подобия. Натурный эксперимент подразделяется на научный эксперимент, комплексные испытания и производственный эксперимент. Научный эксперимент характеризуется широким использованием средств автоматизации проведения, применением весьма разнообразных средств обработки информации, возможностью вмешательства человека в процесс проведения эксперимента. В соответствии с этим появилось новое научное направление — автоматизация научного эксперимента и новая специализация в рамках специальности АСУ — АСНИ (автоматизированные системы научных исследований и комплексных испытаний). Одна из разновидностей эксперимента — комплексные испытания, когда вследствие повторения испытаний объектов в целом (или больших частей системы) выявляются общие закономерности характеристик качества, надежности этих объектов. В этом случае моделирование осуществляется путем обработки и обобщения сведений о группе однородных явлений. Наряду со специально организованными испытаниями возможна реализация натурального моделирования путем обобщения опыта, накопленного в ходе производственного процесса, т. е. можно говорить о производственном эксперименте. Здесь на базе теории подобия обрабатывают статистический материал по производственному процессу и получают его обобщенные характеристики. Необходимо помнить про отличия эксперимента от реального протекания процесса. Оно заключается в том, что в эксперименте могут появиться отдельные критические ситуации и определиться границы устойчивости процесса. В ходе эксперимента новые факторы и возмущающие воздействия вводят в процесс функционирования объекта.

Другим видом реального моделирования является физическое, отличающееся от натурального тем, что исследование проводится на установках, которые сохраняют природу явлений и обладают физическим подобием. В процессе физического моделирования задаются некоторые характеристики внешней среды и исследуется поведение либо реального объекта, либо его модели при заданных или создаваемых искусственно воздействиях внешней среды. Физическое моделирование может протекать в реальном и нереальном (псевдореальном) масштабах времени или рассматриваться без учета времени. В последнем случае изучению подлежат так называемые «замороженные» процессы, фиксируемые в некоторый момент времени. Наибольшую сложность и интерес с точки зрения корректности получаемых результатов представляет физическое моделирование в реальном масштабе времени.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каковы причины появления автоматизированного управления?
2. Расскажите историю становления и развития теории автоматизированного управления.
3. Как Вы понимаете смысл определений управления в широком смысле?
4. Что такое «алгоритм управления»?
5. Сформулируйте понятие «процесс управления».
6. В каких взаимоотношениях находятся объект управления и управляющий орган?
7. Перечислите задачи, решаемые системой управления.
8. В чем состоит принципиальная разница между автоматическим и автоматизированным управлениями?
9. Раскройте понятие «система», «элемент», «подсистема».
10. Поясните содержание понятий «структура» и «связь».
11. Что такое «иерархия»?
12. Что вкладывается в понятия «состояние», «поведение» и «модель» системы?
13. Раскройте понятие «цель». Какова его роль в управлении?
14. Большая (сложная) система как основной вид систем в теории автоматизированного управления.
15. Назовите и поясните характерные особенности больших систем.
16. Перечислите основные этапы управления.
17. Что является объектом теории автоматизированного управления?
18. Как Вы понимаете предмет теории автоматического управления?
19. Раскройте содержание трех циклов математических основ теории автоматизированного управления и кибернетики.
20. Расскажите о вероятностных методах в кибернетике.
21. Какие Вы знаете методы оптимизации, применяемые в кибернетике?
22. Дайте классификацию методов дискретной математики и поясните их роль в теории автоматизированного управления.
23. Охарактеризуйте основные кибернетические модели вероятностной природы.
24. Какое различие Вы можете найти между вероятностными и детерминированными моделями?
25. Какая роль в кибернетике отводится дискретным моделям?
26. Расскажите о важности кибернетических моделей в проектировании АСУ.
27. Что Вы знаете о теории искусственного интеллекта?
28. Какова роль моделирования в теории автоматизированного управления?
29. По каким признакам классифицируется моделирование?
30. Дайте классификацию основных видов моделирования.
31. Поясните роль математического моделирования.
32. Что такое имитационное и ситуационное моделирование?
33. Расскажите, что Вы знаете о структурно-системном моделировании.

Глава 2

МЕТОДОЛОГИЯ ПОСТРОЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Автоматизированные системы управления представляют собой сложные человеко-машинные комплексы. Это совокупность крупных подсистем, симбиоз данных и знаний, экономико-математических моделей, инструментальных и технических средств, средств связи и оргтехники, а также специалистов, предназначенных для обработки информации и принятия решений. Поэтому в этой главе приводится подробная классификация АС по различным признакам. Знание классификации АС позволяет студентам свободно ориентироваться в большом многообразии АСУ. Здесь же приводятся основные методологические принципы построения АС, которые существенно отличаются от методов проектирования в общем машиностроении, энергетике, радиотехнике, электронике. Во второй главе также рассмотрены этапы разработки АС, отражающие как порядок и последовательность создания АСУ, так и многочисленный накопленный полувекковой опыт. Заключает главу список задач, решаемых на всех стадиях проектирования АС. Таким образом, содержание второй главы помогает студентам глубоко понять концептуальный процесс создания АСУ. При этом студент должен уяснить одну очень важную мысль: современные информационные технологии реализуются в АС, составляя их главную, содержательную (семантическую) часть.

2.1. Классификация автоматизированных систем

В настоящее время АС получили широчайшее распространение. Их классификация осуществляется по ряду признаков. В зависимости от решаемой задачи можно выбрать разные признаки классификации. При этом одна и та же АС может характеризоваться одним или несколькими признаками. В качестве признаков классификации АС используются следующие признаки:

- направление деятельности;
- область и специфика применения;
- охватываемая территория;

- организация информационных процессов;
- назначение;
- структура и др.

Укрупненная (общая, глобальная) классификация АС. При самом общем (глобальном) рассмотрении АС ее можно представить состоящей из двух частей: функциональной и обеспечивающей. Например, автоматизированным системам организационного типа, таким, как автоматизированные системы управления предприятием (АСУП), системам управления объединением, фирмой, отраслью присуще наличие очень большого числа различных целей, которые система стремится достичь одновременно. У АСУП можно выделить следующие цели: повышение вероятности выполнения плана, снижение себестоимости продукции, повышение качества продукции, выход продукции на международный рынок, повышение престижа предприятия в административно-территориальном районе и т. п. Между целями могут существовать как взаимоподдержка, так и состязательность. Взаимоподдержка выражается в том, что достижение одной цели способствует достижению другой (или других) целей; состязательность выражается в том, что ради большего достижения других целей приходится поступиться степенью достижения данной цели.

В свою очередь, каждая из глобальных целей может быть разбита на некоторое множество локальных целей или целей более низкого уровня. Аналогично каждая из локальных целей может быть разбита на некоторое множество подцелей следующего уровня иерархии.

Функциональная подсистема — это часть автоматизированной системы, которой поставлена в соответствие одна или несколько целей (подцелей) системы управления. Таким образом, функциональная часть АС — это некоторый набор функциональных подсистем. В самом простейшем случае функциональная подсистема состоит из управляющей части и объекта управления (рис. 2.1).

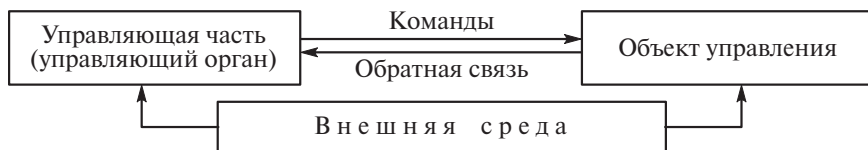


Рис. 2.1. Общее представление функциональной части АС

Управляющая часть воздействует на объект управления посредством выдачи команд, желая привести объект управления в некоторое требуемое состояние. *Команды* — это распорядительная информация. Поскольку управляющей части безразлично состояние объекта управления, то всегда присутствует обратная связь, это — осведомительная информация.

Взаимодействие управляющей части с объектом управления осуществляется в некоторой среде, которая в общем случае вредит управлению.

Применительно к АСУП традиционно выделяют следующие функциональные подсистемы:

- технической подготовки производства;
- технико-экономического планирования;
- оперативного управления производством;
- материально-технического снабжения;
- управления кадрами;
- управления качеством продукции;
- финансовая подсистема и др.

Цели этих подсистем следуют из их названия.

Применительно к АСУ ВУЗ можно выделить следующие функциональные подсистемы: абитуриент, расписание, текущая успеваемость, экзаменационная сессия и т. п.

Практически в любой подсистеме функциональной части АС решаются следующие функциональные задачи:

- планирование, т. е. разработка расписания деятельности объекта управления на некоторый календарный отрезок времени;
- контроль, т. е. сбор первичной информации о состоянии объекта управления и внешней среды;
- регулирование, т. е. сопоставление собранного круга данных с некоторыми запланированными (или нормативными) величинами;
- выдача управляющих воздействий — команд, подаваемых на объект управления в случае отклонения реальных параметров производственного процесса от запланированных или нормативных величин.

Применительно к АСУП управляющие воздействия могут быть:

- экономические (выдача заработной платы, премий, начисление штрафов);
- технологические (введение нового оборудования, изменение существующей технологии);
- административные (объявления благодарностей, административных взысканий и т. д.).

При решении любой из перечисленных глобальных функциональных задач из структуры системы может быть выделена часть, которая обеспечивает их решение — *обеспечивающая часть АС*.

Она включает в себя математическое, информационное, программное, техническое, лингвистическое и другие виды обеспечения АС.

Иногда в литературе обеспечивающую часть называют автоматизированной системой обработки данных (или АСОД) или информационно-вычислительной системой (ИВС).

Математическое обеспечение — это набор математических формул, соотношений, алгоритмов, математических моделей, методик, предназначенных для решения задач управления и обработки информации.

Информационное обеспечение применительно к АСУП — совокупность единой системы классификаторов, кодов технико-экономической информации, унифицированной системы документации, а также массивов информации, используемых в АС. Проще говоря, информационное обеспечение — это вся информация, используемая для решения задач управления и обработки информации.

Программное обеспечение — это набор рабочих программ, пакетов программ, пакетов прикладных программ, программных комплексов и т. п. Проще говоря, это все программы, используемые для решения задач управления и обработки информации с помощью ЭВМ.

Техническое обеспечение — все технические средства, используемые для автоматизированного решения задач управления и обработки информации.

Лингвистическое обеспечение — это набор языковых средств, реализующий дружественный интерфейс между пользователем и ЭВМ в целях повышения эффективности общения человека с машиной.

Классификация АС по направлению деятельности (области применения) показана на рис. 2.2. Как видно из этого рисунка, в промышленной сфере превалирует иерархическое построение АС.

С этих позиций прежде всего можно выделить два очень больших класса систем — это АСУ технологическими процессами (АСУТП) и АС организационного типа. В АСУТП объектом управления является технологический процесс, понимаемый в широком смысле этого понятия, это собственно технологичес-

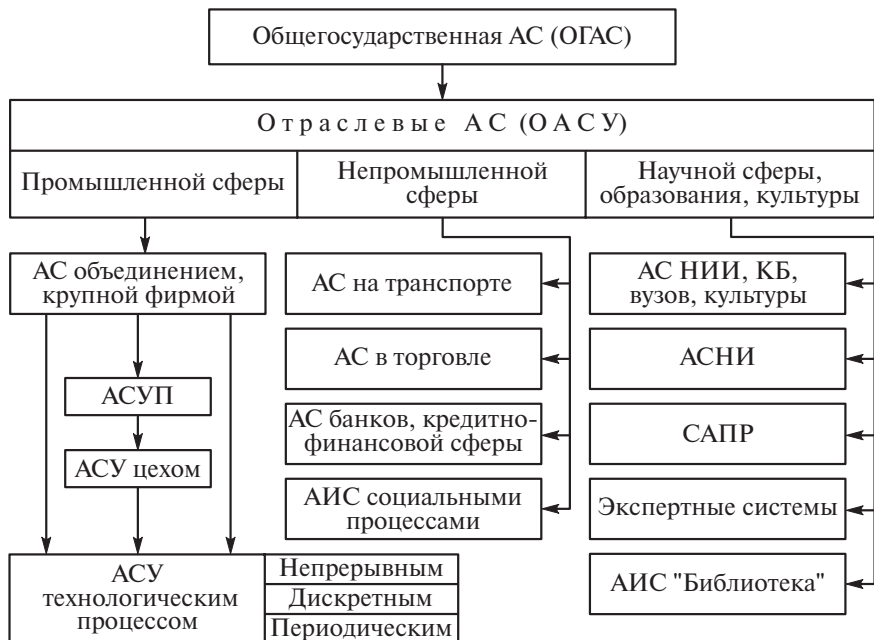


Рис. 2.2. Классификация АС по направлению деятельности (области применения)

кий процесс, а также, например, процесс управления полетом ракеты или самолета, движением корабля, управление химическим или ядерным реактором и т. п. В организационных системах объектом управления является коллектив людей (предприятие, отрасль, дивизия и т. п.). Другое различие между этими системами заключается в виде основного носителя информации. В АСУТП этим носителем является сигнал (электрический, механический, гидравлический, радиосигнал и т. п.), в организационных системах основной носитель — документ. Следующий класс систем — интегрированные системы. Они представляют собой совокупность одной организационной системы и нескольких АСУТП, причем организационная система располагается на верхнем уровне иерархии, а АСУТП — на одном или нескольких нижних уровнях.

По сфере (специфике) применения АС различаются следующим образом (рис. 2.3).

Из этого класса АС исторически первыми стали применяться АС на производстве.

Классификация АС по организации информационных процессов.

В зависимости от организации информационных процессов АС делятся на два больших класса: управляющие и информационные. В информационных системах (ИС) управление отсутствует, например: автоматизированные системы научных исследований — АСНИ, «Библиотека», системы автоматизированного проектирования — САПР, экспертные системы — ЭС и др. В отличие от чисто информационных систем в таких АИС, как автоматизированные системы управления технологическими процессами — АСУТП, АСУ предприятиями — АСУП, управление занимает важное место и бывает либо автоматическим, либо автоматизированным.

Информационно-поисковые системы (ИПС) — в них объектом управления является процедура поиска требуемой информации в очень больших объемах этой информации. Типичный пример — различные библиотечные системы, системы продажи билетов на транспортные средства и т. п.

Системы автоматизированного проектирования (САПР) — в них объектом управления является процесс проектирования изделий любой природы (станка, самолета, ЭВМ, АСУ и т. п.).

Следующий класс систем — АС научных исследований и комплексных испытаний (АСНИ). Здесь объектом управления является процесс исследования объекта любой работы (исследования процесса работы двигателя, полета самолета, работы реактора и т. п.).

В последнее время активно развиваются гибкие автоматизированные производства (ГАП). ГАП — некоторая производственная единица, функционирующая на основе безлюдной технологии и находящаяся под управлением единой программы. Переналадка производства (естественно, в некоторых пределах) с выпуска одного изделия на другое сводится к замене только программного обеспечения.

Для всех этих классов систем характерны общие черты АС, а именно наличие всех вышеперечисленных видов обеспечения и человека как основного звена, принимающего решения. Все вышеперечисленные классы систем оперируют с данными, или, проще говоря, с некоторыми цифрами, хранящимися в памяти системы.



Рис. 2.3. *Классификация АС по специфике применения*



Рис. 2.4. Классификация АС по назначению

Интеллектуальные системы (экспертные системы) в отличие от предыдущих систем оперируют со знаниями, хранящимися в банке знаний. Типичные примеры — это медицинские экспертные системы, геологические экспертные системы и т. п. База знаний (или банк знаний) формируется в результате обобщения знаний ведущих ученых, практиков, а также информации, хранящейся в монографиях, статьях, книгах, и т. п.

Классификация АС по назначению. Данная классификация приведена на рис. 2.4.

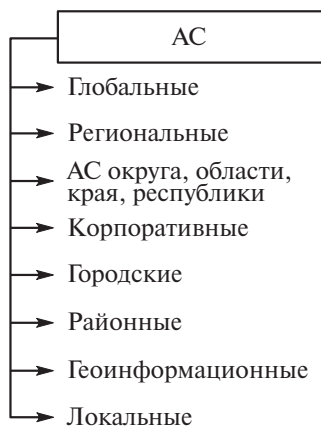


Рис. 2.5. Классификация АС по территориальному признаку

Классификация АС по территориальному признаку. Эта классификация показана на рис. 2.5. АС различаются также по уровню развития в зависимости от поколений ЭВМ, на которых они базируются.

Следует отметить информационные технологии (ИТ) и АС, с помощью которых регулируются микро-социальные процессы. Это ИТ и АС денежно-кассовых операций, распределения мест на транспорте, в гостиницах, метеорологической и другой справочной информации.

Разнообразие ИТ и АС постоянно растет.

2.2. Основные принципы построения автоматизированных систем

Накопленный опыт разработки и эксплуатации АС позволяет сформулировать ряд принципов их построения, соблюдение которых является необходимым условием создания эффективных систем. Мы рассмотрим эти принципы применительно к системам управления производством, но они в полной мере применимы и к системам других классов.

1. Принцип системного подхода.

Это основополагающий принцип. Суть его заключается в том, что проектируемый объект должен рассматриваться с позиций более высокого уровня. Так, например, проектируемая задача должна рассматриваться с позиций функциональной подсистемы, в которую она входит; проектируемая подсистема — с позиций системы и т. п.

Проектирование автоматизированной системы должно начинаться с тщательного системного анализа объекта управления, управляющей части и внешней среды. Необходимо выяснить все множество факторов, под влиянием которых находится система, а также все множество факторов, на которые влияет сама система. Параллельно с этим необходимо выяснить все множество целей, стоящих перед проектируемой системой. Для каждой цели необходимо разработать один или несколько критериев эффективности, которые являются численной мерой степени достижения целей. Необходимо вскрыть весь комплекс вопросов, которые необходимо решить для того, чтобы проектируемая система наилучшим образом соответствовала бы поставленным целям и критериям эффективности. Применительно к АС организационного управления производственными процессами в этот комплекс вопросов должны включаться не только технические вопросы, но также вопросы экономические и организационные. Внедрение АС должно сопровождаться усовершенствованием экономических показателей и методов экономического стимулирования, а также изменением существующих и узаконенных форм документов, изменением маршрутов их движения, изменением функциональных обязанностей работников аппарата управления и т. п.

2. Принцип новых задач.

Суть его заключается в том, что совершенно недостаточно ограничиться тем, чтобы переложить на ЭВМ и другие технические средства сложившиеся формы, методы и задачи управления. Главное внимание следует уделить тем огромным возможностям, которые открывает использование современной вычислительной техники и программного обеспечения. Особое внимание следует обратить на те задачи, которые в существующей системе управ-

ления вследствие большого объема или вычислительных сложностей не решаются или решаются в неполной степени.

3. Принцип первого руководителя.

Успешная реализация двух первых принципов возможна лишь в том случае, если разработка и внедрение АС находятся в непосредственном ведении первых лиц организации заказчика (директор или главный инженер). При этом на системотехника возлагается задача четкого распределения функций между организацией заказчика и организацией разработчика.

Функциями заказчика являются:

- формулировка целей системы, критериев эффективности, общей концепции системы (совместно с руководящим составом организации разработчика),
- определение приоритетов и очередности ввода различных задач управления (совместно с разработчиками системы),
- участие в разработке информационной базы системы,
- реализация организационных мероприятий (изменение структуры и функций аппарата управления), выделение соответствующих площадей под технические средства системы, организация финансирования разработки, выделение соответствующих штатных единиц для персонала, обслуживающего разработанную систему.

Функции разработчика (помимо перечисленных выше):

- разработка технического задания на проектируемую систему (совместно с руководством организации заказчика),
- разработка технического проекта (разработка структуры системы, алгоритмов решения задач, информационной базы каждой задачи, выбор комплекса технических средств),
- разработка рабочего проекта (разработка форм документов, рабочих программ, инструкций по эксплуатации),
- внедрение разработанной системы в эксплуатацию (совместно с персоналом, эксплуатирующим систему).

4. Принцип непрерывного развития системы.

Он предусматривает возможность введения новых задач и совершенствования уже внедренных задач без ухудшения качества решения эксплуатируемых задач и тем более без исключения возможности решения хотя бы одной эксплуатируемой задачи. Системы, обладающие этими качествами, называют открытыми системами.

5. Принцип разумной типизации проекта.

Разрабатывая столь дорогостоящие изделия, каким является автоматизированная система, системотехник, естественно, стремится к тому, чтобы предлагаемые им решения подходили бы как можно более широкому кругу заказчиков. Однако типизация, естественно, приводит к ухудшению предлагаемых решений, пос-

кольку она не позволяет учитывать всю специфику объекта управления. На первых этапах разработки АС была попытка разработки универсальной программы для подсистемы материально-технического снабжения. Эта программа оказалась очень медленно действующей в силу своей универсальности. Применительно к этому примеру принцип «разумной типизации» заключается в разумном увеличении скорости выполнения конкретной программы по сравнению с универсальной.

6. Принцип автоматизации документооборота.

В автоматизированных системах совершенно недостаточно ограничиться выполнением расчетов на ЭВМ по тем или иным моделям, необходимо автоматизировать все стадии обработки информации, а именно сбор первичной информации, ее передачу, обработку, хранение и доведение полученных результатов до конкретных пользователей данной АС.

7. Принцип единой информационной базы.

Суть его заключается в том, что на магнитных носителях накапливается и постоянно обновляется информация, необходимая для решения не отдельных, а всех задач управления.

8. Принцип однократности ввода и многократности использования информации.

Он непосредственно следует из предыдущего принципа. Информация о любом документе, объекте или событии должна вводиться в систему только один раз. Невыполнение этого принципа приводит к тому, что, например, об одном и том же событии может появиться несколько противоречивых мнений, что засоряет память системы и неизбежно выводит ее из строя. Многократность использования означает, что на любой уровень управления, от министра до начальника участка, информация должна поступать из единой информационной базы. При этом, конечно, формы представления этой информации, степень ее детализации и т. п. для каждого уровня должны быть различными.

9. Принцип комплексности задач и рабочих программ.

Большинство задач, решаемых в рассматриваемых системах, тесно связаны между собой, например задачи подсистем технико-экономического планирования и материально-технического снабжения. Между этими подсистемами идет постоянный обмен информацией и раздельное решение этих задач существенно снижает эффективность всей системы.

10. Принцип согласованности пропускных способностей различных элементов системы.

В простейшем случае для последовательных участков системы пропускная способность каждого последующего элемента должна быть не меньше, чем у предыдущего.

2.3. Этапы разработки АС

Чрезвычайно важно при разработке АС соблюдать определенный порядок этой разработки, отражающий многочисленный накопленный опыт.

Этапы разработки удобно иллюстрировать в виде сетевых графиков, в которых кружки отражают события, а стрелки — действия (процессы, операции). Подробный сетевой график может содержать до тысячи операций, поэтому рассмотрим укрупненный сетевой график (рис. 2.6).

Операция ①→②. Как правило, разработка АС начинается с предварительного ознакомления с существующей на данном объекте системой управления. Целью этого ознакомления является определение целесообразности разработки АС на данном объекте. Эту работу выполняет группа (4—5 системотехников высшей квалификации). На этом этапе в самом общем виде формулируются цели предполагаемой АС и намечаются возможные пути повышения эффективности управления за счет автоматизации. Работа заканчивается предоставлением докладной записки руководству организациям заказчика и разработчика. Этот этап может отсутствовать, если имеется директивное решение вышестоящей организации (министерства, ведомства, фирмы и т. п.).

Операция ②→③ — формирование коллективов у разработчика и заказчика. Здесь осуществляется укрупненное изучение существующей системы управления. В работе принимает участие старший состав системотехников. Глобальная цель этого этапа — уточнение целей управления, анализ критериев эффективности, с помощью которых количественно оценивается степень достижения целей, анализ ограничений как по режимам функционирования системы, так и по ресурсам, необходимым на ее разработку (финансы, специалисты, техника, время). В результате выпускается документ «Технико-экономическое обоснование», утверждаемый у руководства организаций заказчика и разработчика.

Операция ③→④ — детальный анализ существующей системы управления. Работа выполняется рядовым составом системотехников. Здесь подробнейшим образом изучается организационная структура автоматизируемого объекта, его функциональная структура, все формы документов, циркулирующих в процессе управления, все информационные потоки (способы доставки документов, частота поступления каждого документа, трудоемкость обработки каждого документа и т. п.), методы принятия управленческих решений, схема распределения ответственности, нормативная база объекта автоматизации, схема всех материальных

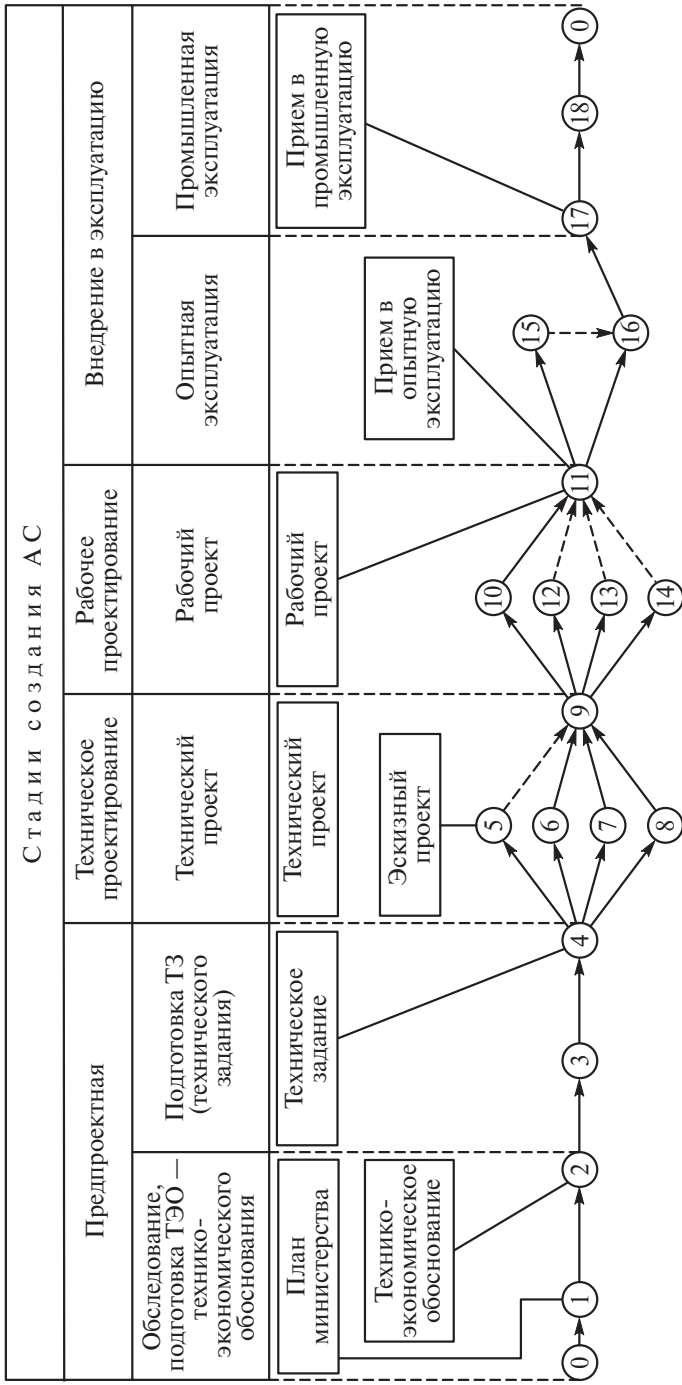


Рис. 2.6. Укрупненный сетевой график

потоков, начиная от поступления сырья и полуфабрикатов и кончая доведением готовой продукции до потребителя.

Операция ③→④ — разработка технического задания (ТЗ) на создаваемую автоматизированную систему. ТЗ содержит описание основных целей создания системы, критериев ее функционирования, назначение и особенности данной системы. В ТЗ указывается состав и характеристики комплексов решаемых задач, а также состав информационного, математического, программного, лингвистического и технического обеспечения. ТЗ — это официальный документ, определяющий требования и сроки создания системы. ТЗ в обязательном порядке согласуется и утверждается у руководства организаций разработчика и заказчика.

Операция ④→⑤ — эскизное проектирование системы. Этот этап имеет место только при проектировании систем, не имеющих аналогов. Основные цели эскизного проектирования: проинформировать руководство о возможных проектных решениях, подготовить сотрудников организации заказчика к переобучению, уточнить требования к структуре системы и ее обеспечивающим подсистемам. Наличие альтернативных вариантов при эскизном проектировании обязательно.

Операция ④→⑥. Специализированные группы ведут разработку одной или нескольких функциональных подсистем (перечень задач, их постановка, алгоритмизация, информационный базис и т. п.).

Операция ④→⑦ — обоснование и выбор комплекса технических средств.

Операция ④→⑧ — предварительный расчет экономической эффективности.

Событие ⑨ — работа всех групп сводится к выпуску технического проекта, его корректировке, согласованию, утверждению.

Операция ⑨→⑩ — разработка и отладка рабочих программ.

Операция ⑩→⑪ — связанная отладка комплексов программ по задачам.

Операция ⑨→⑫ — разработка и выпуск инструкций по эксплуатации технических средств.

Операция ⑨→⑬ — разработка и выпуск рабочих инструкций персоналу автоматизированной системы.

Операция ⑨→⑭ — уточненный расчет экономической эффективности.

Событие ⑪ — выпуск рабочего проекта.

Операция ⑪→⑮. Если технические средства были подготовлены заранее, то проводится опытная эксплуатация системы, если нет, то операция ⑪→⑯ — монтаж и отладка технических средств.

Операция ⑯→⑰ — опытная эксплуатация системы на подготовленных средствах.

Операция ⑰→⑱ — передача в промышленную эксплуатацию.

Операция ⑱→① — возможна частичная модернизация и доработка системы.

Если рассматривать процесс создания системы еще более укрупненно, то можно выделить четыре ГОСТИрованных этапа (стадии):

1-я стадия — предпроектная (ее еще называют стадией ТЗ), здесь два крупных направления работ — это обследование и выпуск ТЗ.

2-я стадия — стадия технического проектирования. Основные работы: разработка структуры системы, разработка перечня задач, решаемых в каждой функциональной подсистеме, постановка и алгоритмизация задач, обоснование и выбор комплекса технических средств, разработка схемы документооборота, разработка структуры и состава информационного, программного и лингвистического обеспечения, расчет экономической эффективности.

3-я стадия — рабочее проектирование. Основные работы: программирование, отладка программ, выпуск комплекта программной документации, выпуск инструкций по эксплуатации технических средств, выпуск должностных инструкций, уточненный расчет экономической эффективности.

4-я стадия — внедрение. Здесь проводится опытная эксплуатация системы совместно с разработчиками этой системы. Затем промышленная эксплуатация, выполняемая силами работников объекта автоматизации.

2.4. Задачи, решаемые на стадиях проектирования АС

Целью данного параграфа является достаточно формальное изложение задач проектирования, распределенных по этапам проектирования. Содержание этого раздела уточняет, дополняет и детализирует материал предыдущего параграфа.

В табл. 2.1 приведены основные направления работ, выполняемых на стадии технического задания.

Стадия технического задания

Этапы	Основные направления работ
Системный анализ проблемной области	<ol style="list-style-type: none"> 1. Изучение целей проектируемой системы 2. Изучение организационной структуры 3. Изучение источников и потребителей информации 4. Изучение входной, промежуточной и выходной информации 5. Изучение методов обработки данных, решения задач и принятия решений
Разработка технического задания	<ol style="list-style-type: none"> 1. Формулировка цели и назначения разработки 2. Формулировка технических требований к общей структуре, автоматизированным подсистемам и задачам, комплексу технических средств, математическому и программному обеспечению, информационному обеспечению, лингвистическому обеспечению, средствам сбора и передачи информации, технологическому процессу обработки информации 3. Формулировка специальных требований 4. Документирование

В табл. 2.2 отражены задачи стадии технического проектирования.

Стадия технического проектирования

Этапы	Основные направления работ
Проработка общесистемных решений	<ol style="list-style-type: none"> 1. Разработка общей структурной схемы 2. Разработка способов сопряжения с внешними абонентами и другими системами 3. Разработка общего алгоритма функционирования 4. Документирование
Разработка функциональных подсистем и задач	<ol style="list-style-type: none"> 1. Разработка целевого назначения и общей характеристики подсистемы 2. Разработка функциональной схемы 3. Разработка информационной базы 4. Разработка математической модели 5. Алгоритмизация 6. Документирование
Разработка информационного обеспечения	<ol style="list-style-type: none"> 1. Разработка состава и структуры информационной базы 2. Распределение информации на машинных носителях 3. Организация обмена данными 4. Организация сбора, передачи и обработки информации 5. Разработка системы классификации и кодирования информации 6. Разработка форм документов 7. Документирование

Этапы	Основные направления работ
Разработка программного обеспечения	<ol style="list-style-type: none"> 1. Разработка структуры общего программного обеспечения 2. Разработка структуры операционной системы и методов организации вычислительного процесса 3. Разработка алгоритмов обмена информацией с внешними абонентами по каналам связи 4. Разработка алгоритмов функционального контроля программного комплекса технических средств 5. Разработка алгоритмов разграничения доступа к информации 6. Разработка состава и структуры специального программного обеспечения 7. Разработка пакета прикладных программ 8. Документирование
Проектирование комплекса технических средств	<ol style="list-style-type: none"> 1. Обоснование и выбор состава и структуры комплекса технических средств 2. Разработка вычислительной системы 3. Разработка системы сбора и телеобработки 4. Разработка системы отображения информации 5. Оценка надежности комплекса технических средств 6. Документирование

В табл. 2.3 отражены основные направления работ, выполняемых на стадии рабочего проектирования.

Стадия рабочего проектирования

Таблица 2.3

Этапы	Основные направления работ
Разработка программного обеспечения	<ol style="list-style-type: none"> 1. Разработка текстов программ 2. Разработка контрольного примера 3. Разработка программной документации: руководства программиста, руководства оператора, эксплуатационной программы и др. 4. Документирование
Разработка технологического процесса функционирования объекта проектирования	<ol style="list-style-type: none"> 1. Разработка технологических процессов: сбора и передачи информации, обработки данных, оценки загрузки комплекса технических средств 2. Документирование
Разработка комплекса технических средств	<ol style="list-style-type: none"> 1. Проектирование общего вида технических устройств 2. Изготовление сборочных чертежей 3. Изготовление принципиальных схем 4. Разработка технической документации: технического описания, инструкции по эксплуатации 5. Документирование

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. По каким признакам классифицируются автоматизированные системы?
2. Расскажите о представлении автоматизированных систем в виде функциональной части.
3. Какие подсистемы входят в обеспечивающую часть АС ?
4. Как классифицируются АС по области применения?
5. Объясните классификацию АС по специфике применения.
6. Приведите примеры АС с различной организацией информационных процессов.
7. Дайте характеристику АС по назначению.
8. Какие АС по территориальному признаку вы знаете?
9. Что такое принципы системного подхода и новых задач в методологии построения АС?
10. В чем заключается принцип первого руководителя?
11. Как Вы понимаете принципы непрерывного развития системы и разумной типизации проекта АС?
12. Сочетаются ли между собой принципы автоматизации документооборота и единой информационной базы АС?
13. Какие существуют этапы разработки АС?
14. Объясните характер работ на предпроектной стадии создания АС.
15. Перечислите основные направления работ, выполняемых разработчиком АС на стадии технического задания.
16. Каков перечень пяти основных подэтапов стадии технического проектирования АС?
17. Дайте характеристику направлениям работ на стадии создания технического проекта.
18. Что такое рабочий проект АС и какие работы характерны для этой стадии разработки АС?
19. В чем разница технического и рабочего проектирования АС?
20. Чем характерен этап внедрения АС в эксплуатацию?

Глава 3

КАТЕГОРИАЛЬНЫЕ ПОНЯТИЯ СИСТЕМНОГО АНАЛИЗА АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Так как большие (сложные) системы в середине XX века стали объектом изучения, проектирования и управления, то потребовалось обобщение методов исследования таких систем. Появилась объективная необходимость в возникновении прикладной науки, устанавливающей связь между абстрактной теорией систем и системной практикой. Это оформилось в научную дисциплину — *системный анализ*.

Системный анализ является междисциплинарной наукой, обобщающей методологию исследования сложных технических, биологических и социальных систем. Для проведения анализа и синтеза сложных систем используется широкий спектр математических методов и моделей. Основу математического аппарата системного анализа составляют линейное и нелинейное программирование, теория принятия решений, теория игр и исследования операций, имитационное и ситуационное моделирование, теория массового обслуживания, теория надежности, теория статистических решений и др.

В настоящее время методы системного анализа получили широкое применение при перспективном и текущем планировании, проектировании различных сложных систем и объектов, управлении производственными и технологическими процессами, прогнозировании отраслей промышленности и сельского хозяйства. Особенно часто к методам системного анализа обращаются при решении задач распределения трудовых ресурсов и запасов, выбора средств транспортировки грузов, составления маршрутов и расписаний перевозок, размещения новых производственных комплексов, сбора информации в АСУ и целого ряда других задач. Причем при решении задач системного анализа наряду со строгим математическим аппаратом применяются эвристические, интуитивные и качественные методы.

Так как в процессе исследования реальной системы обычно приходится сталкиваться с разнообразными проблемами, а быть профессионалом в каждом из них одному человеку невозможно, то специалист, занимающийся системным анализом, должен иметь соответствующее образование и опыт, необходимые для анализа конкретных проблем. Это предъявляет особые требования к системным аналитикам в области широты эрудиции и мышления, умения привлекать к работе узких специалистов, организации коллективной деятельности.

3.1. Системность как общее свойство материи

Современные научные данные и современные системные представления позволяют говорить о мире как о бесконечной иерархической системе систем. Причем части системы находятся в развитии, на разных стадиях развития, на разных уровнях системной иерархии и организации. Системность как всеобщее свойство материи проявляется через следующие составляющие: системность практической деятельности, системность познавательной деятельности и системность среды, окружающей человека.

Рассмотрим практическую деятельность человека, т. е. его активное и целенаправленное воздействие на окружающую среду. Покажем, что человеческая практика системна. Отметим очевидные и обязательные признаки системности: *структурированность системы, взаимосвязанность составляющих ее частей, подчиненность организации всей системы определенной цели*. По отношению к человеческой деятельности эти признаки очевидны. Всякое осознанное действие преследует определенную цель. Во всяком действии достаточно просто увидеть его составные части, более мелкие действия. При этом легко убедиться, что эти составные части должны выполняться не в произвольном порядке, а в определенной их последовательности. Это и есть та самая определенная, подчиненная цели взаимосвязанность составных частей, которая и является признаком системности.

Название для такого построения деятельности — алгоритмичность. Понятие алгоритма возникло сначала в математике и означало задание точно определенной последовательности однозначно понимаемых операций над числами или другими математическими объектами. В настоящее время понятие алгоритма применяется к различным отраслям деятельности. Так говорят не только об алгоритмах принятия управленческих решений, об алгоритмах обучения, алгоритмах написания программ, но и об алгоритмах изобретательства. Алгоритмизируются такие виды деятельности, как игра в шахматы, доказательство теорем и т. п. При этом делается отход от математического понимания алгоритма. Важно сознавать, что в алгоритме должна сохраняться логическая последовательность действий. При этом допускается, что в алгоритме определенного вида деятельности могут присутствовать неформализованные виды действия. Важно лишь, чтобы определенные этапы алгоритма успешно, хотя бы и не осознанно, выполнялись человеком.

подавляющее большинство элементов творческой деятельности, совершаемой человеком, являются неосознанной реализацией определенных алгоритмизируемых закономерностей, т. е. реализацией неосознаваемых, но объективно существующих и формализуемых критериев. Таким образом, во-первых, всякая деятельность алгоритмична. Во-вторых, не всегда алгоритм реальной деятельности осознается, ряд процессов человек выполняет интуитивно, т. е. его способность решать некоторые задачи доведена до автоматизма. Это есть признак профессионализма, который вовсе не означает, что в действиях профессионала отсутствует алгоритм. В-третьих, в случае неудовлетворенности ре-

зультатом деятельности возможную причину неудачи следует искать в несовершенстве алгоритма, т. е. пытаться выявить алгоритм, исследовать его, искать «слабые места», устранять их, т. е. совершенствовать алгоритм и, следовательно, повышать системность деятельности. Таким образом, явная алгоритмизация любой практической деятельности является важным средством ее развития.

Системными являются также результаты практической деятельности. Следует отметить, что роль системных представлений в практике постоянно увеличивается, что растет сама системность человеческой деятельности. Данный тезис можно пояснить на примере проектирования технических объектов. Если раньше перед разработчиками новых образцов техники ставилась задача создания работоспособного объекта, то в настоящее время практика ставит задачу создания новых объектов с некоторыми оптимальными свойствами. Т. е. к разрабатываемым образцам еще на этапе проектирования предъявляются требования оптимальности. Цели, которые ставятся перед разработчиками, таким образом, являются более глобальными, более сложными.

Далее отметим, что *системным является само мышление.* Успешное решение поставленной задачи зависит от того, насколько системно подходит специалист к ее анализу. Неудачи в решении тех или иных проблем связаны с отходом от системности, с игнорированием части существенных взаимосвязей компонентов системы. Разрешение возникшей проблемы осуществляется путем перехода на новый, более высокий уровень системности. В связи с этим можно отметить, что системность не столько состояние, сколько процесс.

Свойство системности присуще процессу познания. Системны знания, накопленные человечеством. В качестве особенности процесса познания отметим наличие аналитического и синтетического образов мышления. *Анализ* — это процесс, состоящий в разделении целого на части, в представлении сложного в виде совокупности более простых компонентов. Но чтобы познать целое, сложное, необходим и обратный процесс — *синтез*. Это относится как к индивидуальному мышлению, так и к общечеловеческому знанию.

Аналитичность человеческого знания находит свое отражение в существовании различных наук, в продолжающейся их дифференциации, во все более глубоком изучении все более узких вопросов. Вместе с тем мы наблюдаем и обратный процесс синтеза знаний. Процесс синтеза проявляется в возникновении междисциплинарных наук, таких как физическая химия, биофизика,

биохимия и т. п. Наконец, наиболее высокая форма синтеза знаний реализуется в виде наук о самых общих свойствах природы. К числу таких синтетических наук относится в первую очередь философия, которая выявляет и отражает общие свойства всех форм существования материи. К синтетическим можно также отнести математику — дисциплину, изучающую всеобщие отношения, взаимосвязи и взаимодействия объектов. К числу синтетических можно отнести и системные науки: кибернетику, теорию систем, теорию организации и т. п. В этих дисциплинах органическим образом соединяются технические, естественнонаучные и гуманитарные знания.

В качестве методологического подхода к анализу явлений и процессов с точки зрения их системности развился диалектический метод. Именно диалектический метод рассматривает объект как комплекс взаимодействующих и взаимосвязанных компонентов, развивающихся во времени. Диалектика является методом познания, обеспечивающим согласование системности знаний и системности мира на любом уровне абстракции.

Свойство системности присуще результатам познания. В технических науках это реализуется в построении адекватных моделей, являющихся отражением исследуемых объектов, моделей, описывающих динамическое поведение материальных объектов.

Системна также среда, окружающая человека. Свойство системности является естественным свойством природы. Как уже отмечалось, окружающий нас мир есть бесконечная система систем, иерархическая организация все более сложных объектов. Причем как в живой, так и неживой природе действуют свои законы организации, являющиеся объективными биологическими или физическими законами.

Системно человеческое общество в целом. Системность человеческого общества выражается во взаимосвязи развития отдельных структур (национальных, государственных, религиозных образований) и в их взаимном влиянии друг на друга. Причем следует отметить, что уровень системности человеческого общества постоянно увеличивается. Системность необходимо, таким образом, рассматривать в историческом аспекте. Если в древнем мире племена жили достаточно отдаленно друг от друга и уровень общения между ними был минимален, то в современном обществе события происходящие в одних государствах находят отклик и имеют влияние в различных частях мира.

Системны взаимодействия человека со средой. В данном аспекте системность выражается в необходимости комплексного учета всех особенностей и возможных воздействий факторов вне-

шней среды на ее состояние в последующие моменты. В случае недостаточной проработки данных вопросов, игнорирования ряда факторов, наблюдается возникновение проблемы в развитии природы, негативное воздействие на хозяйственную и культурную деятельность человека. Примеров тому можно привести множество. Скажем, строительство гидроэлектростанций в равнинной части континента привело к заболачиванию мест, к выводу земель из севооборота, к нарушению экологической ситуации в данном регионе, а в некоторых случаях привело к изменению климата. Применение различных химикатов ненадлежащего качества и в необоснованном количестве привело к непоправимым последствиям в развитии региона Аральского моря. Примеры такого плана можно продолжать и продолжать. Таким образом, можно сделать вывод, что игнорирование системности взаимодействия человека со средой приводит к возникновению проблемы в развитии среды обитания и, соответственно, во взаимодействии природы и общества.

3.2. Место системного анализа в системных представлениях

Применения системных представлений для анализа сложных объектов и процессов рассматривают системные направления, включающие в себя: системный подход, системные исследования, системный анализ (системологию, системотехнику и т. п.). За исключением системотехники, область которой ограничена техническими системами, все другие термины часто употребляются как синонимы. Однако в последнее время системные исследования начали применять в экономике, социологии, психологии и других науках.

Системный подход. Этот термин начал применяться в первых работах, в которых элементы общей теории систем использовались для практических приложений. Используя этот термин, подчеркивали необходимость исследования объекта с разных сторон, комплексно, в отличие от ранее принятого разделения исследований на физические, химические и др. Оказалось, что с помощью многоаспектных исследований можно получить более правильное представление о реальных объектах, выявить их новые свойства, лучше определить взаимоотношения объекта с внешней средой, другими объектами. Заимствованные при этом понятия теории систем вводились не строго, не исследовался вопрос, каким классом систем лучше отобразить объект, какие свойства и закономерности этого класса следует учитывать при

конкретных исследованиях и т. п. Иными словами, термин «системный подход» практически использовался вместо терминов «комплексный подход», «комплексные исследования». Ярким примером реализации системного подхода является знаменитый «План ГОЭЛРО» электрификации России после гражданской войны 1918—1922 гг.

Системные исследования. В работах под этим названием понятия теории систем используются более конструктивно: определяется класс систем, вводится понятие структуры, а иногда и правила ее формирования и т. п. Это был следующий шаг в системных направлениях. В поисках конструктивных рекомендаций появились системные направления с разными названиями: системотехника, системология и др. Для их обобщения стал применяться термин «системные исследования». Часто в работах использовался аппарат исследования операций, который к тому времени был больше развит, чем методы конкретных системных исследований.

Системный анализ. В настоящее время системный анализ является наиболее конструктивным направлением. Этот термин применяется неоднозначно. В одних источниках он определяется как «приложение системных концепции к функциям управления, связанным с планированием». В других — как синоним термина «анализ систем» (*Э. Квейд*) или термина «системные исследования» (*С. Янг*). Однако независимо от того, применяется он только к определению структуры целей системы, к планированию или к исследованию системы в целом, включая и функциональную и обеспечивающую части, работы по системному анализу существенно отличаются от рассмотренных выше тем, что в них всегда предлагается методология проведения исследований, делается попытка выделить этапы исследования и предложить методику выполнения этих этапов в конкретных условиях. В этих работах всегда уделяется особое внимание определению целей системы, вопросам формализации представления целей. Некоторые авторы даже подчеркивают это в определении: системный анализ — это методология исследования целенаправленных систем (*Д. Киланд, В. Кинг*).

Термин «системный анализ» впервые появился в связи с задачами военного управления в исследованиях RAND Corporation (1948), а в отечественной литературе получил широкое распространение после выхода в 1969 г. книги *С. Оптнера* «Системный анализ для решения деловых и промышленных проблем».

В начале работы по системному анализу в большинстве случаев базировались на идеях теории оптимизации и исследования

операций. При этом особое внимание уделялось стремлению в той или иной форме получить выражение, связывающее цель со средствами, аналогичное критерию функционирования или показателю эффективности, т. е. отобразить объект в виде хорошо организованной системы.

Так, например, в ранних руководящих материалах по разработке АСУ рекомендовалось представлять цели в виде набора задач и составлять матрицы, связывающие задачи с методами и средствами достижения. Правда, при практическом применении этого подхода довольно быстро выяснялась его недостаточность, и исследователи стали прежде всего обращать внимание на необходимость построения моделей, не просто фиксирующих цели, компоненты связи между ними, а позволяющих накапливать информацию, вводить новые компоненты, выявлять новые связи и т. д., т. е. отображать объект в виде развивающейся системы, не всегда предлагая, как это делать.

Позднее системный анализ начинают определять как «процесс последовательного разбиения изучаемого явления на подпроцессы» (*С. Янг*) и основное внимание уделяют поиску приемов, позволяющих организовать решение сложной проблемы путем расчленения ее на подпроблемы и этапы, для которых становится возможным подобрать методы исследования и исполнителей. В большинстве работ стремились представить многоступенчатое расчленение в виде иерархических структур типа «дерева», но в ряде случаев разрабатывались методики получения вариантов структур, определяемых временными последовательностями функций.

В настоящее время системный анализ развивается применительно к проблемам планирования и управления, и в связи с усилением внимания к программно-целевым принципам в планировании этот термин стал практически неотделим от терминов «*целеобразование*» и «*программно-целевое планирование и управление*». В работах этого периода системы анализируются как целое, рассматривается роль процессов целеобразования в развитии целого, роль человека. При этом оказалось, что в системном анализе не хватает средств: развиты в основном средства расчленения на части, но почти нет рекомендаций как при расчленении не утратить целое. Поэтому наблюдается усиление внимания к роли неформализованных методов при проведении системного анализа. Вопросы сочетания и взаимодействия формальных и неформальных методов при проведении системного анализа не решены. Но развитие этого научного направления идет по пути их решения.

3.3. Развитие системного анализа

С позиций современных научных представлений системность всегда была методом любой науки. Возможно, что принципы системности применялись не всегда осознанно, но, тем не менее, любой ученый прошлого, который возможно и не помышлял о системном подходе, так или иначе имел дело с системами и моделями объектов или процессов. Ранее всего системные проблемы были осознаны философами. Следует отметить, что обсуждение системных проблем в таких дисциплинах, как философия, логика, математика, осуществлялось еще древними учеными.

Первым в явной форме вопрос о научном подходе к управлению сложными системами поставил М.А. Ампер. Он впервые выделил кибернетику как специальную науку об управлении государством, обозначил ее место в ряду других наук и сформулировал ее системные особенности. Идеи системности применительно к управлению государством развивались в работах польского ученого Б. Трентовского. Он отмечал, что действительно эффективное управление должно учитывать все важнейшие внешние и внутренние факторы, влияющие на объект управления. В своих работах Трентовский отмечал, что при выработке управляющего воздействия необходимо учитывать национальные особенности населения с учетом временного аспекта: «При одной и той же политической идеологии кибернет (в современной терминологии, лицо, принимающее решение) должен управлять различно в Австрии, России или Пруссии. Точно так же и в одной и той же стране он должен управлять завтра иначе, чем сегодня». Трентовский рассматривает общество как систему, которая развивается путем разрешения противоречий. И все-таки общество середины 19-го столетия было не готово к восприятию системных представлений. Прошло еще более полувека, прежде чем системная проблематика прочно заняла свое место в научных публикациях. К числу основоположников теории систем можно заслуженно отнести российского ученого, академика Е.С. Федорова. Основные научные результаты были достигнуты им в области минералогии. Он установил, что существует только 230 типов кристаллической решетки, тем не менее любое вещество при определенных условиях может кристаллизоваться. Таким образом, было показано, что великое многообразие кристаллов и минералов использует для своего строения ограниченное количество типов структур. Далее им были отмечены аналогичные закономерности в области архитектурных и музыкальных конструкций, языковых построений, строения вещества и ряда других систем.

Развивая системные представления, Федоров установил ряд других закономерностей развития систем, в частности им было замечено такое свойство систем, как самоорганизация, способность к приспособлению, к повышению стройности.

Следующим этапом в развитии системных представлений явились работы А.А. Богданова, который в начале XX века начал создавать теорию организации (тектологию). Основная идея теории Богданова заключается в том, что все существующие объекты и процессы имеют определенный уровень организованности. Далее им отмечено, что уровень организованности тем выше, чем сильнее свойства целого отличаются от простой суммы свойств комплекующих элементов. Именно анализ свойств целого и его частей был впоследствии заложен в качестве основной характеристики понятия сложной системы. Заслугой Богданова явилось также то, что он изучает не только статическое состояние структур, а занимается исследованием динамического поведения объектов, уделяет внимание вопросам развития организации, подчеркивает значение обратных связей, указывает на необходимость учета собственных целей организации, отмечает роль открытых систем. Он подчеркивает роль моделирования и математических методов как потенциальных методов решения задач теории организации.

Позднее идеи теории организации развивались в трудах выдающихся представителей отечественного естествознания И.И. Шмальгаузена, В.Н. Беклемишева и ряда других специалистов, вклад которых во многих отношениях явился решающим в формировании вышеназванной теории.

Вклад русских и советских исследователей в развитие теории систем и формирование системных представлений явился определяющим, поскольку большинство развиваемых ныне идей связано с работами Богданова и трудами его последователей. Однако нельзя не отметить также и зарубежных ученых, работы которых являются основополагающими в области развития теории систем и системного анализа. В первую очередь следует обратить внимание на труды австрийского ученого Л. Берталанфи, который в 50-х годах двадцатого столетия организовал в Канаде центр системных исследований. Им было опубликовано большое количество работ, в которых он исследовал взаимодействие систем с окружающей средой. Он подчеркивает большое значение обмена системы веществом, энергией и энтропией с внешним миром, отмечает, что в системе устанавливается динамическое равновесие, которое может быть направлено в сторону усложнения организации, функционирование системы является не просто откликом на изменение внешних условий, а сохранением старого или уста-

новлением нового внутреннего равновесия системы. В своих работах Берталанфи исследует общие закономерности, присущие любым достаточно сложным организациям материи как биологической, так и общественной природы. Берталанфи и организованная им школа последователей в своих трудах пытаются придать общей теории систем формальный характер.

Массовое распространение системных представлений, осознание системности мира, общества и человеческой деятельности связано с именем американского математика Н. Винера. В 1948 г. он опубликовал книгу «Кибернетика» и далее «Кибернетика и общество». В своих трудах он развивает идеи управления и связи в животном мире и машинах, анализирует с позиций кибернетики процессы, происходящие в обществе. Н. Винером и его последователями было указано, что предметом кибернетики является исследование систем. Причем отмечается, что хотя при изучении системы на каком-то этапе потребуется проводить учет ее конкретных свойств, для кибернетики в принципе несущественно, какова природа системы. То есть для изучения систем различных типов, будь она физической, биологической, экономической, организационной или вовсе представленной в виде модели, кибернетика предлагает единые подходы к ее исследованию.

Существенное место в развитии кибернетики занимают советские ученые. Можно отметить многочисленные работы академика А.И. Берга, Харкевича, Н.П. Бусленко, Н.Н. Моисеева. Фундаментальный вклад в развитие кибернетики внес также академик А.Н. Колмогоров. Так, в период, когда в Советском Союзе кибернетику считали лженаукой и в стране шли горячие дискуссии о ее сути, были сформулированы достаточно общие и полные определения кибернетики. Приведем эти определения: «Кибернетика — это наука об оптимальном управлении сложными динамическими системами» (А.И. Берг); «Кибернетика — это наука о системах, воспринимающих, хранящих, перерабатывающих и использующих информацию» (А.Н. Колмогоров).

Наконец, отметим достижения в области исследования систем бельгийской школы во главе с И. Пригожиным. Ученые этой школы исследовали механизм самоорганизации систем. Они отмечают, что в результате взаимодействия с окружающей средой система может перейти в неравновесное состояние. В результате такого взаимодействия изменяется организованность системы. Переломные точки, в которых наблюдается неустойчивость неравновесных состояний, называются точками бифуркации. Таким образом, согласно теории И. Пригожина, материя не является пассивной субстанцией, ей присуща спонтанная активность.

Системный анализ как дисциплина сформировался в результате возникновения необходимости исследовать и проектировать сложные системы, управлять ими в условиях неполноты информации, ограниченности ресурсов и дефицита времени. Системный анализ является дальнейшим развитием целого ряда дисциплин, таких как исследование операций, теория оптимального управления, теория принятия решений, экспертный анализ, теория организации эксплуатации систем и т. д. Для успешного решения поставленных задач системный анализ использует всю совокупность формальных и неформальных процедур. Перечисленные теоретические дисциплины являются базой и методологической основой системного анализа. Таким образом, системный анализ — *междисциплинарный курс, обобщающий методологию исследования сложных технических, биологических и социальных систем*. Широкое распространение идей и методов системного анализа, а главное — успешного их применения на практике, стало возможным только с внедрением и повсеместным использованием ЭВМ. Именно применение ЭВМ как инструмента решения сложных задач позволило перейти от построения теоретических моделей систем к широкому их практическому применению. В связи с этим Н.Н. Моисеев считает, что *системный анализ — это совокупность методов, основанных на использовании ЭВМ и ориентированных на исследование сложных систем — технических, экономических, экологических и т. д.*

Центральной проблемой системного анализа является проблема принятия решения. Применительно к задачам исследования, проектирования и управления сложными системами проблема принятия решения связана с выбором определенной альтернативы в условиях различного рода неопределенности. Неопределенность обусловлена многокритериальностью задач оптимизации, неопределенностью целей развития систем, неоднозначностью сценариев развития системы, недостаточностью априорной информации о системе, воздействием случайных факторов в ходе динамического развития системы и прочими условиями. Учитывая данные обстоятельства, системный анализ можно определить как дисциплину, занимающуюся проблемами принятия решений в условиях, когда выбор альтернативы требует анализа сложной информации различной физической природы.

Главным содержанием дисциплины «Системный анализ» являются сложные проблемы принятия решений, при изучении которых неформальные процедуры, представления здравого смысла и способы описания ситуаций играют не меньшую роль, чем формальный математический аппарат.

3.4. Методики и процедуры системного анализа

3.4.1. Принципы, этапы и процедуры системного анализа

В системном анализе можно выделить три главных направления. Эти три направления соответствуют трем этапам, которые всегда присутствуют в исследовании сложных систем. Перечислим их:

1. Постановка задачи исследования.
2. Построение модели исследуемого объекта.
3. Решение поставленной математической задачи.

Постановка задачи исследования. На данном этапе формулируется цель анализа. Цель исследования предполагается внешним фактором по отношению к системе. Таким образом, цель становится самостоятельным объектом исследования. Цель должна быть формализована. Задача системного анализа состоит в проведении необходимого анализа неопределенностей, ограничений и формулировании, в конечном счете, некоторой оптимизационной задачи:

$$f(x) \rightarrow \max, \quad x \in G. \quad (3.1)$$

Здесь x — элемент некоторого нормированного пространства G , определяемого природой модели, $G \subset E$, где E — множество, которое может иметь сколь угодно сложную природу, определяемую структурой модели и особенностями исследуемой системы. Таким образом, задача системного анализа на этом этапе трактуется как некоторая оптимизационная проблема. Анализируя требования к системе, т. е. цели, которые предполагает достигнуть исследователь, и те неопределенности, которые при этом неизбежно присутствуют, исследователь должен сформулировать цель анализа на языке математики. Язык оптимизации оказывается здесь естественным и удобным, но вовсе не единственно возможным.

Построение модели, т. е. формализация изучаемой системы, процесса или явления. Построение модели есть описание процесса на языке математики. При построении модели осуществляется математическое описание явлений и процессов, происходящих в системе. Поскольку знание всегда относительно, то описание на любом языке отражает лишь некоторые стороны происходящих процессов и никогда не является абсолютно полным. С другой стороны, следует отметить, что при построении модели необходимо уделять основное внимание тем сторонам изучаемого процесса, которые интересуют исследователя. Глубоко ошибочным является желание при построении модели системы отразить все стороны существования системы. При проведении системного анализа, как правило, интересуются динамичес-

ким поведением системы. Причем, при описании динамики с точки зрения проводимого исследования есть первостепенные параметры и взаимодействия, а есть несущественные в данном исследовании параметры. Таким образом, качество модели определяется соответствием выполненного описания тем требованиям, которые предъявляются к исследованию, соответствием получаемых с помощью модели результатов ходу наблюдаемого процесса или явления. Построение математической модели есть основа всего системного анализа, центральный этап исследования или проектирования любой системы. От качества модели зависит результат всего системного анализа.

Решение поставленной математической задачи. Только этот третий этап анализа можно отнести собственно к этапу, использующему в полной степени математические методы. Хотя без знания математики и возможностей ее аппарата успешное выполнение двух первых этапов невозможно, так как и при построении модели системы, и при формулировании цели и задач анализа широкое применение должны находить методы формализации. Однако отметим, что именно на завершающем этапе системного анализа могут потребоваться тонкие математические методы. Но следует иметь в виду, что задачи системного анализа могут иметь ряд особенностей, которые приводят к необходимости применения наряду с формальными процедурами эвристических подходов. Причины, по которым обращаются к эвристическим методам, в первую очередь связаны с недостатком априорной информации о процессах, происходящих в анализируемой системе. Также к таковым причинам можно отнести большую размерность вектора X и сложность структуры множества G . В данном случае трудности, возникающие в результате необходимости применения неформальных процедур анализа, зачастую являются определяющими. Успешное решение задач системного анализа требует использования на каждом этапе исследования неформальных рассуждений. Ввиду этого проверка качества решения, его соответствие исходной цели исследования превращается в важнейшую теоретическую проблему.

Сформулированные три этапа проведения системного анализа являются как бы основой решения любой задачи проведения системных исследований. Суть их состоит в том, что необходимо построить модель исследуемой системы, т. е. дать формализованное описание изучаемого объекта, сформулировать критерий решения задачи системного анализа, т. е. поставить задачу исследования и далее решить поставленную задачу. Указанные три этапа проведения системного анализа являются укрупненной схемой решения задачи. В действительности задачи системного анализа

являются достаточно сложными. Поэтому перечисление этапов не может быть самоцелью. Практикующему системотехнику требуется представлять методику выполнения каждого из этапов. В данном разделе сосредоточим внимание на содержании и последовательности операций выполнения системного анализа.

Использование правильной методики гарантирует исследователю, что он не будет искать решения неверно поставленной задачи. Грамотное проведение системного анализа предупреждает также и возможность неверного решения правильно поставленной задачи. Если исследовательская группа руководствовалась правильной методикой, то разработанные модели адекватны изучаемой проблеме и допустимы с точки зрения реализации вычислительного процесса, выполняются ограничения на выделяемые средства и сроки выполнения работ, а внедрение результатов системного анализа осуществляется квалифицированно и эффективно. В этом случае работы по системному анализу достигают цели.

Методика системного анализа. Методики, реализующие принципы системного анализа в конкретных условиях, направлены на то, чтобы формализовать процесс исследования системы, процесс поставки и решения проблемы. Методика системного анализа разрабатывается и применяется в тех случаях, когда у исследователя нет достаточных сведений о системе, которые позволили бы выбрать адекватный метод формализованного представления системы.

Общим для всех методик системного анализа является формирование вариантов представления системы (процесса решения задачи) и выбор наилучшего варианта. Положив в основу методики системного анализа эти два этапа, их затем можно разделить на подэтапы. Например, первый этап можно разделить следующим образом:

1. Отделение (или ограничение) системы от среды.
2. Выбор подхода к представлению системы.
3. Формирование вариантов (или одного варианта — что часто делают, если система отображена в виде иерархической структуры) представления системы.

Второй этап можно представить следующими подэтапами:

1. Выбор подхода к оценке вариантов.
2. Выбор критериев оценки и ограничений.
3. Проведение оценки.
4. Обработка результатов оценки.
5. Анализ полученных результатов и выбор наилучшего варианта (или корректировка варианта, если он был один).

В настоящее время трудно привести примеры методик, в которых все этапы были бы проработаны равноценно. Например, в за-

рубежных методиках (ПАТТЕРН, ПРОФИЛЕ, ППБ и др.), предназначенных, как правило, для анализа относительно хорошо структурированных военно-технических проблем, обычно более детально проработаны методы проведения одного этапа — этапа оценки и анализа структуры. Принятый обычно за основу при проведении второго этапа метод экспертных оценок не всегда (например, в области образования, медицины и других проблем, связанных с человеком) удастся реализовать на практике. Поэтому перспективным представляется сочетание экспертных оценок с количественными.

В принципе за основу при разработке методики системного анализа можно взять этапы проведения любого научного исследования или этапы исследования и разработки, принятые в теории автоматического управления. Однако специфической особенностью любой методики системного анализа является то, что она должна опираться на понятие системы и использовать закономерности построения, функционирования и развития систем. Здесь необходимо подчеркнуть, что при практическом применении методик системного анализа рассматривается следующее: часто после выполнения того или иного этапа (подэтапа) возникает необходимость возвратиться к предыдущему или еще более раннему этапу, а иногда и повторить процедуру системного анализа полностью. Это проявление закономерности саморегулирования, самоорганизации, которую при разработке методики можно учитывать сознательно, ввести правила, определяющие, в каких случаях необходим возврат к предыдущим этапам.

Отметим также, что методика проведения системного анализа и руководящие принципы не являются универсальными, каждое исследование имеет свои особенности и требует от исполнителей интуиции, инициативы и воображения, чтобы правильно определить цели проекта и добиться успеха в их достижении. Перейдем к формулированию последовательности работ по проведению системного анализа. Как уже было отмечено, специалисты по системному анализу приводят различные схемы его выполнению. Схемы проведения системного анализа представляются в виде алгоритмов. То обстоятельство, что системный анализ оперирует не только формализованными, но и неформализованными процедурами не означает, что нельзя говорить об алгоритмах системного анализа. Неоднократно имели место попытки создать достаточно общий, универсальный алгоритм системного анализа. Тщательный анализ имеющихся в литературе алгоритмов показывает, что они имеют большую степень общности в целом и различия в частностях, деталях.

Далее изложим основные процедуры алгоритма проведения системного анализа, которые являются обобщением последовательнос-

ти этапов проведения системного анализа и отражают общие закономерности проведения работ системного анализа. При этом нельзя утверждать, что предлагаемая схема работ по проведению системного анализа является универсальной. Алгоритм является прагматической моделью деятельности. Было бы неправильно утверждать, что один алгоритм является более правильным, чем другой, что реализация одного из них является системным анализом, а другого нет. Выбрав конкретный алгоритм выполнения работ по системному анализу, необходимо следовать предписаниям именно данного алгоритма. Если бы был выбран другой алгоритм, то работы велись бы согласно схеме действий, предписываемых моделью другого алгоритма. Следует заметить, что различные алгоритмы системного анализа могут быть взаимозависимыми. Так, ряд этапов могут совпадать. С другой стороны, в одних из них может уделяться большее внимание одним вопросам, в других расширенное решение находят другие стороны. Соотношение алгоритмов проведения системного анализа такое же, как алгоритмов программирования. Одна и та же, скажем, вычислительная задача может быть решена различными способами. Существуют различные численные методы реализации одних и тех же процедур, разная квалификация исполнителей, опыт работы, предпочтения в использовании тех или иных процедур, в конце концов, существуют разные языки программирования. Естественно, что разные программисты реализуют одну и ту же вычислительную задачу с помощью разных программ. Одна программа будет изящна, другая старомодна, но все они будут решать одну и ту же задачу. Точно так же системный аналитик может использовать тот или иной алгоритм системных исследований. Важно, чтобы все они позволяли решать задачи системного анализа и приводили к достижению поставленной цели.

Основные процедуры системного анализа следующие:

- определение целей системного анализа;
- изучение структуры системы, анализ ее компонентов, выявление взаимосвязей между отдельными элементами;
- сбор данных о функционировании системы, исследование информационных потоков, наблюдения и эксперименты над анализируемой системой;
- построение моделей;
- проверка адекватности моделей, анализ неопределенности и чувствительности;
- исследование ресурсных возможностей;
- формирование критериев;
- генерирование альтернатив;
- реализация выбора и принятия решений;
- внедрение результатов анализа.

3.4.2. Определение целей системного анализа

Формулирование проблемы. Для традиционных наук начальный этап работы заключается в постановке формальной задачи, которую надо решать. В исследовании сложной системы это промежуточный результат, которому предшествует длительная работа по структурированию исходной проблемы. Начальный пункт определения целей в системном анализе связан с формулированием проблемы. Здесь следует отметить следующую особенность задач системного анализа. Необходимость системного анализа возникает тогда, когда заказчик уже сформулировал свою проблему, то есть проблема не только существует, но и требует решения. Однако системный аналитик должен отдавать себе отчет в том, что сформулированная заказчиком проблема представляет собой приблизительный, рабочий вариант. Причины, по которым исходную формулировку проблемы необходимо считать в качестве первого приближения, состоят в следующем. Система, для которой формулируется цель проведения системного анализа, не является изолированной. Она связана с другими системами, входит как часть в состав некоторой надсистемы. Например, автоматизированная система управления отделом или цехом на предприятии является структурной единицей АСУ всего предприятия, точно также как АСУ предприятия имеет связи с отраслевой системой. Сама система, в свою очередь, состоит из подсистем. И поэтому, формулируя проблему для рассматриваемой системы, необходимо учитывать, как решение данной проблемы отразится на системах, с которыми связана данная система. Неизбежно планируемые изменения затронут и подсистемы, входящие в состав данной системы, и надсистему, содержащую данную систему. Таким образом, к любой реальной проблеме следует относиться не как к отдельно взятой, а как к объекту из числа взаимосвязанных проблем.

Другая причина того, что к сформулированной заказчиком проблеме следует относиться как к первоначальному рабочему варианту, состоит в том, что она (проблема) является его рабочей моделью, его взглядом на проблемную ситуацию. В реальной жизни необходимо учитывать позиции всех заинтересованных сторон. Учет мнений всех заинтересованных сторон приводит к дополнениям, уточнениям первоначального варианта описанной проблемы. Следовательно, системное исследование проблемы должно начинаться с ее расширения до системы проблем, связанных с исследуемой, без учета которых она не может быть решена. Это расширение должно происходить как с учетом связей

данной системы с над- и подсистемами, так и с точки зрения углубления данной проблемы, ее детализации.

Для формулирования системы проблем необходимо сформировать перечень заинтересованных лиц, так или иначе связанных с работами по системному анализу. В данный перечень следует включать, во-первых, клиента, который ставит проблему, заказывает и оплачивает системный анализ. Именно заказчик формулирует исходную проблему системного анализа. Далее включаются лица, принимающие решения, от полномочий которых зависит решение проблемы. Необходимо учитывать мнения активных участников решения проблемы, поскольку на них лягут основные работы по реализации принятых решений. Следующий контингент — пассивные участники, те, на ком скажутся последствия решения проблемы. На этапе формулирования проблемы необходимо учитывать, к каким изменениям приведут внедрения мероприятий проведенного системного анализа и как это отразится на пассивных участниках. И, наконец, требуется включать в перечень самого системного аналитика и его сотрудников, главным образом для того, чтобы предусмотреть возможность минимизации его влияния на остальных заинтересованных лиц.

При формулировании системы проблем системный аналитик должен следовать некоторым рекомендациям. Во-первых, за основу должно браться мнение заказчика. Как правило, в качестве такового выступает руководитель организации, для которой проводится системный анализ. Как было отмечено выше, именно он генерирует исходную формулировку проблемы. Далее системный аналитик, ознакомившись со сформулированной проблемой, должен уяснить задачи, которые были поставлены перед руководителем, ограничения и обстоятельства, влияющие на поведение руководителя, противоречивые цели, между которыми он старается найти компромисс. Насколько это возможно, следует выяснить личные качества руководителя, его склонности и предубеждения. Далее системный аналитик должен изучить организацию, для которой проводится системный анализ. Необходимо тщательно ознакомиться с существующей иерархией управления, функциями различных групп, а также предыдущими исследованиями соответствующих вопросов, если таковые проводились. Аналитик должен воздерживаться от высказывания своего предвзятого мнения о проблеме и от попыток втиснуть ее в рамки своих прежних представлений ради того, чтобы использовать желательный для себя подход к ее решению. Наконец, аналитик не должен оставлять непроверенными утверждения и замечания руководителя. Как уже отмечалось, проблему, сформулированную

руководителем, необходимо, во-первых, расширять до комплекса проблем, согласованных с над- и подсистемами, и, во-вторых, согласовывать ее со всеми заинтересованными лицами.

Следует также отметить, что каждая из заинтересованных сторон имеет свое видение проблемы, отношение к ней. Поэтому при формулировании комплекса проблем необходимо учитывать, какие изменения и почему хочет внести та или другая сторона. Кроме того, проблему необходимо рассматривать всесторонне, в том числе и во временном, историческом плане. Требуется предвидеть, как сформулированные проблемы могут измениться с течением времени или в связи с тем, что исследование заинтересует руководителей другого уровня. Формулируя комплекс проблем, системный аналитик должен дать развернутую картину того, кто заинтересован в том или ином решении.

Определение целей. После того как сформулирована проблема, которую требуется преодолеть в ходе выполнения системного анализа, переходят к определению цели. Определить цель системного анализа — это означает ответить на вопрос, что надо сделать для снятия проблемы. Сформулировать цель — значит указать направление, в котором следует двигаться, чтобы разрешить существующую проблему, показать пути, которые уводят от существующей проблемной ситуации.

Формулируя цель, требуется всегда отдавать отчет в том, что она имеет активную роль в управлении. В определении было отражено, что цель — это желаемый результат развития системы. Таким образом, сформулированная цель системного анализа будет определять весь дальнейший комплекс работ. Следовательно, цели должны быть реалистичными. Задание реалистичных целей направит всю деятельность по выполнению системного анализа на получение определенного полезного результата. Важно также отметить, что представление о цели зависит от стадии познания объекта, и по мере развития представлений о нем цель может переформулироваться. Изменение целей во времени может происходить не только по форме, в силу все лучшего понимания сути явлений, происходящих в исследуемой системе, но и по содержанию, вследствие изменения объективных условий и субъективных установок, влияющих на выбор целей. Сроки изменения представлений о целях, старения целей различны и зависят от уровня иерархии рассмотрения объекта. Цели более высоких уровней долговечнее. Динамичность целей должна учитываться в системном анализе.

При формулировании цели нужно учитывать, что на нее оказывают влияние как внешние по отношению к системе факторы, так и внутренние факторы. При этом внутренние факторы явля-

ются такими же объективно влияющими на процесс формирования цели факторами, как и внешние.

Далее следует отметить, что даже на самом верхнем уровне иерархии системы имеет место множественность целей. Анализируя проблему, необходимо учитывать цели всех заинтересованных сторон. Среди множества целей желательно попытаться найти или сформировать глобальную цель. Если этого сделать не удастся, то следует проранжировать цели в порядке их предпочтения для снятия проблемы в анализируемой системе.

Исследование целей заинтересованных в проблеме лиц должно предусматривать возможность их уточнения, расширения или даже замены. Это обстоятельство является основной причиной итеративности системного анализа.

На выбор целей субъекта решающее влияние оказывает та система ценностей, которой он придерживается. Поэтому при формировании целей необходимым этапом работ является выявление системы ценностей, которой придерживается лицо, принимающее решение. Так, например, различают технократическую и гуманистическую системы ценностей. Согласно первой системе природа провозглашается как источник неисчерпаемых ресурсов, человек — царь природы. Всем известен тезис Мичурина: «Мы не можем ждать милостей от природы. Взять их у нее — наша задача». Гуманистическая система ценностей говорит о том, что природные ресурсы ограничены, что человек должен жить в гармонии с природой и т. д. Подробный анализ этих систем ценностей выполнен В.П. Зинченко (1988 г.). Практика развития человеческого общества показывает, что следование технократической системе ценностей приводит к пагубным последствиям. С другой стороны, полный отказ от технократических ценностей тоже не имеет оправдания. Необходимо не противопоставлять эти системы, а разумно дополнять их и формулировать цели развития системы с учетом обеих систем ценностей.

3.4.3. Анализ структуры системы

Любая задача системного анализа начинается с построения модели исследуемой системы. Для решения задачи построения модели необходимо вначале произвести изучение структуры системы, выполнить анализ ее компонентов, выявить взаимосвязи между отдельными элементами. Чтобы обоснованно проводить анализ структуры системы, необходимо рассмотреть ряд понятий и определений, характеризующих строение и функционирование системы.

Структура отражает определенные взаимосвязи, взаиморасположение составных частей системы, ее устройство, строение.

При описании системы недостаточно перечислить элементы, из которых она состоит. Требуется отобразить систему путем расчленения ее на подсистемы, компоненты и элементы и показать, каким путем обеспечивается в объекте выполнение поставленной цели. Для выполнения такой процедуры и вводят понятие структуры. Таким образом, структура отражает наиболее существенные взаимоотношения между элементами и их группами, которые мало меняются при изменениях в системе и обеспечивают существование системы и ее основных свойств. Структура характеризует организованность системы, устойчивую упорядоченность ее элементов и связей. *Структура системы* — состав системы и схема связей между ее элементами. Понятие структуры можно определить следующим образом: совокупность отношений, заданных на множестве подсистем и элементов, образующих некоторую систему, называется структурой этой системы.

Следующее понятие, которое будем рассматривать, — *связь*. Данное понятие входит в любое определение системы наряду с понятием *элемент* и обеспечивает возникновение и сохранение структуры и целостных свойств системы. Понятие «связь» характеризует одновременно и статическое строение системы, и динамическое ее поведение. Связь определяют как ограничение степени свободы элементов. Связь характеризуется направлением, силой и характером. По первому признаку связи делят на направленные и ненаправленные. По второму признаку различают сильные и слабые связи. По характеру выделяют связи подчинения, равноправные, генетические, связи управления. Различают также связи по направленности процессов — прямые и обратные. Обратные связи могут быть положительными, сохраняющими тенденции происходящих в системе изменений того или иного параметра, и отрицательными — противодействующими тенденциям изменения выходного параметра. Обратная связь направлена на сохранение, стабилизацию требуемого значения параметра. Обратная связь является основой приспособления систем к изменяющимся условиям существования, основой саморегулирования и развития систем.

Следующее понятие — *цель системы* — важное понятие, лежащее в основе развития систем. Цели системы — желательные состояния или результаты поведения системы. *Глобальная цель системы* — конечное состояние, к которому стремится система в силу своей структурной организации. Цель можно также определить следующим образом: цель — это субъективный образ (абстрактная модель) несуществующего, но желаемого состояния среды, которое решило бы возникшую проблему. В практических приложениях цель — это идеальное устремление, которое позволяет коллективу увидеть перспективы или реальные возможности,

обеспечивающие своевременное завершение очередного этапа на пути к идеальным устремлениям. Цель достигается путем решения задач. *Задачи системы* — цели, которые желательно достичь к определенному моменту времени в пределах определенного периода функционирования системы.

Для описания системы создается ее модель. *Модель* — это отражение структуры системы, ее элементов и взаимосвязей направленное на отображение определенной группы свойств. Создание модели системы позволяет предсказывать ее поведение в определенном диапазоне условий.

Формы представления структур. Структурные представления являются средством исследования систем. Одну и ту же систему можно представить различными структурами, необходимый выбор которых обусловлен содержанием исследований, проводимых на данном этапе. Принятый способ описания структур — графическое изображение. В таком графе элементы, компоненты, подсистемы и прочие объекты системы отображаются в виде вершин графа, связи между объектами представляют в виде дуг. Рассмотрим основные способы представления структур.

Сетевые структуры представляют отображение взаимосвязи объектов между собой. Их применяют для представления организационных структур, для изображения структурных схем систем, для представления информационного обеспечения и т. д. С помощью сетевых структур отображаются пространственные взаимосвязи между элементами, как правило, одного иерархического уровня. Примером сетевой структуры может служить структурная схема ЭВМ (рис 3.1), элементами которой являются

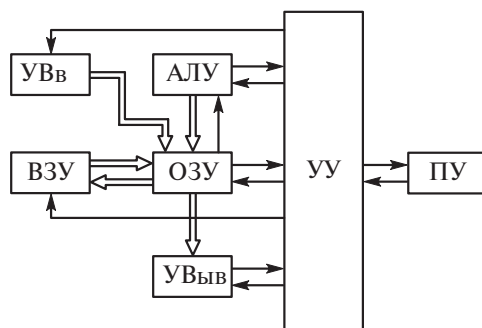


Рис. 3.1. Упрощенная структурная схема ЭВМ: УВв — устройство ввода информации, ВЗУ — внешнее запоминающее устройство, АЛУ — арифметико-логическое устройство, ОЗУ — оперативное запоминающее устройство, УУ — устройство управления, ПУ — пульт управления, УВыв — устройство вывода информации

устройство ввода информации, оперативное запоминающее устройство, внешнее запоминающее устройство, арифметико-логическое устройство, устройство вывода информации, устройство управления и пр. На рисунке стрелками показаны связи между элементами.

Различают следующие виды сетевых структур. *Линейные структуры* со строго упорядоченным взаимоотношением элементов «один к одному». Примером линейной структуры может служить схема одного из каналов (любого) аварийной защиты энергоблока ядерной энергетической установки (ЯЭУ). Каналы строятся по принципу линейного соединения группы устройств: датчик — блок питания — вторичный прибор, регистрирующий информацию с датчика (рис. 3.2).

Древовидная структура представляет собой объединение многих линейных подструктур. Примером может служить схема подсистемы аварийной защиты энергоблока ЯЭУ. Подсистема состоит из группы однотипных каналов, каждый из которых дублирует работу других (рис 3.3).

Кольцевая структура (циклическая) имеет замкнутые контуры в соответствующих графах. С помощью циклических структур изображаются схемы циркуляции информации в системах.

Обобщенная сетевая структура характеризуется многочисленными межэлементными связями.

Иерархические структуры представляют собой декомпозицию системы в пространстве. Применяются, прежде всего, для описания подчиненности элементов в структурах управления. Термин *иерархия* означает соподчиненность, порядок подчинения низших по должности лиц высшим. В настоящее время концепция



Рис. 3.2. *Схема канала аварийной защиты энергоблока ЯЭУ:*
 КНК — камера нейтронная компенсирующая (датчик нейтронного потока), БП — блок питания, УЗС — устройство защиты по скорости нарастания мощности (вторичный прибор)

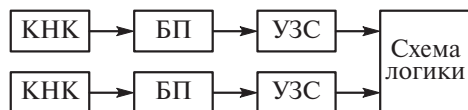


Рис. 3.3. *Схема подсистемы аварийной защиты энергоблока ЯЭУ:*
 схема логики — логическая схема выработки сигнала аварийной защиты



Рис. 3.4. Фрагмент схемы ЭВМ

иерархии распространена на любой согласованный по подчиненности порядок объектов. В иерархических структурах важно лишь выделение уровней соподчиненности, а между уровнями и между компонентами в пределах уровня, в принципе, могут быть любые взаимоотношения. Примером применения иерархической структуры может служить изображение схемы ЭВМ с детализацией на каждом новом уровне иерархии (рис 3.4).

Таким образом, рассмотрены основные понятия, с помощью которых осуществляется решение задачи анализа системы. Для решения задачи в том виде, в котором она была сформулирована в п. 3.4.1, на данном этапе необходимо произвести изучение структуры системы, анализ ее компонентов, выявление взаимосвязей между отдельными элементами, т. е. осуществить структурную декомпозицию системы. Основное содержание процедур, выполняемых на данном этапе, состоит в том, чтобы подготовить информацию к проведению работ по построению модели системы.

3.4.4. Сбор данных о функционировании системы

Решение задачи, сформулированной в предыдущем параграфе, а именно: изучение структуры системы, анализ ее компонентов, выявление взаимосвязей между отдельными элементами, — преследует цель отразить статическое состояние системы. Однако при проведении системного анализа исследователя интересуют вопросы, касающиеся изучения свойств системы. Свойства системы реализуются в процессе ее функционирования, т. е. в процессе динамического поведения системы. Чтобы построить модель системы, которая имела бы возможность отражать свойства и характеристики системы, реализующиеся в процессе ее функционирования во времени, необходимо, помимо структуры

системы, знать ее параметры. Поэтому следующим этапом работ при проведении системного анализа является сбор данных о функционировании системы и исследование информационных потоков.

Основное содержание данного этапа состоит в идентификации параметров системы с целью последующего включения их в модель. Этот этап связан с определением числовых значений параметров системы в режиме ее функционирования. Параметры системы подразделяются на внутренние и внешние.

Внешние параметры системы — характеристики функционирования системы, служащие показателями качества ее работы как единого целого. В качестве примера внешних параметров рассмотрим параметры автоматизированной системы. К таковым относятся:

- общая производительность системы по обработке данных;
- объем передаваемой информации;
- достоверность выходной экспериментальной информации;
- точность получения результатов (для информации, заданной количественно);
- количественные характеристики надежности системы;
- объем используемой в системе аппаратуры (объем памяти, количество преобразователей формы информации, количество внешних устройств и т. д.);
- время задержки с момента поступления в систему исходных данных до момента выдачи окончательных результатов (во время решения определенной задачи);
- стоимость системы (с учетом разработки математического обеспечения);
- показатели удобства системы в эксплуатации (наличие элементов «психологического комфорта») и др.

Внутренние параметры системы — характеристики, показывающие особенности технических решений, принятых при организации системы в целом и отдельных технических средств, входящих в состав системы, а также в совокупности влияющие на значения внешних параметров системы. Примерами внутренних параметров автоматизированных систем являются:

- вид и характеристики сигналов для представления информации в системе, в каналах связи — при обмене информацией между отдельными звеньями системы;
- способ кодирования информации; вид приоритетности при приеме и обработке информации от различных источников;
- способ организации программы-диспетчера;
- быстродействие отдельных элементов и т. д.

Ограничения, накладываемые на внутренние параметры, связаны с конечностью распространения электромагнитных волн, с конечностью возможных типов используемых блоков и элементов аппаратуры системы, с дискретностью числа ячеек запоминающих устройств и т. д. Разделение на внешние и внутренние параметры весьма условно. Обычно к внешним относят те параметры, на которые задаются ограничения, определяемые назначением системы или вызванные условиями ее функционирования.

При описании параметров системы определению подлежат: идентификатор параметра, единицы измерения, диапазон изменения, качественные характеристики (однозначный — многозначный, регулируемый — не регулируемый), место применения в модели. Параметры отражают свойства системы. Одни из них определить достаточно просто. Например, такие параметры, как объем памяти, количество внешних устройств, стоимость системы, способ кодирования информации, вид приоритетности, при приеме и обработке информации от различных источников можно получить на основании изучения документации на систему. Другие же параметры определяются опосредованно, на основании обработки информации, полученной в результате наблюдений за работой системы. К таким параметрам относятся параметры, характеризующие надежность системы, качество функционирования, точность получения количественных результатов и т. п. Наблюдения с целью сбора данных могут проводиться в процессе функционирования системы, либо же для сбора данных организуются специальные экспериментальные исследования. В таком случае говорят, что данные получены в результате *пассивного эксперимента*. Во втором случае имеет место *активный эксперимент*. Активный эксперимент проводится по специально составленному плану с использованием методов планирования эксперимента. При этом предусматривается возможность изменения входных параметров, влияющих на процесс функционирования системы. Исследуется изменение выходных параметров системы в зависимости от уровней входных параметров. Результаты испытаний фиксируются с помощью измерений, т. е. изображения результатов опыта в виде символов, номеров или чисел. Измерение — это алгоритмическая операция, которая данному наблюдаемому состоянию системы или процесса ставит в соответствие определенное обозначение: число, номер или символ. Такое соответствие обеспечивает то, что результаты измерений содержат информацию об исследуемой системе. Требуемая информация в виде оценок параметров получается путем преобразования результатов измерения, или, как еще говорят, с помощью обработки экспериментальных данных.

Современное понимание измерений существенно шире только количественных измерений. Есть наблюдаемые явления, в принципе не допускающие числовой меры, но которые можно фиксировать в «слабых» шкалах, и эти результаты учитывают в моделях, получая качественные, но вполне научные выводы. Расплывчатость некоторых наблюдений признана их неотъемлемым свойством, которому придана строгая математическая форма, и разработан формальный аппарат работы с такими наблюдениями. Погрешности измерений также являются неотъемлемым, естественным и неизбежным свойством процесса измерения. Причинами этого являются наличие неопределенностей в процессе наблюдения, квантование результатов измерения, шумы аппаратуры и т. д. Широкое распространение получили статистические измерения, т. е. оценивание функционалов распределения вероятностей по зафиксированным наблюдениям значений реализации случайного процесса. Этот класс измерений имеет особое значение потому, что большое число входных и выходных показателей реализуется в виде случайных величин.

3.4.5. Исследование информационных потоков

В случае, когда анализируют социотехнические системы (организационные, человеко-машинные, автоматизированные), помимо определения параметров системы для построения модели важное значение приобретают вопросы исследования информационных потоков, циркулирующих в системе. Анализ информационных потоков позволяет выявить схему работы объектов управления, обеспечивает информационное отображение объекта управления, взаимосвязь между его элементами, структуру и динамику информационных потоков. Изучаются формы документов и недокументированных сообщений. В процессе изучения информационных потоков анализируются следующие группы документов:

1. Официальные положения и инструкции, регламентирующие функции подразделений и определяющие сроки и процедуры обработки информации и принятия решений.
2. Входные документы, источники которых находятся вне системы.
3. Систематически обновляемые записи в виде картотек или книг, используемые в процессе работы.
4. Промежуточные документы, получаемые и используемые в процессе обработки данных.
5. Выходные документы.

Анализ информационных потоков осуществляется с помощью специально разработанных методов: графического, метода с использованием сетевой модели, графоаналитического и метода с использованием графов типа «дерево». *Графический метод* применяется для описания потоков информации главным образом на макроуровне, когда решается задача анализа общей схемы работы объектов управления. Здесь отношения между элементами потока, в виде которых выступают документы, изображают структурно-информационно-временной схемой. На схеме приводятся краткие пояснения, описывающие движение информации и материальных потоков.

Метод с использованием сетевой модели состоит в следующем. В качестве события сетевой модели фигурирует определенный документ. Если документ представляет собой результат выполнения какой-либо работы, то он является конечным, если же он будет использоваться в дальнейшем ходе выполнения работ, такой документ будет начальным. Под работой понимается определенная задача или функция, выполняемая элементом органа управления.

Графоаналитический метод основан на анализе матрицы смежности информационного графа. В данном случае исходными для анализа информационных потоков являются данные о парных отношениях между наборами информационных элементов, формализуемые в виде матрицы смежности. Под *информационными элементами* понимают различные типы входных, промежуточных и выходных данных. *Матрица смежности* — квадратная бинарная матрица с количеством строк (и столбцов), равным количеству информационных элементов. В каждой позиции матрицы смежности записывают единицу, если между соответствующими элементами матрицы существует отношение, то есть информация одного документа используется при формировании другого, и в соответствующей позиции ставится ноль, если отношения нет. Далее матрице смежности ставится в соответствие граф информационных взаимосвязей. Множеством вершин графа является множество информационных элементов, дуги отражают взаимосвязи между элементами. Дуга присутствует, если в матрице смежности отношение между элементами отмечено единицей, и отсутствует в противном случае. Анализ графа позволяет выявить информационную зависимость между входными, промежуточными и выходными документами, характер зависимости, установить направление движения информации. Графоаналитический метод является развитием метода с использованием сетевой модели и позволяет проводить более детальный анализ информационных потоков.

Метод с использованием графов типа «дерево» применяют для описания системы потоков информации. Строится граф взаимосвязи показателей и так называемые графы расчетов, описывающие преобразование информации в процессе формирования отдельных показателей. При построении дерева взаимосвязи показателей ребра ориентируют с учетом иерархии от исходных к результирующим. Такой подход позволяет строить графы с более высокой степенью укрупнения. Полученный комплекс графов отражает процесс движения и преобразования информации в системе.

3.4.6. Построение моделей системы

Материалы, изложенные в предыдущих двух параграфах, решают задачи изучения структуры системы, выявления параметров, характеризующих функционирование системы и влияющих на эффективность и качество ее работы, анализа информационных потоков, циркулирующих в системе. Данные этапы являются предварительными этапами работы по построению модели системы, цель этих этапов — выявление основных структурных элементов, динамических и информационных компонент системы. После выяснения этих вопросов переходят к решению основной задачи — построению модели системы.

Построение математической модели системы есть процесс формализации определенных сторон существования, жизнедеятельности системы, ее поведения с точки зрения конкретной решаемой задачи. Различают статические и динамические модели. *Статическая модель* отражает конкретное состояние объекта. Примером статической модели является структурная схема системы. *Динамическая модель* описывает процесс изменения состояний системы. При решении задач системного анализа цели исследования заключаются в изучении характеристик системы, прогнозировании путей развития системы, сравнении вариантов развития и т. п., т. е. интересуются, в основном, вопросами динамического поведения систем. Следовательно, можно сказать, что динамические модели находят более широкое применение, чем статические.

Следующий вопрос, на котором следует остановиться при обсуждении подходов к построению математической модели, это целевое предназначение модели. Перед тем как приступить к созданию математической модели, необходимо уяснить существо решаемой задачи, для которой создается данная модель. Ошибочной будет разработка модели системы, описывающая все сто-

роны, все аспекты существования и развития системы. Такая модель будет излишне громоздка и скорее всего непригодна для проведения каких-либо серьезных исследований. Модель всегда должна быть конкретной, нацеленной на решение поставленной задачи. Для оценки характеристик надежности системы необходимо строить модель надежности, для решения задач прогнозирования развития производственных процессов необходимо строить производственную модель, для решения экономических задач — экономическую модель. Если перед системными аналитиками ставится задача исследования ряда аспектов, то целесообразнее создавать несколько моделей, а не пытаться разрабатывать всеобъемлющую одну модель. Правда, в этом случае необходимо, чтобы разные модели, отражающие различные аспекты существования и развития системы, были взаимосвязаны по входным и выходным параметрам и характеристикам системы. Такая взаимосвязь достигается путем проведения итеративных расчетов на моделях. То есть осуществляется последовательный расчет моделей. Те параметры, которые известны до проведения расчетов, задаются в качестве входных в каждой из моделей, где их присутствие необходимо. Недостающие параметры получают расчетным путем и последовательно включаются в модели от первой к последующим по мере проведения расчетов. На начальном этапе эти параметры заменяются оценками, принадлежащими области определения параметра. По мере получения результатов модели должны уточняться и процесс расчетов по уточненным моделям должен повторяться. В этом заключается итеративность процесса. Расчеты прекращаются, когда исследователь отмечает сходимость процессов уточнения параметров.

Далее рассмотрим типы математических моделей. Выделяют два класса моделей: аналитические и имитационные. В *аналитических моделях* поведение сложной системы записывается в виде некоторых функциональных соотношений или логических условий. Наиболее полное исследование удастся провести в том случае, когда получены явные зависимости, связывающие искомые величины с параметрами сложной системы и начальными условиями ее изучения. Однако это удастся выполнить только для сравнительно простых систем. Для сложных систем исследователю приходится идти на упрощение реальных явлений, дающее возможность описать их поведение и представить взаимодействия между компонентами сложной системы. Это позволяет изучить хотя бы некоторые общие свойства сложной системы, например, оценить устойчивость системы, характеристики надежности и т. п. Для построения математических моделей имеется

мощный математический аппарат (функциональный анализ, исследование операций, теория вероятностей, математическая статистика, теория массового обслуживания и т. д.). Наличие математического аппарата и относительная быстрота и легкость получения информации о поведении сложной системы способствовало повсеместному и успешному распространению аналитических моделей при анализе характеристик сложных систем.

Когда явления в сложной системе настолько сложны и многообразны, что аналитическая модель становится слишком грубым приближением к действительности, системный аналитик вынужден использовать *имитационное моделирование*. В имитационной модели поведение компонента сложной системы описывается набором алгоритмов, которые затем реализуют ситуации, возникающие в реальной системе. Моделирующие алгоритмы позволяют по исходным данным, содержащим сведения о начальном состоянии сложной системы, и фактическим значениям параметров системы отобразить реальные явления в системе и получить сведения о возможном поведении сложной системы для данной конкретной ситуации. На основании этой информации аналитик может принять соответствующие решения. Отмечается, что предсказательные возможности имитационного моделирования значительно меньше, чем у аналитических моделей.

Вопрос о том, какой модели следует отдать предпочтение при проведении исследований характеристик системы, не является очевидным. Аналитическая модель имеет некоторые преимущества по сравнению с имитационной моделью. Во-первых, аналитическая модель дает решение поставленной задачи в законченной форме. Во-вторых, применение аналитической модели обеспечивает глубину анализа. С помощью аналитических моделей можно проводить исследование характеристик в некоторой области определения параметров, в которой модель адекватна описываемым явлениям или процессам. Применение аналитических моделей позволяет получить решение в виде функциональной зависимости исследуемых характеристик от параметров модели. Имитационная модель за один цикл ее применения производит расчет характеристик в одной точке. Для получения функциональной зависимости выходной характеристики от параметров модели необходимо провести многократные расчеты на имитационной модели.

С другой стороны, построить аналитическую модель для сложной системы очень трудно. При таком построении требуется принимать существенные упрощающие предположения, которые могут привести к тому, что построенная модель будет неадекватна описываемым процессам или явлениям. В этом смысле имитаци-

онные модели имеют преимущества, так как они могут быть построены в самых общих предположениях о функционировании системы. Следовательно, имитационные модели могут быть более адекватны. К недостаткам аналитических моделей относится также и то, что простая модификация проекта или изменение предположений о функционировании элементов структуры может потребовать коренной перестройки модели, в то время как у имитационной модели потребуются изменить лишь входную информацию.

Рассмотрим простой пример. Пусть необходимо оценить характеристики надежности системы, структура которой известна. Если проводить расчеты в предположении об отсутствии восстановительных мероприятий после отказов элементов, то аналитическая модель для такого расчета строится с использованием логико-вероятностного подхода. Если изменить предположения и считать, что после отказа элементов осуществляется восстановление и потоки отказов и восстановлений пуассоновские, то для расчета надежности используются уравнения Колмогорова—Чепмена. Если же будем предполагать восстановление элементов, но потоки отказов или восстановлений будем описывать не пуассоновским, а каким-нибудь другим распределением, то для построения моделей расчета надежности необходимо использовать аппарат теории восстановления. То есть, для решения одной и той же задачи при смене предположений о характере функционирования системы приходится для построения аналитической модели полностью менять теоретический аппарат. В имитационной модели в этом случае меняются лишь входные данные. Таким образом, на основании вышесказанного нельзя однозначно решить, какая модель лучше. Как аналитическая, так и имитационная модели являются полезным инструментом исследования и об их соответствии решаемым проблемам надо судить в контексте конкретного применения. В задачах системного анализа целесообразно проводить комбинированные исследования, использующие как аналитические, так и имитационные модели.

3.4.7. Проверка адекватности моделей. Анализ неопределенности и чувствительности

После того как модель построена, необходимо удостовериться в ее качестве. С этой целью выполняют ряд операций, а именно: осуществляют проверку адекватности модели процессу, объекту или явлению, для которых она построена, проверку непротиворечивости модели, неопределенности, чувствительности,

реалистичности и работоспособности. Рассмотрим существо каждой из проводимых работ.

Проверка адекватности моделей. Важный вопрос, который интересует исследователя после того, как построена модель исследуемого явления или процесса, это проверка адекватности модели. Проверить *адекватность модели* — это значит установить, насколько хорошо модель описывает реальные процессы, происходящие в системе, насколько качественно она будет прогнозировать развитие данных процессов. Проверка адекватности модели проводится на основании некоторой экспериментальной информации, полученной на этапе функционирования системы или при проведении специального эксперимента, в ходе которого наблюдаются интересующие системного аналитика процессы. Проверка адекватности модели заключается в доказательстве факта, что точность результатов, полученных по модели, будет не хуже точности расчетов, произведенных на основании экспериментальных данных. Если иметь в виду целевое предназначение моделируемого объекта, то под адекватностью модели нужно понимать степень ее соответствия этому предназначению. В качестве примера, иллюстрирующего необходимость решения вопроса об адекватном описании результатов наблюдений соответствующими моделями, рассмотрим регрессионную модель, с помощью которой описали поведение некоторого процесса. Рассмотрим два рисунка (рис. 3.5, а и б) с одинаковым расположением экспериментальных точек и, следовательно, одинаковым разбросом относительно линии регрессии. Различие в этих рисунках состоит в том, что модели, изображенные на них, построены на основании разного количества экспериментальных данных. В связи с этим имеем различный средний разброс в экспериментальных точках факторного пространства. Разброс в точках показан отрезками прямых, численно равных величине доверительного интервала, построенного для функции отклика.

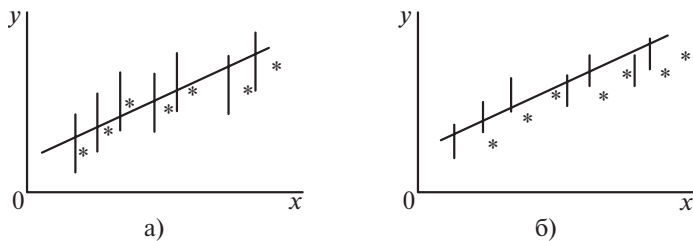


Рис. 3.5. Проверка адекватности модели: а — объем экспериментальных данных мал; б — объем экспериментальных данных велик.

Линейная модель регрессии адекватна в первом случае (рис. 3.5, *a*), т. к. разброс в точках того же порядка, что и разброс относительно линии регрессии. Во втором случае (рис. 3.5, *b*) не все отрезки прямых, численно равных величине доверительного интервала, накрывают линию регрессии. Следовательно, в этом случае требуется более сложная модель, чтобы точность ее предсказания была сравнима с точностью экспериментальных данных.

В первом случае модель обладает удовлетворительными точностными характеристиками по сравнению с экспериментальной информацией, на основании которой она построена. Во втором случае точность предсказания модели хуже точности экспериментальных данных. Таким образом, модель адекватна экспериментальным данным только в первом случае.

Непротиворечивость модели. Целью данного этапа является проверка факта: дает ли модель не противоречащие логике результаты при вариации величин важнейших параметров, особенно в тех случаях, когда их значения близки к экстремальным. Чтобы ответить на этот вопрос, необходимо проанализировать характер реакции модели на изменения соответствующих входных параметров. Для проверки непротиворечивости модели в первую очередь анализируют, какие результаты дает модель при нулевых значениях входных параметров, в том числе в нулевой момент времени, далее исследуется состояние модели на границе области определения входных параметров, например, в точке бесконечность, если она входит в область определения. Можно привести такой пример. При решении задачи анализа надежности сложной системы вычисляют значение коэффициента готовности. Асимптотические модели для описания коэффициента готовности хорошо известны, они приведены в соответствующей литературе. В случае расчета коэффициента готовности системы в моменты времени, сравнимые с наработками элементов системы до отказа приходится строить неасимптотические модели. Они гораздо сложнее асимптотических. Для проверки непротиворечивости неасимптотических моделей можно использовать известные результаты, полученные для описания асимптотического поведения коэффициента готовности. Для этого необходимо в неасимптотической модели задать время, равное бесконечности, и провести сравнение результатов расчета с асимптотической моделью. Если результаты совпали, то это может служить подтверждением того, что модель дает результаты, не противоречащие известным ранее.

Анализ неопределенности модели. Поскольку модель системы только стремится отобразить реальность, то неизбежно существ-

зование упрощений, допущений и идеализаций сложных процессов и явлений, происходящих в системе. Следствием этих упрощений и идеализаций будут неопределенности в итоговых результатах, получаемых в процессе применения модели. Природа возникновения неопределенностей многогранна. Выделяют следующие источники неопределенностей в соответствующих моделях: обусловленные неполнотой моделей, неадекватностью моделей и неопределенностью исходных параметров.

Неопределенности, обусловленные неполнотой моделей, возникают из-за того, что при построении моделей системный аналитик не предусмотрел некоторые стороны развития моделируемых процессов, происходящих в системе. Иными словами, при разработке модели системы были не учтены отдельные особенности существования и развития систем. Это может быть сделано сознательно, когда аналитик считает, что данные особенности системы не играют большой роли и ими можно пренебречь. Иногда это происходит в результате недостаточной проработанности вопросов, связанных с изучением структуры и динамического поведения систем. В результате мы имеем недостаток полноты модели, который приводит к неопределенности в результатах и выводах и который трудно проанализировать и определить количественно.

Второй тип неопределенностей связан с неадекватностью моделей. Даже в тех случаях, когда в модели учтены все особенности существования и развития систем, последовательность событий и логические особенности функционирования систем, заложенные в модель, не точно отражают реальность. Существуют неопределенности, вызванные неадекватностью концептуальных и математических моделей, числовой аппроксимацией, ошибками в вычислительных программах и ограничениями вычислительного процесса. Эти неопределенности рассматриваются как часть анализа неопределенности моделей, для оценки их относительной значимости проводятся исследования чувствительности результатов моделирования.

Третий тип неопределенностей — неопределенность исходных параметров. Параметры различных моделей точно не известны. Причиной этого является недостаточность данных, используемых при статистическом оценивании входных параметров, невозможность точного описания поведения персонала, работающего в составе анализируемой системы, наличие допущений, принятых при составлении модели. Эта третья категория неопределенностей при современном состоянии методологии может быть наиболее успешно охарактеризована численно.

Анализ чувствительности модели. Анализом чувствительности модели называют процедуру оценки влияния допусков входных параметров на ее выходные характеристики. Проводят анализ чувствительности следующим образом: задают отклонение входного параметра в правую и левую сторону от его среднего значения и фиксируют, как при этом изменяются выходные значения характеристик модели. В качестве величины отклонения обычно принимают среднее квадратическое отклонение. Практическая сторона анализа чувствительности модели к изменению входных параметров состоит в том, что устанавливается степень зависимости выходных параметров от входных характеристик. Эту степень влияния затем можно проранжировать и выявить наиболее значимые входные параметры. Если в ходе проверки модели на чувствительность к изменению входных параметров установлено, что ряд параметров приводит к незначительным изменениям выходных характеристик, сравнимых с точностью проведения расчетов на модели, то данные входные параметры можно вывести из модели. Таким образом, анализ чувствительности модели может привести к упрощению модели и исключению из нее незначимых факторов.

Установить реалистичность модели — значит ответить на вопрос: соответствует ли модель тем частным случаям, для которых уже имеются фактические данные. Одним из способов проверки реалистичности модели может служить метод прогнозирования назад. Т. е. в модели задаются требуемые входные параметры и производится расчет некоторого события, которое уже имело место, или же рассчитываются характеристики системы на время, которое система отработала, и оценки этих характеристик можно получить по реальным данным. Если результаты расчета на модели дают хорошее совпадение с практикой, то можно считать, что модель реалистична.

Работоспособность. Цель анализа работоспособности модели выяснить, насколько модель практична и удобна в эксплуатации. Во-первых, модель должна обеспечивать результат за разумное время. Если модель используется в процессе выработки и принятия решения, то необходимо, чтобы расчеты можно было выполнить в пределах сроков, установленных для подготовки соответствующих решений. Если это условие не выполняется, смысл модели пропадает, так как теряется ее предназначение. Во-вторых, трудозатраты и ресурсы, требуемые для эксплуатации модели, должны укладываться в установленные лимиты машинного времени и фонда зарплаты. Должно выполняться условие практической целесообразности.

Следующий аспект проверки модели связан с анализом допущений и предположений, принятых при построении модели. На этом этапе проверки работоспособности оценивается качество модели, ее свойства в условиях воздействия реальных внешних возмущений и параметров. Суть данной процедуры состоит в том, чтобы удостовериться, что при составлении модели «не выплеснули ребенка вместе с водой». Принятие некоторых допущений и ограничений может привести к тому, что модель не будет отражать сути происходящих явлений и процессов. Здесь следует отметить, что эта задача взаимосвязана с задачей проверки адекватности модели.

3.4.8. Исследование ресурсных возможностей

Для того, чтобы модель начала давать результаты, чтобы она работала, необходимы затраты ресурсов. Модель нужно не только воплотить в надлежащем виде, но и обеспечить возможность получения решения нужного качества и к нужному моменту времени. Не требует пояснений то обстоятельство, что даже самое обоснованное решение становится ненужным, если оно появляется после того, как истекли сроки, выделенные для принятия решения, т. е. когда необходимость в нем уже отсутствует. Поэтому при реализации моделей необходимы ресурсы, которые позволяют обеспечить выполнение условий качества и своевременности. Принципиальное значение имеет вопрос, в какой степени обеспечено ресурсами управление ходом выполнения задач системного анализа.

Рассмотрим основные виды ресурсов, используемых при реализации задач системного анализа. Выделяют энергетические, материальные, временные и информационные ресурсы. Характеристику ресурсов начнем с *энергетических ресурсов*. Обычно энергетические затраты на реализацию модели значительно меньше, чем затраты энергии, потребляемые самой системой, для которой разработана модель. Поэтому в обычной ситуации энергетическими ресурсами, как правило, пренебрегают. Однако актуальность обеспечения энергетическими ресурсами возникает в тех случаях, когда модельные исследования проводятся на объектах, работающих в относительно автономных условиях. Примерами таких объектов являются научно-исследовательские морские суда, летательные аппараты, космические станции. Энергетические возможности таких объектов ограничены. Следовательно, прежде чем приступить к организации модельных исследований в таких условиях, требуется обосновать обеспеченность готовящихся исследований с точки зрения достаточности энергетических ресурсов.

Следующий вид ресурсов — материальные ресурсы. *Материальные ресурсы* — это достаточно обширная категория. Сюда можно отнести и людские ресурсы, требуемые для реализации моделей, и ресурсы обеспеченности проводимых исследований необходимым оборудованием, приборами и инструментами, и канцелярские товары и принадлежности и т. п. В случае решения задачи путем моделирования на ЭВМ в качестве материальных ресурсов выступают объем памяти и машинное время. Указанные ресурсы ограничивают возможности решения задач большой размерности в реальном масштабе времени. С подобными проблемами приходится сталкиваться при решении задач в экономических, социальных, метеорологических, организационно-управленческих, сложных технических системах. В случае нехватки ресурсов для решения подобных задач необходимо проводить реконструкцию модели. Самое тривиальное действие, которое можно применить в данном случае, это провести декомпозицию модели системы на совокупность связанных моделей меньшей размерности.

Временные ресурсы. Практика решения задач системного анализа такова. Заказчик работ заключает с системными аналитиками, которые выступают в роли исполнителей работ, договор. В данном договоре оговариваются сроки выполнения работ по проведению системного анализа. Как правило, эти сроки являются ограничивающим временным фактором на выполняемые работы. Таким образом, исследования, проводимые с помощью моделей, должны по времени укладываться в рамки, оговоренные договором.

Наконец, *информационные ресурсы.* Количество и качество информации, используемой при построении моделей систем, различно. Если при построении модели используется достоверная информация в достаточно представительном объеме, то это является одним из условий построения хорошей модели. Качество и полнота информации, представленной в модели, обеспечивает принятие обоснованных решений и является гарантией успешного управления. В свою очередь, ограниченность информации приводит к большой степени неопределенности результатов, получаемых в ходе расчетов на модели. Решения, принимаемые на основе таких моделей, будут обладать слабой степенью обоснованности.

Таким образом, при построении и реализации моделей следует уделять внимание обеспечению процесса использования моделей всеми видами ресурсов. Даже самая качественная модель в смысле адекватности описания происходящих в системе процессов может на практике оказаться бесполезной, если она не обеспечена в надлежащем объеме всеми видами ресурсов, необходимых для ее успешного применения.

3.4.9. Формирование критериев

Критерий — это способ сравнения альтернатив. Необходимо различать понятия «критерий» и «критериальная функция». Критерием качества альтернативы может служить любой ее признак, значение которого можно зафиксировать в порядковой или более сильной шкале. После того как критерий сформирован, т. е. найдена характеристика, которая будет положена в основу сравнения альтернатив, появляется возможность ставить задачи выбора и оптимизации.

Задача формирования критериев решается непосредственно после того, как сформулированы цели системного анализа. Ситуация становится понятной, если к критериям относиться как к количественным моделям качественных целей. Задача системного анализа состоит в том, чтобы формализовать проблемную ситуацию, возникающую в ходе системного анализа. Этой цели как раз и служит этап формирования критериев. Сформированные критерии в некотором смысле должны заменять цели. От критериев требуется как можно большее сходство с целями, чтобы оптимизация по критериям соответствовала максимальному приближению к целям. Выполняя данный этап, необходимо осознавать, что критерии не могут полностью совпадать с целями. Одной из причин этого является то, что критерии и цели формулируются в разных шкалах: цели в номинальных, критерии в более сильных, допускающих упорядочение. Критерий является отображением ценностей, воплощенных в целях, на параметры альтернатив, допускающие упорядочение. Определение значения критерия для данной альтернативы является косвенным измерением степени ее пригодности как средства достижения цели.

Обсуждая вопрос формирования критериев, следует сказать, что это достаточно трудная и серьезная задача. Редко бывает так, что решение лежит на поверхности. Зачастую для формирования хорошего критерия, адекватно отражающего цель системного анализа, приходится прибегать к неформализуемым процедурам. Неформализуемые, творческие, эвристические этапы играют важную роль в процессе формирования критериев. Решая задачи системного анализа, возникает ситуация, когда невозможно предложить один критерий, адекватно отражающий цель исследования. Даже одну цель редко удается выразить одним критерием, хотя к этому необходимо стремиться. Критерий, как и всякая модель, лишь приближенно отображает цель, адекватность одного критерия может оказаться недостаточной. Поэтому решение может состоять не обязательно в поиске более адекватного критерия, оно может вы-

ражаться в использовании нескольких критериев, описывающих одну цель по-разному и дополняющих друг друга. Еще более усложняется задача в случае, когда сформулировано несколько целей системного анализа, отражающих разные системы ценностей. В этом случае исследователь тем более вынужден формировать несколько критериев и в последующем решать многокритериальную задачу. Таким образом, можно отметить, что многокритериальность является способом повышения адекватности описания цели. Однако, введение многокритериальности в задачах системного анализа не должно быть самоцелью. Качество постановки задачи заключается не только и не столько в количестве критериев, сколько в том, чтобы они достаточно адекватно описывали цель системного анализа. Критерии должны описывать по возможности все важные аспекты цели, но при этом желательно минимизировать число необходимых критериев.

Формирование критериев отражает цель, которую ставит заказчик. Но при постановке и решении задач системного анализа необходимо учитывать не только цели, на решение которых он направлен, но и возможности, которыми обладают стороны для решения поставленных задач, и которые позволяют снять выявленные проблемы. В первую очередь необходимо учитывать ресурсы, имеющиеся у сторон. К ресурсам следует отнести: денежные ресурсы, которые заказчик согласен выделить системным аналитикам для решения поставленной задачи; ресурсы исполнителя, сюда относят людские ресурсы, ресурсы вычислительные (наличие вычислительной техники, ее количество и т. д.), материальные ресурсы, требуемые для решения задач (например, наличие канцелярских товаров, транспорта, ресурсов связи); временные ресурсы, сроки решения задач системного анализа, как правило, оговариваются. При формулировке задачи системного анализа необходимо также учитывать интересы окружающей среды. Окружающая среда хоть и играет пассивную роль, но необходимо учитывать, что любая система существует внутри нее, взаимодействует с ней. Поэтому при постановке задачи системного анализа необходимо следовать принципу «не навредить», не предпринимать ничего, что противоречило бы законам природы. Чтобы удовлетворить условиям непревышения количества имеющихся ресурсов, в постановку задачи системного анализа вводят ограничения.

Между целевыми критериями и ограничениями имеется сходство и различия. Общее заключается в том, что и критерий и ограничения являются математической формулировкой некоторых условий. В некоторых задачах оптимизации они могут выступать равноправно. Однако на этапе формирования критериев

целевой критерий открывает возможности для генерирования новых альтернатив в поисках лучшей из них, а ограничение заведомо уменьшает их число, запрещая некоторые из них. Одними целевыми критериями можно жертвовать ради других, ограничения же исключить нельзя, они должны четко соблюдаться.

При формулировании задач системного анализа встречаются случаи, когда ограничения задаются завышенными. Это может привести к нереальности достижения целей системного анализа. В этом случае необходимо ставить вопрос об ослаблении ограничений. Приведем пример. Слишком высокие требования к характеристикам надежности системы могут привести к необходимости чрезвычайных дополнительных финансовых вложений. А это в свою очередь может привести к неэффективности разработки и эксплуатации объекта, для которого проводится анализ. Таким образом, формулируя ограничения, необходимо руководствоваться соображениями здравого смысла. В качестве приема, позволяющего найти наилучшие соотношения между критериями и ограничениями, можно порекомендовать использование итерационных процедур. После проведения определенных вычислений и установления факта завышенности требований, сформулированных в ограничениях, можно эти требования ослабить и попытаться решить задачу заново.

В заключение данного параграфа перечислим основные критерии, наиболее часто встречающиеся в анализе сложных технических систем. Это экономические критерии — прибыль, рентабельность, себестоимость; технико-экономические — производительность, надежность, долговечность; технологические — выход продукта, характеристики качества и пр.

3.4.10. Генерирование альтернатив

Следующим этапом системного анализа является создание множества возможных способов достижения сформулированной цели. Иными словами, на данном этапе необходимо сгенерировать множество альтернатив, из которых затем будет осуществляться выбор наилучшего пути развития системы. Данный этап системного анализа является очень важным и трудным. Важность его заключается в том, что конечная цель системного анализа состоит в выборе наилучшей альтернативы на заданном множестве и в обосновании этого выбора. Если в сформированное множество альтернатив не попала наилучшая, то никакие самые совершенные методы анализа не помогут ее вычислить. Трудность этапа обусловлена необходимостью генерации достаточно полного

множества альтернатив, включающего на первый взгляд даже самые нереализуемые.

Генерирование альтернатив, т. е. идей о возможных способах достижения цели, является настоящим творческим процессом. Существует ряд рекомендаций о возможных подходах к выполнению рассматриваемой процедуры. Во-первых, необходимо сгенерировать как можно большее число альтернатив. Способы генерации таковы:

- поиск альтернатив в патентной и журнальной литературе;
- привлечение нескольких экспертов, имеющих разную подготовку и опыт;
- увеличение числа альтернатив за счет их комбинации, образования промежуточных вариантов между предложенными ранее;
- модификация имеющейся альтернативы, т. е. формирование альтернатив, лишь частично отличающихся от известной;
- включение альтернатив, противоположных предложенным, в том числе и «нулевой» альтернативы (не делать ничего, т. е. рассмотреть последствия развития событий без вмешательства системотехников);
- интервьюирование заинтересованных лиц и более широкие анкетные опросы;
- включение в рассмотрение даже тех альтернатив, которые на первый взгляд кажутся надуманными;
- генерирование альтернатив, рассчитанных на различные интервалы времени (долгосрочные, краткосрочные, экстренные).

При выполнении работы по генерированию альтернатив важно создать благоприятные условия для сотрудников, выполняющих данный вид деятельности. Большое значение имеют психологические факторы, влияющие на интенсивность творческой деятельности. Поэтому необходимо стремиться к созданию благоприятного климата на рабочем месте сотрудников.

Существует еще одна опасность, возникающая при выполнении работ по формированию множества альтернатив, о которой необходимо сказать. Если специально стремиться к тому, чтобы на начальной стадии было получено как можно больше альтернатив, т. е. стараться сделать множество альтернатив как можно более полным, то для некоторых проблем их количество может достичь многих десятков. Для подробного изучения каждой из них потребуются неприемлемо большие затраты времени и средств. Поэтому в данном случае необходимо провести предварительный анализ альтернатив и постараться сузить множество на ранних этапах ана-

лиза. На этом этапе анализа применяют качественные методы сравнения альтернатив, не прибегая к более точным количественным методам. Тем самым осуществляется грубое отсеивание.

Приведем теперь методы, используемые в системном анализе для проведения работы по формированию множества альтернатив.

Методы коллективной генерации идей. Методы коллективной генерации идей известны, так же и как метод мозгового штурма, или мозговой атаки. Данный метод является методом систематической тренировки творческого мышления, нацеленным на открытие новых идей и достижение согласия группы людей на основе интуитивного мышления. Техника мозгового штурма состоит в следующем. Собирается группа лиц, отобранных для генерации альтернатив. Главный принцип отбора заключается в подборе специалистов разных профессий, опыта работы и квалификации. Данная группа обсуждает проблему, причем заранее оговаривается, что приветствуются любые идеи, возникшие как индивидуально, так и по ассоциации при выслушивании предложений других участников. Приветствуются даже идеи, лишь незначительно улучшающие высказывания предыдущих выступающих. При обсуждении придерживаются ряда правил:

- необходимо обеспечить как можно большую свободу мышления участников мозгового штурма и высказывания ими новых идей;
- допускается высказывание любых идей, даже если вначале они кажутся сомнительными и абсурдными;
- не допускается критика, не объявляется ложной и не прекращается обсуждение ни одной идеи;
- приветствуется высказывание как можно большего числа идей, особенно нетривиальных.

Разработка сценариев. В некоторых проблемах искомое решение должно описывать реальное поведение объекта в будущем, определять реальный ход событий. В таких случаях альтернативами являются различные последовательности действий и вытекающих из них событий, которые могут произойти с системой в будущем. Эти последовательности имеют общее начальное состояние и различные траектории движения развития системы. Это различие и приводит к проблеме выбора. Такие гипотетические альтернативные описания поведения системы в будущем называются *сценариями*.

Сценарии-альтернативы — это логически обоснованные модели поведения проблемносодержащей системы в будущем, которые после принятия решения можно рассматривать как прогноз изменения состояний системы. Разработка сценариев относится к типичным неформализуемым процедурам. Для составления сце-

нариев привлекаются специалисты, которые должны знать общие закономерности развития систем. При составлении сценариев проводят анализ внутренних и внешних факторов, влияющих на развитие системы, определяют источники этих факторов, целенаправленно анализируют высказывания ведущих специалистов в научных публикациях по рассматриваемой тематике. Сценарий является предварительной информацией, на основе которой проводится дальнейшая работа по прогнозированию развития системы. Сценарий помогает составить представление о проблеме, затем приступают к более тщательным, как правило, количественным процедурам анализа.

Морфологические методы. Основная идея морфологических методов состоит в систематическом переборе всех мыслимых вариантов решения проблемы или развития системы путем комбинирования выделенных элементов или их признаков. Системный аналитик определяет все мыслимые параметры, от которых может зависеть решение проблемы и представляет их в виде матриц-строк. Затем в этой матрице определяются все возможные сочетания параметров по одному из каждой строки. Полученные таким образом варианты подвергаются оценке и анализу с целью выбора наилучшего варианта решения проблемной ситуации. Методологию морфологического анализа можно проиллюстрировать на примере, касающемся разработки системы телевизионной связи. Рассмотрим табл. 3.1.

Таблица 3.1

Независимая переменная	Значение переменной
Цвет изображения	Черно-белое Одноцветное Двухцветное Трехцветное Семицветное
Размерности изображения	Плоское изображение Объемное изображение
Градации яркости	Непрерывные Дискретные (оцифрованные)
Звуковое сопровождение	Без звука Монофонический звук Стереофонический звук
Передача запахов	Без передачи запахов С сопровождением запахов
Наличие обратной связи	Без обратной связи С обратной связью

Эта таблица порождает 384 различные возможные системы телевизионной связи. Современному телевизионному вещанию соответствует только одна альтернатива. Таким образом, анализируя данную таблицу, можно сказать, что у телевидения широкие возможности для дальнейшего развития.

Деловые игры. Деловыми играми называется имитационное моделирование реальных ситуаций. В процессе моделирования участники игры ведут себя таким образом, будто они в реальности выполняют порученную им роль. Реальная ситуация в данном случае заменяется некоторой моделью. Чаще всего деловые игры используются для обучения, однако их с успехом применяют и для экспериментального генерирования альтернатив, особенно в слабо формализованных ситуациях. Важная роль в деловых играх отводится руководителю игры, тому, кто управляет моделью, регистрирует ход игры и обобщает ее результаты.

Методы экспертного анализа. Методы экспертного анализа разрабатывались для решения задачи структурирования и системной организации процесса получения и кодирования данных и знаний, источником которых является человек-эксперт. Методы экспертного анализа применяются для решения слабоформализованных задач. Суть методов состоит в подборе группы экспертов, являющихся специалистами в рассматриваемой области знаний. Перед ними формулируется задача, скажем, изложить свое мнение по проблеме, требующей решения, предложить пути развития системы, обосновать траекторию изменения состояний системы в будущем и т. п. После получения ответов появляется как бы коллективное мнение, коллективный взгляд на решаемую проблему. В результате обработки экспертных ответов получают наиболее вероятный прогноз по развитию системы.

Метод «Дельфи». Метод «Дельфи» — итеративная процедура при проведении мозговой атаки, которая должна снизить влияние психологических факторов при проведении обсуждений проблемы и повысить объективность результатов. В отличие от традиционного подхода к достижению согласованности мнений экспертов путем открытой дискуссии метод «Дельфи» предполагает полный отказ от коллективных обсуждений. Это делается для того, чтобы уменьшить влияние таких психологических факторов, как присоединение к мнению наиболее авторитетного специалиста, нежелание отказаться от публично выраженного мнения, следование за мнением большинства. В методе «Дельфи» прямые дебаты заменены тщательно разработанной программой последовательных индивидуальных опросов, проводимых в форме анкетирования. Ответы экспертов обобщаются и

вместе с новой дополнительной информацией поступают в распоряжение экспертов, после чего они уточняют свои первоначальные ответы. Такая процедура повторяется несколько раз до получения приемлемой сходимости совокупности высказанных мнений.

Методы типа дерева целей. Метод типа дерева целей, или дерева направлений прогнозирования, подразумевает использование иерархической структуры, полученной путем разделения общей цели на подцели, а их, в свою очередь, на более детальные составляющие — новые подцели, функции и т. д. Древовидные иерархические структуры используются при исследовании вопросов совершенствования организационных систем.

Таким образом, рассмотрены методы, которые находят применение при решении задачи генерирования альтернатив. Важным моментом при решении данного вопроса является итеративность. Суть ее состоит в том, что на любой последующей стадии системного анализа должна быть возможность порождения новой альтернативы и включения ее в состав анализируемых. При рассмотрении слабоструктурированных проблем в качестве метода анализа используют следующий подход. Берут за основу одну подходящую альтернативу и производят ее пошаговое улучшение.

3.4.11. Реализация выбора и принятия решений

Целевое предназначение всего системного анализа состоит в том, чтобы в результате осуществить выбор. Выбор или принятие решения есть суть поставленной задачи системного анализа, конечный итог всей работы. Заказчик формулирует перед системным аналитиком проблему, которая стоит перед ним. Его интересуют прагматичные вопросы. Например, сформулировать мероприятия, которые гарантировали бы быстрое развитие предприятия с обеспечением максимальной прибыли. Или же предложить наилучшее решение по обеспечению стабильного электроснабжения некоторого региона. Системный аналитик должен ответить на вопросы: «Что лучше — построить новую электростанцию или провести модернизацию действующих, но выработавших свой ресурс? Какова будет надежность электростанции после проведения работ по модернизации? Будет ли на допустимом уровне риск от ее эксплуатации?» Заказчика, в общем-то, не интересует, каким способом будет выработано то или иное решение. Для него важно, чтобы оно было обосновано и отвечало на поставленный вопрос.

Все описанные ранее этапы работ являлись предварительными, направленными на изучение проблемной ситуации. Для того, чтобы обоснованно подойти к решению задачи выбора изучается система и строится ее модель, изучаются цели, которые ставит перед собой (и, естественно, системными аналитиками) заказчик, исследуются возможные пути развития анализируемой системы, т. е. генерируются альтернативы. После столь тщательной проработки проблемной ситуации наступает завершающий этап — этап принятия решения. Процедура принятия решения представляет собой действие над множеством альтернатив, в результате которого получается подмножество выбранных альтернатив. Желательно, чтобы это была одна альтернатива. Сужение множества альтернатив возможно, если есть способ сравнения альтернатив между собой и определения наиболее предпочтительных. Для того чтобы имелась возможность сравнивать альтернативы необходимо выработать критерий предпочтения. Проблема выбора сама по себе достаточно сложна. Она допускает существенно различающиеся математические постановки задач. Отметим основные сложности, возникающие при решении задач выбора и принятия решений:

- множество альтернатив может быть конечным, счетным или бесконечным;
- оценка альтернативы может осуществляться по одному или по нескольким критериям;
- критерии могут иметь количественное выражение или допускать только качественную оценку;
- режим выбора может быть однократным или повторяющимся, допускающим обучение на опыте;
- последствия выбора могут быть точно известны, иметь вероятностный характер или иметь неоднозначный исход, не допускающий введение вероятностей.

Различные сочетания перечисленных вариантов приводят к многообразным задачам выбора. Для решения задач выбора предлагаются различные подходы, наиболее распространенным из которых является критериальный подход. Основным предположением критериального подхода является следующее: каждую отдельно взятую альтернативу можно оценить конкретным числом — значением критерия. Критерии, на основе которых осуществляется выбор, имеют различные названия — критерий качества, целевая функция, функция предпочтений, функция полезности и т. д. Объединяет их то, что все они служат решению одной задачи — задачи выбора.

Сравнение альтернатив сводится к сравнению результатов расчетов соответствующих критериев. Если далее предположить, что выбор любой альтернативы приводит к однозначно определяемым последствиям и заданный критерий численно выражает оценку этих последствий, то наилучшей альтернативой является та, которая обладает наибольшим значением критерия. Задача поиска наилучшей альтернативы, простая по постановке, часто оказывается сложной для решения, поскольку метод ее решения определяется размерностью и типом множества альтернатив, а также видом критериальной функции. Однако на практике сложность отыскания наилучшей альтернативы многократно возрастает, так как оценивание вариантов приходится проводить на основании нескольких критериев, качественно различающихся между собой. Если в результате сравнения по нескольким критериям получилось, что одна альтернатива обладает наилучшими значениями по всем критериям, то ситуация выбора не представляет затруднений, именно эта альтернатива и будет наилучшей. Однако такая ситуация встречается лишь в теории. На практике дело обстоит гораздо сложнее. В данной ситуации приходим к необходимости решения многокритериальных задач. Подходы к решению таких задач известны — это метод сведения многокритериальной задачи к однокритериальной, метод условной максимизации, поиск альтернативы с заданными свойствами, нахождение паретовского множества альтернатив. Выбор альтернативы на основании критериального подхода предполагает, что выполненным являются несколько условий: известен критерий, задан способ сравнения вариантов и метод нахождения лучшего из них. Однако этого оказывается недостаточно. При решении задач выбора необходимо учитывать условия, при которых осуществляется выбор, и ограничения задачи, так как их изменение может привести к изменению решения при одном и том же критерии.

Оптимизационный подход нашел широкое применение в задачах системного анализа. Это обусловлено тем, что понятие оптимальности получило строгое и точное представление в математических теориях. Оптимизационный подход прочно вошел в практику проектирования и эксплуатации технических систем, сыграл важную роль в формировании современных системных представлений, широко используется в административном управлении. Нахождение оптимальных вариантов особенно важно для оценки состояния современной техники и определения перспектив ее развития. Знание параметров оптимальной альтернативы позволяет составить представление о принципиально наилучшаемых возможностях технических объектов. Сравнение с

оптимальными параметрами помогает решить вопрос о целесообразности дальнейших усилий по улучшению показателей качества изделий. Однако у оптимизационного подхода есть свои ограничения, требующие внимательного и осторожного обращения с ним. Остановимся на особенностях, накладывающих ограничения на применение оптимизационного подхода и требующих учета при решении задач принятия решений:

1. Оптимальное решение часто оказывается чувствительным к изменению условий задачи. Следует учитывать, что иногда такие изменения могут привести к выбору существенно отличающихся альтернатив.

2. Обычно система, для которой принимается решение, входит в структуру более общей системы, т. е. является ее подсистемой, и решения оптимальные для этой подсистемы могут входить в противоречие с целями надсистемы, т. е. возникает необходимость увязывать критерии подсистем с критериями надсистем.

3. Необходимо очень тщательно и скрупулезно подходить к выбору и обоснованию критерия. Критерий должен выбираться из анализа цели исследования, при этом надо помнить, что он характеризует цель лишь косвенно, иногда хуже, иногда лучше, но всегда приближенно.

4. Помимо критериев в оптимизационной задаче немаловажную роль играют ограничения. Анализ существа проблемной ситуации и качественное обоснование ограничений задачи имеют значительное влияние на принимаемое решение. Нередко даже небольшие изменения в ограничениях отражаются на принимаемом решении. Еще больший эффект получается, когда одни ограничения заменяются другими. Не задав всех необходимых ограничений, можно одновременно с оптимизацией основного критерия получить непредвиденные и нежелательные эффекты.

Таким образом, можно сделать вывод о том, что оптимизация — это мощное средство повышения эффективности, но использовать его необходимо осторожно, особенно при работе со сложными проблемными ситуациями. Проблема еще более обостряется, когда речь идет о принятии решений в организационных или социальных системах. Можно констатировать, что оптимизационные задачи, которые удается поставить при исследовании сложных систем, имеют обоснованный характер, если описывают хорошо структурированные системы, и являются заведомо приближенными, если относятся к системе в целом. Поэтому отметим, что оптимизационный подход является не единственным при решении задач выбора и принятия решений. Существуют другие методы, которые дополняют оптимизационный выбор.

Отметим их. Одним из таких методов является экспертный метод. Он применяется в тех случаях, когда при исследовании сложных систем возникают проблемы, которые не удается представить в виде формальных математических задач. В таких случаях прибегают к услугам экспертов, лиц, чья интуиция и опыт могут уменьшить сложность проблемы. И, наконец, для решения задач выбора и принятия решений в сложных проблемных ситуациях создаются специальные человеко-машинные, проблемно-ориентированные системы. Системы поддержки решений ориентированы не на автоматизацию функций лица, принимающего решение, а на предоставление ему помощи в поиске хорошего решения.

Подводя итог, отметим, что проблема выбора и принятия решений — центральная проблема системного анализа. Налицо сложности, которые возникают перед системным аналитиком. Но, с другой стороны, имеется развитый математический и эвристический аппарат, который является мощным оружием, помогающим обоснованно подходить к проблеме выбора.

3.4.12. Внедрение результатов анализа

Системный анализ является прикладной наукой, его конечная цель — изменение существующей ситуации в соответствии с поставленными целями. Окончательное суждение о правильности и полезности системного анализа можно сделать лишь на основании результатов его практического применения. Конечный результат будет зависеть не только от того, насколько совершенны и теоретически обоснованы методы, применяемые при проведении анализа, но и от того, насколько грамотно и качественно реализованы полученные рекомендации.

В настоящее время вопросам внедрения результатов системного анализа в практику уделяется повышенное внимание. В этом направлении можно отметить работы Р. Акоффа и П. Чеклэнда. Следует заметить, что практика системных исследований и практика внедрения их результатов существенно различаются для систем разных типов. Согласно классификации, введенной П. Чеклэндом, системы делятся на три типа: естественные, искусственные и социотехнические системы. В системах первого типа связи образованы и действуют природным образом. Примерами таких систем могут служить экологические, физические, химические, биологические и т. п. системы. В системах второго типа связи образованы в результате человеческой деятельности. Примерами могут служить всевозможные технические системы.

В системах третьего типа, помимо природных связей, важную роль играют межличностные связи. Такие связи обусловлены не природными свойствами объектов, а культурными традициями, воспитанием участвующих в системе субъектов, их характером и др. особенностями.

Системный анализ применяется для исследования систем всех трех типов. В каждой из систем есть свои особенности, требующие учета при организации работ по внедрению результатов. Наиболее велика доля слабоструктурированных проблем в системах третьего типа. Следовательно, наиболее сложна практика внедрения результатов системных исследований в этих системах.

При внедрении результатов системного анализа необходимо иметь ввиду следующее обстоятельство. Работа осуществляется на клиента (заказчика), обладающего властью, достаточной для изменения системы теми способами, которые будут определены в результате системного анализа. В работе должны непосредственно участвовать все заинтересованные стороны. Заинтересованные стороны — это те, кто отвечает за решение проблемы, и те, кого эта проблема непосредственно касается. В результате внедрения системных исследований необходимо обеспечить улучшение работы организации заказчика с точки зрения хотя бы одной из заинтересованных сторон, при этом не допускаются ухудшения этой работы с точки зрения всех остальных участников проблемной ситуации.

Говоря о внедрении результатов системного анализа, важно отметить, что в реальной жизни ситуация, когда сначала проводят исследования, а затем их результаты внедряют в практику встречается крайне редко, лишь в тех случаях, когда речь идет о простых системах. Социотехнические системы изменяются с течением времени как сами по себе, так и под влиянием исследований. В процессе проведения системного анализа изменяется состояние проблемной ситуации, цели системы, персональный и количественный состав участников, соотношения между заинтересованными сторонами. Кроме того, следует заметить, что реализация принятых решений влияет на все факторы функционирования системы. Этапы исследования и внедрения в такого типа системах фактически сливаются. То есть идет итеративный процесс. Проводимые исследования оказывают влияние на жизнедеятельность системы и это видоизменяет проблемную ситуацию, ставит новую задачу исследований. Новая проблемная ситуация стимулирует дальнейшее проведение системного анализа и т. д. Таким образом, проблема постепенно решается в ходе активного исследования.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Объясните, почему системность является общим свойством материи.
2. Что такое системные представления?
3. Расскажите о системном подходе и системных исследованиях.
4. Раскройте термин «системный анализ».
5. Какие Вы знаете определения системного анализа?
6. Из каких этапов состоит системный анализ?
7. Как Вы представляете себе методику системного анализа?
8. Перечислите процедуры системного анализа.
9. В чем состоит формулирование проблемы при системном анализе?
10. Дайте определение цели системного анализа.
11. Каков порядок анализа структуры системы?
12. С какой целью осуществляется сбор данных о функционировании системы?
13. Какие Вы знаете методы анализа информационных потоков?
14. Зачем необходимо построение модели системы?
15. Как осуществляется проверка адекватности модели системы?
16. В чем состоит анализ неопределенности и чувствительности модели системы?
17. Расскажите о видах ресурсов, используемых при реализации задач системного анализа.
18. С какой целью выполняется формирование критериев для системного анализа?
19. Каким образом производится генерирование альтернатив при системном анализе?
20. Перечислите методы, используемые в системном анализе для проведения работы по формированию множества альтернатив.
21. Какие сложности возникают при решении задач выбора и принятия решений?
22. Каким образом осуществляется внедрение результатов системного анализа?

Глава 4

МОДЕЛИ АНАЛИЗА СТРУКТУРЫ АСУ

В этой главе изложены цели и задачи структурного анализа АСУ с позиций организации, функций управления, используемых алгоритмов, с позиций используемых технических средств.

Рассмотрены три обычно используемых уровня описания структуры АСУ. Приводится аппарат формализации описания структуры, для чего излагаются минимально необходимые сведе-

ния из теории графов. Рассмотрен алгоритм упорядочивания, позволяющий формально разместить элементы структуры по уровням структуры. Изложена последовательность введения числовой функции, что позволяет определять максимальные и минимальные пути на структуре. Приведенный алгоритм декомпозиции структуры АСУ позволяет минимизировать число элементов, составляющих систему, исходя только из структурных свойств этой системы. Методика анализа потоков информации является мощным средством решения множества практических задач. Рассмотренные математические модели структурно-топологических характеристик систем позволяют количественно оценить качество структуры и ее элементов с позиций общесистемного подхода. Предложена методика построения моделей функционирования организационных систем.

В результате изучения материала данной главы студенты должны знать принципы, методы и средства системного анализа, основные классы моделей организационных структур, методы их формализации и алгоритмизации. Они должны уметь выполнять общесистемный анализ и необходимые общесистемные расчеты.

4.1. Цели и задачи структурного анализа АСУ

Одной из главных задач структурного анализа АСУ является построение наглядной формальной модели, отображающей процесс взаимодействия между элементами или подсистемами, составляющими систему, а также их взаимодействие с внешней средой.

При создании АСУ их структурные модели могут рассматриваться с различных позиций: с позиции организации, функций управления, используемых алгоритмов, используемых технических средств и т.п. В соответствии с этим могут быть выделены следующие аспекты структуры одной и той же системы и, как следствие, следующие модели:

- организационная структура и ее модель;
- функциональная структура и ее модель;
- алгоритмическая структура и ее модель;
- техническая структура и ее модель и т.п.

Организационная структура. Как правило, она отображает собой структуру управления, которая сложилась на объекте автоматизации (например, на предприятии) и которая совершенствуется при внедрении АСУ. Эта структура является основной и именно с нее следует начинать анализ и последующий синтез АСУ.

Основными элементами организационной структуры являются подразделения аппарата управления и отдельные лица-операторы, которые связаны с анализом документооборота и процедурой принятия управленческих решений.

Основные направления совершенствования организационной структуры:

- сокращение излишних подразделений или операторов;
- минимизация связей между этими элементами системы (подразделения и операторы);
- повышение пропускной способности этих связей;
- упорядочение документооборота;
- ликвидация циклов в движении документов и т.п.

При анализе организационной структуры решаются следующие задачи:

- описание состава организации и построение ее структурной схемы;
- определение функций отдельных подразделений и операторов;
- описание материальных и информационных связей;
- построение обобщенной структурной информационной модели организации.

Функциональная структура. Она отображает функции, выполняемые отдельными элементами системы (подразделениями и операторами) в составе организационной структуры. В принципе одной и той же организационной структуре может соответствовать несколько различных функциональных структур.

Основные направления совершенствования функциональной структуры:

- устранение параллелизма в выполнении функций управления;
- освобождение элементов системы от выполнения функций, не предусмотренных соответствующими должностными обязанностями;
- перераспределение функций управления с целью оптимизации;
- создание максимально четких контуров ответственности.

При анализе функциональной структуры решаются следующие основные задачи:

- изучаются функции управления в структурных подразделениях системы;
- выбирается состав функций, подлежащих автоматизации;
- определяется взаимосвязь этих функций;
- составляется обобщенная функциональная структура задач управления.

Алгоритмическая структура. Она отображает совокупность используемых алгоритмов и последовательность их декомпозиции, что позволяет в дальнейшем перейти к созданию программного обеспечения.

Основные направления совершенствования алгоритмической структуры:

- использование стандартных (типовых) алгоритмов обработки информации;
- повышение точности, скорости и надежности вычислений;
- сокращение требуемых объемов памяти;
- совершенствование отдельных алгоритмов.

При анализе алгоритмической структуры решаются следующие задачи:

- выделение комплексов задач, отдельных задач, алгоритмов, модулей алгоритмов и т. п.;
- определение их информационно-логической взаимосвязи;
- определение последовательности их реализации.

Техническая структура. Техническая структура отображает перечень и взаимосвязь технических устройств, используемых для построения системы. При анализе технической структуры решаются следующие задачи:

- определяются элементы, участвующие в основных информационных процессах: регистрация и подготовка информации, сбор и передача, хранение и обработка, воспроизведение и выдача;
- составляется формальная структурная модель системы технических средств с учетом топологии расположения элементов, информационной и энергетической их взаимосвязи, а также связи с внешней средой.

4.2. Уровни описания структуры АСУ

Независимо от перечисленных аспектов рассмотрения системы, общая задача структурного анализа состоит в том, чтобы, исходя из заданного описания элементов и непосредственных связей между ними, получить заключение о структурных свойствах системы и ее основных подсистем. Применительно к автоматизированным системам используется три уровня их описания:

- наличие связей;
- наличие и направление связей;
- наличие и направление связей и вид и направление движения сигналов, которые определяются взаимодействием элементов.

На первом уровне, когда исходят лишь из наличия или отсутствия связей, изучаемой системе может быть поставлен в соответствие неориентированный граф, вершины которого отображают элементы системы, а ребра — существующие между элементами связи.

Основные задачи, решаемые на этом уровне:

- определение связности (целостности) системы. Если система оказывается несвязной, то ставят задачу выделения изолированных связных подсистем со списками входящих в них элементов;
- выделение циклов;
- определение минимальных и максимальных последовательностей элементов (цепей), разделяющих элементы друг от друга.

На втором уровне, когда задано направление связей, изучаемой системе соответствует ориентированный граф, направление дуг которого совпадает с направлением связей.

Результаты структурного анализа на этом уровне оказываются более содержательными. К задачам анализа на этом уровне относят:

- определение связности (целостности) системы;
- топологическая декомпозиция системы с выделением сильно связанных подсистем;
- нахождение входных и выходных полюсов системы и в соответствии с этим выделение пунктов приема и выдачи информации;
- выделение уровней в системе и определение их взаимосвязей;
- определение максимальных и минимальных путей;
- определение характеристик топологической значимости элементов;
- получение информации о слабых местах в структуре.

На третьем уровне описания связей не только учитывается наличие и направление связей, но и раскрывается состав и характер сигналов взаимодействия элементов. Система отображается с помощью специально вводимых схем или моделей сопряжения.

Основные задачи на этом уровне:

- определение характера сигналов (входные, выходные, управляющие и т.п.);
- построение моделей функционирования элементов системы и самой системы.

4.3. Формализация описания структуры методами теории графов

Принцип представления структуры в виде графа чрезвычайно прост. Чаще всего элементам системы ставят в соответствие вершины графа, а связям — ребра.

Рассмотрим некоторые основные определения, непосредственно связанные с задачами структурного анализа АСУ.

4.3.1. Способы формализованного задания графа

А. Графическое представление.

Это наиболее наглядный способ представления отношений между элементами, его недостаток — относительная сложность использования ЭВМ для анализа.

Б. Матричное представление.

Матрица смежности вершин для неориентированного графа имеет вид:

$$A = \|a_{ij}\|, \quad i, j = \overline{1, n},$$

где n — число вершин графа,

$$a_{ij} = \begin{cases} 1, & \text{если существует связь между } i \text{ и } j \text{ вершинами;} \\ 0, & \text{в противном случае.} \end{cases}$$

Для ориентированного графа матрица смежности

$$A = \|a_{ij}\|, \quad i, j = \overline{1, n},$$

задается следующим образом:

$$a_{ij} = \begin{cases} 1, & \text{если из вершины } i \text{ можно перейти в вершину } j; \\ 0, & \text{в противном случае.} \end{cases}$$

Матрица инциденций

$$B = \|b_{ij}\|, \quad i = \overline{1, n}, \quad j = \overline{1, m},$$

где n — число вершин, m — число ребер, определяется следующим образом:

для неориентированного графа:

$$b_{ij} = \begin{cases} 1, & \text{если } i\text{-тая вершина инцидентна } j\text{-тому ребру} \\ 0, & \text{в противном случае.} \end{cases}$$

для ориентированного графа:

$$b_{ij} = \begin{cases} 1, & \text{если } i\text{-тая вершина — начало } j\text{-того ребра;} \\ -1, & \text{если } i\text{-тая вершина — конец } j\text{-того ребра;} \\ 0, & \text{если } i\text{-тая вершина не инцидентна } j\text{-тому ребру.} \end{cases}$$

Рис. 4.1. иллюстрирует это положение.

В. Множественное представление.

В этом случае для ориентированного графа $G(V)$ задается множество вершин V и соответствие G , которое показывает, как связаны между собой вершины. Для каждой вершины i соответствие G определяет множество вершин $G(i)$, в которые можно непосредственно попасть из вершины i . Это множество называется множеством правых инцидентий.

Множество $G^{-1}(i)$ определяет все вершины, из которых можно непосредственно попасть в вершину i . Это множество называется множеством левых инцидентий.

Таким образом, ориентированный граф задается перечислением (списком) множеств вида $G(i)$, либо множеств вида $G^{-1}(i)$ для всех вершин графа. Такой способ оказывается наиболее компактным и эффективным при задании исходной информации о структуре для решения задач синтеза, особенно для иерархических структур.

Пример. Пусть структура системы имеет вид, показанный на рис. 4.2. Необходимо представить ее рассмотренными способами.

Строим граф системы (рис. 4.3), матрицы смежности (табл. 4.1) и инцидентий (табл. 4.2).

Множественное задание структуры: $G(1) = (2,3)$, $G(2) = 0$, $G(3) = (5,4)$, $G(4) = (2)$, $G(5) = (1,2)$. Или $G^{-1}(1) = (5)$, $G^{-1}(2) = (1, 5, 4)$, $G^{-1}(3) = (1)$, $G^{-1}(4) = (3)$, $G^{-1}(5) = (3)$.

Таблица 4.1

$i \backslash j$	1	2	3	4	5
1	0	1	1	0	0
2	0	0	0	0	0
3	0	0	0	1	1
4	0	1	0	0	0
5	1	1	0	0	0

Таблица 4.2

$i \backslash j$	1	2	3	4	5	6	7
1	1	1	-1	0	0	0	0
2	-1	0	0	-1	-1	0	0
3	0	-1	0	0	0	1	1
4	0	0	0	0	1	0	-1
5	0	0	1	1	0	-1	0

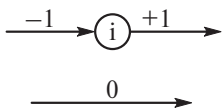


Рис. 4.1. Правило построения b_{ij}

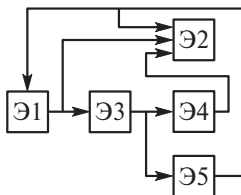


Рис. 4.2. Структура системы

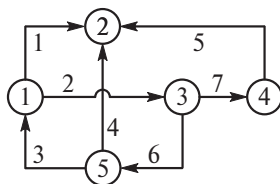


Рис. 4.3. Граф системы

Определение цепи, пути, цикла, контура. *Цепью* называется такая последовательность ребер $E_0, E_1, \dots, E_{k-1}, E_k$, когда каждое ребро E_{k-1} соприкасается одним из концов с ребром E_k . Цепь можно обозначить последовательностью вершин, которые она содержит. Например, для графа, представленного на рис. 4.4, цепью будет (1, 4, 3, 5) или (1, 3, 4) (рис. 4.5).

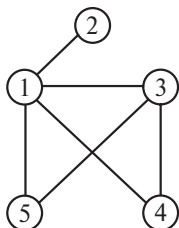


Рис. 4.4. Вид графа

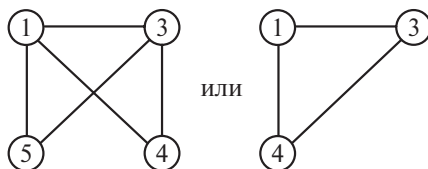


Рис. 4.5. Вид цепи

Понятие цепи обычно используется для неориентированных графов.

Путем называется такая последовательность дуг, когда конец каждой предыдущей дуги совпадает с началом последующей. Например, для графа (рис. 4.3) последовательность дуг (1, 3), (3, 4), (4, 2) является путем (рис. 4.6). Понятие пути обычно используется для ориентированных графов.

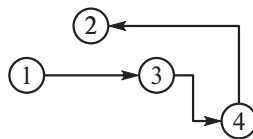


Рис. 4.6. Вид пути

Циклом называется такая конечная цепь, которая начинается и заканчивается в одной вершине. Например, на рис. 4.5 цепь (1, 4, 3) является циклом. Понятие цикла имеет смысл только для неориентированных графов.

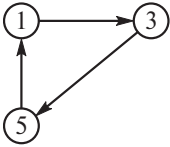


Рис. 4.7.
Вид контура

Контуром называется такой конечный путь, у которого начальная вершина первой дуги совпадает с конечной вершиной последней дуги. Например, для графа (рис. 4.3) последовательность дуг (1, 3), (3, 5), (5, 1) есть контур (рис. 4.7).

Длиной цепи (пути) называют число ребер (дуг), входящих в цепь (путь) графа.

Матрица смежности вершин графа A является матрицей непосредственных путей графа, имеющих длину, равную единице. Общее число транзитных путей длиной λ может быть получено в результате возведения в λ -тую степень матрицы A :

$$A^\lambda = A^{\lambda-1} A.$$

Элемент матрицы $A^\lambda a_{ij}(\lambda)$ определяет число путей длиной λ от вершины i к вершине j .

На рис. 4.8 приведен пример определения элементов матрицы.

Степень вершины. Число ребер, инцидентных вершине i неориентированного графа, называют *степенью вершины i* и обозначают $\rho(i)$.

Для графа, представленного на рис. 4.4:

$$\rho(1) = 4; \rho(2) = 1; \rho(3) = 3; \rho(4) = 2; \rho(5) = 2. \quad \sum_{i=1}^5 \rho(i) = 12.$$

Или в общем виде $0,5 \sum_{i=1}^n = m$, где n — число вершин, m — число ребер графа.

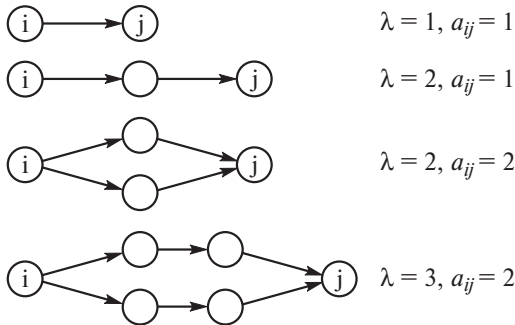


Рис. 4.8. Пример элементов матрицы A^λ

Число дуг ориентированного графа, которые имеют начальной вершиной вершину i , называют полустепенью исхода вершины i и обозначают через $\rho^+(i)$. Аналогично, число дуг, которые имеют своей конечной вершиной вершину j , называют полустепенью захода вершины j и обозначают через $\rho^-(j)$. Для графа, представленного на рис. 4.3:

$$\rho^+(1) = 2; \rho^+(2) = 0; \rho^+(3) = 2; \rho^+(4) = 1; \rho^+(5) = 2; \sum_{i=1}^n \rho^+(i) = 7.$$

$$\rho^-(1) = 1; \rho^-(2) = 3; \rho^-(3) = 1; \rho^-(4) = 1; \rho^-(5) = 1; \sum_{j=1}^n \rho^-(j) = 7.$$

Или в общем виде $\sum_{i=1}^n \rho^+(i) = \sum_{j=1}^n \rho^-(j)$, где m — число дуг графа,

n — число вершин графа.

К понятию связности графа. Для неориентированных графов вводится понятие слабой связности или просто связности. Граф $G(V)$ называется слабо связным (связным), если для любых вершин графа i и j существует цепь из вершины i в вершину j .

Для ориентированных графов вводится дополнительное понятие сильной связности. Граф $G(V)$ называется сильно связным, если для любых вершин графа i, j существует путь из вершины i в вершину j .

Граф на рис. 4.4 является слабо связным. На рис. 4.9 представлен сильно связный граф, на рис. 4.10 — несвязный, распадающийся на два сильно связных подграфа.

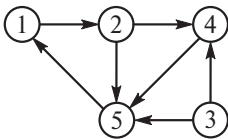


Рис. 4.9. Вид сильно связного графа

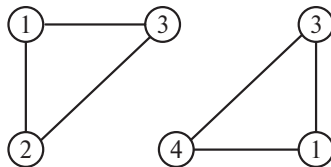


Рис. 4.10. Вид несвязного графа, распадающегося на два сильно связных подграфа

4.3.2. Порядковая функция на графе

Целью введения порядковой функции на графе без контуров является разбиение множества вершин графа на непересекающиеся подмножества, упорядоченные таким образом, что если вершина входит в подмножество с номером i , то следующая за ней вершина — в подмножество с номером, большим i . Полученные непересекающиеся подмножества называются *уровнями*.

Алгоритм упорядочения (или алгоритм введения порядковой функции) сводится к следующему:

- в подмножество нулевого уровня N_0 включаются все вершины i , для которых $G^{-1}(i) = 0$ (иначе говоря, для которых не существует множества левых инцидентий, или, еще проще, — вершины, в которые ниоткуда нельзя попасть). Проводится последовательная нумерация этих вершин: $1, 2, \dots, i$;
- в подмножество первого уровня N_1 включаются все вершины i , для которых $G^{-1}(i) \in N_0$, т. е. для которых вершины уровня N_0 являются множеством левых инцидентий. Проводится последовательная нумерация этих вершин: $i + 1, i + 2, \dots, i + r$;
- в подмножество второго уровня N_2 включаются все вершины i , для которых $G^{-1}(i) \in (N_0 \vee N_1)$. Проводится последовательная нумерация вершин: $i + r + 1, i + r + 2, \dots, i + r + p$;
- в подмножество третьего уровня N_3 включаются все вершины i , для которых $G^{-1}(i) \in (N_0 \vee N_1 \vee N_2)$. Проводится последовательная нумерация вершин и т. д.

Данный процесс повторяется до тех пор, пока не будут пронумерованы все вершины графа.

Рассмотренный алгоритм нумерации приводит к тому, что в матрице смежности вершин графа $a_{ij} = 0$ при $i > j$, т. е. матрица становится треугольной.

Для графов, имеющих контуры, сначала необходимо выделить сильно связанные подграфы (см. ниже «Топологическая декомпозиция структур»). И, рассматривая эти выделенные подсистемы как элементы системы, для них вводить порядковую функцию.

Пример введения порядковой функции. В результате обследования некоторой организационной системы был получен граф информационно-логической взаимосвязи между решаемыми задачами, представленный на рис. 4.11. Необходимо определить, в какой последовательности следует решать эти задачи, решение каких задач следует начинать одновременно,

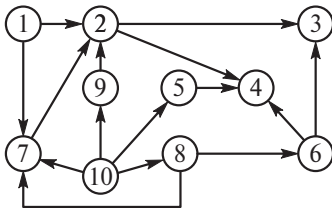


Рис. 4.11. Вид неупорядоченного графа

необходимое число копий решений, сколько тактов следует хранить результаты решения задачи.

Составляем матрицу смежности анализируемого графа (табл. 4.3).

Таблица 4.3

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

В соответствии с рассмотренным алгоритмом переходим к множественному представлению графа. (Напомним, что множество левых инциденций $G^{-1}(i)$ определяет все вершины, из которых можно непосредственно попасть в вершину i .) Из исходного множественного представления удаляем пустое множество левых инциденций и соответствующие этому множеству вершины. Получаем следующий столбец, над которым продельваем аналогичную операцию и т. д. Удаляемым вершинам последовательно присваиваются новые номера.

$$\begin{aligned} G^{-1}(1) &= (0) \\ G^{-1}(2) &= (1, 7, 9) & G^{-1}(2) &= (7, 9) & G^{-1}(2) &= (7) \\ G^{-1}(3) &= (2, 6) & G^{-1}(3) &= (2, 6) & G^{-1}(3) &= (2,6) \\ G^{-1}(4) &= (2, 5, 6) & G^{-1}(4) &= (2, 5, 6) & G^{-1}(4) &= (2,6) \\ G^{-1}(5) &= (10) & G^{-1}(5) &= (0) \\ G^{-1}(6) &= (8) & G^{-1}(6) &= (8) & G^{-1}(6) &= (0) \\ G^{-1}(7) &= (1, 8, 10) & G^{-1}(7) &= (8) & G^{-1}(7) &= (0) \\ G^{-1}(8) &= (10) & G^{-1}(8) &= (0) \\ G^{-1}(9) &= (10) & G^{-1}(9) &= (0) \\ G^{-1}(10) &= (0) \\ \\ G^{-1}(2) &= (0) \\ G^{-1}(3) &= (2) & G^{-1}(3) &= (0) \\ G^{-1}(4) &= (2) & G^{-1}(4) &= (0) \end{aligned}$$

Результаты преобразований сведены в табл. 4.4.

Таблица 4.4

Уровень	Условия включения	Включаемые вершины	Новая нумерация
N_0	$G^{-1}(i) = 0$	(1, 10)	(1, 2)
N_1	$G^{-1}(i) \in N_0$	(5, 8, 9)	(3, 4, 5)
N_2	$G^{-1}(i) \in N_0 \vee N_1$	(6, 7)	(6, 7)
N_3	$G^{-1}(i) \in N_0 \vee N_1 \vee N_2$	(2)	(8)
N_4	$G^{-1}(i) \in N_0 \vee N_1 \vee N_2 \vee N_3$	(3, 4)	(9, 10)

На основании табл. 4.4 строим преобразованный граф. Его вершины в новом обозначении размещаем по найденным уровням (внутри кружков помещаем новые обозначения, рядом — старые). Соединяем старые обозначения вершин дугами в соответствии с ранее найденной матрицей смежности.

Строим матрицу смежности упорядоченного графа (табл. 4.5). Убеждаемся в том, что она оказывается треугольной.

Таблица 4.5

$$A = \begin{pmatrix} * & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & * & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & * & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & * & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & * & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & * \end{pmatrix}$$

где **...* — главная диагональ матрицы смежности.

Если рассматривать уровни как такты движения информации, то рис. 4.12 непосредственно дает ответы на вопросы, сформулированные в начале примера.

Примечание. Задача упорядочения может быть решена с помощью матрицы инцидентий. Алгоритм упорядочения в этом случае выглядит следующим образом:

1. Составляется матрица инцидентий по правилам, изложенным выше.
2. Из матрицы вычеркиваются строчки, состоящие только из 0 и +1, и столбцы, соответствующие +1.
3. Отмечается порядок вычеркивания.
4. На последнем этапе на соответствующем уровне размещаются оставшиеся вершины.
5. Уровень будет равен порядку вычеркивания минус единица.

В качестве **примера** рассмотрим граф, представленный на рис. 4.13.

На основании рис 4.13 строим матрицу инцидентий $B = \|b_{ij}\|$ (табл. 4.6).

Таблица 4.6

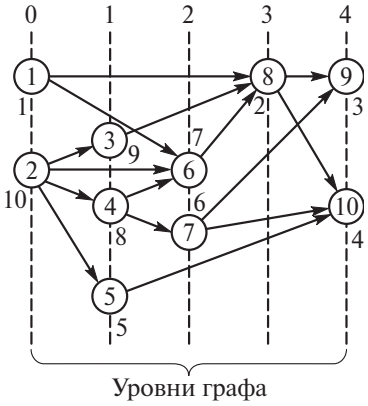
$$B = \|b_{ij}\| = \begin{vmatrix} +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & +1 & +1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & +1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & 0 & +1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$


Рис. 4.12. Вид упорядоченного графа

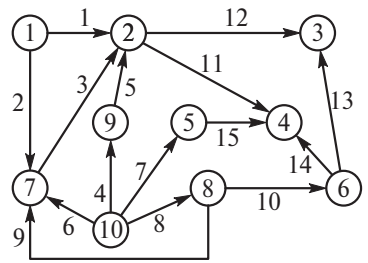


Рис. 4.13. Вид неупорядоченного графа

Первое вычеркивание. Вычеркнуты вершины 1 и 10 (табл. 4.7).
 * — обозначение пустой клетки.

Таблица 4.7

*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	-1	*	-1	*	*	*	0	0	+1	+1	0	0	0
*	*	0	*	0	*	*	*	0	0	0	-1	-1	0	0
*	*	0	*	0	*	*	*	0	0	-1	0	0	-1	-1
*	*	0	*	0	*	*	*	0	0	0	0	0	0	+1
*	*	0	*	0	*	*	*	0	-1	0	0	+1	+1	0
*	*	+1	*	0	*	*	*	-1	0	0	0	0	0	0
*	*	0	*	0	*	*	*	+1	+1	0	0	0	0	0
*	*	0	*	+1	*	*	*	0	0	0	0	0	0	0
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Второе вычеркивание. Вычеркнуты вершины 5, 8 и 9 (табл. 4.8).

Таблица 4.8

*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	-1	*	*	*	*	*	*	*	+1	+1	0	0	*
*	*	0	*	*	*	*	*	*	*	0	-1	-1	0	*
*	*	0	*	*	*	*	*	*	*	-1	0	0	-1	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	0	*	*	*	*	*	*	*	0	0	+1	+1	*
*	*	+1	*	*	*	*	*	*	*	0	0	0	0	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Третье вычеркивание. Вычеркнуты вершины 6 и 7 (табл. 4.9)

Таблица 4.9

*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	+1	+1	*	*	*
*	*	*	*	*	*	*	*	*	*	0	-1	*	*	*
*	*	*	*	*	*	*	*	*	*	-1	0	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Результат четвертого вычеркивания. Вычеркнута вершина 2 (табл. 4.10).

Таблица 4.10

*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Оставшиеся вершины 3 и 4 размещаются на следующем уровне.

Полученный результат использования алгоритма вычеркивания сводим в табл. 4.10 (а).

Таблица 4.10 (а)

Порядок вычеркивания	1	2	3	4	5
Вершины	1, 10	5, 8, 9	6, 7	2	3, 4
Уровни	0	1	2	3	4

4.3.3. Числовая функция на графе

Числовую функцию на графе задают обычно либо на вершинах, либо на дугах (ребрах) графа.

Числовая функция на вершинах графа считается заданной, если каждой i -той вершине a_i графа $G(V)$, $a_i \in V$, ставится в соответствие некоторое число $l_i = l(a_i)$ из некоторого множества L .

Числовая функция на дугах (ребрах) для ориентированного (неориентированного) графа считается заданной, если каждой дуге $(a_i a_j)$ или ребру ставится в соответствие число $q = q(a_i a_j)$ из некоторого множества Q . В некоторых случаях числовая функция на графе задается комбинированным способом как на вершинах, так и на дугах.

Значение функции на пути S через вершины $a_1, a_2, \dots, a_i, \dots, (a_i \in S)$ при задании числовой функции на вершинах графа определяется либо в соответствии с аддитивной формой

$$l_S = \sum_{a_i \in S} l(a_i),$$

либо в соответствии с мультипликативной формой

$$l_S = \prod_{a_i \in S} l(a_i),$$

Аналогичным образом определяется значение функции на пути через вершины $a_1, a_2, \dots, a_i, \dots, (a_i \in S)$ при задании числовой функции на дугах (ребрах) графа:

$$q_s = \sum_{(a_i a_j) \in S} q(a_i a_j),$$

$$q_s = \prod_{(a_i a_j) \in S} q(a_i a_j).$$

В соответствии с данными определениями может быть поставлена задача нахождения путей через множество вершин (дуг), обладающих определенным свойством, с максимальным (минимальным) значением числовой функции. Такие пути называются максимальными (минимальными). Определение максимальных (минимальных) путей на графе чаще всего формализуется в виде задачи динамического программирования. При этом предполагается, что все вершины в графе упорядочены, и контура в графе отсутствуют. Рассмотрим пример.

Пример нахождения максимального пути. Пусть в задаче календарного планирования возникла необходимость определения максимального пути из вершины a_1 в вершину a_7 для графа, представленного на рис. 4.13а.

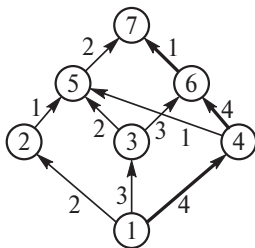


Рис. 4.13а. К нахождению пути максимальной длины

Для вершины a_1 принимаем $q_S^{\max}(a_1 a_1) = 0$. Для вершин a_2, a_3, a_4 $q_S^{\max}(a_1 a_2) = 2$; $q_S^{\max}(a_1 a_3) = 3$; $q_S^{\max}(a_1 a_4) = 4$. Для вершины a_5 $q_S^{\max}(a_1 a_5) = \max(2+1; 3+2; 4+1) = 5$. Для вершины a_6 $q_S^{\max}(a_1 a_6) = \max(3+3; 4+4) = 8$. Для вершины a_7 $q_S^{\max}(a_1 a_7) = \max(5+2; 8+1) = 9$.

Значение функции на максимальном пути для данного примера равно девяти. Этот путь выделен жирными стрелками.

4.4. Топологическая декомпозиция структур АСУ

Проведение топологической структуры АСУ, представленной в виде ориентированного графа $G(V)$, связано с выделением в этой структуре сильно связных подсистем. Напомним, что ориентированный граф $G(V)$ называется сильно связным, если для любых вершин i, j существует путь из вершины i в вершину j .

Для рассмотрения основного алгоритма декомпозиции целесообразно ввести следующие понятия.

Множество вершин, достижимых из вершины i , называется *достижимым множеством* $R(i)$. Достижимое множество определяется следующим образом:

$$R(i) = (i) \vee G(i) \vee G^2(i) \vee \dots \vee G^\lambda(i) \vee \dots, \quad (4.1)$$

где $G(i)$ — множество вершин, достижимых из вершины i с использованием пути длиной, равной единице; G^λ — множество вершин, достижимых из вершины i с помощью путей длиной λ .

При этом предполагается, что сама вершина i достижима с помощью пути, длиной 0 и включена во множество $R(i)$. Это предположение отражается в соотношении (4.1) введением (i) .

В соответствии с выражением (4.1) множество $R(i)$ может быть получено последовательным слева направо объединением множеств до тех пор, пока текущее множество $R(i)$ не перестанет увеличиваться по размеру при выполнении очередной операции объединения. Число объединений, естественно, зависит от вида графа, но, очевидно, всегда $\lambda \leq n$, где n — число вершин графа.

Контрдостижимым множеством $Q(i)$ графа $G(V)$ называется множество таких вершин, когда из любой вершины этого множества можно достигнуть вершину i .

Аналогично построению $R(i)$ можно построить $Q(i)$, используя следующее выражение:

$$Q(i) = (i) \vee G^{-1}(i) \vee G^{-2}(i) \vee \dots \vee G^{-\lambda}(i) \vee \dots, \quad (4.2)$$

где $G^{-1}(i)$ — множество вершин, из которых можно достигнуть i -тую вершину с помощью путей, длина которых равна единице, $G^{-2}(i)$ — то же самое, но с помощью путей, длина которых равна двум и т.д. (рис. 4.14).

Так как $R(i)$ является множеством вершин, достижимых из i -ой вершины, а $Q(j)$ — множеством вершин, из которых можно достичь вершину j , то множество $R(i) \cap Q(j)$ является множеством таких вершин, каждая из которых принадлежит по крайней мере одному пути, идущему от i -той вершины к j -той, что иллюстрируется рис. 4.15. Эти вершины называются существенными или неотъемлемыми относительно двух кольцевых вершин i и j .

В свою очередь, множество

$$R(i) \cap Q(j) \quad (4.3)$$

определяет сильно связный подграф графа $G(V)$, содержащий i -тую вершину, поскольку все существенные вершины, принадлежащие множеству (4.3), достижимы из i -той вершины и, кроме того, из каждой такой вершины достижима вершина i , т.е. все эти вершины взаимодостижимы (рис. 4.16).

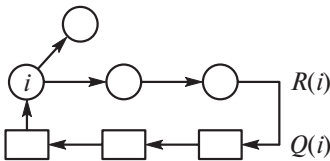


Рис. 4.14. Вид достижимых и контрдостижимых множеств

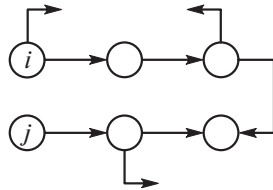


Рис. 4.15. Вид существенных вершин

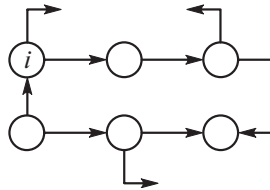


Рис. 4.16. Вид сильно связного подграфа

Из введенных выше определений имеем следующий алгоритм декомпозиции.

Алгоритм декомпозиции:

1. В исходном графе $G(V)$ производим нумерацию вершин.
2. Для i -той вершины ($i = 1$) определяем множество $R(1)$ и множество $Q(1)$.
3. Находим сильно связный подграф G_1 , включающий множество вершин $V_1 = R(1) \cap Q(1)$.
4. Все вершины, принадлежащие $G_1(V_1)$, удаляются из исходного графа $G(V)$.

Далее пункты 2, 3, 4 повторяются для $i = 2, 3, 4, \dots$ до тех пор, пока все вершины исходного графа не будут сгруппированы в соответствующие сильно связные подграфы.

Пример топологической декомпозиции. Пусть в распределенной АСУ пункты обработки информацией обмениваются данными так, как это изображено с помощью графа, представленного на рис. 4.17. Возникла необходимость в сокращении числа этих пунктов, исходя только из структурных свойств анализируемой системы. (Объединение будем производить без учета производительности, надежности и т. п., учитывая только структурные свойства схемы.)

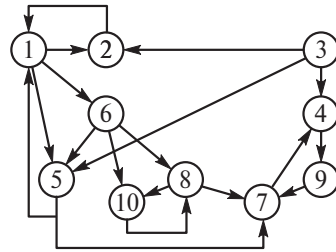


Рис. 4.17. Вид исходного графа

В соответствии с рассмотренным алгоритмом определяем множества $R(i)$ и $Q(i)$.

Полагаем $i = 1$ и находим, используя формулы (4.1) и (4.2), достижимое $R(1)$ и контрдостижимое $Q(1)$ множества:

$$R(1) = (1, 2, 4, 5, 6, 7, 8, 9, 10), \quad Q(1) = (1, 2, 3, 5, 6).$$

Используя соотношение (4.3), находим сильно связный подграф, содержащий вершину 1:

$$V_1 = R(1) \cap Q(1), \quad V_1 = (1, 2, 5, 6).$$

После удаления сильно связного подграфа $G_1(V_1)$ исходный граф $G(V)$ имеет вид (рис. 4.18).

Полагаем $i = 2$, но вершина 2 входит в выделенный подграф V_1 , следовательно, $i = 3$.

$R(3) = (3, 4, 7, 9)$, $Q(3) = (3)$, $V_2 = (3)$. Затем удаляем сильно связный подграф $G_2(V_2)$ (рис. 4.19).

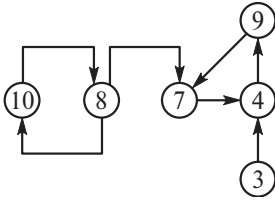


Рис. 4.18. Вид графа после удаления $G_1(V_1)$

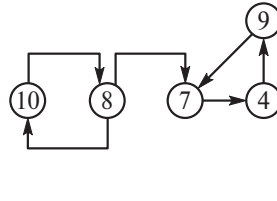


Рис. 4.19. Вид графа после удаления $G_1(V_1)$ и $G_2(V_2)$

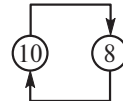


Рис. 4.20. Вид графа после удаления $G_1(V_1)$, $G_2(V_2)$ и $G_3(V_3)$

Полагаем $i = 4$, тогда $R(4) = (4, 7, 9)$, $Q(4) = (4, 7, 8, 9, 10)$, $V_3 = (4, 7, 9)$. Удаляем сильно связный подграф $G_3(V_3)$ (рис.4.20). Полагаем $i = 8$, тогда $R(8) = (8, 10)$, $G(8) = (8, 10)$, $V_4 = (8, 10)$. Итак, окончательно имеем:

1. $G_1(V_1) = G_1(1, 2, 5, 6)$:

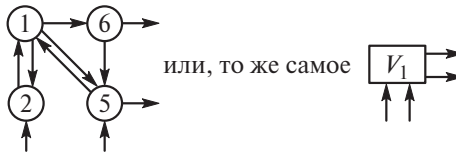


Рис. 4.21. Вид подграфа G_1

2. $G_2(V_2) = G_2(3)$:

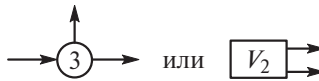


Рис. 4.22. Вид подграфа G_2

3. $G_3(V_3) = G_3(4, 7, 9)$:

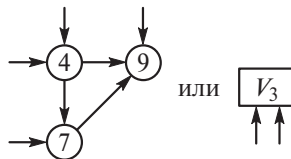


Рис. 4.23. Вид подграфа G_3

$$4. G_4(V_4) = G_4(8, 10):$$

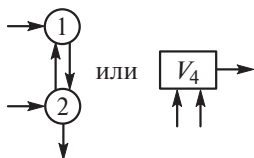


Рис. 4.24. Вид подграфа G_4

Объединяем полученные сильно связанные подграфы в соответствии с исходным графом:

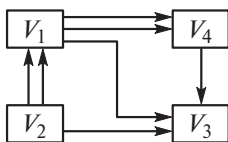


Рис. 4.25. Вид сильно связанных подграфов

Или окончательно:

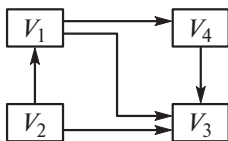


Рис. 4.26. Результат декомпозиции исходного графа

4.5. Модели описания и анализа потоков информации в АСУ

Основным носителем информации в организационных системах является документ. Как для всей системы в целом, так и для ее отдельных подсистем, все документы можно разделить на следующие группы: входные документы, возникающие вне системы и поступающие на ее входы; выходные документы, т. е. документы, покидающие систему; промежуточные документы, которые

используются для целей управления и учета, а также для формирования выходных документов.

Между документами существуют отношения вхождения и порядка.

Отношение вхождения

$$X_j = X_{j1}, X_{j2}, \dots, X_{jn}$$

означает, что документ X_j каким-то образом формируется из документов $X_j = X_{j1}, X_{j2}, \dots, X_{jn}$.

Отношение порядка « X_i следует за X_j » $\{\rightarrow (X_j) \rightarrow (X_i) \rightarrow\}$ означает, что документ X_i может быть сформирован только тогда, когда закончится формирование документа X_j .

Потоки информации в организационных системах образуются не только движением документов, но и движением элементов документов (реквизитов, показателей), а также различных сообщений, данных и т. п. Будем поэтому говорить об элементах потоков информации, имея в виду все перечисленные объекты.

Если элементам потока информации X_1, X_2, \dots, X_n поставить в соответствие вершины графа X_1, X_2, \dots, X_n и каждую пару вершин X_i и X_j соединить дугой, идущей от X_i к X_j , в том и только том случае, если X_i является входом для X_j , то полученный граф называют *информационным графом*.

Для использования формальной методики анализа информационных потоков необходимо провести реальное обследование этих потоков на объекте автоматизации и затем представить полученные результаты в виде матрицы смежности информационного графа

$$A = \|a_{ij}\|, \quad i, j = 1, n.$$

Будем формально, т.е. используя соответствующий аппарат высшей алгебры, возводить эту матрицу в степень N до тех пор, пока

$$A^N \neq 0, \quad \text{а} \quad A^{N+1} = 0,$$

если информационный граф не содержит контуров, либо $n \geq N$ раз, если они есть, где n — число вершин.

Для возведения матрицы A в степень удобно воспользоваться рекуррентным соотношением

$$A^\lambda = A^{\lambda-1} \cdot A.$$

Определяем *матрицу достижимости* $A(B)$ как сумму одноименных элементов всех предыдущих матриц.

Полученная последовательность матриц позволяет формально определить все свойства анализируемого графа.

Рассмотрим эти свойства.

1. *Порядком* π_j элемента X_j (для краткости в дальнейшем просто j) называется длина наибольшего пути, связывающего j -тый элемент с i -тым ($i, j = 1, n, i \neq j$).

Формально π_j определяется с помощью следующего соотношения:

$$\left. \begin{aligned} \sigma_i(1 = \pi_j) &> 0; \\ \sigma_i(1 = \pi_j + 1) &= 0. \end{aligned} \right\} \quad (4.4)$$

В этом соотношении $\sigma_j(\lambda)$ — сумма элементов j -того столбца матрицы A , т. е.

$$\sigma_j = \sum_{i=1}^n a_{ij}(\lambda).$$

Например, пусть мы хотим найти элементы первого порядка, стало быть, полагаем $\pi_j = 1$. Обращаемся к 1-й строке соотношения (4.4), то есть $\lambda = 1$, и далее к матрице A^1 . Пусть фрагмент матрицы A^1 имеет следующий вид (табл. 4.11):

Таблица 4.11

$i \backslash j$	1	2	3	4	5
σ_i	0	0	1	3	2

В соответствии с первой строкой выражения (4.4) выписываем те j , для которых $\sigma_j > 0$. Это $j = 3, 4, 5$.

Теперь обращаемся ко 2-й строке соотношения (4.4). Добавляем 1, следовательно, $\lambda = 2$ и нам следует обратиться к матрице A^2 .

Пусть фрагмент матрицы A^2 имеет вид (табл. 4.12):

Таблица 4.12

$i \backslash j$	1	2	3	4	5
σ_i	0	0	0	1	2

В соответствии со второй строчкой соотношения (4.4) выписываем такие j , для которых $\sigma_j = 0$. Это $j = 1, 2, 3$. Совместно соотношение (4.4) выполняется только для $j = 3$. Стало быть, в данном примере элемент 3 есть элемент первого порядка. Для того чтобы найти элементы второго порядка, надо положить $\pi_j = 2$ и, обратившись к матрицам A^2 и A^3 , проделать те же самые процедуры.

Физический смысл π_j — это номер такта, к которому «готовы» все составляющие элемента x_j .

2. Число $N = \max_j \pi_j$ называют *порядком информационного графа*. Если для N справедливо соотношение

$$\left. \begin{aligned} A^N &\neq 0 \\ A^{(N+1)} &= 0 \end{aligned} \right\}, \quad (4.5)$$

то соответствующая схема называется N -тактной.

Записанное условие возможно только в случае отсутствия контуров.

3. Признаком контура является появление ненулевых элементов на главной диагонали любой из матриц A^λ . Наличие контура свидетельствует либо об ошибке в обследовании, либо о неправильно спроектированном документообороте. В любом случае необходим содержательный экономический анализ с целью устранения контура.

4. Равенство нулю суммы элементов j -того столбца матрицы смежности, т. е. $\sigma_j (\lambda = 1) = 0$ является формальным признаком для выделения входных элементов потока. Значение $k = \sigma_j (\lambda = 1) > 0$ равняется числу элементов, участвующих в формировании элемента j .

Пусть фрагмент матрицы A имеет вид, представленный в табл. 4.13.

Таблица 4.13

$i \backslash j$	1	2	3	4	5	σ_i
1						2
2						
3						
4						0
5						0
σ_j	0	0	0	2	4	

} Выходы

{ Входы

Из матрицы следует, что элементы 1, 2, 3 – входные элементы и, например, для формирования элемента 5 требуется четыре других элемента.

5. Аналогично свойству 4, равенство нулю суммы элементов i -той строки матрицы смежности информационного графа $\sigma_j(\lambda = 1) = 0$ служит формальным признаком для выделения выходных элементов потока, а число $L = \sigma_i(\lambda = 1) > 0$ равно числу элементов, в которые входит элемент i . Например, судя по предыдущей матрице, элемент 1 используется дважды.

6. Если при некотором $i = j$ одновременно

$$\left. \begin{aligned} \sigma_i(\lambda = 1) &= 0 \\ \sigma_j(\lambda = 1) &= 0 \end{aligned} \right\}$$

то к анализируемой схеме этот элемент не имеет отношения (ошибка обследования).

7. Число путей длиной λ от элемента i к элементу j определяется элементом $a_{ij}(\lambda)$ матрицы A^λ .

8. Число всевозможных путей от элемента i к элементу j определяется элементом $a_{ij}(\Sigma)$ матрицы $A(\Sigma)$, где

$$A(\Sigma) = \sum_{\lambda=1}^N A^\lambda.$$

Матрица $A(\Sigma)$ есть сумма всех одноименных элементов всех матриц A^λ , а элемент этой этой матрицы равен сумме всех одноименных элементов всех матриц.

$$a_{ij}(\Sigma) = \sum_{\lambda=1}^N a_{ij}(\lambda).$$

9. По аналогии со свойствами 4 и 5 отличные от нуля элементы j -того столбца матрицы достижимости $A(\Sigma)$ указывают все элементы потока, которые участвуют в формировании j , а отличные от нуля элементы i -той строки этой матрицы указывают все элементы, при формировании которых используется элемент i .

10. Максимальное значение порядка элементов i -той строки матрицы смежности A , которые не равны нулю, определяет номер такта τ_i , после которого элемент i уже не используется и он может быть «погашен» в памяти системы.

Пример. Пусть задан фрагмент матрицы A (табл. 4.14).

Таблица 4.14

π_j	0	0	3	2	1
$i \backslash j$	1	2	3	4	5
1	0	0	0	1	1

Из табл. 4.14 следует, что π_j поразрядно равняется 00321. Тогда для элемента 1 искомый номер такта $\tau_1 = 3$.

11. Число тактов, в течение которых элемент 1 должен храниться в памяти системы определяется соотношением $t_i = \tau_i - \pi_i$.

Для нашего примера $t_1 = 3 - 0$.

12. Анализ структуры всех путей, связывающих элементы i и j , позволяет выявить как дублирующие связи, так и избыточные элементы. Это позволяет улучшить свойства анализируемого информационного потока.

Пример анализа потоков информации. Пусть схеме движения оперативной отчетности в подсистеме оперативного управления производством соответствует информационный граф, представленный на рис. 4.27. Необходимо формально выявить все свойства данного информационного графа.

Прежде всего, необходимо составить матрицу смежности исходного графа A .

Последовательность матриц $A^{(\lambda)}$ и $A(\Sigma)$ позволяет выяснить свойства анализируемого потока.

Алгоритм формализованного возведения матрицы A в степень.

Для определения i -той строчки матрицы $A^{(\lambda)}$ $a_i(\lambda)$:

1. Выписывается известная нам i -тая строчка матрицы $A^{(\lambda-1)}$ $a_i(\lambda-1)$.

2. В этой выписанной строчке отмечаются ненулевые элементы.

3. Из матрицы A выписываются строчки, соответствующие этим ненулевым элементам.

4. Искомая i -тая строчка будет равняться сумме выписанных строк матрицы A . Причем, если ненулевой элемент не равен единице, то соответствующую строчку надо домножить на величину этого элемента.

Итак, имеем граф

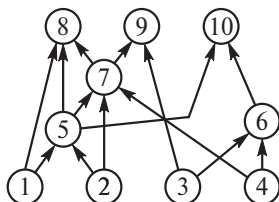


Рис. 4.27. Вид исходного графа

Составляем его матрицу смежности A (табл. 4.15).

Таблица 4.15

π_j	0	0	0	0	1	1	2	3	3	2	
$i \backslash j$	1	2	3	4	5	6	7	8	9	10	σ_i
1					1			1			2
2					1		1				2
3						1			1		2
4						1	1				2
5							1	1		1	3
6										1	1
7								1	1		2
8											0
9											0
10											0
σ_j	0	0	0	0	2	2	3	3	2	2	

Определяем матрицу A^2 : $A^2 = A \cdot A$

$$\begin{array}{r}
 a_1(\lambda = 1) = 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0 \\
 \\
 \\
 a_5(\lambda = 1) = 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \\
 + a_8(\lambda = 1) = 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\
 \hline
 a_1(\lambda = 2) = 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \\
 a_2(\lambda = 1) = 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0 \\
 \\
 a_5(\lambda = 1) = 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \\
 + a_7(\lambda = 1) = 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0 \\
 \hline
 a_2(\lambda = 2) = 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 1\ 1
 \end{array}$$

и т. д.

Окончательно имеем (табл. 4.16):

Таблица 4.16

$i \backslash j$	1	2	3	4	5	6	7	8	9	10	σ_i
1							1	1		1	3
2							1	2	1	1	5
3										1	1
4								1	1	1	3
5								1	1		2
6											0
7											0
8											0
9											0
10											0
σ_j	0	0	0	0	0	0	1	5	3	4	

Определим матрицу A^3 : $A^2 \cdot A$

$$a_1(\lambda = 2) = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix}$$

$$a_7(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$$

$$a_8(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

$$a_{10}(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

$$a_1(\lambda = 3) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$$

$$a_2(\lambda = 2) = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1 & 1 \\ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix}$$

$$a_7(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$$

$$2a_8(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

$$a_{10}(\lambda = 1) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

$$a_2(\lambda = 3) = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$$

и т. д.

Окончательно имеем:

Таблица 4.17

$i \backslash j$	1	2	3	4	5	6	7	8	9	10	σ_i
1							1	1			2
2							1	1			2
3											
4											
5											
6											
7											
8											
9											
10	0	0	0	0	0	0	0	0	0	0	0
σ_j	0	0	0	0	0	0	2	2	0	0	0

Матрица $A^4 = 0$.

Находим $\sigma_j(\lambda) \quad \lambda = \overline{1, N}$
и $\sigma_j(\lambda) \quad \lambda = \overline{1, N}$.

Определяем матрицу достижимости $A(\Sigma)$: $A(\Sigma) = \sum_{\lambda=1}^N A^\lambda$

(табл. 4.18).

Таблица 4.18

$i \backslash j$	1	2	3	4	5	6	7	8	9	10	σ_i
1					1		1	3	1	1	
2					1		2	3	2	1	
3						1			1	1	
4						1	1	1	1	1	
5							1	2	1	1	
6										1	
7								1	1		
8											
9											
10											
σ_j	0	0	0	0	0	0	1	5	3	4	

Последовательность вычисленных матриц позволяет формально определить все свойства анализируемого информационного потока.

Рассмотрим эти свойства в той же последовательности, в которой они излагались в теории.

1. Определение порядка элементов.

Для этого используется система
$$\left. \begin{array}{l} \sigma_i(\lambda = \pi_j) = 0; \\ \sigma_j(\lambda = \pi_j + 1) = 0. \end{array} \right\}$$

Определяем элементы нулевого порядка, для чего полагаем π_j . $\sigma_j(\lambda = 0) > 0$ для $j = \underline{1}, \underline{2}, \underline{3}, \underline{4}, \underline{5}, \underline{6}, \underline{7}, \underline{8}, \underline{9}, \underline{10}$, так как любая матрица в степени 0 дает 1.

$\sigma_j(\lambda = 1) = 0$ для $j = \underline{1}, \underline{2}, \underline{3}, \underline{4}$ — это следует из матрицы A . Совместно указанное соотношение выполняется для $j = 1, 2, 3, 4$.

Следовательно элементы X_1, X_2, X_3, X_4 есть элементы нулевого порядка.

Определяем элементы 1-го порядка, полагая $\pi_j = 1$.

$$\left. \begin{array}{l} \sigma_i(\lambda = 1) > 0 \text{ для } j = \underline{5}, \underline{6}, \underline{7}, \underline{8}, \underline{9}, \underline{10} \text{ (из матрицы } A^1); \\ \sigma_j(\lambda = 2) = 0 \text{ для } j = 1, 2, 3, 4, \underline{5}, \underline{6} \text{ (из матрицы } A^2). \end{array} \right\}$$

Совместно заданные условия выполняются для $j = 5, 6$. Следовательно, X_5, X_6 — элементы первого порядка.

Определяем элементы 2-го порядка, полагая $\pi_j = 2$.

$$\left. \begin{array}{l} \sigma_i(\lambda = 2) > 0 \text{ для } j = \underline{7}, \underline{8}, \underline{9}, \underline{10}; \\ \sigma_j(\lambda = 3) = 0 \text{ для } j = 1, 2, 3, 4, 5, 6, \underline{7}, \underline{10}. \end{array} \right\}$$

Совместно заданные условия выполняются для $j = 7, 10$. Следовательно, X_7, X_{10} — элементы второго порядка.

Полагаем $\pi_j = 3$.

$$\left. \begin{array}{l} \sigma_i(\lambda = 3) > 0 \text{ для } j = \underline{8}, \underline{9}; \\ \sigma_j(\lambda = 4) = 0 \text{ для } j = 1, 2, 3, 4, 5, 6, 7, \underline{8}, \underline{9}, \underline{10}. \end{array} \right\}$$

Совместно заданные условия выполняются для $j = 8, 9$. Следовательно, X_8, X_9 — элементы третьего порядка.

2. Определение «тактности» системы.

$$N = \max_j \pi_j = 3.$$

Данная схема является трехтактной.

3. *Отсутствие ненулевых элементов на главной диагонали любой из матриц свидетельствует о том, что в анализируемом документообороте контуров нет.*

4. *Определение входных элементов потока.*

$\sigma_j(\lambda = 1) = 0$ для $j = 1, 2, 3, 4$. Следовательно, элементы X_1, X_2, X_3, X_4 — входные элементы.

Например, $\sigma_6(\lambda = 1) = 2$. Это означает, что в X_6 входят два элемента, и т. д.

5. *Определение выходных элементов потока.*

$\sigma_i(\lambda = 1) = 0$ для $i = 8, 9, 10$. Следовательно, элементы X_8, X_9, X_{10} — выходные элементы.

Например, $\sigma_5(\lambda = 1) = 3$. Это означает, что X_5 используется для формирования 3-х других элементов.

6. *Определение висящих вершин.*

Ситуация $\sigma_j(\lambda = 1) = \sigma_i(\lambda = 1) = 0$ отсутствует, т.е. висящих вершин нет.

7. *Определение числа путей длиной λ .*

Например, $a_{28}(\lambda = 2) = 2$. Это означает, что от X_2 к X_8 имеются два различных пути длиной два.

8. *Определение всевозможных путей между двумя элементами.*

Например, $a_{18}(\Sigma) = 3$. Это означает, что от X_1 к X_8 имеются три различных пути длиной 1.

9. *Определение всех элементов, участвующих в формировании данного.*

Например, отличные от нуля элементы 8-го столбца матрицы $A(\Sigma)$ указывают все элементы потока, участвующие в формировании X_8 , т.е. $X_1, X_2, X_3, X_4, X_5, X_7$, причем, например, X_5 — дважды; а, например, ненулевые элементы 5-й строки матрицы $A(\Sigma)$ указывают все элементы, при формировании которых используется X_5 , т.е. X_7, X_8, X_9, X_{10} , причем для формирования X_8 элемент X_5 используется дважды.

10. *Определение номера такта, после которого данный элемент может быть «погашен» в памяти системы.*

Например, X_1 уже не используется после $\tau_1 = 3$, так как, судя по A , для формирования X_1 используется X_5 с $\tau_5 = 1$ и X_8 с $\tau_8 = 3$. Максимум равен 3.

11. *Определение числа тактов хранения.*

Например, для X_1 $t_1 = \tau_1 - \pi_1 = 3 - 0 = 3$.

12. *Рассмотрим столбцы матрицы $A(B)$, соответствующие выходным элементам.*

Например, столбец, соответствующий X_8 . Как указывалось, эта матрица задает число всех связей между элементами. В фор-

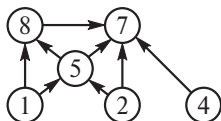


Рис. 4.28. Вид фрагмента графа, связанного с формированием X_8 .

мировании X_8 участвуют элементы X_1, X_2, X_4, X_5 и X_7 , причем X_1 и X_2 — трижды, а X_5 — дважды.

Наличие большого числа ненулевых и неединичных элементов в столбце $j = 8$ свидетельствует о необходимости проведения содержательного анализа фрагмента общей схемы потока, связанной с формированием X_8 (рис. 4.28). Быть может, удастся упростить фрагмент за счет исключения излишних связей или промежуточных элементов.

4.6. Структурно-топологические характеристики систем и их применение

При проведении анализа системы целесообразно оценить количественно качество структуры системы и ее элементов с позиций общесистемного подхода. Рассмотрим основные структурно-топологические характеристики. Сначала выделим основные виды структур с точки зрения топологии внутренних связей.

Виды топологических структур. Виды топологических структур рассмотрим на примере пяти элементов.

Связность структуры. Данная характеристика позволяет выделить наличие обрывов, висячие вершины и т. д. Для неориенти-

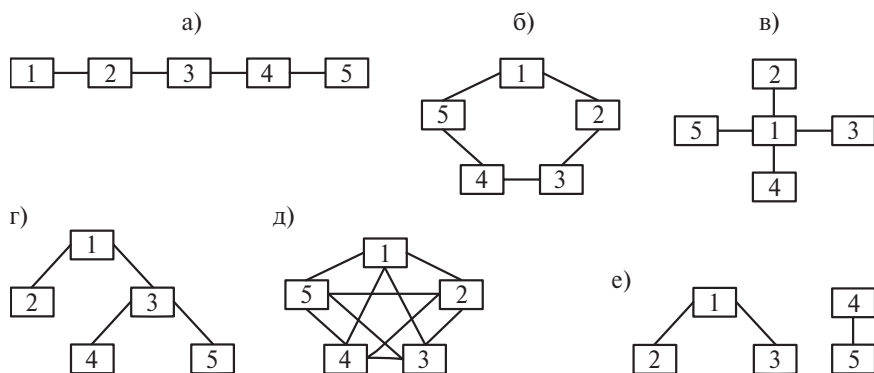


Рис. 4.29. Виды топологических структур: а) последовательная структура; б) кольцевая структура; в) радиальная структура; г) древовидная структура; д) структура типа полный граф; е) несвязная структура

рованных графов связность всех элементов в структуре соответствует выполнению условия:

$$0,5 = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \geq n - 1, \quad i \neq j. \quad (4.6)$$

В соотношении (4.6) a_{ij} — элемент матрицы смежности. Правая часть (4.6) определяет необходимое минимальное число связей в структуре неориентированного графа, содержащего n вершин. Коэффициент 0,5 берется в силу того, что одна и та же связь a_i и a_j учитывается дважды.

Соотношение (4.6) можно переписать в виде

$$0,5 = \sum_{i=1}^n \sum_{j=1}^n a_{ij} = m,$$

т.е. это не что иное, как число ребер в неориентированном графе.

Определим m как функцию от n для основных типов структур в общем виде.

Последовательная структура. Очевидно $m = n - 1$.

Кольцевая структура. Очевидно $m = n$.

Радиальная структура. Очевидно $m = n - 1$.

Древовидная структура. Трудно сказать в общем виде, так как параметр определяется видом дерева.

Структура полный граф. Для первого элемента число связей $n - 1$, так как он связан со всеми элементами, кроме себя. У второго элемента $n - 2$ связей, т.к. нет связи с самим собой, и связь с первым мы уже учли. Аналогично у третьего — $n - 3$ и т.д. У последнего — 0, так как все связи учтены. Итак, имеем $m = (n - 1) + (n - 2) + (n - 3) + \dots + 0 = 0,5n(n - 1)$.

Для несвязной структуры эта характеристика не имеет смысла.

Структурная избыточность. Это структурный параметр, отражающий превышение общего числа связей над необходимым минимальным числом связей. Разделим в соотношении (4.6) все члены на $n - 1$ и разность обозначим через R . Тогда будем иметь

$$R = 0,5 \sum_{i=1}^n \sum_{j=1}^n a_{ij} \frac{1}{n - 1} - 1; \quad (4.7)$$

$$R = \frac{m}{n - 1}, \quad (4.8)$$

где R — структурная избыточность.

Найдем структурную избыточность для рассмотренных типовых структур в общем виде, учитывая ранее найденную зависимость $m = m(n)$.

1. Последовательная структура: $m = n - 1$; $R = \frac{n-1}{n-1} - 1 = 0$.
2. Кольцевая структура: $m = n$; $R = \frac{n}{n-1} - 1 = \frac{1}{n-1}$.
3. Радиальная структура: $m = n - 1$; $R = \frac{n-1}{n-1} - 1 = 0$.
4. Структура полный граф: $m = \frac{n(n-1)}{2}$; $R = \frac{n(n-1)}{2(n-1)} - 1 = \frac{n-2}{2}$.

Данная структурная характеристика используется для косвенной оценки экономичности и надежности исследуемой системы. Для систем с избыточностью — $R > 0$, для систем с минимальной избыточностью — $R = 0$, для несвязных систем — $R < 0$.

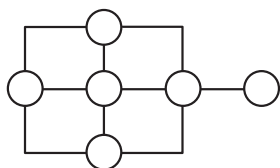


Рис. 4.30. Вид графа

Очевидно, что чем больше R , тем потенциально более надежна система, но структурная избыточность может быть достигнута за счет увеличения связности только одних вершин (см. рис. 4.30).

Поэтому вводят параметр ϵ^2 , учитывающий неравномерность распределения связей или их несимметричность.

Вспомним понятие «степень вершины» ρ_i , т. е. число ребер, инцидентных вершине i . Было показано, что

$$0,5 \left(\sum_{i=1}^n \rho_i \right) = m.$$

Если связи распределены равномерно, то, очевидно, все c_i одинаковы. Тогда

$$\sum_{i=1}^n \rho_i = n\bar{\rho} \quad \text{и} \quad \bar{\rho} = \frac{2m}{n},$$

где $\bar{\rho}$ — средняя степень вершины.

Теперь можно определить квадратичные отклонения распределения степеней вершин от равномерного.

Среднеквадратичное отклонение равно

$$\epsilon^2 = \sum_{i=1}^n (\rho_i - \bar{\rho})^2 = \sum_{i=1}^n \rho_i^2 - 2 \sum_{i=1}^n \rho_i \bar{\rho} + \sum_{i=1}^n (\bar{\rho})^2; \quad 2 \sum_{i=1}^n \rho_i \bar{\rho} = 2\bar{\rho} \sum_{i=1}^n \rho_i.$$

$$\begin{aligned} \text{Так как } \sum_{i=1}^n \rho_i &= 2m \text{ и } \bar{\rho} = 2\frac{m}{n}, \text{ то } 2 \sum_{i=1}^n \rho_i \bar{\rho} = 2 \cdot 2\bar{\rho}\frac{m}{n} \cdot n \cdot 2\frac{m}{n} = \\ &= 8\frac{m^2}{n}; \quad 2 \sum_{i=1}^n \bar{\rho}^2 = 4 \cdot n\frac{m^2}{n^2} = \frac{4m^2}{n}. \end{aligned}$$

И, окончательно,

$$\begin{aligned} \varepsilon^2 &= \sum_{i=1}^n \rho_i^2 - 8\frac{m^2}{n} + 4\frac{m^2}{n}; \\ \varepsilon^2 &= \sum_{i=1}^n \rho_i^2 - \frac{4m^2}{n}. \end{aligned} \quad (4.9)$$

Показатель ε^2 характеризует недоиспользованные возможности заданной структуры, имеющей m ребер и n вершин.

Рассмотрим структурную неравномерность для типовых схем.

1. Последовательная структура:

$$m = n - 1. \quad \text{Тогда } \frac{4m^2}{n} = \frac{4(n-1)^2}{n}.$$

$$\sum_{i=1}^n \rho_i^2 = 1^2 + \underbrace{2^2 + 2^2 + \dots + 2^2}_{n-2} + 1^2 = 2 + 2^2(n-2) = 4n - 6.$$

$$\varepsilon^2 = 4n - 6 - \frac{4(n-2)^2}{n} = \frac{2(n-2)}{n}.$$

2. Кольцевая структура:

$$m = n, \quad \sum_{i=1}^n \rho_i^2 = \underbrace{2^2 + 2^2 + \dots + 2^2}_n = 2^2 n.$$

$$\varepsilon^2 = 2^2 n - \frac{4n^2}{n} = 0.$$

3. Радиальная структура:

$$m = n - 1, \quad \sum_{i=1}^n \rho_i^2 = (n-1)^2 + \underbrace{1^2 + 1^2 + \dots + 1^2}_{n-1} = n(n-1).$$

$$\varepsilon^2 = n(n-1) - \frac{4(n-1)^2}{n} = \frac{(n-1)(n-2)}{n}.$$

4. Структура полный граф:

$$m = \frac{n(n-1)}{2}, \quad \frac{4m^2}{n} = \frac{4}{n} \cdot \frac{n^2(n-1)^2}{4} = n(n-1)^2.$$

$$\sum_{i=1}^n \rho_i^2 = \underbrace{(n-1)^2 + (n-1)^2 + \dots + (n-1)^2}_n = n(n-1)^2.$$

$$\varepsilon^2 = n(n-1)^2 - n(n-1)^2 = 0.$$

Структурная компактность. Для ее оценки вводится параметр, отображающий близость элементов между собой. *Близость* 2-х элементов i и j определяется через минимальную длину пути для ориентированного графа (цепи — для неориентированного) — d_{ij} . Тогда сумма всех минимальных путей (цепей) между всеми элементами отражает общую структурную близость элементов в анализируемой структуре. Обозначим эту величину через Q и, в соответствии с определением, будем иметь:

$$Q = \sum_{i=1}^n \sum_{j=1}^n d_{ij}, \quad i \neq j. \quad (4.10)$$

Рассмотрим этот параметр для основных структур.

1. Последовательная структура.

Непосредственно из ее изображения имеем:

Для 1-го элемента

$$\sum_{j=1}^n d_{1j} = d_{12} + d_{13} + d_{14} + d_{15} + \dots + d_{1n} = 1 + 2 + 3 + 4 + \dots + n - 1.$$

Для 2-го элемента

$$\sum_{j=1}^n d_{2j} = d_{21} + d_{23} + d_{24} + d_{25} + \dots + d_{2n} = 1 + 1 + 1 + 3 + \dots + n - 2.$$

Для 3-го элемента

$$\sum_{j=1}^n d_{3j} = d_{31} + d_{33} + d_{34} + d_{35} + \dots + d_{3n} = 2 + 1 + 1 + 2 + \dots + n - 3.$$

Для n -го элемента

$$\sum_{j=1}^n d_{nj} = n - 1 + \dots + 4 + 3 + 2 + 1.$$

Перепишем это иначе:

$$Q_1 \quad \begin{array}{c} 1 + \\ 2 + 1 + \end{array} \left[\begin{array}{c} 1 + 2 + 3 + 4 + \dots + (n-1) \\ 1 + 2 + 3 + 4 + \dots + (n-2) \\ 1 + 2 + 3 + 4 + \dots + (n-3) \end{array} \right]_{n-1}$$

$$\underbrace{\hspace{15em}}_{n-1}$$

$$\begin{array}{c} (n-1) + \dots + 4 + 3 + 2 + 1 \\ (n-2) + \dots + 4 + 3 + 2 + 1 \\ (n-3) + \dots + 4 + 3 + 2 + 1 \end{array} \left| \begin{array}{c} + 1 \\ + 1 + 2 \end{array} \right. \quad Q_2$$

Теперь очевидно, что $Q_1 = Q_2$ и $Q = 2Q_1$.
Рассмотрим Q_1 . Сумма натуральных чисел по столбцам равна

$$Q_1 = 1(n-1) + 2(n-1-1) + 3(n-1-1-1) + \dots + (n-1).$$

$$Q_1 = 1(n-1) + 2(n-2) + 3(n-3) + \dots + (n-1).$$

$$Q_1 = \sum_{i=1}^{n-1} i(n-i).$$

И, окончательно,

$$Q = 2 \sum_{i=1}^{n-1} i(n-i). \tag{4.11}$$

Раскроем последнее выражение

$$Q = 2n \sum_{i=1}^{n-1} i - 2 \sum_{i=1}^{n-1} i^2.$$

Известна справочная формула

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}.$$

Откуда

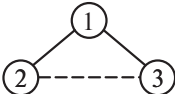
$$2 \sum_{i=1}^n i^2 = \frac{n(n-1)(2n-1)}{3}.$$

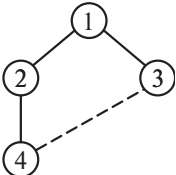
И, окончательно,

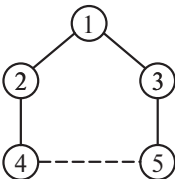
$$Q = \frac{1}{3} n(n^2 - 1). \tag{4.12}$$

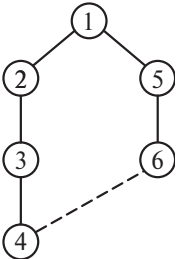
2. Кольцевая структура. Имеем для 1-го элемента ($i=1$)

$n = 2$  $d_{12} = 1;$

$n = 3$  $\sum_{j=1}^3 d_{1j} = 1 + 1;$

$n = 4$  $\sum_{j=1}^4 d_{1j} = 1 + 2 + 1;$

$n = 5$  $\sum_{j=1}^5 d_{1j} = 1 + 2 + 1 + 2;$

$n = 6$  $\sum_{j=1}^6 d_{1j} = 1 + 2 + 3 + 1 + 2.$

Теперь очевидно, что для n – нечетного

$$\sum_{j=1}^n d_{1j} = 1 + 2 + 3 + \dots + \frac{n-1}{2} + 1 + 2 + 3 + \dots + \frac{n-1}{2};$$

$$\sum_{j=1}^n d_{1j} = 2 \cdot \left(1 + \frac{n-1}{2}\right) \left(\frac{n-1}{4}\right) = \frac{n^2-1}{4}.$$

Мы нашли сумму всех минимальных путей (цепей) для 1-го элемента (в случае нечетного числа элементов). И это справедливо для всех n элементов. Окончательно (для n – нечетного)

$$Q = \frac{n(n^2+1)}{4}. \quad (4.13)$$

Для n – четного имеем

$$\sum_{j=1}^n d_{1j} = \left[1 + 2 + 3 + \dots + \frac{n}{2} \right] + \left[1 + 2 + 3 + \dots + \frac{n}{2} \right] - \frac{n}{2} = \frac{n^2}{4}.$$

И, окончательно, для n – четного

$$Q = \frac{n^3}{4}. \quad (4.14)$$

3. Радиальная структура.

Непосредственно из ее изображения (см. п. 4.6.1) имеем для центрального элемента

$$\sum_{j=1}^n = 1(n-1).$$

Для второго и, в силу симметричности, любого из оставшихся элементов, имеем

$$\sum_{j=1}^n d_{2j} = \sum_{j=1}^n d_{3j} = \dots = 1 + 2(n-2) = 2n-3.$$

Первый член в последнем выражении 1 – это путь до 1-го элемента, второй 2 – для любого другого, а их $(n-2)$, так как нет связи с самим собой, а связь с первым уже учтена. И таких связей, очевидно, $(n-1)$. Тогда

$$\begin{aligned} Q &= (n-1) + (2n-3)(n-1); \\ Q &= 2(n-1)^2. \end{aligned} \quad (4.15)$$

4. Структура полный граф.

У каждого элемента есть связи со всеми, кроме себя. Это – $(n-1)$, а всего вершин n . Итак, имеем

$$Q = n(n-1).$$

Для оценки структурной компактности часто используется относительный показатель

$$Q_{\text{отн}} = \frac{Q}{Q_{\text{min}}} - 1, \quad (4.16)$$

где $Q_{\text{min}} = n(n-1)$ – минимальное значение компактности для структуры типа полный граф.

Структурную компактность можно характеризовать и другим параметром – *диаметром структуры*:

$$d = \max_{i,j} d_{ij}.$$

У полного графа указанная величина равна единице, поэтому он и обладает максимальной компактностью.

Учитывая преобладающий информационный характер связей в АСУ, можно сказать, что как величина $Q_{\text{отн}}$, так и d интегрально оценивают инерционность процессов в системе, а при равных значениях ε^2 и R (равномерность и структурная избыточность) их возрастание отражает увеличение числа разделяющих связей в структуре, характеризуя тем самым снижение общей надежности.

Степень централизации структуры.

1. Наименее «централизованными» и наиболее «размытыми» с этих позиций являются структуры кольцевая и полный граф, в которых элементы нагружены абсолютно одинаково. Для этих двух структур структурная компактность

$$Q = \sum_{i=1}^n \sum_{j=1}^n d_{ij} = n \sum_{i=1}^n d_{ij},$$

где $n \sum_{i=1}^n d_{ij}$ – сумма расстояний от i -того элемента до всех остальных.

Рассмотрим отношение $\frac{Q}{n}$. Для кольцевой структуры

$$\sum_{j=1}^n d_{ij}$$

$\sum_{i=1}^n d_{ij}$ равны между собой. Для полного графа эти величины хотя

и другие (по сравнению с предыдущим случаем), но тоже равны между собой.

Тогда для этих структур, введя обозначение, получим

$$2z_i = \frac{Q}{n} = \frac{\sum_{i=1}^n d_{ij}}{\sum_{j=1}^n d_{ij}} = n.$$

Для любой другой структуры эта величина будет больше.

Обозначим

$$2z_{\max} = \frac{Q}{\left[\sum_{j=1}^n d_{ij} \right]_{\min}}.$$

Для того чтобы эта величина для кольцевой структуры и полного графа (один крайний случай) начиналась с нуля, вычтем из нее n .

$$\alpha' = 2z_{\max} - n = Q \left[\sum_{j=1}^n d_{ij} \right]_{\min}^{-1} - n.$$

Проведем нормализацию. Для этого рассмотрим другой крайний случай – радиальную структуру. Было получено

$$Q = 2(n-1)^2, \quad \left[\sum_{j=1}^n d_{ij} \right]_{\min} = n-1.$$

Тогда для радиальной структуры

$$\alpha' = 2(n-1)^2(n-1)^{-1} - n = n-2.$$

Это максимальное значение α' . У всех других структур α' будет меньше. Потребуем, чтобы эта величина не превышала единицу. Тогда

$$\alpha = \alpha' (n-2)^{-1} = (2z_{\max} - n)(n-2)^{-1}.$$

2. Но, с другой стороны, чем более компактна структура, тем легче центру управлять ею. Поэтому введем новый параметр β' , выражаемый через z_{\max} и возрастающий с ростом компактности. В соответствии с определением

$$\beta' = \left[\frac{\sum_{i=1}^n d_{ij}}{Q} \right]_{\max} = (2z_{\max})^{-1}.$$

Для нашего предельного случая (радиальная структура), как указывалось,

$$Q = 2(n-1)^2, \quad \left[\sum_{j=1}^n d_{ij} \right]_{\min} = n-1.$$

Тогда

$$\beta' = (n - 1)[2(n - 1)^2]^{-1} = [2(n - 1)]^{-1}.$$

Потребуем также, чтобы в этом предельном случае эта величина равнялась бы единице:

$$\beta = \beta' 2(n - 1).$$

Тогда

$$\beta = 2(n - 1)(2z_{\max})^{-1} = (n - 1)(z_{\max})^{-1}.$$

Теперь введем понятие *индекса центральности*, определив его как

$$\gamma = \alpha\beta = \frac{(n - 1)(2z_{\max} - n)}{(n - 2)z_{\max}}. \quad (4.17)$$

Для структур с максимальной степенью централизации (радиальная) $\gamma = 1$, для структур с равномерным распределением связей (кольцевая и полный граф) $\gamma = 0$.

Сравнительный анализ топологических структур. Результаты вычисления топологических характеристик для типовых структур (для случая 5-ти элементов) сведены в таблицу 4.16. Параметры этой таблицы для всех структур, кроме древовидной, вычислялись по приведенным выше формулам (4.8—4.17). Для древовидной структуры расчеты произведем непосредственно исходя из определений параметров.

Таблица 4.16

Структура \ Параметр	R	ϵ^2	Q	$Q_{\text{отн}}$	d	γ
Последовательная	0	1,2	40	1,0	4	0,7
Кольцевая	0,25	0	30	0,5	2	0
Радиальная	0	7,2	32	0,6	2	1,0
Древовидная	0	3,2	36	0,8	3	0,81
Полный граф	2,5	0	20	0	1	0
Несвязанная	-0,25	—	—	—	—	—

Структурная избыточность R

$$R = \frac{m}{n - 1} - 1 = \frac{4}{5 - 1} - 1 = 0.$$

Среднеквадратичное отклонение

$$\varepsilon^2 = \sum_{j=1}^n \rho_j^2 - \frac{4m^2}{n} = 2^2 + 1^1 + 3^2 + 1^2 + 1^2 - \frac{4 \cdot 4^2}{5} = 3,2.$$

Структурная компактность

$$Q = \sum_{i=1}^n \sum_{j=1}^n d_{ij}, \quad i \neq j.$$

$$\sum_{j=1}^5 d_{1j} = d_{12} + d_{13} + d_{14} + d_{15} = 1 + 1 + 2 + 2 = 6,$$

$$\sum_{j=1}^5 d_{2j} = 1 + 2 + 3 + 3 = 9,$$

$$\sum_{j=1}^5 d_{3j} = 1 + 2 + 1 + 1 = 5,$$

$$\sum_{j=1}^5 d_{4j} = 2 + 3 + 1 + 2 = 8,$$

$$\sum_{j=1}^5 d_{5j} = 2 + 3 + 1 + 2 = 8.$$

Структурная компактность

$$Q = 6 + 9 + 5 + 8 + 8 = 36.$$

Относительная структурная компактность

$$Q_{\text{отн}} = \frac{Q}{Q_{\text{min}}} - 1,$$

где

$$Q_{\text{min}} = n \cdot (n - 1) = 5 \cdot 4 = 20.$$

$$Q_{\text{отн}} = \frac{36}{20} - 1 = 0,8.$$

Степень централизации

$$\gamma = \frac{(n-1) \cdot (2z_{\max} - n)}{(n-2)z_{\max}}, \quad \text{где} \quad 2z_{\max} = \frac{Q}{\left[\sum_{j=1}^n d_{ij} \right]_{\min}}.$$

$$2z_{\max} = \frac{36}{5} = 7,2 \quad \text{и} \quad \gamma = \frac{(5-1)(7,2-5)}{(5-2) \cdot \frac{7,2}{2}} = 0,81.$$

Анализ табл. 4.16 показывает следующее.

1. Для несвязных структур структурная избыточность $R < 0$, для структур без избыточности (последовательная, радиальная, древовидная) $R = 0$; для структур с избыточностью по связям (кольцевая, полный граф) $R > 0$.

2. Структуры (последовательная, радиальная, древовидная) с $R = 0$ различаются по показателю ε^2 , наибольшую неравномерность связей имеет радиальная структура.

3. Наибольшую близость элементов (показатель $Q_{\text{отн}}$) имеет структура типа полный граф, наименьшую — последовательная.

4. Радиальная и древовидная структуры, имеющие одинаковые или близкие значения R , $Q_{\text{отн}}$, d , значительно отличаются по показателям ε^2 и γ , что соответствует физическому смыслу, ибо отход от полной централизации в структуре ведет к большей равномерности распределения связей по элементам.

Рассмотренные выше структурные характеристики были получены только на основе информации о составе элементов и их связях. Дальнейшее развитие методологии структурных параметров для решения задач структурного анализа может быть основано на учете неструктурной информации за счет введения числовых функций на графах. Это позволяет, наряду с составом элементов и направленностью их взаимодействия, учитывать при решении задач другие стороны их функционирования (временные, надежностные, стоимостные и т. п.).

Рассмотрим далее третий уровень формализации задач структурного анализа, когда учитывается состав сигналов взаимосвязи элементов и их вид.

Напомним, что на первом уровне учитывалось лишь наличие связи между элементами, на втором — наличие и направление этой связи.

Пример расчета структурно-топологических характеристик. Пусть необходимо для структуры, представленной на рис. 4.31, вычислить все ее структурно-топологические характеристики, по полученным результатам охарактеризовать структуру.

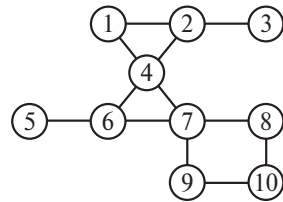


Рис. 4.31. Вид исходного графа

1. Связность структуры.

Для неориентированных графов связность всех элементов в структуре соответствует выполнению следующего условия:

$$0,5 \sum_{i=1}^n \sum_{j=1}^n a_{ij} \geq n - 1, \quad i \neq j,$$

где a_{ij} — элемент матрицы смежности.

В данном случае имеем $0,5 \cdot 24 > 9$. Следовательно, граф является связным.

2. Структурная избыточность R .

$$R = 0,5 \sum_{i=1}^n \sum_{j=1}^n a_{ij} \frac{1}{n-1} - 1 \quad \text{или} \quad R = \frac{m}{n-1} - 1,$$

где m — число ребер, n — число вершин.

В данной структуре $n = 10$, $m = 12$ и

$$R = \frac{12}{10-1} - 1 = \frac{1}{3} > 0.$$

Поскольку $R > 0$, то в данной системе присутствует структурная избыточность.

3. Среднеквадратичное отклонение ε^2 .

Справедливо следующее соотношение:

$$\varepsilon^2 = \sum_{i=1}^n \rho_i^2 - \frac{4}{n} m^2, \quad \rho_i^2 \text{ — квадрат степени } i\text{-той вершины.}$$

Для данной системы:

$$\varepsilon^2 = 2^2 + 3^2 + 1 + 4^2 + 1 + 3^2 + 4^2 + 2^2 + 2^2 + 2^2 - \frac{4 \cdot 12^2}{10} = 10,4.$$

4. Структурная компактность.
 Воспользуемся соотношением

$$Q = \sum_{i=1}^n \sum_{j=1}^n d_{ij}, \quad i \neq j,$$

где d_{ij} — минимальная длина цепи между i -той и j -той вершинами.

$$\sum_{j=1}^{10} d_{1j} = 1 + 2 + 1 + 3 + 2 + 2 + 3 + 3 + 4 = 21, \quad (j \neq 1);$$

$$\sum_{j=1}^{10} d_{2j} = 1 + 1 + 1 + 3 + 2 + 2 + 3 + 3 + 4 = 20, \quad (j \neq 2);$$

$$\sum_{j=1}^{10} d_{3j} = 2 + 1 + 2 + 4 + 3 + 3 + 4 + 4 + 5 = 28, \quad (j \neq 3);$$

$$\sum_{j=1}^{10} d_{4j} = 1 + 1 + 2 + 2 + 1 + 1 + 2 + 2 + 3 = 15, \quad (j \neq 4);$$

$$\sum_{j=1}^{10} d_{5j} = 3 + 3 + 4 + 2 + 1 + 2 + 3 + 3 + 4 = 25, \quad (j \neq 5);$$

$$\sum_{j=1}^{10} d_{6j} = 2 + 2 + 3 + 1 + 1 + 1 + 2 + 2 + 3 = 17, \quad (j \neq 6);$$

$$\sum_{j=1}^{10} d_{7j} = 2 + 2 + 3 + 1 + 2 + 1 + 1 + 1 + 2 = 15, \quad (j \neq 7);$$

$$\sum_{j=1}^{10} d_{8j} = 3 + 3 + 4 + 2 + 3 + 2 + 1 + 2 + 1 = 21, \quad (j \neq 8);$$

$$\sum_{j=1}^{10} d_{9j} = 3 + 3 + 4 + 2 + 3 + 2 + 1 + 2 + 1 = 21, \quad (j \neq 9);$$

$$\sum_{j=1}^{10} d_{10j} = 4 + 4 + 5 + 3 + 4 + 3 + 2 + 1 + 1 = 27, \quad (j \neq 10).$$

Следовательно, $Q = 210$.

Относительный показатель $Q_{\text{отн}} = \frac{Q}{Q_{\text{min}}} - 1$, где $Q_{\text{min}} = n(n-1)$ — структурная компактность для полного графа.

$$Q_{\text{отн}} = \frac{210}{10(10-1)} - 1 = 1,3.$$

5. Степень централизации в структуре γ .
Воспользуемся соотношением (4.17)

$$\gamma = \frac{(n-1)(2z_{\text{max}} - n)}{(n-2)z_{\text{max}}}.$$

Подставляя числовые значения, получим:

$$z_{\text{max}} = \frac{210}{2 \cdot 15} = 7; \quad \gamma = \frac{9}{8} \cdot \frac{2 \cdot 7 - 10}{7} \approx 0,643.$$

Мы вычислили основные структурно-топологические характеристики заданной структуры. Эти характеристики имеют следующие числовые значения:

- структурная избыточность $R = 0,33$;
- среднеквадратичное отклонение $\varepsilon^2 = 10,4$;
- структурная компактность $Q = 210$, $Q_{\text{отн}} = 1,3$;
- степень централизации в структуре $\gamma = 0,643$.

Структура является связной. Для того, чтобы судить о степени близости анализируемой структуры к типовым структурам по различным параметрам, необходимо составить таблицу, аналогичную таблице 4.16, и заполнить ее данными для случая $n = 10$. Для древовидной структуры необходимо учесть также, что $m = 12$. Сравнение этих данных с данными анализируемой структуры позволит получить ответы на поставленные вопросы.

Итак, заполняем табл. 4.17 для случая $n = 10$, $m = 12$. В строку «Анализируемая структура» подставляем только что рассчитанные данные.

Структурная избыточность

На основании выражений (4.6—4.17) произведем требуемые вычисления. Напомним, что

$$R = \frac{m}{n-1} - 1.$$

1. Последовательная структура:

$$m = n - 1; \quad R = \frac{n-1}{n-1} - 1 = 0.$$

2. Кольцевая структура:

$$m = n; \quad R = \frac{n}{n-1} - 1 = \frac{1}{n-1} = 0,1.$$

3. Радиальная структура:

$$m = n - 1; \quad R = \frac{n-1}{n-1} - 1 = 0.$$

4. Древоподобная структура (в данном конкретном случае):

$$R = \frac{m}{n-1} - 1 = \frac{12}{10-1} - 1 = 0,33.$$

5. Структура полный граф:

$$m = \frac{n(n-1)}{2}; \quad R = \frac{n(n-1)}{2(n-1)} - 1 = \frac{n-2}{2} = 4.$$

Среднеквадратичное отклонение

$$\varepsilon^2 = \sum_{i=1}^{10} \rho_i^2 - \frac{4m^2}{n}.$$

1. Последовательная структура:

$$\varepsilon^2 = \frac{2(n-2)}{n} = 1,6.$$

2. Кольцевая структура:

$$\varepsilon^2 = 0.$$

3. Радиальная структура:

$$\varepsilon^2 = \frac{(n-1)(n-2)^2}{2} = 57,6.$$

4. Древоподобная структура (в данном конкретном случае):

$$\begin{aligned} \varepsilon^2 &= 2^2 + 3^2 + 3^2 + 3^2 + 2^2 + 1^1 + 3^2 + 1^2 + \\ &+ 1^2 + 1^2 + 1^2 + 1^2 - \frac{4 \cdot 9^2}{10} = 7,6. \end{aligned}$$

5. Структура полный граф:

$$\varepsilon^2 = 0.$$

Структурная компактность

$$Q = \sum_{i=1}^n \sum_{j=1}^n d_{ij}, \quad i \neq j.$$

1. Последовательная структура:

$$Q = \frac{1}{3} n(n^2 - 1) = 330.$$

2. Кольцевая структура:

$$Q = \frac{n^3}{4} = 250.$$

3. Радиальная структура:

$$Q = 2(n - 1)^2 = 162.$$

4. Древоподобная структура (в данном конкретном случае):

$$\sum_{i=1}^{10} d_{1j} = 1 + 1 + 2 + 2 + 2 + 2 + 3 + 3 + 4 = 20,$$

$$Q = 20 + 17 + 23 + 21 + 25 + 31 + 31 + 29 + 29 + 31 = 257.$$

5. Структура полный граф:

$$Q = n(n - 1) = 90.$$

Относительная структурная компактность

$$Q_{\text{отн}} = \frac{Q}{Q_{\text{min}}} - 1.$$

1. Последовательная структура:

$$Q_{\text{отн}} = \frac{330}{90} - 1 = 2,67.$$

2. Кольцевая структура:

$$Q_{\text{отн}} = \frac{250}{90} - 1 = 1,78.$$

3. Радиальная структура:

$$Q_{\text{отн}} = \frac{162}{90} - 1 = 0,8.$$

4. Древоподобная структура (в данном конкретном случае):

$$Q_{\text{отн}} = \frac{257}{90} - 1 = 1,85.$$

5. Структура полный граф:

$$Q_{\text{отн}} = \frac{90}{90} - 1 = 0.$$

Степень централизации

$$\gamma = \frac{(n-1)(2z_{\max} - n)}{(n-2)z_{\max}}, \quad \text{где } 2z_{\max} = \frac{Q}{\left[\sum_{j=1}^n d_{ij} \right]}.$$

1. Последовательная структура.

Можно показать, что для N -ного элемента (т. е. элемента, находящегося на месте N) справедливо соотношение

$$\sum_{j=1}^n d_{ij} = \frac{i(i+1)}{2} + \frac{(n-i)(n-i-1)}{2}.$$

Для $n = 10$ и $N = 5$ и $N = 6$:

$$\begin{aligned} \left[\sum_{j=1}^{10} d_{5j} \right]_{\min} &= \left[\sum_{j=1}^{10} d_{6j} \right]_{\min} = \frac{4(4+1)}{2} + \frac{(10-4)(10-4-1)}{2} = \\ &= \frac{5(5+1)}{2} + \frac{(10-5)(10-5-1)}{2} = 25; \end{aligned}$$

$$2z_{\max} = \frac{330}{25} = 13,2;$$

$$\gamma = \frac{(10-1)2(13,2-10)}{(10-2) \cdot 13,2} = 0,18.$$

2. Кольцевая структура:

$$2z_{\max} = n \quad \text{и} \quad \gamma = 0.$$

3. Радиальная структура:

$$2z_{\max} = \frac{2(n-1)^2}{n-1} = 2(n-1). \quad \gamma = \frac{(n-1)(2(n-1)) - n}{2(n-2)(n-1)} = 1.$$

4. Древоподобная структура (в данном конкретном случае):

$$2z_{\max} = \frac{257}{17} = 15,12. \quad \gamma = \frac{(10-1)(15,12-10)2}{(10-2)15,12} = 0,76.$$

5. Структура полный граф:

$$2z_{\max} = \frac{n(n-1)}{n-1} = n \quad \text{и} \quad \gamma = 0.$$

Итак, имеем:

Таблица 4.17

Структура \ Параметр	R	ϵ^2	Q	$Q_{\text{отн}}$	d	γ
Последовательная	0	1,6	330	2,67	9	0,18
Кольцевая	0,1	0	250	1,78	2	0
Радиальная	0	57,6	162	0,8	2	1,0
Древоподобная	0,33	7,6	257	1,85	5	0,76
Полный граф	4	0	90	0	1	0
Анализируемая структ.	0,33	10,4	210	1,3	5	0,643

Из табл. 4.17 следует, что по своим параметрам анализируемая структура ближе всего к древоподобной структуре.

Если заранее изучить свойства типовых структур, то можно с большой степенью уверенности судить о свойствах анализируемой структуры по близости соответствующих параметров.

4.7. Модели функционирования организационной системы

Основные допущения и определения. Для построения математической модели структурного сопряжения элементов в системе введем ряд допущений:

1. Входной сигнал $x(t)$, поступающий к элементу в момент времени t , будем рассматривать как совокупность элементарных сигналов $x_1(t)$, $x_2(t)$, ..., $x_m(t)$, одновременно возникающих на входах элемента.

2. Выходной сигнал $y(t)$ аналогично может рассматриваться как совокупность элементарных выходных сигналов $y_1(t), y_2(t), \dots, y_n(t)$.

3. Элементарные сигналы передаются в системе независимо друг от друга по элементарным каналам.

4. К входному контакту любого элемента может быть подключен не более чем один элементарный канал.

5. К выходному контакту любого элемента может быть подключено любое количество элементарных каналов.

Итак, пусть система S состоит из N элементов. Для i -го элемента, обозначим его через C_i , в соответствии с введенными допущениями мы имеем:

$\left[X_j^i \right]_1^{m_i} = \left\{ X_1^i, X_2^i, \dots, X_{m_i}^i \right\}$ — множество входных контактов элемента;

$\left[x^i(t) \right]_1^{m_i} \in \left[X_j^i(t) \right]_1^{m_i}$ — множество входных сигналов этого элемента;

$\left[Y_j^i \right]_1^{n_i} = \left\{ Y_1^i, Y_2^i, \dots, Y_{n_i}^i \right\}$ — множество выходных контактов элемента;

$\left[y^i(t) \right]_1^{n_i} \in \left[Y_j^i(t) \right]_1^{n_i}$ — множество выходных сигналов этого элемента.

Графически это выглядит следующим образом (рис. 4.31):

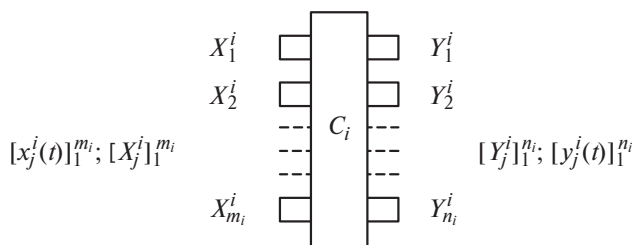


Рис. 4.31. Вид модели элемента

Взаимодействие системы с внешней средой рассматривается как обмен сигналами между внешней средой и элементами системы, причем для всех этих сигналов справедливы все отмеченные выше допущения. В соответствии с этим внешнюю среду можно представить в виде фиктивного элемента системы C_0 , при-

чем сигнал, выдаваемый нашей системой, воспринимается внешней средой как входной сигнал

$$x^0(t) = \{x_1^0(t), x_1^0(t), \dots, x_{m_0}^0(t)\},$$

а сигнал, поступающий на нашу систему из внешней среды, является выходным сигналом элемента C_0 , имитирующего внешнюю среду, и он состоит из элементарных сигналов

$$y^0(t) = \{y_1^0(t), y_1^0(t), \dots, y_{n_0}^0(t)\}.$$

Он воспринимается одним или несколькими элементами нашей системы.

Одноуровневая схема сопряжения. Введем некоторый оператор R так, чтобы

$$Y_l^k = R(X_j^i).$$

В этом соотношении i и k – номера элементов, j и l – номера контактов. Причем

$$X_j^i \in \bigcup_{i=1}^N [X_j^i]_1^{m_i} \text{ — есть один из множества входных контактов.}$$

Совершенно аналогично:

$$Y_j^i \in \bigcup_{i=1}^N [Y_j^i]_1^{n_i} \text{ — есть один из множества выходных контактов.}$$

В последних записях

m_i – число входных контактов i -того элемента и соответственно число входных сигналов,

n_i – число выходных контактов i -того элемента и соответственно число выходных сигналов.

Оператор R называют оператором сопряжения. Он ставит в соответствие входному контакту X_j^i выходной контакт Y_l^k , связанный с ним элементарным каналом. Причем, если к контакту не подключен никакой элементарный канал, то оператор R считается неопределенным на этом X_j^i .

Обычно оператор R задается в виде таблицы, в которой на пересечении строк с номерами элементов i и столбцов с номерами контактов j располагается пара чисел (k, l) , указывающая номер элемента k и номер контакта l , с которым соединен контакт X_j^i .

Рассмотрим **пример**. Пусть структура некоторой системы отображена на рис. 4.32. Необходимо разработать ее оператор сопряжения.

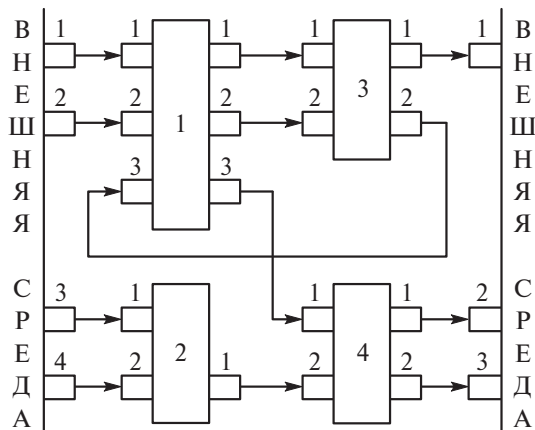


Рис. 4.32. Вид структуры системы

Оператор сопряжения составляется только по входу, именно поэтому возникло требование подключения к входу только одного элементарного канала.

Для простоты составления оператора сопряжения представим отдельно входы всех элементов системы (рис. 4.33).

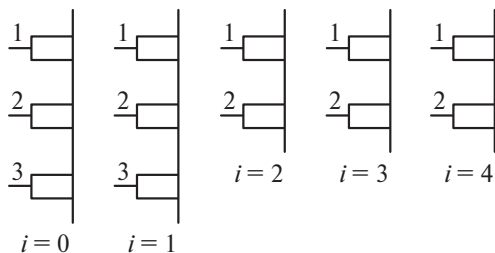


Рис. 4.33. Входы элементов

В таблице 4.18 представлен разработанный оператор сопряжения анализируемой системы.

Таблица 4.18

$i \backslash j$	1	2	3
0	(3, 1)	(4, 1)	(4, 2)
1	(0, 1)	(0, 2)	(3, 2)
2	(0, 3)	(0, 4)	—
3	(1, 1)	(1, 2)	—
4	(1, 3)	(2, 1)	—

Если ввести двойную нумерацию контактов, учитывающую как номер элемента, так и номер контакта, то полученная матрица, отражающая связи в данной схеме, будет представлять собой не что иное, как матрицу смежности ориентированного графа, вершинами которого являются контакты, а дугами — элементарные каналы. Достоинством такого представления оператора сопряжения является то, что к нему может быть применен весь рассмотренный выше аппарат, связанный с матрицей смежности. Недостаток — чрезвычайная громоздкость этой матрицы.

Многоуровневая схема сопряжения. Задание оператора R рассмотренным способом определяет одноуровневую схему сопряжения. Однако точно такой же формальный подход может быть применен и для построения многоуровневых схем сопряжения. Для этого надо учесть тот факт, что любая подсистема S_μ , с одной стороны, сама является системой, содержащей некоторое число элементов, с другой стороны, ее же можно рассматривать как некоторый элемент системы S более высокого уровня.

Подсистема S_μ как самостоятельная система должна иметь контакты $X_j^{(0)\mu}$ и $Y_l^{(0)\mu}$, характеризующие источники и потребители внешней для нее среды (рис. 4.34).

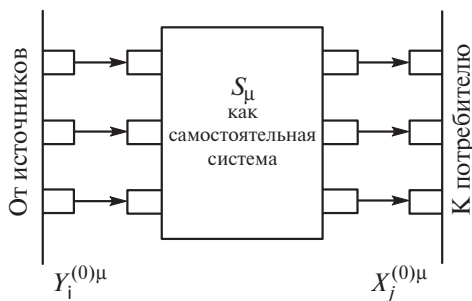


Рис. 4.34. Подсистема как самостоятельная система

С другой стороны, как элемент системы S подсистема S_μ должна содержать входные \bar{X}_j^μ и выходные \bar{Y}_l^μ контакты для связи ее с другими подсистемами (рис. 4.35).

Соответствующие контакты \bar{X}_j^μ и $Y_l^{(0)\mu}$ при $j = l$, а также $X_j^{(0)\mu}$ и \bar{Y}_l^μ при $j = l$ объединяются в «двойные» контакты на границах подсистемы S_μ .

Объединяя рис. 4.34 и рис. 4.35, получим рис. 4.36.

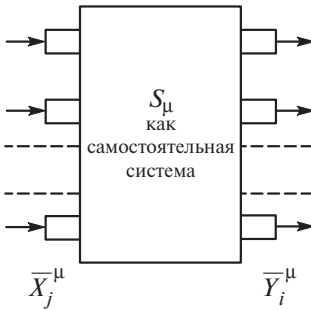


Рис. 4.35. Подсистема как элемент системы

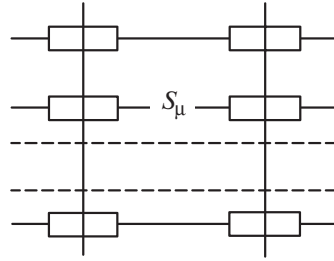


Рис. 4.36. Представление подсистемы S_μ

Введем некоторый оператор R так, чтобы

$$Y_l^{(k)} = R_\mu^{(1)}(X_j^{(i)}), \quad \mu = 1, 2, \dots$$

Этот оператор данному входному контакту $X_j^{(i)}$ элемента C_i подсистемы S_μ ставит в соответствие выходной контакт $Y_l^{(k)}$ той же подсистемы, соединенный с $X_j^{(i)}$ элементарным каналом (если такое соединение в подсистеме S_μ существует). Операторы $R_\mu^{(1)}$ $\mu = 1, 2, \dots$ называются внутренними операторами сопряжения подсистем S_μ . Способы задания $R_\mu^{(1)}$ $\mu = 1, 2, \dots$ такие же, как и у обычного оператора сопряжения.

Операторы $R_\mu^{(1)}$ $\mu = 1, 2, \dots$ называют операторами сопряжения первого уровня.

Рассмотрим теперь подсистему S_μ как элемент системы S . С этих позиций она характеризуется множеством входных и выходных контактов: $[\bar{X}_j^{(\mu)}]_1^{m_\mu}$ и $[\bar{Y}_l^{(\mu)}]_1^{n_\mu}$. Элемент C_0 , представляющий внешнюю среду системы S , будем интерпретировать как подсистему S_0 с входными контактами и $[\bar{X}_j^{(0)}]_1^{m_0}$ и выходными контактами $[\bar{Y}_l^{(0)}]_1^{n_0}$.

Введем оператор

$$\bar{Y}_l^{(k)} = R^{(2)}(\bar{X}_j^{(\mu)}),$$

реализующий отображение множества всех входных контактов подсистем S_0, S_1, \dots, S_μ системы S в множество всех выходных контактов, который данному контакту $\bar{X}_j^{(\mu)}$ ставит в соответствие контакт $\bar{Y}_l^{(k)}$, соединенный с $\bar{X}_j^{(\mu)}$, если такое соединение в S существует. Оператор $R^{(2)}$ будем называть оператором сопряжения второго уровня.

Совокупность внутренних одноуровневых схем сопряжения всех подсистем S ($\mu = 1, 2$) вместе со схемой сопряжения второго уровня называется двухуровневой схемой сопряжения системы S .

Рассмотрим **пример** многоуровневой схемы сопряжения.

Пусть, например, система S состоит из двух подсистем S_1 и S_2 , (т.е. $\mu = 1, 2$) и пусть в S_1 входят элементы C_1 и C_2 , а в подсистему S_2 входят элементы C_3 и C_4 . Структура этой системы представлена на рис. 4.37.

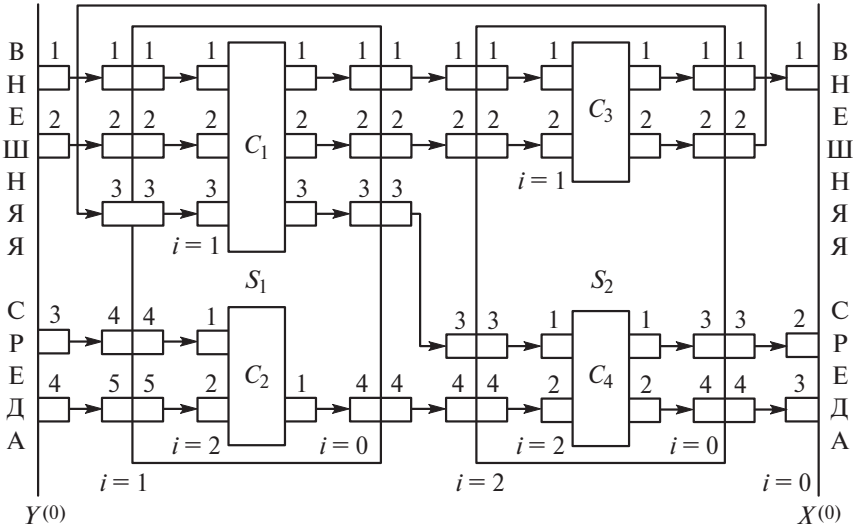


Рис. 4.37. Структура системы

В соответствии со сказанным выше подсистема S_j как самостоятельная система имеет контакты $Y_1^{(0)1}, Y_2^{(0)1}, Y_3^{(0)1}, Y_4^{(0)1}, Y_5^{(0)1}, X_1^{(0)1}, X_2^{(0)1}, X_3^{(0)1}, X_4^{(0)1}$, а как элемент системы S имеет контакты $\bar{X}_1^1, \bar{X}_2^1, \bar{X}_3^1, \bar{X}_4^1, \bar{X}_5^1, \bar{Y}_1^1, \bar{Y}_2^1, \bar{Y}_3^1, \bar{Y}_4^1$, (рис. 4.38).

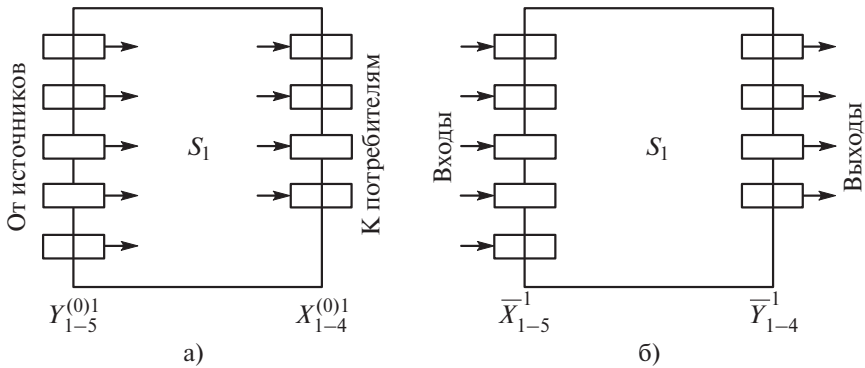


Рис. 4.38. Подсистема как самостоятельная система (а) и как элемент (б)

Аналогичным образом соответствующие множества могут быть построены и для подсистемы S_2 .

В соответствии с введенными определениями разрабатываем внутренние операторы сопряжения. Для облегчения разработки внутренних операторов $R_1^{(1)}$ и $R_2^{(1)}$ подсистемы S_1 и S_2 как самостоятельные системы представлены на рис. 4.39 и рис. 4.40. Внутренние операторы $R_1^{(1)}$ и $R_2^{(1)}$ представлены в таблицах 4.19 и 4.20 соответственно.

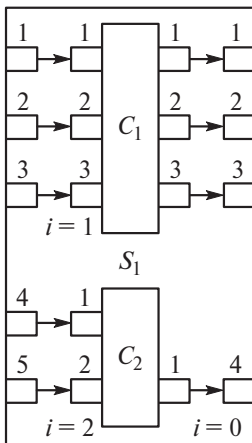


Рис. 4.39. К определению $R_1^{(1)}$

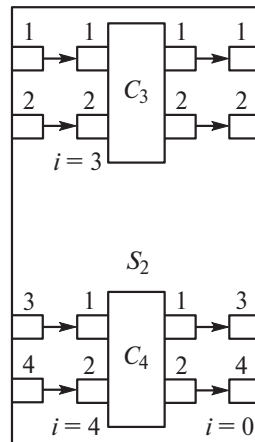


Рис. 4.40. К определению $R_2^{(1)}$

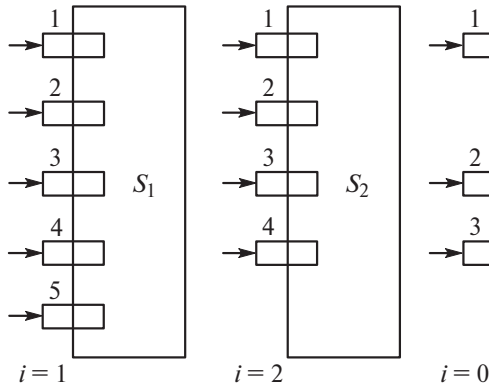
Таблица 4.19

$i \backslash j$	1	2	3	4
0	(1, 1)	(1, 2)	(1, 3)	(2, 1)
1	(0, 1)	(0, 2)	(0, 3)	—
2	(0, 4)	(0, 5)	—	—

Таблица 4.20

$i \backslash j$	1	2	3	4
0	(3, 1)	(3, 2)	(4, 1)	(4, 2)
3	(0, 1)	(0, 2)	—	—
4	(0, 3)	(0, 4)	—	—

Для облегчения разработки оператора сопряжения второго уровня $R^{(2)}$ на рис. 4.41 изображены входы подсистем S_1 , S_2 , и внешней среды.

Рис. 4.41. К определению $R^{(2)}$

В табл. 4.21 представлен оператор сопряжения второго уровня $R^{(2)}$. Нулевая строка соответствует внешней среде, первая строка — подсистеме S_1 , вторая строка — подсистеме S_2 .

Таблица 4.21

$i \backslash j$	1	2	3	4	5
0	(2, 1)	(2, 3)	(2, 4)	—	—
1	(0, 1)	(0, 2)	(2, 2)	(0, 3)	(0, 4)
2	(1, 1)	(1, 2)	(1, 3)	(1, 4)	—

Совокупность внутренних одноуровневых операторов сопряжения всех подсистем вместе с оператором сопряжения второго уровня называется двухуровневой схемой сопряжения системы.

Развитие подобного подхода легко обобщить на создании многоуровневых структурных моделей.

Оператор преобразования. Итак, в соответствии с введенными выше определениями имеем:

$x_j^i(t) \in \bigcup_{i=0}^N [x_j^i(t)]_1^{m_1}$ — один из множества элементарных входных сигналов, воздействующих на i -тый элемент системы;

$y_l^k(t) \in \bigcup_{i=0}^N [y_l^k(t)]_1^{n_1}$ — один из множества элементарных выходных сигналов i -того элемента системы.

Введем некоторый оператор $F(t)$, такой, чтобы

$$y_l^k(t) = F(t)\{x_j^i(t)\}.$$

Назовем его *оператором преобразования* системы. Он ставит в соответствие входному сигналу $x_j^i(t)$ выходной сигнал $y_l^k(t)$, связанный с ним некоторым преобразованием.

Рассмотрим оператор преобразования i -того элемента $F_i(t)$. Его можно задать указаниями следующих 3-х параметров:

- операции или команды, т. е. указания, что надо делать, — f_a ;
- адреса, т. е. указания номера элемента и номера входного контакта, над которым выполняется команда i/X_j^i ;
- номера такта, на котором выполняется команда, — t_e .

Тогда оператор преобразования системы $F(t)$ (табл. 4.22) может быть задан с помощью матрицы, у которой по строкам перечислены все используемые команды f_a (проще в порядке их использования), а по столбцам — такты, на которых используется данная команда t_e (проще в порядке возрастания номера такта); на пересечении соответствующих строк и столбцов указывается адрес элемента и входного контакта i/X_j^i , над которым выполняется команда.

Таблица 4.22

	t_1	t_2
f_1			
f_2			
			i/X_j^i

Перед разработкой $F(t)$ граф, отображающий принципиальную схему, должен быть упорядочен по тактам (с помощью одной из методик, изложенных выше).

Примеры команд: сложить, вычесть, начислить %, ввести документ в систему и т. п.

Алгоритм разработки модели функционирования организационной системы. Введем допущения:

1. Будем считать, что любой документ, циркулирующий в системе, состоит из элементарных частей, содержащих только один параметр – реквизит.

2. Назовем эту элементарную часть документа полем документа.

3. Будем считать, что связь между документами осуществляется через связь между полями.

4. Будем считать, что формирование любого поля осуществляется за два полутакта:

- подача выходной информации к данному полю данного документа;
- преобразование входной информации в соответствии с некоторой функцией.

При этом данное преобразование может осуществляться как с учетом уже содержащейся в этом поле информации, так и без этого учета, но всегда результирующая информация остается в данном поле (рис. 4.42).



Рис. 4.42. К определению поля документа

Таким образом, одно и то же поле до выполнения второго полутакта является входным, после выполнения второго полутакта – выходным. В этом смысле можно считать, что поле имеет несколько независимых в пространстве входов, поскольку в одно и то же поле может вноситься информация как из других полей данного документа, так и из других полей других документов, и только один выход.

5. Будем рассматривать поле P как элемент системы D , D – документ. В соответствии с материалом предыдущего параграфа:

$$X_j^i \in \bigcup_{i=0}^N [X_j^{(i)}]_1^{m_i} \text{ — один из множества входов } i\text{-того документа;}$$

$$Y_j^i \in \bigcup_{i=0}^N [Y_j^{(i)}]_1^{n_i} \text{ — один из множества выходов } i\text{-того документа.}$$

В заключение приведем обобщенный алгоритм разработки формальной модели функционирования.

Исходные данные:

- а) схема документооборота;
- б) функциональные обязанности операторов системы.

Эти данные получаются либо в результате обследования уже функционирующей организационной структуры, либо при ее проектировании.

В соответствии с функциями организационной системы разрабатывается алгоритм обработки информации.

1. Производится нумерация элементов системы.
2. Для каждого элемента системы нумеруются все его входы и выходы.
3. Производится упорядочение элементов системы.
4. Разрабатывается оператор сопряжения R .
5. Разрабатывается оператор преобразования $F(t)$.
6. Разрабатывается моделирующий алгоритм.

Подобное описание может быть использовано для любого уровня:

- первый уровень — уровень полей документов, тогда поле является элементом системы, документ — системой;
- второй уровень — уровень документов, тогда документ является элементом, комплект документов — системой;
- третий уровень — уровень комплектов документов, тогда комплект документов будет элементом, организация в целом (например, банк) — системой (рис. 4.43).

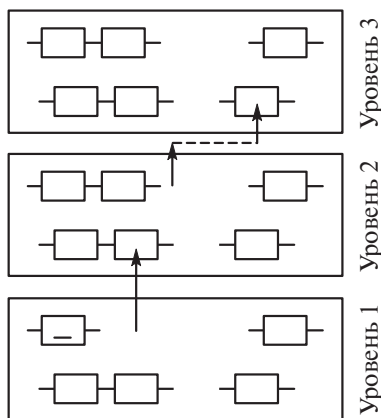


Рис. 4.43. Уровни описания

4.8. Пример разработки модели функционирования организационной системы

Словесное описание процесса функционирования. Пусть у некоторого пользователя домашней локальной сети, именуемого впоследствии клиентом, возникла необходимость продлить свой абонемент пользования Интернетом на какой-то срок. С этой целью он прибывает в офис администраторов сети, где производят эту операцию.

Введём следующие допущения, не снижающие общности постановки задачи:

- клиент располагает требуемой суммой,
- очередь клиентов отсутствует.

На время обслуживания клиента формируется система, состоящая из клиента, администратора и ЭВМ-администратора. Рассмотрим функции всех элементов системы:

Функции клиента:

- Передача администратору удостоверения пользователя сети (УПС), в котором, в числе прочего, содержится логин этого пользователя.
- Словесное указание, на какой месяц нужно оформить абонемент.
- Оплата стоимости абонемента.
- Получение УПС и чека.

Функции администратора:

- Идентификация клиента по данным УПС.
- Ввод в ЭВМ оплаченного срока действия абонемента и суммы, которую заплатил клиент.
- Передача клиенту УПС и чека об оплате.

Функции ЭВМ:

- Активирование пользования Интернетом для данного клиента на оплачиваемый срок.
- Выдача чека, в котором указываются фамилия администратора, принявшего платеж, логин клиента, оплаченный срок действия абонемента, сумма, которую заплатил клиент, дата.

Организационная схема системы. Организационная схема составляется на основе словесного описания, приведенного в предыдущем разделе.

Итак, на время обслуживания клиента формируется система: клиент, администратор, ЭВМ, структурная схема которой представлена на рис. 4.44.

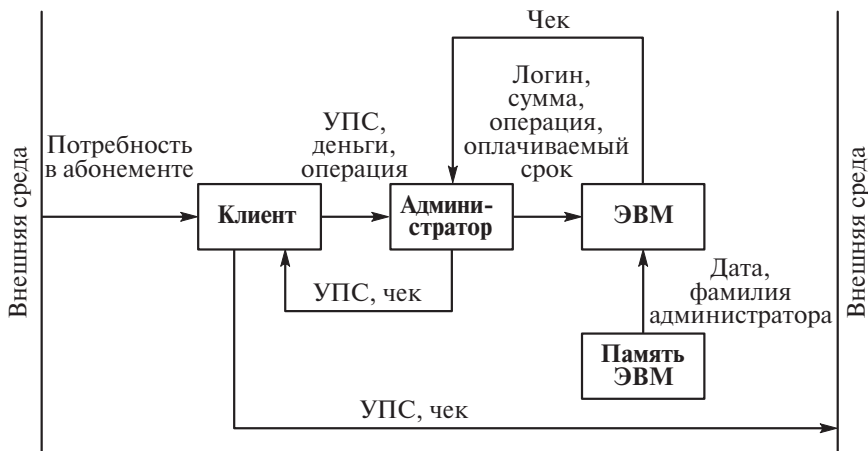


Рис. 4.44. Организационная схема системы

Описание документов системы. Описание документооборота составляется на основе материала двух предыдущих параграфов. Как следует из них, документооборот включает в себя два документа: УПС и чек.

Рассмотрим упрощенную схему документооборота (рис.4.45), на которой отразим только те взаимосвязи между документами, которые используются для реализации рассматриваемой операции.

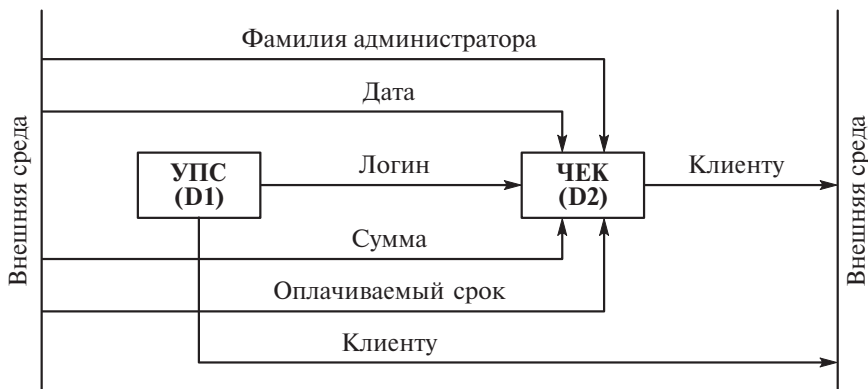


Рис. 4.45. Укрупненная схема документооборота

На рис. 4.46 и 4.47 представлены стилизованные, но очень близкие к подлиннику формы указанных документов. Содержание этих документов раскрывается в словесном описании. Поля данных документов пронумерованы в произвольном порядке.

ФИО пользователя _____ 1
Логин пользователя _____ 2
№ УПС _____ 3

Рис. 4.46. Документ D1. Форма УПС

Фамилия администратора _____ 1
Логин пользователя _____ 2
Оплаченный срок _____ 3
Сумма _____ 4
Дата _____ 5

Рис. 4.47. Документ D2. Форма чека

Укрупненный алгоритм обработки информации. На рис. 4.48 представлен алгоритм обработки информации, составленный на основе словесного описания процесса функционирования системы.

На рис. 4.48 обозначены:

t_i – i -тый такт движения документов;

F_i – обозначение i -той процедуры, содержание которой раскрыто в названии i -того блока алгоритма;

j_i – i -тая команда условного перехода.

В отличие от F_i она не изменяет содержимого документов. Такт движения документов не учитывает команды условного перехода, а такт работы алгоритма учитывает их.

Конкретные условия:

j_1 – есть ли у клиента потребность оформить абонемент на пользование Интернетом на следующий месяц;

j_2 – проверка существования в сети пользователя, соответствующего данному УПС.

Разработка принципиальной схемы документооборота. Для этого необходимо перенумеровать все документы и все поля этих документов. Это сделано в разделе «Описание документов». При-

нципиальная схема разрабатывается на основе всего предыдущего материала. На схеме знаком «*» обозначены операции над всем документом в целом. Для удобства чтения схемы внешняя среда, обозначим ее через 0, будет состоять из 3-х составляющих: 01 – клиент; 02 – администратор; 03 – ЭВМ.

Изобразим все документы, участвующие в документообороте. Сначала изображаем со всеми входными и выходными контактами внешнюю среду и все документы, участвующие в обработке информации. Слева – выходы внешней среды, справа – входы внешней среды. На каждом документе слева – входы, а справа – выходы. Нумеруем по порядку все выходные контакты внешней среды от 1 до 6. Затем нумеруем по порядку, начиная с 7, все выходы документов. Заводим все входные и выходные контакты, как внешней среды, так и всех документов в общую шину. Затем реализуем блоки алгоритма $[t_1, F_1; t_2, F_2; t_3, F_3; t_4, F_4; t_5, F_5]$ (рис. 4.49).


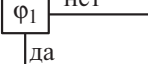

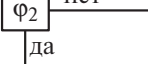
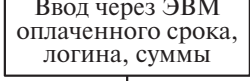
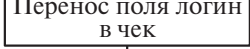

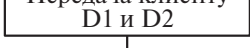

Такт, процедура	Алгоритм	Исполнитель
t_0		Клиент
t_1, F_1		Клиент
t_2, F_2		Администратор
t_3, F_3		Администратор, ЭВМ
t_4, F_4		ЭВМ
t_5, F_5		ЭВМ
t_5, F_5		ЭВМ
t_0		Клиент, администратор, ЭВМ
		

Рис. 4.48. Алгоритм обработки информации

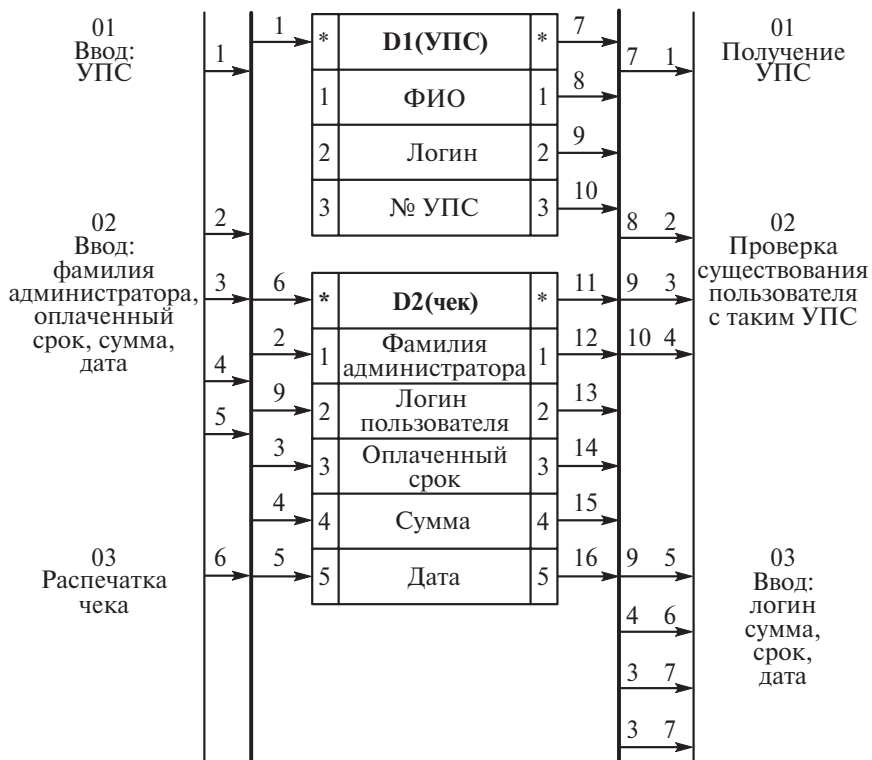


Рис. 4.49. Принципиальная схема документооборота

Оператор сопряжения. На основе принципиальной схемы и алгоритма обработки информации разрабатываем оператор сопряжения R (табл.4.23).

Таблица 4.23

$i \backslash j$		*	1	2	3	4	5	6	7	8	9
		0	01		D1/*	D2/*					
02					D1/1	D1/2	D1/3				
03								D1/2	02/4	02/3	03/5
D1		01/1									
D2		03/6	02/2	D1/2	02/3	02/4	02/5				

Оператор преобразования $F(t)$. На основе алгоритма обработки информации и принципиальной схемы документооборота раскрываем все операторы $F(t)$. На каждом такте t_j ($j = 1-5$) выполняется один оператор $F_i(t)$.

t_1, F_1 – передача оператору УПС. Вводим элементарную команду f_1 – передача документа. $F_1 = f_1(D1/*)$.

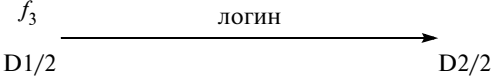
t_2, F_2 – ввод в память ЭВМ логина, срока и суммы. Вводим элементарную команду f_2 – ввод документа в целом или полей документа. $F_2 = f_2(D1/2, 02/3, 02/4)$.

t_3, F_3 – перенос поля логин документа D1 в соответствующее поле документа D2. Вводим элементарную команду f_3 – перенос. $F_3 = f_3(D1/2 \rightarrow D2/2)$. t_4, F_4 – ввод в чек полей: фамилия администратора, оплаченный срок, сумма, дата. $F_4 = f_2(D2/1, D2/3, D2/4, D2/5)$.

t_5, F_5 – передача клиенту D1/* и D2/*. $F_5 = f_1(D1/*, D2/*)$.

Результаты сводим в табл. 4.24 – оператор преобразования системы.

Таблица 4.24

$f_i \backslash t_j$	t_1	t_2	t_3	t_4	t_5
f_1	D1/*				D1/* D2/*
f_2		D1/2 02/3 02/4		D2/1 D2/3 D2/4 D2/5	
f_3					

Примечание. В операции «Перенос», очевидно, необходимо указывать два адреса: откуда брать и куда направлять. В табл. 4.24 это показано с помощью стрелки.

Моделирующий алгоритм. Разработаем некоторый автомат A, состояниями которого были бы такты документооборота t_j , где $j = 1, 2, 3, 4, 5$, а выходным алфавитом этого автомата – элементарные команды f_i , $i = 1, 2, 3$. Как отмечалось выше, на каждом такте t_j выполняется одна или несколько команд f_i , имеющих ненулевой список аргументов. Эта информация определяется на основе оператора преобразования $F(t)$ (табл. 4.24). На такте t_0 никакие элементарные команды не выполняются. Данный автомат моделирует переходы системы из одного состояния в другое толь-

ко в зависимости от условий φ_l . Причем если «да», то φ_l , если «нет», то $\bar{\varphi}_l$. Поведение автомата описывается табл. 4.25, которая составляется на основе алгоритма обработки информации. В отличие от команд f_i, j_l также переводит систему из одного состояния в другое, но она не изменяет содержимого ни одного из документов.

Таблица 4.25

Номер перехода	Текущее состояние	Условие	Следующее состояние
1	t_0	$\bar{\varphi}_1$	t_0
2	—	φ_1	t_1
3	t_0	$\bar{\varphi}_2$	t_3
4	—	φ_2	t_4
5	t_2	—	t_5
6	t_3	—	t_4
7	t_4	—	t_5
8	t_5	—	t_0

Таким образом, модель функционирования организационной системы представлена автоматом, оператором преобразования $F(t)$, оператором сопряжения R и моделирующим алгоритмом (автоматом).

Для ее реализации необходимо закодировать элементарные команды f_i в виде подпрограмм, а сам автомат — в виде главного модуля. При работе автомата необходимо пользоваться оператором $F(t)$, чтобы определять элементарную команду, выполняемую в текущем такте, и оператором R , чтобы устанавливать источники информации, подаваемой на входы документов, обрабатываемых с помощью f_i .

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Перечислите цели и задачи структурного анализа АСУ.
2. Рассмотрите пути совершенствования организационной, функциональной, алгоритмической структур.
3. Назовите уровни описания структуры АСУ.
4. Сформулируйте формализованные способы описания структуры АСУ.

5. Какую цель преследует введение порядковой функции?
6. Рассмотрите алгоритм введения порядковой функции.
7. Что представляет собой числовая функция на графе?
8. Рассмотрите топологическую декомпозицию структур АСУ.
9. Как определяется порядок элементов?
10. Что является формальным признаком для выделения входных и выходных элементов потока?
11. Приведите пример анализа потоков информации в АСУ.
12. Дайте характеристику основных структурно-топологических характеристик систем.
13. Приведите основные виды топологических структур.
14. Как определяется связность структуры?
15. Каким образом определяется избыточность структуры?
16. Зачем вводят параметр ε^2 ?
17. Что представляет собой компактность структуры?
18. Как определяется степень централизации структуры?
19. Проведите сравнительный анализ топологических структур.
20. Определите одноуровневую схему сопряжения.
21. Что представляет собой многоуровневая схема сопряжения?
22. Каким образом разрабатывается оператор преобразования?
23. Приведите алгоритм разработки модели функционирования организационной системы.
24. Что представляет собой алгоритм, моделирующий процесс функционирования организационной системы?
25. Рассмотрите пример модели функционирования организационной системы.

Глава 5

МОДЕЛИ СИНТЕЗА СТРУКТУРЫ АСУ

В главе рассмотрены некоторые вопросы синтеза моделей структуры АСУ. Сначала производится формализация общей задачи синтеза в самом общем случае. Затем разрабатываются математические выражения для частных критериев оптимизации и ограничений в частных задачах оптимизации. Основными параметрами в рассматриваемых моделях являются затраты на реализацию функциональных задач АСУ и время решения этих задач. Различные комбинации этих параметров и составляют суть той или иной частной задачи синтеза оптимальной структуры АСУ.

В результате изучения данной главы студенты должны знать подходы к постановке и решению задач подобного класса, уметь анализировать полученные результаты.

Авторы сознательно ограничили рамки излагаемого материала, посвященного синтезу моделей структуры АСУ. Это вызвано, во-первых, ограниченностью объема учебника, во-вторых, требованиями соответствующего Государственного образовательного стандарта высшего профессионального образования и, в-третьих, наличием специального курса «Проектирование АСОИУ».

5.1. Формализация общей задачи синтеза структуры АСУ

Структура системы — это способ организации системы из отдельных элементов с их взаимосвязями, которые зависят от распределения функций и целей, выполняемых системой. Таким образом, структура (одно из определений) — это способ организации целого из составных частей.

В зависимости от задачи исследования в понятие структуры системы может вкладываться различный смысл. Так, при разработке структуры АСУ в это понятие входит, например, определение множества элементов системы и связей между ними, распределение задач, возлагаемых на технические средства АСУ, по уровням и элементам системы и выбор комплекса технических средств, обеспечивающего их своевременное решение.

Основными проблемами, возникающими при разработке структуры АСУ, являются:

- 1) определение необходимого числа уровней иерархии;
- 2) установление между уровнями правильных взаимоотношений, что связано с задачами согласования целей элементов различных уровней и оптимальным стимулированием их работы;
- 3) распределение ответственности;
- 4) выбор конкретных схем управления и создание контуров принятия решения;
- 5) организация информационных потоков;
- 6) выбор соответствующих технических средств.

Все эти вопросы взаимосвязаны и образуют сложную проблему.

Рассмотрим задачу синтеза структуры АС в самом общем виде. Для ее формализации введем следующие обозначения:

P — множество возможных принципов построения системы или ее элементов. Возможные принципы бывают обычно заданы и выбираются при синтезе системы;

π — выбранные принципы построения системы или ее элементов. Очевидно, $\pi \in P$;

F — множество взаимосвязанных функций (операций), выполняемых системой. Каждому набору принципов π построения системы соответствует некоторое множество функций $F(\pi)$, из которого при проектировании системы необходимо выбрать подмножество $f \in F(\pi)$, достаточное для реализации выбранных принципов π ;

A — множество возможных взаимосвязанных элементов системы. Подобными элементами, например, могут быть узлы системы, технические средства, пункты обслуживания, отдельные исполнители, коллективы и т. п.;

a — выбранные взаимосвязанные элементы системы. Введем также операцию отображения W элементов множества F на элементы множества A . Оптимальное отображение должно обеспечивать экстремум некоторой (или некоторых) целевой функции при выполнении заданных ограничений.

В общем случае задача синтеза оптимальной структуры состоит в определении:

$$\pi \in P; \quad (5.1)$$

$$f \in F(\pi); \quad (5.2)$$

$$a \in A; \quad (5.3)$$

$$[f \in F(\pi)] W [a \in A]. \quad (5.4)$$

Если заданы принципы построения системы, то синтез оптимальной структуры состоит в определении (5.2), (5.3) и (5.4). Если заданы принципы построения системы и выполняемые ею функции — то в определении (5.4) и (5.3). Если заданы принципы построения системы, выполняемые ею функции и элементы системы — то в определении (5.4), т. е. в нахождении оптимального отображения множества выполняемых функций на множестве взаимосвязанных элементов.

Задача анализа состоит в определении характеристик системы при заданных условиях (5.1) — (5.4). Если для некоторых элементов возникает проблема большой нагрузки, то условия с (5.1) по (5.4) должны учитывать правила функционирования элементов. В ряде случаев эти правила определяются при синтезе, поскольку от них может зависеть распределение функций и взаимосвязей в системе.

5.2. Частные задачи синтеза оптимальной структуры АСУ

Сложность синтеза оптимальной структуры АСУ приводит к тому, что на практике ставят и решают более частные задачи синтеза, такие, например, как оптимальное распределение возлагаемых на АСУ функций по заданным уровням и углам системы, определение оптимальной реализации функций в АСУ, выбор комплекса технических средств, обеспечивающего качественную реализацию функций, и т. д.

Рассмотрим некоторые частные постановки задач формализованного распределения множества решаемых задач между узлами АСУ при различных критериях и ограничениях.

5.2.1. Частные критерии оптимизации

а) Минимизация затрат на реализацию задач в АСУ.

$$\min \sum_{i=1}^I \sum_{j=1}^J W_{ij} x_{ij}, \quad (5.5)$$

где $i = \overline{1, I}$ — множество функциональных задач, реализуемых в АСУ, $j = \overline{1, J}$ — множество обслуживающих узлов системы управления, W_{ij} — затраты на реализацию i -той задачи в j -том узле.

Кроме того, пусть $x_{ij} = \begin{cases} 1 \\ 0 \end{cases}$; $x_{ij} = 1$, если i -тая задача выполняется в j -том узле и $x_{ij} = 0$ — в противном случае.

б) Минимизация общего времени решения всех задач АСУ

$$\min \sum_{i=1}^I \sum_{j=1}^J t_{ij} x_{ij}, \quad (5.6)$$

где t_{ij} — время решения i -той задачи в j -том узле.

в) Минимизация максимального времени решения задач в АСУ

$$\min \left(\max_{i=1}^I \sum_{j=1}^J t_{ij} x_{ij} \right). \quad (5.7)$$

Возможна оптимизация по более сложным критериям, включающим (5.5) — (5.7), а также использование критериев более общего типа, таких, как получение максимальной прибыли, обеспечение требуемого времени готовности системы и т. д.

5.2.2. Ограничения в частных задачах синтеза

а) На связи между задачами, т. е. задан граф

$$\sigma_E. \quad (5.8)$$

б) На связи между узлами, т. е. задан граф

$$\sigma_M. \quad (5.9)$$

в) На общие затраты на реализацию задач в АСУ

$$\sum_{i=1}^I \sum_{j=1}^J W_{ij} x_{ij} \leq W_{\text{доп}}. \quad (5.10)$$

г) На затраты на реализацию задач в узлах

$$\sum_{i=1}^I W_{ij} x_{ij} \leq W_{i_{\text{доп}}}. \quad (5.11)$$

д) На загрузку каждого узла

$$\sum_{i=1}^I \lambda_{ij} t_{ij} x_{ij} \leq \pi_j, \quad j = \overline{1, J}, \quad (5.12)$$

где λ_i — интенсивность поступления i -той задачи на решение. Возможны дополнительные требования к равномерности загрузки узлов.

е) На общее время решения всех задач

$$\sum_{i=1}^I \sum_{j=1}^J t_{ij} x_{ij} \leq T. \quad (5.13)$$

ж) На время решения отдельных задач

$$\sum_{i=1}^J t_{ij} x_{ij} \leq \tau_p, \quad i = \overline{1, I}. \quad (5.14)$$

5.2.3. Первая частная задача синтеза оптимальной структуры АСУ

Необходимо так распределить i задач $i = \overline{1, I}$ между j узлами $j = \overline{1, J}$, чтобы обеспечить минимум общих затрат (5.5) или минимум общего времени решения (5.6) при выполнении ограничений на загрузку каждого из узлов (5.12), или на затраты в каждом j -том узле (5.11).

Математическая модель этой задачи может быть записана следующим образом: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (5.15)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \leq b_j, \quad j = \overline{1, J}, \quad (5.16)$$

$$\sum_{i=1}^J x_{ij} \leq 1, \quad i = \overline{1, I}, \quad x_{ij} = \{1, 0\}. \quad (5.17)$$

В этих соотношениях:

a_{ij} — затраты (время решения) i -той задачи в j -том узле;

b_j — допустимые затраты (загрузка) в j -том узле;

x_{ij} — переменная, равная 1, если i -тая задача решается в j -том узле, и равная 0 — в противном случае.

Условие (5.17) означает, что каждая задача должна решаться только в одном узле.

Наиболее удобным для решения данного класса задач является метод «ветвей и границ». Применительно к данной задаче он заключается в направленном движении по вершинам дерева, полученного путем фиксирования части переменных x_{ij} , $x_{ij} = \{0, 1\}$.

Вершины первого уровня получают, поочередно закрепляя первую задачу за первым узлом, вторым и т. д., т. е. фиксируя $x'_{ij} = 1$ для $j = 1, 2, 3, \dots$ при $i = 1$.

Вершины второго уровня получают, фиксируя $x''_{ij} = 1$ для $j = 1, 2, 3, \dots$ при $i = 2$ и т. д. Для \leq каждой вершины находят оценку

$$\sum_{i < i^*} a_{ij} + \sum_{i \geq i^*} \bar{a}_{ij}, \quad (5.18)$$

где i^* — число рассмотренных уровней ветвления; $\bar{a}_{ij} = \min_j a_{ij}$.

Стратегия ветвления может быть улучшена за счет использования специфических свойств рассматриваемой задачи, что существенно при решении задач большой размерности.

Вначале из матрицы коэффициентов $\|a_{ij}\|$ системы (5.15) исключаем все элементы, для которых выполняется условие $a_{ij} > b_j$, $i = \overline{1, I}$, $j = \overline{1, J}$. При этом для любой строчки возможны следующие варианты:

- исключены все элементы a_{ij} , тогда решение отсутствует;
- остался лишь один элемент a_{ij} , он обязательно входит в оптимальное решение, если оно существует. Значение b_j заменяется на $b_j - a_{ij}$, и этот элемент в дальнейшем поиске не участвует;
- осталось несколько элементов, они участвуют в дальнейшем поиске оптимального решения.

5.2.4. Вторая частная задача синтеза оптимальной структуры АСУ

Необходимо так распределить i задач $i = \overline{1, I}$ между j узлами $j = \overline{1, J}$, чтобы обеспечить минимум общих затрат (5.5) или минимум общего времени решения (5.6) при выполнении ограничений на общее время решения (5.13) или общие затраты (5.10) соответственно.

Математическая модель этой задачи может быть записана следующим образом: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (5.19)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J b_{ij} x_{ij} \leq B. \quad (5.20)$$

$$\sum_{j=1}^J x_{ij} \leq 1, \quad x_{ij} = \{1, 0\}. \quad (5.21)$$

В этих соотношениях:

a_{ij} — затраты (время решения) i -той задачи в j -том узле;

b_{ij} — время решения (затраты) i -той задачи в j -том узле;

B — общее время решения (затраты) всех задач.

Для решения этой задачи прежде всего берутся минимальные элементы в каждой строке матрицы коэффициентов $\|a_{ij}\|$ и проверяется выполнение условия (5.20) для соответствующих элементов матрицы коэффициентов $\|b_{ij}\|$.

Если условие (5.20) выполняется, это и будет оптимальным решением.

Если условие (5.20) не выполняется, то из матрицы коэффициентов $\|a_{ij}\|$ и $\|b_{ij}\|$ исключают те элементы, которые не могут

войти ни в одно допустимое решение. Для этого последовательно рассматриваются все элементы матрицы $\|a_{ij}\|$ и проверяется условие

$$\sum_{i=1}^{l-1} \bar{b}_{ij} + b_{lj} + \sum_{i=l+1}^I b_{ij} \leq B, \quad l = \overline{1, I}, \quad j = \overline{1, J}, \quad (5.22)$$

где \bar{b}_{ij} — минимальный элемент в соответствующей строке;

b_{lj} — рассматриваемый элемент, $j \neq l$.

Иначе говоря, каждая задача последовательно закрепляется за каждым из узлов и проверяется выполнение условия (5.20) в лучшем случае.

Если условие (5.22) нарушается, то соответствующий элемент b_{lj} не входит в допустимое решение и исключается из матрицы $\|a_{ij}\|$. Из матрицы $\|a_{ij}\|$ исключается соответствующий элемент a_{ij} .

Из условия (5.21) следует, что в каждой строке может быть только один элемент. Поэтому $\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij}$ без учета выра-

жения (5.18) равен $\sum_{i=1}^I \min \sum_{j=1}^J a_{ij} x_{ij}$. Отсюда, если для элементов

одновременно выполняются условия $a_{ij} \geq a_{lj}$ и $b_{ij} \geq b_{lj}$, ($j \neq l$), то эти элементы могут быть исключены из рассмотрения.

Хотя исключение элементов не всегда приводит к оптимальному решению, однако объем вычислений резко сокращается.

Далее используется метод «ветвей и границ». В отличие от предыдущей задачи, ветвление осуществляется с учетом ограничения (5.20), что существенно сокращает число рассматриваемых вариантов. Оценка для каждой вершины находится по элементам матрицы (5.19) аналогично предыдущей задаче (5.18). Ограничение при этом имеет вид

$$\sum_{i < i^*} b_{ij} + \sum_{i \geq i^*} \bar{b}_{ij}, \quad (5.23)$$

где i^* — уровень ветвления;

$$\bar{b}_{ij} = \min_j b_j.$$

5.2.5. Третья частная задача синтеза оптимальной структуры АСУ

Необходимо так распределить i задач $i = \overline{1, I}$ между j узлами $j = \overline{1, J}$, чтобы обеспечить минимум общих затрат (5.5) или минимум общего времени решения (5.6) при выполнении ограничений на общее время решения (5.13) и загрузку узлов (5.12), либо на общие затраты (5.10) и загрузку узлов (5.12) соответственно.

Математическая модель этой задачи может быть записана в следующем виде: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (5.24)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J b_{ij} x_{ij} \leq B, \quad (5.25)$$

$$\sum_{j=1}^J c_{ij} x_{ij} \leq c_j, \quad j = \overline{1, J}, \quad (5.26)$$

$$\sum_{j=1}^J x_{ij} \leq 1, \quad x_{ij} = \{0, 1\}. \quad (5.27)$$

Для решения этой задачи прежде всего из матриц коэффициентов $\|a_{ij}\|$, $\|b_{ij}\|$, $\|c_{ij}\|$ исключаются элементы, которые заведомо не могут войти в оптимальное решение. Исключение элементов b_{ij} и c_{ij} из матриц систем (5.25) и (5.26) осуществляется аналогично рассмотренной выше, т. е. исключаются все элементы, для которых не выполняется условие (5.22). Оценка для матрицы коэффициентов (5.24) находится аналогично оценке системы (5.18) в первой задаче.

5.3. Примеры частных задач синтеза оптимальной структуры АСУ

Пример 1.¹ Сначала решим методом «ветвей и границ» следующую задачу: найти минимум

$$\min \sum_{i=1}^4 \sum_{j=1}^4 a_{ij} x_{ij} \quad (5.28)$$

¹ Пример имеет целью продемонстрировать процедуру ветвления.

при

$$\sum_{i=1}^4 a_{ij}x_{ij} = 1, \quad (5.29)$$

$$\sum_{j=1}^4 a_{ij}x_{ij} = 1, \quad (5.30)$$

$$x_{ij} = \{0, 1\}, \quad (5.31)$$

$$a_{ij} = \begin{vmatrix} 8 & 5 & 6 & 4 \\ 5 & 3 & 2 & 2 \\ 1 & 7 & 3 & 4 \\ 2 & 3 & 8 & 5 \end{vmatrix}. \quad (5.32)$$

Условие (5.29) означает, что каждый узел может решать только одну задачу. Условие (5.30) означает, что каждая задача может решаться только в одном узле.

Будем изображать множество вариантов кружками, в верхней части которых проставлен номер множества, а в нижней — значение нижней границы (рис. 5.1).

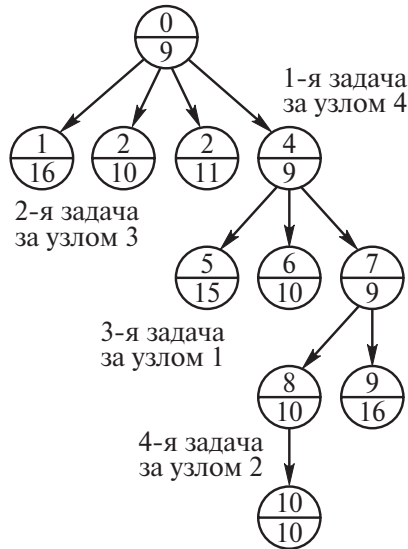


Рис. 5.1. Процедура ветвления

Для вычисления нижней границы используется соотношение

$$\sum_{i < i^*} b_{ij} + \sum_{i \geq i^*} \bar{b}_{ij}, \quad (5.33)$$

где $\bar{a}_{ij} = \min_j a_{ij}$, i^* — число рассмотренных уровней ветвления.

Для исходного множества (обозначим его через «0») соотношение (5.3.6) имеет вид $\sum_{i \geq 0}^4 a_{ij} = \min_j a_{ij}$, т. е. из матрицы (5.32)

выбираются минимальные числа, причем условие (5.29) может нарушаться. Итак,

$$\sum_{i \geq 0}^4 \bar{a}_{ij} = 4 + 2 + 1 + 2 = 9.$$

Вершины первого уровня получим, поочередно закрепляя первую задачу за первым, вторым, третьим и четвертым узлами. Соответствующие значения нижней границы представлены в табл. 5.1.

Вершины второго уровня получим, закрепив первую задачу за четвертым узлом. Соответствующие значения нижней границы представлены в табл. 5.2.

Из табл. 5.2. следует, что вторую задачу следует закрепить за третьим узлом.

Вершины третьего уровня получим, закрепив первую задачу за четвертым узлом и вторую за третьим. Соответствующие значения нижней границы представлены в табл. 5.3.

Из табл. 5.3 следует, что задача 3 должна быть закреплена за узлом один. Четвертая задача однозначно закрепляется за узлом два.

Таблица 5.1

j	$\sum_{i < i^*} a_{ij}$	$\sum_{i \geq i^*} \bar{a}_{ij}$	
1	8	+ 2 + 3 + 3	16
2	5	+ 2 + 1 + 2	10
3	6	+ 2 + 1 + 2	11
4	4	+ 2 + 1 + 2	9

$i = 1$

Таблица 5.2

j	$\sum_{i < i^*} a_{ij}$	$\sum_{i \geq i^*} \bar{a}_{ij}$	
1	4 + 5	+ 3 + 3	15
2	4 + 3	+ 1 + 2	12
3	4 + 2	+ 1 + 2	9

Таблица 5.3

j	$\sum_{i < i^*} a_{ij}$	$\sum_{i \geq i^*} \bar{a}_{ij}$	
1	4 + 2 + 1	+ 3	10
2	4 + 2 + 7	+ 2	15

Окончательный ответ представлен в матрице

$$x_{ij} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{vmatrix}.$$

Значение целевой функции равно 10.

Пример 2. Рассмотрим решение первой частной задачи синтеза оптимальной структуры. Найти

$$\min \sum_{i=1}^4 \sum_{j=1}^4 a_{ij} x_{ij}$$

при ограничениях

$$\sum_{i=1}^4 a_{ij} x_{ij} \leq b_j, \quad a_{ij} = \begin{vmatrix} 8 & 5 & 6 & 4 \\ 5 & 3 & 2 & 2 \\ 1 & 7 & 3 & 4 \\ 2 & 3 & 8 & 5 \end{vmatrix},$$

$$\sum_{i=1}^4 x_{ij} = 1, \quad x_{ij} = \{0, 1\}, \quad b_j = \{3 \quad 1 \quad 5 \quad 3\}.$$

В соответствии с ранее рассмотренным алгоритмом производим упрощение матрицы $\|a_{ij}\|$, для чего исключаем элементы, для

которых выполняется условие $a_{ij} > b_j$. Первая строчка после исключения не содержит ни одного элемента, т. е. первая задача не может быть решена: решение отсутствует.

Пусть $b_j = |3 \ 6 \ 5 \ 3|$. Тогда после исключения $\|a_{ij}\|$ имеет вид

$$a_{ij} = \begin{vmatrix} - & 5 & - & - \\ - & 3 & 2 & 2 \\ 1 & - & 3 & - \\ 2 & 3 & - & - \end{vmatrix}.$$

Первая строчка содержит только один элемент $a_{12} = 5$, следовательно, он обязательно войдет в решение. В отличие от рассмотренного ранее примера, мы сняли условие, согласно которому один узел может быть загружен только одной задачей. Ресурс на второй узел равен 6, следовательно, остается резерв: $6 - 5 = 1$,

Далее процедура аналогична рассмотренной выше, но каждый раз ищутся минимальные элементы в столбцах и проверяется, не нагружен ли данный узел.

Итак, $x_{12} = 1$. Имеем

$$a_{ij} = \begin{vmatrix} - & 3 & 2 & 2 \\ 1 & - & 3 & - \\ 2 & 3 & - & - \end{vmatrix}, \quad b_{ij} = |3 \ 1 \ 5 \ 3|, \quad i = 2, 3, 4.$$

Выбираем минимальные элементы в каждой строке. Загрузка не превышает заданную. Окончательно

$$x_{ij} = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} \quad \text{или} \quad x'_{ij} = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix}.$$

Значение целевой функции в первом случае $5 + 2 + 1 + 2 = 10$, во втором $5 + 2 + 1 + 2 = 10$. Варианты равнозначны.

Пример 3. Рассмотрим числовое решение задачи минимизации общих затрат при ограничениях на общее время решения, т. е. будем искать

$$\min \sum_{i=1}^4 \sum_{j=1}^4 a_{ij} x_{ij} \quad (5.34)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J b_{ij} x_{ij} \leq B, \quad (5.35)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad (5.36)$$

$$x_{ij} = \{0, 1\}. \quad (5.37)$$

Пусть

$$\|a_{ij}\| = \begin{vmatrix} 3 & 7 & 2 & 4 \\ 4 & 8 & 1 & 3 \\ 6 & 9 & 6 & 2 \\ 6 & 10 & 7 & 1 \\ 7 & 5 & 3 & 1 \end{vmatrix}, \quad \|b_{ij}\| = \begin{vmatrix} 1,5 & 3 & 2 & 9 \\ 2 & 6 & 5 & 10 \\ 3 & 7 & 6 & 11 \\ 4 & 8 & 7 & 12 \\ 4 & 9 & 8 & 5 \end{vmatrix}, \quad B \leq 20.$$

Сначала находим минимальные элементы в каждой строке матрицы $\|a_{ij}\|$ и проверяем, удовлетворяется ли условие (5.35) по одноименным элементам матрицы $\|b_{ij}\|$:

$$\sum_{i=1}^I b_{ij} = 2 + 5 + 11 + 12 + 5 > 20.$$

Условие (5.35) не выполняется, и задачу «в лоб» решить не удастся. Приступим к упрощению матрицы. Для матрицы $\|b_{ij}\|$ последовательно для всех элементов проверяется условие

$$\sum_{i=1}^{l-1} \bar{b}_{ij} + b_{lj} + \sum_{i=l+1}^I b_{ij} \leq B, \quad l = \overline{1, I}, \quad j = \overline{1, J}. \quad (5.38)$$

где \bar{b}_{ij} — минимальные элементы строк; b_{lj} — рассматриваемый элемент.

$$\begin{aligned} & 1,5 + 2 + 3 + 4 + 4 = 14,5, \\ \text{Для } i = 1: & \quad 3 + 2 + 3 + 4 + 4 = 16, \\ & \quad 2 + 2 + 3 + 4 + 4 = 15, \\ & \quad 9 + 2 + 3 + 4 + 4 = 22. \end{aligned}$$

Элемент b_{14} не удовлетворяет условию (5.38), он исключается из матрицы $\|b_{ij}\|$, и одноименный элемент a_{14} исключается из матрицы $\|a_{ij}\|$. Аналогично для $i = 2$:

$$\begin{aligned} & 2 + 1,5 + 3 + 4 + 4 = 14,5, \\ & 6 + 1,5 + 3 + 4 + 4 = 18,5, \\ & 5 + 1,5 + 3 + 4 + 4 = 17,5, \\ & 10 + 1,5 + 3 + 4 + 4 = 22,5. \end{aligned}$$

$$\begin{array}{l}
 \text{Для } i = 3: \\
 \quad 3 + 1,5 + 2 + 4 + 4 = 14,5, \\
 \quad 7 + 1,5 + 2 + 4 + 4 = 18,5, \\
 \quad 6 + 1,5 + 2 + 4 + 4 = 17,5, \\
 \quad 11 + 1,5 + 2 + 4 + 4 = 22,5. \\
 \\
 \text{Для } i = 4: \\
 \quad 4 + 1,5 + 2 + 3 + 4 = 14,5, \\
 \quad 8 + 1,5 + 2 + 3 + 4 = 18,5, \\
 \quad 7 + 1,5 + 2 + 3 + 4 = 17,5, \\
 \quad 12 + 1,5 + 2 + 3 + 4 = 22,5.
 \end{array}$$

Легко видеть, что для $i = 5$ все элементы удовлетворяют условию (5.38).

Поскольку в каждой строчке может быть только один элемент и в обеих матрицах осуществляется минимизация, то при одновременном выполнении условия

$$a_{ij} > a_{lj}, \quad b_{ij} > b_{lj}$$

эти элемент могут быть исключены из рассмотрения.

После соответствующих упрощений матрицы имеют вид

$$\|a_{ij}\| = \begin{vmatrix} 3 & - & 2 & - \\ 4 & - & 1 & - \\ 6 & - & - & - \\ 6 & - & - & - \\ 7 & 5 & 3 & 1 \end{vmatrix}, \quad \|b_{ij}\| = \begin{vmatrix} 1,5 & - & 2 & - \\ 2 & - & 5 & - \\ 3 & - & - & - \\ 4 & - & - & - \\ 4 & 9 & 8 & 5 \end{vmatrix}.$$

Из матрицы $\|a_{ij}\|$ выбираем минимальные элементы и подсчитываем время решения: $2 + 5 + 3 + 4 + 5 = 19 < 20$. Задача решена. Если бы это не удалось, пришлось бы вести ветвление и каждый минимальный вариант проверять на условие (5.35). Итак, ответ:

$$x_{ij} = \begin{vmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}.$$

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Рассмотрите основные проблемы, возникающие при разработке структуры АСУ.
2. Приведите общую формулировку синтеза структуры АСУ.
3. Что представляют собой частные критерии оптимизации?

4. Какие основные виды ограничений необходимо учитывать в частных задачах оптимизации?
5. Что представляет собой первая частная задача оптимизации структуры АСУ?
6. Приведите алгоритм решения первой частной задачи оптимизации.
7. В чем суть второй частной задачи оптимизации структуры АСУ?
8. Каковы отличительные особенности алгоритма решения второй частной задачи оптимизации?
9. Что представляет собой третья частная задача оптимизации структуры АСУ?
10. Рассмотрите алгоритм решения третьей частной задачи оптимизации структуры АСУ.

Глава 6

МОДЕЛИ И ПРОЦЕСС ПРИНЯТИЯ РЕШЕНИЙ В АСУ

В этой главе рассмотрены формальные процедуры и формальный аппарат принятия решений человеком в контуре автоматизированного управления. В отличие от автоматических систем, в которых после их запуска роль человека сводится к контролю за работой системы, в автоматизированных системах человек является главным определяющим звеном этих систем, поэтому при проектировании АСУ необходимо учитывать такие «человеческие» факторы, как индивидуальная и групповая психология, пропускная способность человека, скорость реакции, допустимые объемы перерабатываемой информации и т.п. При этом процесс принятия решений сопровождается и подкрепляется использованием соответствующих формальных моделей.

В главе прежде всего анализируются проблема и процесс принятия решений. Формулируется общая постановка задачи принятия решений. Приводится классификация задач принятия решений. Рассматриваются однокритериальная задача принятия решений, принятие решений в условиях риска и неопределенности, многокритериальные задачи принятия решений. Материал главы сопровождается соответствующими примерами.

В результате изучения данной главы студенты должны знать подходы к постановке и решению задач подобного класса, уметь производить необходимые расчеты и анализировать полученные результаты.

6.1. Проблема принятия решений в больших системах

Проблема принятия решений составляет суть любой целенаправленной человеческой деятельности. Несмотря на все многообразие ситуаций и условий, в которых производится выбор решения, сам процесс выбора носит достаточно универсальный характер.

Ситуации, в которых осуществляется выбор, характеризуют следующие основные черты:

1) *Наличие цели (целей).*

Необходимость принятия решения диктуется только наличием цели, которую необходимо достичь. Если цель отсутствует, то и нет никакой необходимости принимать решение.

2) *Наличие альтернативных линий поведения.*

Решения принимаются только тогда, когда существует более одного способа их достижения. Причем каждый из способов может характеризоваться различной вероятностью достижения цели, а также, различными затратами, необходимыми для достижения целей.

3) *Наличие ограничивающих факторов.*

Естественно, что лицо, принимающее решение, не обладает бесконечными возможностями. Все множество ограничивающих факторов можно разбить на три основные группы:

- экономические факторы (деньги, производственные и людские ресурсы, время и т.п.),
- технические факторы (габариты, вес, энергопотребление, надежность, точность и т.п.),
- социальные факторы, которые учитывают требования человеческой этики и морали, а также экологические требования.

Проблему принятия решений проиллюстрируем на примере выбора оптимального варианта проекта.

Применительно к АСУ объектом проектирования в зависимости от этапа разработки могут быть собственно АСУ, функциональные или обеспечивающие подсистемы, технические устройства, задачи, алгоритмы или программы.

Каждый вариант проекта характеризуется определенной совокупностью параметров. Все их в принципе можно разделить на две группы: *внешние* и *внутренние*.

К *внешним* параметрам отнесем те параметры, которые характеризуют объект с точки зрения заказчика. Например, применительно к функциональной задаче внешними параметрами могут

быть: время решения задачи, требуемые объемы памяти, надежность решения и т.п. К *внутренним* параметрам отнесем те параметры, которые характеризуют объект с точки зрения разработчика и которые, вообще говоря, безразличны заказчику. В том же примере такими параметрами могут быть: способ кодирования, виды электрических сигналов, может даже тип ЭВМ и т.п.

В самом общем случае каждый из внешних параметров

$$g_1, g_2, \dots, g_n$$

каким-то образом зависит от внутренних параметров

$$a_1, a_2, \dots, a_m,$$

т. е.

$$\begin{aligned} g_1 &= g_1(a_1, a_2, \dots, a_m), \\ g_2 &= g_2(a_1, a_2, \dots, a_m), \\ &\dots \\ g_n &= g_n(a_1, a_2, \dots, a_m). \end{aligned}$$

Таким образом, каждому варианту проекта, или (что то же самое) вектору внутренних параметров,

$$\bar{a} = (a_1, a_2, \dots, a_m) \tag{6.1}$$

соответствует вектор внешних параметров

$$\bar{g}(\bar{a}) = [g_1(\bar{a}), g_2(\bar{a}), \dots, g_n(\bar{a})]. \tag{6.2}$$

Будем называть допустимым вариантом проекта такой набор значений внутренних параметров:

$$\bar{a}_{\text{доп}} = (\bar{a}_{1, \text{доп}}, \bar{a}_{2, \text{доп}}, \dots, \bar{a}_{m, \text{доп}}), \tag{6.3}$$

при котором удовлетворяются все заданные ограничения.

Основными ограничениями, как следует из вышеизложенного, являются:

1. Ограничения, вытекающие из ограниченности ресурсов:

$$C_k \leq C_{k, \text{доп}},$$

где $k = \overline{1, l}$; C_k — k -тый вид ресурса, l — число ресурсов.

Особо следует отметить ограничение на стоимость.

2. Ограничение, связанное со сроком разработки

$$T \leq T_{\text{доп}}.$$

3. Ограничения, вытекающие из необходимости поддержания внешних параметров в заданном диапазоне

$$g_{i, \text{доп}}^{\min} \leq g_i \leq g_{i, \text{доп}}^{\max}, \quad i = \overline{1, n}.$$

4. Ограничения, вытекающие из необходимости поддержания внутренних параметров в заданном диапазоне:

$$a_{i, \text{доп}}^{\min} \leq a_i \leq a_{i, \text{доп}}^{\max}, \quad j = \overline{1, m}.$$

Теперь сформулируем задачу оптимального проектирования: ищется вариант проекта (набор внутренних параметров) (6.1), который принадлежал бы множеству допустимых проектов (6.3) и обращал бы в оптимум вектор внешних параметров (6.2).

Иначе говоря, требуется найти

$$a_{\text{opt}} = (a_j, \text{opt}), \quad j = \overline{1, m},$$

$$g(g_i) = \text{opt } g(g_i), \quad i = \overline{1, n},$$

где opt — оператор оптимизации. Он определяет выбранный принцип оптимизации.

6.2. Процесс принятия решений

Процесс принятия управленческих решений — это преобразование исходной информации (информации состояния) в выходную информацию (информацию управления) — *приказ*.

Принято делить решения на формальные и творческие. Если преобразование информации выполняется с помощью математических моделей, то выработанное решение считается формальным; если решение принимается в результате скрытой работы интеллекта лица, принимающего решения, то это решение считается творческим. Такое деление в достаточной степени условно, поскольку ни чисто формального, ни чисто творческого решения в природе не существует. Если решения принимаются с помощью математических моделей, то знания и опыт человечества (элементы творчества) используются при создании этих моделей, а интуиция (элемент творчества) используется в момент, когда лицо, принимающее решение, задает то или иное значение исходной информации или из множества альтернативных вариантов в качестве решения выбирает один. Если основным инструментом выбора решений является интеллект человека, то

формальные методы, носителем которых является вся наука, скрыто присутствуют в его знаниях и опыте.

В соответствии с делением решений на творческие и формальные все множество проблем, соответствующих любому процессу принятия решений, делят на два принципиально различных класса: это проблемы концептуального характера и проблемы формально-математического, или вычислительного, характера.

К концептуальным проблемам относят сложные логические проблемы, которые невозможно решить с помощью только формально математических методов и ЭВМ. Очень часто эти проблемы уникальны в том смысле, что они решаются впервые и не имеют прототипов в прошлом. Обычно концептуальные проблемы решаются на уровне руководителей с привлечением группы экспертов. В качестве экспертов выступают высококвалифицированные специалисты из различных областей науки и техники. При решении концептуальных проблем формально математические методы играют только вспомогательную роль, а главное значение придается эрудиции, опыту и интуиции эксперта. К числу концептуальных проблем относят, в частности, такие проблемы, как анализ и выбор целей, выявление совокупности показателей, характеризующих следствия принятого решения, выбор из этих показателей критерия оптимальности и т.п. Формализация эвристических процедур является содержанием нового научного направления, которое называется «Неформальная теория принятия решений». В дальнейшем мы будем предполагать, что цели управления, соответствующие им критерии оптимальности и ограничения заданы и обсуждению не подлежат. Иначе говоря, в дальнейшем мы будем изучать количественную, или формальную теорию принятия решений.

Процесс принятия решений является сложной итерационной процедурой. Основные этапы этого процесса и их последовательность представлены на рисунке 6.1. Процесс начинается с появления соответствующих стимулов. В том или ином виде осуществляется первоначальная формулировка проблемы. Эта формулировка позволяет приступить к определению целей, критериев, ограничений, составляется список альтернатив, производится сбор информации и прогноз развития проблемы. На этом заканчивается 1-й этап. Вся эта работа позволяет уточнить и конкретизировать проблему. После этого последовательно разрабатывается постановка задачи, математическая модель, метод решения, алгоритм решения, производится оценка альтернатив и выбор среди них оптимальной. Проводимый в заключение 2-го этапа

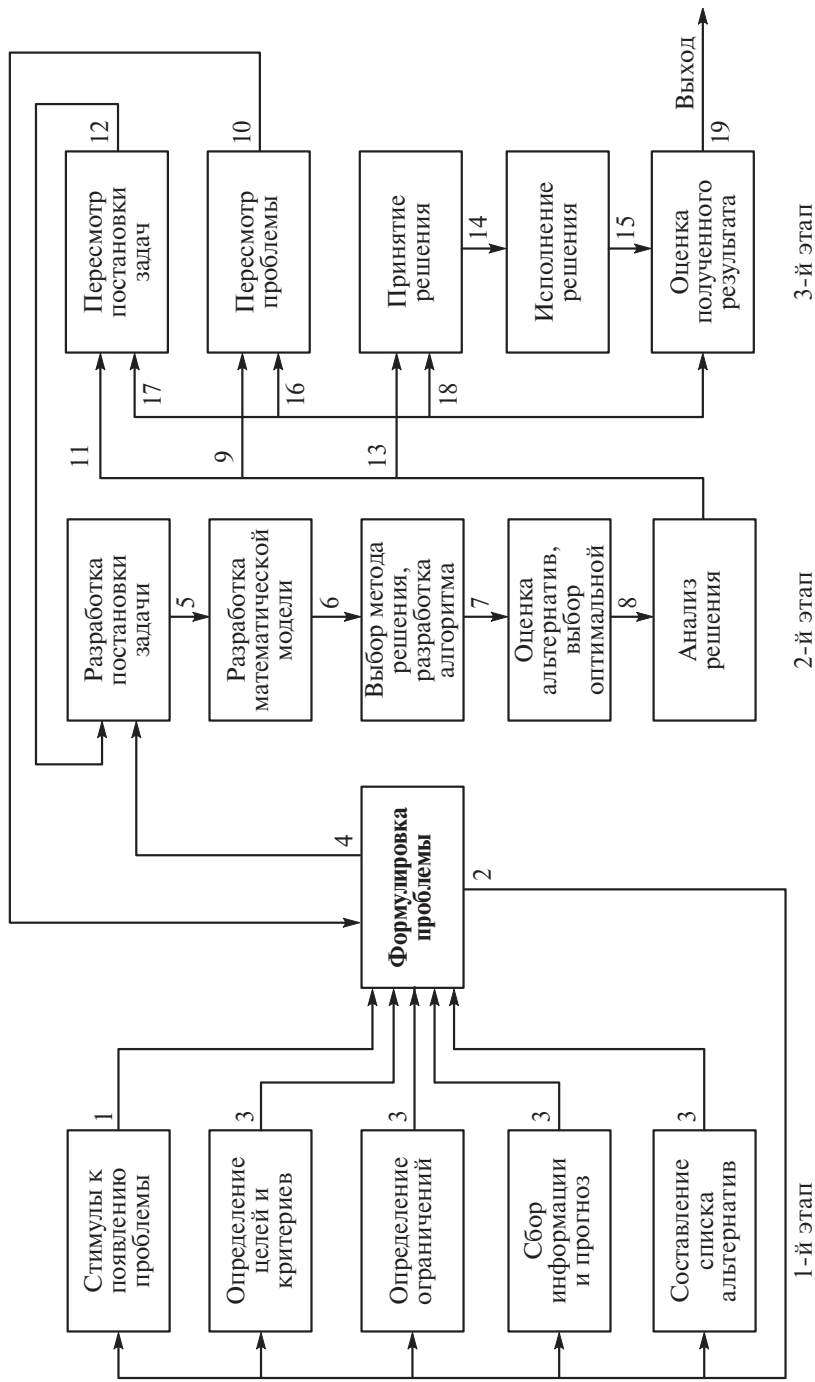


Рис. 6.1. Процесс принятия решений

анализ решения может привести либо к пересмотру проблемы и, следовательно, к повторению 1-го и 2-го этапов, либо к пересмотру постановки задачи и, следовательно, к повторению только 2-го этапа. Указанные возвраты могут быть и неоднократными. В результате переходят к 3-му этапу: принятие решения, исполнение решения, оценка полученного результата. Последняя процедура может привести к точно таким же последствиям, как и анализ решения. В результате проблема оказывается разрешенной.

6.3. Общая постановка задачи принятия решений

Пусть эффективность выбора того или иного решения определяется некоторым критерием F , допускающим количественное представление. В самом общем случае все факторы, от которых зависит эффективность выбора, можно разбить на две группы:

- *контролируемые (управляемые) факторы*, выбор которых определяется лицом, принимающим решения. Обозначим их через X_1, X_2, \dots, X_l ;
- *неконтролируемые (неуправляемые) факторы*. Они характеризуют условия, в которых осуществляется выбор; и лицо, принимающее решение, не может повлиять на их величину. В состав неконтролируемых факторов включают и время t . Неконтролируемые факторы, в зависимости от информированности о них лица, принимающего решения, можно разделить на три подгруппы:
 - *детерминированные неконтролируемые факторы* — это неслучайные фиксированные величины, значение которых в точности известно. Обозначим их через A_1, A_2, \dots, A_p ;
 - *стохастические неконтролируемые факторы* — случайные величины с известными законами распределения. Обозначим их через Y_1, Y_2, \dots, Y_q ;
 - *неопределенные неконтролируемые факторы*, для каждого из которых известна только область, внутри которой находится неизвестный закон их распределения. Обозначим эти величины через Z_1, Z_2, \dots, Z_r .

В соответствии с выделенными факторами критерий оптимальности можно представить в следующем виде:

$$F = F(X_1, X_2, \dots, X_l, A_1, A_2, \dots, A_p, Y_1, Y_2, \dots, Y_q, Z_1, Z_2, \dots, Z_r, t). \quad (6.4)$$

Величины X, A, Y, Z в самом общем случае могут быть скалярами, векторами, матрицами.

Величины контролируемых (управляемых) параметров в общем случае обычно ограничены естественным рядом причин, например, ограниченностью ресурсов. Математически эти ограничения можно записать в следующем виде:

$$q_i = q_i(X_1, X_2, \dots, X_p, A_1, A_2, \dots, A_p, Y_1, Y_2, \dots, Y_q, Z_1, Z_2, \dots, Z_p, t), \quad (6.5)$$

причем это выражение может быть меньше или равно, равно, больше или равно $b_i, i = 1, n$.

Условия (6.5) определяют области $W_{x_1}, W_{x_2}, \dots, W_{x_l}$ — пространства, внутри которых расположены допустимые значения управляемых факторов X_1, X_2, \dots, X_l . Совершенно аналогично можно расписать ограничения и на области значений неконтролируемых параметров. Поскольку критерий оптимальности F есть количественная мера достижения целей управления, то математически цель управления выражается в стремлении к максимально возможному увеличению (или уменьшению) значения критерия оптимальности F , т. е.

$$F \rightarrow \max(\text{или } \min).$$

Средством достижения этой цели является выбор управлений X_1, X_2, \dots, X_l , принадлежащих к областям их допустимых значений $W_{x_1}, W_{x_2}, \dots, W_{x_l}$.

Таким образом, общая постановка задачи принятия решений может быть сформулирована следующим образом: при заданных значениях фиксированных и неконтролируемых факторов A_1, A_2, \dots, A_p стохастических неконтролируемых факторов Y_1, Y_2, \dots, Y_q с учетом неопределенных факторов Z_1, Z_2, \dots, Z_l найти $X_{1 \text{ opt}}, X_{2 \text{ opt}}, \dots, X_{l \text{ opt}}$, принадлежащее областям их допустимых значений $W_{x_1}, W_{x_2}, \dots, W_{x_l}$, которые по возможности обращали бы в максимум (минимум) критерий оптимальности F .

6.4. Классификация задач принятия решений

Воспользуемся классификацией, в основу которой положены четыре важных классификационных признака (рис. 6.2):

- 1) количество целей управления и соответствующих им критериев оптимальности;
- 2) наличие или отсутствие зависимости критерия оптимальности и ограничений от времени;

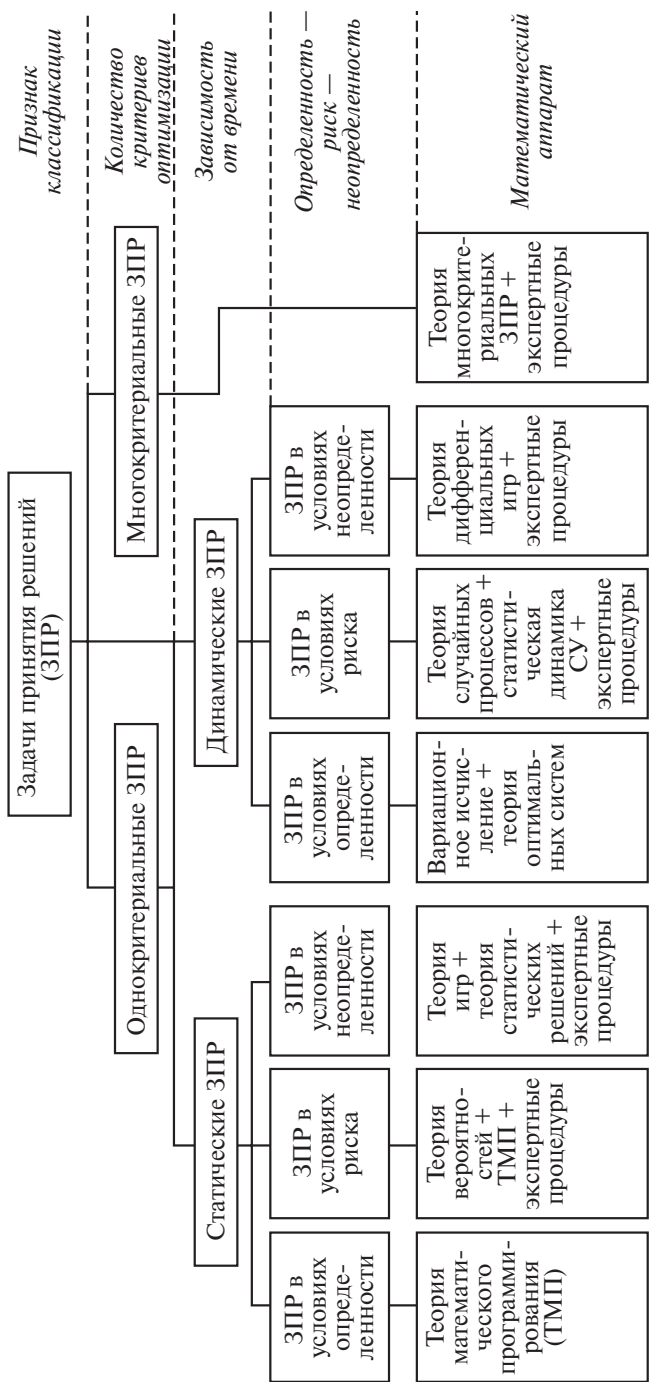


Рис. 6.2. Классификация ЗПР и методов их решения

- 3) наличие случайных и неопределенных факторов, этот признак называют признаком «определенность — риск — неопределенность»;
- 4) используемый для их решения математический аппарат.

По первому классификационному признаку ЗПР делятся на одноцелевые или однокритериальные (скалярные) и многоцелевые или многокритериальные (векторные) ЗПР.

По второму классификационному признаку ЗПР делится на статические (не зависящие от времени) и динамические (зависящие от времени) ЗПР. Динамическим ЗПР присущи две особенности:

- 1) в качестве критерия оптимальности в динамических ЗПР выступает не функция, как в статических ЗПР, а функционал, зависящий от функций времени;
- 2) в составе ограничений обычно присутствуют так называемые дифференциальные связи, описываемые дифференциальными уравнениями.

По признаку «определенность — риск — неопределенность» ЗПР подразделяют на три больших подкласса:

- 1) принятие решений в условиях определенности, или детерминированные ЗПР. Они характеризуются однозначной детерминированной связью между принятым решением и его исходом;
- 2) принятие решений при риске, или стохастические ЗПР. Любое принятое решение может привести к одному из множества возможных исходов, причем каждый исход имеет определенную вероятность появления. Предполагается, что эти вероятности заранее известны лицу, принимающему решения;
- 3) принятие решений в условиях неопределенности. Любое принятое решение может привести к одному из множества возможных исходов, вероятности появления которых неизвестны.

Деление задач принятия решений по используемому для их решения математическому аппарату показано на рис. 6.2.

6.5. Однокритериальные задачи принятия решений

Пусть исход управляемого мероприятия зависит от выбранного решения (стратегии управления) и некоторых неслучайных, фиксированных факторов, полностью известных лицу, принимающему решение. Стратегии управления могут быть представлены в виде значений n -мерного вектора $X = (x_1, x_2, \dots, x_n)$, на компо-

ненты которого наложены ограничения, обусловленные рядом естественных причин и имеющие вид

$$\begin{aligned} g_i &= g_i(A_i, X) \{ \leq, =, \geq \} b_i; \\ i &= \overline{1, m}; m \{ \leq, =, \geq \} n, \end{aligned} \quad (6.6)$$

где A_i — некоторый массив фиксированных неслучайных параметров.

Условия (6.6) определяют область Ω_X допустимых значений стратегий X .

Эффективность управления характеризуется некоторым численным критерием оптимальности F

$$F = F(X, C), \quad (6.7)$$

где C — массив фиксированных, неслучайных параметров.

Массивы A_i и C характеризуют свойства объектов, участвующих в управлении, и условия протекания управления.

Перед лицом, принимающим решение, стоит задача выбора такого значения $\bar{X} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$ вектора управления $X = (x_1, x_2, \dots, x_n)$ из области Ω_X его допустимых значений, которое максимизирует значение критерия оптимальности F , а также значение \bar{F} этого максимума

$$\bar{F} = F(\bar{X}, C) = \max_{X \in \Omega_X} F(X, C), \quad (6.8)$$

где область Ω_X представляется условием (6.6).

В (6.8) символы \bar{F} и \bar{X} обозначают максимально достижимое в условиях (6.6) значение критерия оптимальности F и соответствующее ему оптимальное значение вектора управления X .

Совокупность соотношений (6.6), (6.7) и (6.8) представляет собой общий вид математической модели однокритериальной статической детерминированной ЗПР.

Задача в такой постановке полностью совпадает с общей постановкой задачи математического программирования. Поэтому весь арсенал методов, разработанных для решения задач математического программирования, может быть использован для решения задач принятия решений данного класса. Мы не будем здесь останавливаться на обзоре соответствующих методов решения.

Рассмотрим **пример** однокритериальной статической детерминированной ЗПР.

Пусть необходимо отображать некоторое количество информационных моделей (например, картографическую информацию). Для отображения любой из моделей всегда требуется решать n различных задач Z_1, Z_2, \dots, Z_n (отображение символов,

отображение векторов, поворот и перемещение изображения, масштабирование и т.п.). Все задачи взаимно независимы. Для решения этих задач могут быть использованы m различных микропроцессоров M_1, M_2, \dots, M_m . В течение времени T микропроцессор M_j может решить a_{ij} задач типа Z_i ($i = \overline{1, n}; j = \overline{1, m}$), т. е. решить задачу Z_i несколько раз по одному и тому же алгоритму, но для различных исходных данных.

Информационную модель можно отображать только в том случае, если она содержит полный набор результатов решения всех задач Z_1, Z_2, \dots, Z_n .

Требуется распределить задачи по микропроцессорам так, чтобы число информационных моделей, синтезированных за время T , было максимально. Иначе говоря, необходимо указать, какую часть времени T микропроцессор M_j , должен занимать решением задачи Z_i .

Обозначим эту величину через x_{ij} (если эта задача не будет решаться на данном микропроцессоре, то $x_{ij} = 0$).

Очевидно, что общее время занятости каждого микропроцессора решением всех задач не должно превышать общего запаса времени T , «доля» — единицы. Таким образом, имеем следующие ограничительные условия:

$$\sum_{i=1}^n x_{ij} \leq 1, \quad j = \overline{1, m}.$$

Общее количество решений N_i задачи Z_i , полученных всеми микропроцессорами вместе,

$$N_i = \sum_{j=1}^m a_{ij} x_{ij}, \quad i = \overline{1, n}.$$

Так как информационная модель может быть синтезирована лишь из полного набора результатов решения всех задач, то количество информационных моделей F будет определяться минимальным из чисел N_i .

Итак, имеем следующую математическую модель: требуется найти такие x_{ij} , чтобы обращалась в максимум функция F .

$$F = \min \sum_{j=1}^m a_{ij} x_{ij}, \quad i = \overline{1, n}$$

при

$$\sum_{i=1}^n x_{ij} \leq 1, \quad j = \overline{1, m}, \quad x_{ij} \geq 0.$$

6.6. Принятие решений в условиях риска

Как отмечалось, каждая выбранная стратегия управления в условиях рынка связана с множеством возможных исходов, причем каждый исход имеет определенную вероятность появления, известную заранее человеку, принимающему решение.

При оптимизации решения в подобной ситуации стохастическую ЗПР сводят к детерминированной. Широко используют при этом следующие два принципа: искусственное сведение к детерминированной схеме и оптимизация в среднем.

В первом случае неопределенная, вероятностная картина явления приближенно заменяется детерминированной. Для этого все участвующие в задаче случайные факторы приближенно заменяются какими-то неслучайными характеристиками этих факторов (как правило, их математическими ожиданиями).

Этот прием используется в грубых, ориентировочных расчетах, а также в тех случаях, когда диапазон возможных значений случайных величин сравнительно мал. В тех случаях, когда показатель эффективности управления линейно зависит от случайных параметров, этот прием приводит к тому же результату, что и «оптимизация в среднем».

Прием «оптимизация в среднем» заключается в переходе от исходного показателя эффективности Q , являющегося случайной величиной:

$$Q = Q(X, A, y_1, y_2, \dots, y_q),$$

где X — вектор управления; A — массив детерминированных факторов; y_1, y_2, \dots, y_q — конкретные реализации случайных фиксированных факторов Y_1, Y_2, \dots, Y_q к его усредненной, статической характеристике, например, к его математическому ожиданию $M[Q]$:

$$F = M[Q] = \iint \dots \int Q(X, A, y_1, y_2, \dots, y_q) \times \\ \times f(y_1, y_2, \dots, y_q) dy_1, dy_2, \dots, dy_q = F(X, A, B). \quad (6.9)$$

Здесь B — массив известных статистических характеристик случайных величин Y_1, Y_2, \dots, Y_q ; $f(y_1, y_2, \dots, y_q)$ — закон распределения вероятностей случайных величин Y_1, Y_2, \dots, Y_q .

При оптимизации в среднем по критерию (6.9) в качестве оптимальной стратегии \bar{X} будет выбрана такая, стратегия, которая, удовлетворяя ограничениям на область Q_x допустимых значений

вектора X , максимизирует значение математического ожидания $F = M[Q]$ исходного показателя эффективности Q , т. е.

$$\begin{aligned}\bar{F} &= F(\bar{X}, A, B) = \max F(X, A, B) = \\ &= \max_{X \in \Omega_X} M[X, A, y_1, y_2, \dots, y_q].\end{aligned}\quad (6.10)$$

В том случае, если число возможных стратегий i конечно $i = \bar{1}, \bar{I}$ и число возможных исходов j конечно $j = \bar{1}, \bar{J}$, то выражение (6.10) переписывается в виде

$$\bar{F} = F(\bar{X}) = \max_{1 \leq i \leq I} [F(X_i)] = \max_{1 \leq i \leq I} \left[\sum_{j=1}^J P_{ij} Q_{ij} \right], \quad (6.11)$$

где Q_{ij} — значение показателя эффективности управления в случае появления j -того исхода при выборе i стратегии управления; P_{ij} — вероятность появления j -того исхода при реализации i -той стратегии.

Из выражений (6.10) и (6.11) следует, что оптимальная стратегия \bar{X} приводит к гарантированному наилучшему результату только при многократном повторении ситуации в одинаковых условиях. Эффективность каждого отдельного выбора связана с риском и может отличаться от средней величины как в лучшую, так и в худшую сторону.

Сравнение двух рассмотренных принципов оптимизации в стохастических ЗПР показывает, что они представляют собой детерминизацию исходной задачи на разных уровнях влияния стохастических факторов. «Искусственное сведение к детерминированной схеме» представляет собой детерминизацию на уровне факторов, «оптимизация в среднем» — на уровне показателя эффективности.

После выполнения детерминизации могут быть использованы все методы, применимые для решения однокритериальных статических детерминированных ЗПР.

Рассмотрим **пример** однокритериальной статической задачи принятия решений в условиях риска.

Для создания картографической базы данных необходимо кодировать картографическую информацию. Использование поэлементного кодирования приводит к необходимости использования чрезвычайно больших объемов памяти. Известен ряд методов кодирования, позволяющих существенно сократить требуемый объем памяти (например, линейная интерполяция, интерполяция классическими многочленами, кубические сплайны и т.п.). Основным показателем эффективности метода кодирования является коэффициент сжатия информации. Однако значение это-

го коэффициента зависит от вида кодируемой картографической информации (гидрография, границы вида графической кодируемой информации и т. п.). Обозначим через Q_{ij} , $i = \overline{1, n}$; $j = \overline{1, m}$ значение коэффициента сжатия i -того метода кодирования для j -того вида информации. Конкретный район, подлежащий кодированию, заранее неизвестен. Однако предварительный анализ картографической информации всего региона и опыт предыдущих работ позволяет вычислить вероятность появления каждого из видов информации. Обозначим через P_{ij} вероятность появления

j -того вида, $\sum_{j=1}^m P_{ij} = 1$. Тогда, используя метод оптимизации в среднем, следует выбрать такой метод кодирования, для которого

$$\bar{F} = \max_{j=1}^m \sum_{i=1}^n P_{ij} Q_{ij}, \quad i = \overline{1, n}.$$

6.7. Принятие решений в условиях неопределенности

Прежде всего отметим принципиальное различие между стохастическими факторами, приводящими к принятию решения в условиях риска, и неопределенными факторами, приводящими к принятию решения в условиях неопределенности. И те, и другие приводят к разбросу возможных исходов результатов управления. Но стохастические факторы полностью описываются известной стохастической информацией, эта информация и позволяет выбрать лучшее в среднем решение. Применительно к неопределенным факторам подобная информация отсутствует.

В общем случае неопределенность может быть вызвана либо противодействием разумного противника, либо недостаточной осведомленностью об условиях, в которых осуществляется выбор решения.

Принятие решений в условиях разумного противодействия является объектом исследования теории игр. Мы здесь не будем касаться этих вопросов.

Рассмотрим принципы выбора решений при наличии недостаточной осведомленности относительно условий, в которых осуществляется выбор. Такие ситуации принято называть «играми с природой».

В терминах «игры с природой» задача принятия решений может быть сформулирована следующим образом. Пусть лицо, принимающее решение, может выбрать один из m возможных вари-

антов своих решений: x_1, x_2, \dots, x_m и пусть относительно условий, в которых будут реализованы возможные варианты, можно сделать n предположений: y_1, y_2, \dots, y_n . Оценки каждого варианта решения в каждом условии (x_i, y_j) известны и заданы в виде матрицы выигрышей лица, принимающего решения: $A = |a_{ij}|$.

Предположим вначале, что априорная информация о вероятностях возникновения той или иной ситуации y_j отсутствует.

Теория статистических решений предлагает несколько критериев оптимальности выбора решений. Выбор того или иного критерия неформализуем, он осуществляется человеком, принимающим решения, субъективно, исходя из его опыта, интуиции и т. п. Рассмотрим эти критерии.

Критерий Лапласа. Поскольку вероятности возникновения той или иной ситуации y_j неизвестны, будем их все считать равновероятными. Тогда для каждой строки матрицы выигрышей подсчитывается среднее арифметическое значение оценок. Оптимальному решению будет соответствовать такое решение, которому соответствует максимальное значение этого среднего арифметического, т. е.

$$\bar{F} = F(\bar{X}, Y) = \max_{1 \leq i \leq m} \left(\frac{1}{n} \right) \sum_{j=1}^n a_{ij}.$$

Критерий Вальда. В каждой строчке матрицы выбираем минимальную оценку. Оптимальному решению соответствует такое решение, которому соответствует максимум этого минимума, т. е.

$$\bar{F} = F(\bar{X}, Y) = \max_{1 \leq i \leq m} \min_{1 \leq j \leq n} a_{ij}.$$

Этот критерий очень осторожен. Он ориентирован на наихудшие условия, только среди которых и отыскивается наилучший и теперь уже гарантированный результат.

Критерий Сэвиджа. В каждом столбце матрицы находится максимальная оценка $\max_{1 \leq i \leq m} a_{ij}$ и составляется новая матрица, элементы которой определяются соотношением

$$r_{ij} = \max_{1 \leq i \leq m} a_{ij} - a_{ij}.$$

Величину r_{ij} называют риском, под которым понимают разность между максимальным выигрышем, который имел бы место, если бы было достоверно известно, что наступит ситуация y_j , и выигрышем при выборе решения x_i в условиях y_j . Эта новая

матрица называется матрицей рисков. Далее из матрицы рисков выбирают такое решение, при котором величина риска принимает наименьшее значение в самой неблагоприятной ситуации, т.е.

$$\bar{F} = F(\bar{X}, Y) = \max_{1 \leq i \leq m} \min_{1 \leq j \leq n} \left(\max_{1 \leq i \leq m} a_j - a_{ij} \right).$$

Сущность этого критерия заключается в минимизации риска. Как и критерий Вальда, критерий Сэвиджа очень осторожен. Они различаются разным пониманием худшей ситуации: в первом случае — это минимальный выигрыш, во втором — максимальная потеря выигрыша по сравнению с тем, чего можно было бы достичь в данных условиях.

Критерий Гурвица. Вводится некоторый коэффициент α , называемый «коэффициентом оптимизма», $0 < \alpha < 1$. В каждой строке матрицы выигрышей находится самая большая оценка $\max_{1 \leq j \leq n} a_{ij}$ и самая маленькая $\min_{1 \leq j \leq n} a_{ij}$.

Они умножаются соответственно на α и $(1 - \alpha)$ и затем вычисляется их сумма. Оптимальному решению будет соответствовать такое решение, которому соответствует максимум этой суммы, т.е.

$$\bar{F} = F(\bar{X}, Y) = \max_{1 \leq i \leq m} \left[\alpha \max_{1 \leq j \leq n} a_{ij} + (1 - \alpha) \min_{1 \leq j \leq n} a_{ij} \right].$$

При $\alpha = 0$ критерий Гурвица трансформируется в критерий Вальда. Это случай крайнего «пессимизма». При $\alpha = 1$ (случай крайнего «оптимизма») человек, принимающий решение, рассматривает на то, что ему будет сопутствовать самая благоприятная ситуация. «Коэффициент оптимизма» α назначается субъективно, исходя из опыта, интуиции и т.п. Чем более опасна ситуация, тем более осторожным должен быть подход к выбору решения и тем меньшее значение присваивается коэффициенту α .

Примером принятия решений в условиях неопределенности может служить рассмотренная выше задача выбора метода кодирования картографической информации, когда вероятности появления того или иного вида этой информации известны.

6.8. Многокритериальные задачи принятия решений

В примере, рассмотренном в предыдущем параграфе, имелся всего один критерий — F но значительно чаще принимаемое решение описывается совокупностью критериев f_1, f_2, \dots, f_k . Кроме того, каждый из этих критериев, назовем их локальными крите-

риями, — характеризуется своим коэффициентом относительной важности. Обозначим эти коэффициенты через $\lambda_1, \lambda_2, \dots, \lambda_k$.

Итак, совокупность локальных или частных критериев f_q , где $q = \overline{1, k}$, образует интегральный или векторный критерий оптимальности принимаемого решения. Обозначим это так:

$$F = \{f_q\},$$

где F — интегральный критерий.

В свою очередь, коэффициенты относительной важности λ_q , $q = \overline{1, k}$ образуют вектор важности: $\bar{\lambda} = \{\lambda_q\}$.

Как и прежде, задача заключается в том, чтобы найти оптимальное значение X из области допустимых значений λ_x . Каждый локальный критерий характеризует одно какое-либо качество принимаемого решения. Например, в задаче выбора ЭВМ локальными критериями могут быть: стоимость, быстродействие, объем оперативной памяти и т. п. Совокупность этих локальных критериев образует интегральный критерий, различный для каждого типа машины, и с помощью него можно производить сравнение различных типов машин, или сравнение качества принимаемого решения. Формально оптимальное решение \bar{X} может быть условно записано следующим образом:

$$\bar{F} = \bar{F}(\bar{X}) = \underset{X \in W_x}{\text{opt}} [F(X), \bar{\lambda}]. \quad (6.12)$$

В этом соотношении, точнее говоря, в этой формальной записи: \bar{F} — оптимальное значение интегрального критерия, \bar{X} — оптимальные значения управляемых параметров задачи, opt — оператор оптимизации, который определяет выбранный принцип оптимизации, $\bar{\lambda}$ — вектор важности.

Область допустимых значений W_x можно разбить на две непересекающиеся подобласти:

1) W_x^C — область «согласия», в которой качество принимаемого решения может быть улучшено по одному или нескольким локальным критериям без ухудшения хотя бы одного из оставшихся локальных критериев.

2) W_x^E — область «компромиссов», в которой улучшение решения по одному или нескольким локальным критериям обязательно приводит к снижению значений $\bar{\lambda}$ одного или нескольких оставшихся локальных критериев.

Пример. Пусть необходимо выбрать одну ЭВМ из двух различных типов, и пусть локальными критериями являются стоимость и быстродействие.

Случай 1 — пусть ЭВМ-1 лучше и по стоимости и по быстродействию, чем ЭВМ-2, и тогда, при переходе от ЭВМ-2 к ЭВМ-1 оба критерия «согласны» улучшить свои значения. Тогда говорят, что оба эти варианта лежат в области согласия, и очевидно, что выбирать следует первый вариант, а второй просто отбрасывается.

Случай 2 — пусть у ЭВМ-1 лучше (меньше) стоимость, но худшее быстродействие, чем у ЭВМ-2. Выбирая ЭВМ-1, мы улучшаем решение по стоимости, но ухудшаем его по быстродействию, выбирая ЭВМ-2, мы ухудшаем решение по стоимости, но улучшаем по быстродействию.

Для того, чтобы выбрать окончательно какой-либо вариант, мы должны найти некоторый компромисс, поэтому говорят, что эти два варианта лежат в области компромиссов.

Поэтому первый этап принятия решений — это разбиение области допустимых значений на область согласия и область компромиссов. Это разбиение позволяет существенно сократить число рассматриваемых вариантов.

Далее необходимо задаться некоторой «схемой компромисса», или, говоря иначе, раскрыть смысл оператора оптимизации — opt — выражения (6.12).

В дальнейшем нам будет удобнее от допустимого пространства управляющих воздействий W_X^C перейти к допустимому пространству локальных критериев W_F^C и тогда расписанная выше модель может быть формализована следующим образом:

$$\bar{F} = \bar{F}(\bar{X}) = \underset{x \in \omega_X^k}{\text{opt}} \left[F(X), \bar{\lambda} \right] = \underset{x \in \omega_F^k}{\text{opt}} \left[F, \bar{\lambda} \right]. \quad (6.13)$$

Рассмотрим основные схемы компромисса, предполагая, что все локальные критерии нормализованы, т. е. все они имеют одинаковую размерность, либо являются безразмерными величинами (это ограничение будет снято). Кроме того, все локальные критерии имеют одинаковую важность (и это ограничение будет снято). И лучшим будет считаться большее значение локального критерия (и это ограничение тоже будет снято).

Необходимо отметить следующее обстоятельство: нет формальных правил выбора лучшей схемы компромисса, т. е. окончательное решение принимает человек.

Принцип равномерности. Он провозглашает целесообразным выбор такого варианта решения, принадлежащего области компромиссов, при котором достигалась бы некоторая «равномерность» показателей по всем локальным критериям.

Используются следующие реализации принципа равномерности: принцип равенства, принцип квазиравенства и принцип максимина (maxmin).

Принцип равенства. Он провозглашает целесообразность выбора такого варианта, при котором все значения локальных критериев равны между собой. Пусть интегральный критерий F задан табл. 6.1.

Таблица 6.1

Лок. кр. № вар.	f_1	f_2	F_3
1	f_{11}	f_{12}	F_{13}
2	f_{21}	f_{22}	F_{23}
3	f_{31}	f_{32}	F_{33}

1, 2, 3 — номера трех вариантов, один из которых надо выбрать; f_1, f_2, f_3 — некоторые локальные критерии (к примеру, f_1 — быстродействие, f_2 — объем оперативной памяти и т.д.); f_{13} — значение локального критерия f_3 для первого варианта и т.п.

Пусть, например, $f_{21} = f_{22} = f_{23}$, а остальные значения локальных критериев для первого и второго вариантов не равны между собой. Тогда вариант 2 признается лучшим.

В общем виде эта модель расписывается следующим образом:

$$\bar{F} = \operatorname{opt}_{x \in \omega_F^k} F = (f_1 = f_2 = \dots = f_k).$$

Принцип квазиравенства. Практически достичь равенства локальных критериев не удастся, тогда лучшим признается вариант, в котором локальные критерии наиболее близки к этому равенству.

Принцип максимина — maxmin. Формально он может быть записан с помощью следующей записи:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \max_{F \in \omega_F^k} \min_{1 \leq g \leq k} .$$

Иначе говоря, для каждого варианта выбирается минимальное значение локального критерия, и окончательный выбор останавливается на варианте, в котором этот минимум достигает своего максимума. В этом случае равномерность обеспечивается за счет «подтягивания» локального критерия с наименьшим значением показателя.

Принцип справедливой уступки. Он основан на сопоставлении прироста и убыли величин локальных критериев. Когда два или более вариантов находятся в области компромиссов (а только такие ситуации мы и рассматриваем), то при переходе от одного варианта к другому один (или несколько) локальный критерий может возрастать, другой (другие) может убывать. Данный принцип и основан на сопоставлении суммарной прибыли и суммарной убыли. Если суммарная прибыль превышает суммарную убыль, то новый вариант предпочтительнее старого, старый вариант отбрасывается и ведется сопоставление оставшегося варианта с новым вариантом. В том случае, если суммарная прибыль меньше суммарной убыли, то отбрасывается новый вариант, а старый вариант сравнивается со следующим вариантом. В том случае, если убыль равняется прибыли, то эти варианты равнозначны.

При этом сравнение может вестись как по абсолютному значению прибыли и убыли — тогда это принцип абсолютной уступки, либо по относительной величине прибыли и убыли — тогда это принцип относительной уступки.

Принцип абсолютной уступки. Формально он может быть выражен с помощью следующего выражения:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F} F = \{ \bar{F} \mid \sum_{j \in J^{(+)}} \Delta f_j \geq \sum_{j \in I^{(-)}} \Delta f_j \}.$$

В этом выражении $J^{(+)}$ — подмножество мажорируемых, т.е. увеличиваемых, критериев; $I^{(-)}$ — подмножество минорируемых, т.е. уменьшаемых, критериев. Причем, как следует из определения, $\Delta f_j > 0$, $\Delta f_i < 0$, Δf_j , Δf_i — абсолютное значение величин приращения, $|$ — символ «такой, при котором». Лучшим по принципу абсолютной уступки считается компромисс, при котором абсолютное значение суммы снижения одного или нескольких критериев не превышает абсолютного значения суммы приращений оставшихся критериев.

Пример: пусть, как и прежде, значения локальных критериев заданы табл. 6.1.

Сравниваем между собой первый и второй варианты. При переходе от первого варианта ко второму мы имеем: Δf_1 — приращение первого критерия, $\Delta f_1 = \Delta f_{21} - \Delta f_{11}$, пусть эта величина оказалась положительной $\Delta f_1 = \Delta f_{21} - \Delta f_{11} > 0$.

Сравниваем эти два варианта по второму критерию: $\Delta f_2 = \Delta f_{22} - \Delta f_{12}$, и пусть эта величина оказалась отрицательной $\Delta f_2 = \Delta f_{22} - \Delta f_{12} < 0$.

Сравниваем между собой эти два варианта по третьему критерию: $\Delta f_3 = \Delta f_{23} - \Delta f_{13}$, и пусть эта величина тоже оказалась отрицательной $\Delta f_3 = \Delta f_{23} - \Delta f_{13} < 0$.

Если выигрыш Δf_1 оказался больше проигрыша $\Delta f_2 + \Delta f_3$, то лучшим следует признать второй вариант, если $\Delta f_1 < \Delta f_2 + \Delta f_3$, то лучшим следует признать первый вариант — проиграли больше, чем выиграли.

После этого худший вариант отбрасывается и совершенно аналогично сравнение ведется выбранного варианта со следующим по порядку вариантом. Таким образом, необходимо просмотреть все варианты.

Можно показать, что принципу абсолютной уступки соответствует модель максимизации суммы локальных критериев:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \sum_{q=1}^k f_q \rightarrow \max,$$

т.е. ищется сумма по строкам всех локальных критериев:

$$\begin{aligned} & f_{11} + f_{12} + f_{13}, \\ & f_{21} + f_{22} + f_{23}, \\ & f_{31} + f_{32} + f_{33}, \end{aligned}$$

и та из этих сумм, которая окажется максимальной, соответствует лучшему варианту.

Недостатком принципа абсолютной уступки является то, что он чрезвычайно чувствителен к значению каждого критерия и поэтому большая величина одного критерия может «погасить» значения других критериев.

Принцип относительной уступки. Формально он может быть записан с помощью следующего выражения:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \{ \bar{F} \mid \sum_{j \in J^{(+)}} x_j \geq \sum_{j \in J^{(-)}} x_j \},$$

где $x_j = \frac{\Delta f_j}{f_{j \max}}$, $x_i = \frac{\Delta f_i}{f_{i \max}}$ есть относительные значения приращения локальных критериев.

В каждом столбце находится максимальное значение локального критерия. После этого необходимо перейти к новой таблице, поделив все числа в столбцах предыдущей таблицы на максимальное значение в каждом столбце. Далее с этой новой таблицей выполняются те же процедуры, что и в принципе абсолютной уступки.

Можно показать, что при использовании этого компромисса оптимальному варианту соответствует модель максимизации произведения локальных критериев, формально:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \prod_{q=1}^k f_q \rightarrow \max.$$

Или, для случая трех критериев: для каждой строчки вычисляется произведение:

$$\begin{aligned} f_{11} \cdot f_{12} \cdot f_{13}, \\ f_{21} \cdot f_{22} \cdot f_{23}, \\ f_{31} \cdot f_{32} \cdot f_{33}, \end{aligned}$$

и среди этих произведений ищется максимум — это и будет лучший вариант по данному компромиссу.

Достоинством этого компромисса является то, что он не требует предварительной нормализации критериев. Эта нормализация осуществляется автоматически за счет деления на максимальные значения в каждом столбце.

Принцип выделения одного оптимизируемого критерия. Этот принцип является самым простейшим: один из локальных критериев объявляется главным и только по нему ищется наилучшее решение. На остальные локальные критерии могут накладываться (или не накладываться) ограничения.

Формально этот принцип может быть записан следующим образом:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \max f_i; \quad f_q > \text{ или } < f_{q \text{ доп}}, \quad i \neq q.$$

Принцип последовательной уступки. Пусть теперь локальные критерии имеют различную важность и пусть, также, самым важным является критерий f_1 , вторым по важности является критерий f_2 , третьим — f_3 и т. д.

Сначала отыскивается вариант, обращающий критерий f_1 в максимум. После этого, исходя из некоторых соображений (например, из точности, с которой мы знаем значение f_1), на критерий f_1 накладывается некоторая «уступка» Δf_1 и при ограничении $f_{1 \max} - \Delta f_1$ выбирается вариант, обращающий в максимум второй по важности критерий f_2 . Совершенно аналогично на критерий f_2 может быть наложена «уступка» Δf_2 , и при соблюдении условий

$$\left. \begin{aligned} f_{1 \max} - \Delta f_1 \\ f_{2 \max} - \Delta f_2 \end{aligned} \right\}$$

выбирается вариант, обращающий в максимум следующий по важности критерий f_3 , и т. д.

Способ хорош тем, что здесь отчетливо видно, ценой какой уступки в одном критерии можно получить выигрыш в других критериях.

Пример. Пусть имеется табл. 6.2.

Таблица 6.2

Лок. кр. № вар.	f_1	f_2	f_3	Оставшиеся варианты
1	21	14		+
2	13	10		+
3	30	10		+
4	7	40		-
5	28	13		+

$f_{1 \max} = 30$. Допустим, что мы согласны допустить уступку $f_1 = 20$. Тогда, при условии $f_1 \geq 30 - 20 = 10$, просматриваем варианты по первому критерию. В столбце «оставшиеся варианты» знаком «-» отмечен отбрасываемый вариант. Среди оставшихся вариантов находится лучший вариант по второму критерию, стало быть, выбираем первый вариант.

Свертка локальных критериев. Весь вышеизложенный материал относился к случаю, когда лучшими считались большие значения локальных критериев, или, иначе говоря, решалась задача максимизации интегрального критерия.

Пусть теперь лучшими будут считаться меньшие значения всех локальных критериев, т.е. пусть необходимо решать задачу минимизации интегрального критерия.

В этом случае исходную таблицу (все ее члены) необходимо умножить на -1 и, используя весь предыдущий аппарат, отыскивать максимум $F = -F$.

Пусть теперь ряд локальных критериев необходимо максимизировать, а оставшиеся критерии необходимо минимизировать. В этом случае для выбора наилучшего варианта можно использовать любое из двух следующих соотношений:

$$\bar{F} = \text{opt}_{F \in \omega_F^k} F = \max_{F \in \omega_F^k} \left\{ \sum_{q=1}^l t_q + \left(\sum_{q=l+1}^k t_q \right)^{-1} \right\}, \quad (6.14)$$

$$\bar{F} = \text{opt}_{F \in \omega_F^k} F = \max_{F \in \omega_F^k} \left\{ \prod_{q=1}^l t_q \times \left(\prod_{q=l+1}^k t_q \right)^{-1} \right\}. \quad (6.15)$$

В этих соотношениях $f_q, q = \overline{1, l}$ — локальные критерии, которые необходимо максимизировать; $f_q, q = \overline{l+1, k}$ — локальные критерии, которые необходимо минимизировать.

Соотношение (6.14) соответствует принципу абсолютной уступки, соотношение (6.15) — принципу относительной уступки.

Способы нормализации локальных критериев. Проблема нормализации локальных критериев возникает во всех задачах векторной оптимизации, когда локальные критерии имеют различные единицы измерения (килограммы, метры, секунды, рубли и т. д.). Исключение составляет лишь метод относительной уступки, в котором нормализация осуществляется автоматически. В основу нормализации положено понятие «идеального вектора», т. е. вектора с идеальными значениями локальных критериев

$$F_u = (f_{1u}, f_{2u}, \dots, f_{ku}).$$

В нормализованном пространстве критериев вместо действительного значения локального критерия f_q рассматривается безразмерная величина $f_{q, n}$,

$$f_{q, n} = \frac{f_q}{f_{q, u}},$$

— действительная величина, поделенная на идеальную величину.

В том случае, если лучшим считается большее значение критерия и если

$$f_{q, u} \neq 0, \quad \text{то } f_{q, n} \in (0, 1).$$

Успешное решение проблемы нормализации во многом определяется тем, насколько удачным окажется выбор параметров идеального вектора. Мы рассмотрим три основных способа задания идеального вектора.

1-й способ. Идеальный вектор определяется некоторыми заданными значениями локальных критериев. Эти заданные значения может определить, например, заказчик разработки. Формальная запись:

$$F_u = F_{\text{зад}} = \{f_{q, \text{зад}}\}, \quad q = \overline{1, k}.$$

Недостаток этого способа — полнейший субъективизм выбора.

2-й способ. Идеальным считается вектор, параметрами которого являются максимально возможные значения локальных критериев.

$$F_u = F_{\text{max}} = \{f_{q, \text{max}}\}, \quad q = \overline{1, k}.$$

3-й способ. В качестве параметров идеального вектора принимается максимально возможный разброс значений соответствующих локальных критериев, т.е.

$$F_u = F_{q, \max} - F_{q, \min} = \{f_{q, \max} - f_{q, \min}\}, \quad q = \overline{1, k}.$$

Необходимо отметить, что нет формальных способов по выбору способа задания идеального вектора.

Способы задания и учета приоритета локальных критериев. Обычно используются три способа задания: с помощью ряда приоритета, вектора приоритета и весового вектора.

Ряд приоритета $\bar{R} = (1, 2, 3, \dots, k)$ указывает на то, что локальные критерии, записанные в скобках левее, более важны, чем локальные критерии, записанные правее, т.е. самым важным является критерий f_1 , вторым по важности является критерий f_2 , затем f_3 и т.д.

В том случае, если имеются равноприоритетные критерии, они выделяются скобками $\bar{R} = \{1, 2, (3, 4, 5), \dots, k\}$, т.е. критерии 3, 4, 5 — равноприоритетны и занимают третье место по важности. Это чисто качественный способ задания приоритетов. При таком способе обычно используется принцип «жесткого приоритета», т.е. не допускается ни малейшего снижения критерия, стоящего левее в ряду приоритета.

Вектор приоритета $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_k)$ — это способ количественного задания приоритетов. Компоненты этого вектора λ_q определяют степень относительного превосходства двух соседних критериев из ряда приоритета, т.е. λ_q определяет, во сколько раз критерий f_q важнее критерия f_{q+1} , в том случае, если f_q и f_{q+1} равны по важности, то, стало быть, и $\lambda_q = 1$. Для удобства λ_q всегда равно единице ($\lambda_k = 1$).

Весовой вектор $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_k)$ представляет собой k -мерный вектор, компоненты которого связаны соотношениями:

$$0 \leq \alpha_q \leq 1, \quad \sum_{q=1}^k \alpha_q = 1, \quad q = \overline{1, k}.$$

Компонента α_q показывает степень относительного превосходства критерия f_q над всеми оставшимися критериями. Обычно если необходимо количественно задавать приоритет критериев, то его задают в виде ряда приоритета, поскольку там сравнение идет только между двумя соседними критериями; затем с помощью соотношения:

$$\alpha_q = \frac{\prod_{i=1}^k \lambda_i}{\sum_{i=1}^k \prod_{i=1}^k \lambda_i} \quad (6.16)$$

переходят к весовому вектору. И тогда выбор наилучшего варианта производится с помощью всего вышеописанного аппарата, только вместо компонент вектора $\{f_1, f_2, \dots, f_k\}$ используются компоненты $\{\alpha_1 f_1, \alpha_2 f_2, \dots, \alpha_k f_k\}$. Такой подход называют *принципом гибкого приоритета*.

Для случая 3-х локальных критериев соотношения (6.16) переписываются в виде

$$\alpha_1 = \frac{\lambda_1 \lambda_2 \lambda_3}{A}, \quad \alpha_2 = \frac{\lambda_2 \lambda_3}{A}, \quad \alpha_3 = \frac{\lambda_3}{A},$$

где $A = \lambda_1 \lambda_2 \lambda_3 + \lambda_2 \lambda_3 + \lambda_3$.

Пример многокритериальной задачи принятия решений. Для отдела ЦКБ необходимо устройство для вывода на печать конструкторских чертежей (плоттер). Имеются плоттеры трех моделей. Каждая модель характеризуется тремя локальными критериями: максимально возможный формат отпечатанного чертежа F (мм), разрешение чертежа R (dpi) и объем буфера V (КБайт). Конкретные значения указанных локальных критериев для каждого из вариантов представлены в таб. 6.3.

Таблица 6.3

Крит. № вар.	F	R	V
1	4	20	64
2	8	14	128
3	10	12	256

Требуется, используя известные схемы компромисса, определить лучший вариант плоттера:

- без учета приоритета локальных критериев;
- с учетом приоритета локальных критериев.

1. Нормализация исходных данных. Поскольку локальные критерии имеют различную размерность, прежде всего необходи-

мо нормализовать исходную таблицу 6.3. Для этого используется соотношение

$$f_{q, \text{норм}} = \frac{f_q}{f_{q, \text{ид}}}.$$

В качестве параметров идеального вектора выбираем максимально возможные значения параметров:

$$F_{\text{max}} = 10; \quad R_{\text{max}} = 20; \quad V_{\text{max}} = 256.$$

Переходим от табл. 6.3 к табл. 6.4, в которой записаны нормализованные значения локальных критериев.

Таблица 6.4

№ вар. \ Крит.	F	R	V
1	0,4	1	0,25
2	0,8	0,7	0,5
3	1	0,6	1

2. Выбор наилучшего варианта без учета приоритета локальных критериев.

Воспользуемся известными схемами компромисса.

2.1. Принцип равномерности.

Известны три модификации этого принципа.

Принцип равенства.

Формально он записывается следующим образом:

$$\bar{F} = \underset{q \in \omega_F^k}{\text{opt}} F = (f_1 = f_2 = \dots = f_k),$$

где \bar{F} — оптимальный вариант, принадлежащий области компромиссов ω_F^k , а f_q — q -тый локальный критерий, $q = \overline{1, k}$.

Вывод: случай, когда все локальные критерии равны между собой, отсутствует, поэтому эта модификация применена быть не может.

Принцип квазиравенства.

Выбираем вариант, у которого в максимальной степени локальные критерии равны между собой.

Вывод: по принципу квазиравенства предпочтение следует отдать второму варианту.

Принцип максимина.

Формально он записывается следующим образом:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \max_{F \in \omega_F^k} \min_{1 \leq q \leq k},$$

где \bar{F} — оптимальный вариант, принадлежащий области компромиссов ω_F^k , а f_q — q -тый локальный критерий, $q = \overline{1, k}$.

Рассматривая таблицу построчно, необходимо в каждой строке выделить минимальное значение критерия:

1-я строка — 0,25; 2-я строка — 0,5; 3-я строка — 0,6 (max).

Просматривая выделенные числа среди них находим максимальное значение, оно соответствует варианту №3.

Вывод: если воспользоваться принципом максимина, то предпочтение следует отдать варианту №3.

2.2. Принцип справедливой уступки.

Известны две модификации этого принципа.

Принцип абсолютной уступки.

Формально он записывается следующим образом:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \{ \bar{F} \mid \sum_{j \in J^{(+)}} \Delta f_j \geq \sum_{j \in I^{(-)}} \Delta f_j \}.$$

В этом выражении $J^{(+)}$ — подмножество мажорируемых, т.е. увеличиваемых критериев; $I^{(-)}$ — подмножество минорируемых, т.е. уменьшаемых критериев. Причем, как следует из определения, $\Delta f_j > 0$, $\Delta f_i < 0$, Δf_j , Δf_i — абсолютное значение величин приращения, | — символ «такой, при котором».

Лучшим, по принципу абсолютной уступки, считается компромисс, при котором значение суммы снижения одного или нескольких критериев не превышает абсолютного значения суммы приращений оставшихся критериев.

Можно показать, что этому принципу соответствует модель максимизации суммы локальных критериев:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \sum_{q=1}^k f_q \rightarrow \max,$$

т.е. ищется сумма по строкам всех локальных критериев:

$$\begin{aligned} & f_{11} + f_{12} + f_{13}, \\ & f_{21} + f_{22} + f_{23}, \\ & f_{31} + f_{32} + f_{33}, \end{aligned}$$

и та из этих сумм, которая окажется максимальной, соответствует лучшему варианту.

Воспользуемся этими соотношениями:

$$\begin{aligned} 1\text{-й вариант} &— 0,4 + 1 + 0,25 = 1,65; \\ 2\text{-й вариант} &— 0,8 + 0,7 + 0,5 = 2,0; \\ 3\text{-й вариант} &— 1 + 0,6 + 1 = 2,6 \quad (\text{max}). \end{aligned}$$

Просматривая значения полученных сумм, выбираем максимальное значение, это и будет наилучшим вариантом.

Вывод: если задаться принципом абсолютной уступки, то предпочтение следует отдать варианту три.

Принцип относительной уступки.

Формально он записывается с помощью следующего соотношения:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \{ \bar{F} \mid \sum_{j \in J^{(+)}} x_j \geq \sum_{j \in J^{(-)}} x_j \},$$

где $x_j = \frac{\Delta f_j}{f_{j \max}}$, $x_i = \frac{\Delta f_i}{f_{i \max}}$ есть относительные значения приращения

локальных критериев.

В каждом столбце находится максимальное значение локального критерия. После этого необходимо перейти к новой таблице, поделив все числа в столбцах предыдущей таблицы на максимальное значение в каждом столбце. Далее, с этой новой таблицей выполняются те же процедуры, что и в принципе абсолютной уступки. Можно показать, что при использовании этого компромисса оптимальному варианту соответствует модель максимизации произведения локальных критериев. Формально:

$$\bar{F} = \operatorname{opt}_{F \in \omega_F^k} F = \prod_{q=1}^k f_q \rightarrow \max.$$

Или, для случая трех критериев: для каждой строчки вычисляется произведение:

$$\begin{aligned} f_{11} \cdot f_{12} \cdot f_{13}, \\ f_{21} \cdot f_{22} \cdot f_{23}, \\ f_{31} \cdot f_{32} \cdot f_{33}, \end{aligned}$$

и среди этих произведений ищется максимум — это и будет лучший вариант по данному компромиссу.

Используя эти соотношения, получим:

$$\begin{aligned} 1\text{-й вариант} &— 0,4 \cdot 1 \cdot 0,25 = 0,1; \\ 2\text{-й вариант} &— 0,8 \cdot 0,7 \cdot 0,5 = 0,28; \\ 3\text{-й вариант} &— 1 \cdot 0,6 \cdot 1 = 0,6 \quad (\text{max}). \end{aligned}$$

Вывод: если в качестве схемы компромисса выбрать принцип относительной уступки, то предпочтение следует отдать варианту три.

2.3. Принцип выделения одного оптимизируемого критерия.

При выборе модели плоттера главным критерием является формат печатаемых материалов F , поэтому, просматривая первый столбец в табл. 6.4, необходимо найти максимальное значение соответствующего критерия:

$$0,4; 0,8; 1 \quad (\max).$$

Максимальное значение соответствует третьему варианту, это и будет оптимальным вариантом.

Вывод: оптимальным вариантом является третий при использовании принципа выделения одного оптимизируемого критерия.

2.4. Принцип последовательной уступки.

Пусть важность локальных критериев соответствует их последовательной записи в табл. 6.3, т. е. самым важным параметром является форма, менее важным — разрешение, и наименее важным — объём памяти. Просматриваем первый столбец табл. 6.4 и выбираем вариант, в котором F достигает максимума:

$$0,4; 0,8; 1 \quad (\max).$$

Пусть теперь мы согласны наложить «уступку» на критерий F и пусть эта «уступка» будет равна $\Delta F = 0,5$; тогда имеем $F_{1, \max} - \Delta f_1 = 1 - 0,5 = 0,5$; таким образом мы допускаем к рассмотрению все варианты, в которых F не хуже, чем 0,5. От табл. 6.4 переходим к табл. 6.5, где отображено это условие.

Таблица 6.5

№ вар. \ Крит.	F	R	V
1	—	—	—
2	0,8	0,7	0,5
3	1	0,6	1

Из рассмотрения исключается первый вариант.

Теперь в табл. 6.5 выбираем наилучший вариант по критерию R — это 0,7. Таким образом, при наложении уступки на критерий F равный 0,5, мы пришли к выбору варианта два.

Совершенно аналогично можно наложить уступку и на второй по важности критерий. Мы не будем этого делать.

Вывод: при использовании принципа последовательной уступки выбирается второй вариант.

3. Выбор наилучшего варианта с учетом приоритета локальных критериев.

Пусть приоритет критериев задан следующим вектором $\bar{\lambda} = (1, 2, 4)$. Переходим к весовому вектору:

$$A = \lambda_1 \lambda_2 \lambda_3 + \lambda_2 \lambda_3 + \lambda_3 = 8 + 8 + 4 = 20;$$

$$\alpha_1 = \frac{\lambda_1 \lambda_2 \lambda_3}{A} = \frac{8}{20} = 0,4;$$

$$\alpha_2 = \frac{\lambda_2 \lambda_3}{A} = \frac{8}{20} = 0,4;$$

$$\alpha_3 = \frac{\lambda_3}{A} = \frac{4}{20} = 0,2.$$

Имеем:

$$\bar{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0,4; 0,4; 0,2).$$

И от табл. 6.4 переходим к табл. 6.6, в которую подставляем вместо $f_i \rightarrow \alpha_i f_i$.

Таблица 6.6

№ вар.	Крит.		
	F	R	V
1	0,16	0,4	0,1
2	0,32	0,28	0,2
3	0,2	0,12	0,2

Теперь воспользуемся известными схемами компромисса.

3.1. Принцип равномерности.

Принцип равенства.

Случай, когда все локальные критерии равны между собой, отсутствует, поэтому эта модификация применена быть не может.

Принцип квазиравенства.

Выбираем вариант, у которого локальные критерии в максимальной степени равны между собой.

Вывод: по принципу квазиравенства предпочтение следует отдать второму варианту.

Принцип максимина.

Рассматривая табл.6.6 построчно, в каждой строке выделяем минимальное значение локального критерия:

1-я строка — 0,1; 2-я строка — 0,2 (max); 3-я строка — 0,12.

Просматривая эти числа, среди них находим максимальное значение, оно соответствует варианту два.

Вывод: если воспользоваться принципом максимина, то предпочтение следует отдать варианту два.

3.2. Принцип справедливой уступки.

Принцип абсолютной уступки.

Воспользуемся здесь и ниже материалом, изложенным в первой части задачи.

$$\begin{aligned} 1\text{-й вариант} & - 0,16 + 0,4 + 0,1 = 0,66; \\ 2\text{-й вариант} & - 0,32 + 0,28 + 0,2 = 0,8; \quad (\max); \\ 3\text{-й вариант} & - 0,2 + 0,12 + 0,2 = 0,52. \end{aligned}$$

Просматривая значения полученных сумм, выбираем максимальное значение, это и будет наилучшим вариантом.

Вывод: если задаться принципом абсолютной уступки, то предпочтение следует отдать варианту два.

Принцип относительной уступки.

$$\begin{aligned} 1\text{-й вариант} & - 0,16 \cdot 0,4 \cdot 0,1 = 0,0064; \\ 2\text{-й вариант} & - 0,32 \cdot 0,28 \cdot 0,2 = 0,01792; \quad (\max); \\ 3\text{-й вариант} & - 0,2 \cdot 0,12 \cdot 0,2 = 0,0048. \end{aligned}$$

Просматривая полученные значения, выбираем максимальное значение, это и будет наилучшим вариантом.

Вывод: если в качестве принципа выбрать принцип относительной уступки, то предпочтение следует отдать варианту два.

3.3. Принцип выделения одного оптимизируемого критерия.

При выборе модели плоттера главным критерием является формат печатаемых материалов F , поэтому, просматривая первый столбец в табл.6.6, необходимо найти максимальное значение:

$$0,16; 0,32 (\max); 0,2.$$

Максимальное значение соответствует второму варианту, это и будет оптимальным вариантом.

Вывод: оптимальным вариантом является второй при использовании принципа выделения одного оптимизируемого критерия.

3.4. Принцип последовательной уступки.

Пусть важность локальных критериев соответствует их последовательной записи в табл.6.6, т.е. важнейшим параметром является форма, менее важным — разрешение, и наименее важным — объём памяти. Просматриваем первый столбец табл. 6.6 и выбираем вариант, в котором F достигает максимума:

$$0,16; 0,32; (\max); 0,2.$$

Пусть теперь мы согласны наложить «уступку» на критерий F , и пусть эта «уступка» будет равна $\Delta F = 0,15$, тогда имеем $F_{1, \max} - \Delta f_1 = 0,32 - 0,15 = 0,17$, таким образом, мы допускаем к рассмотрению все варианты, в которых F не хуже, чем $0,17$. От табл. 6.6 переходим к табл. 6.7, где отображено это условие:

Таблица 6.7

№ вар. \ Крит.	F	R	V
1	—	—	—
2	0,32	0,28	0,2
3	0,2	0,12	0,2

Из рассмотрения исключается первый вариант. Теперь в табл. 6.7 выбираем наилучший вариант по критерию R , это 0,28. Таким образом, при наложении уступки на критерий F , равный 0,17, мы пришли к выбору второго варианта.

Вывод: при использовании принципа последовательной уступки выбирается второй вариант.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Рассмотрите основные условия, в которых осуществляется выбор решения.
2. Перечислите основные группы ограничивающих факторов.
3. Что такое внешние и внутренние параметры?
4. Рассмотрите математическую модель оптимального проектирования.
5. Расскажите, как реализуется процесс принятия решений.
6. Сформулируйте общую постановку задачи принятия решений.
7. Приведите классификацию задач принятия решений.
8. В чем сущность однокритериальной задачи принятия решений?
9. Как могут приниматься решения в условиях риска?
10. В чем отличия решений в условиях риска и в условиях неопределенности?
11. Рассмотрите основные критерии оптимальности выбора решений в условиях неопределенности.
12. В чем сущность многокритериальной задачи принятия решений?
13. Что представляет собой принцип равномерности?
14. В чем заключается принцип справедливой уступки?
15. В чем отличия принципа абсолютной уступки от принципа относительной уступки?
16. Поясните принцип выделения одного оптимизируемого критерия.
17. Что представляет собой принцип последовательной уступки?
18. Что такое свертка локальных критериев?
19. Рассмотрите способы нормализации локальных критериев.
20. Рассмотрите способы задания и учета приоритета локальных критериев.
21. Что представляет собой ряд приоритета?
22. Что такое вектор приоритета? Как задается весовой вектор?
23. Каким образом обычно количественно задается приоритет критериев?
24. Чем отличаются друг от друга «жесткий» и «гибкий» приоритеты?

Раздел 2

ПРИКЛАДНЫЕ ВОПРОСЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

Глава 7

ВИДЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

Всякое управление предполагает наличие объекта управления и управляющего органа. Объекты управления чрезвычайно разнообразны: предприятие, сложная технологическая установка, отрасль народного хозяйства, живой организм или его часть, учебное заведение, город, область, республика. Управление представляет собой сложный информационный процесс. Желательно, чтобы управление осуществлялось наилучшим образом, было оптимальным.

В этой главе рассматриваются различные виды управления: централизованное, децентрализованное и иерархическое. Также анализируются типовые организационные структуры управления производством.

Так как в контуре управления сложной системы обязательно наличие человека (т. е. сложная система всегда является эргатической, или человеко-машинной системой), то на человека возлагается часть функций управления, причем наиболее ответственных. Ибо только человек способен учесть при управлении системой такие неформализуемые пока факторы, как социальные, психологические, моральные и физиологические.

7.1. Централизованное и децентрализованное управление

Централизованный вид управления предполагает реализацию всех процессов управления объектами в едином центральном органе управления. Этот орган собирает информацию о состоянии всех объектов управления, осуществляет ее обработку и каждому объекту управления выдает свою собственную управляющую команду (рис. 7.1). Управляющий орган воздействует на объект управления посредством выдачи команд. Эту информацию называют *распорядительной*. Поскольку органу управления не безразлично

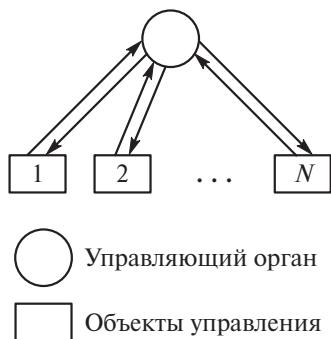


Рис. 7.1. *Централизованная структура управления*

поведение объекта управления, всегда присутствует обратная связь — это *осведомительная* информация. Она сообщает органу управления о состоянии объекта управления.

Достоинства такой структуры:

- создается принципиальная возможность реализации глобально-оптимального управления системой в целом, поскольку каждое управляющее воздействие вырабатывается на основе всей информации о системе;
- достаточно просто реализуются процессы информационного взаимодействия;
- исключается необходимость в пересылках промежуточных результатов процессов управления;
- достаточно легко осуществляется корректировка оперативно изменяющихся данных;
- возникает возможность достижения максимальной эксплуатационной эффективности при минимальной избыточности технических средств.

С системотехнической точки зрения основными недостатками структуры с централизованным управлением являются:

- необходимость для органа управления собирать, запоминать и обрабатывать чрезвычайно большие объемы информации;
- необходимость наличия запоминающих устройств очень большого объема;
- необходимость использования вычислительных средств очень высокой производительности;
- чрезвычайно высокие требования по надежности ко всем элементам технического обеспечения и ко всем элементам программного обеспечения, потому что выход из строя любого элемента приводит к выходу из строя всей системы;
- высокая суммарная протяженность и перегруженность каналов связи при наличии территориально разнесенных объектов управления.

В централизованном виде управления существует еще одна разновидность управления — так называемая *централизованная рассредоточенная структура*.

В рассматриваемой структуре (рис. 7.2) сохраняются все свойства и особенности принципа централизованного управления, а

именно: выработка управляющего воздействия на каждый объект управления осуществляется на основе информации о состоянии всей совокупности объектов управления. Однако в отличие от полностью централизованной структуры она не имеет четкой локализации в едином управляющем органе.

Алгоритмы управления в этом случае состоят из совокупности взаимосвязанных алгоритмов управления объектами.

В процессе функционирования каждый управляющий орган производит прием и обработку соответствующей информации, а также выдачу управляющих команд на подчиненные объекты. Кроме того, каждый локальный орган управления по мере необходимости вступает в процесс информационного взаимодействия с другими органами управления.

Достоинства такой структуры:

- снижение требований к объемам памяти, производительности и надежности каждого управляющего органа без снижения качества управления;
- суммарная протяженность и стоимость каналов связи в такой системе может быть снижена.

К недостаткам следует отнести:

- усложненность информационных процессов вследствие необходимости обмена данными между центрами обработки;
- сложность корректировки хранимой в памяти системы информации;
- значительная избыточность технических средств и, следовательно, повышение расходов на их приобретение, монтаж и эксплуатацию;
- сложность синхронизации процессов обмена информацией.

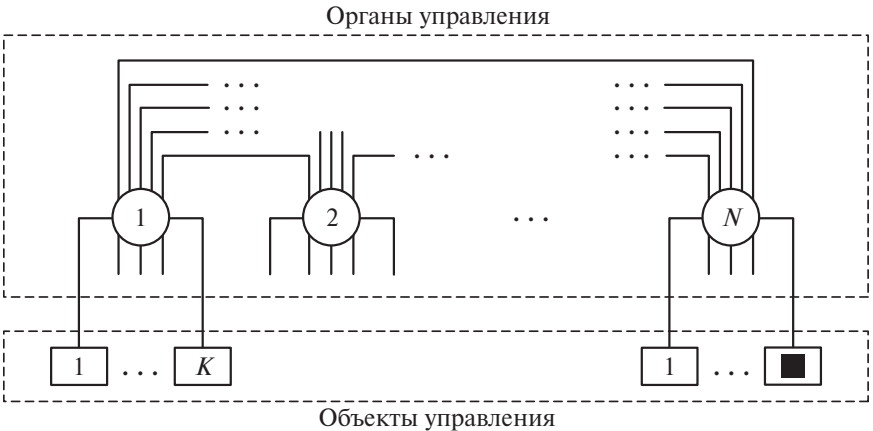


Рис. 7.2. Централизованная рассредоточенная структура управления

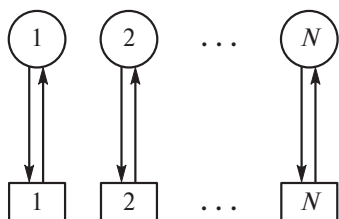


Рис. 7.3. Децентрализованная структура управления

Децентрализованное управление предполагает распределение функций управления по отдельным элементам сложной системы (рис 7.3).

Построение системы с такой структурой возможно только в случае независимости объектов управления по материальным, энергетическим, информационным и иным ресурсам. Для выработки управляющего воздействия на каждый объект необходимо

информация только о состоянии этого объекта и безразлична информация о состоянии других объектов.

Фактически децентрализованная структура представляет собой совокупность нескольких независимых систем, каждая из которых обладает своей информационной, алгоритмической, технической и прочими базами.

Децентрализованное управление свободно от многих недостатков чисто централизованного управления. Однако по сравнению с централизованным у децентрализованного управления имеются очевидные недостатки. К их числу относятся:

- возможная разность управления при одинаковых объектах управления;
- высокая стоимость системы управления;
- сокращение перерабатываемой информации может привести к снижению качества управления.

Поэтому в сложных системах управления, как правило, относится к промежуточному типу, когда степень централизации находится между двумя крайними случаями чисто централизованного и децентрализованного управления.

7.2. Иерархическое управление

В науке и технике известны случаи, когда из двух различных принципов возможен третий, в котором исключаются недостатки обоих принципов и объединяются только преимущества. К такому случаю относится иерархическое управление, в котором наличие суммирование преимуществ как централизованного, так и децентрализованного управления и отсутствие их недостатков.

Понятие «иерархия» (греч. *hierarchia*, от *hieros* — священный и *arche* — власть) — это расположение частей и элементов целого в порядке от высшего к низшему. Термин «иерархия» был введен

во второй половине V в. афинским епископом Дионисием Ареопагитом в трактатах «О небесной иерархии» и «О церковной иерархии» и до XIX в. употреблялся для характеристики организации христианской церкви. В науке понятие «иерархия» стало разрабатываться со второй половины XIX в. и применялось для описания сословно-классового деления общества и характеристики структуры власти, в особенности бюрократии. В XX в. с появлением общей теории систем понятие «иерархия» стало применяться для описания любых системных объектов. Иерархически организованные формы существуют во всех сферах объективной реальности: неорганической, биологической, социальной.

В общей теории организации иерархия характеризует принцип управления, обеспечивающий эффективное функционирование организации; в лингвистике различают иерархии уровней языка; в теории графов — иерархически построенный граф («дерево»).

Иерархический принцип управления — это принцип построения многоступенчатых систем управления, при котором функции управления распределяются между соподчиненными частями системы. В соответствии с иерархическим принципом обобщенные управляющие сигналы подсистем высшего уровня поступают для управления подчиненными подсистемами и, наоборот, конкретные осведомительные и задающие сигналы низших уровней иерархии используются для формирования управляющих сигналов вышестоящих.

Идея иерархической структуры представлена на рис. 7.4.

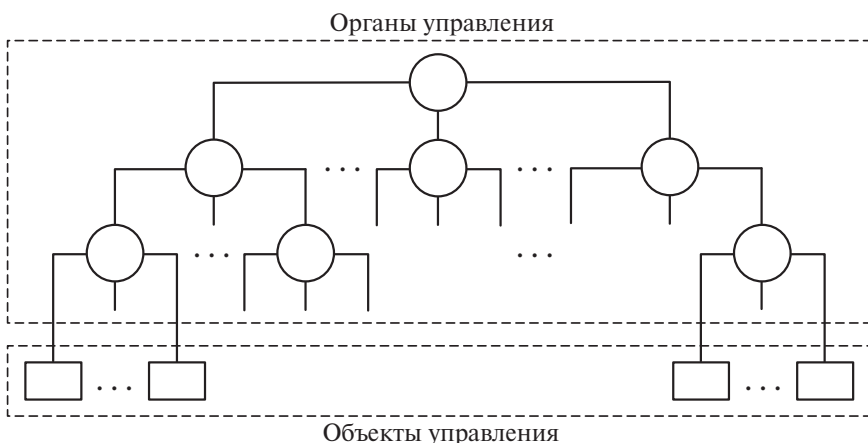


Рис. 7.4. Иерархическая структура

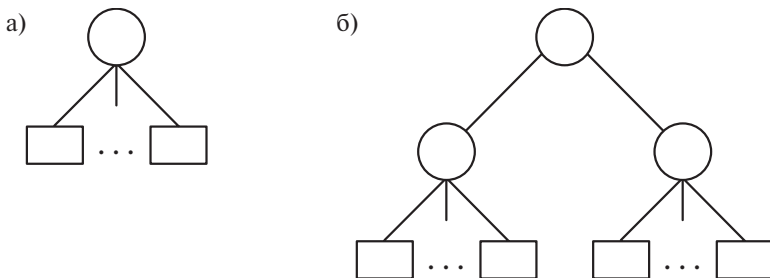


Рис. 7.5. Суть иерархического управления

С ростом масштабов объектов управления чрезвычайно увеличивается необходимый объем перерабатываемой информации и сложность алгоритмов управления. В результате управлять централизованным способом становится уже невозможно, хотя именно такое управление обеспечивает наилучшую согласованность принимаемых решений. Так возникает необходимость разделения функций и органов управления. Такое разделение, позволяя справиться с информационными трудностями каждого локального органа управления, порождает необходимость согласования принимаемых этими органами решений, т. е. порождает необходимость создания нового органа управления более высокого уровня.

Таким образом, идея иерархического управления сводится к следующему: пусть мы имеем централизованную систему, в которой орган управления не справляется с задачами управления (рис.7.5, а). Тогда для его разгрузки на этот же уровень добавляется еще один орган управления. Для согласования принимаемых ими решений на более высокий уровень управления добавляется еще один новый орган (рис 7.5, б).

Так, по мере роста сложности управления выстраивается иерархическая структура. Поэтому главной причиной иерархического управления является несоответствие между сложностью управляемого объекта и способностью управляющего органа получать и обрабатывать информацию.

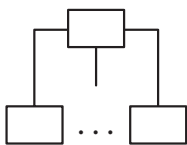


Рис. 7.6. Иерархия управляемой части

Кроме того, применительно к промышленным организационным системам объекты управления могут иметь собственную иерархию, которая чаще всего не совпадает с иерархией управляющей части (рис. 7.6). Иерархическая организация таких

систем возникла не из потребностей управления, а под влиянием объективных тенденций научно-технического прогресса. Такими тенденциями являются:

- рост сложности выпускаемых изделий;
- рост сложности технологии производства;
- рост скорости смены номенклатуры выпускаемых изделий;
- увеличение числа экономических единиц, занятых производством одного и того же изделия.

Это объективные системообразующие факторы, определяющие иерархию производственных объектов в АС. Управляющие процессы в сложных системах требуют своевременного формирования правильных решений, которые, во-первых, приводили бы к поставленным целям, во-вторых, были бы взаимно согласованы, в-третьих, были бы своевременны. Каждое такое решение требует постановки соответствующих задач управления. Их совокупность образует иерархию задач управления, которая в ряде случаев значительно сложнее иерархии объекта управления. Решение этой иерархии задач управления является главной функцией органов управления.

Таким образом, в организационных иерархических системах имеет место:

- иерархия объектов управления;
- иерархия управляющей части, которая, в свою очередь, содержит:
 - а) иерархию органов управления,
 - б) иерархию задач управления.

Всем организационным иерархическим системам присущи следующие основные особенности:

- последовательное вертикальное расположение подсистем, или, иначе говоря, *вертикальная соподчиненность*;
- *приоритет действий*, или, иначе говоря, право вмешательства подсистем верхнего уровня в действия подсистем нижнего уровня;
- *зависимость действий подсистем верхнего уровня* от фактического исполнения подсистемами нижнего уровня своих функций;
- элементы верхнего уровня имеют дело с более широкими аспектами поведения системы в целом, и, следовательно, они не могут реагировать на такие изменения в окружающей среде или в самом управляемом объекте, с которыми имеют дело элементы более низких уровней;
- чем выше уровень управления, тем больший период времени требуется для формирования управляющих воздействий. Это связано с тем, что степень неопределенности ин-

формации о состоянии системы повышается с номером уровня. Степень неопределенности зависит от непрерывно изменяющегося состояния внешней среды, от стохастического характера функционирования системы, от индивидуальных характеристик людей, участвующих в процессе управления, от многозначности критериев оценки качества функционирования. В связи с этим за малый период управления не успевает накопиться необходимое количество информации, достаточное для формирования правильного управляющего воздействия. Поэтому если длительность периода мала, то управляющая часть может реализовать только упрощенные управляющие алгоритмы, соответствующие более грубым методам переработки информации и принятию управляющих решений;

- чем ниже уровень управления, тем меньше эта неопределенность, и, следовательно, тем меньший период времени требуется для выработки правильных управляющих воздействий.

7.3. Основные типы иерархий

Определение многоуровневой иерархической структуры. Понятие многоуровневой иерархической структуры нельзя определить одной краткой и сжатой формулировкой. Как часто бывает в науке, дадим определение «многоуровневая иерархическая структура» перечислением присущих ей наиболее существенных свойств. К ним относятся следующие:

- вертикальная соподчиненность;
- право вмешательства;
- взаимозависимость действий.

Вертикальная соподчиненность означает, что любая иерархия состоит из последовательно расположенных вертикально взаимодействующих подсистем, составляющих данную систему (вертикальная декомпозиция), как показано на рис. 7.7. Заметим, что как входы X , так и выходы Y могут быть распределены по всем уровням, хотя чаще всего обмен со средой происходит на более низком уровне. Взаимодействие между уровнями не обязательно происходит только между каждыми двумя близлежащими уровнями, как для простоты показано на рис. 7.7.

Право вмешательства, или приоритет действий подсистем верхнего уровня, носит для нижележащих уровней обязывающий характер. В системах с детерминированным алгоритмом выполнения работы вмешательство обычно проявляется в виде изменения пара-

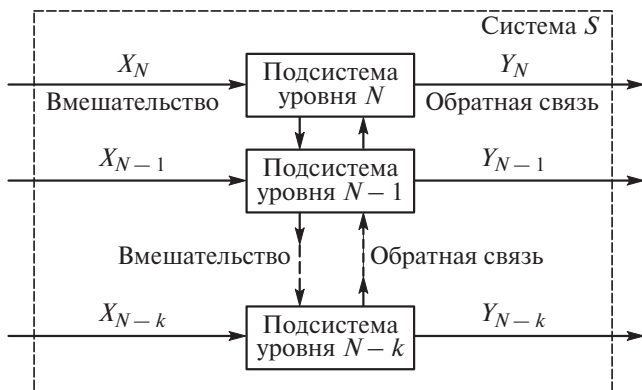


Рис. 7.7. Вертикальное взаимодействие между уровнями иерархий:
 X — входы, Y — выходы

метров подсистем нижележащего уровня. В системах же с недетерминированным алгоритмом выполнения приоритет действий задает последовательный порядок получения решений на разных уровнях. Чтобы подчеркнуть значение приоритета в установлении порядка действий, элементы верхнего уровня называют вышестоящими (*supremel*) и нижнего — нижестоящими (*infimal*).

Взаимозависимость действий вышестоящих и нижестоящих уровней структуры. Хотя вмешательство (приоритет действий) направлено сверху вниз в виде отдачи приказов или команд, успешность функционирования системы и фактически элементов любого уровня зависит от поведения всех элементов системы. Так как само понятие приоритета подразумевает, что вмешательство предшествует действиям более низких уровней, успешность работы верхнего уровня зависит не только от осуществляемых им действий, но и от соответствующих реакций нижних уровней, точнее от их суммарного эффекта. Поэтому можно считать, что качество работы всей системы обеспечивается обратной связью, т. е. реакциями на вмешательство, информация о которых направляется снизу вверх. Такая взаимозависимость действий особенно очевидна в случае, когда обмен со средой происходит в основном (или исключительно) на самом нижнем уровне системы.

Для классификации иерархий вводятся три понятия уровней: 1) уровень описания, или абстрагирования; 2) уровень сложности принимаемого решения; 3) организационный уровень. Чтобы различать эти уровни, употребляются следующие термины: «страта», «слой» и «эшелон». Термин «уровень» является родовым, относящимся к любому из этих понятий. При описании реальных иерар-

хических систем могут использоваться все три понятия; случаи, когда применяется только одно, являются исключением.

Страты. Уровни описания, или абстрагирования. Напомним, что основная дилемма при описании систем состоит в том, что приходится находить компромисс между простотой описания и необходимостью учета многочисленных факторов и характеристик сложной системы. Разрешение этой дилеммы ищется в иерархическом описании. Система задается семейством моделей, каждая из которых описывает поведение системы с точки зрения различных уровней абстрагирования. Для каждого уровня иерархии существует ряд характерных особенностей и переменных, законов и принципов, с помощью которых описывается поведение системы. Чтобы такое иерархическое описание было эффективным, необходима как можно большая независимость моделей для различных уровней системы. Таким образом, процесс расчленения системы на уровни, характеризующие технологические, информационные и экономические аспекты ее функционирования, называется *стратифицированием системы*, а сами уровни — *стратами*. На каждой страте в иерархии структур имеется свой собственный набор переменных, которые позволяют в значительной степени ограничить изучение только одной стратой. Независимость страт открывает возможность для более глубокого и детального изучения системы, однако предположение о независимости страт может привести к неполному пониманию поведения системы в целом. Приведем несколько примеров систем, требующих стратифицированного описания.

1. *Модель ЭВМ.* Функционирование ЭВМ обычно описывается не менее чем на двух стратах (рис. 7.8).

На первой страте система описывается на языке физических законов, управляющих работой и взаимодействием ее составных частей, в то время как на второй страте имеют дело с абстрактными нефизическими понятиями: такими, как двоичные разряды, файлы, информационные потоки и т. д. На страте физических законов представляет интерес правильное функционирование различных электронных компонентов, а на страте обработки информации — проблемы вычисления, программирования и т. п.

2. *Автоматизированный промышленный комплекс.* Этот комплекс обычно моделируют на трех стратах: 1) физические процессы обработки материалов и преобразования энергии; 2) управление в обработке информации; 3) экономика производства с точки зрения его производительности и прибыльности (рис. 7.9). Заметим, что на любой из этих трех страт мы имеем дело с одним предметом — основным физическим продуктом. На первой страте он рассматривается как физический объект, который подлежит обработке в соответс-

твии с физическими законами, на второй страте его рассматривают как управляемую переменную; на третьей страте это уже товар как экономическая категория. Для каждого из этих аспектов системы имеется свое описание, однако система остается одной и той же.

3. *Структура системы управления народным хозяйством страны.* Система управления народным хозяйством страны может быть

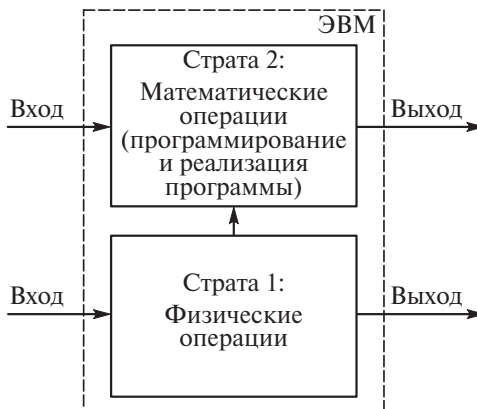


Рис. 7.8. Стратифицированное представление ЭВМ

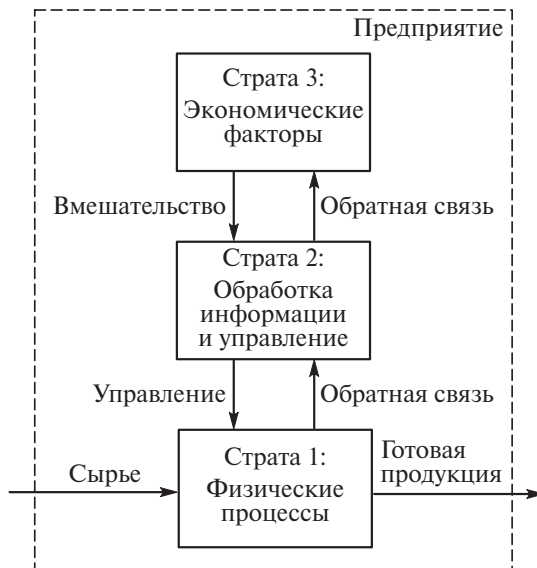


Рис. 7.9. Стратифицированное представление автоматизированного машинного производства

представлена в виде семи страт (рис. 7.10). Первые три нижние страты относятся к проблематике, связанной с решением задач автоматического или автоматизированного управления производством. На этих уровнях большую роль в процессе управления играют автоматические средства, а не человек. На остальных (верхних) уровнях осуществляется административное и организационное (планирование экономики) управление, и большее значение в процессе этого управления принадлежит людям, а не автоматическим устройствам.

Примеров стратифицированного описания систем можно привести очень много. Однако и приведенных примеров достаточно, чтобы сформулировать общие свойства стратифицированного описания систем:

1. Выбор страт, в терминах которых описывается данная система, зависит от наблюдателя, его знания и заинтересованности в деятельности системы, хотя для многих систем некоторые страты кажутся естественными, внутренне им присущими.
2. Аспекты описания функционирования системы на различных стратах в общем случае не связаны между собой. Поэтому

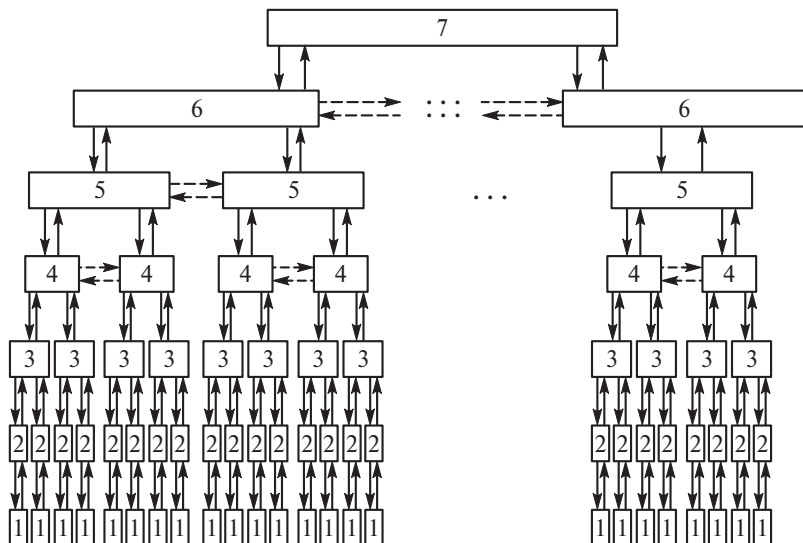


Рис. 7.10. Стратифицированное представление системы управления народным хозяйством страны (по В.А. Трапезникову):

- 1 — страта 1: автоматические регуляторы агрегатов; 2 — страта 2: системы управления агрегатами; 3 — страта 3: СУ автоматическими линиями; 4 — страта 4: цеховые СУ; 5 — страта 5: СУ предприятиями; 6 — страта 6: СУ отраслями народного хозяйства; 7 — страта 7: СУ народным хозяйством

принципы и законы, используемые для характеристики системы на любой страте, в общем случае не могут быть выведены из принципов, используемых на других стратах.

3. Существует асимметричная зависимость между условиями функционирования системы на различных стратах.

4. На каждой страте имеется свой собственный набор терминов, концепций и принципов.

5. Понимание системы возрастает при последовательном переходе от одной страты к другой: чем ниже спускаемся по иерархии, тем более детальным становится раскрытие системы; чем выше поднимаемся, тем яснее становится смысл и значение всей системы.

Слои. Уровни сложности принимаемого решения. Почти в любой реальной ситуации принятия решения существуют две предельно простые, но чрезвычайно важные особенности:

1) когда приходит время принимать решения, принятие и выполнение решения нельзя откладывать;

2) неясность относительно последствий различных альтернативных действий и отсутствие достаточных знаний об имеющихся связях препятствуют достаточно полному формализованному описанию ситуации, необходимому для рационального выбора действий.

Эти два фактора приводят к основной дилемме принятия решения: с одной стороны, необходимо действовать немедленно, с другой же — столь же необходимо, прежде чем приступить к действиям, попытаться лучше понять ситуацию. При принятии решения в сложных ситуациях решение этой дилеммы ищут в иерархическом подходе (рис. 7.11). Сложная проблема принятия решения разбивается на семейство последовательно расположенных более простых подпроблем $D_i, i = \overline{1, n}$, так что решение всех подпроблем позволяет решить и исходную проблему P . Такая иерархия называется иерархией слоев принятия решений, а вся система (D на рис. 7.11) — *многослойной системой* (принятия решений).

Важным видом иерархий является функциональная иерархия принятия решений или управления. Эта иерархия возникает в связи с тремя основными аспектами проблемы принятия решения в условиях полной неопределенности (рис. 7.12). Функциональная иерархия, изображенная на рис. 7.12, состоит из трех слоев:

1. Слой выбора: задача этого слоя — выбор способа действий.

2. Слой общения, или адаптации: задача слоя — конкретизация множества неопределенностей, с которым имеет дело слой выбора.

3. Слой самоорганизации: задача слоя — выбор структуры, функции и стратегии, которые используются на нижележащих слоях, таким образом, чтобы по возможности приблизиться к глобальной цели.

Многоэшелонные системы: организационные иерархии. Это понятие иерархии подразумевает, что: 1) система состоит из семейства четко выделенных взаимодействующих подсистем; 2) некоторые из подсистем являются принимающими решения элементами и 3) принимающие решения элементы располагаются иерархически в том смысле, что некоторые из них находятся под влиянием или управляются другими решающими элементами. Блок-схема такого типа приведена на рис. 7.13. Уровень в такой системе называется эшелон. Такие системы называются многоэшелонными, многоуровневыми или многоцелевыми.

Для эффективного использования многоуровневой структуры существенно, чтобы элементам принятия решения была предоставлена некоторая свобода действий. Должно быть проведено рациональное распределение усилий по принятию решений между элементами различных уровней. Только при этом условии будет оправдано само существование иерархии. По характеру иерархического расположения образующих систему элементов различают следующие системы принятия решений:

- одноуровневые одноцелевые системы;
- одноуровневые многоцелевые системы;
- многоуровневые многоцелевые системы.

Отметим еще несколько свойств, характерных для любых иерархических структур:

1) элемент верхнего уровня иерархии имеет дело с более крупными подсистемами или с более широкими аспектами поведения системы в целом;

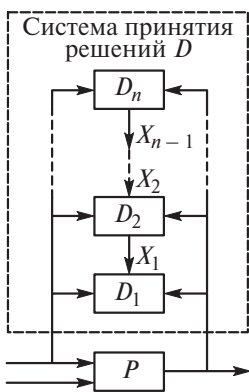


Рис. 7.11. Многослойная иерархия системы принятия решений

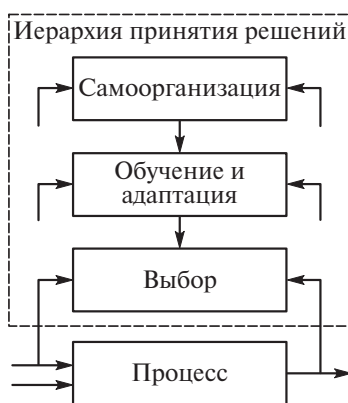


Рис. 7.12. Функциональная многослойная иерархия принятия решений

2) период принятия решения для элемента верхнего уровня больше, чем для элементов нижних уровней;

3) элемент верхнего уровня имеет дело с более медленными аспектами поведения всей системы;

4) описания и проблемы на верхних уровнях менее структурированы, содержат больше неопределенностей и более трудны для формализации.

Преимущества иерархических систем управления (ИСУ). ИСУ нашли широкое распространение. Универсальный характер ИСУ обусловлен рядом преимуществ, которыми они обладают по сравнению с другими СУ, в частности с системами централизованного управления:

1) свобода локальных действий (в течение интервала времени, обусловленных моментами поступления управляющих воздействий со стороны вышележащего уровня);

2) возможность целесообразно сочетать различные для каждого из уровней системы локальные критерии оптимальности и глобальный критерий оптимальности системы в целом;

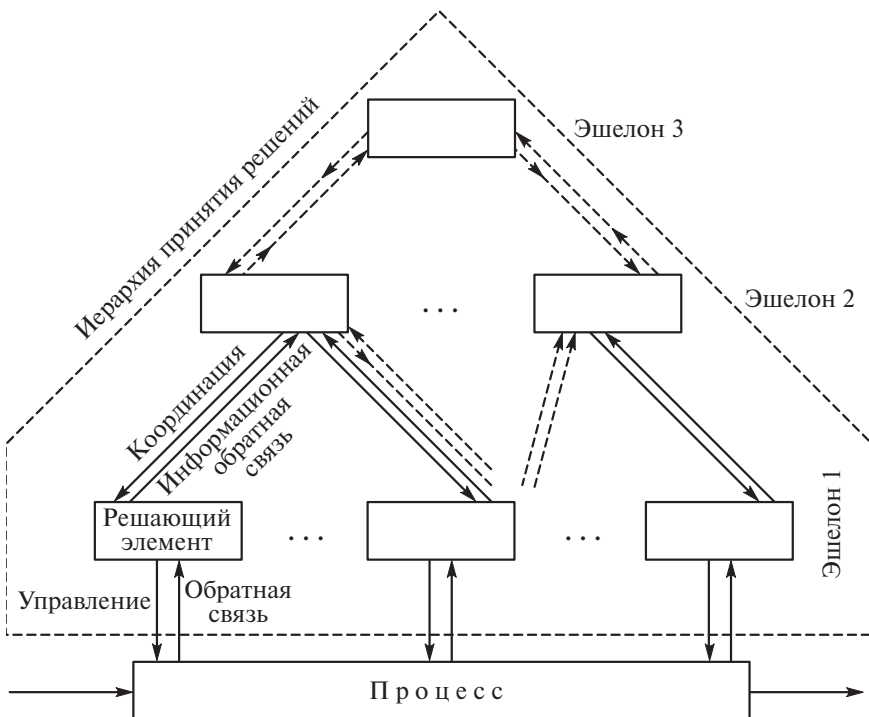


Рис. 7.13. Многоуровневая организация иерархий; многоэшелонная система

3) отсутствие необходимости пропускать очень большие потоки информации через один управляющий орган, так как при использовании ИСУ информация с нижнего уровня передается на верхний в обобщенном виде;

4) повышенная надежность СУ и большие возможности введения элементной избыточности в систему на необходимом уровне управления;

5) гибкость и адаптивность СУ;

6) универсальность при решении однотипна в целом, но в деталях управления может отличаться;

7) в ряде случаев — экономическая целесообразность по сравнению с СУ иной структуры.

7.4. Формализация иерархических понятий

Координация. При оптимизации взаимодействия между уровнями в ИСУ весьма важной является проблема координации. Пусть система состоит из n подсистем первого уровня S_1, \dots, S_n , управляющих процессов P и одной подсистемы второго (высшего) уровня S_0 (рис. 7.14). Цель высшей подсистемы влиять на низшие таким образом, чтобы достигалась общая цель, заданная для всей системы. Это составляет содержание понятия координации. Схема, представленная на рис. 7.14, является простейшей, хотя она обладает наиболее важными характеристиками многоуровневых ИСУ и может служить в качестве элементарного блока при построении более сложных ИСУ. На рис. 7.14 приняты следующие обозначения: $u_i, i = \overline{1, n}$ — управляющие воздействия на подпроцесс P_i ; X — входные воздействия на процесс P ; β_i — об-

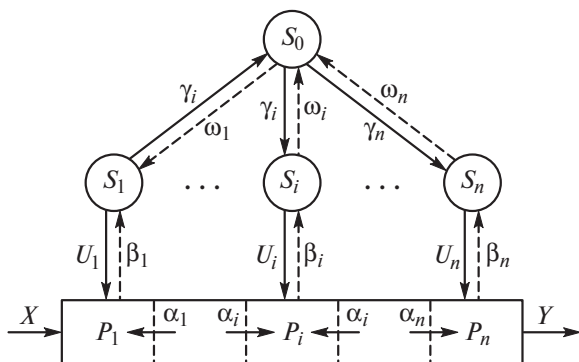


Рис. 7.14. Двухуровневая иерархическая система управления

ратная связь подсистем S_i с подпроцессом P_i ; γ_i — координирующее воздействие на подсистему S_i ; ω_i — обратная связь координатора S_0 с подсистемой S_i ; Y — выход процесса P ; α_i — связи подпроцесса P_i с процессом P , $\alpha = \{\alpha_1, \dots, \alpha_n\}$.

Так как подсистемы низшего уровня действуют в направлении достижения собственных локальных целей, то между ними возникает конфликт, в результате которого глобальная цель системы может быть не достигнута. Поэтому задача координатора состоит в уменьшении последствий такого конфликта. Очевидно, что конфликт не возникнет, если входы связей α_i , желательные с точки зрения каждой подсистемы S_i , совпадают с теми, которые фактически имеют место в процессе работы системы. Следовательно, главный вопрос, который возникает в связи с задачей координации подсистем низшего уровня, состоит в том, каким образом входы связей учитываются этими подсистемами.

Возможны три способа анализа видов связей, каждому из которых соответствует свой способ координации: 1) подсистемы низшего уровня решают свои локальные задачи, предполагая входы связей равными тем, которые предсказаны подсистемой высшего уровня; 2) подсистемы низшего уровня решают локальные задачи, предполагая входы связей находящимися в пределах некоторых диапазонов значений, предсказанных подсистемой высшего уровня; 3) входы связей рассматриваются подсистемами низшего уровня как добавочные переменные, которые можно свободно выбирать. Соответствующие способы координации называются *координацией предсказаний взаимодействий*, *координацией оценки взаимодействий* и *координацией баланса взаимодействий*.

Сущность принципов координации рассмотрим на примере двухуровневой ИСУ, изображенной на рис. 7.15. Первый уровень (регуляторы S_1 и S_2) управляют объектами P_1 и P_2 , подавая на вход их управляющие воздействия соответственно u_1 и u_2 . Второй уровень (координатор S_0) управляет регуляторами S_1 и S_2 , подавая на них координирующие воздействия — соответственно γ_1 и γ_2 . Вмешательство координатора проявляется в том, что от значений γ_1 и γ_2 зависят управляющие воздействия u_1 и u_2 и это обозначается в виде $u_1(\gamma_1)$ и $u_2(\gamma_2)$. В общем случае γ_1 и γ_2 могут зависеть одновременно от γ_1 и γ_2 ,

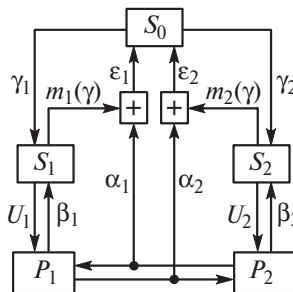


Рис. 7.15. Структура двухуровневой ИСУ с двумя объектами управления

тогда это обозначается как $u_1(\gamma_1)$ и $u_2(\gamma_2)$, где $\gamma = \{\gamma_1, \gamma_2\}$. Система называется *координируемой*, если найдены такие значения $\hat{\gamma}$, что $u_1(\hat{\gamma})$ и $u_2(\hat{\gamma})$ удовлетворяют общей цели, поставленной перед системой. Значения управляющих воздействий u_1 и u_2 , удовлетворяющих условию координируемости, обозначаются через $\hat{u}_1(\gamma)$ и $\hat{u}_2(\gamma)$. Для осуществления процесса координации существенное значение имеют величины α_1 и α_2 , характеризующие перекрестные связи между объектами управления P_1 и P_2 . Текущие значения этих величин α_1 и α_2 , передаются координатору S_0 путем сопоставления их со значениями $\hat{m}_1(\gamma)$ и $\hat{m}_2(\gamma)$, удовлетворяющими условиям координируемости системы. Ошибки рассогласования $\varepsilon_1 = \alpha_1 - \hat{m}_1(\gamma)$ и $\varepsilon_2 = \alpha_2 - \hat{m}_2(\gamma)$ используются для построения алгоритма функционирования координатора S_0 .

Стратегия координации, заключающаяся в том, что управляющие воздействия верхнего уровня распределяются между подсистемами соседнего нижнего уровня таким образом, что каждая из этих подсистем становится автономной относительно всех других подсистем этого же уровня, называется *принципом предсказания взаимодействия*. Другими словами, управление $\hat{u}_1(\gamma) = \{\hat{u}_1(\gamma), \dots, \hat{u}_n(\gamma)\}$ удовлетворяет глобальной цели системы всякий раз, когда $\gamma_1 = \alpha_1(\gamma)$, ..., $\gamma_n = \alpha_n(\gamma)$, т. е. взаимодействия точно предсказуемы.

Стратегия координации, при которой значения управляющих воздействий $\hat{u}_1(\gamma)$ и $\hat{u}_2(\gamma)$ удовлетворяют глобальной цели системы, когда $\alpha_1(\gamma) = \hat{\alpha}_1(\gamma)$ и $\alpha_2(\gamma) = \hat{\alpha}_2(\gamma)$, называется *принципом баланса взаимодействия*. Если же последние соотношения заменяются на $\alpha_1(\gamma) \in A_1^Y$ и $\alpha_2(\gamma) \in A_2^Y$ — допустимый диапазон изменения взаимодействий α_1 и α_2 , то принцип координации именуется *принципом оценки взаимодействий*.

При поддержании параметров системы в заданных пределах фактическое взаимодействие должно принадлежать предсказанному значению.

Выбор той или иной стратегии координации производится на основе сопоставления результатов теоретических расчетов, моделирования и эвристических соображений. Теоретические расчеты сводятся к построению соответствующей итерационной процедуры, базирующейся на одном из известных, но специально для этой цели модифицированном методе оптимального управления. В частности, разработаны различные градиентные и интегральные процедуры для обеспечения координации $\varepsilon_i = 0$.

При исследовании более сложных ИСУ, имеющих больше двух уровней, характер задач при переходе от уровня к уровню бу-

дет существенно изменяться. Так, если для нижних уровней характерны именно описанные выше методы координации, то для средних уровней задачи координации могут быть уже иными, а для верхних уровней, на которых решаются задачи чисто экономического характера и долгосрочного планирования и прогнозирования, они приобретают иной, еще более сложный характер. Считается, что по мере перехода от нижних уровней к верхним решение задач все более и более затрудняется, так как приходится оперировать все с менее и менее достоверной информацией. Ее объема обычно не хватает для качественного осуществления процесса управления. Однако уже хорошо известно, что только решение задач для всех уровней, а не только для нижних, позволяет действительно достичь существенных экономических результатов при использовании ИСУ.

Теория координации в настоящее время находится на начальном этапе развития.

Декомпозиция. Принцип декомпозиции (децентрализации) состоит в разбиении системы на подсистемы, обладающие требуемыми свойствами. Расчленение глобальной задачи на локальные подзадачи осуществляется из соображений, связанных с необходимостью децентрализовать управление сложной системой, привлекая для этого управляющие органы составляющих ее подсистем. Отсутствие у центрального органа детальной информации о возможностях отдельных подсистем вызывает необходимость в иерархической структуре, при которой вышестоящий орган собирает в некоторой агрегированной форме информацию о подведомственных ему подсистемах, а затем спускает им в той или иной форме указания о требуемых или ожидаемых от них действиях. Ознакомившись с этими указаниями, подсистемы могут направить наверх свои предложения, в которых они лучшим образом учли свои внутренние возможности. На основе этой информации центральный орган корректирует свои указания и процесс повторяется до окончательного согласования. Дополнительная информация, передаваемая в локальные подсистемы на каждом шаге, обычно представляет собой целевой функционал или/и вектор ограничений локальных задач.

Математическое исследование описанного процесса получило развитие в форме задач блочного программирования, а точнее, в форме итеративных алгоритмов решения таких задач. С этой целью используются два основных класса таких процедур: Данцига — Вульфа и Корнай — Липтака. При использовании декомпозиционных методов исходная задача сводится к решению последовательности задач меньшей размерности, каждая из которых

решается с учетом ее специфических свойств. Метод Данцига — Вульфа — это один из наиболее старых и известных декомпозиционных методов, который был разработан Дж. Данцигом и Ф. Вульфом (США) в 1960 г. В этом методе центр спускает предприятиям информацию в виде цен, а получает от них информацию в виде предполагаемых объемов затрат и выпусков. Декомпозиционный метод Данцига — Вульфа относится к задачам линейного программирования, в которых искомые переменные группируются в блоки, связанные в единую задачу либо дополнительными общими ограничениями, либо дополнительными общими переменными, входящими во все блоки одновременно.

В методе Данцига — Вульфа рассматривается задача линейного программирования, ограничения которой разделены на два блока, т. е. требуется найти максимум функции

$$\sum_{i=1}^n c_i x_i \Rightarrow \max; \quad (7.1)$$

при ограничениях

$$\sum_{i=1}^n A_i^0 x_i = B^0; \quad (7.2)$$

$$\sum_{i=1}^n A_i^1 x_i = B^1; \quad (7.3)$$

$$x_i \geq 0, \quad (7.4)$$

где $C = (c_1, \dots, c_n)$ — вектор строки; $B = (B^0, B^1)^T = (b_1, \dots, b_m, b_{m+1}, \dots, b_{m+k})^T$ — $(m+k)$ — мерный вектор ограничения задачи; $A_i = (A_i^0, A_i^1)^T = (a_{1,i}, \dots, a_{m,i}, a_{m+1,i}, \dots, a_{m+k,i})^T$ — $(m+k)$ — мерный i -тый вектор условий $i = \overline{1, n}$; T — знак транспонирования, $X = (x_1, \dots, x_n)$ — вектор переменных.

Для обоснования иерархического построения СУ наибольший интерес представляют итеративные методы решения задач блочного программирования, которые в отличие от монотонных методов (например, симплекс-метода) не позволяют получить решение за конечное число шагов и процесс приближения к решению не является монотонным. Однако они обеспечивают приближение к оптимальному решению при увеличении числа итераций. Задачи итерационного типа обычно рассматриваются как двухуровневые.

В алгоритме Корнай — Липтака центр спускает план в натуральных показателях, а информация в форме оценок в центр поступает от предприятий. Обозначим через $u_i = \overline{1, n}$ вектор размерности v , координаты которого представляют собой объемы соответствующих общих ресурсов, выделенных i -тому блоку. Векторы выбираются так, чтобы $\sum_{i=1}^n u_i = B^2$. Тогда задача (7.1)—(7.4) распадается на n отдельных задач вида

$$c_i x_i \Rightarrow \max; \quad A_i^0 x_i = u_i; \quad A_i^1 x_i = B^1; \quad x_i \geq 0. \quad (7.5)$$

Двойственная к ней задача будет иметь вид

$$(u_i y_i + B^1 z_i) \Rightarrow \min; \quad (7.6)$$

$$A_i^0 y_i = A_i^1 z_i \geq c. \quad (7.7)$$

Пусть Y_i — множество допустимых планов задачи (7.6)—(7.7), $X_i(u_i)$ — множество допустимых планов задачи (7.5) при фиксированном векторе u_i . Будем считать, что $Y_i \neq \emptyset$ для всех $i = \overline{1, n}$.

Введем $(v + n)$ — мерный вектор $u = (u_1, \dots, u_n, u_{n+1}, \dots, u_{n+v})^T$ и множество

$$U = \{u / \sum u_i = B^2, X_i(u_i) \neq \emptyset, \forall i = \overline{1, n}\}.$$

Назовем центральным планом задачи (7.1)—(7.4) и U — множеством допустимых центральных планов. Через $\varphi_i(u_i)$ обозначим оптимальное значение целевой функции в задаче (7.6), (7.7) при фиксированном векторе u_i и положим

$$\varphi(u) = \sum_{i=1}^n \varphi_i(u_i).$$

Тогда задача (7.1)—(7.4) эквивалентна задаче

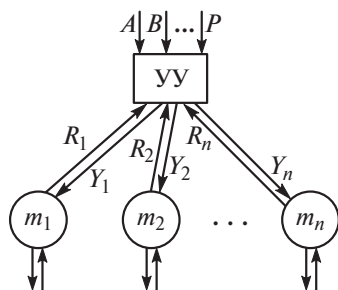
$$\varphi(u) \Rightarrow \max_{u \in U}, \quad (7.8)$$

которая решается сведением ее к матричной игре и решением последней итеративным методом Брауна — Робинсона или другими способами.

Необходимо подчеркнуть, что блочные алгоритмы математического программирования — очень грубые модели процессов планирования и управления. Более совершенные модели должны учи-

тивать активность элементов системы, т. е. наличие у них собственных целей, характеристики средств переработки и достоверность информации, запаздывание при ее обмене и другие факторы.

При использовании метода декомпозиции для решения задач планирования и оперативного управления имеются существенные различия. В планировании результаты решения задачи используются лишь после его окончания. Задача при этом может решаться на одном или нескольких уровнях. В оперативном управлении промежуточные результаты в процессе решения задачи используются для управления. При этом существенной становится динамика процесса и учет разного масштаба времени на различных уровнях системы.



7.16. Интерпретация принципа агрегации

Агрегация. Агрегация является одним из методов синтеза сложных систем, т. е. процесса построения системы, обладающей заданными свойствами, путем объединения отдельных подсистем с известными характеристиками. Таким образом, агрегация может рассматриваться как противоположность декомпозиции. Ряд существенных результатов использования принципа агрегации получен Р. Куликовским.

Рассмотрим ИСУ, показанную на рис. 7.16. Управляющее устройство (УУ) получает от системы высшего уровня некоторое количество ресурсов A, B, \dots, P и распределяет их оптимальным образом между объектами управления $M_i, i = 1, n$. Информация о выходных эффектах, или продуктах каждого объекта управления (ОУ), поступает на УУ и в агрегированном виде Y_i и определяет валовую продукцию системы. Пусть известны характеристики оптимального управления ОУ в виде

$$A_i^\alpha B_i^\beta \dots P_i^p = Q_i^q. \quad (7.9)$$

Тогда задача состоит в определении таких значений $A_i^0, B_i^0, \dots, P_i^0$, переменных $A_i, B_i, \dots, P_i, i = 1, n$, которые максимизируют суммарный выход

$$Y = \sum_{i=1}^n Y_i = \sum_{i=1}^n [Q_i^q A_i^{-\alpha} B_i^{-\beta} \dots P_i^{-p}]^{1/q} \Rightarrow \max \quad (7.10)$$

при соблюдении условий

$$\sum_{i=1}^n A_i = A; \quad \sum_{i=1}^n B_i = B; \dots; \quad A_i \geq 0; \quad B_i \geq 0; \dots$$

Можно показать, что оптимальные значения $A_i^0, B_i^0, \dots, P_i^0$ определяются из линейных уравнений:

$$\frac{A_i^0}{A} = \frac{B_i^0}{B} = \dots = \frac{P_i^0}{P} = \frac{Q_i^0}{Q},$$

где $Q = \sum_{i=1}^n Q_i$. Подставляя значения $A_i^0, B_i^0, \dots, P_i^0$ в (7.10), получаем

$$A^\alpha B^\beta \dots P^\rho = Q^q, \quad q = \alpha + \beta + \dots + \rho.$$

Как видно, характеристики оптимального управления для агрегированной системы имеют такой же аналитический вид, как и характеристики подсистем (7.9). Так как каждая подсистема полностью определена значением коэффициента Q_j , то можно сказать, что процесс агрегации сопровождается n -кратным сжатием информации, содержащейся в характеристиках оптимального управления агрегированных подсистем.

Конечно, существует много характеристик оптимального управления, которые не описываются уравнениями типа (7.9). Тогда вопрос агрегации и оптимизации превращается в трудную задачу нелинейного программирования.

В рассмотренном случае агрегации (7.9), (7.10) предполагалось, что выходные эффекты Y_i аддитивным образом содействуют достижению общей цели Y , т. е.

$$Y = \sum_{i=1}^n \varepsilon_i Y_i, \quad \varepsilon > 0.$$

Однако во многих производственных системах имеется ряд критериев качества, которые лишь в совокупности определяют качество процесса функционирования всей системы.

Задачи, решаемые системами, обычно имеют настолько большую размерность, что изучение их в исходных переменных не позволяет оптимизировать процесс и получить представления о влиянии отдельных ОУ системы на общую эффективность. Вы-

ход из этого положения состоит в том, что нужно либо выделять существенные переменные, по которым и производится оптимизация, либо агрегировать переменные. При агрегировании переменных получается модель существенно меньшей размерности. Агрегирование переменных, а также переход от агрегированных величин к исходным могут быть выполнены различными способами.

Отметим, что не при любом выборе агрегированных переменных от исходной модели можно перейти к адекватной агрегированной модели, т. е. к такой модели, состояние выходов которой совпадает с агрегатами состояния выходов исходной модели, если состояние входов агрегированной модели совпадает с агрегатами состояния входов исходной модели. В тех случаях, когда построенная агрегированная модель адекватна исходной, говорят, что агрегирование совместно.

Предположим, что для некоторого объекта построена модель в виде системы уравнений

$$y_i = f_i(x_i), \quad i = \overline{1, n},$$

связывающих вход модели x с её выходами y . Тогда агрегирование этой модели сводится к введению некоторой замены переменных

$$\begin{aligned} \varepsilon_k &= \psi_k(x_1, \dots, x_n), \quad k = \overline{1, m}; \quad m < n; \\ r_k &= \phi_k(y_1, \dots, y_n), \end{aligned} \quad (7.11)$$

задающей связи между переменными x и y исходной модели с агрегатами ε , r и k введению системы уравнений

$$r_k = \mu_k(\varepsilon_1, \dots, \varepsilon_m), \quad k = \overline{1, m}, \quad (7.12)$$

связывающих входы ε агрегированной модели с ее выходами r .

Условие совместности при таком агрегировании имеет вид

$$\phi_k(f_1(x), \dots, f_n(x)) = \mu_k(\psi_1(x), \dots, \psi_m(x)), \quad k = \overline{1, m}, \quad (7.13)$$

для любых $X = (x_1, \dots, x_n)$.

В качестве примера рассмотрим применение принципа агрегирования в линейных экономических моделях, в частности в моделях межотраслевого баланса. Подобные задачи состоят в определении объемов x_1, x_2, \dots, x_n полных выпусков каждого из n видов продукции, необходимых для обеспечения заданных объ-

емов $y_i, i = \overline{1, n}$ конечных выпусков и проверки допустимости полученного вектора $X = (x_1, \dots, x_n)$ по некоторым сводным показателям, например, по стоимости или производственным мощностям. Одной из наиболее известных моделей межотраслевого баланса, на которых решаются подобные задачи, является модель «затраты — выпуск»

$$X = AX + Y, \quad (7.14)$$

где $X = (x_1, \dots, x_n)$ — вектор полного выпуска продукции; $Y = (y_1, \dots, y_n)$ — вектор конечного выпуска продукции; $A = \|a_{ij}\|$ — матрица коэффициентов прямых материальных затрат.

Современному уровню разделения труда в промышленности соответствует номенклатура из десятков тысяч различных видов продукции, что и определяет размерность n исходной модели (7.14). Агрегирование этой модели означает уменьшение размерности объединением продуктов в укрупненную (агрегированную) номенклатуру, состоящую из $m (m < n)$ агрегированных показателей. Тогда в соответствии с (7.11) и (7.12) можно ввести переменные $E = \{\varepsilon_1, \dots, \varepsilon_m\}$ и $R = (r_1, \dots, r_m)$, такие, что

$$E = CX, \quad R = DY, \quad (7.15)$$

где $C = \|c_{ij}\|$, $D = \|d_{ij}\|$ — матрицы одинаковых размеров $m \times n$, ранга m , и матрицу H , задающую связь между агрегатами E и R :

$$E = EH + R. \quad (7.16)$$

Систему уравнений (7.15) и (7.16) называют линейной системой биматричного агрегирования модели «затраты — выпуск».

Условие совместности (7.13) в данном случае имеет вид

$$(\Phi_m - H)C = D(\Phi_n - A), \quad (7.17)$$

где Φ_n — единичная матрица порядка $k, k = m, n$.

Агрегированная модель (7.16) с точностью до обозначений в размерности совпадает с исходной моделью «затраты — выпуск» (7.14). Это позволяет применить описанный аппарат агрегирования и к модели (7.16). В результате ряда последовательных шагов агрегирования можно построить многоступенчатую систему агрегирования модели «затраты — выпуск», соответствующую иерархической структуре управления промышленным производством.

В сложных ИСУ модели нижних уровней и глобальная модель отличаются как размерностью, так и составом переменных. Поэ-

тому при конструировании модели ИСУ признана необходимой различная степень укрупнения показателей разных уровней. Для решения таких задач разработаны специальные методы итеративного агрегирования, позволяющие увязывать решения, получающиеся на верхних уровнях, с решениями, формирующимися на нижних уровнях.

Основная идея метода итеративного агрегирования заключается в следующем. Пусть дана задача математического программирования

$$f(X) \Rightarrow \max; \quad g(X) \leq 0; \quad x \geq 0. \quad (7.18)$$

Здесь $X = \{x_i\}_1^n$ — вектор элементов x_i , размерностью n ; $g(X) = \{g_j(X)\}_1^m$.

Разобьем множество индексов $N = \{1, \dots, n\}$ на l пересекающихся подмножеств $J_k, k = 1, l$ и подставим определенным образом в соответствие группе переменных $x_i, i \in J_k$ одну агрегированную переменную x_k . Тогда задача (7.18) в новых переменных переписывается в виде

$$\bar{f}(X) \Rightarrow \max; \quad \bar{g}(X) \leq 0; \quad x \geq 0, \quad (7.19)$$

где

$$X = \{x_k\}_1^l; \quad \bar{g}(X) = \{\bar{g}_j(X)\}_1^m, \quad l < m.$$

Правила перехода от задачи (7.18) к задаче (7.19) составляют первую часть алгоритма, а именно агрегирование. Полученный в результате решения задачи (7.19) агрегированный вектор X определенным образом агрегирует до исходной размерности n . Это вторая часть процесса — дезагрегирование. Детализация решения позволяет перейти к новой задаче и т. д., пока с заданной степенью точности не будет построено детализированное решение, удовлетворяющее условиям (7.18).

Алгоритмы итеративного агрегирования могут быть реализованы в разных вариантах, одни из которых конечношаговые, а другие — бесконечношаговые итеративные процессы, причем последние нестационарны.

В заключение заметим, что методы агрегации разработаны в настоящее время недостаточно. Особенно это относится к методам итеративного агрегирования — весьма важному подклассу математического программирования.

7.5. Принципы управления сложными системами

Управление в сложных системах принципиально отличается от традиционного представления об управлении, в частности от того, что принято называть «оптимальным управлением» (точнее — «программным управлением»), т. е. переводом системы в желаемое состояние по некоторому оптимальному пути. Это очевидно: сложные системы слабопредсказуемы, определить как желаемое, так и практически достижимое состояние невозможно, тем более невозможно выбрать и навязать системе «оптимальный» (в детерминистическом или статистическом смысле) путь перехода, поскольку структура и функции системы не взаимоопределимы. По содержанию и механизму действия управление сложными системами, в том числе самоуправление, наиболее близко к физиологическим процессам возбуждения и торможения, иначе говоря, внешнего и внутреннего стимулирования. Прямые и обратные связи, все виды и формы воздействия (если они не приводят к разрушению системы) — не более чем стимулы, возбуждающие или тормозящие внутрисистемные процессы, ход и последствия которых в основном определяются самой системой.

Проблема управления сложными системами состоит в исследовании влияния возбуждающих и тормозящих стимулов на поведение системы и конечный результат и в использовании стимулирования для достижения требуемой эффективности системы. Возбуждение может перейти в торможение и наоборот: при изменении уровня стимула и состояния системы, поэтому априорная оценка характера воздействия затруднительна. Управление должно достигаться ценой относительно малого энергоресурса. Типичным в этом смысле является информационное управление, при котором энергоресурс управления незначителен по сравнению с энергоресурсом объектов управления. Сложная система обладает не только большим энергоресурсом, но и большой динамической инерционностью.

Сформулируем общую задачу управления сложной системой в следующем виде:

$$\frac{dY(t)}{dt} = \Phi \left\{ t, Y(t), Y(t - \tau), \int_{-\infty}^{t - \tau} Y(s - \delta_1) dF_1[(s, t), t], \dots \right. \\ \left. \dots, \int_{-\infty}^{t - \tau} Y(s - \delta_k) dF_k[(s, t), t], x(t, t_x), u(t, \tau_u) \right\}, \quad (7.20)$$

где \underline{Y} — выход системы; $x(t, \tau_x) \in X$ — воздействие среды; F_i , $i = \overline{1, k}$ — известные функции; $u(t, \tau_u) \in U$ — область возможных и допустимых управлений; τ_x, τ_u, δ_i — запаздывания. При учете предыстории $t_0 - \tau_{\max} \leq t \leq t_0$.

Требуется найти управление и $\hat{u}(t, \hat{\tau}_u)$, обеспечивающее высокую эффективность системы

$$\mathcal{E} = \mathcal{E}[Y(t), T, C, \hat{u}(t, \hat{\tau}_u), x(t, \tau_x)] \geq \mathcal{E}^*, \quad (7.21)$$

где C — стоимость системы.

Система считается управляемой, если:

$$\left. \begin{aligned} \mathcal{E}[Y(t), T, C] &\geq \mathcal{E}^* \quad \text{при} \quad u(X) \in U, \\ P_u(u(t, \tau_u(t))) &\geq P^*, \\ T & \\ \int_0^T P_u(t) dt &= E^* \leq E_s, \end{aligned} \right\} \quad (7.22)$$

где $P(u)$ — мощность, потребляемая для реализации управления и при задержках τ_u , E_s — энергоресурс системы; P^* , E^* — допустимые значения мощности и энергии управления на интервале $[0, T]$.

Значимость функций, входящих в Φ выражения (7.20), неодинакова для систем различного класса: $Y(t)$ представляет собой состояние выходов системы к началу управления; $Y(t - \tau)$ при $t_0 - \tau_{\max} \leq t \leq t_0$ — поведение системы на интервале, предшествующем управлению (предыстория системы);

$$\left\{ \int_{-\infty}^{t-\tau} Y(s - \delta_i) dF_i((s, t), t) \right\}, \quad i = \overline{1, k}, \{\delta_i\} = \delta,$$

множество, определяющее типовые свойства системы, ее способность к управлению и внутренние тенденции, некоторую относительно стабильную (поскольку интегрирование ведется от $-\infty$) линию поведения и управляемости, ее внутреннюю мотивацию.

Управление и воздействие среды могут быть независимы, но могут быть и зависимы, если управляющая система располагает априорной информацией относительно X или оценивает x на интервале $[0, T]$, а среда может иметь информацию относительно U или оценивать u на интервале $[0, T]$.

Рассмотрим частные случаи.

1. Функция Φ такова, что влияние τ на $\left\{ \int_{-\infty}^{t-\tau} Y(s - \delta_i) dF_i((s, t), t) \right\}$,

$i = \overline{1, k}$, пренебрежимо мало; в этом случае

$$\frac{dY}{dt} = \Phi[t, Y(t), x(t, \tau_x), u(t, \tau_u)]. \quad (7.23)$$

Если, кроме того, Φ — линейная функция и $\tau_u = 0, x = 0$, то

$$\frac{dY}{dt} = AY + u + f(t). \quad (7.24)$$

Это задача программного управления, характерная для технических систем, которая решается известными методами. Если $x \neq 0$, то

$$\frac{dY}{dt} = AY + x + u + f(t) \quad (7.25)$$

и это есть управление в условиях содействия или противодействия. В случае $f(t) = 0$ и

$$\frac{dY}{dt} = AY + u + \xi, \quad (7.26)$$

где ξ — случайная функция, то управление стохастическое.

2. Функция Φ слабо зависит от $\left\{ \int_{-\infty}^t Y(s) dF_i((s, t), t) \right\}$, $i = \overline{1, k}$,

в этом случае

$$\frac{dY}{dt} = \Phi[t, Y(t), Y(t - \tau), x(t - \tau_x), u(t - \tau_u)], \quad (7.27)$$

управление не опирается на мотивацию, но существенно зависит от ситуации на интервале $[t - \tau, t]$. Это *ситуационное управление*. При линейной Φ и $f(t) = 0$

$$\frac{dY}{dt} = A_1 Y(t) + A_2 Y(t - \tau) + x(t - \tau_x) + u(t - \tau_u), \quad (7.28)$$

где A_1 и A_2 — матрицы коэффициентов. Решение (7.28) (если оно существует) достигается типовыми методами. Задачи ситуационного управления (7.27) и (7.28) характерны для производственных систем.

3. Функция Φ линейная и от t непосредственно не зависит. Тогда

$$\begin{aligned} \frac{dY(t)}{dt} = & A_1 Y(t) + A_2 Y(t - \tau) + \int_{-\infty}^{t-\tau} Y(s - \delta_1) dF_1((s, t), t) + \dots \\ & \dots + \int_{-\infty}^{t-\tau} Y(s - \delta_k) dF_k((s, t), t) + x(t, \tau_x) + u(t - \tau_u). \end{aligned} \quad (7.29)$$

Этот случай характерен для нейтральной среды (случайной либо, по крайней мере, не зависящей от u). Достижение высокой эффективности возможно только путем приспособления сложившихся на интервале $[-\infty, t - \tau]$ свойств системы к изменению ситуации, складывающейся на интервале $[t - \tau, T]$, т. е. путем адаптации системы, средством которой является управление. Это *адаптивное управление*, применяемое в случае, если влияние традиций не очень сильно, во всяком случае их можно перестроить на относительно коротком интервале времени.

4. Функция Φ зависит от t

$$\begin{aligned} \frac{dY}{dt} = & \Phi \left[t, Y(t), Y(t - \tau), \int_{-\infty}^{t-\tau} Y(t - \delta_1) dF_1(s, t), t), \dots \right. \\ & \left. \dots + \int_{-\infty}^{t-\tau} Y(t - \delta_1) dF_k((s, t), t) + x(t, \tau_x) + u(t, \tau_u) \right]. \end{aligned} \quad (7.30)$$

Строгое эффективное управление невозможно. Управление должно влиять на внутреннюю мотивацию системы; это достижимо, если мотивация системы известна (хотя бы частично). Оптимальных (в смысле $\max \mathcal{E}$) решений не существует. *Управление рефлексивное.*

Рефлексивное управление может оказаться эффективным, если его применять на интервале времени $T_u \in T$. В этом случае влияние τ и δ незначительно и

$$\begin{aligned} \frac{dY}{dt} = & \Phi \left[t, Y, \int_{-\infty}^{T_u} Y(t) dF_1((s, t), t), \dots, \int_{T_u}^t Y(t) dF_k((s, t), t), \dots \right. \\ & \dots, \int_{T_u}^t Y(t) dF_{1u}((s, t), u(t, \tau_u), \dots \\ & \dots, \left. \int_{T_u}^t Y(t) dF_{ku}((s, t), u(t, \tau_u), x(t, \tau_x)) \right]. \end{aligned} \quad (7.31)$$

Управление эффективно оценивается на интервале $(0, T)$ и изменяет мотивацию системы на интервале $[t - T_u, t]$, таким образом, что она начинает действовать в соответствии с намерениями управляющей системы.

5. Управление начинается с некоторого фиксированного момента времени

$$\begin{aligned} \frac{dY}{dt} = \Phi^* \left[t, Y(t), Y(t - \tau), \int_{t^*}^{t - \tau} Y(t - \delta_1) dF_1^*((s, t), t), \dots \right. \\ \left. \dots, \int_{t^*}^{t - \tau} Y(t - \delta_k) dF_k^*((s, t), t), x(t, \tau_x), u(t, \tau_u), \right] \end{aligned} \quad (7.32)$$

где $\Phi^* = \Phi^*(u^*, \tau^*, \Phi)$, $F_i^* = F_i^*(u^*, \tau^*, F_i)$

В результате управления u^* на интервале $[t^* - \tau^*, T^*]$, функция Φ меняется на Φ^* , F_i — на F_i^* , преобразуется структура системы. Это процесс формирования новой системы, начинающей функционировать в момент t^* под действием: внутренних факторов, т. е. взаимодействия достигающих определенного уровня процессов (самоорганизация), или законсервированной и стимулируемой в момент $t_u^* = t^* - \tau^*$ программы, или под действием внешних факторов (организация). В новой системе мотивация накапливается на интервале $[t^*, t - \tau]$ и действует новое управление $u(t, \tau_u)$.

Управление самоорганизацией (или организацией) состоит в 1) разрушении старой структуры до уровня элементов, которые требуются для новой системы и подлежат сохранению; 2) создании новой структуры; 3) подготовке системы к восприятию управления $u(t, \tau_u)$; 4) блокировании неблагоприятного (в частности, мешающего самоорганизации) воздействия среды, по отношению к которой преобразующаяся система беззащитна.

Рассмотрим виды управления сложной системой.

Адаптивное управление с раздражательным механизмом. В теории управления рассматриваются методы адаптации к стохастической ситуации, оптимальные в среднем. Однако для сложных систем характерны неповторяющиеся ситуации поведения. Для единичных ситуаций известные методы стохастической адаптации малопригодны.

Существует метод, находящийся на стыке стохастической оптимизации и целенаправленного поведения, — *метод массовых проб*. Ситуация не стохастична, она не описывается статистическими законами, и с этим ничего нельзя сделать. Но можно пре-

вернуть систему в квазистохастическую, а затем ее оптимизировать. Это возможно сделать при наличии достаточно большого ресурса, который позволит провести над ситуацией достаточно большое число экспериментов и выявить закономерности. Неопределенность ситуации не будет раскрыта, но ценой некоторого расхода ресурсов она будет исключена. Рассмотрим действие подражательного механизма, лежащего в основе метода.

Пусть система S , состоящая из N однотипных подсистем $S_i \left(S = \sum_{i=1}^N S_i \right)$, функционирует в изменяющейся среде X автономно (без внешнего управления), а ее поведение описывается переменной $Z(t)$. Тогда

$$\dot{Z}(t) = A_1 Z(t - \tau) + A_2 Z(t) + u(t - \tau_u) + \xi(t), \quad (7.33)$$

где $\xi(t)$ — нестационарный случайный процесс, характеристики которого зависят от среды.

Если переменные $Z_i \in Z$ взаимно независимы, то

$$\begin{aligned} \dot{Z}(t) &= A_1 Z(t - \tau) + A_2 Z(t) + u(t - \tau_u) + \xi(t), \quad i = \overline{1, N}, \\ \mathcal{E}_i &= \mathcal{E}_i(Z_i, T), \quad \mathcal{E} = \sum_{i=1}^N \mathcal{E}_i = \mathcal{E}(\{Z_i\}, t). \end{aligned} \quad (7.34)$$

Естественным представляется решение

$$\begin{aligned} u_i(t) &= u_{1i}(t) + u_{2i}(t); \\ u_{2i}(t) &= -\xi(t - \tau_u), \quad i = \overline{1, N}, \end{aligned}$$

реализация которого требует исследования процессов $\xi_i(t)$, экстраполяции их на интервал $[0, \tau_{u2}]$ (желательно, конечно, чтобы интервал был минимальным, а лучше всего, чтобы $\tau_{u2} = 0$) и построения компенсирующих процессов. Управления $u_{1i}(i = \overline{1, N})$ выражают целевую функцию системы. Это *программное управление*. Система должна обладать большими возможностями для выполнения (7.34), которых может и не быть.

Поставим вопрос: какими свойствами должна обладать система S , чтобы наращивать свою эффективность (или, по крайней мере, сохранять эффективность на некотором допустимом уровне) при существенных (в том числе неблагоприятных) изменениях X ? Для определенности примем, что критерием эффективности является энергетический ресурс \mathcal{E} ; это не снижает общности. Подсистемы S_i взаимодействуют между собой и ориентированы

на повышение индивидуальной эффективности (индивидуального энергоресурса подсистем), которую они не умеют предсказать, но способны оценить увеличение или уменьшение.

Сформируем механизм поведения подсистем S_i . Пусть подсистема S_i имеет $U = \{u_j\}$, $j = \overline{1, G}$ возможных способов действия. Осуществляя u_j -тое действие, подсистема S_i проявляет его, а другие подсистемы или, по крайней мере, некоторые из них могут узнать об этом. Способ связи между системами не имеет значения. Эффективность системы

$$\mathcal{E}_i(u_i) = \mathcal{E}_{1i}(u_j, \mathcal{E}) - \mathcal{E}_{2i}(u_j),$$

где \mathcal{E}_{1i} — энергия, извлекаемая из среды при действии u_j ; \mathcal{E}_{2i} — энергозатраты на осуществление действия u_j . Зависимость $\mathcal{E}_{1i}(\mathcal{E})$ обусловлена средоформирующим влиянием S , зависящим от эффективности \mathcal{E} .

Если $0 < \mathcal{E}_i(u_j) < \mathcal{E}^*$, действие u_j называют *рациональным*. Если $\mathcal{E}^* < \mathcal{E}_i(u_j) < \mathcal{E}_*$, действие u_j *эффективно*. Если $\mathcal{E}_i(u_j) < 0$, действие u_j *нерационально*: S_i бесполезно теряет энергию и ее остатка может не хватить для выполнения других действий. Если $\mathcal{E}_i(u_j) > \mathcal{E}_*$, действие u_j *индивидуалистично*: S_i мешает возможности эффективно функционировать другим подсистемам.

Таким образом, предположительная целевая функция состоит в поддержании индивидуальной эффективности в диапазоне $(\mathcal{E}^*, \mathcal{E}_*)$. Эта целевая функция обеспечивает как индивидуальное, так и системное (групповое) благополучие. По-видимому, существует некоторое оптимальное значение $\mathcal{E}_{i\text{opt}}(u_{j\text{opt}})$, при котором достигается $\max(\{\mathcal{E}_{i\text{opt}}\})$. Поскольку $u_{j\text{opt}}$ зависит от среды, подсистемы S_i и действий остальных подсистем высшего уровня, определить его не просто. Во всяком случае полагают, что ни S_i , ни S в целом неведомы понятия целей и целевой функции, эффективности, оптимальности, что анализ ситуации им недоступен, а способ выбора действия опирается на некий автоматизм и предельно прост.

Допустим, что каждая подсистема S_i может действовать в соответствии с одним из следующих критериев: 1) с вероятностью p_1 выбирать наугад любое действие из алфавита $u_j \in U$; 2) с вероятностью p_2 повторить то, что делает другая подсистема, от которой принят сигнал (подражать соседу, $p_1 + p_2 = 1$). Чем удачнее оказался очередной выбор, тем дольше S_i его удерживает: увеличение энергии способствует стабилизации функции, уменьшение стимулирует смену действия; если хватает энергии, смена происходит, если не хватает — смены не происходит. С точки зрения информационного ресурса такой способ выбора действия весьма экономичен: механизм случайного выбора элементарен, а подража-

ние требует анализа G типов хорошо различимых непрерывающихся сигналов. Вряд ли можно придумать что-либо проще такой комбинации автономии и целенаправленности. Природа, широко используя подобную схему, сформировала богатейшие возможности адаптации и развития. Каждая из подсистем может действовать либо полностью автономно, либо координируя свои действия с действиями других подсистем. Поскольку управления сверху нет и нет средств для предварительного «обмена мнениями», остается единственная возможность — подражания лучшим (в каком-то смысле) образцам действий других подсистем.

Можно предположить, что «удачливые» подсистемы сохраняют способ действия, обнаружив его эффективность, на длительное время, а «неудачливые» будут его менять, подражая тем подсистемам, которые действуют относительно стабильно. По-видимому, успеха можно ожидать в том случае, если состояния среды для подражающей подсистемы и образца подражания близки. В этом случае «втягивание» подсистем в подражание переходит в приспособление к сложившейся ситуации. Таким образом формируется общесистемный язык управления.

Оценим эти предположения на следующей математической модели изменения эффективности. Пусть при любом неэффективном действии (область A) $\mathcal{E} = \mathcal{E}_0 e^{-\alpha t}$, а при эффективном (область B) $\mathcal{E} = \mathcal{E}_0 e^{\beta t}$, где \mathcal{E}_0 — эффективность подсистемы в момент смены действия.

Рассмотрим два варианта подражания. В первом варианте подсистемы изменяют действия случайным образом через интервалы, распределенные по случайному закону, с математическим ожиданием m_τ и дисперсией σ^2 . Для областей A и B имеем

$$m_{\tau_A} = m_{\tau_0} (e^{-\alpha_1 \mathcal{E}} + e^{\alpha_2 / \mathcal{E}}) \quad \text{и} \quad m_{\tau_B} = m_{r_0} e^{\beta \mathcal{E}}.$$

m_{τ_A} сначала падает, а с уменьшением \mathcal{E} неограниченно возрастает.

Поведение других подсистем учитывается при смене действия следующим образом. Вероятность p_{ij} перехода i -той подсистемы к j -тому действию в момент времени t

$$p_{ij} = [N_j(t) / N + p_1(1 - N_j(t) / N)],$$

где N_j — число подсистем, выполняющих j -тое действие в течение времени выполнения i -той подсистемой действия, предшествовавшего смене:

$$f(t) = \frac{1}{\sqrt{2\pi\sigma_{\tau_i}}} \int_0^t \exp\left\{-\frac{[s - m_{\tau_i}(s)]^2}{\sigma_{\tau_i}^2}\right\} ds.$$

Таким образом, каждая подсистема j -тое действие ориентирует на число выполняющих его подсистем и, кроме того, применяет свободный поиск, в простейшем случае $p_i = 1 / G$.

Во втором варианте поступают следующим образом. Ранжируют действия u_j по признаку взаимного различия и упорядочивают множество G так: $u_1 \varepsilon_{12} u_2 \varepsilon_{23} \dots \varepsilon_{j-1j} u_j \varepsilon_{j,j+1} \dots \varepsilon_{n-1,n} u_n$, где $\varepsilon_{j,j+1} < 1$ — числовые оценки различия, причем $(\forall_j)(\varepsilon_{j-1,j} < \varepsilon_{j-1,j+1})$.

Далее примем

$$p_{ij} = k_j p_2 \sum_{j=1}^n \sum_{k=1}^N \frac{\xi_{kj} t_{kj}}{p_{j1}},$$

где k_j — нормирующий множитель $\sum_{j=1}^n \frac{p_{ij}}{p_2} = 1$;

$$\xi_{kj} = \begin{cases} \text{если } k\text{-тая система производит } u_j\text{-тое действие;} \\ \text{если } k\text{-тая система не производит } u_j\text{-того действия;} \end{cases}$$

t_{kj} — время, в течение которого k -тая система производит u_j -тое действие; p_{j1} — различие между u_1 -тым в u_j -тым действиями.

Процесс адаптации S_i и X протекает следующим образом. Вначале все S_i выбирают действие из алфавита G случайно. Те системы, которым сопутствовала удача, сохраняют выбранный способ действия, остальные изменяют его в соответствии с распределением вероятностей, которое изменяется ввиду появления эффективных образцов. Число «удачливых» подсистем увеличивается до тех пор, пока растет эффективность S , затем устанавливается динамическое равновесие. Так как не все S_i действуют одинаково, стабильным будет распределение функций, при котором

$$\begin{aligned} \mathcal{D} = \sum_{j=1}^N \mathcal{D}_i(u_j) &= \sum_{i=1}^{N_1} \mathcal{D}_i(u_1) + \sum_{i=1}^{N_2} \mathcal{D}_i(u_2) + \dots \\ &\dots + \sum_{i=1}^{N_j} \mathcal{D}_i(u_j) + \dots + \sum_{i=1}^{N_G} \mathcal{D}_i(u_G) \Rightarrow \max, \end{aligned}$$

где N_1, N_2, \dots, N_G — число подсистем, выполняющих действия u_1, u_2, \dots, u_G соответственно.

Таким образом, ведущим фактором «естественного сбора» в групповом взаимодействии является подражательный механизм

выбора индивидуальных действий. Ориентируясь на индивидуальный успех (подражание действию, увеличивающему индивидуальную эффективность), каждая подсистема повышает эффективность системы и наращивает свои силы для дальнейшего взаимодействия со средой, а если нужно — для сопротивления и борьбы.

Случайность формирует изменчивость, расширяет диапазон поиска, а подражание — целесообразность.

Адаптация требует определенного времени и, если за это время среда изменит свои свойства, процесс может разрушиться. Это не означает, что эффективность S не может быть высокой; нестационарный процесс непрерывного изменения действий S может сформировать достаточно эффективное совместное поведение S_i . Все зависит от соотношений свойств S и X .

Если бы существовала надсистема S_0 , способная учесть все эти свойства и осведомленная о критерии эффективности \mathcal{E} системы S , она могла бы (по крайней мере в принципе) определить такую программу распределения действий между S_i , при которой достигалась бы максимальная эффективность S . Изменение X потребовало бы новой программы и т. д. Это означало бы оптимальное управление системой S со стороны S_0 . Однако неизвестно, что произойдет, если в какой-то момент времени S отключится и окажется предоставленной самой себе: при оптимальном распределении функция системы максимально эффективна, но целесообразная смена функций при изменении среды может оказаться недопустимо продолжительной. Поэтому «оптимальное» управление бывает невыгодным: система не может приспособиться к новому изменению среды. Это, например, может произойти так: все S_i будут выполнять весьма эффективные в среде X_1 одинаковые действия в области, далекой от той, которая окажется целесообразна в среде X_2 . При изменении среды ни одна подсистема не достигнет целесообразного действия, так как преждевременно израсходует энергоресурс на случайный поиск, а подражать некому.

При малых возможностях случайного поиска и интенсивном подражании система теряет ресурс и в течение длительного времени малоэффективна. Поэтому в часто изменяющихся и особенно нестационарных средах системы с малой «индивидуальной свободой» могут «не выжить». С увеличением вероятности случайного поиска длительность начального этапа адаптации сокращается, зато подсистемы медленнее наращивают эффективность, что также ограничивает их возможности к адаптации в случайных

средах. Следовательно, у систем со слабым коллективизмом живучесть недостаточна. Существует оптимальное соотношение между индивидуальной свободой и подражательной способностью, обеспечивающее максимальную эффективность.

Возможности адаптивного управления расширяются с увеличением глубины памяти подсистем. В дополнение к предыдущему примем, что переход подсистемы от u_j -того к u_k -тому действию требует затрат энергии Q_{jk} . После m переходов эффективность i -той подсистемы определяется рекуррентной формулой

$$\mathcal{E}_i(m) = \mathcal{E}_i(m-1) - Q_{jk} = \mathcal{E}_i(m-l) - \sum_{j=1}^l Q_j, j, k = 1, 2, \dots,$$

где Q_j — прирост энергии за время j -того действия.

Специализированная система с большим ресурсом и высокой интенсивностью управления, но в силу специализации с большим временем адаптации (запаздывания) легко справляется с небольшим увеличением противодействия, однако гибнет, если это противодействие неожиданно и сильно возрастает. Менее специализированные системы имеют меньший ресурс, но в силу больших способностей к адаптации (меньше запаздывание) легче переживают катаклизм. Это позволяет им в дальнейшем приспособиться к сильно противодействующей среде. Если эффективность исчислять на малом интервале времени T , то преимущество специализированных систем неоспоримо, при $T \rightarrow \infty$ (условно) преимущество за адаптивными системами.

Исследовать динамику адаптивного управления можно при помощи системы взаимодействующих автоматов. Каждый автомат предельно прост, поведение его элементарно, тем не менее, коллективное поведение оказывается настолько гибким и многогранным, что производит впечатление разумности.

Серьезное усложнение адаптивного управления большими системами состоит в том, что наряду с подражанием в процессе случайного поиска обнаруживаются и закрепляются действия не схожие, а дополняющие, т. е. такие, которые усиливают деятельность других подсистем и системы в целом. В дальнейшем могут включаться новые подсистемы, выполняющие другие дополняющие действия, что еще больше увеличивает совместную эффективность. Формируется большая группа подсистем, внутри которой возникает распределение функций. Если среда изменяется медленно, специализация действия групп сможет сохраняться надолго, закрепляться и совершенствоваться.

Ввиду высокой эффективности системы потребность в широком поиске отпадает, подсистемы будут осуществлять поиск только вблизи закрепившегося рационального действия, которое станет стабильной функцией. Качество выполнения функции повышается до тех пор, пока она способствует повышению эффективности системы. Так, начиная с автономных случайных действий, посредством механизма подражания формируются целенаправленность и эффективное поведение в сложной ситуации. Целенаправленность эта слепа и в неблагоприятно сложившейся ситуации может усилить неблагоприятный результат. Если вероятность ошибочной ориентации процесса невелика, рассмотренный механизм действует эффективно.

Из всех способов формирования целенаправленности подражание требует наименьшего информационного ресурса: программу действий не нужно хранить в памяти, она сосредоточена во внешнем образце, в группах с распределением функций одну и ту же функцию могут выполнять несколько подсистем (со сдвигом во времени) и служить друг для друга образцом. Направленное распределение функций формируется эффективнее, если в «действиях сотрудничества» имеется общий фрагмент — образец для подражания. Схема адаптивного управления показана на рис. 7.17.

Рефлексивное управление. Пусть имеются две взаимодействующие системы S_1 и S_2 . Сущность рефлексивного управления состоит в том, что взаимодействующей системе (содействующей или противодействующей) посредством передачи информации «внушаются» определенные данные относительно $Z(t)$, которые стимулируют желательный выбор $v(t)$. В распоряжении системы S_1 помимо u_1 имеется воздействие $v_1 \in V_1$ и этот способ влияния на систему S_2 используется не только для эффективного воздействия, но и для передачи системе S_2 такой информации, чтобы натолкнуть S_2 на выбор определенного $v_2(t)$. Вместо компенсации v_2 (как в программном управлении) или адаптации к v_2 нужно каким-либо способом заставить вторую систему повлиять на v_2 так, как это нужно первой системе. При этом может оказаться, что вторая система поступает аналогично, осуществляя рефлексивное управление первой системой.

Примеры рефлексивного управления. В содействующих системах: игры « n против n », обучение, воспитание, руководство. В противодействующих системах: игры « n против n », конкуренция, спортивные соревнования, управление противником.

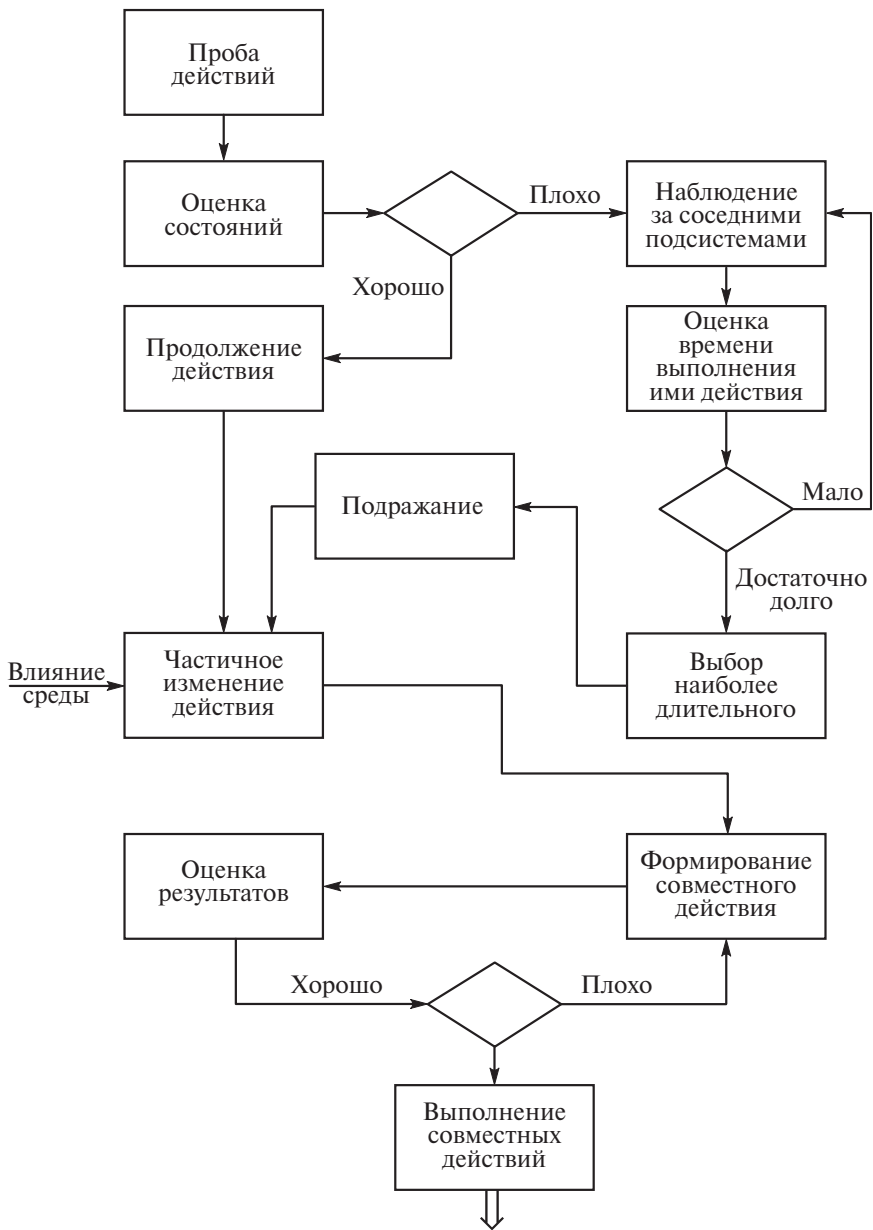


Рис. 7.17. Схема адаптивного управления сложной системой

Рефлексивное управление обладает рядом особенностей:

1. Имеет отражательный характер. Чтобы первая система могла принять решение о воздействии на вторую, в первой системе создается представление о возможной реакции второй системы путем мысленного восстановления процесса. Такая рефлексия может носить многократный характер (обладает иерархической системой рангов), но практически число рангов рефлексии при управлении невелико, поскольку высшие ранги рефлексии быстро вырождаются.

2. Велика роль мотивации, определяющей цели рефлексивного управления. Мотивация при этом извлекает из памяти опыт прошлого, заставляет мышление оценивать его результаты с учетом конкретной внешней обстановки настоящего, в которой происходит работа взаимодействующих систем.

3. Имеет недостоверный характер. Так, союзник может не понять цели и намерения управляющей системы и прореагировать в силу своего понимания цели воздействия; противник может вскрыть замысел и цель воздействия и прореагировать на него, исходя из своей оценки обстановки.

4. При выборе дозволенных приемов, способов и средств воздействия играют роль морально-этические принципы и правовые нормы, являющиеся ограничивающими условиями.

5. Чем масштабнее цели рефлексивного управления, тем сложнее и многостороннее по охвату различных сторон деятельности «управляемой» системы проводимый комплекс «управляющих» воздействий.

6. Рефлексивное управление динамично, особую роль в нем приобретает прогнозирование ожидаемых результатов; глубина рефлексивного управления определяется прочностью связей между системами и соотношением сил.

По содержанию рефлексивное управление бывает простое и сложное. При простом управлении оно достигается (в вероятностном смысле) с помощью передачи тем или иным способом информации взаимодействующей системе для принятия ею решения, выгодного управляющей стороне.

Переданная информация должна побудить такие мотивы поведения, которые стимулируют желательный выбор действий второй системы $v_2(t)$. Первая система при этом располагает информацией о возможностях второй системы, т. е. ей известно множество возможных действий второй системы v_2 , ее функционал эффективности $\mathcal{E}_2(Z_2, T)$, ограничения по ресурсам $W_2(v_2)$, а также уравнение

$$\dot{Z}_1(t) = A_1 Z_1(t) + A_2 Z_1(t - \tau_1) + u_1(t - \tau_u) + v_2(t - \tau_n) + \xi(t), \quad (7.35)$$

характеризующее изменение свойств системы с учетом взаимодействия во времени. Первая система помимо передачи информации может оказать воздействие $v_1(t) \in V_1$ на вторую систему (из множества воздействий первой системы, воспринимаемых второй системой: «обольщать выгодой», «пугать ущербом», «угрожать уничтожением» и т. д.).

Более сложный и глубокий тип рефлексивного управления проявляется при воздействии не на процесс отображения обстановки в управляемой системе, а на сам процесс принятия решения: подмена эффективного алгоритма неэффективным и, наоборот, исключение или включение оптимизации решения, снижение или повышение уровня психической устойчивости при принятии решений, выработка ориентации у второй системы и т. д. Рефлексивное управление должно охватывать как подготовку данных и процесс принятия решения, так и стоящую за решением программу действий второй стороны.

Сложное рефлексивное управление осуществляется через управление самой рефлексией, т. е. отражением окружающей действительности в управляемой системе.

Необходимо учитывать двоякую роль рефлексии — созидательную и разрушающую. Созидательная роль рефлексии заключается в определенной свободе при выборе поведения, что, в свою очередь, убыстряет адаптацию к определенному классу ситуаций на основе анализа вариантов собственных действий, особенно когда допустимое время принятия решения позволяет осуществить последовательную рефлексия высоких рангов.

Рефлексия оказывает разрушительное действие, когда мыслительная деятельность происходит по стереотипам. Если принимающий решение начнет размышлять о внутренней ценности результатов, оценивать предполагаемые последствия, то мыслительная деятельность замедляется, сопровождается большим числом ошибок и установившийся алгоритм принятия решения может оказаться нарушенным.

Особый случай представляет рефлексивное управление, осуществляемое не непосредственно, а через программы, заложенные в автоматические или автоматизированные системы управления (например, «рефлексивное управление роботами»). Принятие решения в системе S_1 о воздействии на S_2 может осуществляться на основе одного из двух принципов: по оценке возможностей системы S_2 или по намерениям S_2 .

Мотивационный механизм рефлексивного управления опирается на «внутриинтеллектуальный» подражательный механизм.

В «мозгу» формируются модели, отображающие ситуации и взаимодействующую систему, себя в этой системе, взаимодействующие системы в себе и т. д., а также модели возможных действий и поведения.

Рефлексирующая система состоит из трех частей, действующих как бы автономно. Одна часть моделирует себя, другая — взаимодействующую систему, третья — осуществляет контроль.

В результате проигрывания множества вариантов отбирается такая, в котором взаимодействующая система получает выгодные (с точки зрения рефлексирующей системы) стимулы к поведению. На основании этих стимулов строится программа рефлексивного управления, которая подвергается непрерывной оперативной корректировке. Основная трудность рефлексии состоит в том, чтобы, осуществляя контроль за моделями взаимодействующих систем, управлять ими, предоставив определенную свободу деятельности.

Средством оперативного исследования $v(t) \in V$ является планомерное применение $u(t) \in U$ таким образом, чтобы сначала выявить тенденцию $v(t)$, а затем использовать ее в интересах повышения $\mathcal{E}(Z, Z, T)$. Процесс рефлексивного управления разбивается на четыре этапа:

1. *Рефлексивная разведка* $u_1 = u_1(t)$ в течение времени $[0, t_1]$. На этом интервале наблюдается закон изменения $Z_1(t)|_{[0, T]}$ и определяется

$$\hat{v}_1(t) = \dot{Z}(t) - \dot{Z}(t - \tau_1) - f_1(Z_1(t - \tau_2)) - f_2(Z_1(t)) - u_1(t), \quad 0 \leq t \leq t_1, \quad (7.36)$$

где « $\hat{\cdot}$ » — знак оценки; $Z_1(t)$ — реализация $Z(t)$.

2. *Рефлексивное управление* (информация или дезинформация) $u_2 = u_2(t)$ на интервале $[t_1, t_2]$ с целью передачи противоположной стороне таких сведений $\hat{Z} \in \hat{Z}$, которые соответствуют замыслу S_1 :

$$\begin{aligned} \dot{\hat{Z}}(t) = & \dot{\hat{Z}}(t - \tau_1) + f_1(\hat{Z}(t - \tau_2)) + \\ & + f_2(\hat{Z}_1(t)) + u_2(t) + \hat{v}_2(t - \hat{\tau}_{2u}), \quad t_1 \leq t \leq t_2, \end{aligned} \quad (7.37)$$

при начальных условиях

$$Z(t) = \varphi_1(t), \quad \sup \tau_i \leq t \leq t_1,$$

где \hat{v}_2 и $\hat{\tau}_2$ — управления и запаздывания, которые желательно формировать путем рефлексивного управления и которые обособлены оценками \hat{v}_1 и $\hat{\tau}_1$, полученными на первом этапе.

3. *Оперативная разведка* $u_3 = u_3(t)$ в течение интервала времени $[t_2, t_3]$ с целью проверки результата рефлексивного управления и принятия решения:

$$\begin{aligned} \hat{v}_3(t) = & \dot{Z}(t) - \dot{Z}(t - \tau_1) - f_1(Z(t - \tau_2)) - \\ & - f_2(Z(t)) - u_3(t), \quad t_2 \leq t \leq t_3, \end{aligned} \quad (7.38)$$

при начальных условиях

$$Z(t) = \varphi_2(t), \quad \sup \tau_i \leq t \leq t_2.$$

4. *Оперативное управление* $u_4 = u_4(t)$ на интервале $[t_3, t_4]$ с целью получения $Z(t)|_{[t_3, T]}$, обеспечивающего эффективность системы $\mathcal{E}(Z, Z, T) \geq \mathcal{E}^*$.

Предполагается, что на втором этапе управления системе S_1 удалось «внушить» системе S_2 представление о $Z(t)$ и $u(t)$, такое, чтобы можно было выбрать соответствующее управление $u_4(t)$:

$$\begin{aligned} \dot{Z}(t) = & \dot{Z}(t - \tau_1) + f_1(Z(t - \tau_2)) + f_2(Z(t)) + \\ & + \hat{v}(t - \hat{\tau}_v) + u(t - \tau_u), \quad t_3 \leq t \leq t_4, \end{aligned} \quad (7.39)$$

при начальных условиях

$$Z(t) = \varphi_3(t), \quad \sup \tau_i \leq t \leq t_3.$$

Отметим, что $W_u(u(t))|_{[0, T]} \leq C_u^*$.

Реализация каждого этапа требует определенного расхода ресурсов, и, разумеется, чем больше их расходуется на рефлексивное управление, тем больше будет получено информации о $v(t)$, но тем меньше останется для оперативного управления, т. е. для оперативной деятельности.

Система S_2 решает аналогичную задачу, применяя рефлексивное управление. Преимущество получит та система, которая эффективнее распределит свои ресурсы по этапам. В настоящее время не разработаны методы построения рекурсивных алгоритмов распределения ресурсов, обеспечивающих рациональное решение. Но если бы такие алгоритмы существовали, это бы ничего не изменило: абсолютного оптимума быть не может, результат зависит от сочетания стратегий $u(t)$ и $v(t)$.

Основная проблема рефлексивного управления состоит в установлении правильного соответствия между результатами первого этапа и решения, принятого на втором этапе. Это означает, что навязываемая другой стороне стратегия управления должна

не только соответствовать цели (достижению $\mathcal{E} \geq \mathcal{E}^*$), но и вытекать из тенденции управления $\nu(t)$. Решающее значение на первом этапе имеет выявление, какой является система S_2 — содействующей или противодействующей. Если она противодействующая, то необходимо превратить ее в объективно содействующую (хотя бы она и стремилась к противодействию).

Большую роль играет запаздывание в управлении τ_u и τ_v . То управление, которое меньше запаздывает, имеет больше шансов на успех. В целом задача рефлексивного управления является математической, а не эвристической. Удельный вес эвристического элемента здесь не больше, чем в любой задаче управления в условиях стохастической неопределенности, где законы распределения выбираются эвристическими средствами.

Алгоритм решения задач (7.36)—(7.39) показан на рис. 7.18.

Рассмотрим случай, описываемый уравнением вида

$$\begin{aligned} \dot{Z}(t) = & \dot{Z}(t - \tau_1) + f_1(Z(t - \tau_2)) + f_2(Z(t)) + \\ & + \xi(t, \nu(t - \tau_v)) + u(t - \tau_u). \end{aligned} \quad (7.40)$$

В этом уравнении случайный процесс $\xi(t)$ зависит от управления $\nu(t)$. Типовые случаи такого влияния — изменение математического ожидания и дисперсии периода автокорреляции $\xi(t)$.

Действие $\nu(t)$ не является строго целенаправленным, оно относится к случайному процессу и, следовательно, рефлексивное управление применимо здесь только частично. Вместе с тем выявление статистических свойств $\xi(t)$ позволяет использовать теорию статистических решений (и, следовательно, программное управление) в сочетании с рефлексивным управлением. В процессе выработки управления (назовем его *ситуационным управлением*) на $\nu(t)$ можно влиять только через $\xi(t)$.

Показателем, определяющим ситуационный риск, является

$$P_c = \hat{\mathcal{E}}(Z, \dot{Z}, T) - \min \mathcal{E}(Z, \dot{Z}, T),$$

где (Z, \dot{Z}, T) — среднее значение ожидаемой эффективности с учетом оперативных и случайных факторов.

Решение (7.40) разбивается на следующие этапы:

1) *предварительная разведка* — выявление свойств $\xi(t, \nu(t))$ и их статистическая оценка;

2) *ситуационная разведка* — выявление свойств $\nu(t)$ и τ_v (обнаружение и распознавание $\nu(t)$);

3) *программно-рефлексивное управление* — воздействие на $\nu(t)$;

4) *оперативная разведка*, устанавливающая результат программно-рефлексивного управления и уточняющая $\xi(t, \nu(t))$;

5) *ситуационное управление*, обеспечивающее достижение $\Theta(Z, \dot{Z}, T) \geq \Theta^*$ при $R_s < R^*$.

Решение (7.36)—(7.39) носит циклический характер: если не хватает времени для решения или ресурсов для реализации требуемого числа циклов, решение приходится не вычислять, а принимать.

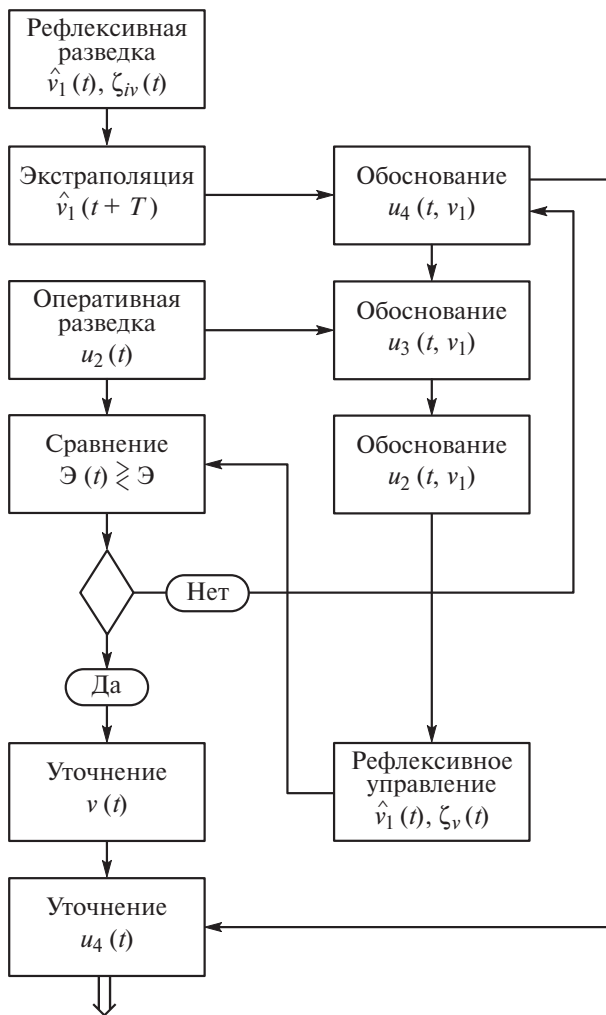


Рис. 7.18. Схема рефлексивного управления сложной системой

Заметим, что в случае (7.35)—(7.39) не предполагается взаимного рефлексивного управления от второй системы S_2 , которая не знает о взаимодействии с S_1 и строит свои стратегии независимо, исходя из свойств случайного процесса $\xi(t)$, который определял ее эффективность, $\mathcal{D}_2 = \mathcal{D}_2(t, \xi(t))$. Тем не менее внешнее проявление этих стратегий может носить характер взаимодействия и выглядеть как содействие или противодействие. Поэтому приходится принимать решения, аналогичные предыдущему случаю (7.40), и только после обнаружения истины переходить к ситуационному управлению (рис. 7.18).

Важнейшим показателем способности к осуществлению рефлексивного управления является *ранг рефлексии*. Если система моделирует только свое поведение, не считаясь с взаимодействующей системой, то это рефлексия нулевого ранга. Учет поведения взаимодействующей системы, рассматриваемой как системы с нулевым рангом рефлексии, означает рефлексии первого ранга. Если система приписывает взаимодействующей системе первый ранг рефлексии и, исходя из этого, строит свое поведение — имеют дело со вторым рангом рефлексии. Математически это записывается для системы S так:

— *нулевой ранг рефлексии*:

$$\dot{Z}(t) = a_{11}Z(t - \tau) + a_{12}Z_1(t) + u_1(t - \tau_u);$$

— *первый ранг рефлексии*:

$$\dot{Z}(t) = a_{11}Z_1(t - \tau_1) + a_{12}Z_1(t) + u_1(t - \tau_{u1}) + \hat{u}(t - \tau_v);$$

$$\dot{Z}_2(t) = a_{21}Z_2(t - \tau_2) + a_{22}Z_2(t) + \hat{u}_2(t - \tau_{u2}),$$

где \hat{u}_2 — оценка u_2 первой системой;

— *второй ранг рефлексии*:

$$\dot{Z}(t) = a_{11}Z_1(t - \tau_1) + a_{12}Z_1(t) + u_1(t - \tau_{u1}) + \hat{v}_2(t - \tau_v);$$

$$\dot{Z}(t) = a_{21}Z_2(t - \tau_2) + a_{22}Z_2(t) + u_2(t - \tau_{u2}) + \hat{v}_1(t - \tau_v),$$

где \hat{v}_2 — оценка v_2 , выполняемая системой S_1 ; \hat{v}_1 — предполагаемая оценка v_1 системой S_2 , которой первая приписывает первый ранг рефлексии.

В принципе, чем выше ранг рефлексии, тем лучше. Но реально дело обстоит сложнее. Рефлексия может быть различной. Скажем, одно лицо обладает высоким рангом рефлексии в области разработки АСУ ТП, другое — АСУП, третье — ОАСУ, четвертое — организационного управления и т. д. Поэтому, хотя способность к рефлексии определяется интеллектуально-волевым комплексом, реальное значение имеет интегральная рефлексия управляющей системы как целого, т. е. системный гомеостаз управления.

7.6. Эргатические системы управления

7.6.1. Особенности человеко-машинных (эргатических) систем управления

Эргатические системы управления (ЭСУ) — это системы, которые включают в качестве элементов как технические системы, так и людей, взаимодействующих с этими системами.

Для эффективного функционирования подобных систем необходимо выбирать рациональные способы взаимодействия людей с техникой на основании выводов эргономики.

Эргатические системы управления делятся на простые, такие, как «автомобиль — водитель», «самолет — летчик», «ЭВМ — исследователь», «управляемый объект — оператор» и т. п., и большие сложные, которыми являются, например, автоматизированные системы управления (АСУ). Различают два основных типа АСУ: системы организационно-экономического, или административного, управления и системы управления технологическими процессами. Для первых объектами управления являются предприятия, отрасли народного хозяйства, министерства, ведомства, т. е. коллективы людей, которые используют различные машины, процессы, приборы, устройства. В АСУ ТП основной формой передачи информации являются различные сигналы (электрические, световые, механические и др.), в системах же организационно-экономического управления основная форма передачи информации — документ. В настоящее время наметилась тенденция слияния двух видов систем в единые интегрированные системы управления, тем самым грани между ними до известной степени стираются.

Особенностью эргатических систем является то, что в контур управления, т. е. в управляющую систему, включен сам человек-оператор или коллектив людей-операторов.

Особенности управления ЭСУ состоят в том, что психофизиологические свойства человека-оператора должны быть включены в параметры (свойства) управляющей системы.

Закон управления для таких систем также может быть спроектирован заранее с гарантией качества управления, как и в технических системах. Если функционирование ЭСУ происходит в условиях неопределенности, то качество управления обеспечивается качеством работы человека-оператора.

Более содержательное обобщение особенностей ЭСУ представлено в табл. 7.1.

Таблица 7.1

Группы особенностей	Особенности и их сущность
Функциональные	Наличие общей задачи и единой цели функционирования для всей системы. Сложность поведения, связанная со случайным характером внешних воздействий и большим количеством обратных связей внутри системы. Устойчивость к внешним и внутренним помехам и наличие самоорганизации и адаптации к различным воздействиям. Надежность системы в целом, построенной из неабсолютно надежных компонентов. Способность к развитию, выражающаяся в способности изменять функции и структуру
Структурные	Большое количество взаимодействующих частей или элементов, составляющих систему — целостное образование. Возможность выделения групп взаимодействующих элементов-подсистем, имеющих свое специальное назначение и цель функционирования. Наличие иерархической структуры связей подсистем и иерархии критериев качества функционирования всей системы. Высокая степень неоднородности состава элементов. Большая территориальная рассредоточенность подсистем (элементов). Динамичность структуры
Изготовления	Значительные затраты на разработку и изготовление. Многообразие возможных допустимых вариантов построения и функционирования системы. Необходимость привлечения для проектирования, создания системы многих научных дисциплин. Несоответствие проектных решений, определенных в документации, реализованным проектным решениям из-за расхождения моделей разработчиков на этапах проектирования. Необходимость ввода в строй одновременно всех элементов
Эксплуатационные	Большой объем циркулирующей в системе информации, эффективная обработка которой вручную практически невозможна. Осуществление прогноза последствий нештатных (аварийных) ситуаций. Невозможность достоверно прогнозировать воздействие на систему непрерывно изменяющейся окружающей среды вследствие неполноты информации о возможных изменениях в среде за период жизненного цикла системы. Необходимость развитой инфраструктуры, обеспечивающей ремонт и восстановление компонентов ЭСУ. Многократное частичное изменение структуры и состава системы в процессе ее функционирования, связанное с непрогнозируемыми изменениями внешней среды, уточнением параметров самой системы и целей ее функционирования

Группы особенностей	Особенности и их сущность
Эргономические	Основной функцией человека в ЭСУ является управление. Способность человека оперировать нечеткими представлениями, воспринимать сложные объекты, процессы или явления как единое целое. Умение творчески, гибко действовать в сложных непредвиденных ситуациях в условиях недостаточной или неполностью достоверной информации. Способность переходить от одних технологий управления к другим в зависимости от конкретных управленческих ситуаций. Непредсказуемость поведения, настроения, работоспособности человека. Субъективный характер принимаемых человеком решений, особенно в условиях острого дефицита времени и отсутствия достаточно полной информации, возможность случайных и преднамеренных ошибок при обработке информации или формировании информационных сообщений. Низкая вычислительная мощность человека, неспособность воспринимать большое число вариантов исходов, прогнозировать результаты принятых решений

7.6.2. Инженерно-психологические проблемы создания и эксплуатации эргатических систем управления

По мере усложнения ЭСУ все ощутимее становятся потери от несоответствия характеристик технических средств возможностям человека. При этом основные трудности связаны не только с совершенствованием технических и программных средств, но и с недостаточным развитием методов учета человеческого фактора при создании и эксплуатации сложных ЭСУ.

Можно выделить следующие инженерно-психологические проблемы, требующие решения в процессе создания и эксплуатации сложных ЭСУ.

Первая проблема: компенсация ошибочных (в первую очередь непреднамеренных, но также и преднамеренных) действий человека, влекущих за собой негативные последствия для функционирования ЭСУ.

В ЭСУ должны быть учтены: забывчивость оператора, возможность его ошибки, непостоянство внимания и т. п.

Если решение, принятое человеком, может привести систему в аварийный режим (контроль осуществляет сама система), то это решение не должно восприниматься, о чем система должна сигнализировать оператору.

Подобные действия в состоянии выполнять лишь сложная система с хорошо развитыми средствами интеллектуальной поддержки операторов.

Вторая проблема: формализация психологических аспектов мыслительной деятельности человека в процессе выработки решений по реализации какой-либо задачи и учет их в системах искусственного интеллекта (ИИ), формирующих соответствующие решения.

Проблема формализации основных схем поведения и психологических характеристик человека-оператора связана с попытками создания математических моделей деятельности человека. Это обусловлено прежде всего необходимостью создания единого языка описания функционирования системы в целом, причем принято считать, что разработка математических моделей деятельности является одним из перспективных путей решения этой проблемы.

Вместе с тем в процессе проектирования деятельности подчас целесообразно автоматизировать те или иные функции человека-оператора, т. е. поручить выполнение их техническим средствам, носящим в себе черты модели, соответствующей деятельности человека.

Движение любого объекта обусловлено его собственными свойствами и действием на него управляющих сил. В целом объект и система управления им образуют динамическую систему, движение которой может быть описано дифференциальными уравнениями. Класс таких дифференциальных уравнений определяется динамикой конкретной системы. Обычно динамическая система описывается сложной системой нелинейных дифференциальных уравнений высокого порядка со случайными параметрами, аналитического выражения для которых до сих пор не существует.

Для всех систем, за исключением простейших, истинное явление можно описать с помощью уравнений лишь приближенно. Это обусловлено тем, что мы не знаем всех факторов, влияющих на систему, или получаем слишком громоздкие уравнения, которые современными средствами решать весьма сложно. Обычно рассматривается небольшое число аспектов поведения ЭСУ.

Основной принцип построения моделей заключается в том, что результаты, получаемые с помощью моделей, должны соответствовать экспериментальным данным и, кроме того, модель должна давать возможность получать новую информацию о системе или объекте.

Третья проблема: определение «границ возможного» в деятельности человека и возможностей техники для оптимального распределения функций между ними.

Пределы функционирования сложных систем определяются условиями и воздействиями, приводящими к срыву деятельности.

В этом смысле срыв операторской деятельности является одной из глобальных проблем, стоящих перед проектировщиками сложных ЭСУ. Цель проектирования прежде всего состоит в том, чтобы избежать, исключить возможность аварий (прекращения деятельности) современных систем, которые неотвратимы при срыве деятельности человека-оператора.

Степень согласованности характеристик технических средств с психофизическими характеристиками человека-оператора определяет эффективность деятельности. Срыв деятельности характеризуется нулевой и даже отрицательной эффективностью. Он может наступить, например, при повышении темпа поступления информации.

Выделяют следующие аспекты срыва операторской деятельности, исследование которых необходимо при проектировании:

- определение критических значений потока информации в зависимости от способов деятельности;
- оценка влияния автоматизации процессов управления на устойчивость операторской деятельности;
- выявление «слабых» звеньев в структуре деятельности в целях проектирования наилучших способов деятельности;
- определение стадий (фаз) срывов деятельности с выявлением необходимых перестроек, переходов от одного способа деятельности к другому при обнаружении возможности срыва деятельности;
- определение допустимых границ изменений функционального состояния оператора;
- определение границ между областями устойчивой деятельности и срывов деятельности, т. е. определение тех требований, которые проектировщики систем могут предъявить к человеку-оператору в соответствии с функциональными возможностями операторов конкретных систем.

Четвертая проблема: формализация основных схем поведения (их еще называют алгоритмами или последовательностями деятельности) человека в зависимости от сложившейся ситуации и предложение оператору (лицу, принимающему решение) лучшей (по какому-то критерию) из них.

К этому классу задач относятся:

- классификация типов поведения,
- моделирование поступков,
- определение траектории поведения,
- формирование поведения и др.

Пятая проблема: определение психологических характеристик человека и их диапазонов для обеспечения комфортного общения человека и техники, использование современных технологий и техники для уменьшения потребности адаптации людей к системе.

Современные средства взаимодействия человека и техники представляют собой сложный комплекс, включающий различные компоненты: планирование, информирование и управление общением; формализацию облика информации, интерпретацию сообщений; представление, обработку данных и принятие решения; обеспечение надежности и др.

Основной тенденцией перспективного развития и совершенствования средств взаимодействия является создание адаптивных интеллектуальных систем, учитывающих целесообразное распределение нагрузки между искусственным интеллектом ЭВМ и интеллектом.

7.6.3. Специфика анализа и синтеза эргатических систем управления

Задачи анализа ЭСУ решаются, как правило, на стадиях их эксплуатации, транспортировки, снятия с эксплуатации и списания. Анализ используется также для исследования вариантов вновь создаваемых ЭСУ с целью выбора лучшего варианта.

Задачи синтеза ЭСУ решаются на этапах их проектирования и создания.

К задачам синтеза ЭСУ относится процесс принятия решения о целесообразности того или иного нововведения и обоснованного выбора направлений предпроектных исследований.

Структурный анализ и синтез ЭСУ направлен на решение следующих задач: описание состава организации ЭСУ и построение ее структурной схемы; формирование рационального числа уровней управления; определение состава и мест размещения звеньев управления; определение функций отдельных подразделений, их структурной схемы; создание рациональной сетевой структуры, обеспечивающей требуемые характеристики устойчивости и оперативности управления; исследование отдельных технических ус-

тройств, входящих в состав ЭСУ; учет психологических характеристик человека-оператора при создании структур ЭСУ; построение обобщенной структурной информационной модели ЭСУ; описание материальных, вещественных и информационных связей.

Функциональный анализ и синтез ЭСУ направлен на решение следующих основных задач: анализ функций управления в структурных подразделениях, выбор состава автоматизируемых функций и определение их взаимодействий; определение способов сбора, хранения и отображения информации, необходимой для функционирования системы управления; определение порядка обработки информации с целью принятия управленческих решений и доведения их до исполнителей; создание системы контроля за доведением решений и их исполнением, а также оценка результатов выполненных решений; учет психологических факторов оператора при управлении сложными ЭСУ.

Особенности *информационного анализа и синтеза ЭСУ* заключаются в исследовании и поиске рациональных способов сопряжения оперативного персонала с техническими средствами и решаемыми задачами управления.

При этом исследуются способы предоставления, ввода и вывода информации, определяется необходимый и достаточный состав формализованных сообщений (указаний, приказов, подтверждений, донесений), обеспечивающих эффективное управление.

Наряду с этим решаются общие задачи анализа и синтеза информационного обеспечения, включающего способы классификации и кодирования информации, языковые средства описания данных, унифицированную систему документации, программные средства обработки информационных массивов, базы и банки данных.

Параметрический анализ и синтез ЭСУ связаны с исследованием и количественной оценкой разнообразных свойств и различных условий функционирования оперативного персонала и используемых технических средств управления. Поэтому процедура выбора показателей, достаточно полно отражающих свойства подобных систем, довольно сложна, и в настоящее время нет четко установленного перечня подобных показателей. На практике для исследования свойств данных систем и их элементов используется несколько сотен различных показателей: количественные, качественные, экономические, технические, общие, комплексные, частные, основные, вспомогательные, специфические, исходные, производственные и т. п.

Из большого количества показателей качества работы людей в ЭСУ наиболее часто используют быстроедействие, напряженность, экономичность, надежность.

Быстроедействие оператора характеризуется длительностью рабочего цикла

$$T_{\text{ц}} = \tau_{\text{пр}} + \sum_{i=1}^n \tau_3^{(i)}, \quad (7.41)$$

где $\tau_{\text{пр}}$ — время производства работ, $\tau_{\text{пр}} = \tau_{\text{ож}} + \tau_{\text{р}}$; $\tau_{\text{ож}}$, $\tau_{\text{р}}$ — время ожидания всех операций работы без пауз между ними соответственно; $\tau_3^{(i)}$ — время между моментами окончания i -той операции и началом $(i + 1)$ -й.

Часто время $\tau_{\text{р}}$ определяют по формуле

$$\tau_{\text{р}} = \frac{\alpha + 1}{R_0},$$

при $\alpha \cong 0,2$ с; $1/R_0 = 0,15 \div 0,35$ с/бит, т. е. в предположении, что в среднем начало работ запаздывает относительно момента выдачи задания на 0,2 с, при средней производительности оператора, равной 1/0,15 — 1/0,35 бит/с.

Быстроедействие конкретного оператора может быть определено с использованием тренажера или хронометрирования на реальном рабочем месте.

Напряженность работы оператора определяется степенью функционального напряжения его организма, нервного или физического, по формуле

$$\gamma = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{y_i}{y_{i_{\text{max}}}} \right)^2}, \quad (7.42)$$

где y_i , $y_{i_{\text{max}}}$ — физиологические количественные показатели напряженности работы в реальных и экстремальных условиях соответственно.

Экономичность оператора определяется как отношение количественного результата его работы к затратам на подготовку и поддержание квалификации оператора.

Надежность оператора характеризует его свойство выполнять заданные функции в течение определенного времени при заданных условиях работы. Показатели надежности человеко-машинных систем должны учитывать свойства и человека, и машины. Однако если методы оценки надежности машин достаточно хорошо разработаны, то расчет надежности людей-операторов

представляет известные трудности и составляет в настоящее время одну из актуальных проблем в теории ЭСУ.

Функциональным понятием теории надежности является понятие отказа — случайного события, состоящего в том, что элемент (оператор) полностью или частично утратил свою работоспособность, в результате чего заданные ему функции не выполняются.

Устойчивые отказы операторов называют биологическими, а временные — психологическими. Причиной первых являются болезни, чрезмерное утомление, засыпание на рабочем месте и т. п.; причиной временных отказов являются случайные ошибки нормального рабочего оператора.

Одним из показателей надежности операторов служит вероятность безотказной работы, определяемая как

$$P_{\text{оп}}(t) = K_{\text{оп}} P_{\text{б}}(t) + P_{\text{п}}(t), \quad (7.43)$$

где $P_{\text{б}}, P_{\text{п}}$ — вероятности биологической и психологической надежности оператора соответственно; $K_{\text{оп}}$ — коэффициент готовности оператора.

$$K_{\text{оп}} = \frac{1 - T_0}{T},$$

где T_0 — время отсутствия оператора на рабочем месте; T — общее время работы.

С учетом $P_{\text{оп}}$ вероятность безотказной оперативной работы человеко-машинной системы определяют мультипликативным показателем

$$P(t) = P_{\text{техн}}(t) \times P_{\text{оп}}(t) \times P_{\text{СВ}}(t), \quad (7.44)$$

где $P_{\text{техн}}(t)$ — вероятность безотказной работы машины; $P_{\text{СВ}}(t)$ — вероятность своевременного выполнения работы.

Выделенные показатели качества работы операторов позволяют определить следующие основные *пути повышения эффективности* их работы:

- обеспечение необходимой степени профессиональной подготовки операторов;
- проектирование аппаратуры в соответствии с требованиями инженерной психологии;
- обеспечение контроля за правильностью действий операторов;
- правильный выбор режимов труда и отдыха операторов;
- исключение информационных перегрузок операторов;
- обеспечение хорошего психологического климата в коллективе операторов и т. п.

Экспериментально установлено, что определенная часть операторов, обладающих соответствующей квалификацией, часто обнаруживают свою несостоятельность при возникновении аварийных (критических) ситуаций.

Статистика указывает также на значительный процент аварий из-за неправильных действий операторов (человеческого фактора): например, в строительстве — больше 70%, в авиации — больше 80%. Таким образом, очень часто ошибочные действия операторов приводят к лавинообразному развитию аварийных ситуаций, к порче и потере дорогостоящей техники, а также к гибели людей.

7.6.4. Типовые противоречия в процессе создания новых эргатических систем управления

При разработке сложных эргатических систем управления наиболее ответственным этапом является процесс принятия решения о целесообразности того или иного нововведения и обоснованного выбора направлений и методов предпроектных исследований. Основной целью деятельности на этом этапе является снижение неопределенности и степени риска, связанных с внедрением дорогостоящих нововведений. Под *нововведением* понимается деятельность, направленная на изменение ЭСУ на всех стадиях ее жизненного цикла и приводящих при этом к разнообразным последствиям в различных видах окружения ЭСУ.

Принятие решения о внедрении нововведения связано со знанием противоречий, которые необходимо разрешить в процессе создания новых систем. Анализ основных тенденций в создании перспективных ЭСУ позволяет сформулировать ряд противоречий, от разрешения которых существенно зависит как облик системы, так и работа по ее созданию. В табл. 7.2 приведены типовые противоречия, разрешаемые в процессе создания новых ЭСУ.

Таблица 7.2

Тип противоречия	Смысл противоречия
Потребность — возможность	С одной стороны, существует потребность в создании новой системы, а с другой — отсутствует возможность ее создания. Это может быть вызвано различными причинами, например отсутствием материальной базы

Тип противоречия	Смысл противоречия
Необходимость предвидеть будущее — неопределенность будущего	При создании ЭСУ необходимо спроектировать ее облик, однако человеку не дано видеть то, чего еще нет в природе (в частности, невозможно предвидеть все последствия от производства и применения создаваемой системы)
Срок создания — темп морального старения	Новизна и сложность создаваемой системы требуют увеличения срока создания, но, с другой стороны, высокий темп НТП ведет к ее быстрому моральному старению
Темпы роста сложности систем — темпы развития методов их проектирования	Темпы роста сложности технических систем (ТС) преобладают над развитием методов их проектирования. В среднем по всем отраслям техники число подсистем ТС удваивается каждые 15 лет. При этом производительность труда в проектировании с начала века выросла всего на 20% (в производстве — на 1000%)
Сложность — надежность	С повышением степени сложности создаваемых ТС падает их надежность
Уникальность ЭСУ — необходимость индустриального подхода	Большинство ЭСУ являются единственными в своем роде, но требование высокой надежности вынуждает применять индустриальный подход к их созданию, основанный на массовости, стандартизации, унификации комплектующих изделий и технологических операций

Содержание нововведений зависит от глубины реорганизации системы или создания более перспективных систем (табл. 7.3).

Таблица 7.3

Уровень реорганизации (1)	Содержание нововведения (2)
Параметрическая реорганизация системы	Изменение свойств, параметров системы или ее элементов
Функциональная реорганизация системы	Изменение функций, содержания решаемых задач, состава целей, оперативных направлений для отдельных элементов существующей системы

Уровень реорганизации (1)	Содержание нововведения (2)
Структурная реорганизация системы	Изменения организационного или пространственного построения элементов существующих систем (варьирование структуры, направления, характера и количества связей; пересмотр мест дислокации и базирования и т. д.)
Полная реорганизация системы проектирования	Создание более перспективных систем, комплексов и отдельных образцов техники, предполагающих формирование новых или преобразование существующих организационных подразделений, осуществляющих обслуживание и применение систем подобного типа

Общую схему разрешения противоречий в процессе создания новых ЭСУ можно представить в виде последовательности формирования ответов на вопросы, стоящие перед исследователем (табл. 7.4).

Таблица 7.4

1. Выявление необходимости нововведения		Нужно ли оно в перспективе?	
		Да	Нет
Нужно ли оно сейчас?	Да Нет	Устойчивая потребность Перспективная потребность	Кажущаяся потребность Мнимая потребность
2. Выявление возможности нововведения		Возможно ли оно в перспективе?	
		Да	Нет
Возможно ли оно сейчас?	Да Нет	Устойчивая возможность Перспективная возможность	Кажущаяся возможность Мнимая возможность
3. Оценка последствий нововведений		Каковы последствия в перспективе?	
		Позитивные	Негативные
Каковы последствия в ближайшем будущем?	Позитивные Негативные	Устойчивые позитивные последствия Позитивные последствия в перспективе	Перспективно-негативные последствия Устойчиво-негативные последствия

Предлагая некое нововведение (например, замысел новой ЭСУ), исследователь задается тремя вопросами.

Первый вопрос — нужно ли оно (нововведение)?

Если в результате анализа потребности на текущий момент и прогнозирования ее изменений в перспективе выясняется, что потребность устойчива или перспективна, то делается вывод о необходимости нововведения. Если потребность оказывается кажущейся или мнимой, то фиксируется отрицательный ответ.

При положительном ответе на первый вопрос исследователь переходит ко второму вопросу.

Второй вопрос — возможна ли реализация этого нововведения?

Если в результате анализа возможности на текущий момент и прогнозирования ее изменений в перспективе выясняется, что возможность устойчива или перспективна, то на вопрос дается положительный ответ, в противном случае — отрицательный ответ. При положительном ответе на второй вопрос исследователь переходит к третьему вопросу.

Третий вопрос — каковы возможные последствия этого нововведения?

Если в результате анализа последствий на ближайшее будущее и прогнозирование их изменений на отдаленную перспективу выяснится, что последствия устойчиво негативны или становятся таковыми в будущем, то характер нововведений отрицательный.

При положительных ответах на поставленные вопросы исследователь может перейти к уточнению характеристик нововведения и к ответам на вопросы:

- какие конечные цели могут быть достигнуты посредством создаваемой ЭСУ;
- какую роль будет играть создаваемая ЭСУ в данной сфере деятельности;
- какими свойствами и строением может обладать ЭСУ при данных предположениях;
- какие принципы и стратегии могут быть положены в основу процесса создания и применения ЭСУ;
- какова степень технического и экономического риска, связанного с созданием ЭСУ?

Ответы на эти вопросы позволят исследователю перейти к формированию проблематики и комплексной программы НИР по выбранным направлениям.

7.7. Типовые организационные структуры управления производством

Организационная структура — это форма распределения задач и полномочий на принятие решений, а также форма распределения ответственности за эти принятые решения.

Рассмотрим типовые организационные структуры управления производством. Реальные структуры, как правило, представляют собой некоторую комбинацию типовых структур.

Линейная структура

Так, например, (рис. 7.19) начальник цеха управляет начальниками своих участков во всей совокупности выполняемых функций управления. Достоинства такой структуры: вся власть и, следовательно, вся мера ответственности возлагается на элемент более высокого уровня, а также невозможность получения элементами нижнего уровня противоречивых, не увязанных между собой распоряжений или команд. Недостатки: необходимость элементу верхнего уровня запоминать и обрабатывать очень большие объемы информации, причем, очень часто, достаточно специальной (специфической).

Функциональная структура

Для разгрузки линейного руководителя (рис. 7.20) в части реализации некоторых специальных функций управления создаются функциональные подразделения (это может быть и один человек). Так, например, в штате цеха может быть подразделение механика цеха, энергетика цеха, может быть бухгалтер цеха и т. п. Таким образом, функциональное управление не отменяет линейное управление, а только несколько ограничивает его деятельность. В свою очередь, функциональные подразделения могут образовывать и свою собственную линейную структуру, например механик цеха непосредственно подчиняется главному механику предприятия.

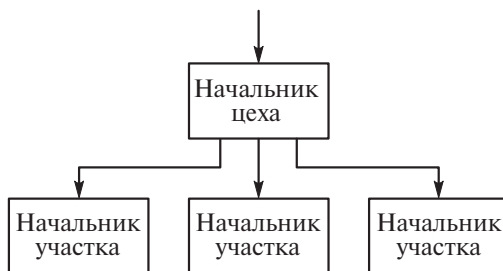


Рис. 7.19. Линейная структура

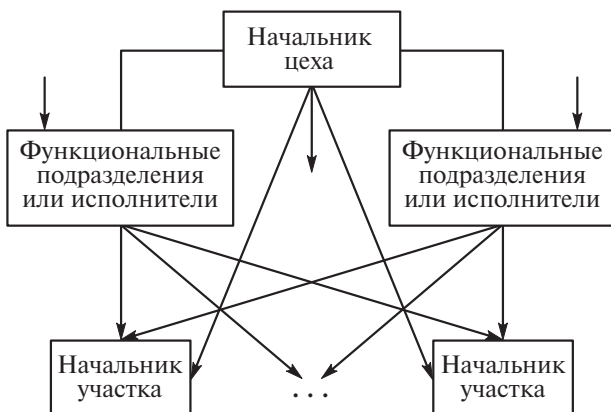


Рис. 7.20. Функциональная структура управления

Достоинство структуры — разгрузка линейного руководителя. Недосток этой структуры — возможность получения элементами нижнего уровня взаимно несогласованных распоряжений.

Линейно-штабная структура

В линейно-штабной структуре (рис. 7.21) устраняется указанный недостаток. Вся полнота власти, а следовательно, и вся мера ответственности за принимаемое решение возлагаются на линейного руководителя. Функциональные подразделения лишены непосредственной возможности воздействовать на элементы нижнего уровня. Они только готовят рекомендации для линейного

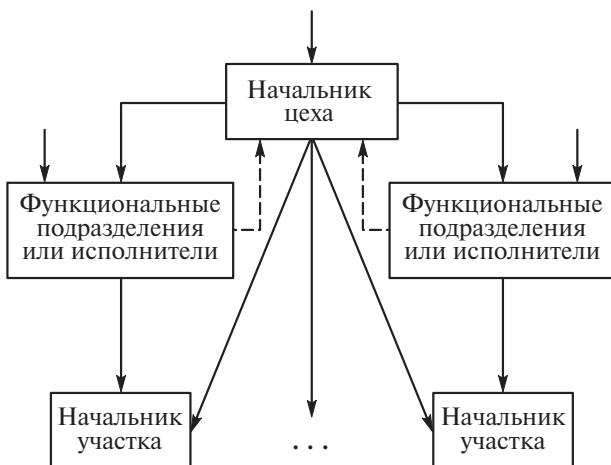


Рис. 7.21. Линейно-штабная структура

руководителя. Не следует считать, что тем самым устраняются возможные противоречия, они только переносятся на более высокий уровень.

Таким образом, формируются два основных вида управления: комплексное управление и функциональное управление.

Комплексное управление — управление во всей совокупности функций. Так, например, министерство управляет главком, отрасль — предприятием, начальник цеха — начальником участка.

Функциональное управление — это управление по реализации некоторых функций со стороны специализированных органов управления. Такими функциями, например, могут быть материально-техническое снабжение, планирование, финансы и т. п.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Перечислите особенности централизованного вида управления.
2. Что такое централизованная рассредоточенная структура управления?
3. В чем Вы видите преимущества и недостатки децентрализованного управления?
4. Раскройте понятия «иерархия» и «иерархический тип управления».
5. Дайте определение многоуровневой иерархической структуры.
6. Какова сущность страты и стратифицирования системы?
7. Объясните структуру и функции многослойной системы принятия решений.
8. Что такое «эшелон» в теории иерархического управления?
9. Расскажите о преимуществах иерархических систем управления.
10. В чем состоит и как решается проблема координации в ИСУ?
11. Раскройте принцип декомпозиции в СУ.
12. С помощью каких математических методов решаются задачи декомпозиции в СУ?
13. Какой смысл вкладывается в понятие «агрегация»?
14. Приведите примеры применения принципа агрегирования.
15. Какие Вы знаете принципы управления сложными системами?
16. Сформулируйте общую постановку задачи управления сложной системой.
17. Что такое стохастическое, адаптивное и рефлексивное управление?
18. Раскройте содержание принципа адаптивного управления с подражательным механизмом.
19. Какие основные блоки входят в схему адаптивного управления сложной системой?
20. Расскажите об особенностях рефлексивного управления.
21. Приведите этапы процесса рефлексивного управления.
22. Дайте определение эргатической системе управления.
23. Какими особенностями обладают эргатические системы управления?
24. Перечислите инженерно-психологические проблемы при создании и эксплуатации эргономических систем управления.

25. В чем вы видите сущность специфики анализа и синтеза эргатических систем управления?
26. Назовите и дайте краткую характеристику показателям качества работы человека-оператора.
27. Каковы пути повышения эффективности оператора сложной системы?
28. Расскажите содержание типовых противоречий, разрешаемых в процессе создания новых эргатических систем управления.
29. Как Вы понимаете смысл организационной структуры управления предприятием?
30. Что такое организационная структура и функциональная структура управления предприятием?
31. Какие преимущества и недостатки у линейной, функциональной и линейно-штабной форм структуры управления предприятием?

Глава 8

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ

Автоматизированное управление реализуется в автоматизированных системах управления (АСУ). АСУ по своей природе являются чрезвычайно большим классом в группе автоматизированных систем (АС). Поэтому в этой главе представлены примеры основных видов АСУ, которые дают студентам полное представление о характерных особенностях АСУ. Начиная с середины 60-х годов XX века, когда начался этап повсеместного внедрения АСУ в различные сферы народного хозяйства, и по настоящее время практически на каждом предприятии промышленности, транспорта, торговли, кредитно-финансовых и др. предприятий имеются АСУ предприятиями (АСУП) и АСУ технологическими процессами (АСУТП). На сегодняшний день эти два вида АСУ являются наиболее распространенными. Учитывая этот фактор, в восьмой главе рассмотрены различные представители АСУТП такие, как гибкие автоматические производства, системы числового программного управления технологическим оборудованием, автоматизированные системы контроля и регулирования вращающихся печей, системы управления и защиты исследовательского ядерного реактора. Кроме АСУ промышленными предприятиями процесс внедрения вычислительной техники в научную сферу привел к появлению автоматизированных систем высокой степени интеллектуализации. К таким системам относятся системы автоматизированного проектирования (САПР) и автоматизированные системы научных исследований (АСНИ). Примеры клас-

сификации, структуры, видов обеспечения, организации взаимодействия человека и ЭВМ приведены в этой главе. АСУ нашли широкое применение и в таких сферах деятельности человека, как транспорт, торговля, учебные заведения, банки. Как пример применения АСУ в таких специфических областях в главе рассмотрены автоматизированные банковские системы управления.

8.1. Автоматизированные системы управления предприятием

8.1.1. Организационная структура предприятия

Организационная структура — это форма распределения задач и полномочий на принятие решений.

Рассмотрим пример укрупненной схемы организационной структуры среднего механообрабатывающего предприятия (рис. 8.1).

Для простоты на рис. 8.1 отсутствуют, например, зам. по кадрам со своим отделом, заместитель по общим вопросам со своим отделом, гл. бухгалтер со своим отделом и т. п.

Управление основным производственным процессом на предприятии осуществляется на двух уровнях: общезаводском (межцеховом) и внутрицеховом.

Под основным производственным процессом понимается совокупность отдельных процессов по обработке и перемещению от одной технологической операции к другой предметов труда (полуфабрикатов, заготовок, деталей, узлов) в соответствии с принятой технологией и организацией производства.

Под общезаводским оперативным управлением понимается часть функций управления, которая связана с планированием, учетом, анализом и регулированием основной производственной деятельности цехов предприятия, исходя из необходимости выполнения производственных заданий.

Объектом управления на общезаводском уровне являются цеха основного производства, склады, а также экспедиционные и транспортные межцеховые службы.

Управляющими элементами на этом уровне являются соответствующие структурные подразделения заводского аппарата управления: аппарат линейного руководства предприятия, планово-диспетчерский отдел, отдел МТС, конструкторское бюро, отдел главного технолога.

Под внутрицеховым оперативным управлением понимается часть функций управления, которая связана с планированием, учетом, анализом и регулированием основного производственного процесса в рамках цеха.

Объектом управления на внутрицеховом уровне являются отдельные системы вида «человек—станок» со всем набором необходимых для осуществления технологических процессов предметов и средств труда (заготовок, инструмента, оснастки, оборудования, измерительных приборов, тары, транспортных средств и т. д.) или определенные группы таких систем (участки, отделения и т. п.).

Управляющими элементами на этом уровне являются: аппарат линейного руководства цехом или складом, планово-диспетчерское бюро или плановая группа цеха, линейное руководство внутрицеховых подразделений (участков, смен и т. п.).

Основные функции, реализуемые в процессе управления, — см. 8.1.3.

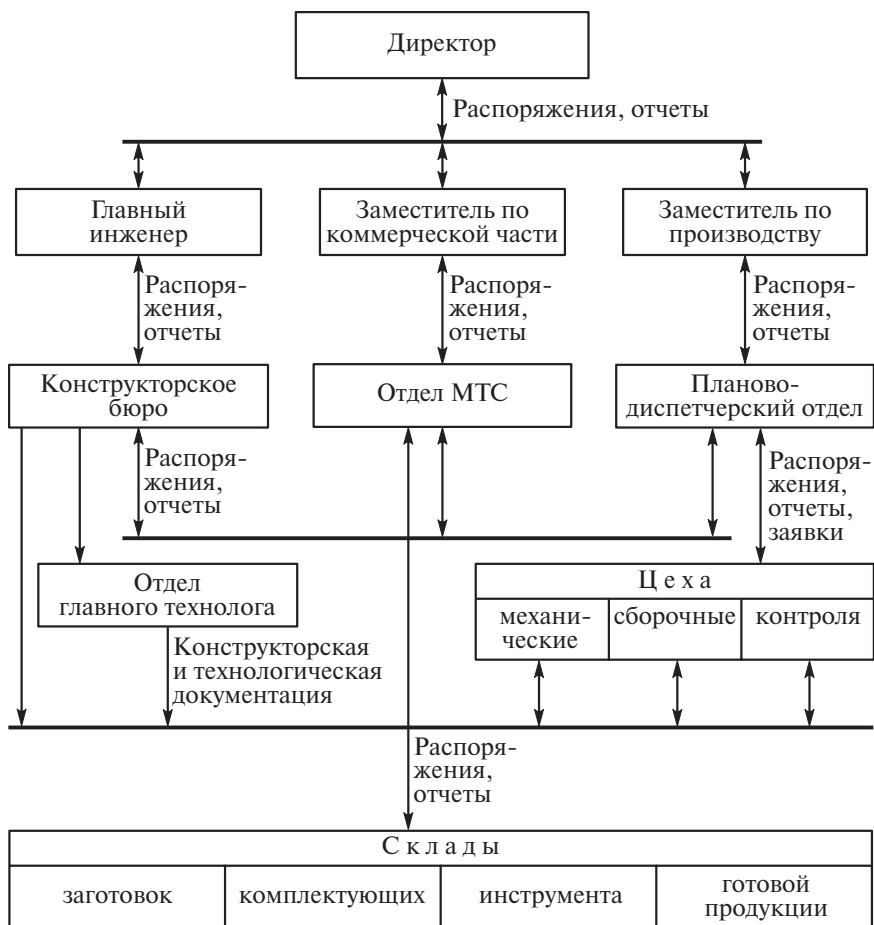


Рис. 8.1. Организационная структура предприятия

8.1.2. Пример документооборота предприятия

Существующая на предприятии система документооборота сложна и громоздка вследствие наличия большого числа различных форм документов, многоэтапности их формирования, большого числа дублирования различных показателей и т. д. Такое положение сложилось вследствие многих субъективных и объективных причин, в том числе, вследствие автономности обработки данных в подразделениях предприятия.

На рис. 8.2. представлена типичная для среднего машиностроительного предприятия группировка документов по их использованию и формированию.

Общее число документов, циркулирующих на предприятии, составляет 1224 формы. По отношению к внешней среде они делятся на входные (199 форм), внутренние (1025), т. е. формируемые на предприятии, и выходные (219), число которых входит в число внутренних (поэтому они выделены на рис. 8.2. пунктиром).

Для формирования 438 форм не требуются данные других документов (64 формы выходят за рамки предприятия) – первичные документы. Остальные 587 форм формируются на основе данных, содержащихся в других документах – производные доку-



Рис. 8.2. Документооборот предприятия

менты. Для принятия решений используется $108 + 237 + 294 = 639$ форм – активные документы. Остальные $91 + 201 + 293 = 585$ форм используются в процессе обработки информации – пассивные документы.

Анализ указанных форм документов показывает, что среднее число повторений одних и тех же показателей в различных документах равно 3,1, а общее число показателей, фигурирующих в потоке информации, превышает 17000.

Если учесть сложность потоков информации, и то, что все документы имеют копии, встречаются в нескольких подразделениях, то станут очевидными те огромные объемы информации, которые необходимо обрабатывать в процессе управления предприятием.

8.1.3. Функциональная структура АСУП

В функциональных подсистемах АСУП реализуются следующие глобальные функции:

- планирование;
- учет;
- контроль и анализ;
- регулирование.

Реализация каждой из этих функций требует решения определенного множества задач.

При решении задач планирования разрабатывается расписание деятельности объекта управления на некоторый определенный интервал времени.

При решении задач учета осуществляется сбор определенного круга данных о состоянии объекта управления и внешней среды.

При решении задач контроля и анализа производится сравнение собранного круга данных с некоторыми запланированными или нормативными величинами.

При решении задач регулирования вырабатываются соответствующие управляющие воздействия на объект управления в случае наличия отклонений, обнаруженных при решении задач контроля.

Перечисленное множество задач решается в соответствующих функциональных подсистемах.

Исходя из сложившейся практики в настоящее время принято выделять следующие функциональные подсистемы:

- технико-экономического планирования (ТЭП);
- технической подготовки производства (ТПП);
- оперативного управления производством (ОУП);
- материально-технического снабжения (МТС);
- бухгалтерского учета (БУ);

- сбыта и реализации продукции (СРП);
- управления кадрами (УК);
- управления качеством продукции (УКП) и некоторые другие.

Основные функции перечисленных подсистем непосредственно следуют из их названия.

Функциональная структура АСУП представлена на рис. 8.3.

Управленческая информация в виде планов, директив, контрольных цифр поступает в ПЭО, где она анализируется, и на ее основе разрабатывается план предприятия. На основании решения комплекса задач планирования формируются задания в дру-

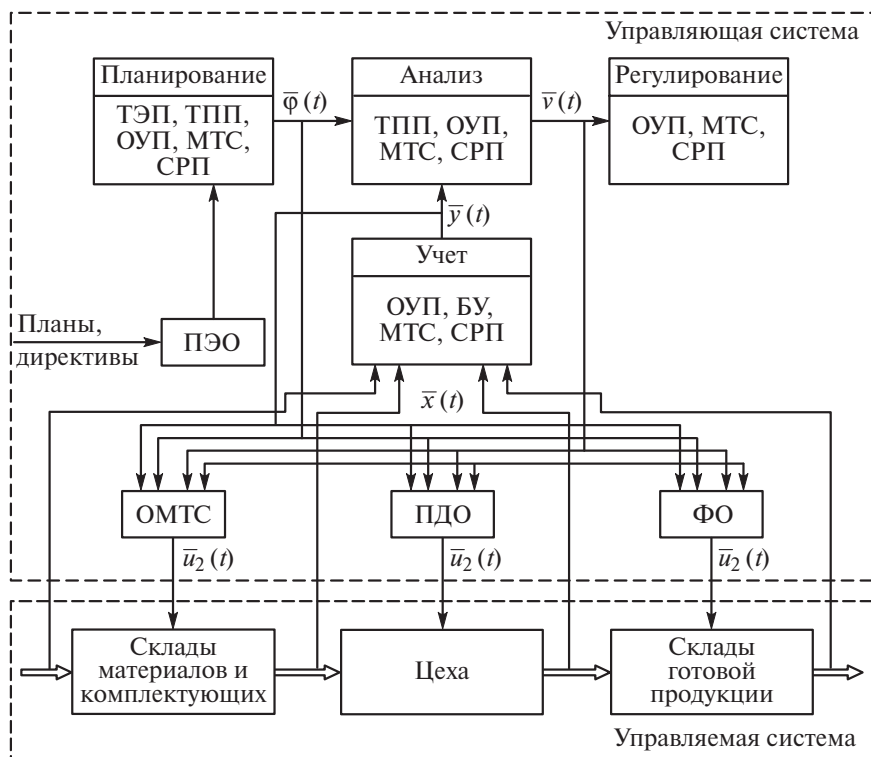
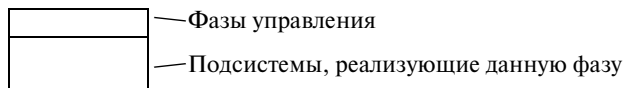


Рис. 8.3. Функциональная структура АСУП

Обозначения: \rightarrow — направление движения информационных потоков; \Rightarrow — направление движения материальных потоков; ПЭО — планово-экономический отдел предприятия; ОМТС — отдел материально-технического снабжения; ПДО — планово-диспетчерский отдел; ФО — финансовый отдел.



гие службы предприятия (вектор управляющих команд — $\varphi(t)$). С целью определения фактического состояния производства в конкретный момент времени в фазу учета поступает информация о состоянии материальных потоков (вектор $x(t)$). В фазе учета вырабатываются сводные данные о фактическом ходе производственного процесса (вектор $y(t)$). В фазе анализа эти фактические данные сравниваются с плановыми заданиями, разработанными ранее, и в случае их расхождения вырабатывается информация (вектор результатов анализа $v(t)$), направленная на ликвидацию обнаруженных расхождений. Причем конкретные управляющие воздействия $\bar{u}_1(t)$, $\bar{u}_2(t)$, $\bar{u}_3(t)$ вырабатываются в соответствующих службах предприятия.

8.1.4. Подсистема оперативного управления основным производственным процессом

В качестве примера функциональных подсистем АСУП мы рассмотрим подсистему управления основным производственным процессом.

Оперативное управление производственным процессом представляет собой комплекс информационных процессов по выработке управляющих воздействий на производственный процесс, который обеспечивает достижение производственным процессом целей, заданных подсистемой ТЭП. Таким образом, с системных позиций подсистема ОУП является объектом управления относительно подсистемы ТЭП, которая выступает в данном случае в качестве управляющей части. Как объект управления подсистема ОУП выполняет следующие функции:

- оперативное планирование;
- оперативный учет;
- оперативный контроль;
- оперативное регулирование (диспетчирование).

Оперативное планирование заключается в детализации периодически выдаваемых подсистемой ТЭП плановых показателей и в конкретизации их по времени (месяц, декада, сутки, смена) и месту (цех, участок, группа оборудования, рабочее место).

Оперативное регулирование сводится к корректировке детализированных планов-графиков по фактическим результатам производственного процесса. Вне зависимости от типа производства (массовое, серийное или единичное) обычно управление осуществляется на двух уровнях:

- общезаводском (межцеховом);
- внутрицеховом.

Объектом управления на производственном участке являются: цеха основного производства, комплекточные и промежуточные склады и склады готовой продукции, а также экспедиционные и транспортные межцеховые службы. С точки зрения структурной организации элементами управляющей части являются соответствующие структурные подразделения заводского аппарата управления, а именно: аппарат линейного руководства предприятием, планово-экономический отдел предприятия, производственно-диспетчерский отдел предприятия.

Объектом управления на внутрицеховом уровне являются отдельные системы вида «человек—станок» со всем набором предметов и средств труда, необходимых для осуществления технологического процесса (заготовки, материалы, инструменты, оснастка, контрольно-измерительная аппаратура, тара и т. п.). Естественно, что системы вида «человек—станок» могут объединяться и в более крупные единицы, такие, как группа рабочих мест, объединенная единым технологическим процессом, производственная линия, участок и т. п.

С позиции организационной структуры элементами управляющей части на этом уровне являются: аппарат линейного руководства цеха или склада, планово-диспетчерское бюро или плановая группа цеха, линейное руководство внутрицеховых подразделений (начальник участка, мастер и т. п.), аппарат линейных диспетчеров цеха. Основные задачи, решаемые на межцеховом и внутрицеховом уровне, представлены в табл. 8.1 и 8.2.

Таблица 8.1

Межцеховой уровень

	Фазы управления			
	Планирование	Учет	Анализ	Регулирование
Решаемые задачи	Формирование квартальных отчетов	Учет выпуска изделий по кварталам	Контроль выполнения плана	Коррекция выполняемых планов
	Формирование сводного плана-графика выпуска изделий цехами	Учет выпуска изделий по календарным срокам	Анализ отставаний цехов от плана-графика	Коррекция плана-графика
	Определение календарных планов межцеховых передач	Учет межцеховых передач	Контроль и анализ причин срыва	Коррекция плана-графика

Внутрицеховой уровень

Решаемые задачи	Фазы управления			
	Планирование	Учет	Анализ	Регулирование
	Составление плана-графика запуска-выпуска деталей	Учет запуска и выпуска деталей производством	Контроль получения комплектующих деталей и инструментов	Расчет сменно-суточных заданий участкам и рабочим местам
	Расчет календарно-плановых нормативов (заделов и опережений)	Учет выполнения календарно-плановых нормативов	Контроль сводных показателей работы цехов	Регулирование календарно-плановых нормативов
	Прогнозирование брака	Учет брака	Анализ причин брака	Коррекция сменно-суточных заданий
Расчет загрузки оборудования	Учет использования оборудования	Анализ причин простоев	Коррекция сменно-суточных заданий	

Как следует из таблиц 8.1 и 8.2, центральной проблемой, решаемой задачами подсистемы ОУП, является составление планов на определенные периоды, которые должны отвечать следующим основным требованиям:

- обеспечивать ритмичность производства, необходимую для выполнения обязательств предприятия перед потребителями;
- обеспечивать сокращение перерывов в движении предметов труда по технологическому процессу;
- обеспечивать достаточно высокую загрузку оборудования и площадей, что позволяет эффективно использовать основные фонды предприятия.

С математической точки зрения (в словесном описании) основную проблему, решаемую в задачах подсистемы ОУП, можно сформулировать следующим образом: имеются некоторые группы оборудования (участки, рабочие места и т. п.), известен требуемый объем выпуска этих деталей, изделий на данный плановый период. Необходимо определить, какую деталь, изделие, в какое время, на какой группе оборудования следует запустить в производство, чтобы оптимизировать выбранный критерий качества

управления при выполнении ограничений, существенных для данного процесса.

Конкретный вид экономико-математических моделей для формирования оптимального плана-графика зависит от очень большого количества факторов конкретного производства.

К основным факторам относятся:

- принятый критерий оптимальности;
- производственная структура предприятия, которая определяет состав цехов, участков, рабочих мест, а также конструктивные и технологические особенности выпускаемых изделий.

Математическим аппаратом, используемым для решения задач, перечисленных в табл. 8.1. и 8.2, является аппарат моделей объемного планирования, календарного планирования и объемно-календарного планирования (см. экономико-математические модели).

8.2. Автоматизированные системы управления технологическим процессом

Основным назначением АСУТП является автоматическая оптимизация технологических процессов производства продукции на предприятии. В АСУТП обеспечивается реализация законов управления и как следствие — наиболее эффективное ведение технологического процесса и высокое качество продукции, т. е. выпускается максимально возможное количество готовой продукции при обеспечении необходимого ее качества и допустимой загрузки технологического оборудования. При этом в идеальном случае затраты должны быть минимальными. Это означает, что технологический комплекс и саму технологию необходимо проектировать таким образом, чтобы оборудование имело наименьшие габариты и массу. Конструкция оборудования должна гарантировать минимум потерь, а технология — максимальное использование исходного продукта и энергии. Наконец, ведение технологического процесса должно обеспечить выпуск готовой продукции с наименьшим полем допусков и минимум брака.

Единственный критерий, способный объединить перечисленные показатели, — экономический. Если стоимостные эквиваленты не установлены, повышение эффективности достигается путем оптимизации одного или двух наиболее существенных параметров. При двух параметрах приходится отыскивать компромиссное решение. Еще более сложны поиски компромисса меж-

ду минимумами расхода трех основных компонентов: материала, энергии и информации.

Управление технологическими процессами осуществляется в результате непосредственного взаимодействия информационных и материально-энергетических процессов. Это взаимодействие происходит в реальном масштабе времени и реализуется путем обмена информацией между технологическим объектом управления (ТОУ) и системой управления. Технологический характер объекта управления является основным признаком систем управления технологическими процессами. Задачами верхнего смежного уровня для систем управления технологическими процессами являются задачи оперативного планирования производства.

ТОУ характеризуются высокой размерностью входных и выходных переменных, сложной неявной связью между переменными на входе и выходе, большой неопределенностью по отношению к внешней среде, нестационарностью, нелинейностью, стохастичностью процессов, высоким уровнем шумов и большим разнообразием помех.

ТОУ представляют собой совокупность технологического оборудования и реализованного на нем по соответствующим инструкциям или регламентам технологического процесса производства. В качестве ТОУ можно рассматривать технологические агрегаты, группы станков, производственные участки, цеха, производственные процессы всего промышленного предприятия в целом, если управление этими процессами носит в основном технологический характер, то есть заключается в рациональном выборе и согласовании режимов работы агрегатов, участков производства. Как правило, ТОУ имеют четыре основных режима функционирования: пуск, остановка, нормальный и аварийный.

АСУТП предназначены для выработки и реализации управляющих воздействий, обеспечивающих желаемый вид процессов в ТОУ. Совместно функционирующие ТОУ и АСУТП образуют автоматизированные технологические комплексы (АТК). АСУТП являются частным видом систем управления с обратными связями и характеризуются наличием самостоятельных функций и целей управления, а также необходимой для реализации этих функций и целей специальной системной организацией. Это человеко-машинные системы управления, обеспечивающие автоматизированный сбор и обработку информации о состоянии и функционировании ТОУ и осуществляющие на основе этой информации оптимизацию протекающих в ТОУ процессов в соответствии с принятым критерием. В качестве характерных особенностей АСУТП надлежит отметить:

- наличие в составе их современных автоматических средств сбора и переработки информации, в первую очередь средств вычислительной техники;
- человек выступает в роли субъекта труда, основные функции которого заключаются в принятии управляющих решений на основе предоставляемой ему информации об объекте управления и данных о возможных альтернативных вариантах решений;
- осуществление переработки технической и технико-экономической информации;
- целью функционирования является повышение эффективности функционирования ТОО в соответствии с принятым критерием управления;
- выработка управляющих воздействий производится в реальном масштабе времени;
- как компонент системы управления промышленным предприятием АСУТП предназначена для целенаправленного проведения технологических процессов и обеспечения спетом управления более высоких уровней оперативной и достоверной информацией технико-экономического характера;
- возможность иметь многоуровневую иерархическую структуру и обеспечивать управление совокупностями АТК, ТОО, вспомогательными процессами (транспортирования, складирования и т. п.), входящими в состав производства;
- получение от соответствующих подсистем АСУП или служб управления предприятием заданий и ограничений (номенклатуры запланированных к выпуску изделий, объемов производства, технико-экономических показателей и т. п.) и обеспечение подготовки и передачи этим подсистемам или службам управления необходимых для их работы данных о состояниях и функционировании АТК. При наличии на предприятии автоматизированной системы технологической подготовки производства (АСТПП) должно быть налажено взаимодействие АСУТП с этой системой. АСУТП должна получать от АСТПП технологическую и другую необходимую ей информацию и передавать АСТПП оперативные данные для корректирования решения АСТПП на последующих временных интервалах.

Цели функционирования АСУТП в каждом конкретном случае могут быть разными. В качестве примера можно указать экономиию топлива, сырья, энергии и других видов производствен-

ных ресурсов; обеспечение безопасности функционирования; повышение качества конечного продукта; снижение затрат живого труда; достижение оптимальных нагрузок оборудования; оптимизацию режимов работы ТОУ и т. п.

В серийном, мелкосерийном и единичном производстве АСУТП создаются на основе групповой технологии, предусматривающей концентрацию изготовления технологически однородной продукции в специализированных подразделениях производства. АСУТП ориентированы на замкнутый цикл изготовления изделий внутри каждого технологического процесса. Это расширяет возможности комплексной автоматизации производственных процессов и позволяет достигать существенно более коротких производственных циклов. Если по каким-то причинам комплексная автоматизация всех технологических процессов производства оказывается невозможной, АСУТП создается как элемент некоторого производственного подразделения (линии, участка, цеха), имеющего более низкий уровень автоматизации, но единую систему управления. Тем самым обеспечивается замкнутый цикл производства и подготавливается возможность объединения отдельных АСУТП линий, участков, цехов в единую автоматизированную систему, управляемую ЭВМ.

8.2.1. Классификация и характерные особенности АСУТП

Одним из главных признаков классификации АСУТП является их разделение по типовым задачам управления, которые характеризуют их назначения:

1. АСУТП обеспечивающие стабилизацию (поддержание) заданного рационального или оптимального технологического режима, т. е. таких технологических параметров, на которые действующие возмущения оказывают существенное влияние. Например, к ним можно отнести системы стабилизации скорости резания при торцовой обработке деталей больших диаметров на металлорежущих станках, системы стабилизации скорости шлифования при уменьшении диаметра шлифовального круга и др.

2. АСУТП, обрабатывающие с установленной точностью заданное или непрерывно задаваемое рациональное (оптимальное) изменение технологического процесса. Такие системы программного или следящего управления применяются в роботах-манипуляторах, станках с ЧПУ, лифтовых подъемниках и пр.

3. АСУТП, самостоятельно (автоматически) выбирающие наилучший по какому-либо признаку технологический режим и

обеспечивающие его поддержание либо автоматически выбирающие наилучшую траекторию перемещения рабочего органа и обеспечивающие ее отработку с установленной точностью. К таким системам оптимального управления относятся АСУТП агрегатов оптимального раскроя материала системы дотационной перестройки, рассчитывающие и реализующие оптимальную программу обжаты металла в отдельных клетях стана непрерывной прокатки и т. д.

Следующим признаком классификации АСУТП является характер производственного процесса — непрерывный и дискретный. К производству с непрерывным циклом относятся, например, следующие: добыча нефти и газа, их транспортировка по трубопроводным системам, крекинг-процессы, производство тепловой и электрической энергии и др. Также весьма широк класс промышленных изделий с технологическими процессами дискретного типа. Различие в объемах выпуска промышленной продукции (от нескольких миллионов до нескольких единиц изделий) приводит к общепринятому разделению технологических процессов дискретного производства на процессы изготовления массовой и крупносерийной продукции (МКС-производства) и процессы изготовления мелкосерийной и единичной продукции (группа МСЕ-производств). В каждой из этих групп традиционно сложились различные подходы к организации многопродуктового производства с широкой номенклатурой выпускаемых изделий, контролю и обеспечению планируемого уровня качества готовой продукции и автоматизации производства.

В МКС-производствах имеются благоприятные условия для создания и использования специализированных автоматов и автоматических линий. Большие затраты на проектирование и внедрение средств автоматизации в этой группе производств быстро окупаются. Выпуск изделий установленной номенклатуры осуществляется, как правило, на нескольких технологических линиях с фиксированным маршрутом и жесткой технологией пооперационной обработки сырья и полуфабрикатов. Перечень видов выпускаемой продукции для МКС-производств относительно постоянный. Освоение каждого нового изделия сопровождается запуском новой технологической линии. Контроль и обеспечение качества продукции в МКС-производстве связаны с применением статистических методов выборочного контроля и анализа состояний процессов и изделий. По уровню технологической и технической подготовки в МКС-производствах складываются благоприятные условия для проведения работ по комплексной механизации и автоматизации производственных процессов.

Иная картина имеет место в МСЕ-производствах, характерных для 80% машиностроительных и приборостроительных предприятий промышленности. Выпуск продукции в этой группе производств организуется при противоречивых требованиях к специализации оборудования и его максимальной загрузке. Организационно-технологические трудности выполнения этих требований при быстроменяющейся номенклатуре изделий были до недавнего времени главными причинами слабого распространения в МСЕ-производствах методов календарного планирования и оперативно-диспетчерского управления. Окупаемость затрат на автоматизацию МСЕ-производств была крайне низкой.

Характерной особенностью современных АСУТП является *высокая производительность переработки информации* при практически неограниченном объеме памяти. Эта особенность базируется на применяемых в системах быстродействующих средствах вычислительной техники: программируемых контроллерах, микропроцессорах, управляющих вычислительных машинах (УВМ). Высокая информационная производительность УВМ обеспечивает реализацию указанного назначения АСУТП.

Применение быстродействующей вычислительной техники обеспечивает вторую характерную особенность АСУТП — их *высокую гибкость*, что позволяет существенно повысить гибкость технологического объекта управления при переходе на новую технологию. Перестройка сводится к «очистке» памяти управляющей машины и вводу с программ носителя новых данных. Даже для весьма сложных технологических комплексов время перенастройки не выходит за пределы нескольких десятков секунд.

Дополняя, а в дальнейшем заменяя человека, АСУ в известной мере копирует его функциональную структуру.

На рис. 8.4 схематически показаны основные функциональные связи, устанавливаемые в системе «человек — ТОУ». Для выполнения заданных операций управления человек должен получить ряд сведений, которые принято называть внешней информацией. Эта неизменная информация хранится в памяти оператора и включает основные характеристики технологического процесса и порядок выполнения операций управления в нормальных и аварийных режимах. Очевидно, отбор этой информации должен быть весьма тщательным, так как перегрузка памяти большим количеством цифр и ситуационных событий неизбежно приводит к частичной потере некоторых данных и нарушению порядка операций управления. За изменением технологических характеристик, представляющих собой совокупность управляющих параметров и возмущающих воздействий, человек следит с

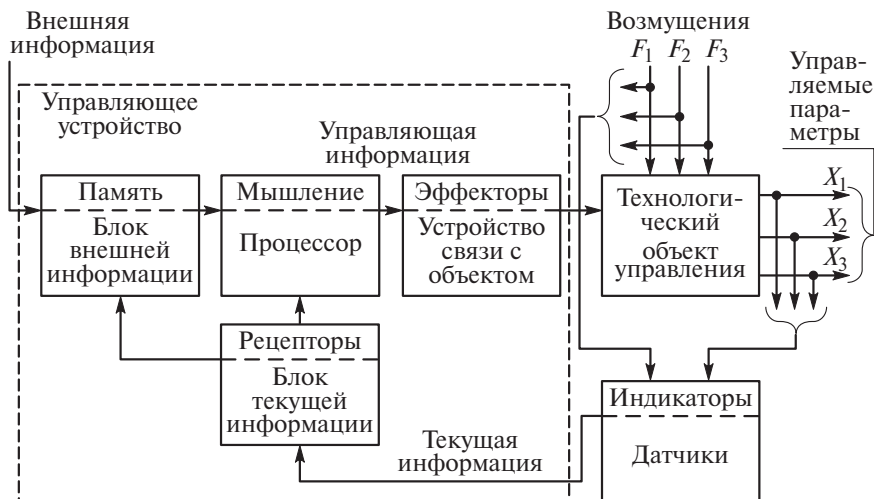


Рис. 8.4. Структура управления технологическим объектом человеком-оператором или АСУТП

помощью органов-рецепторов, из которых наибольшую нагрузку несет зрение. Человеку приходится наблюдать за показаниями комплекса приборов, некоторые предельные параметры могут фиксироваться с помощью звуковых сигналов. Восприятие текущей информации также может нарушаться, что вынуждает ограничивать количество индикаторов. Сопоставление исходной и текущей информации и принятие решения осуществляются центральной нервной системой. После выработки управляющей информации в результате принятия решения в периферийную нервную систему поступают сигналы. Их следствием является реализация команд управления с помощью эффекторов — рук, ног, голоса. По аналогичному принципу строится функциональная структура любой АСУ, в том числе и АСУТП. На рис. 8.4 указаны рецепторы, память, мышление, эффекторы, присущие человеку-оператору, и их аналоги в АСУТП. Сохраняя в качестве самостоятельных блоки внешней и текущей информации, процессор и устройство связи с объектом, АСУТП выполняет две функции: информационную и управляющую. Информационную функцию АСУТП обычно определяют как функцию, целью которой является сбор, преобразование, хранение информации о состоянии ТОО, передача ее для последующей обработки.

Управляющая функция АСУТП имеет своей целью выработку решений и осуществление управляющих воздействий на ТОО.

Как видно, в исполнении задач информационной и управляющей функций принимает участие процессор, совмещающий переработку, хранение и выработку управляющей информации, передаваемой в виде управляющего воздействия ТООУ по каналам устройства связи.

Рассмотрим функциональные блоки, показанные на рис. 8.4. В блок внешней информации управляющего устройства, содержащий внешние и оперативные запоминающие устройства, для длительного хранения вводятся априорная информация об объекте и технологические инструкции. В эти же запоминающие устройства от общего пульта управления вводится также исходная оперативная информация о текущем задании. Текущая информация о технологическом процессе с помощью датчиков технологических параметров вводится в блок текущей информации управляющего устройства.

Основными узлами управляющего устройства являются процессоры — логические и арифметические блоки, в которых априорная, исходная и текущая информация преобразуется в управляющие команды. Сложность этих устройств зависит от сложности задач, решаемых АСУ. На выходе системы установлены устройства связи с объектом, предназначенные для преобразования управляющих команд в управляющие воздействия достаточной мощности.

8.2.2. Структура АСУТП

Техническая реализация АСУТП ведет к централизованным и децентрализованным многоуровневым (иерархическим) структурам многопроцессорных управляющих вычислительных комплексов. В целом иерархические структуры представляют собой комбинацию двух структурных систем управления. Первое из этих представлений связано с пространственной (горизонтальной) декомпозицией, использующей особенности слабых связей; между отдельными подсистемами АТК (рис. 8.5). Второе — с многоступенчатой (вертикальной) декомпозицией задач различного характера, решаемых с помощью АСУТП (рис. 8.6). На первом, нижнем, уровне (рис. 8.5, *a*) обеспечивается задача стабилизации параметров после каждой операции, допускающей контроль перерабатываемого продукта. На втором уровне (рис. 8.5, *b*) решается задача согласования значений управляющих воздействий для каждой операции с целью обеспечения требуемого значения параметра конечного продукта. Эта задача может решаться как задача оптимального управления, например, исходя

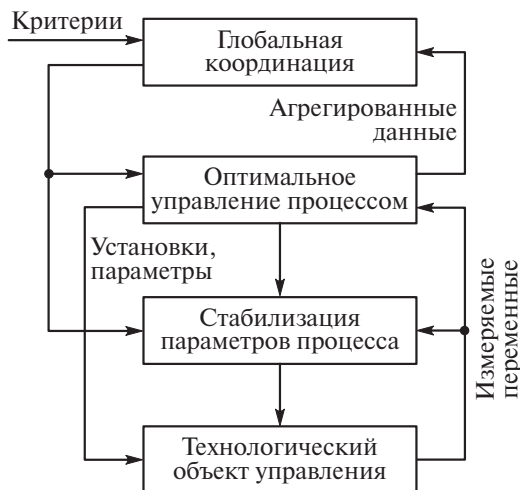


Рис. 8.6. Вертикальная декомпозиция задач АК

из условий оптимального расхода ресурса. На третьем уровне (рис. 8.5, с) системы управления определяются необходимые значения параметров конечного продукта и перераспределения материальных потоков с целью выпуска соответствующего его количества с различными требуемыми параметрами (глобальная координация).

Решение задач на каждом уровне осуществляется с помощью моделей, соответствующих различным уровням описания технологического процесса. Очевидно, что в зависимости от вида задач, решаемых при автоматизации технологического процесса, система управления может содержать разное число уровней. Кроме того, для разных технологических процессов характерно разное взаимодействие задач управления параметрами продукта и интенсивностью материальных потоков, что также находит свое отражение в структуре систем управления. Решение задач на каждом уровне осуществляется с разным ритмом, что соответствует различным темпам изменения параметров модели технологического процесса на различных уровнях описания. На более низких уровнях эти изменения характеризуются более высокой частотой, что требует внесения более частых корректирующих воздействий.

Уровень совершенства АСУТП зависит от совершенства элементов ее структуры и характеризуется степенью «самостоятельности» в выработке и реализации управляющих воздействий при ведении технологического процесса. Сложность элементов внутренней структуры АСУТП находится в тесной связи с характером

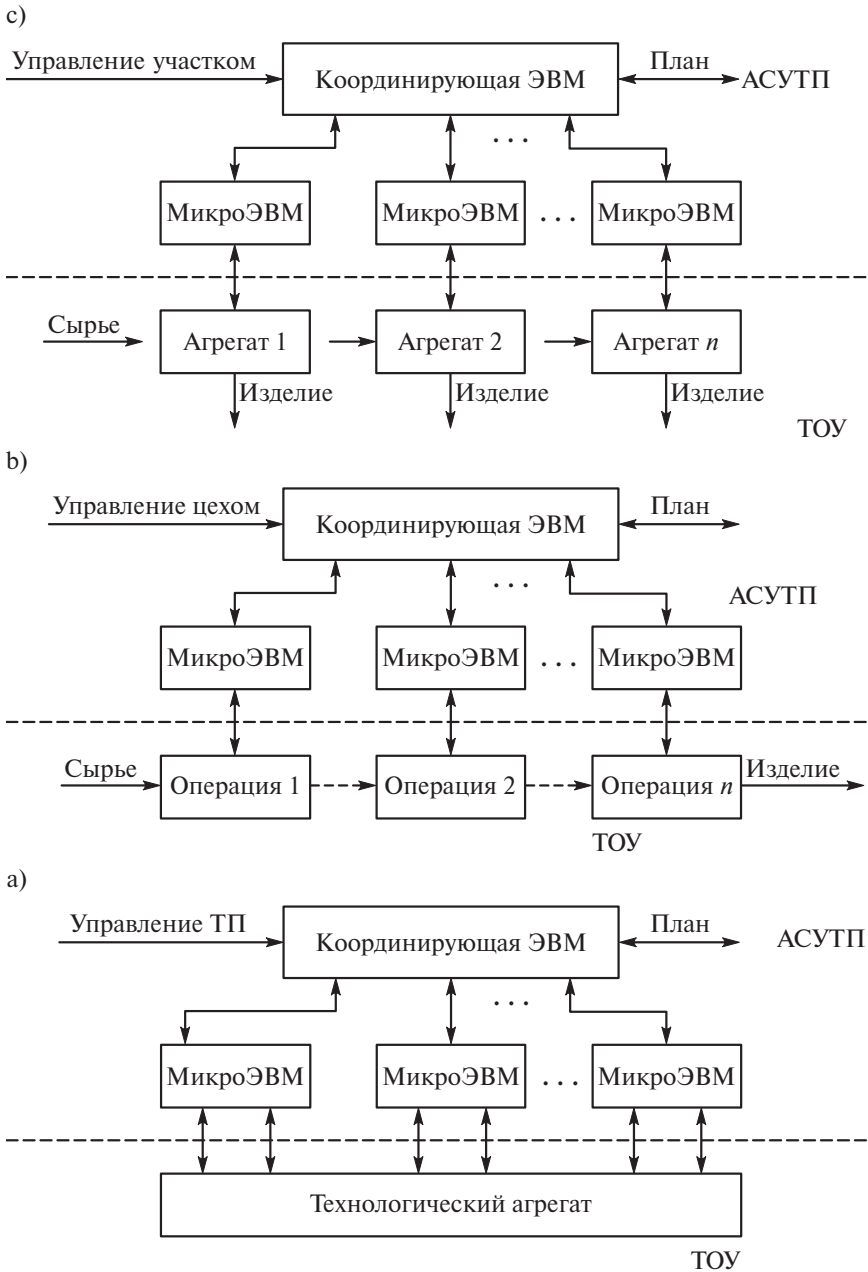


Рис. 8.5. Горизонтальная декомпозиция задач автоматизированного технологического комплекса: а – нижний уровень; б – средний уровень; с – верхний уровень иерархии

технологического процесса. Если этот процесс допустимо рассматривать детерминированным, то его модель будет проще.

В тех случаях, когда ход технологического процесса не удается свести к детерминированному, т. е. возможны некоторые отклонения случайного характера (изменение качества материала, его физических свойств, характеристик рабочего органа и т. п.), внутренняя структура АСУТП существенно усложняется. В таких системах априорная информация содержит формулировку цели управления и математическую модель процесса, которая позволяет определить наилучший путь достижения этой цели. Чем большее число случайных факторов подлежит учету, тем сложнее математическая модель процесса и элементы внутренней структуры АСУТП. Все это приводит к удорожанию системы, усложнению эксплуатации и существенному снижению надежности. Поэтому рациональный путь создания работоспособных АСУТП повышенной надежности состоит в совершенствовании не только отдельных элементов системы, но и технологического процесса, совершенствование которого должно быть направлено на повышение детерминированности последнего, в результате чего исключается или сводится до минимума влияние случайных факторов. Только одновременное совершенствование системы управления и технологического процесса может привести к созданию надежно функционирующих автоматизированных управляемых комплексов.

Во внешней структуре АСУТП могут быть выделены системы различных уровней или подсистемы АСУТП, выделяемые по функциональному или структурному признаку, отвечающему конкретным целям задач. Если функции системы ограничены одним механизмом, то она называется *локальной*. Примером тому может служить система безотходного пореза металла, управляющая подвижным упором ножниц обжимного прокатного стана. Эта локальная АСУТП обеспечивает при раскросе минимум отхода металла с учетом срочности выполнения заказов. Она располагает информацией о длинах раската, стандартных длинах слябов, объемах и сроках выполнения.

Если механизмы, управляемые локальными системами тесно связаны между собой технологическим процессом, то для их взаимной координации создается АСУТП. Объединение локальных АСУТП в узловое может быть осуществлено по территориальному или функциональному признаку. В этом случае устанавливается общая ситуационная связь расположенных на одном агрегате рабочих органов или механизмов. Комплексными АСУТП являются, например, системы управления печами, клетями и механизмами склада продукции прокатного стана, автоматизиро-

ванные станочные линии, роботизированные комплексы станков, комплекс агрегатов и механизмов обжимного прокатного стана, тонколистового стана горячей прокатки, роторный землеройный комплекс.

В этих системах комплекс технологических объектов подчинен общим технологическим закономерностям, в связи с чем здесь не могут применяться локальные АСУТП различной производительности. Построение такого комплекса исходит из единой для всех объектов заданной производительности.

Организованные в единую систему комплексные АСУТП, дополненные информационными узлами и автоматизированными системами переработки информации отделов и служб предприятия, образуют автоматизированную систему управления предприятием часто называемую интегрированной АСУ (ИАСУ).

Иерархия АСУТП определяет порядок подчинения взаимосвязанных подсистем общей системы, которая в пределах отрасли промышленности основывается на выработке определенной стратегии, реализуемой но оптимальным законам с учетом возможностей конкретных предприятий, цехов, технологических комплексов, механизмов. Сущность иерархического принципа состоит в том, что каждый из уровней вырабатывает задание — установку, обязательную для подчиненного ему уровня.

В последние годы при разработке АСУТП технологических линий и комплексов ранги иерархии сократились в связи с применением прямого цифрового управления, которое осуществляется централизованно с помощью общего вычислительного управляющего комплекса. Происходит замена структурной иерархии алгоритмической. Здесь иерархическую структуру приобретает алгоритм работы управляющей вычислительной машины.

С повышением ранга уровня усложняются решаемые задачи. Это требует более совершенной техники, однако характер задач (стабилизация, программное, оптимальное управление) и функции отдельных составляющих систем управления остаются неизменными.

Таким образом, АСУТП представляет собой комплекс технических средств технического, информационного, математического и программного обеспечения для управления технологическими объектами, который обеспечивает оптимальный при заданной структуре и технических средствах уровень автоматизации сбора и переработки информации для формирования управляющих сигналов и передачи их без потерь и искажения на исполнительные механизмы в целях достижения наиболее эффективной работы технологического объекта управления в целом.

8.2.3. Гибкие производственные системы (ГПС)

В настоящее время эффективность и рентабельность работы промышленных предприятия ставится в зависимость от решения неотложной задачи создания гибкой системы планирования и управления производственными ресурсами, обеспечивающей быстрое реагирование на изменяющиеся требования спроса к объему, номенклатуре и качеству дискретной продукции. Перспектива решения этой многоплановой задачи — создание предприятий с быстро перенастраиваемым производством.

В проблеме создания гибких МКС- и МСЕ-производств можно выделить организационный, технический и социальный аспекты.

Ведущим принципом организации производства становится групповая организация, реализуемая на основе предметной специализации, унификации технологических процессов и централизации выдачи производственным системам (участкам, линиям, цехам) сменно-суточных заданий по номенклатуре групп однородных изделий.

Выделение в номенклатуре продукции дискретного производства групп деталей или изделия, однородных по конструктивно-технологическим признакам, позволяет проводить в МСЕ-производствах технологическую унификацию и специализацию оборудования и переналаживаемой оснастки для обработки изделий партиями. Применение групповой обработки в МСЕ-производствах позволяет увеличить пропускную способность рабочих мест, что открывает возможность использования специализированного оборудования, быстропереналаживаемых многономенклатурных поточных линий с высокой степенью автоматизации. При групповом МСЕ-производстве появляется возможность внедрения прогрессивных методов контроля качества изделий и использования результатов контроля для принятия решений по компенсации источников неоднородности и нестабильности качества выпускаемой продукции.

Иными словами, решение вопросов организации и обеспечения гибкости производства на основе научных принципов группового производства приводит к стиранию различий между МКС- и МСЕ-производствами в подходах к планированию и управлению, к оценке эффективности комплексной автоматизации и сроков ее окупаемости.

Технологические возможности обеспечения гибкости производства связываются с достигнутым прогрессом в области микроэлектроники и вычислительной техники. Появление интеграль-

ных схем с высоким уровнем интеграции и создание микропроцессорных средств управления оказали огромное воздействующее влияние на сферу промышленного производства и его техническое перевооружение. В настоящее время средства ВТ применяются не только для автоматизации технологических процессов, но и для автоматизации процессов транспортирования, складирования, манипулирования, для автоматизации процессов сборки и контроля качества. Появилась реальная возможность осуществлять переналадку производства путем перепрограммирования контроллеров управления оборудованием, станков с числовым программным обеспечением (ЧПУ), адресуемых транспортных систем, обрабатывающих центров, обслуживающих роботов и манипуляторов. В расширении этой возможности для обеспечения гибкости производства важную роль призваны сыграть САПР изделий, технологии и оборудования, проектирования инструмента и групповой переналаживаемой оснастки, а также АСТПП. Развитие и практическая реализация идей адаптации в программных и технических средствах позволяют достигать «безлюдного» режима работы на технологических участках, линиях и в цехах.

ГПС — это система, допускающая иерархическую организацию с комплексно-автоматизированным производственным процессом, работа всех компонентов которой (технологического оборудования, транспортных и складских средств, погрузочно-разгрузочных устройств, мест комплектации, средств измерения и контроля и т. п.) координируется как единое целое системой управления, обеспечивающей быстрое изменение программ функционирования элементов при смене объектов производства.

Как подсистема промышленного комплекса ГПС определяется исходя из различных позиций. Например, можно понимать в качестве ГПС реализацию АСУТП в МСЕ-производствах. Построение ГПС в рамках интегрированной системы управления производством, включающей САПР и АСТПП означает переход не только к «безлюдной», но и к «безбумажной» промышленной технологии.

Независимо от физической природы процессов производства ГПС классифицируются по уровням организационной структуры: ГПМ — гибкий производственный модуль; ГАУ — гибкий автоматизированный участок; ГАЦ — гибкий автоматизированный цех; ГАЗ — гибкий автоматизированный завод. На самом низком уровне организационной структуры производства ГПМ определяется как ГПС, состоящая из единицы технологического оборудования, оснащенной автоматизированным устройством программного управления и средствами автоматизации технологического

процесса. Она функционирует автономно и осуществляет многократные циклы, применяется для встраивания в ГПС более высокого уровня организации. В ГПМ могут включаться некоторые вспомогательные средства. Например, на рис. 8.7 приводится пример ГПМ механической обработки, который состоит из станка с ЧПУ, системы контроля и управления, вспомогательного оборудования — манипулятора, накопителей, системы уборки стружки.

Совокупность взаимодействующих ГПМ, объединенных автоматизированной системой управления, функционирующая по технологическому маршруту, который предусматривает возможность изменения последовательности использования технологического оборудования, образует ГПС вида ГАУ. В ГАУ определяется и реализуется множество операций во времени и определенной логической последовательности. Структура ГАУ допускает включение его в качестве подсистемы ГПС в более высокого уровня организации, а также использование в условиях действующего неавтоматизированного производства. На базе ГАУ может быть реализована безлюдная технология. Безлюдный режим функционирования, как правило, не превышает по времени третьей части суток (обычно это ночная смена). При недостаточных уровнях надежности технологического оборудования и степени автоматизации отдельных операций управление ГАУ строится на основе предельного упрощения (сокращения) взаимодействия между отдельными ГПМ. Участие человека (оператора) в ГАУ необходимо для коррекции технологических режимов и диагностики состояния оборудования и систем управления. Операции загрузки и разгрузки, транспортирования, установки, позиционирования и снятия инструмента, загрузки программного обеспечения, пуска и выключения системы управления могут быть полностью автоматизированы.

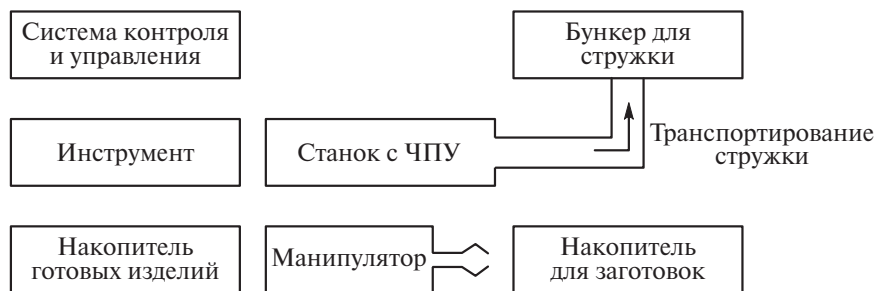


Рис. 8.7. Пример гибкого производственного модуля

Совокупность ГАУ, взаимодействие и функционирование которых обеспечивается распределенной системой управления, способных в течение определенного интервала времени выпускать технологически законченные изделия заданной номенклатуры и в установленных объемах, образует ГПС вида ГАП. Объединение ГАП на уровне их систем управления приводит к построению ГПС максимального уровня организации — гибкому автоматизированному заводу (ГАЗ).

При построении ГПС обычно рассматриваются две базовые альтернативные концепции. Согласно одной из них производство реализуется на основе узкоспециализированного технологического оборудования, обладающего высокими показателями надежности и живучести и имеющего встроенные системы контроля и управления. Использование сравнительно недорогого узкоспециализированного оборудования, оснащенного системами контроля и управления, позволяет изменять конфигурацию ГАУ при изменениях номенклатуры выпускаемых изделий, осуществляя переход от одного вида оборудования к другому.

Иной способ организации ГПС связан с применением многофункционального универсального дорогостоящего оборудования. В этом случае создаются ГАУ с многофункциональными перенастраиваемыми материальными и информационными потоками. Надежность функционирования таких ГАУ может быть обеспечена за счет многофункциональности даже при отказах (технологического оборудования в отдельных режимах его использования).

Основное качество ГПС — это гибкость. Применительно к задачам управления под гибкостью понимают способность ГПС обеспечивать возможность сохранения показателей эффективности функционирования производства при действии технологических и операционных возмущений.

К технологическим (внутренним) возмущениям относят простой ГПС (независимо от вызвавших эти простои причин), изменения технологических характеристик оборудования и инструмента, вариации параметров изделий на технологически, маршрутах и т. п. Операционные (внешние) возмущения могут быть обусловлены, например, изменениями объемов партий, численности партий каждого наименования.

Гибкость — это системная характеристика. При ее определении обычно используются более элементарные понятия технологической, операционной, производственной и интерактивной гибкости ГПС. Технологическая гибкость оценивается инвариантностью показателей эффективности функционирования ГПС к

действию технологических возмущений при реализации некоторого зафиксированного планового задания. Операционная гибкость оценивается по величине отклонений показателей эффективности ГПС при изменениях номенклатур обрабатываемых изделий при отсутствии технологических возмущений. Изменения номенклатуры обычно разрешены только в пределах допустимого множества наименований выпускаемой продукции. Под допустимым множеством наименований понимается множество, для которого системой технологической подготовки производства решается задача выбора технологии обработки, инструмента и оснастки, а на базе данных систем управления ГПС содержатся все необходимые для производственного процесса данные. При этом для каждого подмножества допустимых наименований сгенерированы архитектура вычислительных (управляющих) ресурсов и необходимое программное обеспечение. Производственная (нововведенческая) гибкость характеризует способность ГПС к расширению допустимого множества наименований. Это качество уже не является присущим только лишь ГПС. Производственная гибкость может быть обеспечена при организации функционирования ГПС в условиях интегрированной системы управления производством, т. е. в условиях ГАП — гибкого автоматизированного производства. В качестве оценки производственной гибкости можно принять величину временного интервала от момента поступления требования в САПР/АСТПП до момента введения всех необходимых для организации производства данных в базу данных ГПС. Интерактивная гибкость характеризует эффективность диалоговых систем в интегрированной системе управления производством. Оценкой интерактивной гибкости является время адаптации пользователя к решению определенного класса задач в ГПС в зависимости от его первоначальной подготовленности. Уменьшение интервалов адаптации, а также поддержание периодов адаптации на необходимом уровне у профессиональных пользователей приводит к необходимости разработки специальных тренажеров и методов организации некоторого аналога деловых игр.

Использование понятий гибкости позволяет ввести функционально-целевое описание ГПС. Перечисленные виды гибкости ГПС помогают получить ответы на вопросы о том, за счет каких ресурсов в производстве можно компенсировать технологические и операционные возмущения, обеспечение каждого вида гибкости осуществляется на разных уровнях автоматизации разными средствами. Технологическая гибкость достигается двумя способами: первый связан с управлением ГПМ (обеспечение инвариантности функционирования ГПМ к действию внешних и

параметрических возмущений технологического типа; придание системам управления ГПМ свойств, обеспечивающих их нормальное функционирование при возможных обрывах связей между центральной ЭВМ и локальными системами возмущения); второй — системный путь — связан с управлением движением материальных потоков, где основной задачей является поддержание эффективного функционирования ГПС при блокировке ГПМ (отказы, дефицит материалов на входе, переполнение накопителя на выходе ГАУ и т. п.).

ГАУ не имеет системных ресурсов для обеспечения технологической гибкости. Такие ресурсы закладываются при проектировании топологии материальных потоков, начиная с уровня гибкого производственного комплекса (ГПК), в который входят ГПМ, объединенные автоматизированной системой управления и автоматизированной транспортно-складской системой, которая автономно функционирует в течение заданного интервала времени как самостоятельная производственная единица либо как подсистема ГПС более высокого уровня организации.

Ресурсы, необходимые для достижения операционной гибкости аналогичным образом, делятся на технические и системные. Технические ресурсы обычно связаны с созданием и использованием в ГПС многофункциональных ГПМ. Системные ресурсы обусловлены наличием в ГПС необходимого множества доступных для выбора технологических маршрутов, использование которых во всяком конкретном случае связано с учетом функциональных возможностей ГПМ.

Системные аспекты обеспечения технологической и операционной гибкости практически мало различаются. Повышение операционной гибкости на системном уровне вызывает повышение технологической гибкости. Однако если последняя характеризуется числом одноименных маршрутов, то операционная — числом разноименных маршрутов. Соответственно и формализация этих задач различна. Различаются и применяемые оценки гибкости, которые связаны с временем перехода от одного подмножества допустимых наименований к другому или с временем перехода от одного подмножества технологических маршрутов (при зафиксированной номенклатуре выпускаемых изделий) к другому.

Свойство интерактивной гибкости ГПС обусловлено наличием в ГПМ средств автоматизированного управления и программного обеспечения интерактивных режимов функционирования и диалога. Достижение производственной гибкости более всего зависит от уровня развития САПР и АСТПП, а также от применяемых технических средств и программного обеспечения.

Рассматривая ГПС с позиций организации управления, будем выделять в качестве подсистем системы управления производством АСУТП и АСУ материальными потоками. В каждой из этих подсистем могут быть реализованы законы управления с обратной связью, а их функционирование направлено на достижение желательных динамических и экономических показателей производства на «оперативных» интервалах времени (часто используют понятие «реальное время»).

Развитие ГАП, включающих интегрированные САПР/АСТПП, играет важнейшую роль в процессе адаптации производства как целостной экономической единицы к требованиям внешней среды по выпуску новых видов изделий и внедрению новых технологий. В данном случае интеграция ГПС с АСНИ, САПР, АСТПП и АСУП приводит к действию механизмов прямого директивного управления производством на всех его уровнях и во всех звеньях.

Естественно, что реальный путь организации управления производством связан как с развитием и совершенствованием принципов, систем и средств управления с обратной связью в реальном времени на базе АСУТП и АСУ материальными потоками, так и с развитием прямого программного управления в условиях единой интегрированной системы управления производством.

Для МКС-производств АСУ материальными потоками мало чем отличаются от АСУТП. В МСЕ-производствах АСУ материальными потоками решают специфические задачи планирования и управления. При планировании определяются типы ГПМ, их местоположение в цехах и на участках, номинальные режимы работы. В ходе планирования определяются технологические маршруты при условии изменения номенклатуры изделий, планируются маршруты на уровне ГАУ при условии замены технологического оборудования, обслуживающих устройств, оснастки и др. Управление материальными потоками состоит в изменении темпов потоков и расчете множеств допустимых отклонений технологических режимов ГПМ, через которые проходят маршруты движения потоков. Заметим, что теоретические основы решения задач управления материальным потоками в МСЕ-производствах пока еще находятся в стадии становления.

8.2.4. Особенности проектирования АСУТП

Основные принципы построения АСУТП. Технологический объект управления и взаимодействующая с ним АСУТП составляют автоматизированный технологический комплекс (АТК). АСУТП включает техническое, программное, информационное обеспе-

чение как совокупность средств сбора, переработки технологической информации и преобразования ее и управляющее воздействие.

Техническое обеспечение — комплекс технических средств (КТС) включает устройства получения информации о технологических параметрах и состоянии технологического оборудования; формирования и передачи информации и системе, локального регулирования и управления; вычислительной техники; представления информации обслуживающему персоналу: передачи информации в смежные и вышестоящие АСУ, а также исполнительные устройства, в основном АСУТП, гидравлические и пневматические системы.

Программное обеспечение состоит из общего программного обеспечения, поставляемого вместе со средствами вычислительной техники (организующие программы, транслирующие программы, библиотека стандартных подпрограмм и др.); специального программного обеспечения, представляющего собой совокупность программ, которые реализуют функции данной АСУТП и обеспечивают заданное функционирование комплекса технических средств.

Информационное обеспечение АСУТП включает в себя единую систему кодирования технологической и технико-экономической информации, справочную и оперативную информацию. Спецификой проектирования АСУТП является применение системного подхода, эффективность которого возрастает с увеличением сложности и размеров системы. Системный подход состоит в том, что разрабатываемая АСУТП рассматривается как сложное целое со свойствами, не всегда присущими ее составляющим, структура и состав которых описаны выше. При системной разработке идут от общего к частному. Общей задаче, сформулированной для системы в целом, подчиняются задачи, решаемые отдельными ее составляющими.

Основные положения системного подхода заключаются в следующем.

1. Четко и конкретно формулируются задачи, поставленные при разработке АСУТП, ее назначение в общей структуре управления промышленным предприятием; устанавливаются критерии эффективности применения АСУТП. Это позволяет определить иерархические границы системы в соответствии с техническим объектом управления (механизм, комплекс механизмов, линия, участок, цех и т. д.). Выявляются задающие (управляющие) воздействия, поступающие на вход АСУТП от системы высшего ранга или от смежных систем. Определяются возможные

возмущающие воздействия, которые прикладываются к технологическому объекту управления.

2. В структуре АСУТП и технологического объекта управления выделяются подсистемы и отдельные элементы и устанавливаются связи между ними. Подсистемы могут выделяться по однородности физической природы технологии. Элементы системы обычно отождествляются с отдельными механизмами или транспортными устройствами и выделяются по их специализации на конкретной операции. Связи между элементами формируются в результате расчленения (декомпозиции) общей системной задачи.

3. Отражением системного подхода является иерархический принцип организации управления, описанный в гл. 6. Задания узловым системам формируются централизованными УВМ исходя из оптимизации технологического процесса всего комплекса. В свою очередь, УВМ отдельных узлов, оптимизируя выполнение полученной уставки, распределяют управляющие воздействия локальным АСУТП и комплексным АСУ. Иерархичность (подчиненность) проявляется здесь в том, что полученная данным уровнем уставка обязательна для выполнения и не подлежит коррекции.

4. Индивидуальная разработка и изготовление большинства элементов допустимы лишь при создании уникальных и специальных систем, когда оптимальный режим работы АСУТП обеспечивается оптимальными режимами составляющих ее элементов.

Промышленные АСУТП должны проектироваться на базе типовых элементов.

5. Повышение эффективности АСУТП достигается в процессе комплексного проектирования, при котором обеспечивается увязка функционирования всех подсистем в интересах поставленной цели — автоматизации технологического процесса.

8.2.5. Порядок разработки АСУТП

Разработка АСУТП от возникновения идеи до ввода ее в эксплуатацию — процесс трудоемкий и длительный. ГОСТы по АСУТП определяют пять стадий создания АСУТП: разработка технического задания, выполнение технического проекта, разработка рабочего проекта, внедрение и анализ функционирования.

Состав и содержание технического задания. При разработке технического задания выделяется четыре этапа.

1. Обследование технологического объекта управления, изучение и определение необходимого объема научно-исследовательских работ (НИР) и плана-графика проведения этих исследований.

2. Этап выполнения предпроектных НИР. Они включают анализ технологического процесса и оборудования как объекта управления и анализ информационных потоков: на этой основе формулируются критерии управления и ограничения. На стадии предпроектных НИР разрабатываются также предварительные математические модели технологического процесса, осуществляется идентификация моделей по экспериментальным данным и предварительно выбираются методы синтеза алгоритмов функционирования и управления.

3. Эскизная разработка АСУТП. Она содержит предварительные разработки функционально-алгоритмической структуры системы, синтез основных алгоритмов контроля и управления и их экспериментальную проверку, на этом этапе выбираются технические средства системы и дается обоснование выбора, определяются задачи модернизации технологического оборудования и совершенствования технологического процесса, осуществляется сравнительный анализ разрабатываемой системы и ее известных аналогов.

4. Собственно разработка технического задания на создание АСУТП. Она включает в себя: формирование полного перечня функциональных задач, решаемых системой; разработку технико-экономического обоснования ее создания; составление плана-графика работ по монтажу, наладке и пуску АСУТП; виды и объемы НИР на стадиях проектирования, внедрения системы и в процессе ее функционирования.

Стадия технического проектирования. Она состоит из ряда этапов.

1. Системотехнический синтез АСУТП. Он включает разработку функциональной структуры системы, постановку всех решаемых задач, разработку перечня измеряемых сигналов и метрологические характеристики измерений; в нем формулируются технические требования к системам контроля и регулирования; разрабатывается информационная и организационная структура системы.

2. Аппаратурно-технический синтез. На этом этапе разрабатываются системы локальной автоматики, состав и компоновка информационно-управляющего комплекса.

3. Разработка заданий на модернизацию технологического объекта управления и строительно-монтажные работы.

Ряд этапов предусматривает подготовку заявок на стандартное и нестандартное оборудование, составление сметы и ведомостей средств технического обеспечения. Отдельные этапы технического проектирования включают расчет ожидаемой техни-

ко-экономической эффективности АСУТП и ее сравнительный анализ с известными аналогами. Составляется патентный формуляр АСУТП. Заключительным этапом стадии технического проекта является техническое проектирование специального информационного и математического обеспечения АСУТП. На этом этапе разрабатываются полные наборы алгоритмов контроля и управления, общий алгоритм функционирования системы в реальном времени и осуществляется их проверка на объекте или модели. Осуществляется разработка схем сбора и обработки информации.

Если выбор некоторых технических решений при нескольких вариантах возможен лишь после проверки их на физической или математической модели, составляется эскизно-технический проект, включающий стадию моделирования.

Рабочее проектирование. Оно осуществляется на основании материалов технического проекта. Рабочее проектирование также состоит из нескольких этапов. Основным является этап разработки рабочей документации технического обеспечения АСУТП. Отдельными этапами рабочего проекта являются: составление заказных спецификаций; подготовка инструкции по эксплуатации, технического описания и регламента работы АСУТП с учетом действия оперативного персонала в предаварийных и аварийных режимах.

Следующим этапом, который может идти параллельно с предыдущими, будет разработка рабочей документации на программное и информационное обеспечение АСУТП. Сюда входят: программирование алгоритмов специального математического обеспечения, описание и инструкции по пользованию этими программами, подготовка таблиц, использования памяти ЭВМ, перечень всех программ, входящих в математическое обеспечение, уточнение схем сбора, обработки и выдачи информации и разработка полной документации по организации нормативно-справочной информации.

Этот комплекс разработок в процессе рабочего проектирования выполняется инженерами-программистами и инженерами-вычислителями.

Стадия внедрения АСУТП предусматривает пять этапов.

1. Проведение комплекса организационно-технических мероприятий, связанных с внедрением системы и обучением ремонтно-эксплуатационного персонала. На этих работах инженер-системотехник выступает в роли соисполнителя с инженерами-технологами, специалистами в области электронной техники, автоматики, вычислительной техники. К этому этапу относится

также модернизация технологического оборудования. На нем инженер-системотехник выступает в роли консультанта либо соисполнителя, если приходится существенно изменять АСУТП.

2. **Наладка АСУТП.** Она складывается из собственно наладки технических средств, отладки программ решения отдельных функциональных задач (информационной и управляющей), отладки программы, реализующей полный алгоритм функционирования системы.

3. **Опытная эксплуатация АСУТП.** Она начинается с включения системы. Определяются ее эксплуатационные характеристики, проводится дополнительная отладка программ и технических средств управления и вновь корректируется эксплуатационная документация. Во всех этих работах принимает участие инженер-системотехник.

4. **Приемо-сдаточные испытания.** Разрабатывается программа, в соответствии с которой проводятся промышленные испытания, оформляются их результаты и проводится анализ этих результатов.

5. **Доработка системы по результатам опытной эксплуатации и промышленных испытаний.**

Анализ функционирования АСУТП. Это заключительная стадия создания системы. Она включает исследования фактической технико-экономической эффективности системы, разработку рекомендаций по ее развитию и совершенствованию и выработку требований к модернизации технологического процесса.

8.2.6. Особенности проектирования АСУТП различного исполнения

Исполнение АСУТП тесно связано с выполняемыми ею функциями, поэтому рассматриваются эти функции для того, чтобы вместе с материалом, изложенным в § 8.2.5, определить особенности проектирования систем различного исполнения.

Из совокупности выполняемых АСУТП функций наиболее важными являются: централизованный контроль, обработка информации, вычислительные и логические операции информационного характера, управляющие воздействия. Все они сводятся к передаче информации для визуальной оценки, переработки ее до уровня информации-советчика, использованию информации для целей управления.

1. *Централизованный контроль.* К нему относятся:

1.1. Непрерывное, периодическое либо по вызову измерение и регистрация значений технологических параметров и показателей состояния технологического оборудования.

1.2. Обнаружение, регистрация и сигнализация отклонений от заданных пределов значений технологических параметров и показателей технологического оборудования.

1.3. Оперативное отображение и регистрация результатов обработки информации, выполняемых комплексом технических средств системы.

1.4. Контроль, регистрация и сигнализация срабатывания блокировок и защит.

2. *Обработка информации.* К ней относятся:

2.1. Косвенное измерение технологических параметров и показателей состояния технологического оборудования.

2.2. Вычисление и анализ обобщенных показателей оценки текущего состояния объекта управления и его подсистем.

2.3. Анализ срабатывания блокировок и защит.

2.4. Диагностика протекания и прогнозирование хода технологического процесса и состояния оборудования.

2.5. Расчет технико-экономических и эксплуатационных показателей работы ТОО.

2.6. Подготовка информации и обмен ею со смежными и вышестоящими АСУТП.

3. *Управляющие функции АСУТП.* К ним относятся:

3.1. Выбор оптимального режима технологического процесса в соответствии с целями системы.

3.2. Формирование сигналов управления для поддержания оптимального режима работы объекта управления.

3.3. Выдача оператору рекомендаций по управлению технологическим процессом.

Необходимым условием принадлежности системы к АСУТП является обязательное выполнение ею информационных и управляющих функций в минимальном объеме, соответствующем пп. 1.1, 1.2, 1.3, 2.6, 3.1 и 3.2. Это обязательное сочетание может наращиваться в зависимости от целей построения АСУТП различных вариантов. Оно предусматривает работу системы в режиме прямого управления, при котором обеспечивается непосредственное воздействие на ход технологического процесса путем передачи сигналов управления непосредственно исполнительным устройствам.

Все АСУТП по схеме создания делятся на следующие типы:

1) вновь разрабатываемые для ряда однотипных или близких по технологии процессов и оборудованию технологических объектов управления (повторяющиеся);

2) вновь разрабатываемые без повторения;

3) типовые системы;

4) повторно реализуемые.

Для повторяющихся АСУТП выполняются все стадии создания и все этапы, описанные в 8.2.5. При оценке однотипности технологических процессов необходимо исходить из соответствия математических моделей технологическим объектам управления.

Для разработки систем без повторения исключается стадия анализа функционирования в процессе эксплуатации, так как в следующем такие АСУТП не будут повторяться, а следовательно, нет необходимости нести дополнительные затраты.

При проектировании типовых АСУТП исключаются стадии внедрения и анализа. В зависимости от выполняемых функций по согласованию с заказчиком могут быть сокращены отдельные этапы, указанные в 8.2.5. Разрабатывая эти типовые АСУТП, нельзя идти по пути непосредственного повторения, особенно в плане формирования аппаратной части.

Для повторно реализуемых систем сокращается объем НИР и осуществляется одностадийное проектирование с выпуском ра-



Рис. 8.8. Структура АСУТП

бочего проекта, объединяющего вторую и третью стадии создания АСУТП. Техническое задание для таких систем составляется на основе предварительного обследования технологического процесса. Оно предусматривает в качестве основного собственно разработку технического задания. Технический проект включает только определение структуры и выбор аппаратуры АСУТП и техническое проектирование специального математического и информационного обеспечения. Особенность разработки АСУТП на уровне локальных и комплексных систем состоит в их расчленении на отдельные элементы со строго определенной целевой функцией каждого из них. В связи с ним при разработке АСУТП могут быть использованы типовые решения систем автоматизированного проектирования (САПР) каждого элемента системы. Исходя из структурной схемы рис. 8.4, можно рассматривать следующие элементы: технологический объект управления, устройство связи с объектом, процессор, устройство кодирования и преобразования информации, информационную часть системы.

Построение САПР каждого элемента включает: постановку задачи проектирования исходя из общих требований к АСУТП; выбор исходного варианта по ранее выполненным установкам; моделирование; формализацию описания частных элементов и расчет нелепой функции: сопоставление результатов с заданием и логическую операцию перехода к оформлению результатов, если выбранные параметры или схема удовлетворяют заданным требованиям либо осуществляется изменение параметров, если эти требования не удовлетворяются. На стадии изменения параметров или схемы лучшим вариантом является диалоговый режим, при котором опыт разработчика, его интуиция существенно сокращают время проектирования.

Известны разработки САПР средств измерения, дискретных устройств, мини- и микроУВМ и др. Все они доводятся до выдачи графической, текстовой и заказной документации, что существенно облегчает процесс разработки АСУТП. Получение частных решений является лишь одним из промежуточных этапов применения САПР. Системный подход предполагает изучение поведения системы до окончания разработки. Проверка работоспособности АСУТП выполняется на модели, которая может быть использована также для уточнения алгоритма управления путем внесения изменений по результатам сопоставления требуемого закона функционирования и полученного при моделировании. В этом случае в основу сопоставления должна быть положена целевая функция всей системы. Как и при решении частных задач, наиболее удобной формой логической части САПР является диалоговый режим.

На рис. 8.8 приведена одна из возможных структур построения САПР АСУТП. В машину вводятся технические условия и в формализованном виде воспроизводится исходный вариант проектируемого элемента системы либо части системы. По этим данным формируется модель либо целевая функция, которые сравниваются с заданными моделью или нелепой функцией. Логический блок осуществляет сравнение по заданной точности; если погрешность оказывается недопустимой, подается сигнал на уточнение параметров либо изменение алгоритма.

На экран дисплея могут быть вызваны заданные и достигнутые в процессе проектирования модели или целевые функции, поэтому представляется возможность вести проектирование в режиме диалога. Меняя параметры схемы либо алгоритм управления, проектант добивается желаемой точности совпадения результатов, полученных при проектировании, с заданными. При совпадении результатов ЭВМ переходит к выдаче графической, текстовой и заказной документации.

8.2.7. Автоматизированная система контроля и регулирования вращающихся печей

В этом параграфе рассматривается задача оптимального управления процессом обжига сырьевого материала во вращающихся печах и пути ее решения с помощью регулятора теплового режима РТР на примере автоматизированной системы, внедренной в ОАО «Комбинат Магнезит» (г. Сатка Челябинской области).

ОАО «Комбинат Магнезит» — крупнейший в стране производитель огнеупорной продукции — недавно отметил свой столетний юбилей. Несмотря на солидный возраст, предприятие динамично развивается и большое внимание уделяет внедрению современных наукоёмких разработок как в управленческой сфере (на предприятии внедряется корпоративная система *R/3*), так и в сфере управления технологическими процессами. Начиная с 1993 года, с участием НТЦ «Лидер» (г. Озерск) было внедрено более десяти автоматизированных систем в нескольких цехах и на различных технологических объектах: вращающихся печах, печах термической обработки, линиях дозирования компонентов шихты, испытательном стенде, а также реализованы системы контроля подачи порошков, контроля и учета работы прессов, контроля работы трубомельниц, контроля и управления деаэраторами, паровыми и водогрейными котлами.

Как на большинстве предприятий, вначале внедрялись информационные системы диспетчеризации и контроля хода тех-

нологических процессов. Однако целью автоматизации, безусловно, является повышение качества ведения технологического процесса, а в идеале и его оптимизация. Именно управляющие системы способны дать реальный экономический эффект, стабилизировать качество продукции. Наиболее трудоемкой в реализации таких систем является разработка управляющих алгоритмов, адекватных по сложности управляемым процессам. Немаловажным является также выбор инструмента, с помощью которого эти алгоритмы можно было бы реализовать на реальном объекте.

Данный параграф посвящен опыту разработки и внедрения управляющей системы, давшей существенный экономический эффект.

Постановка задачи. Задача контроля и регулирования процесса обжига материала во вращающихся печах решалась с момента установки их на комбинате, а именно с 30-х годов XX века.

Проблема в разное время решалась по-своему, но в целом методика решения оставалась прежней: человек наблюдал ход технологического процесса и выдавал задание на выработку управляющих воздействий либо «вручную», либо (в последние десятилетия) посредством каких-то контуров регулирования и управления. Необходимость полной автоматизации процесса ни у кого сомнений не вызывала, хотя в возможность практического достижения поставленной цели мало кто верил.

В начале 90-х годов предпринимались попытки автоматизации регулирования процесса обжига, в частности, на печах ЦМП-3 (ЦМП — цех магнезиальных порошков) с помощью системы САРА. Успех данной системы также был весьма ограниченным: какое-то время процесс шел в пределах установленных границ режима обжига, а потом печь «глохла», так как процесс выходил за допустимые границы и требовалось вмешательство обжигальщика.

Новый этап развития систем управления режимом обжига начался в мае 1993 года после установки системы МАИС. Система контроля и управления вращающейся печью прошла в своем развитии четыре этапа:

1. Установка IBM PC совместимого контроллера на вращающейся печи 7 ЦМП-3 для демонстрации возможности управления печью через компьютер (февраль 1993 года);

2. Переход к управлению комплексом печей 7—8 ЦМП-3 с применением методов распределенной сетевой обработки данных, исполнители НТЦ «Лидер» (г. Озерск) и ОАО «Комбинат Магнезит» (г. Сатка), проект ПКИ УРМА (г. Екатеринбург, ноябрь 1995 года);

3. Установка системы контроля и управления печами 4—6 ЦМП-3 и первой версии регулятора теплового режима (октябрь 1996 года);

4. Замена контроллеров системы контроля и управления печей 4—6 ЦМП-3 на контроллеры Micro PC (январь 2001 года).

После первых дней эксплуатации системы стало очевидным, что эффективность внедрения будет зависеть от того, удастся ли решить важнейшие задачи оптимизации хода технологического процесса обжига на уровне, недостижимом для человека-оператора, и минимизации влияния человеческого фактора на результаты производственного процесса в целом.

Краткое описание процесса обжига. Процесс обжига сырого магнезита во вращающейся печи является чрезвычайно сложным, с точки зрения возможности автоматизации. Если рассматривать процесс обжига, не вникая в детали, то он представляет собой процесс термического разложения сырого магнезита, усложненный наличием разного рода примесей в составе исходного сырья и значительным влиянием газодинамических характеристик печи. Процесс обжига сырого магнезита сопровождается нагревом магнезита, испарением влаги, содержащейся в сыром магнезите, плавлением легкоплавких примесей и т. д.

В ходе обжига сырого магнезита происходит уменьшение толщины слоя, снижение объема материала, причем степень изменения объема материала зависит от большого числа факторов, наиболее значимыми из которых являются химический и фракционный состав поступающего сырья, распределение температур по длине печи, или температурный профиль печи (положение и протяженность зон подогрева, каустизации, обжига), условия течения потока газов в пространстве печи. Различие степени изменения объема материала приводит к изменениям условий течения потока газов и, как следствие, газодинамических характеристик печи в целом.

Нестабильность газодинамических характеристик печи вызывает отклонения в режиме горения топлива, вследствие чего изменяется положение факела, влияющее на положение зоны обжига. Температура в этой зоне и ее положение в значительной степени определяют производительность печи и качество порошка на выходе.

Процесс обжига во вращающихся печах, работающих на смеси каустического и сырого магнезита, является более сложным: кроме обжига сырья, происходит спекание каустического магнезита, часть подаваемого каустического магнезита уносится потоком отходящих газов за пределы рабочей зоны печи; при этом со-

отношение количества каустического магнезита, подаваемого в печь, и количества порошка на выходе из печи составляет приблизительно 1,3—1,5, а при обжиге сырого магнезита аналогичное соотношение равно 2,2—2,6.

Управление процессом обжига. Процесс обжига во вращающейся печи, работающей на смеси, может управляться по пяти каналам:

- регулирование загрузки сырого магнезита;
- регулирование загрузки каустического магнезита;
- регулирование расхода топлива (природного газа);
- регулирование разрежения в пылевой камере;
- регулирование скорости вращения печи.

В литературе описываются и практически применяются всевозможные способы управления печью, основанные на отдельном или комбинированном с другими использовании перечисленных каналов регулирования.

Практики, профессионально занимающиеся управлением процессом обжига, в течение длительного периода времени применяли и применяют в настоящее время управление процессом обжига посредством использования всех пяти каналов регулирования.

Традиционное управление ходом технологического процесса. Традиционно управление процессом обжига в нормальном режиме происходит следующим образом: загрузка сырого магнезита устанавливается в пределах 7—10 т/ч, чтобы пыль по длине печи перемещалась более равномерно и не скапливалась в пересыпном колодце:

- загрузка каустического магнезита регулируется для управления температурой в пылевой камере;
- разрежение в пылевой камере регулируется в небольших пределах для стабилизации положения зоны обжига;
- подача газа изменяется с целью облегчения управления изменением температуры и положением зоны обжига: если температура начинает быстро изменяться, изменения температуры парируются регулированием подачи газа; изменение подачи газа является дополнительным каналом управления положением факела, а значит, и зоны обжига;
- скорость вращения печи регулируется в зависимости от хода технологического процесса, при падении температуры скорость вращения печи уменьшается, соответственно уменьшается загрузка сырого и каустического магнезита.

Данный способ управления время от времени приводит к перегрузке печи, в результате чего температура в зоне обжига падает, зона обжига смещается в сторону выхода из печи (горячей го-

ловки печи), качество материала падает, резко растет температура на выходе из холодильника, существует вероятность выхода брака и загорания транспортерных лент. Для возвращения процесса в нормальный режим печь ставится «на скорость»: загрузка снижается на 30—50%, скорость вращения печи уменьшается, а иногда, когда этих мер недостаточно, увеличиваются подача газа и разрежение в пылевой камере с целью смещения зоны обжига в сторону загрузочного конца печи при помощи «вытягивания» факела.

Постановка печи «на скорость» приводит к ряду негативных последствий.

Увеличивается тепловая нагрузка на футеровку: уменьшение загрузки и уменьшение скорости вращения печи, сопровождаемое при необходимости увеличением подачи газа, ведет к быстрому разогреву футеровки, то есть к тепловому удару.

Возрастает вероятность обрыва наvara и возникновения аварийных ситуаций: тепловые удары приводят к появлению местных температурных перенапряжений, что вызывает обрыв наvara; обрыв больших участков наvara является причиной остановки технологического процесса из-за резкого повышения температуры на выходе из холодильника и угрозы возгорания транспортерных лент; кроме того, большие куски наvara с температурой свыше 1000 градусов могут плотно закупорить пересыпной колодец и привести к длительному выходу печи из строя.

Уменьшается средняя загрузка печи: как правило, оптимально организованный непрерывный технологический процесс выгоднее периодического, каковым является обжиг с постановкой печи на подогрев.

Увеличивается удельный расход топлива: уменьшение средней загрузки и разогрев футеровки приводят к увеличению удельного расхода топлива на единицу продукции.

Качество готовой продукции становится непостоянным: изменяется температурный режим печи, перемещается зона обжига, изменяется толщина слоя материала, процесс обжига протекает неравномерно.

Изменяется газодинамический режим печи: нарушается оптимальное соотношение топливо/воздух, изменяется скорость перемещения газов в пространстве печи, изменяются величины разрежений по длине печи вследствие изменения толщины слоя материала.

Увеличивается пылевынос: уменьшение температуры в зоне обжига приводит к увеличению длины зоны каустизации, увеличение толщины слоя материала вызывает увеличение скорости движения газов в пространстве печи.

Существует тезис о самостабилизации хода технологического процесса обжига в силу действия комплекса обратных связей: перегрузка печи ведет к увеличению толщины слоя материала и, как следствие, к росту перепада разрежений по длине печи и удлинению зоны каустизации из-за смещения зоны обжига в сторону горячей головки печи; это влечет за собой значительный рост пылевыноса, при котором уносится существенное количество материала, толщина слоя уменьшается и процесс обжига нормализуется. Однако цена такой «самостабилизации» — перерасход сырья, топлива, рост пылевыноса, ухудшение качества продукции, а также риск пропуска необожженного материала, что является предпосылкой к аварии (загоранию транспортерных лент и полной остановке производства).

Автоматизированное управление ходом технологического процесса. Автоматизированное управление ходом технологического процесса основной своей целью имеет стабилизацию с высокой точностью температурного режима вращающейся печи с внесением минимальных возмущений в ход процесса, а также управление плавным изменением параметров режима при необходимости оптимальной настройки хода технологического процесса.

Таким образом, автоматизированное управление ходом технологического процесса реализуется как адаптивное управление загрузкой сырого магнетита, расходом природного газа и разрежением в зависимости от изменения температур в пылевой камере, в газоходе, в пересыпном колодце и от статистических данных о загрузке и температуре в газоходе.

Адаптивность управления реализуется через изменение коэффициентов передачи регулятора и через использование в качестве настроечных коэффициентов не постоянных величин, а определенным образом обработанных значений самих контролируемых величин; таким образом, регулятор теплового режима вращающейся печи представляет собой многоконтурную многосвязную адаптивную систему с элементами прогнозирования.

В результате выполнения проекта по разработке регулятора теплового режима (РТР) был сделан вывод о том, что в значительной мере неудачи предшественников объясняются стремлением автоматизировать труд обжигальщика, а не управление технологическим процессом обжига. В результате автоматика не могла выполнить работу обжигальщика лучше, чем сам обжигальщик. Это и было основной причиной, почему такие системы не приживались на производстве.

Алгоритм регулирования заключается в выработке параметров загрузки печи, расхода природного газа и разрежения, требу-

емых для оптимизации хода технологического процесса обжига, с учетом значений следующих наблюдаемых и вычисляемых параметров технологического процесса:

- средней величины загрузки за последний час;
- средней величины загрузки за последние пять минут;
- скорости изменения температуры в пылевой камере;
- средней скорости изменения температуры в пылевой камере за последние пять минут;
- ускорения изменения температуры в пылевой камере;
- скорости изменения температуры в газоходе;
- ускорения изменения температуры в газоходе;
- разности средней температуры в пылевой камере и заданной температуры;
- разности прогнозируемой и текущей температур в пылевой камере;
- разности прогнозируемой и заданной температур в пылевой камере.

Основные принципы построения регулятора теплового режима.

Главным в работе регулятора теплового режима (рис. 8.9) является принцип внесения минимальных возмущений в ход технологического процесса. Регулятор постоянно рассчитывает значение загрузки в текущий момент, необходимое для оптимального управления технологическим процессом.

Под оптимальным управлением понимается изменение загрузки сырого магнезита настолько, насколько это необходимо для удержания хода технологического процесса в заданных границах, без перегрузки или недогрузки печи. Необходимое воздействие на печь достигается путем управления загрузкой таким



Рис. 8.9. Регулятор теплового режима

образом, что среднее значение загрузки, в конечном итоге определяющее ход технологического процесса, остается величиной практически постоянной в промежутке времени, сравнимом с общим временем движения материала в печи.

Классификация параметров регулирования. Все параметры регулятора теплового режима (РТР) — и расчетные, и измеряемые — делятся на три основные группы:

- параметры, определяющие динамику печи, изменение и регулирование режима;
- параметры, определяющие статику печи, стабилизацию режима;
- параметры, определяющие характеристики системы «печь—регулятор». На динамику печи влияют следующие параметры:
 - текущая температура отходящих газов в газоходе;
 - текущая температура отходящих газов в пылевой камере;
 - текущая величина загрузки сырого магнетита;
 - скорость изменения температуры в пылевой камере за пять секунд;
 - ускорение изменения температуры в пылевой камере за пять секунд;
 - значение прогнозируемой температуры в пылевой камере;
 - заданная величина загрузки сырого магнетита;
 - средняя скорость изменения температуры в пылевой камере;
 - заданная скорость изменения температуры в пылевой камере;
 - скорость изменения температуры в газоходе за пять секунд;
 - значение прогнозируемой температуры в газоходе;
 - разность температур в газоходе и в пылевой камере.

На статику печи оказывают влияние следующие параметры:

- разность текущей и заданной температур отходящих газов в пылевой камере;
- температура термопары в пересыпном колодце;
- температура материала на выходе из холодильника;
- средняя величина загрузки сырого магнетита за пять минут;
- средняя температура в пылевой камере за три часа;
- средняя температура в пересыпном колодце за шесть часов;
- средняя величина загрузки сырого магнетита за час;
- заданная температура в пересыпном колодце;
- заданная температура в пылевой камере.

Характеристику системы «печь—регулятор» определяют, в числе прочих, следующие параметры:

- значение температуры в пылевой камере, полученное в предыдущем цикле измерения;
- значение коэффициента существенных изменений (КСИ);
- положение исполнительного механизма при загрузке сырого магнезита;
- ограничение минимума задания температуры в пылевой камере;
- средняя разность заданной и текущей величин загрузки сырого магнезита;
- положение ключа включения ограничения задания температуры в пересыпном колодце;
- коэффициент передачи регулятора теплового режима;
- среднее значение коэффициента адаптации (КА);
- коэффициент передачи дозатора (код/загрузка);
- положение ключа включения регулятора загрузки;
- величина тока контроля работы тиристорного блока управления загрузкой;
- код, определяющий величину тока, выдаваемого на тиристорный блок управления загрузкой;
- задание поддерживаемой регулятором величины загрузки;
- задание положения исполнительного механизма.

Логика работы регулятора. Вообще логика работы регулятора может быть достаточно проста, если рассматривать печь как трубу, в которую засыпается сырой магнезит, высыпается порошок, увеличение загрузки приводит к уменьшению температуры и наборот.

Практически же печь является сложной физико-химической, термо- и газодинамической системой; логика работы регулятора примерно соответствует сложности системы, поэтому «совсем просто» регулятор описать невозможно. Для понимания логики работы регулятора требуется знание того, как происходит управление вращающейся печью.

Основным требованием к регулятору, как уже было отмечено, является требование управлять процессом с внесением минимальных возмущений в его ход, поэтому при отстроенном регуляторе, удовлетворяющем этому требованию, печь «идет ровно», берет такую загрузку, какую вообще способна взять, производительность печи остается постоянно высокой, как и качество материала на выходе. По наблюдениям регулятору требуется примерно 2—3 суток, чтобы довести печь до состояния, когда она даст максимально возможную производительность при заданном качестве материала на выходе и практически неизменном расходе газа. Из этого состояния любой «толчок» (редкое отклонение па-

раметров) выводит печь примерно на 3—6 часов; таким образом, если печь «дергать» (пускать/останавливать, греть/загружать), оптимальных параметров хода технологического процесса она вообще никогда не достигнет.

Понятно, что требование полного отсутствия «толчков» по причинам разного характера является практически невыполнимым; поэтому регулятор теплового режима сделан многорежимным и самонастраивающимся.

Самонастройка регулятора основана на статистической обработке параметров технологического процесса и определении таких значений, при которых печь будет гарантированно работать, хотя и не в оптимальном режиме.

Дальнейшее уточнение значений параметров (точная настройка технологического процесса) происходит в процессе регулирования, при этом определяется градиент изменений и вычисляется оптимальное направление изменения параметров технологического процесса.

Внешние возмущения, влияющие на ход технологического процесса, имеют большую или меньшую значимость в зависимости от близости параметров технологического процесса к экстремальным значениям, то есть к таким, когда технологический процесс протекает наиболее эффективно.

При разработке регулятора принято, что изменение параметров извне (вручную) невозможно, чтобы исключить случаи непреднамеренного вывода технологического процесса обжига печи за пределы допустимых значений.

Структура системы. Система контроля и управления вращающейся печью (на примере печей 4—6 ЦМП-3) состоит из четырех функционально самостоятельных частей, соединенных с диспетчерской цеха и между собой через технологическую информационную сеть (ТИС) с протоколом IPX/SPX:

- резервированный контроллер, расположенный в щите КИП печей 4—6;
- вычислительная платформа с функционирующим на ней регулятором теплового режима;
- автоматизированные рабочие места обжигальщика;
- серверы МАИС, Novell NetWare, Windows NT, расположенные в здании цеха КИПиА.
- Резервированный контроллер выполняет функции, связанные со сбором и первичной обработкой технологической информации, а также управляющие функции:
- получение сигналов с датчиков температуры, расхода, давления, разрежения в виде стандартных токовых сигналов;

- получение сигналов с объекта управления в дискретной форме (=24 В и 220 В 50 Гц);
- выдачу дискретных управляющих сигналов (=24 В и 220 В 50 Гц);
- выдачу аналоговых управляющих сигналов (0—20 мА);
- контроль положения переключателей на щите КИП;
- выдачу сигналов о выходе значений параметров за пределы регламентных и аварийных установок;
- ограничение величины управляющих воздействий регламентными границами;
- контроль достоверности значений основных технологических параметров;
- контроль исправности УСО;
- обмен данными посредством ТИС.

Вычислительная платформа представляет собой системный блок персонального компьютера, конфигурация которого отвечает требованиям работы в тяжелых условиях. В частности, вместо жесткого диска на вычислительной платформе установлен модуль загрузочного флэш-диска FDM фирмы Fastwel. Настроечные конфигурации (база конфигурации МАИС) загружаются в стандартном варианте через ТИС с сервера Novell NetWare, а в случае отказа сети или иных обстоятельств автоматически перебираются альтернативные варианты: загрузка базы конфигурации с одного или другого АРМ обжигальщика, либо с флэш-диска. На вычислительной платформе функционирует регулятор теплового режима, управляющий ходом технологического процесса вращающейся печи посредством получения и обработки данных с контроллера через ТИС и расчета и выдачи величин управляющих воздействий через ТИС на контроллер.

На АРМ обжигальщика выполняются функции по отображению и архивированию технологической информации:

- отображение параметров технологического процесса посредством графических мнемосхем;
- просмотр графиков изменения параметров по данным из памяти контроллера (ретроспектива до 8 часов);
- просмотр графиков изменения параметров по данным из архива на жестком диске АРМ обжигальщика (ретроспектива до 999 суток).

Таким образом, управление ходом технологического процесса осуществляется независимо от работоспособности устройств верхнего уровня.

Для расширения возможностей и повышения надежности работы АСУТП используются серверы Novell NetWare, МАИС, Windows NT.

Серверы Novell NetWare используются в качестве файл-серверов для хранения данных, архивов и т. п., а также для автоматизации контроля, управления, эксплуатации и сопровождения программного обеспечения.

Серверы МАИС выполняют функции по обработке данных, получаемых через ТИС в соответствии с определенными алгоритмами (статистика, контроль нарушений технологии, учет расходов энергоресурсов и сырья, ведение протоколов наблюдений метрологических характеристик системы и т. п.).

Сервер Windows NT используется как сервер приложений под Windows.

Технические средства. Система контроля и управления вращающимися печами реализована на базе IBM PC совместимых аппаратных средств и стандартной сетевой аппаратуры Ethernet, Arcnet.

Учитывая жесткие условия эксплуатации (высокая запыленность), контроллеры смонтированы в шкафу со степенью защиты IP55 серии technopac II electronic фирмы Schroff.

В контроллерах используются следующие модули и платы:

- процессорные платы 5066 фирмы Octagon Systems;
- универсальные модули ввода-вывода UNIO96-1 фирмы Fastwel;
- платы дискретного ввода TBI-24/0 и дискретного вывода TBI-0/24 фирмы Fastwel;
- модули УСО дискретного ввода 70G-IAC5A и дискретного вывода 70G-OAC5A фирмы Grayhill;
- модули аналогового ввода с гальванической изоляцией ADC32G фирмы «Риус»;
- модули аналогового вывода AO16-C8 фирмы Fastwel;
- источники питания фирмы Artesyn Technologies;
- оборудование для создания резервированных систем: разветвители, коммутаторы, модули контроля работы процессоров (арбитры) и т. д. — разработки фирмы «Лидер».

Общее количество входных/выходных сигналов составляет порядка 300, а общее количество параметров системы, включая расчетные, лабораторные и специальные, — порядка 1000.

Поскольку система выполняет ответственные управляющие функции, к ее техническим средствам предъявляются очень жесткие требования по надежности. Требуемые показатели надежности достигаются с помощью резервирования основных компонентов системы (рис. 8.10).

Имеется два идентичных комплекта процессорных плат и модулей УСО (основной и резервный контроллеры), каждый из

которых смонтирован в отдельном крейте со своим блоком питания. Все входные сигналы через разветвитель подключаются к обоим комплектам УСО. Выходные сигналы подключаются к программно управляемому коммутатору. Основной и резервный контроллеры работают синхронно и параллельно: производится ввод и обработка сигналов с объекта, выполняются расчеты, формируются управляющие воздействия, но сигналы на исполнительные механизмы проходят только с основного контроллера. Работоспособность контроллеров проверяется специальным модулем-арбитром. В случае выхода из строя основного контроллера (пропадание питания, «зависание» программы или другое событие, повлекшее превышение времени отклика) производится перекоммутация выходных цепей на резервный контроллер, который получает статус основного. То есть деление на основной и резервный контроллеры в достаточной степени условно.

Таким образом в системе контроля и управления вращающимися печами реализовано «горячее» резервирование с «безударным» подхватом всех функций. Аналогичный подход реализован и на других технологических объектах ОАО «Комбинат Магнит», где внедрена система МАИС и присутствуют управляющие функции.

Преимущества использования резервирования достаточно очевидны. Кроме высокой живучести системы, обеспечивается также удобство эксплуатации и ремонта, ведь замену и добавление модулей, подключение новых точек, перекоммутацию сигналов можно производить без остановки системы. Такое решение позволяет при проектировании новых объектов автоматизации смело закладывать «бесщитовой» вариант системы управления, когда показывающие и регистрирующие приборы исключаются полностью, а количество органов ручного управления сводится к минимуму. Ясно, что капитальные затраты и последующие экс-

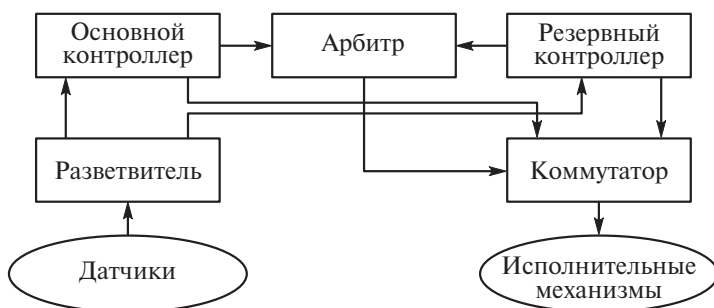


Рис. 8.10. Структура резервированного контроллера

платационные расходы в этом случае существенно меньше. Что касается стоимости, то, как показывает наш опыт, удорожание, связанное с выбором резервированной системы вместо нерезервированной, составляет не более 20—25% от общей стоимости проекта.

Программное обеспечение. В качестве инструментальных программных средств при реализации системы использовалась система МАИС, разработанная НТЦ «Лидер» (г. Озерск).

Это полностью отечественная разработка, учитывающая специфику приборного парка и исполнительных механизмов российских предприятий. Разработчики могут адаптировать программные средства к особенностям конкретного предприятия и конкретной технологии. В процессе внедрения и отладки системы контроля и управления вращающимися печами были внесены дополнения в базовое программное обеспечение специально для решения нестандартных задач, связанных, в частности, с реализацией регулятора теплового режима:

- встроенные программные адаптивные фильтры нижних частот до десятого порядка включительно для обработки аналоговых входных сигналов;
- набор компонентов для создания программных адаптивных фильтров нижних частот до сотого порядка включительно;
- возможность выполнения вычислений в формате с двойной точностью;
- возможность построения регуляторов любого уровня сложности и с любыми законами регулирования, в том числе с использованием распределенной сетевой модели вычислений;
- возможность внесения изменений в управляющие алгоритмы в режиме on-line через ТИС с немедленной проверкой результата без перезагрузки контроллера, а значит, и без влияния на ход технологического процесса;
- возможность дистанционного контроля и тестирования работы АСУТП в целом, а также любого из элементов, в том числе и каналов ввода-вывода УСО.

Управляющие алгоритмы, алгоритмы обработки данных (статистика), учетные задачи и т. д., а также алгоритмы функционирования собственно регулятора теплового режима реализованы средствами технологического языка ТЛ и оформлены в виде программ, которые могут выполняться как в контроллерах, так и на вычислительных платформах. Эти программы могут работать по вызову оператора, циклически с заданным периодом, по наступ-

лению события в ходе технологического процесса в реальном масштабе времени, в многозадачном режиме с использованием приоритетов.

Следует отметить особенности системы МАИС:

- используется широко распространенная IBM PC совместимая архитектура на нижнем и верхнем уровнях системы, а также стандартная сетевая аппаратура (Arcnet, Ethernet);
- работает в стандартных сетях Novell и Microsoft;
- система построена на архитектуре «клиент—сервер»;
- простота настройки на конкретный технологический объект, малые сроки внедрения;
- наличие развитого технологического языка программирования;
- программирование верхнего и нижнего уровней АСУТП ведется в рамках единого проекта;
- все функции, связанные с управлением технологическими объектами, реализуются на нижнем уровне системы (в контроллерах), и их выполнение не зависит от работоспособности устройств верхнего уровня;
- возможность создания систем с полным резервированием информационных и управляющих функций;
- корректировка информационного обеспечения и управляющих алгоритмов осуществляется без остановки контроллеров;
- возможность администрирования всех компонентов системы по сети;
- наличие распределенной сетевой модели вычислений;
- открытый формат представления данных в архиве;
- экспорт данных в различные распространенные форматы БД, что позволяет передавать данные в корпоративную систему уровня предприятия.

Заключение. В настоящее время регулятор работает устойчиво при любых изменениях параметров материалов на входе печи, обеспечивая заданное качество порошка на ее выходе с максимально возможной для текущего режима производительностью.

В ходе технологического процесса удается оптимизировать температуру зоны обжига в диапазоне 1840...1940 °С, причем это делается без применения кислородного дутья, а только за счет решения многофакторной задачи управления реально работающим промышленным тепловым агрегатом.

По результатам проведенных исследований в течение определенного промежутка времени средний расход топлива сократился на 7% при повышении производительности печи на 3,5% и одно-

временном снижении пылевыноса на 3,8%, что в конечном счете позволило снизить срок окупаемости программно-технического комплекса на базе МАИС для системы управления 3 печами до 6 месяцев.

Проводимые в течение 8 лет исследования по разработке и настройке РТР, защищенные патентом «Способ управления процессом обжига сырьевого материала во вращающейся печи» RU 2139482 С1 от 12.03.98, могут быть использованы в качестве базы для построения регуляторов теплового режима на вращающихся печах других типов, а также в качестве основы для построения регуляторов хода технологического процесса на других объектах.

8.2.8. Система числового программного управления технологическим оборудованием NC-2000

В качестве примера современных АСУТП рассмотрим систему числового программного управления NC-2000, предназначенную для управления сложным и высокопроизводительным оборудованием. Описываемая система является частью замкнутой технологической цепочки современного гибкого автоматизированного производства, построенного на основе интегрированных информационных связей, поддерживаемых аппаратно. Она участвует в информационной сети оперативно-технологического назначения цеха и завода, выполняя свои функции без участия человека.

Идея гибкого производства. По мере развития общества потребности человека неуклонно растут. Добиваясь определенного уровня благосостояния, человек стремится к новым вершинам — такова его природа. В то же время, находя пути удовлетворения своих потребностей, человек вынужден становиться производителем материальных благ. А став производителем, вынужден постоянно работать над совершенствованием технологии производства, повышая его эффективность и улучшая качество продукции. На сегодняшний день никто из нас не мыслит себе жизнь без домашней и офисной техники, посуды, упаковки и т. д., имеющих современный дизайн. А это — пластмассовые и металлические корпуса, состоящие из массы сложных поверхностей, которые подчас невозможно даже отобразить на чертеже.

Уровень современной технологии позволяет создавать к памяти компьютера трехмерные модели деталей сложной формообразующей оснастки штампов и пресс-форм и изготавливать их на станках с числовым программным управлением (ЧПУ) в условиях единичного производства. При этом процесс подготовки уп-

равляющих программ для станка с ЧПУ выполняется автоматически на основе трехмерной модели изделия. Сложная деталь изготавливается «с листа», без создания опытных образцов. Естественно, что при наличии на предприятии потока индивидуальных заказов остро встает проблема организации и планирования производства, решаемая также при помощи современных автоматизированных систем.

Идея гибкого автоматизированного производства не является новой, но при современном уровне развития компьютерных технологий и сетевых решений приобретает особое значение. В то же время задача создания такого производства достаточно сложна. Она выдвигает повышенные требования как к программному обеспечению инженерного корпуса, так и к системам управления оборудованием.

Современные требования к системам ЧПУ. Наш взгляд на систему ЧПУ металлообрабатывающего оборудования за последние 10—15 лет претерпел кардинальные изменения. Ранее единственной тенденцией развития станков с ЧПУ была тенденция превращения их в обрабатывающие центры. При этом станки оснащались магазинами, рассчитанными на большое количество инструментов. Для встраивания станка в автоматическую линию он комплектовался быстросменными столами-спутниками и их накопителями. Программирование осуществлялось вручную.

С появлением мощных (и особенно персональных) компьютеров стало возможным создавать управляющие программы (УП) для станков с ЧПУ автоматически, сократив до минимума количество ручного труда. Это позволило разрабатывать УП для обработки сложных трехмерных поверхностей, используемых чаще всего в формообразующих деталях оснастки штампов и прессформ. При этом требования, предъявляемые к станкам с ЧПУ, изменились. Произошло это в основном благодаря увеличению размера и структурным изменениям УП, а также существенному увеличению доли основного времени обработки при соответствующем уменьшении доли вспомогательного времени. УП обработки одной поверхности теперь достигает нескольких мегабайтов (а иногда и десятков мегабайтов) и состоит из множества «коротких кадров» — программных блоков, описывающих короткие перемещения инструмента (чем выше точность обработки, тем короче эти перемещения, так как их длина определяет точность аппроксимации). Обработка при этом ведется преимущественно одним инструментом много часов подряд, а станок теряет уже ставшее привычным оснащение: магазины, столы-спутники и т. д. Разумеется, речь здесь идет о станке, предназначенном для изго-

товления сложной формообразующей оснастки. Количество управляемых координат при том возросло до пяти. Но вместе с этим значительно выросли требования к системе ЧПУ. Учитывая имеющийся на данный момент практический опыт, можно сказать, что современная система ЧПУ, предназначенная для сложной трехмерной обработки, должна обеспечивать следующее.

1. Возможность ввода (приема с внешнего носителя или через сеть) управляющих программ неограниченного размера, их редактирования и исполнения как единого целого.

2. С целью уменьшения основного времени обработки:

- опережающая (по отношению к исполнению) расшифровка кадров УП;
- возможность отработки движения без снижения до нуля скорости в конце перемещения, описанного в каждом отдельном кадре (при соблюдении условия отсутствия превышения максимальных ускорений по осям); это позволяет обрабатывать сложную траекторию, описываемую в УП и состоящую из множества «коротких кадров», на скорости, близкой к заданной скорости подачи;
- определение предельных ускорений по управляемым осям с учетом динамических характеристик станка;
- возможность перехода с одной траектории обработки на другую на рабочей подаче без торможений и разгонов по трехмерной петле, рассчитанной системой САМ (computer-aided manufacturing); для этого требуется интеграция с САМ-системой или с САМ-сервером через сеть;
- возможность работы на повышенных скоростях рабочих подач (до 60 м/мин); с этой целью помимо достаточной для перекрытия требуемого диапазона регулирования разрядности ЦАП необходимо, чтобы время гарантированной реакции системы управления движением было относительно малым (около 200 мкс).

3. С целью снижения времени переналадки:

- доступ к файлам и ресурсам конструкторского и технологического бюро через стандартную сеть, включая поддержку стандартных (в том числе распределенных) баз данных;
- встроенная функция трехмерной коррекции траектории движения инструмента на величину его радиуса;
- возможность интеграции с САМ-сервером для выполнения полноценной коррекции УП по результатам предыдущих операций (в том числе трехмерной коррекции траектории движения инструмента на величину его радиуса);

- привязка набора УП, подпрограмм, корректоров, параметров системы и служебной информации к конкретному изделию (проекту);
 - возможность параллельно с процессом обработки выполнять редактирование или эмуляцию работы другой УП, ввод УП с дискеты (в том числе с использованием многотомных архивов), доступ к сети, включая обращение к САМ-серверу;
 - возможность автоматизированного измерения (или поиска) баз заготовки (детали), контроля размеров детали и инструмента.
4. С целью увеличения коэффициента загрузки оборудования в условиях единичного и мелкосерийного производства:
- возможность работы совместно с системой управления верхнего уровня на основе стандартных сетевых технологий;
 - возможность информационной поддержки систем планирования и диспетчеризации на уровне цеха или участка.
5. С целью повышения надежности системы:
- повышение ресурса узлов ЧПУ за счет применения отлаженных серийных модулей (плат);
 - применение вместо традиционных реле высоконадежных твердотельных силовых модулей (с гальванической изоляцией прочностью не менее 1500 В) и оптоэлектронных датчиков положения.
6. С целью увеличения ремонтпригодности и уменьшения времени поиска неисправностей:
- наглядное представление сигналов электроавтоматики в соответствии с электрической схемой станка и приведенными в техническом описании алгоритмами работы;
 - наличие подсистемы диагностики и выдачи сообщений оператору;
 - возможность непосредственного управления исполнительными устройствами;
 - конфигурирование ЧПУ из стандартных узлов, из которых также могут быть построены системы управления другим оборудованием завода и с использованием по возможности стандартного базового программного обеспечения.
7. С целью обеспечения гибкости системы:
- возможность постоянной доработки системы ЧПУ в соответствии с непрерывно растущими требованиями современного производства;
 - возможность быстрой адаптации к любому технологическому оборудованию (в том числе не металлорежущему);

- возможность интеграции со сложными автономными системами (например, с системами технического зрения);
- возможность выполнения необходимых измерений и объема детали-прототипа с целью создания трехмерной математической модели или построения УП для копирования;
- наличие гибкой архитектуры системы, реализуемой в зависимости от поставленной задачи;
- надежная поддержка фирмой-разработчиком и обновление версий базового программного обеспечения.

Особенности системы NC-2000. В результате многолетней работы над созданием систем управления технологическим оборудованием современного производства НПП «Модель» разработало систему ЧПУ NC-2000. Данная система управления в значительной степени отвечает приведённому набору требований. Система ЧПУ NC-2000 впервые была продемонстрирована в Москве на выставке «Передовые технологии автоматизации — 2001» в декабре 2001 года.

Систему NC-2000 создавали как систему нового поколения, воплощая в ней весь имеющийся опыт. Конструктивно система построена на основе отлаженных стандартных серийно выпускаемых базовых узлов и плат, производимых фирмами Advantech, Fastwel, Grayhill и Omron. Использование стандартных узлов позволило решить сразу несколько важных задач. Первая из них — существенное сокращение сроков разработки системы при одновременном снижении трудозатрат. Вторая — повышенная надежность комплектующих, так как их серийно выпускают хорошо зарекомендовавшие себя на мировом рынке фирмы. Третья — универсальность и доступность применяемых комплектующих: они поставляются на российский рынок в массовом порядке, на их основе могут быть построены самые разнообразные системы управления, включая большие распределенные системы АСУТП, при сохранении однородности ремонтной базы.

Основной особенностью системы NC-2000 является принцип ее построения. Это распределенная система с гибкой архитектурой. Варианты архитектуры системы NC-2000 показаны на рис. 8.11, а примерная структура производственного участка с ее использованием — на рис. 8.12. Как видно из рисунков, ядром системы является базовая станция, построенная в конструктиве промышленной рабочей станции фирмы Advantech. Базовая станция управляет всей системой на уровне команд, обеспечивает пользовательский интерфейс, сетевые возможности, а также отвечает за взаимодействие с автономными внешними устройс-

твами (такими, например, как система технического зрения) и системами верхнего уровня. Работает станция под управлением Windows-NT.

Блок (контроллер) адаптивного управления выполняет функцию регулировки параметров технологического процесса или подстройки параметров оборудования в реальном масштабе времени. В зависимости от сложности процесса адаптивного управления, уровня требований к надежности и точности управления,

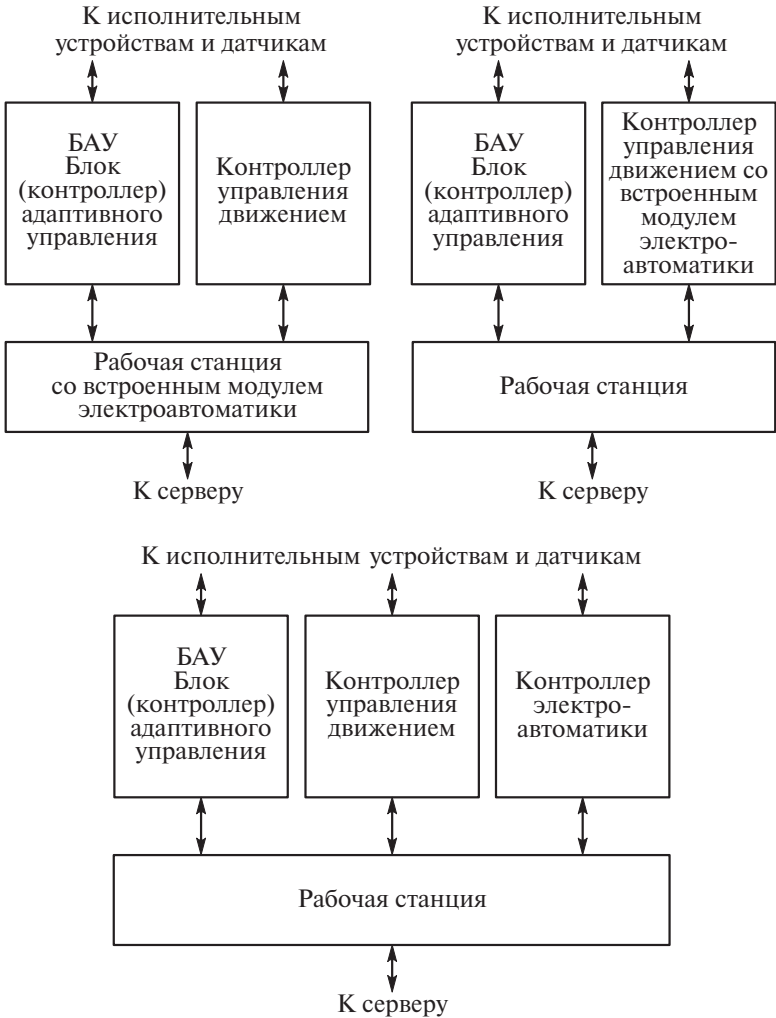


Рис. 8.11. Варианты архитектуры системы NC-2000

временных характеристик цепи обратной связи эта функция может быть реализована на специально выделенном контроллере либо быть дополнительной функцией базовой станции или любого контроллера, например контроллера управления движением.

На нижнем уровне NC-2000 расположены контроллеры, отвечающие за управление технологической машиной в режиме жесткого реального времени. Это контроллеры управления движением, контроллеры электроавтоматики, адаптивного управления и т. д. Все контроллеры объединены между собой и с базовой станцией в единую сеть и имеют индивидуальный сетевой адрес. Каждый контроллер получает задание от базовой станции, выполняет его и посылает базовой станции информацию о ходе выполнения задания.

Данная архитектура системы позволяет получить наивысшую скорость ее реакции на внешнее воздействие, сохранив обширные сетевые, графические и вычислительные возможности. Время отработки одного цикла управления движением NC-2000 не более 200 мкс, при этом один контроллер может управлять одновременно 32 осями. Гарантированное время реакции контроллеров системы может достигать 50 мкс. Количество контроллеров в системе NC-2000 в основном определяется сетевыми возможностями.

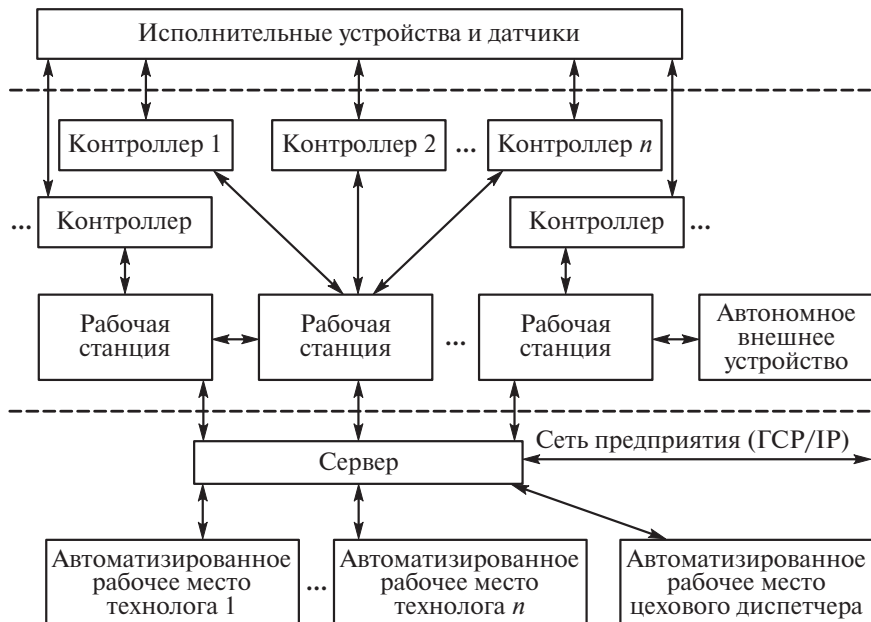


Рис. 8.12. Пример производственного участка с использованием NC-2000

Траектория движения инструмента по обрабатываемой поверхности представляет собой кусочно-ломаную линию, приближающуюся к теоретическому контуру поверхности с заданной точностью.

В обычных системах ЧПУ, чтобы перейти на следующий отрезок этой линии, узел станка должен сначала остановиться, дабы исключить «перебег» конечной точки отрезка и не нарушить точность обработки. Поэтому станок постоянно работает в режиме разгона/торможения и при малой длине отрезков (не более 10 мкм для достижения точности трехмерной поверхности IT8) не успевает выходить на заданную скорость подачи, что существенно снижает его реальную производительность. Чтобы преодолеть такую ситуацию, нужно заменить полное торможение торможением до некоторой величины скорости, при которой отклонение реальной траектории от запланированной не превышает заданного допуска, а для этого требуется информация о последующих перемещениях инструмента. Поэтому контроллер управления движением NC-2000 принимает к исполнению от базовой станции пакет буферизованных кадров, программно описывающих отдельные последовательные отрезки запланированной траектории движения инструмента, просматривает его и на основе полученной информации производит расчет текущей скорости подачи и управляет приводами. Число буферизованных кадров УП — не менее 100 (задается в параметрах системы). Расчет участков разгона/торможения производится из условия отсутствия превышения максимального ускорения в системе станка. Разгон/торможение может выполняться как по S-кривой, так и по линейному закону. В результате при реализации управляющей программой сложной траектории, представленной множеством «коротких кадров», полного торможения в конце отдельного кадра (перемещения, отрезка траектории) не происходит и скорость движения инструмента относительно заготовки стремится к заданному значению подачи. В процессе отработки траектории контроллер управления движением постоянно посылает базовой станции информацию о реальном положении узлов станка.

В момент загрузки и начала работы системы базовая станция устанавливает связь со всеми контроллерами, входящими в конфигурацию системы. В случае нарушения сетевого обмена контроллеры переходят в автономный режим работы. Базовая станция пытается восстановить с ними связь. После того как связь будет восстановлена, работа системы продолжится с точки останова. В системе предусмотрена возможность аварийной перезагрузки базовой станции, по окончании которой станция автома-

тически устанавливает связь с контроллерами и продолжает выполнение задания с точки останова. Во время перезагрузки базовой станции контроллеры находятся в автономном режиме, что позволяет не выключать технологическую машину и повторно не выводить оси в ноль, то есть не проводить поиск референтной точки инкрементными датчиками положения узлов машины.

Следует также добавить, что система NC-2000 позволяет вести программное управление несколькими процессами одновременно. В ней возможен как запуск нескольких процессов в пределах одной машины (разделение ресурса), так и нескольких процессов на разных машинах, имеющих свои контроллеры (локальный ресурс). Это позволяет использовать NC-2000 для управления автоматическими линиями, в том числе распределенными.

Еще одна особенность NC-2000 — она позволяет запускать на базовой станции стандартные Windows-приложения, включая CAD/CAM-системы. Таким образом, если на базовой станции использовать достаточно мощную процессорную плату, она может выполнять функции рабочего места конструктора-технолога в составе лабораторного комплекса. В случае упрощенного варианта системы базовая станция может выполнять и функции контроллера электроавтоматики (логического контроллера). При этом программное обеспечение контроллера запускается на базовой станции отдельным приложением и обменивается данными с ядром NC-2000.

Реализация системы. Перейдем теперь к технической стороне вопроса. Как уже говорилось, базовая станция выполнена в конструктиве промышленной рабочей станции фирмы Advantech. Внутри корпуса рабочей станции устанавливается процессорная плата со встроенным видеоадаптером и контроллером Ethernet. Там же устанавливается накопитель данных на жестком диске или накопитель на твердотельном диске (в зависимости от условий эксплуатации). Если NC-2000 применяется только в качестве системы управления станка с ЧПУ и не предусматривается использование дополнительных ресурсоемких приложений (например, CAD/CAM), вполне подходит процессорная плата с процессором Pentium MMX 266, имеющая 32 Мбайт оперативной памяти. При этом, если не используется скоростная эмуляция УП, загрузка процессора не превышает 30%. В таких условиях вполне нормально выполняются офисные приложения Windows.

Работая в среде Windows NT широко используются все возможности операционной системы, начиная с управления процессами и кончая графикой. В то же время считается, что чело-

веко-машинный интерфейс станка с ЧПУ (или любой другой технологической машины в цеховых условиях) должен быть максимально простым для персонала. Наладчик станка с ЧПУ не должен отвлекаться на работу с интерфейсом, он решает и без того сложные и требующие внимания задачи. В связи с этим интерфейс NC-2000 специально не использует мышь и не имеет перемещаемых произвольно окон. В этом смысле большим достоинством рабочих станций является наличие функциональной клавиатуры. Благодаря ей без труда строится двух-, трехуровневый интерфейс без использования «всплывающих» меню. Разумеется, мышь необходима, если в процессе работы системы управления используются обычные приложения Windows, но это уже дополнительная функция.

Все контроллеры системы NC-2000 строятся на основе промышленных шасси преимущественно фирмы Advantech. Это дорогая техника для применения в условиях цеха, если нет отрицательных температур. В жестких условиях эксплуатации (например, в условиях литейного производства) используется техник фирмы Octagon Systems. Изделия этой фирмы эксплуатируются в широком диапазоне рабочих температур, не имеют вращающихся деталей и хорошо подходят для контроллеров, применяющихся на необслуживаемых объектах. Они традиционно используются и хорошо себя зарекомендовали. Имеется опыт их непрерывной безотказной эксплуатации в течение 8 лет.

Контроллеры оснащаются платами с процессором 486 или Pentium MMX — этого вполне достаточно. Как правило, они не имеют человеко-машинного интерфейса. В редких случаях требуются простейшие терминалы в виде индикаторной панели. Загрузка контроллера осуществляется с флэш-диска при включении питания.

В качестве плат расширения в контроллере управления движением применяются в основном платы квадратурных счетчиков и ЦАП. Это универсальная схема построения системы управления движением. В качестве датчиков положения мы используем фотоимпульсные инкрементные датчики линейных и угловых перемещений. Контроллер электроавтоматики оснащается в основном платами дискретного ввода и вывода. В качестве плат дискретного ввода применяются платы PCL-733. Они обеспечивают гальваническую изоляцию входных сигналов от объекта управления. Для дискретного вывода чаще всего используется плата PCL-731. Она не имеет гальванической изоляции, но обладает достаточной нагрузочной способностью для подключения силовых твердо тельных модулей фирмы Grayhill. В разработках обыч-

но используются модули серии 70G. Они рассчитаны на ток до 3,5 А, чего вполне достаточно для непосредственного питания большинства исполнительных устройств электроавтоматики, таких как электромагниты, электромуфты, магнитные пускатели и т. д. Модули 70G имеют индивидуальный предохранитель, защищающий модуль от перегрузки, и индикатор включенного состояния (светодиод). Они легко устанавливаются и заменяются в случае необходимости. Решая с их помощью задачи коммутации цепей и сопряжения устройств, удастся существенно повысить надежность всей системы. Для включения более мощных, а также трехфазных нагрузок можно использовать силовые модули фирмы Omron.

Заключение. Система NC-2000 построена по модульному принципу и позволяет гибко конфигурировать ее в зависимости от потребностей объекта управления. Поэтому область применения NC-2000 не ограничивается металлорежущими станками: она разработана как универсальное базовое ядро для построения систем управления любыми машинами. В данный момент NC-2000 используется нами для построения систем управления металлорежущим станком фрезерной группы, промышленным роботом и автоматической линией по переработке пластмасс. Мы следуем принципу общности программного обеспечения. В результате, совершенствуя базовое ядро в одной предметной области, мы улучшаем все системы управления, построенные на его основе. Каким бы уникальным ни был объект, система управления им будет создана как развивающаяся и совершенствующаяся, а не как «тупиковая ветвь». Это облегчает удовлетворение постоянно растущих требований современного производства.

Отсутствие за последние три года отказов во внедренных системах ЧПУ подтвердило правильность выбора базовых технических средств.

8.2.9. АСУТП для резервуаров с реагентами цеха редких металлов

В этом разделе описываются новая автоматизированная система мониторинга и управления технологическими процессами в резервуарах с реагентами цеха редких металлов и комплекс мероприятий по ее поэтапному внедрению без остановки производства и в сжатые сроки. Благодаря выбранной элементной базе автоматизированная система отвечает требованиям высокой надежности в сочетании с достаточной производительностью по обработке измерительной информации.

Введение. Наряду с дальнейшим совершенствованием на Жезказганском заводе редких металлов («Жезказганредмет») технологического оборудования и существующей технологии извлечения редких металлов (рения, осмия и др.) важнейшим направлением повышения эффективности производства и качества конечного продукта является автоматизация на основе применения новейших измерительно-вычислительных комплексов для контроля технологических параметров.

Требования к качеству, надежности и безопасности систем мониторинга и управления технологическими процессами особенно возросли с введением на большинстве предприятий горно-металлургического комплекса Республики Казахстан программы управления качеством на основе международных стандартов ISO 9000 и ISO 14000. Это обстоятельство приводит к необходимости постоянно модернизировать соответствующие информационно-управляющие системы.

Важнейшей задачей, которую необходимо решить при создании автоматизированной системы управления технологическими процессами извлечения редких металлов, является разработка методов и средств контроля технологических параметров в резервуарах с реагентами.

В зависимости от конкретных условий на предприятии (технологических особенностей объектов, финансовых возможностей) процесс выполнения работ по оснащению новейшими измерительно-вычислительными комплексами контроля и управления может быть разовым или поэтапным. Важным критерием выбора здесь является возможность проведения модернизации без остановки технологического процесса на объекте.

Здесь приведем описание поэтапного выполнения работ по созданию системы мониторинга и управления технологическими процессами в резервуарах с реагентами цеха редких металлов.

Система учета реагентов в резервуарах. В системе учета реагентов в резервуарах использован объемно-массовый метод измерения массы жидкости (ГОСТ 26976—86). В соответствии с этим методом для измерения массы продукта необходимо измерить уровень, среднюю температуру и приведенную плотность жидкости в резервуарах, а также провести калибровку резервуара (составить калибровочные таблицы). Система учета осуществляет постоянное измерение уровня и температуры жидкости в резервуарах и производит вычисление ее массы, используя введенную относительную плотность продукта и данные из калибровочных таблиц.

На основании анализа результатов обследования резервуаров цеха редких металлов были определены технические и метроло-

гические требования к датчикам для контроля температуры и уровня жидкости в технологических емкостях.

Необходимо отметить, что в технологических емкостях цеха редких металлов находится агрессивная жидкость, поэтому для контроля ее уровня необходимо использовать бесконтактные измерительные средства. Наиболее полно удовлетворяют этому требованию уровнемеры, основанные на применении акустических и электромагнитных методов.

Акустические уровнемеры. Принцип действия акустических уровнемеров основан на локации уровня звуковыми импульсами, проходящими через газовую среду и отражающимися от границы раздела «газ — контролируемая среда». Для измерения уровня акустический преобразователь крепится на верхней крышке резервуара и не контактирует с контролируемой средой. Этот факт обеспечивает универсальность акустических уровнемеров по отношению к свойствам контролируемых сред.

Однако серийно выпускаемые отечественные акустические уровнемеры имеют существенные конструктивные недостатки, основными из которых являются недостаточная мощность звукового излучателя и плохая герметичность конструкции акустического преобразователя. Первый недостаток приводит к отказам в работе прибора при образовании конденсата на поверхности излучателя, а также к неустойчивой работе при наличии интенсивных испарений и резервуарах. Второй недостаток значительно затрудняет возможность использования этих приборов для контроля уровня агрессивных сред.

Локация уровня может производиться и снизу; при этом определяется толщина слоя жидкости над источником и приемником ультразвуковых колебаний, которые крепятся снаружи ко дну резервуара. На наш взгляд, локация снизу в ряде случаев (например, при измерении уровней находящихся под давлением жидкостей или сжиженных газов) может оказаться предпочтительнее, так как источник и приёмник функционируют в более благоприятных условиях и появляется возможность увеличения максимально допустимой высоты измеряемого уровня.

Результаты экспериментальных исследований показали, что акустические уровнемеры позволяют контролировать уровень реагентов с необходимой точностью в резервуарах с емкостью до 15 куб. м.

Радарные уровнемеры. Экспериментальные исследования, проведенные как в лабораторных условиях, так и непосредственно на баке №1—1 цеха редких металлов, показали высокую эффективность и точность уровнемеров, использующих методы из-

мерения на основе микроволнового электромагнитного излучения, в частности, радиолокационные (радарные) методы.

Принцип действия радарных уровнемеров также основан на измерении времени распространения микроволн от датчика до поверхности продукта и обратно. В настоящее время уровнемеры этого типа находят не большее применение для измерения уровня жидких сред. Это объясняется тем, что они удачно совмещают в себе высокую точность (погрешность порядка $\pm 1,0$ мм), надежность и простоту обслуживания. Высокая надежность определяется бесконтактностью метода, а высокая точность — частотой/длиной волны излучаемого сигнала и незначительным (всего несколько процентов даже для насыщенного водяного пара) ослаблением сигнала используемого диапазона при прохождении через газовую среду над измеряемым уровнем.

На баке № 1—1 были успешно проведены работы по монтажу, наладке и внедрению в эксплуатацию радарного уровнемера типа УЛМ-11. Работа уровнемера УЛМ-11 основана на принципе радиолокации в миллиметровом диапазоне волн. Уровнемер работает на частоте 94 ГГц, максимальная ошибка измерения ± 1 мм, диапазон измерений до 30 м.

Высокая точность измерений обеспечивается использованием прецизионной системы фазовой автоподстройки частоты и применением спектрального анализа с высокой разрешающей способностью. Благодаря использованию сверхкоротких электромагнитных волн уровнемер имеет малые размеры антенны по сравнению с другими уровнемерами. Это позволяет размещать весь уровнемер, включая и антенну, непосредственно на фланце резервуара с его внешней стороны.

Анализ различных типов радарных уровнемеров позволяет сделать вывод, что соотношение точности и размеров антенны в значительной мере зависит от используемой частоты излучения. Высокая точность уровнемеров, работающих на частотах 10 ГГц, достигается за счет использования антенны больших размеров, что усложняет их монтаж и эксплуатацию. Уровнемеры, работающие в более коротковолновом диапазоне, при той же точности используют антенны с меньшими геометрическими размерами, что значительно упрощает их монтаж и эксплуатацию. Данное обстоятельство послужило основанием для выбора в качестве первичного датчика радарного уровнемера типа УЛМ-11 для резервуаров, имеющих объем более 15 куб. м.

Специализированный измерительно-вычислительный комплекс. В ОАО «Казчерметавтоматика» разработан специализированный измерительно-вычислительный комплекс (ИВК) для контроля

температуры и уровня реагентов в технологических емкостях цеха редких металлов (ИВКТ-7685). Структурная схема ИВК приведена на рис. 8.13.

ИВК разработан на основе использования модуля микроконтроллера типа CPU 188-5MX в формате MicroPC фирмы Fastwel. Модуль CPU188-5MX позволяет наиболее полно реализовать задачи, поставленные на этапе разработки комплекса, при его относительно невысокой стоимости. Его функциональные возможности допускают подключение внешних устройств и организацию связи с компьютером верхнего уровня АСУТП через интерфейсы RS-485 или RS-232; через универсальный порт ввода-вывода реализуются функции управления индикацией, контроля цепей термопреобразователей сопротивления, управления сигнальными и исполнительными устройствами. Данная плата широко используется в разработках ОАО «Казчерметавтоматика», и имеется положительный опыт ее эксплуатации в условиях резко континентального климата в составе ряда измерительно-вычислительных комплексов для контроля технологических параметров на предприятиях горно-металлургического комплекса Республики Казахстан.

ИВК работает следующим образом. При включении питания прибор автоматически производит самотестирование для определения работоспособности. При обнаружении внутренней неисправности прибор прекращает работу. По завершении теста определяется наличие подключенных датчиков, установленных на технологических емкостях (баках, резервуарах), состояние датчиков записывается в ОЗУ. Далее определяется первый по списку резервуар и происходит обработка информации, полученной с датчиков уровня и термопреобразователей сопротивления. Вычисляются объем заполнения (согласно конфигурации бака), температура (согласно номинальным статистическим характеристикам, рассчитанным по ГОСТ 6651—84) и записываются в ОЗУ прибора. Параллельно информация передается в компьютер верхнего уровня и на выносное табло. Кроме этого, при достижении установленного порога (он задается с клавиатуры) срабатывает устройство сигнализации (звуковое, световое), после чего происходит определение номера текущего резервуара и опрос клавиатуры. Если была нажата клавиша, происходит обработка, то есть изменение установленных параметров и возвращение в основной текст программы. Если клавиша не нажата, то опрашиваются датчики следующего резервуара, то есть программа работает в цикле. Блок-схема алгоритма работы ИВК приведена на рис. 8.14.

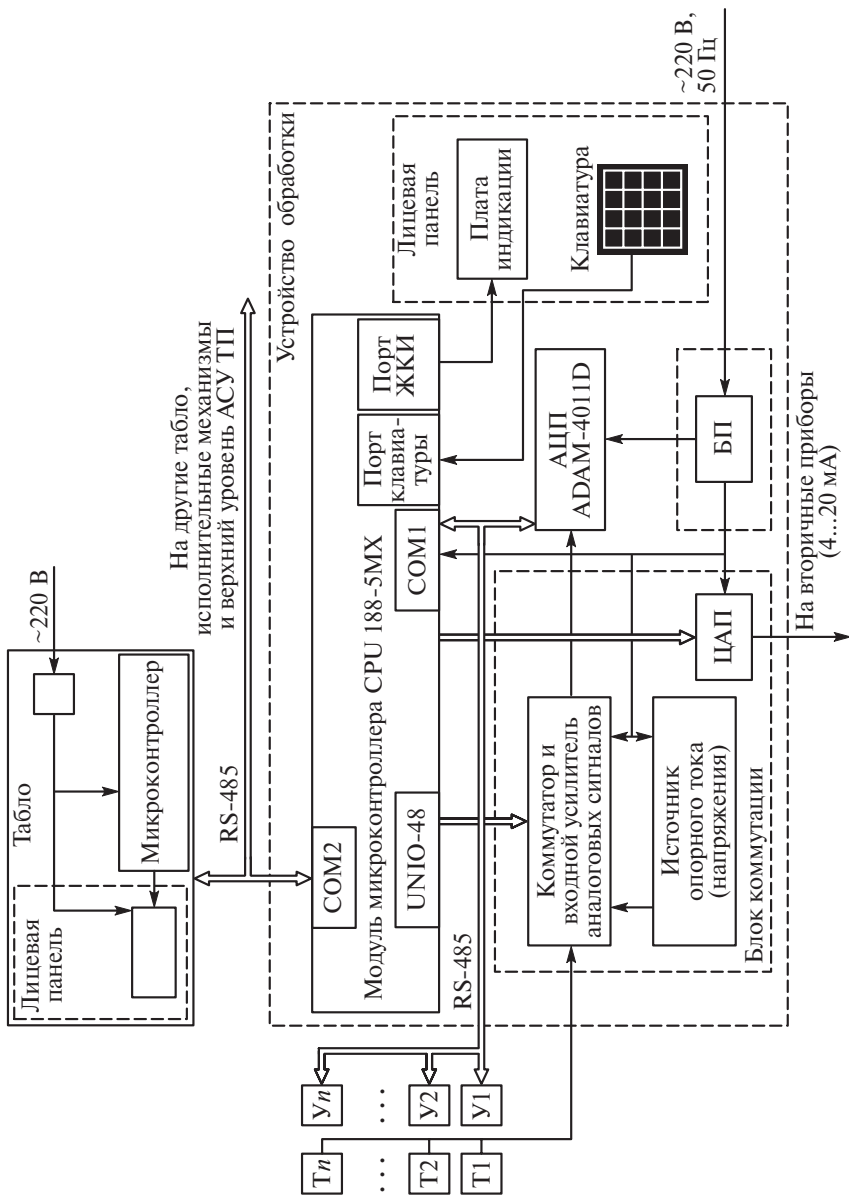


Рис. 8.13. Структурная схема ИВКТ-7685

Условные обозначения: Т – датчик температуры; У – датчик уровня; АЦП – аналого-цифровой преобразователь;; ЦАП – цифро-аналоговый преобразователь; БП – блок питания; ЖКИ – жидкокристаллический индикатор;; UNIO-48 – универсальный порт дискретного ввода-вывода (48 каналов)

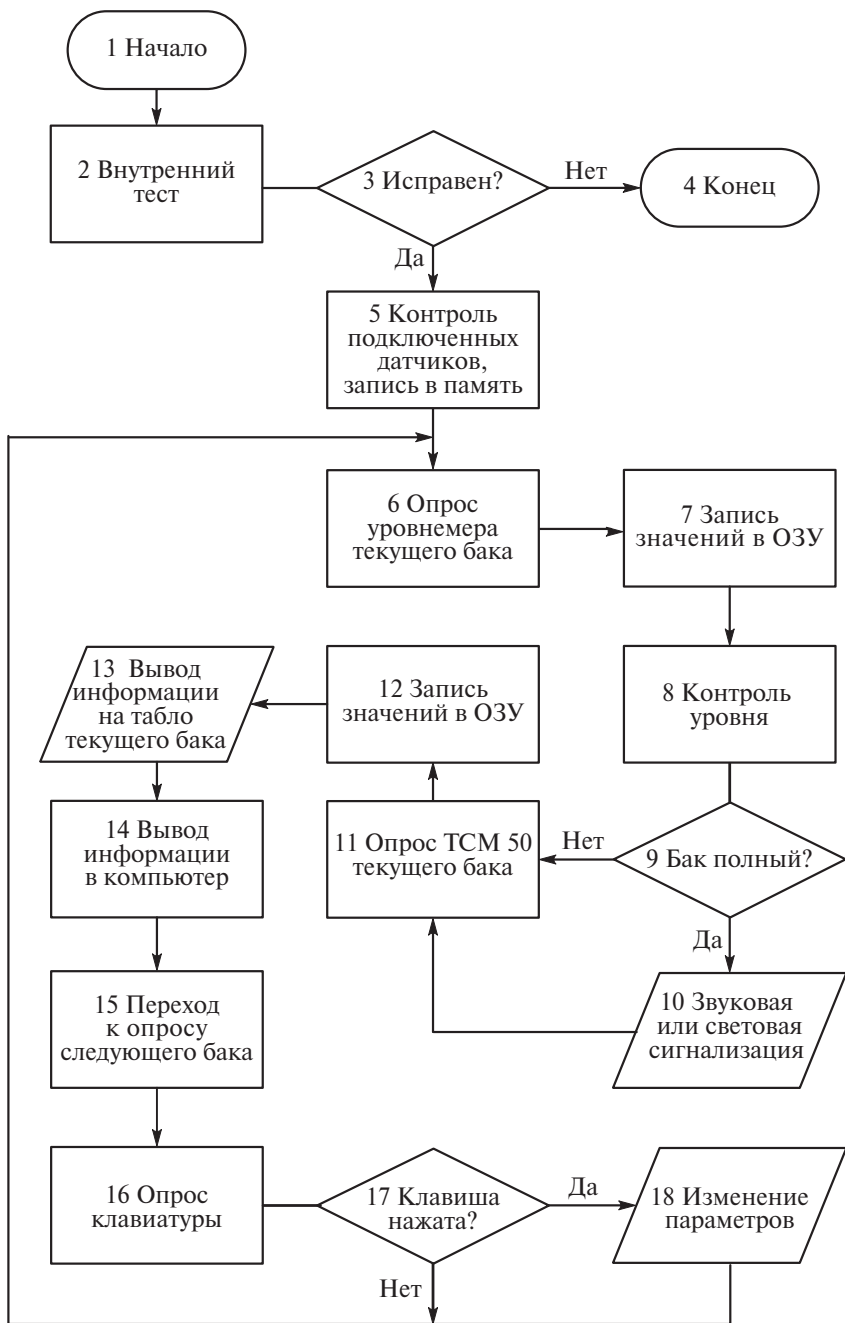


Рис. 8.14. Блок-схема алгоритма работы ИВК

Аналоговые сигналы с термопреобразователей сопротивлений типа ТСМ-50 через коммутатор поступают на вход АЦП ADAM-4011 D (фирма Advantech) и по интерфейсу RS-485 передаются в цифровом виде на модуль микроконтроллера, где происходит определение температуры технологической жидкости в резервуаре по заложенной номинальной статистической характеристике ТСМ-50 (ГОСТ 6651—84) с дальнейшей передачей полученных данных на табло и в компьютер более высокого уровня. Источник опорного тока (напряжения) формирует измерительные токи для ТСМ-50 и опорные напряжения для проведения автоматической калибровки аналоговых каналов.

Опрос уровнемеров и получение информации от них происходит по интерфейсу RS-485 в цифровом виде. Модуль микроконтроллера CPU 188-5MX производит вычисление объема по полученным данным и также передает информацию на табло и компьютер более высокого уровня.

Клавиатура служит для ввода основных параметров, определяющих рабочие диапазоны прибора (размеры резервуара, диапазон измерения температуры, коэффициент для компенсации изменений расстояния от уровнемера до поверхности жидкости).

Плата индикации используется оператором для контроля изменений параметров прибора, так как данная информация на табло не отображается. Кроме того, на плате индикации с помощью клавиатуры можно просмотреть текущие значения параметров любой технологической емкости.

Прибор аналогично работает по остальным каналам и отслеживает все изменения и в других резервуарах.

Для работы специализированного ИВК с датчиками через шину fieldbus предусмотрен вариант исполнения комплекса с возможностью подключения интеллектуальных интерфейсных плат фирмы Hilscher, обеспечивающих доступ к различным промышленным сетям. В этом случае платы ИВК и интерфейсные платы размещаются в монтажном каркасе с 8-разрядной шиной ISA.

Кроме того, комплекс допускает подключение датчиков и исполнительных устройств через AS-интерфейсы и шлюзы фирмы Pepperl+Fuchs, что расширяет коммутационные возможности ИВК и номенклатуру допустимых устройств нижнего уровня.

Обеспечение точности измерений. Для обеспечения точности измерений в алгоритме работы специализированного комплекса ИВКТ-7685 предусмотрена возможность автоматической калибровки аналоговых каналов с целью компенсации погрешностей и устранения дрейфа коэффициента усиления и смещения нулево-

го уровня входного усилителя аналоговых сигналов, входящего в состав блока коммутации.

Процесс автоматической калибровки входного усилителя осуществляется по следующему алгоритму. Через коммутатор аналоговых сигналов к нему подключаются два источника опорных напряжений, параметры которых U_1 и U_2 известны. Источники опорных напряжений реализованы на микросхеме REF 195.

Величина выходных напряжений входного усилителя при подключении первого и второго источника опорных напряжений соответственно составит:

$$\begin{aligned}U_{a1} &= U_0 + kU_1, \\U_{a2} &= U_0 + kU_2.\end{aligned}$$

Здесь U_{a1} — величина выходного напряжения входного усилителя при подключении к нему источника опорного напряжения, B ; U_{a2} — величина выходного напряжения входного усилителя при подключении к нему второго источника опорного напряжения, B ; U_0 — величина смещения нулевого уровня входного усилителя на его выходе, B ; k — коэффициент усиления; U_1 — величина напряжения первого источника опорного напряжения, B ; U_2 — величина напряжения второго источника опорного напряжения, B .

Решая полученную систему линейных уравнений относительно коэффициента усиления и величины смещения нулевого уровня входного усилителя на его выходе, получим:

$$\begin{aligned}k &= (U_{a1} - U_{a2}) / (U_1 - U_2), \\U_0 &= [U_{a1} + U_{a2} - k(U_1 + U_2)] / 2.\end{aligned}$$

Следовательно, исходя из замеренного на выходе входного усилителя параметра напряжения U_a , можно расчетным путем определить неизвестную величину входного напряжения U_g :

$$U_g = (U_a - U_0) / k.$$

Таким образом, путем автоматической калибровки входного усилителя исключаются погрешности, возникающие из-за дрейфа коэффициента усиления и смещения нулевого уровня входного усилителя.

Заключение. В ОАО «Казчерметавтоматика» работа над специализированным измерительно-вычислительным комплексом для контроля температуры и уровня реагентов в технологических емкостях цеха редких металлов успешно завершена проведением

Государственных приемочных испытаний с получением сертификационных документов в полном объеме. Осуществлена работа по монтажу, наладке и внедрению в опытно-промышленную эксплуатацию автоматизированной системы мониторинга и управления технологическими процессами в резервуарах с реагентами цеха редких металлов Дочернего государственного предприятия (ДГП) «Жезказганредмет».

Это внедрение позволило:

- повысить точность измерения уровня реагентов в резервуарах, снизив погрешность с ± 10 мм (ручной способ с использованием мерной штанги) до ± 1 мм, благодаря чему удалось стабилизировать параметры технологических процессов и увеличить выход конечной продукции;
- устранить необходимость ручного труда при измерении уровня реагентов в резервуарах;
- улучшить условия труда (снизить загазованность рабочих мест из-за отказа от измерений вручную и связанных с этим открываний специальных люков резервуаров, обеспечить возможность дистанционного контроля и управления, повысить безопасность труда и др.).

8.2.10. Подсистема АСУ З-03Р системы управления и защиты исследовательского ядерного реактора ПИК

В этом разделе представлен принципиально новый подход к задаче построения интегрированного канала формирования сигналов управляющей системы безопасности, используемого в аппаратуре системы управления и защиты для исследовательского ядерного реактора ПИК Петербургского института ядерной физики Российской академии наук.

Введение. Системы обеспечения безопасной работы реакторных установок возникли вследствие необходимости контроля состояния активной зоны реактора, в первую очередь нейтронного потока, затем теплогидравлических параметров. С введением в эксплуатацию электростанций, работающих на ядерном топливе, системам контроля и обеспечения безопасной работы ядерных реакторов стали уделять большее значение.

Значительную часть атомных объектов составляют исследовательские реакторы и критические стенды, предназначенные для научных исследований в области ядерной физики, физики твердого тела, технологии элементов ядерных реакторов, медицины и т. д. Контроль, управление и обеспечение безопасной работы таких ядерно-опасных объектов также немаловажны.

ЗАО «СНИИП-СИСТЕМАТОМ» («ССА», до 1994 г. — отделение НИЦ СНИИП) занимается разработкой аппаратуры обеспечения безопасности ядерных реакторов более тридцати лет. За это время были подтверждены основные концепции построения аппаратуры, изготовлены и установлены системы управления и защиты ядерных реакторов на Кольской, Балаковской, Калининской, Ново-Воронежской, Ровенской, Южно-Украинской, Запорожской АЭС, на АЭС в Ловиизе (Финляндия), Козлодуге (Болгария), Моховце (Словакия) и других. Оснащены аналогичными системами контроля и управления защитой исследовательские реакторы центра ядерных исследований (ЦЯИ) в г. Тажура (Ливия) и института ядерных исследований (ИЯИ) в г. Далат (Вьетнам). Аппаратура, установленная на этих ядерно-опасных объектах, обеспечивает по сей день их надежную работу. Например, АЭС в Ловиизе введена в строй в 1975 г., исследовательский реактор при ИЯИ в г. Далат введен в эксплуатацию в 1984 г., и до настоящего времени аппаратура на этих объектах находится в рабочем состоянии.

Основные принципы физики работы ядерных реакторов остаются неизменными. Однако аппаратура, осуществляющая управление ядерными реакторами, как аналоговая, так и цифровая и особенно вычислительная техника, за последние четверть века шагнула далеко вперед. Бурное развитие и повышение надежности микросхем с высокой степенью интеграции и микроконтроллеров открыло перспективы дальнейшего развития и модернизации систем контроля и управления защитой. Аппаратура системы контроля и управления защитой реакторных установок, разрабатываемая в «ССА», прошла все этапы развития электронной техники: первые комплексы были выполнены с применением аналоговой техники (середина 1970-х годов), затем была создана и введена в постоянную эксплуатацию первая в СССР аппаратура контроля нейтронного потока на основе цифровой техники (1987 г., Кольская АЭС). Началось широкое внедрение цифровой техники в систему контроля теплогидравлических параметров, аппаратуру формирования логических сигналов управления реакторной установкой, устройства отображения и регистрации контролируемых параметров.

На рубеже столетий (1999 г.) был разработан и введен в постоянную эксплуатацию комплекс аппаратуры системы контроля и управления защитой на втором блоке Кольской АЭС (фотография Кольской АЭС представлена на первой странице обложки журнала). Аппаратура формирования защитных сигналов по параметрам нейтронного потока и теплогидравлическим параметрам

рам этого комплекса традиционно была выполнена на основе хорошо зарекомендовавшей себя цифровой аппаратуры разработок середины 1980-х годов. Наряду с этим аппаратура формирования управляющих сигналов аварийной защиты была выполнена на логических матрицах фирмы Altera. Практически вся аппаратура систем контроля, не входящая в состав системы управления аварийной защитой реактора (это аппаратура контроля перегрузки, автоматический регулятор мощности, аппаратура контроля реактивности, аппаратура сигнализации первопричины аварийной ситуации, устройства отображения и протоколирования информации), была выполнена с применением процессорной техники на основе изделий фирм Octagon Systems, Advantech, Analog Devices, Grayhill, Planar, Bopla, Arlesyn Technologies.

С 1997 г. в «ССА» получило развитие новое направление в области разработки систем контроля и управления защитой ядерных реакторов — применение специализированных микроконтроллеров в аппаратуре управляющей системы безопасности и управления технологической автоматикой. Применение микроконтроллеров подняло аппаратуру на более высокий технический уровень, повысило ее надежность и помехоустойчивость, привело к сокращению объема аппаратуры, сделало ее гибкой и легко адаптируемой к конкретному проекту, более эргономичной по отношению к обслуживающему персоналу.

Структура комплекса. Комплекс аппаратуры АСУЗ-03Р является составной электронной частью системы управления и защиты (СУЗ) исследовательского ядерного реактора ПИК. Он осуществляет контроль и обеспечивает безопасность эксплуатации реакторной установки во всех режимах: во время пуска реактора, работы на заданном уровне мощности, перегрузки (загрузки) топлива, а также при возникновении аварийной ситуации и при остановленном реакторе. В соответствии с НП-033-01 «Общие положения обеспечения безопасности исследовательских ядерных установок» (ОПБ ИЯУ) элементы комплекса подразделяются на следующие классы безопасности: 2УН, 3Н, 3У, 4.

Данный комплекс представляет собой логически завершенную структуру с наличием постоянного программного и аппаратного контроля подсистем управления аварийной защитой, оперативного контроля текущего состояния, долговременной архивации и документирования информации, автоматической предпусковой проверки. Комплекс выполнен на основе модулей с применением стандартных логических элементов, логических матриц с высокой степенью интеграции, специализированных микроконтроллеров, разработанных в «ССА», промышленных компьютеров и модулей ввода-выво-

да фирмы Octagon Systems, преобразователей сигналов фирм Analog Devices и Grayhill, электролюминесцентных дисплеев фирмы Planar, преобразователей напряжения фирмы Artesyn Technologies, промышленной рабочей станции фирмы Advantech.

В состав комплекса АСУЗ-03Р (рис. 8.15) входят четыре канала контроля и формирования защитных сигналов по параметрам нейтронного потока, три канала контроля и формирования защитных сигналов по теплогидравлическим параметрам, три канала формирования управляющих сигналов для защитной системы безопасности и технологической автоматики, четыре канала контроля реактивности, четыре регулятора мощности, один канал контроля положения рабочих органов.

Контроль реактивности и регулирования мощности реакторной установки осуществляется по информации, поступающей от блоков детектирования нейтронного потока, участвующих в формировании сигналов защиты по параметрам нейтронного потока.

Комплекс АСУЗ-03Р также обеспечивает контроль нейтронного потока по двум независимым каналам при помощи аппаратуры, расположенной на резервном щите управления (РЩУ).

Контроль оперативной информации и состояния исправности реакторной установки реализуется посредством нескольких устройств, работающих одновременно:

- три аналоговых дисплея пульта оператора блочного щита управления (БЩУ) и один аналоговый дисплей РЩУ, предназначенные для отображения в виде гистограмм и в цифровом виде значений теплогидравлических параметров, параметров нейтронного потока, реактивности, величины разбаланса, положения рабочих органов, состояния исправности блоков и устройств комплекса;
- четыре строчных цифровых дисплея, предназначенные для отображения поканального значения мощности нейтронного потока и величины обратной скорости изменения плотности нейтронного потока (период разгона реактора);
- самопишущие регистраторы для непрерывной записи поканальной и усредненной величины мощности нейтронного потока, поканального и минимального значения периода разгона реактора;
- рабочая станция и принтер, предназначенные для отображения, архивирования, протоколирования параметров комплекса во всех режимах работы реактора, а также диагностики блоков, участвующих в формировании защитных сигналов, и проверки систем технологической автоматики в режиме останова реактора.

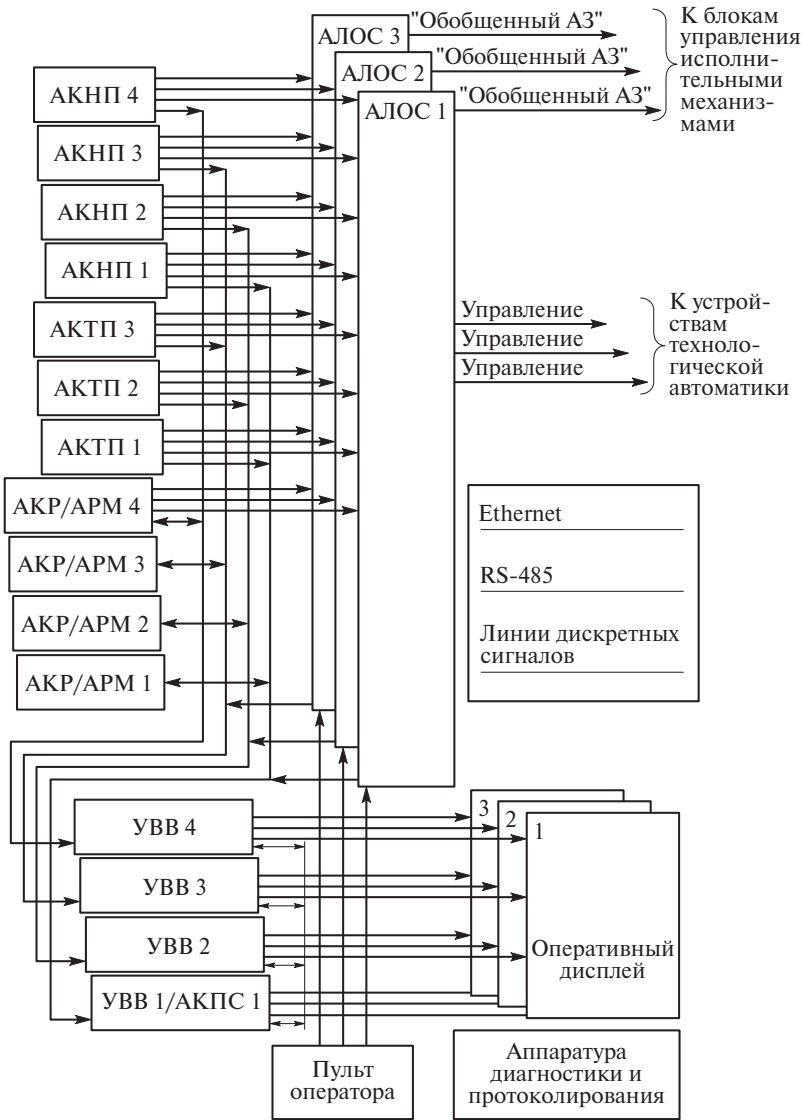


Рис. 8.15. Структурная схема комплекса АСУЗ-03Р:

АКНП – аппаратра контроля нейтронного потока; АКТП – аппаратра контроля теплогидравлических параметров; АЛОС – аппаратра логической обработки сигналов; АКР – аппаратра контроля реактивности; АРМ – автоматический регулятор мощности; АКПС – аппаратра контроля положения стержней; УВВ – управление вводом-выводом информации; «Обобщенный АЗ» – обобщенный управляющий сигнал аварийной защиты; «Управление» – сигналы управления устройствами технологической автоматики

Комплекс АСУЗ-03Р передает информацию в систему верхнего уровня РАКУРС — распределенный автоматизированный комплекс управления реакторной установкой — и получает от этой системы сигналы единого времени.

Аппаратура комплекса АСУЗ-03Р размещена в различных помещениях здания исследовательского реактора:

- блоки детектирования, усилители, преобразователи, блок сигнализации разгона — в реакторном зале;
- устройства накопления и обработки информации, устройства коммутации — в отдельных помещениях аппаратуры СУЗ;
- устройства отображения информации, самопишущие регистраторы, блок сигнализации разгона, органы управления реакторной установкой, аппаратура диагностики и протоколирования (АДП) — в помещении БЩУ;
- устройство накопления и обработки информации, устройства отображения и регистрации информации — в помещении РЩУ.

Основные принципы построения комплекса. При разработке комплекса АСУЗ-03Р были реализованы следующие принципы структурного построения, концепции организации обмена информацией и требования к программному обеспечению.

Комплекс имеет традиционную канальную («ниточную») структуру и включает в себя аппаратуру, начиная от устройств детектирования и заканчивая устройством формирования обобщенного управляющего сигнала аварийной защиты («Обобщенный АЗ») и управляющих сигналов для технологической автоматики. Структура комплекса АСУЗ-03Р обеспечивает полную конструктивную, электрическую и функциональную независимость каналов.

В комплексе используется принцип проектирования устройств и блоков двойного функционального назначения. Аппаратура формирования сигнала «Обобщенный АЗ», выполняющая функции защиты, контроля и управления, выполнена в виде единого устройства накопления и обработки информации (УНО) с блочной архитектурой, каждый блок — составная часть канала контроля и защиты.

Аппаратура, формирующая сигналы для защитной системы безопасности, выполнена с применением специализированных микроконтроллеров и классифицируется как аппаратура класса 2УН.

Аппаратура, выполняющая функции контроля, построена с использованием микропроцессоров промышленного назначения фирмы Octagon Systems и является аппаратурой класса 3Н и 3У.

Единая для всего комплекса АСУЗ-03Р аппаратура, обеспечивающая диагностику, архивацию, протоколирование, предпусковую проверку, передачу информации на верхний уровень, — аппаратура диагностики и протоколирования (АДП) — реализована на базе промышленного компьютера фирмы Advantech. Это аппаратура 4-го класса.

Программное обеспечение для аппаратуры класса 2 разработано на ассемблере, является новым, не содержащим встроенной операционной системы MS-DOS и BIOS. Остальное программное обеспечение разработано на языке Паскаль и функционирует под управлением MS-DOS совместимой операционной системы.

Входной информацией для комплекса являются значения плотности потока тепловых нейтронов, унифицированные токовые сигналы 4...20 мА, сигналы с выходов термопреобразователей сопротивления (ТПС), поступающие от измерительных устройств и датчиков тешюгидравлических параметров, а также сигналы с выходов переменных сопротивлений ПТП-22, контролирующих положение рабочих органов, и дискретные сигналы состояния обеспечивающих систем.

Обмен информацией между блоками и устройствами комплекса осуществляется как по стандартным интерфейсам RS-485 и Ethernet, так и в виде обмена частотными и дискретными сигналами.

Все входные и выходные линии связи между устройствами комплекса, а также между комплексом и другими системами гальванически развязаны и выдерживают испытательное напряжение 1500 В.

Электропитание комплекса осуществляется от четырех вводов надежного источника напряжения (220 В/50 Гц) первой категории.

Электропитание АДП и самопишущих регистраторов производится от сети переменного напряжения (220 В/50 Гц) БЩУ.

Электропитание остальных блоков и устройств комплекса осуществляется от шины постоянного напряжения 24 В, которое вырабатывается устройствами УНО.

Аппаратура комплекса изготовлена с применением как конструктивов, разработанных в «ССА», так и стандартных конструктивов и элементов конструкций фирмы Schroff.

Все это позволило унифицировать модули (узлы), стандартизировать блоки, объединить в одну стойку несколько функционально завершенных блоков. Таким образом был создан интегрированный канал УНО, формирующий управляющий сигнал для защитной системы безопасности и системы нормальной эксплуатации, выполняющий следующие функции:

- контроль уровня и скорости нарастания плотности нейтронного потока, формирование соответствующих защитных сигналов;
- контроль теплогидравлических параметров и формирование соответствующих защитных сигналов;
- логическая обработка сигналов и формирование управляющих сигналов аварийной защиты;
- логическая обработка и формирование управляющих сигналов для устройств технологической автоматики;
- групповое и индивидуальное управление рабочими органами;
- контроль положения рабочих органов (стержней);
- контроль реактивности;
- регулирование мощности;
- управление потоком информации.

В состав интегрированного канала входит электронная часть, обрабатывающая и формирующая сигналы «ниточных» каналов контроля. Функциональная интеграция происходит на уровне формирования управляющего сигнала аварийной защиты и сигналов для устройств технологической автоматики. Конструктивная интеграция заключается в совмещении логической части «ниточных» каналов контроля в одном устройстве — УНО.

Функциональная схема интегрированного канала управляющей системы безопасности представлена на рис. 8.16.

Описание составных частей комплекса.

Каналы контроля и аварийной защиты по параметрам нейтронного потока. Каналы данного типа включают аппаратуру контроля нейтронного потока (АКНП), состоящую из устройства детектирования нейтронного потока (УДНП), блока защиты по параметрам нейтронного потока (БЗНП), блока задания уставок по мощности АКНП (ЗУ-АКНП), цифрового дисплея (ЦД-АКНП), а также оперативные дисплеи пульта оператора и самопишущие регистраторы. Структурная схема канала АКНП представлена на рис. 8.17.

Уровень мощности реактора контролируется шестью независимыми каналами измерения плотности потока тепловых нейтронов. Четыре из них предназначены для контроля параметров и формирования сигналов аварийной защиты по мощности и периоду разгона реактора, два — только для контроля параметров с РЩУ. Необходимая информация по шести каналам отображается на дисплеях пульта оператора БЩУ и РЩУ.

Блоки детектирования АКНП размещены в каналах ионизационных камер (каналы ИК). Это сухие вертикальные окруженные свинцовым экраном каналы.

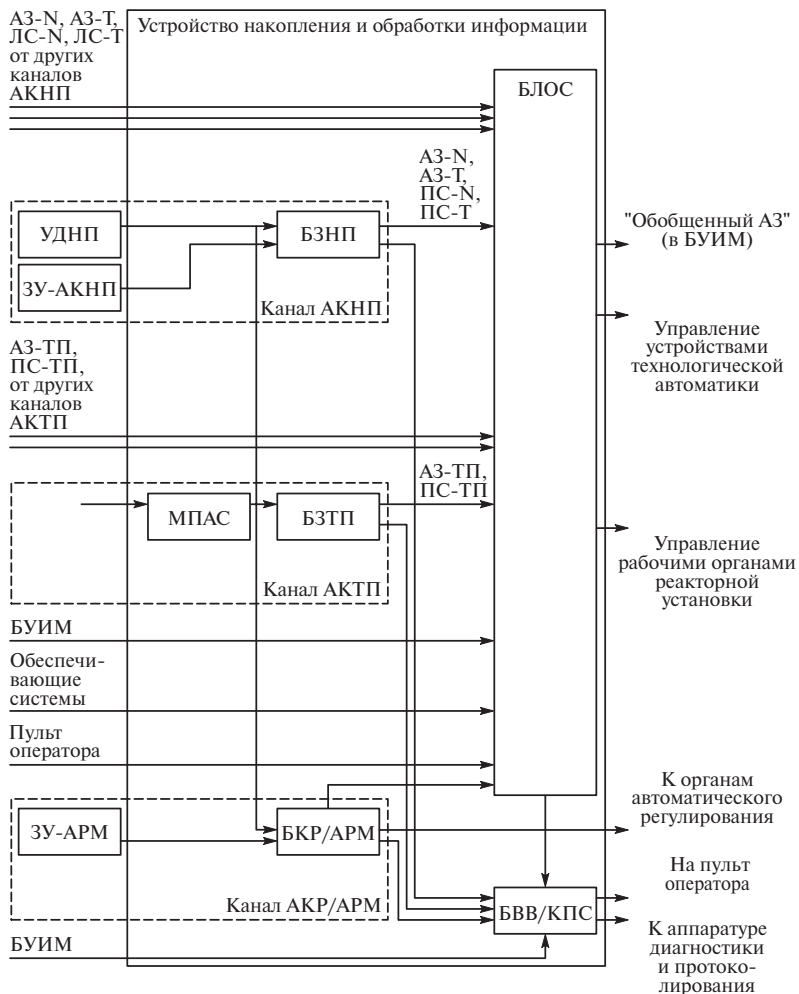


Рис 8.16. Структурная схема интегрированного канала формирования сигналов управляющей системы безопасности исследовательского реактора: УДНП – устройство детектирования нейтронного потока; БЗНП – блок защиты по параметрам нейтронного потока; ЗУ-АКНП – блок задания уставок по мощности АКНП; БЗТП – блок защиты по теплогидравлическим параметрам; БЛОС – блок логической обработки и формирования сигналов для блоков управления исполнительными механизмами (БУИМ), устройств технологической автоматики и группового и индивидуального управления рабочими органами (РО); БКР/АРМ – блок контроля реактивности и автоматического регулирования мощности; ЗУ-АРМ – блок задания уставок АРМ; БВВ/КПС – блок контроля положения стержней и ввода-вывода информации; АЗ и ПС – сигналы аварийной защиты и предупредительной сигнализации по мощности (N), периоду разгона (Т), теплогидравлическим параметрам (ТП); МПАС – модуль преобразования аналоговых сигналов

При номинальном уровне мощности с учетом свинцовых экранов в местах расположения блоков детектирования плотности потока тепловых нейтронов не превышает $1,2 \times 10^{10}$ нейтр./ $(\text{см}^2 \cdot \text{с})$ (примем сокращенное обозначение «нейтр.» — нейтрон), мощность дозы гамма-излучения — не более 1×10^6 Р/ч.

Для измерения плотности потока тепловых нейтронов используется устройство детектирования УДПН-20Р, состоящее из блоков детектирования БДПН-28Р и БДПН-23Р, коробки соединительной КС-1, двух блоков преобразования БПХ-67Р и БПХ-67Р1, блока вспомогательного БХ-152Р. Устройство УДПН-20Р преобразует плотность потока нейтронов в частоту следования импульсов; частота следования импульсов пропорциональна тепловой мощности реактора.

Блоки детектирования БДПН-28Р и БДПН-23Р, установленные в каждом канале ИК, размещены на разной высоте относи-

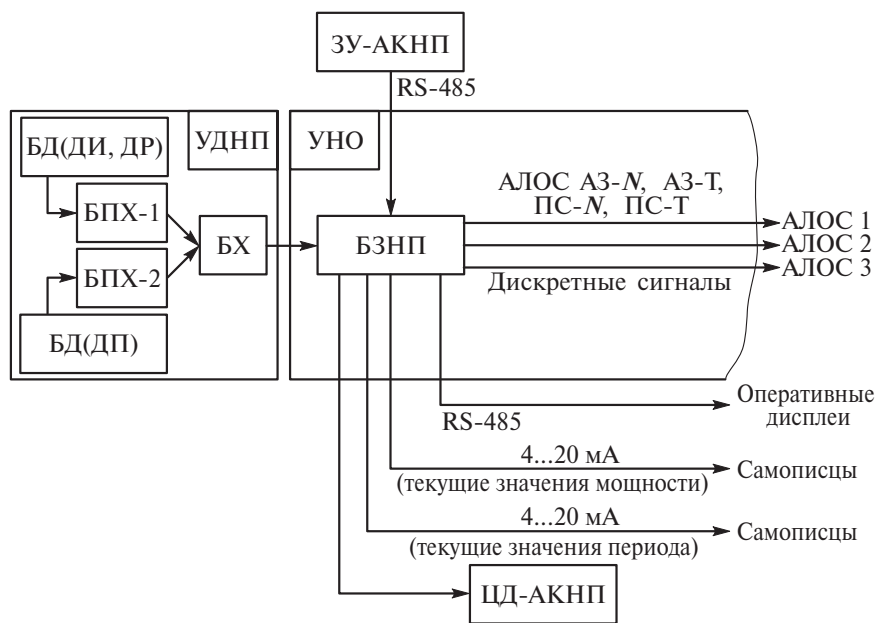


Рис 8.17. Структурная схема канала АКНП:

УДНП — устройство детектирования нейтронного потока; БД (ДИ, ДР) — блок детектирования диапазонов источника и рабочего; БД (ДП) — блок детектирования диапазона промежуточного; БПХ 1 и БПХ 2 — блоки преобразования и усиления сигналов; БХ — блок вспомогательный; БЗНП — блок защиты по параметрам нейтронного потока; ЗУ-АКНП — блок задания уставок по мощности АКНП; ЦД-АКНП — цифровой дисплей, отображающий мощность и период канала АКНП; АЗ-*N*, ПС-*N*, АЗ-*T*, ПС-*T* — сигналы аварийной защиты и предупредительной сигнализации по мощности (*N*) и по периоду разгона реактора (*T*)

тельно центра активной зоны реактора. Блок БДПН-28Р находится по центру активной зоны, блок БДПН-23Р смещен по высоте относительно центра активной зоны. Блок БДПН-28Р изготовлен на основе импульсной ионизационной урановой камеры деления типа КНК-15, работает в двух режимах (импульсном и токовом) и обеспечивает контроль плотности нейтронного потока в диапазоне источника (ДИ) от 1×10^0 до 1×10^6 нейтр./ $(\text{см}^2 \cdot \text{с})$ ($1 \times 10^{-8} \dots 1 \times 10^{-2} \% N_{\text{НОМ}}$, $N_{\text{НОМ}}$ — номинальное значение мощности реакторной установки) и в рабочем диапазоне (ДР) от 1×10^7 до $1,2 \times 10^{10}$ нейтр./ $(\text{см}^2 \cdot \text{с})$ ($1 \times 10^{-1} \dots 1,2 \times 10^2 \% N_{\text{НОМ}}$). Блок детектирования БДПН-23Р на основе ионизационной урановой камеры деления предназначен для обеспечения перекрытия диапазонов ДИ и ДР (промежуточный диапазон — ДП) и измерения плотности нейтронного потока в пределах от 1×10^5 до 1×10^8 нейтр./ $(\text{см}^2 \cdot \text{с})$ ($1 \times 10^{-3} \dots 1 \times 10^0 \% N_{\text{НОМ}}$).

Таким образом обеспечивается контроль мощности реактора в диапазоне от 1×10^{-8} до $1,2 \times 10^2 \% N_{\text{НОМ}}$ ($1 \times 10^0 \dots 1,2 \times 10^{10}$ нейтр./ $(\text{см}^2 \cdot \text{с})$), при этом перекрытие поддиапазонов составляет более одного десятичного порядка и может корректироваться перемещением блока детектирования промежуточного диапазона (БДПН-23Р) относительно центра активной зоны в период настройки канала АКНП. В процессе эксплуатации блоки детектирования не перемещаются.

Частотные сигналы от устройства УДПН-20Р поступают в блок защиты по параметрам нейтронного потока БПМ-80Р устройства УНО для вычисления текущего значения мощности и периода разгона реактора и формирования поканальных сигналов аварийной защиты и предупредительной сигнализации по мощности (АЗ- N , ПС- N) и по периоду (АЗ- T , ПС- T). В блоке БПМ-80Р функции приема частотных сигналов от устройства детектирования, вычисления текущих значений мощности и периода, приема текущего значения уставок АКНП, формирования аварийных и предупредительных сигналов, управления встроенными дисплеем и клавиатурой, а также передачи информации по интерфейсам RS-485 возложены на один модуль, выполненный на базе микроконтроллера.

Формирование сигналов АЗ- N и ПС- N осуществляется при превышении текущего значения мощности над установленным в ЗУ-АКНП (БКЦ-73Р) значением уставки для АЗ- N . При этом величина уставки для ПС- N определяется соотношением (уставка для АЗ- M)/(уставка для ПС- N) = 120/110 в диапазоне контроля мощности от 1×10^{-7} до $1,2 \times 10^2 \% N_{\text{НОМ}}$. Дискретность задания уставки для АЗ- N в диапазоне от 1×10^{-7} до $1 \% N_{\text{НОМ}}$ — 10%, в диапазоне от 1 до $1,2 \times 10^2 1 \% N_{\text{НОМ}}$ — 1%. Количество бло-

ков задания уставок по мощности равно числу каналов АКНП. Блоки БКЦ-73Р расположены на пульте оператора, передача значения уставки АЗ-*N* осуществляется по последовательному интерфейсу RS-485.

Формирование сигналов АЗ-Т, ПС-Т осуществляется, если текущее значение периода разгона реактора превысит значение уставки для АЗ-Т. Величина уставки для ПС-Т определяется как $2 \times$ (уставка для АЗ-Т). Величина уставки для АЗ-Т задается непосредственно с помощью встроенной клавиатуры блока БПМ-80Р и может принимать целочисленные значения от 0 до 999 с.

Блок БПМ-80Р формирует и передает по последовательному интерфейсу RS-485 информацию в БЩУ для последующего отображения значений мощности и периода на цифровом дисплее ЦД-АКНП (БИЦ-63Р). Значения контролируемых параметров отображаются также на дисплеях пульта оператора (блоки БИЦ-52Р) и регистрируются самописцами типа РП-160.

Каналы контроля и аварийной защиты по теплогидравлическим параметрам. Каналы этого типа включают аппаратуру контроля теплогидравлических параметров (АКТП), состоящую из устройств, измеряющих давление, температуру, уровни и расход воды реактора, и блока защиты по теплогидравлическим параметрам (БЗТП), а также оперативные дисплеи пульта оператора. Структурная схема канала АКТП изображена на рис. 8.18.

В устройствах УНО блок защиты по теплогидравлическим параметрам БПМ-79Р осуществляет вычисление значений контро-

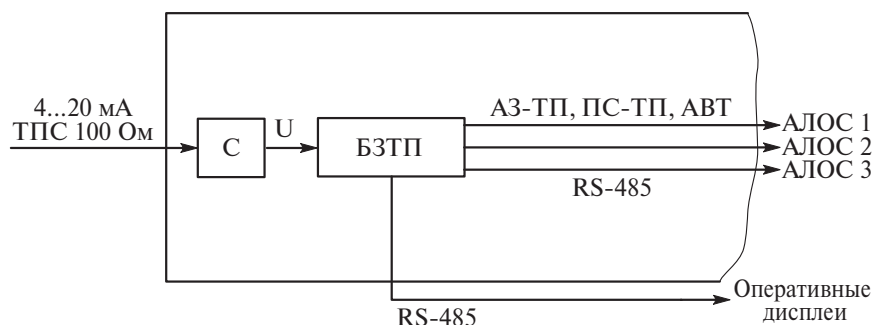


Рис. 8.18. Структурная схема канала АКТП:

БЗТП – блок защиты по теплогидравлическим параметрам; С – модули преобразования и гальванической развязки фирмы Analog Devices; АЗ-ТП и ПС-ТП – сигналы аварийной защиты и предупредительной сигнализации по теплогидравлическим параметрам; АВТ – сигналы управления устройствами технологической автоматики; ТПС – термопреобразователи сопротивления

лируемых величин и формирование сигналов аварийной защиты (АЗ-ТП), предупредительной сигнализации (ПС-ТП) и сигналов управления устройствами технологической автоматики (АВТ) по результатам сравнения текущего значения каждого параметра с соответствующими уставками. В блоке БПМ-79Р эту функцию осуществляет один модуль, выполненный с применением микроконтроллера, обеспечивающего, кроме того, управление 24 аналого-цифровыми преобразователями, встроенными дисплеем и клавиатурой, а также передачу данных по интерфейсам RS-485.

Первичными преобразователями, контролирующими давление, расход и уровень воды, являются преобразователи типа «Сапфир-22М» класса точности 0,5. Диапазон изменения выходного сигнала таких преобразователей от 4 до 20 мА. Датчиками температуры являются платиновые термопреобразователи сопротивления (100 Ом, класс допуска А). Прием сигналов от «Сапфир-22М» и термопреобразователей сопротивления, а также обеспечение гальванической развязки входного сигнала осуществляется при помощи модулей фирмы Analog Devices.

Значения уставок хранятся в энергонезависимой памяти микроконтроллера и могут быть изменены с помощью встроенных дисплея и клавиатуры блока БПМ-79Р.

Блок БПМ-79Р также формирует и передает по RS-485 информацию для последующего отображения контролируемых параметров на оперативных дисплеях пульта оператора.

Каналы логической обработки сигналов и формирования управляющих сигналов для аварийной защиты и технологической автоматики. Каналы данного типа образованы аппаратурой логической обработки сигналов (АЛОС), главным элементом которой является блок обработки сигналов (БЛОС — БФМ-11Р), и оперативными дисплеями пульта оператора. Структурная схема канала АЛОС представлена на рис. 8.19.

Каналы АЛОС обеспечивают управление защитой в аварийном режиме при отклонении параметров реактора от установленных значений и отказе оборудования обеспечивающих систем, а также в нормальном режиме эксплуатации по сигналам с пульта оператора.

Алгоритмы управления реакторной установкой разработаны Научно-исследовательским и конструкторским институтом энерготехники (НИКИЭТ). В соответствии с этими алгоритмами АЛОС осуществляет формирование обобщенного управляющего сигнала аварийной защиты от четырех каналов АКНП по логике 2 из 4, от трех каналов АКТП по логике 2 из 3 и от сигналов оборудования обеспечивающих систем: при аварии дизель-генератора, при неисправности агрегатов бесперебойного питания, при

неисправности щита постоянного тока, при наличии сигнала аварийной защиты от источников холодных нейтронов, при закрытии отсечных задвижек системы аварийного охлаждения реактора центрального экспериментального канала, при сигнале аварии от системы индустриальной антисейсмической защиты, при неисправности системы жидкостного регулятора.

Сигналы о состоянии органов управления реакторной установкой, расположенных на пульте оператора, поступают в каждый канал АЛОС по независимым линиям связи с гальванической развязкой, и только после логической обработки этих сигналов осуществляется ручное и (или) автоматическое управление реакторной установкой.

Сигнал «Обобщенный АЗ» и сигналы управления рабочими органами реакторной установки поступают от каждого устройства УНО в блоки управления исполнительными механизмами (БУИМ, эти блоки для реактора ПИК называются БУШД — блоки управления шаговыми двигателями) по независимым линиям связи, где также происходит мажорирование этих сигналов по

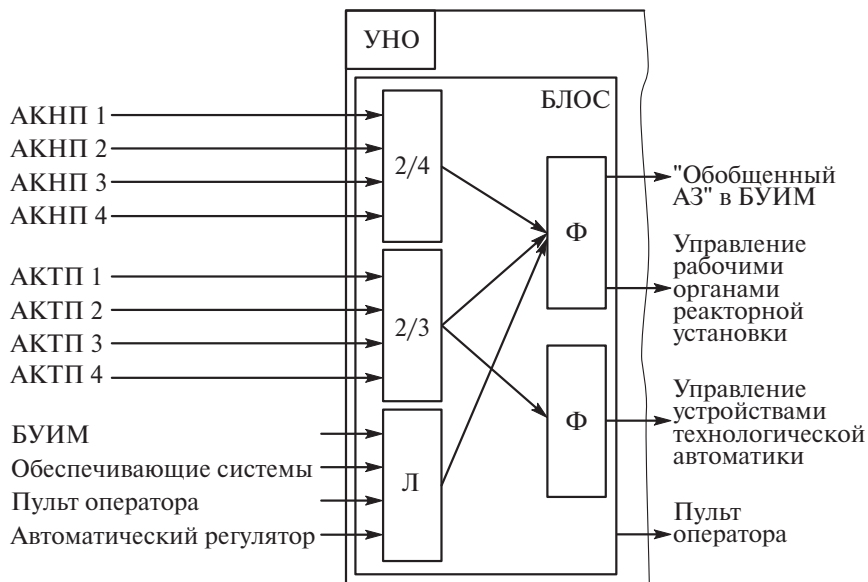


Рис. 8.19. Структурная схема канала АЛОС:

БЛОС — блок логической обработки сигналов; 2/4 — обработка сигналов от АКНП по мажоритарной логике 2 из 4; 2/3 — обработка сигналов от АКТП по мажоритарной логике 2 из 3; Л — логическая обработка сигналов; Ф — формователи сигналов для БУИМ и устройств технологической автоматики

логике 2 из 3. Каждый канал АЛОС обеспечивает мажорирование по логике 2 из 3 значений превышения уставок для АВТ по теплогидравлическим параметрам, которые поступают из каналов АКТП, формирование и последующую передачу дискретных сигналов для устройств технологической автоматики. Исправность выходных линий этих сигналов контролируется на дисплеях пульта оператора.

Минимальное время формирования сигнала «Обобщенный АЗ» по каналам АКНП — 20 мс, по каналам АКТП — 50 мс.

Каждый канал АЛОС выполняет функцию индивидуального определения времени падения рабочих органов с момента формирования сигнала «Обобщенный АЗ» до момента срабатывания нижнего концевого выключателя.

Блок обработки сигналов БФМ-11Р выполнен с применением модуля микроконтроллера, осуществляющего функции контроля и передачи информации, и модулей на логических матрицах, выполняющих функцию формирования обобщенных управляющих сигналов для аварийной защиты и устройств технологической автоматики. Обмен информацией между модулями производится по общей шине.

Каналы контроля реактивности и регулирования мощности реактора. Функции контроля реактивности (АКР) и автоматического регулятора мощности (АРМщ) реактора выполняет блок БНО-70Р, расположенный в устройствах УНО-200Р1 и УНО-200Р2. Причем блок БПМ-70Р в устройствах УНО-200Р1 выполняет одну функцию — контроль реактивности, а функция регулирования мощности находится в «горячем» резерве; блок БНО-70Р, расположенный в устройстве УНО-200Р2, реализует две функции — контроль реактивности и регулирование мощности. Блок БНО-70Р выполнен с применением микропроцессора и модулей фирмы Octagon Systems, а также модулей на дискретных элементах, разработанных в «ССА».

Для контроля реактивности и автоматического регулирования мощности реактора используется информация от тех же устройств детектирования, которые предназначены для формирования сигналов аварийной защиты и контроля АКНП. Эта информация ретранслируется блоком БПМ-80Р в виде частотного сигнала. Для вычисления реактивности используется точечная модель ядерного реактора с учетом шести групп запаздывающих нейтронов. Вычисленное значение реактивности передается на оперативные дисплеи пульта оператора.

Регулятор мощности обеспечивает формирование сигналов управления органами автоматического регулирования реактор-

ной установки с целью подъема мощности до заданного уровня и автоматического поддержания мощности на заданном уровне в диапазоне контроля от 1×10^{-5} до $1 \times 10^2 \% N_{\text{ном}}$. Автоматическое регулирование мощности осуществляется как по выбранному (одному из четырех), так и по среднему для четырех каналов АК-НП значению мощности. АРМщ обеспечивает автоматическое увеличение мощности с заданным периодом 35 или 70 с, начиная с мощности не менее $1 \times 10^{-5} \% N_{\text{ном}}$ до заданного уровня и стабилизацию уровня мощности в диапазоне $1 \times 10^{-2} \dots 1 \times 10^2 \% N_{\text{ном}}$.

Органов автоматического регулирования два: нижняя и верхняя шторы, перекрывающие центр активной зоны.

АРМщ обеспечивает формирование сигнала, пропорционального относительному отклонению мощности реактора от заданного значения, сравнение этого сигнала с обратным периодом реактора и преобразование полученного значения в дискретные сигналы положительного и отрицательного разбаланса. Сигналы положительного и отрицательного разбаланса передаются в БУ-ИМ. Информация о величине разбаланса и положении органов автоматического регулирования представлена на оперативном дисплее пульта оператора.

Выбор канала, по которому производится регулирование мощности, и переход на регулирование по среднему значению осуществляются с помощью переключателя, расположенного на пульте оператора. Установка заданных значений периода и мощности автоматического регулятора осуществляется при помощи блока задания уставки АРМщ — БКЦ-75Р, расположенного также на пульте оператора. Обмен информацией между АРМщ и блоком БКЦ-75Р выполняется по последовательному интерфейсу RS-485.

Аппаратура контроля положения стержней и передачи информации. Аппаратура контроля положения стержней расположена в УНО-200Р2 (блок БВЦ-228Р1) и предназначена для преобразования информации от датчиков положения рабочих органов ПТП-22 с последующим представлением на пульте оператора.

Блоки БВЦ-228Р и БВЦ-228Р1 осуществляют прием информации по интерфейсу RS-485, предварительную ее архивацию в течение 300 мс и форматирование для передачи по интерфейсу RS-485 на блоки связи БВЦ-225Р с целью последующего отображения на дисплеях пульта оператора БЩУ и оперативном дисплее РЩУ, а также поддерживают связь по интерфейсу Ethernet с аппаратурой диагностики и протоколирования. Блок БВЦ-228Р выполнен с применением микропроцессора и модулей фирмы Octagon Systems, преобразователей фирмы Analog Devices, модулей на дискретных элементах, разработанных в «ССА».

Аппаратура отображения, регистрации, сигнализации и управления. Аппаратура контроля реакторной установки и состояния исправности комплекса АСУЗ-03Р представлена блоками, расположенными на БЩУ и пульте оператора.

В состав БЩУ входят четыре (по количеству каналов АКНП) цифровых дисплея БИЦ-63Р, отображающих текущие значения мощности и периода. Цифровой дисплей принимает информацию по интерфейсу RS-485 непосредственно от БЗТП. Информация из каналов АКНП выводится на два самопишущих регистратора РП-160 с возможностью выбора контролируемого канала.

На БЩУ и в реакторном зале установлен оптико-звуковой сигнализатор — блок БСР-38Р, предназначенный для звуковой и световой сигнализации в режимах начального пуска и перегрузки топлива.

Оперативная информация на пульте оператора представлена на трех независимых дисплеях, выполненных с применением электролюминесцентных матриц EL640.480-AA1 фирмы Planar. Каждый дисплей функционирует под управлением блока связи БВЦ-225Р и клавиатуры РСМ-301. Блок связи БВЦ-225Р, выполненный на основе модулей фирмы Octagon Systems, осуществляет вывод информации на дисплей и приём сигналов с клавиатуры РСМ-301 для управления отображением. Информация на каждый блок связи БВЦ-225Р поступает от трех устройств УНО-200Р1 и устройства УНО-200Р2 по независимым интерфейсам RS-485.

По функциям представления информации оператору три дисплея идентичны, переключение слайдов осуществляется при помощи клавиатуры РСМ-301. На каждом оперативном дисплее отображаются три основных слайда («АКНП», «АКТП», «Сигнализация») и два неоперативных («Время падения рабочих органов» и «Исправность выходов в систему технологической автоматики»).

На пульте оператора размещены четыре блока задания уставок АКНП, блок задания уставок АРМщ и ключи управления реакторной установкой.

Аппаратура диагностики и протоколирования. Аппаратура диагностики и протоколирования (АДП) представлена рабочей станцией AWS-825Р фирмы Advantech, выполненной для промышленных применений, и принтером LaserJet 6L. Рабочая станция принимает информацию от устройств УНО-200Р1, УНО200Р2 и УНО-200Р3 по интерфейсу Ethernet. АДП связана с системой верхнего уровня РАКУРС по второму каналу Ethernet, что обеспечивает гальваническую развязку комплекса АСУЗ-03Р и систе-

мы РАКУРС. От этой системы через АДП поступают сигналы единого времени.

АДП предназначена для архивации, отображения и вывода на принтер значений параметров нейтронного потока, теплогидравлических параметров, состояния (положения) органов управления реакторной установкой на пульте оператора. АДП выполняет функцию регистрации первопричины возникновения аварийной ситуации. Максимальное время архивирования — 24 часа с дискретностью 100 мс.

При помощи АДП осуществляется автоматическая предпусковая проверка комплекса АСУЗ — 03Р по функции прохождения обобщенных управляющих сигналов аварийной защиты и предупредительной сигнализации и проверка срабатывания дискретных выходов к устройствам технологической автоматики. Результаты проверок распечатываются на принтере.

Аппаратура резервного шита управления. Каналы АКНП для РЩУ идентичны уже описанным каналам защиты и выполнены с применением аналогичных блоков и устройств. Отличие заключается только в том, что сигналы АЗ-*N*, ПС-*N*, АЗ-*T*, ПС-*T* не принимают участия в формировании управляющего сигнала аварийной защиты.

Аппаратура РЩУ состоит из двух независимых каналов контроля по параметрам нейтронного потока, аппаратуры отображения и регистрации. Каждый канал содержит устройство детектирования УДПН-20Р и устройства контроля УНО-200РЗ (блок БПМ-80Р). Аппаратура РЩУ расположена в отдельном здании. На оперативный дисплей РЩУ выводится текущая информация по двум каналам контроля параметров нейтронного потока и вся текущая информация о состоянии СУЗ, аналогичная информации на дисплеях пульта оператора. На самопишущем регистраторе РП-160 фиксируются значения мощности, усредненной по двум каналам АКНП РЩУ.

Заключение. Комплекс АСУЗ-03Р выполнен в соответствии с требованиями МАГАТЭ, Правил ядерной безопасности исследовательских реакторов и ГОСТ Российской Федерации; разработка, изготовление и испытания проводились под надзором ГАН РФ (ГосАтомНадзор России).

Комплекс АСУЗ-03Р создан с применением передовых технологий в области разработки, процессов конструирования и изготовления. Это привело к сокращению сроков изготовления и поставки оборудования на объект. Применение микропроцессорной техники позволило построить комплекс, адаптируемый к различным проектам.

Программное обеспечение для аппаратуры АКНП, АКТП, АЛОС является вновь созданным продуктом класса 2, ранее не применявшимся, разработанным для определенной конфигурации технических средств. Отсутствие в этом программном обеспечении базовых и операционных систем делает его «прозрачным» для процесса верификации вплоть до уровня кода программ. Процесс разработки и верификации осуществлялся согласно МЭК 60880.

Аппаратура комплекса АСУЗ-03Р в рамках Государственных приемо-сдаточных испытаний прошла все виды испытаний, регламентируемых техническим заданием: функциональные, климатические, механические, электромагнитные — и готовится к отправке на объект. В настоящее время на основе опыта создания комплекса АСУЗ-03Р, функциональной и конструктивной формализации интегрированного канала разрабатываются аналогичные системы управления и защиты для исследовательских реакторов ряда атомных научных центров. «ССА» является сторонником консервативной политики в области систем безопасности атомных реакторов, предусматривающей применение и развитие технических средств на «жесткой» логике.

Тем не менее считается необходимым и важным накопить опыт эксплуатации аппаратно-программного комплекса АСУЗ-03Р на исследовательских реакторах для подготовки объективных предложений по использованию принципиально новых технических решений в аналогичных системах для АЭС.

8.3. Системы автоматизированного проектирования

Современные задачи, возникающие перед наукой и техникой, вызывают необходимость проектирования все более сложных технических объектов в сжатые сроки. Удовлетворить противоречивые требования повышения сложности объектов, сокращения сроков и повышения качества проектирования с помощью простого увеличения численности проектировщиков нельзя, так как возможность параллельного проведения проектных работ ограничена и численность инженерно-технических работников в проектных организациях страны не может быть сколько-нибудь заметно увеличена. Выходом из этого положения является широкое применение вычислительной техники для решения проектных задач (автоматизация проектирования).

Цель автоматизации проектирования — обеспечить бездефектное проектирование, снизить материальные затраты, сократить

сроки проектирования и ликвидировать рост количества инженерно-технических работников, занятых проектированием.

Знание математического аппарата, применяемого в инженерных исследованиях, умение пользоваться математическими моделями при оптимальном проектировании реальных объектов и систем, знание программных и технических средств САПР и умение пользоваться ими в качестве инструмента проектировщика должны позволить современным инженерам ставить и решать задачи автоматизации проектирования по отраслям техники.

8.3.1. Общие сведения о проектировании

Процесс создания описания нового объекта может выполняться различными способами. Если весь процесс проектирования осуществляет человек, то проектирование называют *неавтоматизированным*. В настоящее время неавтоматизированное проектирование сложных объектов практически не применяют. Наибольшее распространение получило проектирование, при котором происходит взаимодействие человека и ЭВМ. Такое проектирование называют автоматизированным. Автоматизированное проектирование, как правило, осуществляется в режиме диалога человека с ЭВМ на основе применения специальных языков общения с ЭВМ.

Проектирование, при котором все преобразования описаний объекта и алгоритма его функционирования осуществляются без участия человека, называют *автоматическим*.

Рассмотрим ряд понятий, которые используются при проектировании объектов.

Первичное описание, представленное в заданной форме, называется заданием на проектирование. В задании на проектирование системы, должны быть сведения о назначении системы и ее параметрах, способах функционирования, конструктивной реализации, изготовления и т. п.

Проектным решением называется промежуточное или конечное описание объекта проектирования, необходимое и достаточное для рассмотрения и определения дальнейшего направления или окончания проектирования.

Проектное решение или их совокупность, удовлетворяющие заданным требованиям, необходимые для создания объекта проектирования, будут являться *результатом проектирования*. В заданные требования должны быть обязательно включены требования к форме представляемого проектного решения.

Документ, выполненный по заданной форме, в котором представлено какое-либо проектное решение, полученное при проектировании, называется *проектным*. Совокупность проектных документов в соответствии с установленным перечнем, в котором представлен результат проектирования, называется *проектом*.

Под *проектной процедурой* понимают формализованную совокупность действий, выполнение которых оканчивается проектным решением. Например, проектными процедурами являются оптимизация, контроль, поиск решения, корректировка, компоновка, проверка правильности трассировки и т. п.

Действие или формализованная совокупность действий, составляющих часть проектной процедуры, алгоритм которых остается неизменным для ряда проектных процедур называется *проектной операцией*. Примерами проектных операций являются составление таблиц с данными вычисления, вычерчивание топологии, ввод и вывод данных и т. п. Соответственно проектная процедура, алгоритм которой остается неизменным для различных объектов проектирования или различных стадий проектирования одного и того же объекта, называется *унифицированной проектной процедурой*.

Выполнение проектных работ можно распределить как во времени, так и по подразделениям проектной организации.

При временном распределении работ по созданию новых объектов процесс проектирования разделяется на *стадии и этапы*. Различают 8 стадий: предпроектные исследования, техническое задание, техническое предложение, эскизный проект, технический проект, рабочий проект, изготовление, отладка, испытание и ввод в действие.

При создании новых объектов выделяют следующие этапы:

- этап научно-исследовательских работ (НИР). Объединяет стадии: предпроектные исследования, техническое задание и часть технического предложения. Здесь проводят исследования по поиску новых принципов функционирования, новых структур, физических процессов, новой элементной базы, технических средств и т. п.;
- этап опытно-конструкторских работ (ОКР). Объединяет стадии: часть технического предложения, эскизный проект, технический проект. Здесь отражаются вопросы детальной конструкторской проработки проекта;
- этап рабочего проектирования. Объединяет стадии: рабочий проект, изготовление, отладка и испытание, ввод в действие. Здесь прорабатывают схемные, конструкторские и технологические решения, проводят испытания, изготовление.

Для этапа НИР в основном используют системы автоматизации научных исследований и экспериментов.

Распределение работ между подразделениями производят с использованием блочно-иерархического подхода (БИП) к проектированию. Этот подход основан на структурировании описаний объекта с разделением описаний на ряд иерархических уровней по степени детальности отображения в них свойств объекта и его частей. Каждому иерархическому уровню присущи свои формы документации, математический аппарат для построения моделей и алгоритмов исследования. Совокупность языков, моделей, постановок задач, методов получения описаний некоторого иерархического уровня часто называют уровнем проектирования.

Уровни проектирования можно выделять не только по степени подробности отражения свойств объекта, но и по характеру отражаемых свойств. Если в первом случае уровни называют горизонтальными, или иерархическими, то во втором — вертикальными, или аспектами.

Методология БИП базируется на трех концепциях: *разбиение и локальная оптимизация; абстрагирование; повторяемость.*

Концепция разбиения позволяет сложную задачу проектирования объекта свести к решению более простых задач с учетом взаимодействий между ними. Локальная оптимизация подразумевает улучшение параметров внутри каждой простой задачи. Абстрагируемость заключается в построении формальных математических моделей, отражающих только значимые в данных условиях свойства объектов. Повторяемость заключается в использовании существующего опыта проектирования.

Основное достоинство БИП — это упрощение процесса проектирования и получение возможности решать задачи проектирования доступными средствами.

Использование БИП помогает: упростить решение проблемы хранения данных, сократить размерность выполняемых программ и время проектирования, применять САПР один раз для объекта (его части) независимо от числа идентичных объектов (его частей).

Весь процесс проектирования можно представить как последовательность этапов, связывающих концептуальное описание объекта и создание этого объекта. Указанную связь реализуют в одном из двух направлений: восходящем или нисходящем. Восходящее проектирование (ВП), т. е. проектирование снизу вверх, характеризуется решением сначала задач низких иерархических уровней с последовательным переходом к решению задач более высоких уровней. Нисходящее проектирование (НП), т. е. проек-

тирование сверху вниз, является противоположным по отношению к ВП.

Отметим, например, что используемая в настоящее время концепция проектирования интегральных микросхем с большой степенью интеграции по модульному принципу — это концепция БИП. В системе БИП конструктор выполняет функциональные, интуитивные и интеллектуальные преобразования на верхних уровнях, а ЭВМ выполняет проектирование на нижних уровнях.

При выделении горизонтальных уровней проектирования производится разделение объекта на блоки и рассмотрение вместо объекта его отдельных блоков. Если на некотором уровне i_1 имеем объект s , то на соседнем, более низком уровне i_2 происходит разделение s на блоки s_1, s_2, \dots, s_j и рассмотрение каждого блока s_1, s_2, \dots, s_j на уровне i_2 с большей степенью детализации, чем на уровне i_1 .

В общем случае при проектировании технических объектов можно выделить несколько вертикальных уровней, основные из них — функциональный, конструкторский, технологический. Описание каждого вертикального уровня в свою очередь делят на иерархические уровни. Ниже приведен пример структурирования описания ЭВМ (табл. 8.3).

Таблица 8.3

Вертикальные уровни

Горизонтальные уровни	Функциональный	Алгоритмический	Конструкторский	Технологический
	Системный	Программирование системы	Шкаф, стойка	Принципиальная схема технологического процесса
	Логический	Программирование модулей	Панель	
	Схемотехнический		ТЭЗ	Маршрутная технология
	Компонентный	Проектирование микропрограмм	Модуль	Технологические операции
Кристалл				
Ячейка				

Функциональное проектирование включает в себя анализ технического задания (ТЗ) и на его основе выбор с системных позиций методики построения и путей реализации вычислительного процесса в объекте; связано с анализом и синтезом блоков объ-

екта; заключается в разработке функциональных и принципиальных схем. Здесь определяют принципы функционирования и важнейшие параметры и характеристики объекта.

Основные задачи функционального проектирования следующие: разработка структурных схем, определение требований к выходным параметрам; анализ и формирование ТЗ на разработку отдельных блоков объекта; синтез функциональных и принципиальных схем полученных блоков; контроль и выработка диагностических тестов; проверка работоспособности синтезируемых блоков; расчеты параметров пассивных компонентов и определение требований к параметрам активных компонентов; формулировка ТЗ на проектирование компонентов; выбор физической структуры, топологии компонентов; расчеты параметров диффузионных профилей и полупроводниковых компонентов, электрических параметров, параметров технологических процессов эпитаксии, диффузии, окисления и др.; вероятностные требования к выходным параметрам компонентов.

Алгоритмическое проектирование заключается в разработке алгоритмов функционирования и создании математического обеспечения объекта.

Конструкторское проектирование заключается в реализации принципиальных схем в заданном конструктивном базисе. При этом решаются вопросы выбора форм и материалов, выбора типоразмеров, компоновки, размещения элементов, трассировки соединений, контроля. Основные задачи конструкторского проектирования следующие: покрытие функциональных схем, т. е. получение принципиальных электрических схем; конструкторский расчет геометрических размеров компонентов и площади размещения; компоновка элементов; размещение элементов с учетом конструкторских схмотехнических и технологических ограничений; трассировка соединений; контроль топологии; проектирование фотошаблонов; выпуск конструкторско-технологической документации.

Технологическое проектирование заключается в решении задач технологической подготовки производства — разработке принципиальной схемы, маршрутов, операций и переходов технологических процессов изготовления деталей, сборки и монтажа узлов, включая выбор оснастки, инструмента, технологического оборудования и т. п.

Функциональное проектирование объекта состоит из четырех основных горизонтальных уровней: системного, логического, схмотехнического, компонентного.

На системном уровне определяют общую структурную схему, структурные схемы основных блоков.

На логическом уровне создают функциональные и принципиальные схемы объекта. Здесь выделяют подуровни — регистровый и вентильный. На регистровом подуровне проектируются устройства из модулей (функциональных узлов) типа регистров, счетчиков, сумматоров, интеграторов и т. п. На вентильном уровне проектируются устройства и модули из отдельных логических вентилей и триггеров.

Алгоритмическое проектирование используется для разработки программного обеспечения объекта. Для больших программных систем обычно используют набор иерархических уровней, два из которых являются основными. На первом планируют всю программную систему и разрабатывают схемы алгоритмов на основе программных модулей. На втором производят программирование модулей на заданном алгоритмическом языке.

Конструкторское проектирование состоит из иерархических уровней проектирования компонентов, БИС, типовых элементов замены, панелей, стоек, шкафов. Здесь в основном используется восходящее проектирование.

Технологическое проектирование состоит из уровней проектирования принципиальной схемы технологического процесса, технологических маршрутов, технологических операций.

8.3.2. Классификация САПР

В общем смысле классификация — система соподчиненных понятий, часто представляемая в виде различных схем, таблиц и используемая как средство для установления связей между этими понятиями или классами объектов, а также для точной ориентировки в многообразии понятий или соответствующих объектов. Классификация фиксирует место объекта в системе, которая указывает на его свойства. В связи с этим она служит средством хранения и поиска информации, содержащейся в ней самой. Классификация создает условия для разработки технически обоснованных норм обеспечения процесса создания, функционирования и стандартизации в области САПР.

Системы автоматизированного проектирования классифицируются по типу, разновидности и сложности объекта проектирования; уровню и комплектности автоматизации проектирования; характеру и числу выпускаемых проектных документов; числу уровней в структуре технического обеспечения (рис. 8.20). На этом

классификация первого уровня закончена. Рассмотрим классификацию второго иерархического уровня.

По типу объекта проектирования различают САПР (рис. 8.21): изделий машиностроения и приборостроения; технологических процессов в машиностроении и приборостроении; объектов строительства; организационных систем.

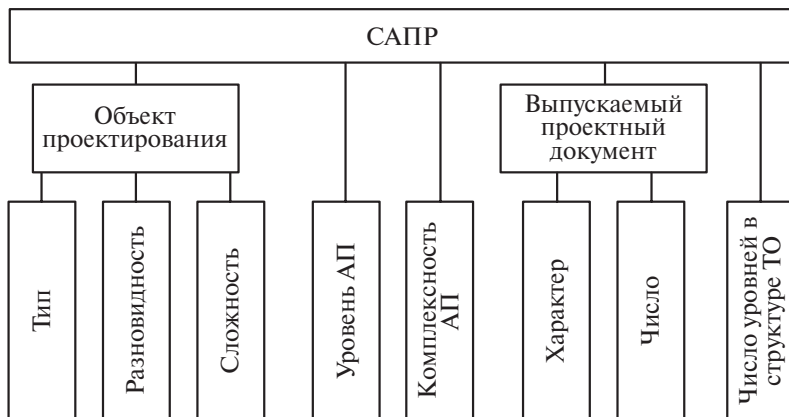


Рис. 8.20. Классификация САПР



Рис 8.21. Классификация САПР по типу (а) и сложности (б) объекта проектирования

Под *организационной системой* понимают совокупность, состоящую из коллектива специалистов и комплекса средств, взаимодействий и взаимосвязей между ними и с внешней средой и алгоритма процесса проектирования, необходимого для выполнения АП.

По разновидности объекта проектирования можно различать САПР РЭА, САПР атомной станции, САПР ракетной системы и т. п.

По сложности объекта проектирования различают САПР (рис. 8.21, б): простых объектов, содержащих до 10^2 составных частей; объектов средней сложности, содержащих от 10^2 до 10^3 составных частей; сложных объектов, содержащих от 10^3 до 10^4 составных частей; очень сложных объектов, содержащих от 10^4 до 10^6 составных частей; объектов очень высокой сложности, содержащих 10^6 и более составных частей.

Следовательно, САПР современных ЭВМ относится к САПР простых объектов, а САПР многопроцессорных вычислительных систем на сверхбольших интегральных микросхемах — к САПР очень сложных объектов.

По уровню автоматизации проектирования различают САПР (рис. 8.22, а) низкоавтоматизированные, в которых число автоматизированных проектных процедур (АПП) составляет 25 % общего числа проектных процедур; среднеавтоматизированные, — от 25 до 50% общего числа проектных процедур, высокоавтоматизированные от 50 до 75 %. В этих системах применяют методы

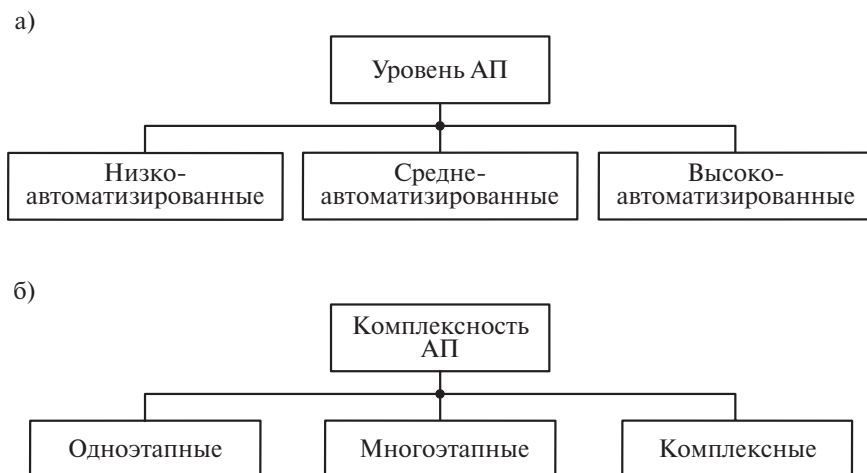


Рис. 8.22. Классификация САПР по уровню (а) и комплексности (б) автоматизации проектирования

многовариантного оптимального проектирования. Например, в гибких автоматизированных производствах (ГАП) для эффективности результатов необходимо использовать САПР средне- и высокоавтоматизированного проектирования.

По комплексности САПР классифицируют так (рис.8.22, б): одноэтапные, выполняющие один этап проектирования из всех установленных для объекта; многоэтапные, выполняющие несколько этапов проектирования из всех установленных для объекта; комплексные, выполняющие все этапы проектирования, установленные для объекта.

По характеру выпускаемых проектных документов различают САПР (рис. 8.23, а): текстовые, выполняющие только текстовые, документы на бумажной ленте или листе; текстовые и графические, выполняющие текстовые и графические документы на бумажной ленте или листе; на машинных носителях, выполняющих документы на магнитных носителях (магнитных лентах, дисках и барабанах); на фотоносителях, выполняющих документы на микрофильмах, микрофишах, фотошаблонах и т. п.; на двух типах носителей; на всех типах носителей.

По количеству выпускаемых проектных документов различают САПР (рис. 8.23, б): малой производительности, выпускающих до 10^5 проектных документов в пересчете на формат А за год;



Рис. 8.23. Классификация САПР по характеру (а) и числу (б) выпускаемых проектных документов

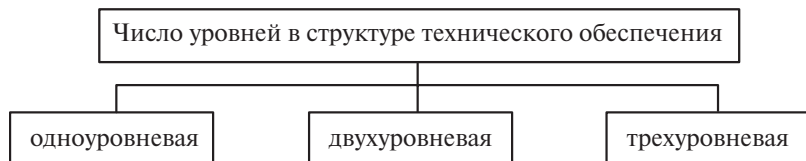


Рис. 8.24. Классификация САПР по числу уровней в структуре технического обеспечения

средней производительности, выпускающих от 10^5 до 10^6 проектных документов; высокой производительности выпускающих свыше 10^6 проектных документов.

В различных отраслях вводятся различные количественные характеристики информации, определяющие производительность САПР.

По числу уровней в структуре технического обеспечения различают САПР (рис. 8.24): одноуровневые, построенные на основе ЭВМ среднего или высокого класса со штатным набором периферийных устройств, который может быть дополнен средствами обработки графической информации; двухуровневые, построенные на основе ЭВМ среднего или высокого класса и одного или нескольких АРМ, включающих мини-ЭВМ; трехуровневые, построенные на основе ЭВМ высокого класса, АРМ и периферийного программно-управляемого оборудования.

Данная классификация — не догма. Теория автоматизации проектирования непрерывно развивается. Появляются новые технические и программные средства ЭВМ, комплексные САПР, поэтому существующие схемы классификации САПР будут видоизменяться и совершенствоваться.

8.3.3. Системный подход к автоматизации проектирования и принципы организации САПР

Рассмотрим основные принципы организации САПР. Целью создания САПР является повышение качества и технико-экономического уровня проектируемых объектов при их создании и применении, повышение производительности труда, сокращение сроков, уменьшение стоимости и трудоемкости проектирования.

Очевидно, что создание САПР в первую очередь должно быть обусловлено технико-экономической целесообразностью.

Состав типовой САПР. Составными структурными частями САПР являются подсистемы, обладающие всеми свойствами систем и создаваемые как самостоятельные. *Подсистемой САПР*

называют выделенную по некоторым признакам часть САПР, обеспечивающую получение законченных проектных решений.

По назначению подсистемы САПР разделяют на проектирующие и обслуживающие. К проектирующим относят подсистемы, выполняющие проектные процедуры и операции, например подсистема логического проектирования, подсистема конструкторского проектирования, подсистема технологического проектирования, подсистема проектирования деталей и сборочных единиц и т. п. К обслуживающим относят подсистемы, предназначенные для поддержания работоспособности проектирующих подсистем, например подсистема информационного поиска, подсистема документирования, подсистема графического отображения объектов проектирования и т. п.

По отношению к объекту проектирования различают объектно-ориентированные (объектные) и объектно-независимые (инвариантные). К объектным относят подсистемы, выполняющие одну или несколько проектных процедур или операций, непосредственно зависящих от конкретного объекта проектирования. К инвариантным относят подсистемы, выполняющие унифицированные проектные процедуры и операции, например функции отработки, не зависящие от особенностей проектируемого объекта.

Подсистемы состоят из компонентов, объединенных общей для данной подсистемы целевой функцией и обеспечивающих функционирование этой подсистемы.

Структурное единство подсистемы обеспечивается связями между компонентами различных обеспечений САПР.

Принципы создания САПР. К ним относят принципы включения, системного единства, развития, комплексности, информационного единства, совместимости, стандартизации.

Принцип включения обеспечивает разработку САПР на основе требований, позволяющих включать эту САПР в САПР более высокого уровня.

Принцип системного единства состоит в том, что при создании, функционировании и развитии САПР связи между подсистемами должны обеспечивать целостность системы.

Согласно принципу развития САПР должна создаваться и функционировать с учетом пополнения, совершенствования и обновления подсистем и компонентов.

Принцип комплексности обеспечивает связность проектирования элементов и всего объекта на всех стадиях, позволяет осуществлять согласование и контроль характеристик элементов и объекта в целом.

Принцип информационного единства состоит в использовании в подсистемах, компонентах и средствах обеспечения САПР единых условных обозначений, терминов, символов, проблемно-ориентированных языков, способов представления информации, соответствующих принятым нормативным документам.

Согласно принципу совместимости языки, символы, коды, информационные и технические характеристики структурных связей между подсистемами, средствами обеспечения и компонентами должны обеспечивать совместное функционирование подсистем и сохранять открытую структуру системы в целом.

Принцип стандартизации заключается в проведении унификации, типизации и стандартизации подсистем и компонентов, инвариантных к проектируемым объектам и отраслевой специфике, а также в установлении правил с целью упорядочения деятельности в области создания и развития САПР.

Подсистемы должны вводиться в действие и функционировать независимо от других подсистем. Единство общесистемных требований обеспечивает проектная служба САПР.

8.3.4. Виды обеспечения САПР

Система автоматизированного проектирования — это организационно-техническая система, состоящая из комплекса средств автоматизации проектирования (КСАП), взаимосвязанного с необходимыми подразделениями проектной организации P_1, P_2, \dots, P_n или коллективом специалистов (пользователей системы) и выполняющая автоматизированное проектирование (рис. 8.25, а). Соответственно *система автоматического проектирования* выполняет автоматическое проектирование без участия человека. Автоматизированное проектирование осуществляется с помощью комплекса средств автоматизации проектирования, который представляет собой совокупность различных видов обеспечения автоматизированного (автоматического) проектирования (АП), необходимых для выполнения АП (рис. 8.25, б).

Математическое обеспечение (МО) АП — это совокупность математических методов (ММет), математических моделей (ММ) и алгоритмов проектирования (АлП), необходимых для выполнения АП, представленных в заданной форме (рис. 8.26, а).

Программное обеспечение (ПО) АП — совокупность машинных программ, необходимых для выполнения АП, представленных в заданной форме. Часть ПО АП, предназначенную для управления проектированием, называют операционной системой (ОС) АП.

Совокупность машинных программ (МП), необходимых для выполнения какой-либо проектной процедуры и представленных в заданной форме, называют пакетом прикладных программ (ППП).

Компонентами ПО являются документы с текстами программ, программы на всех видах носителей, эксплуатационные документы. Программное обеспечение разделяют на общесистемное (ОПО) и прикладное (ППО). Компонентами ОПО являются трансляторы (Т) с алгоритмических языков, эмуляторы (Э), супервизоры (С) и др. Компонентами ППО являются программы (МП) и пакеты прикладных программ для АП (рис. 8.26, б).

Информационное обеспечение (ИО) АП — совокупность сведений, необходимых для выполнения АП, представленных в заданной форме. Основной частью ИО являются автоматизированные банки данных, которые состоят из баз данных (БД) САПР и систем управления базами данных (СУБД). В ИО входят нормативно-справочные документы, задания государственных планов, прогнозы технического развития, типовые проектные решения,

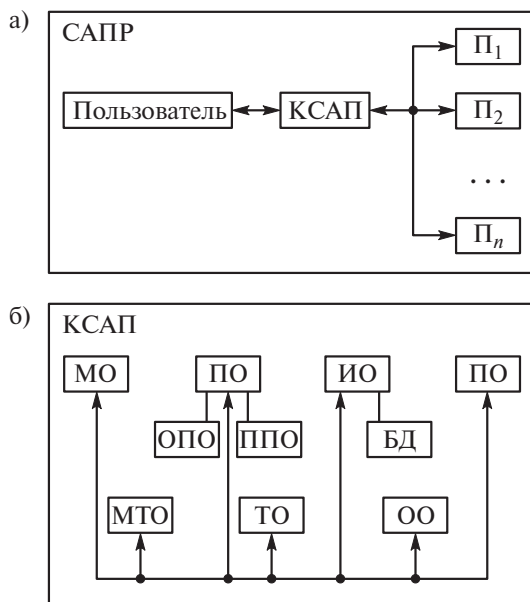


Рис. 8.25. Система автоматизированного (автоматического) проектирования (а) и комплекс средств автоматизированного (автоматического) проектирования (б); виды обеспечения:

МО – математическое, МТО – методическое, ПО – программное, ИО – информационное, ЛО – лингвистическое, ТО – техническое, ОО – организационное

системы классификации и кодирования технико-экономической информации, системы документации типа ЕСКД, ЕСТД, файлы и блоки данных на машинных носителях, фонды нормативные, плановые, прогнозные, типовых решений, алгоритмов и программ и т. п. (рис. 8.27).

Управление автоматизированным банком данных осуществляют проектировщики, при этом необходимо обеспечить целос-

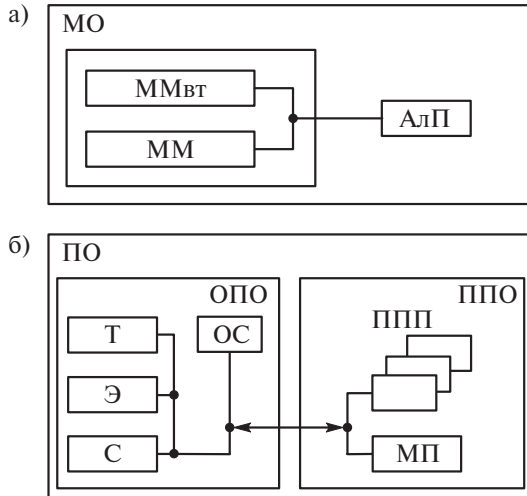


Рис. 8.26. Математическое (а) и программное (б) обеспечения САПР

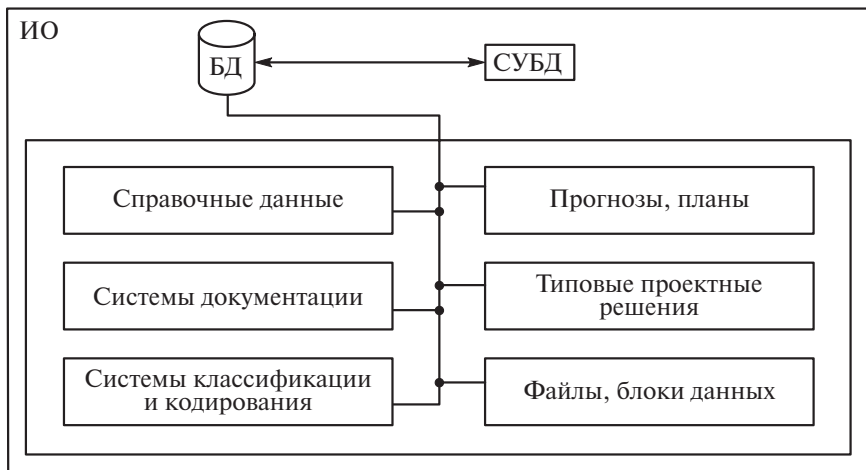


Рис. 8.27. Информационное обеспечение САПР

тность, правильность данных, эффективность и функциональные возможности СУБД. Проектировщик организует и формирует БД, определяет вопросы использования и реорганизации. База данных составляется с учетом характеристик объектов проектирования, процесса проектирования, действующих нормативов и справочных данных. При создании автоматизированных банков данных одним из основных является принцип информационного единства, заключающийся в использовании единой терминологии, условных обозначений, символов, единых проблемно-ориентированных языков, способов представления информации, единой размерности данных физических величин, хранящихся в БД. Автоматизированные банки данных должны обладать гибкостью, надежностью, наглядностью и экономичностью. Гибкость заключается в возможности адаптации, наращивания и изменения средств СУБД и структуры БД. Реорганизация БД не должна приводить к изменению прикладных программ. Для одновременного обслуживания пользователей должен быть организован параллельный доступ к данным. При использовании интерактивных методов проектирования необходимо использовать режим диалога.

Все версии СУБД должны генерироваться в соответствии с имеющимся комплексом ТС САПР. Доступ к информации БД должен обеспечиваться пользователями различных уровней.

Для надежности необходимо иметь возможность восстановления информации и ПО в случае разрушения, обеспечивать стандартные реакции на ошибочный запрос. Для наглядности информация должна представляться в обычной и удобной для пользователя форме. Для обеспечения экономичности необходимо исключить неоправданное дублирование данных, обеспечить автоматизацию сбора статистических данных о содержании и использовании информации банка для организации эффективного распределения памяти, обеспечить наличие средств для тиражирования баз данных.

Техническое обеспечение (ТО) АП — это совокупность взаимосвязанных и взаимодействующих технических средств, предназначенных для выполнения АП. Пример ТО АП показан на рис. 8.28. Технические средства (ТС) выполняют определенную функцию в САПР и представляют собой компоненты ТО. Вообще говоря, компонент САПР — это элемент средства обеспечения, выполняющий определенную функцию. К ТС относятся устройства вычислительной и организационной техники, средства передачи данных, измерительные и другие устройства.

Различают следующие группы ТС:

- *подготовка и ввод данных*. Группа предназначена для автоматизации подготовки и редактирования данных при вводе в ЭВМ алфавитно-цифровой и графической информации. Группа дает возможность кодирования информации, нанесения данных на машинные носители, ввода данных в ЭВМ, визуального контроля и редактирования данных при вводе информации;
- *передача данных*. Группа предназначена для обеспечения дистанционной связи технических средств по телефонным, телеграфным и специальным каналам связи;
- *программная обработка данных*. Группа включает в себя универсальные или специализированные ЭВМ, обеспечивающие прием цифровых данных с устройства ввода или каналов связи, их программной обработки, накопления и вывода на машинные носители, устройства отображения и каналы связи. Позволяет изменить производительность путем наращивания ЭВМ, использовать мультипрограммный и диалоговый режимы работы;
- *отображение и документирование данных*. Группа предназначена для оперативного представления и документиро-

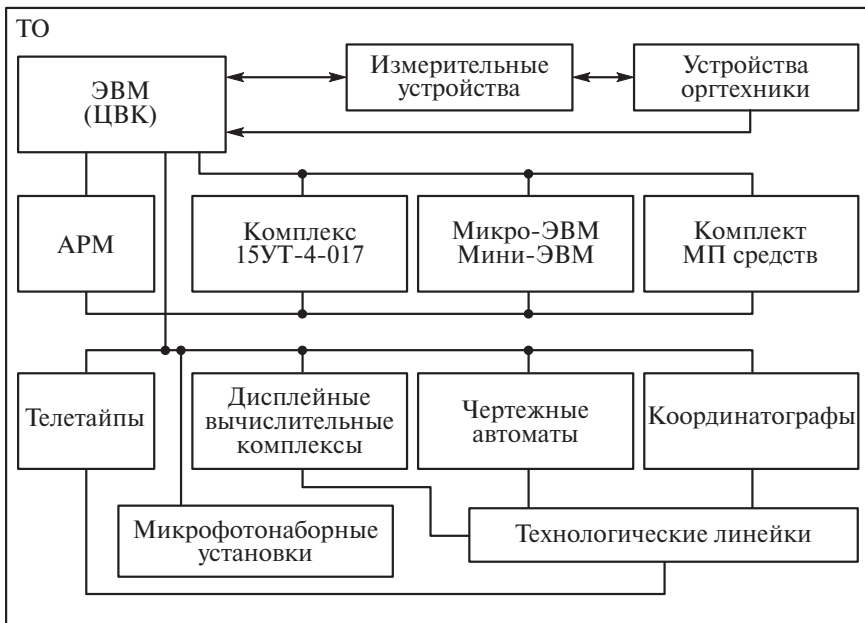


Рис 8.28. Техническое обеспечение САПР

вания проектных решений. Здесь используют печатающие устройства и графопостроители, микрофильмы, микрофиши и устройства отображения визуальной информации; — *архив проектных решений*. Группа предназначена для обеспечения хранения, контроля, восстановления и размножения данных о проектных решениях и справочных данных.

Компоненты ТО создаются на базе серийных средств вычислительной техники общего назначения и специализированных технических средств. В настоящее время преимущественно используют двухуровневую иерархическую структуру комплекса ТС САПР. Структура включает в себя компоненты центрального вычислительного комплекса (ЦВК) и компоненты терминального комплекса (ТК). Центральный ВК строят на основе ЭВМ, вычислительных систем и сетей ЭВМ. Терминальный комплекс САПР строят на основе автоматизированных рабочих мест, терминальных станций с использованием микропроцессоров, мини- и микроЭВМ.

Лингвистическое обеспечение (ЛО) АП — совокупность языков проектирования (ЯП), включая термины и определения, правила формализации естественного языка и методы сжатия и развертывания текстов, необходимых для выполнения АП, представленных в заданной форме (рис.8.29).

Методическое обеспечение (МТО) — совокупность документов, устанавливающих состав и правила отбора и эксплуатации средств обеспечения АП, необходимых для выполнения АП. Отметим, что в некоторых работах и документах методическое обеспечение понимается более широкой в качестве компонентов включает МО и ЛО.

Организационное обеспечение (ОО) АП — совокупность документов, устанавливающих состав проектной организации и ее подразделений, связи между ними, их функции, а также форму представления результата проектирования и порядок рассмотре-



Рис. 8.29. Лингвистическое обеспечение САПР

ния проектных документов, необходимых для выполнения АП. Компонентами ОО САПР являются методические и руководящие материалы, положения, инструкции, приказы и другие документы, обеспечивающие взаимодействие подразделений проектной организации при создании и эксплуатации САПР.

В настоящее время при создании ЭВМ пятого поколения с перестраиваемой архитектурой и коммутацией, способных принимать решения в условиях расплывчатости и неопределенности, перспективными являются САПР, позволяющие адаптироваться к внешним условиям проектирования и имеющие возможность настраиваться на заданный класс создаваемых объектов.

8.3.5. Математические модели проектируемых объектов в САПР

Классификация математических моделей. Математическая модель (ММ) технического объекта есть совокупность математических объектов (чисел, переменных, матриц, множеств и т. п.) и отношений между ними, которая адекватно отображает свойства технического объекта, интересующие инженера, разрабатывающего этот объект. Выполнение проектных операций и процедур в САПР основано на оперировании ММ. С их помощью прогнозируются характеристики и оцениваются возможности предложенных вариантов схем и конструкций, проверяется их соответствие предъявляемым требованиям, проводится оптимизация параметров, разрабатывается техническая документация и т. п.

В САПР для каждого иерархического уровня сформулированы основные положения математического моделирования, выбран и развит соответствующий математический аппарат, получены типовые ММ элементов проектируемых объектов, формализованы методы получения и анализа математических моделей систем. Сложность задач проектирования и противоречивость требований высокой точности, полноты и малой трудоемкости анализа обуславливают целесообразность компромиссного удовлетворения этих требований с помощью соответствующего выбора моделей. Это обстоятельство приводит к расширению множества используемых моделей и развитию алгоритмов адаптивного моделирования.

Функциональные и структурные модели. В проектных процедурах, связанных с функциональным аспектом проектирования, как правило, используются ММ, отражающие закономерности процессов функционирования объектов. Такие модели называют функциональными. Типичная функциональная модель представ-

ляет собой систему уравнений, описывающих либо электрические, тепловые, механические процессы, либо процессы преобразования информации.

В то же время в процедурах, относящихся к конструкторскому аспекту проектирования, преобладает использование математических моделей, отражающих только структурные свойства объекта, например его геометрическую форму, размеры, взаимное расположение элементов в пространстве. Такие модели называют *структурными*. Структурные модели чаще всего представляются в виде графов, матриц инцидентий и смежности, списков и т. п.

Как правило, функциональные модели более сложные, поскольку в них отражаются также сведения о структуре объектов. Однако при решении многих задач конструирования использование сложных функциональных моделей не оправдано, так как нужные результаты могут быть получены на основе более простых структурных моделей. Функциональные модели применяют преимущественно на завершающих этапах верификации описаний объектов, предварительно синтезированных с помощью структурных моделей.

Иерархия математических моделей в САПР. Блочный-иерархический подход к проектированию радиоэлектронной аппаратуры (РЭА) включает в качестве своей основы иерархию математических моделей. Деление моделей по иерархическим уровням (уровням абстрагирования) происходит по степени детализации описываемых свойств и процессов, протекающих в объекте. При этом на каждом иерархическом уровне используют свои понятия «система» и «элементы». Так, система k -того уровня рассматривается как элемент на соседнем более высоком $(k-1)$ -м уровне абстрагирования.

Представим структуру некоторого объекта множества элементов (рис. 8.30) и связей между ними. Выделим в соответствии с блочно-иерархическим подходом в структуре объекта некоторые подмножества элементов и назовем их блоками (на рисунке показаны штриховыми линиями). Пусть состояние каждой связи характеризуется одной фазовой переменной v_i , z_j или u_k . Здесь v_i относится к внутренним связям между элементами данного блока, z_j , u_k относятся к выходам и входам блока соответственно. Теперь поясним важные для функциональных моделей понятия полной модели и макромоделей.

Полная модель блока есть модель, составленная из моделей элементов с учетом межэлементных связей, т. е. модель, описывающая как состояние выходов, так и состояние каждого из эле-

ментов блока. Моделями элементов блока А являются уравнения, связывающие входные и выходные переменные:

$$\left. \begin{aligned} f_1(v_1, u_1) &= 0; \\ f_2(v_2, u_2) &= 0; \\ f_3(v_2, u_3, v_1) &= 0; \\ \dots\dots\dots \\ f_9(v_9, u_8, z_2) &= 0. \end{aligned} \right\} \quad (8.1)$$

Полная модель блока есть система уравнений

$$F(V, U) = 0, \quad Z = \psi(V, U), \quad (8.2)$$

где V, Z и U — векторы внутренних, выходных и входных фазовых переменных блока.

При большом количестве элементов размерность вектора V и порядок системы уравнений (8.2) становятся чрезмерно большими и требуют упрощения.

При переходе к более высокому иерархическому уровню упрощения основаны на исключении из модели вектора внутренних переменных V . Полученная модель представляет собой систему уравнений

$$\varphi(Z, U) = 0 \quad (8.3)$$

существенно меньшей размерности, чем полная модель (8.2), и называется *макромоделью*. Следовательно, макромодель уже не описывает процессы внутри блока, а характеризует только про-

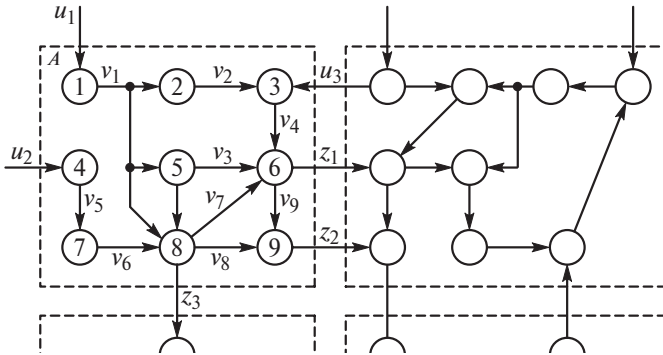


Рис. 8.30. Представление структуры объекта

цессы взаимодействия данного блока с другими в составе системы блоков.

Модели (8.2) и (8.3) относятся друг к другу как полная модель и макромоделю на n -м уровне иерархии. На более высоком ($n-1$)-м уровне блок А рассматривается как элемент и макромоделю (8.3) становится моделью элемента А. Следовательно, модели (8.1) и (8.3) относятся друг к другу как модели элементов соседних иерархических уровней. Из моделей типа (8.3) может быть составлена полная модель системы на ($n-1$)-м уровне.

Микро-, макро- и метауровни. В зависимости от сложности объекта при его проектировании используют большее или меньшее число уровней абстракции. Объединение уровней, родственных по характеру используемого математического аппарата, приводит к образованию трех укрупненных уровней — микро-, макро- и метауровня — в иерархии функциональных моделей для большинства проектируемых сложных объектов.

На *микроуровне* используют математические модели, описывающие физическое состояние и процессы в сплошных средах. Для моделирования применяют аппарат уравнений математической физики. Примерами таких уравнений служат дифференциальные уравнения в частных производных—уравнения электродинамики, теплопроводности, упругости, газовой динамики. Эти уравнения описывают поля электрического потенциала и температуры в полупроводниковых кристаллах интегральных схем, напряженно-деформированное состояние деталей механических конструкций и т. п. К типичным фазовым переменным на микроуровне относятся электрические потенциалы, давления, температуры, концентрации частиц, плотности токов, механические напряжения и деформации. Независимыми переменными являются время и пространственные координаты. В качестве операторов F и ψ в уравнениях (8.2) фигурируют дифференциальные и интегральные операторы. Уравнения (8.2), дополненные краевыми условиями, составляют ММ объектов на микроуровне. Анализ таких моделей сводится к решению краевых задач математической физики.

На *макроуровне* производится дискретизация пространств с выделением в качестве элементов отдельных деталей, дискретных электрорадиоэлементов, участков полупроводниковых кристаллов. При этом из числа независимых переменных исключают пространственные координаты. Функциональные модели на макроуровне представляют собой системы алгебраических или обыкновенных дифференциальных уравнений, для их получения и решения используют соответствующие численные методы.

В качестве фазовых переменных фигурируют электрические напряжения, токи, силы, скорости, температуры, расходы и т. д. Они характеризуют проявления внешних свойств элементов при их взаимодействии между собой и внешней средой в электронных схемах или механических конструкциях.

На *метауровне* с помощью дальнейшего абстрагирования от характера физических процессов удается получить приемлемое по сложности описание информационных процессов, протекающих в проектируемых объектах. На метауровне для моделирования аналоговой РЭА широко применяют аппарат анализа систем автоматического управления, а для моделирования цифровой РЭА — математическую логику, теорию конечных автоматов, теорию массового обслуживания. Математические модели на метауровне — системы обыкновенных дифференциальных уравнений, системы логических уравнений, имитационные модели систем массового обслуживания.

Формы представления моделей. Для представления моделей используют следующие основные формы:

- *инвариантная форма* — запись соотношений модели с помощью традиционного математического языка безотносительно к методу решения уравнений модели;
- *алгоритмическая форма* — запись соотношений модели и выбранного численного метода решения в форме алгоритма;
- *аналитическая форма* — запись модели в виде результата аналитического решения исходных уравнений модели; обычно модели в аналитической форме представляют собой явные выражения выходных параметров как функций внутренних и внешних параметров;
- *схемная форма*, называемая также графической формой, — представление модели на некотором графическом языке, например на языке графов, эквивалентных схем, диаграмм и т. п. Графические формы удобны для восприятия человеком. Использование таких форм возможно при наличии правил однозначного истолкования элементов чертежей и их перевода на язык инвариантных или алгоритмических форм.

Модели в алгоритмической и аналитической формах называют соответственно *алгоритмическими* и *аналитическими*. Среди алгоритмических моделей важный класс составляют *имитационные модели*, предназначенные для имитации физических или информационных процессов в объекте при задании различных зависимостей входных воздействий от времени. Собственно имитацию названных процессов называют имитационным моделированием. Результат имитационного моделирования — зависимости фазо-

вых переменных в избранных элементах системы от времени. Примерами имитационных моделей являются модели электронных схем в виде систем обыкновенных дифференциальных уравнений или модели систем массового обслуживания, предназначенные для имитации процессов прохождения заявок через систему.

Требования к математическим моделям. Основными требованиями, предъявляемыми к математическим моделям, являются адекватность, универсальность и экономичность.

Адекватность. Модель считается адекватной, если отражает заданные свойства объекта с приемлемой точностью. Точность определяется как степень совпадения значений выходных параметров модели и объекта. Пусть ε_j — относительная погрешность модели по j -тому выходному параметру:

$$\varepsilon_j = (\tilde{y}_j - y_j)/y_j,$$

где \tilde{y}_j — j -тый выходной параметр, рассчитанный с помощью модели; y_j — тот же выходной параметр, имеющий место в моделируемом объекте.

Погрешность модели ε_M по совокупности учитываемых выходных параметров оценивается одной из норм вектора $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_M)$, например

$$\varepsilon_M = \max_{j=1,m} |\varepsilon_j| \quad \text{или} \quad \varepsilon_M = \sqrt{\sum_{i=1}^m \varepsilon_j^2}.$$

Точность модели различна в разных условиях функционирования объекта. Эти условия характеризуются внешними параметрами. Если задаться предельной допустимой погрешностью $\varepsilon_{\text{пред}}$, то можно в пространстве внешних параметров выделить область, в которой выполняется условие

$$\varepsilon_M < \varepsilon_{\text{пред}}.$$

Эту область называют *областью адекватности* (ОА) модели. Возможно введение индивидуальных предельных значений $\varepsilon_{\text{пред } j}$ для каждого выходного параметра и определение ОА как области, в которой одновременно выполняются все m условий вида $|\varepsilon_j| \leq \varepsilon_{\text{пред } j}$.

Определение областей адекватности для конкретных моделей — сложная процедура, требующая больших вычислительных затрат. Эти затраты и трудности представления ОА быстро растут с увеличением размерности пространства внешних параметров. Определение ОА — более трудная задача, чем, например, задача

параметрической оптимизации, поэтому для моделей вновь проектируемых объектов ОА не рассчитывают.

Однако для моделей унифицированных элементов расчет областей адекватности становится оправданным в связи с однократностью определения ОА и многократностью их использования при проектировании различных систем. Знание ОА позволяет правильно выбирать модели элементов из числа имеющихся и тем самым повышать достоверность результатов машинных расчетов.

В общем случае ОА может иметь произвольную форму, сведения о которой выражаются громоздко, и неудобна в использовании, поэтому на практике вместо истинных ОА применяют те или иные их аппроксимации. На рис. 8.31 дано графическое представление области адекватности и аппроксимирующего ее гиперпараллелепипеда. Такое представление удобно для двумерных случаев.

Возможно использование и других аппроксимаций ОА, например областей с линеаризованными границами в виде участков гиперплоскостей, областей в форме гиперсфер и т. п.

Универсальность. При определении ОА необходимо выбрать совокупность внешних параметров и совокупность выходных параметров y_j , отражающих учитываемые в модели свойства. Типичными внешними параметрами при этом являются параметры нагрузки и внешних воздействий (электрических, механических, тепловых, радиационных и т. п.). Увеличение числа учитываемых внешних факторов расширяет применимость модели, но существенно удорожает работу по определению ОА. Выбор совокупности выходных параметров также неоднозначен, однако для большинства объектов число и перечень учитываемых свойств и соответствующих им выходных параметров сравнительно невелики, достаточно стабильны и составляют типовой набор выходных параметров. Например, для макромоделей логических элементов БИС такими выходными параметрами являются уровни выходного напряжения в состояниях логических «0» и «1», запасы помехоустойчивости, задержка распространения сигнала, рассеиваемая мощность.

Если адекватность характеризуется положением и размерами ОА, то универсальность модели определяется числом и составом учитываемых в модели внешних и выходных параметров.

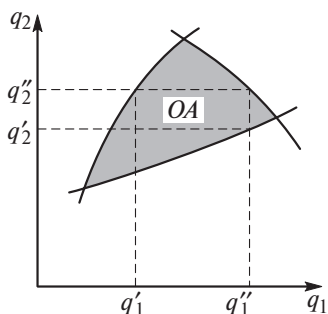


Рис. 8.31. Пример области адекватности

Экономичность. Экономичность модели характеризуется затратами вычислительных ресурсов для ее реализации, а именно затратами машинного времени T_M и памяти P_M . Общие затраты T_M и P_M на выполнение в САПР какой-либо проектной процедуры зависят как от особенностей выбранных моделей, так и от методов решения.

В большинстве случаев при реализации численного метода происходят многократные обращения к модели элемента, входящего в состав моделируемого объекта. Тогда удобно экономичность модели элемента характеризовать затратами машинного времени, получающимися при обращении к модели, а число обращений к модели должно учитываться при оценке экономичности метода решения.

Экономичность модели по затратам памяти оценивается объемом оперативной памяти, необходимой для реализации модели.

Требования широких областей адекватности, высокой степени универсальности, с одной стороны, и высокой экономичности, с другой, являются противоречивыми. Наилучшее компромиссное удовлетворение этих требований оказывается неодинаковым в различных применениях. Это обстоятельство обуславливает использование в САПР многих моделей для объектов одного и того же типа — различного рода макромоделей, многоуровневых, смешанных моделей и т. п.

Методы получения моделей элементов. Получение моделей элементов (моделирование элементов) в общем случае — процедура неформализованная. Основные решения, касающиеся выбора вида математических соотношений, характера используемых переменных и параметров, принимает проектировщик. В то же время такие операции, как расчет численных значений параметров модели, определение областей адекватности и другие, алгоритмизированы и решаются на ЭВМ. Поэтому моделирование элементов обычно выполняется специалистами конкретных технических областей с помощью традиционных средств экспериментальных исследований и средств САПР.

Методы получения функциональных моделей элементов делят на теоретические и экспериментальные. *Теоретические методы* основаны на изучении физических закономерностей протекающих в объекте процессов, определении соответствующего этим закономерностям математического описания, обосновании и принятии упрощающих предположений, выполнении необходимых выкладок и приведении результата к принятой форме представления модели. *Экспериментальные методы* основаны на использовании внешних проявлений свойств объекта, фиксируе-

мых во время эксплуатации однотипных объектов или при проведении целенаправленных экспериментов.

Несмотря на эвристический характер многих операций моделирования, имеется ряд положений и приемов, общих для получения моделей различных объектов. Достаточно общий характер имеют методика макро моделирования, математические методы планирования экспериментов, а также алгоритмы формализуемых операций расчета численных значений параметров и определения областей адекватности.

8.3.6. Информационное обеспечение САПР

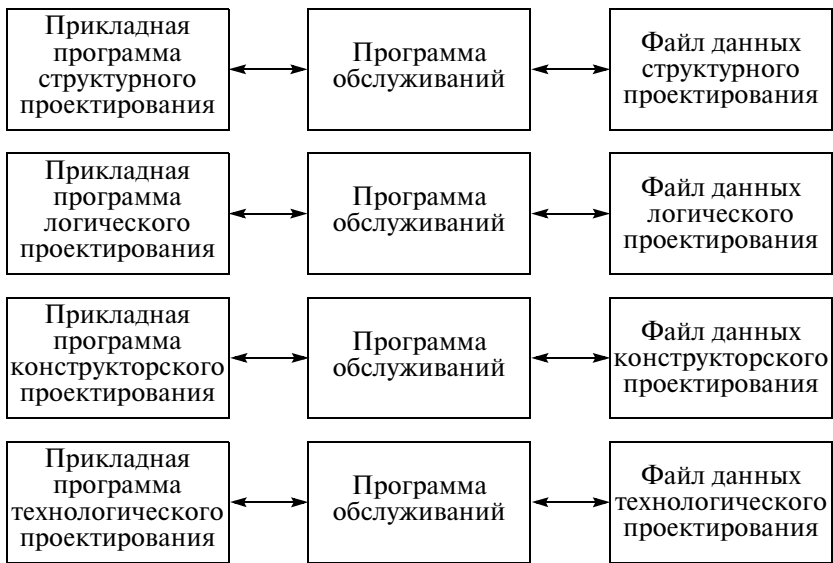
Основные понятия и определения. Основу информационного обеспечения подсистем САПР составляет совокупность данных, которые необходимы для выполнения процесса проектирования. Совокупность компонентов ИО образует информационную базу, называемую базой данных (БД) САПР.

Данные — это сведения о некоторых фактах, позволяющие делать определенные выводы. Взаимосвязанные данные часто называют *системой данных*, и хранимые данные называют информационным фондом. Основное назначение информационного обеспечения — предоставлять пользователям САПР достоверную информацию в определенном виде.

Объектом считают любой предмет, событие, понятие и т. п., о которых приводятся данные. Все объекты характеризуются атрибутами. Например, объект ЭВМ можно характеризовать такими атрибутами: скоростью вычислений, объемом оперативной памяти, числом элементарных операций, числом процессоров, габаритами, количеством мультиплексных каналов и т. п. Сведения, содержащиеся в каждом атрибуте, называют *значениями данных*.

Атрибуты, по значениям которых определяют значения других атрибутов, называют *идентификаторами объекта* или *ключевыми элементами данных*. Отметим, что один и тот же объект могут идентифицировать несколько элементов данных. Их тогда считают кандидатами в идентификаторы. Проблему выбора идентификатора из нескольких кандидатов решает пользователь САПР.

Объединение значений связанных атрибутов называют *записью данных*. Упорядоченную совокупность записей данных называют *файлом данных* или *набором данных*. В настоящее время разработаны универсальные подпрограммы, реализующие методы доступа к файлам, их обработку и обслуживание. Эти подпрограммы являются частью ОС ЭВМ. На рис. 8.32, а показано вза-



а)



б)

Рис. 8.32. Взаимнооднозначное соответствие между прикладными программами и файлом данных (а) и пример использования универсальных методов доступа (б)

имнооднозначное соответствие между прикладными программами и файлами данных в САПР печатных плат. На рис. 8.32, б показано использование универсальных методов доступа вместо единичных подпрограмм обслуживания файлов данных.

При обработке данных с расположением одних и тех же элементов данных в нескольких файлах возникают трудности, связанные с избыточностью данных, что требует нескольких процедур ввода, обновления и формирования; опасностью нарушения непротиворечивости данных, что связано с хранением одной и той же информации в нескольких местах; ограниченностью разделения данных; ограничениями по доступности данных; негибкостью к изменениям; сложностью управления процессом решения задачи.

Существует много адекватных и в то же время неформальных определений базы данных. Приведем те из них, которые получили наибольшее распространение.

База данных — это совокупность специально организованных данных, рассчитанных на применение в большом количестве прикладных программ. *База данных* — это набор файлов, работа с которыми обеспечивается специальным пакетом прикладных программ — системой управления базой данных — с целью создания массивов данных, их обновления и получения справок. *База данных* — это совокупность данных, отображающая состояние объектов и их отношений в рассматриваемой предметной области. *База данных* — совокупность всех тех данных, которые обрабатываются более чем в одной программе (модуле).

Приведем основное различие между БД и файлом данных. Файл данных имеет несколько назначений, но соответствует одному представлению хранимых данных. База данных имеет также несколько назначений, но соответствует различным представлениям о хранимых данных.

Сформулируем основные требования к БД:

1. Целостность данных — их непротиворечивость и достоверность.
2. Организация БД должна обеспечивать согласование времени выборки данных прикладными программами с частотами их использования прикладными программами САПР.
3. Универсальность, т. е. наличие в БД всех необходимых данных и возможности доступа к ним в процессе решения проектной задачи.
4. Открытость БД для внесения в нее новой информации.
5. Наличие языков высокого уровня взаимодействия пользователей с БД.

6. Секретность, т. е. невозможность несанкционированного доступа к информации и ее изменений.

7. Оптимизация организации БД — минимизация избыточности данных.

Одним из принципов построения САПР является информационная согласованность частей ее программного обеспечения, т. е. пригодность результатов выполнения одной проектной процедуры для использования другой проектной процедурой без их трудоемкого ручного преобразования пользователем. Отсюда вытекают следующие условия информационной согласованности:

- использование программами одной и той же подсистемы САПР единой БД;
- использование единого внутреннего языка для представления данных.

Комплексная автоматизация процесса проектирования объекта предполагает информационную согласованность не только отдельных программ подсистем САПР, но и самих подсистем между собой. Способом достижения этой согласованности является единство информационного обеспечения.

Основные способы информационного согласования подсистем САПР достигаются либо созданием единой БД, либо сопряжением нескольких БД с помощью специальных программ, которые перекодируют информацию, приводя ее к требуемому виду.

8.3.7. Программное обеспечение САПР

Программное обеспечение (ПО) совместно с информационным являются звеньями, связывающими все виды обеспечения САПР в единую систему проектирования.

К программному обеспечению САПР предъявляется ряд системных требований, от степени удовлетворения которых в значительной мере зависит эффективность всей системы автоматизированного проектирования, а именно требование эффективной программной реализации алгоритмов, требование информационной согласованности программ друг с другом и с системой управления базы данных, требование модульности и наращиваемости ПО САПР.

Программное обеспечение САПР состоит из двух составных частей — общего и специального ПО. Общее ПО предназначено для планирования и организации процесса выполнения прикладных программ и определяется существующей номенклатурой операционных систем и комплексов программ технического обслуживания. Специальное ПО представляет

собой пакеты прикладных программ, реализующих алгоритмы выполнения проектных операций и процедур. Эти прикладные программы необходимы группе пользователей, занимающихся автоматизированным проектированием в конкретной проблемной области.

Функции и состав общего программного обеспечения. Под операционной системой (ОС) ЭВМ понимают комплект системных программ, управляющих всеми ресурсами системы: оперативной памятью, временем работы процессора, каналами обмена данными и периферийными устройствами, программными модулями и др. Рассмотрим функции и состав ОС ЭВМ. ОС ЭВМ предназначена для автоматизации обработки данных в вычислительных системах, построенных на основе моделей ЭВМ. ЭВМ увеличивает пропускную способность вычислительной системы; уменьшает время реакции системы и повышает надежность ее функционирования. Это достигается благодаря непрерывной обработке потоков заданий, их размещению в системных очередях, предварительному планированию и инициированию выполнения очередных заданий.

Основными компонентами ОС ЭВМ являются управляющие и обрабатывающие программы.

Управляющие программы ОС ЭВМ. Управляют вычислительным процессом, реализуют функции управления задачами, заданиями и данными следят за выполнением обрабатывающих программ и прикладных программ, составленных пользователем.

Программа управления, задачами (супервизор) управляет всем ходом вычислительного процесса в ЭВМ и предназначена для наблюдения и управления всеми задачами в системе. Супервизор выделяет задачам необходимые ресурсы для их решения в одно- или мультипрограммном режиме.

В однопрограммном режиме образуется только одна задача, в рамках которой выполняются как программы ОС, так и программы пользователей, при этом все технические средства ЭВМ находятся в распоряжении этой задачи.

При необходимости одновременного решения нескольких не зависящих друг от друга задач супервизор реализует либо мультипрограммный режим, либо режим разделения времени. В этом случае супервизор производит переключение с решения одной задачи на решение другой, осуществляет динамическое распределение памяти и других ресурсов ЭВМ, обрабатывает через систему прерываний поступающие запросы.

Одновременное выполнение нескольких задач в мультипрограммном режиме достигается благодаря тому, что при их выпол-

нении всегда возникают паузы, связанные с ожиданием завершения операции ввода и вывода, с истечением заданного интервала времени, с ожиданием дополнительных данных для продолжения выполнения программы и др. На период ожидания выполнение данной задачи откладывается и процессор переходит к выполнению следующей задачи и т. д. Для оператора эти программы выполняются одновременно, при этом ОС обеспечивает динамическое распределение памяти и возможность работы по приоритету со многими задачами и заданиями.

Программы управления заданиями (планировщик заданий и главный планировщик) выполняют функции планирования вычислительного процесса. Задание представляет собой внешнюю независимую единицу работы, выполняемой ОС. Задания состоят из пунктов или шагов, связанных между собой прямо или косвенно. Выполнение одного пункта задания это выполнение некоторой обрабатываемой программы.

Управление прохождением одного или нескольких заданий через вычислительную систему осуществляют программы управления заданиями. Планируя структуру задания, программист может использовать любые программы, которые включены в библиотеку ОС. Перед обработкой задание объединяется с другими, образуя один входной пакет заданий. Планировщик состоит из планировщика заданий и главного планировщика.

Планировщик заданий подготавливает задания для выполнения и планирует их выполнение последовательно или с учетом приоритетов. Главный планировщик устанавливает связь оператора с ОС, выводит сообщения на консоль оператора и вводит с консоли его команд.

Язык управления заданиями содержит девять типов операторов: задания (JOB), исполнительный (EXEC), описания данных (DD), командный, разделительный, пустой, процедуры (PROC), конца процедуры (PEND), комментариев. Эти операторы имеют соответствующие параметры, которые позволяют планировщику заданий регулировать выполнение заданий и пунктов заданий, отыскивать и размещать данные, распределять ресурсы ввода/вывода, а главному планировщику — осуществлять связь с оператором вычислительной системы.

Программы управления данными предназначены для организации хранения, поиска данных и обмена данными между оперативной памятью и внешними устройствами.

Операционная система ЭВМ допускает четыре типа организации данных: последовательную, индексно-последовательную, прямую и библиотечную.

При последовательной организации наборов данных записи подвергаются обработке в порядке, совпадающем с их физической последовательностью на внешнем носителе информации.

Индексно-последовательная организация представляет возможность адресоваться непосредственно к записи без считывания предыдущих записей или без информации о расположении данной записи.

Прямая организация наборов данных допускает возможность произвольной обработки записей.

Наборы данных с библиотечной организацией состоят из последовательно организованных разделов, каждый из которых имеет свое имя.

Последовательные наборы данных могут быть организованы на любом внешнем носителе, а остальные вышеперечисленные способы организации наборов данных — только на устройствах прямого доступа (магнитных дисках или барабанах).

Все имеющиеся в системе наборы данных зарегистрированы в каталоге системы. Каталогизация позволяет обращаться к наборам данных только по имени, уменьшает степень ручного вмешательства в процессы идентификации и хранения информации, сводит к минимуму ошибки обслуживающего персонала. Каталог системы устроен подобно оглавлению библиотеки и расположен в устройствах прямого доступа.

Обрабатывающие программы. Служат для подготовки прикладных программ к исполнению на ЭВМ. Подготовка к исполнению включает в себя трансляцию исходной программы, ее редактирование и загрузку. К обрабатывающим программам относятся трансляторы с основных алгоритмических языков (ФОРТРАН, ПЛ/1, ПАСКАЛЬ, АЛГОЛ и др.), редактор связей, программы отладки и вспомогательные программы (утилиты).

Трансляторы предназначены для перевода текста исходных программ (модулей) на машинный язык, построения объектного модуля в форме, удобной для его последующей обработки, вывода на печать листингов исходных программ, а также диагностического сообщения о лексических, синтаксических и семантических ошибках.

Объектный модуль представляет собой последовательность машинных команд. Он не содержит стандартных подпрограмм, необходимых для работы программы пользователя. *Исходный модуль* в простейшем случае размещается на перфокартах, а объектный модуль и промежуточные наборы данных — на магнитных дисках.

Для организации машинной реализации к объектному модулю необходимо добавить недостающие стандартные программы.

Эту работу называют *редактированием*, а соответствующую программу — *редактором связей* или просто *редактором*. Недостающие стандартные программы редактор связей берет из библиотеки стандартных подпрограмм. Редактор создает из объектного модуля и стандартных подпрограмм *загрузочный модуль*, который может рассматриваться в качестве выполняемой машинной программы. Загрузка загрузочного модуля в оперативную память для непосредственного исполнения осуществляется программой выборки.

После получения загрузочного модуля происходит выполнение программы пользователя. В процессе выполнения программы ОС контролирует ее выполнение, а результаты контроля выводит на алфавитно-цифровое печатающее устройство. Взаимосвязь функционирования обрабатывающих программ в ОС ЭВМ представлена на рис. 8.33.

К обрабатываемым программам относятся также программа сортировки-объединения для организации расположения информации во внешней памяти, программы-утилиты для преобразования набора данных и редактирования его записей, для копирования набора данных с носителя на носитель, перегруппировки записей и упорядочения массивов для распечатки сведений об имеющихся в библиотеках программах и данных и др.

Операционная система имеет модульную структуру, которая позволяет приспособлять систему к конкретным конфигурациям используемых технических средств. Процесс создания конкретной структуры ОС, учитывающей особенности конфигурации ЭВМ и задач потребителя, называют генерацией системы.

Функции и состав специального программного обеспечения. В состав программного обеспечения САПР входят пакеты прикладных программ (ППП), ориентированные на решение определенных задач проектирования и реализуемые как надстройка над ОС. Основу ППП составляет множество программных модулей, каждый из которых является программой реализации определенной проектной процедуры либо программой реализации некоторого алгоритма (или фрагмента алгоритма) проектирования.

Специальное ПО САПР может иметь собственную ОС или же использовать одну из базовых ОС ЭВМ. Программное обеспечение с собственной ОС имеет сложную структуру. В состав такого ПО входят универсальный или специализированный монитор САПР, организующий вычислительный процесс в соответствии с принятым алгоритмом проектирования; транслятор или интерпретатор с входного языка; набор программных модулей, составляющих тело ППП; набор обслуживающих программ и т. п. Типовая структура ПО САПР представлена на рис. 8.34.

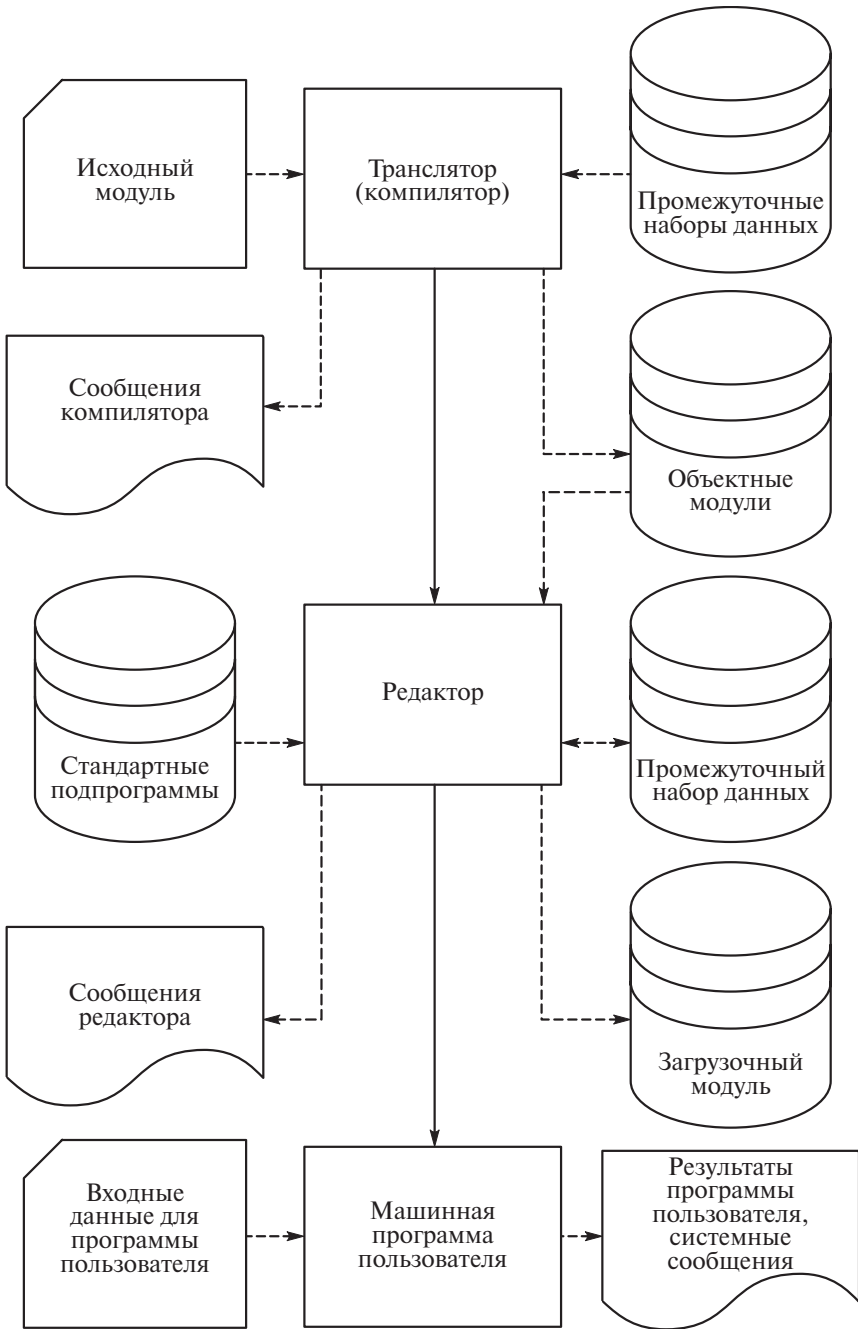


Рис. 8.33. Взаимосвязь функционирования обрабатывающих программ в ОС ЭВМ

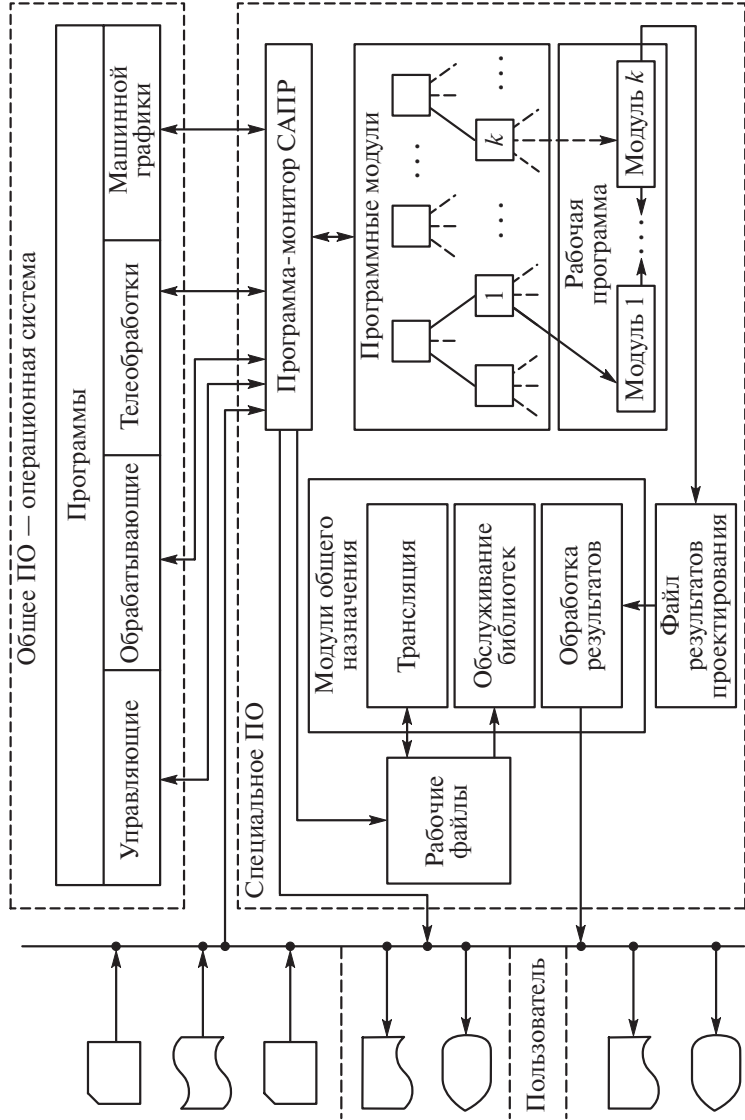


Рис. 8.34. Типовая структура ПО САПР

Обращение к специальному ПО реализуется в форме запроса, который содержит требования к выполнению проектных процедур и исходные данные для проектирования. Последовательность подключения программных модулей для выполнения проектных процедур определяется *программой-монитором* САПР. Взаимосвязь отдельных программных модулей обеспечивает возможность организации сложных маршрутов проектирования.

Непосредственную реализацию проектных процедур производят с помощью *программных модулей*. Каждый программный модуль в свою очередь может разделяться на более мелкие модули. В подобной структуре должны быть четко определены функции каждого программного модуля. Программы специального ПО группируются в ППП, каждый из которых проблемно ориентирован на обслуживание задач отдельных подсистем САПР. Поэтому в состав ППП входят как программные модули, ориентированные на реализацию узкоспециализированных проектных процедур и, как правило, создаваемые разработчиками конкретных САПР, так и модули общего назначения (оптимизации, математических моделей используемых элементов и т. п.). Модули общего назначения иногда называют *библиотечными модулями*.

В соответствии с алгоритмом проектирования для каждого нового объекта и маршрута его проектирования автоматически составляются рабочие программы. В зависимости от требований ТЗ и конкретных условий на каждом этапе разработки могут использоваться различные маршруты проектирования и соответственно различные рабочие программы.

Связь программных модулей осуществляется с помощью буферных файлов. В САПР используется два типа буферных файлов — рабочие и результатов проектирования. В рабочих файлах запоминается информация о структуре проектируемого объекта, типе и параметрах составляющих его элементов, в файле результатов накапливаются результаты автоматизированного проектирования, которые после обработки документируются и выводятся на устройства отображения информации.

Модульная структура ПО САПР позволяет легко наращивать функциональные возможности системы путем расширения набора программных модулей и их модификаций, а также позволяет пользователю легко развивать систему.

Многообразие языков программирования, сложность проектных процедур и разнообразие вариантов маршрутов проектирования требуют концентрации усилий разработчиков специального ПО САПР. Цикл разработки программного обеспечения включает в себя: анализ требований, предъявляемых к САПР; определе-

ние точного описания функций и проектных процедур (спецификаций), реализуемых с помощью ПО; разработку алгоритмов реализации функций, проектных процедур; программных модулей с использованием алгоритмических языков высокого уровня и методов структурного программирования; тестирование программ; эксплуатацию и сопровождение.

На первом этапе разработки ПО определяются требования, выполнение которых позволяет получить требуемое решение поставленной проблемы. Анализ требований способствует лучшему пониманию проблемы и компромиссных ситуаций, что помогает выбору наилучшего решения.

На этапе определения спецификаций задаются структура входных и выходных данных, возможные типы проектных процедур и маршрутов проектирования. Чем детальнее разработаны проектные процедуры и маршруты проектирования, тем меньше вероятность возникновения ошибок и тем легче организовать информационный интерфейс между программными модулями.

При организации разработки ПО целесообразно структуру пакета прикладных программ разбивать на отдельные по возможности независимые части таким образом, чтобы ответственность за реализацию каждой такой части можно было возложить либо на одного разработчика, либо на группу. При этом для каждой части ППП должны быть сформулированы предъявляемые к ней требования: реализуемые проектные процедуры, размерность и др.

Проверку корректности ПО и его отладку производят на этапе тестирования. Тестирование подразделяют на три стадии: автономное, комплексное и системное. При автономном тестировании каждый программный модуль проверяют с помощью данных, подготавливаемых программистом. Модуль, прошедший автономное тестирование, подвергают комплексному тестированию, при котором проверяют отдельные группы программных модулей. В результате комплексного тестирования возможно обнаружение ошибок, пропущенных при автономном тестировании. При системном тестировании испытывают ППП с помощью независимых тестов.

Практика разработки ПО показывает, что наибольшее число ошибок связано с разработкой информационного интерфейса. Большие трудности возникают при разработке интерфейса между программными модулями, написанными разными программистами. Поскольку число таких интерфейсов при N исполнителях составляет $N(N-1)/2$ и возрастает пропорционально квадрату числа исполнителей, проблема становится весьма сложной при разработке ПО группой из нескольких человек, так как взаимо-

действие программистов друг с другом снижает производительность их труда и требует дополнительных затрат на тестирование. Решение вопросов унификации и стандартизации должно быть основополагающим при разработке специального ПО САПР.

Мониторы и языковые процессоры. К наиболее сложным программам САПР относятся мониторы ППП и языковые процессоры.

Монитор — это управляющая программа диалогового взаимодействия пользователя с комплексом технических средств САПР. В функции монитора входит: управление работой программного канала обмена с периферийными устройствами; создание различных режимов работы комплекса технических средств и управление ими; загрузка абсолютных программ (загрузочных модулей, настроенных на конкретные адреса) и управление ходом их выполнения; выбор программных модулей в соответствии с маршрутом проектирования; настройка связей выбранных программных модулей; распределение памяти и т. д.

Языковые процессоры предназначены для преобразования информации, выраженной на входном языке, в рабочую программу. Они могут генерировать рабочую программу способами *компиляции* или *интерпретации*, их соответственно называют компиляторами и интерпретаторами.

Компилятор осуществляет лексический и синтаксический анализ для трансляции программы с языка высокого уровня, выполняет основную работу по составлению объектной программы — генерирует объектные модули и команды обращения к библиотечным модулям, а также формирует управляющие предложения для редактора связей относительно состава используемых объектных модулей для формирования загрузочного модуля.

8.3.8. Техническое обеспечение САПР

Организация технических средств САПР. Техническое обеспечение САПР совместно с ПО является инструментальной базой САПР, в среде которой реализуются другие виды обеспечения САПР.

Компоненты технического обеспечения САПР включают в себя средства вычислительной и организационной техники, средства передачи данных, измерительные и другие устройства или их сочетания, обеспечивающие функционирование САПР. Совокупность компонентов ТО образует комплекс технических средств САПР.

Требования к составу и структуре ТС формируются исходя из общих требований к структуре САПР; эффективного решения

выделенного класса задач проектирования, активного включения пользователя в процесс проектирования; возможности работы с графическим материалом, включая процессы как ввода и обработки, так и вывода информации.

Классификация ТС САПР: ТС подготовки и ввода данных; передачи данных; программы обработки данных; архива проектных решений.

Логическая структура технических средств САПР приведена на рис. 8.35. Технические средства САПР объединены в группы взаимодействующего оборудования. Среди этих групп выделяют группы базовой конфигурации. *Базовая конфигурация* — это минимальный состав, позволяющий решать задачи определенного класса.

Существует ряд требований, которые необходимо учитывать при разработке базовых конфигураций: унификация проектных решений; построение развивающейся системы, предусматривающее наращивание и совершенствование компонентов технических средств; физическая совместимость, предусматривающая совместное функционирование всех компонентов комплекса; модульность конфигурации, требующая, чтобы компоненты системы были универсальными и типовыми; минимизация стоимости; согласованность основных параметров компонентов системы.

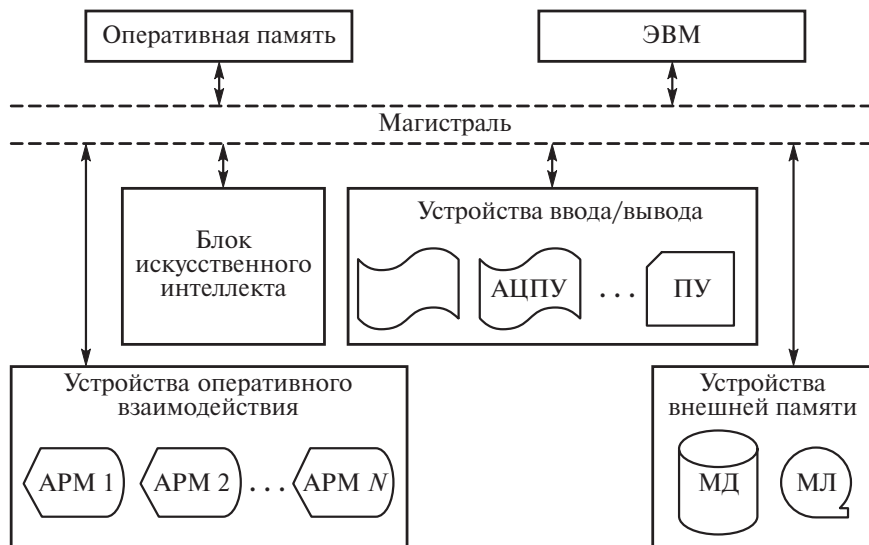


Рис. 8.35. Логическая структура технических средств САПР

Одним из основных компонентов базовых конфигураций является автоматизированное рабочее место. Серийно выпускаются комплексы АРМ-Р, АРМ-М, 15УТ-4-017 и др.

Комплекс АРМ-Р представляет собой совокупность технических и программных средств для организации проектирования радиоаппаратуры, а также для выполнения различных работ, связанных с вводом/выводом, редактированием графической и текстовой информации. Центральным элементом в комплексе является мини-ЭВМ (рис. 8.36), процессор которой связан с общей шиной, с периферийным оборудованием и оперативной памятью.

Общесистемное программное обеспечение АРМ-Р в мультипрограммном режиме поддерживает различные конфигурации аппаратных средств комплекса, ввод/вывод графических данных, подготовку информации для дальнейшей передачи.

Для приближения пользователя к средствам САПР созданы диалоговые комплексы АРМ-2. Разработано несколько модификаций, где для программной обработки данных используют мини-ЭВМ. Комплекс АРМ-2.01 предназначен для разработки радиоэлектронной аппаратуры и машиностроительных объектов, позволяет осуществлять ввод графической и алфавитно-цифровой информации, ее редактирование, отображение и выпуск документации на машинных носителях. Одновременно могут работать восемь пользователей. Технические средства АРМ2-01 вклю-

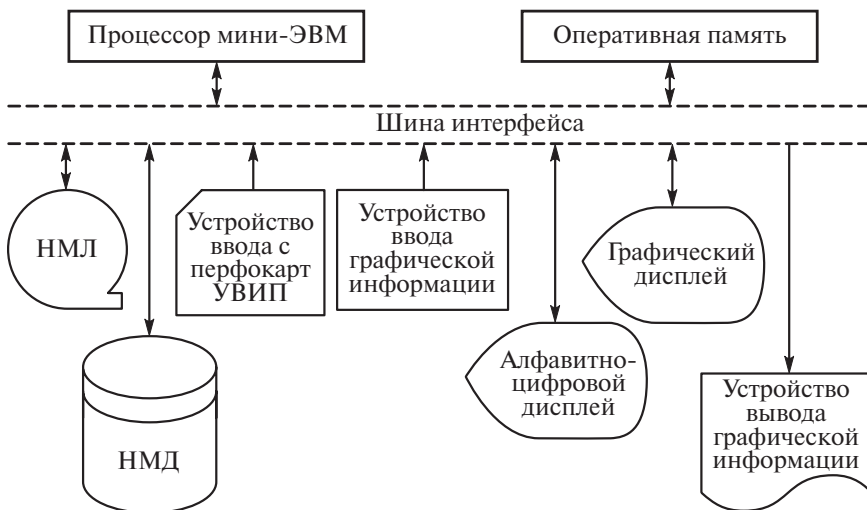


Рис. 8.36. Схема АРМ

чают мини-ЭВМ, терминалы подготовки и ввода графических данных, экранные пульта, графопостроители. Базовое программное обеспечение позволяет описывать на проблемно-ориентированном языке, вводить в ЭВМ и редактировать графическую и текстовую информацию, вести диалог с ЭВМ, создавать на магнитных дисках архив результатов проектирования, выводить информацию на специальные устройства.

Комплексы АРМ-2.02 используют для проектирования деталей машиностроения, технологических процессов их изготовления; АРМ-2.03 — для создания управляющих программ для станков с числовым программным управлением и проектирования цифровой РЭА; АРМ-2.04 — для создания и редактирования программ и выпуска текстовой документации; АРМ-2.05 — для отладки управляющих программ и микропрограмм, исследования их в реальном масштабе времени, занесения программ в постоянные запоминающие устройства ЭВМ с программируемой логикой работы.

В состав системы 15УТ-4-017 входят мини-ЭВМ, рабочие места, оборудованные кодировщиками, алфавитно-цифровые и графические дисплеи, координатографы. Система позволяет проектировать топологию БИС и редактировать информацию в интерактивном режиме. При этом анализируют и редактируют эскиз кристалла и топологию компонентов и кристалла, преобразовывают топологию кристалла в информацию для управления микрофотонаборной установкой. Программное обеспечение системы состоит из следующих блоков:

- системного обеспечения — включает в себя диспетчер системы и набор вспомогательных программ;
- программного обеспечения логического анализа и схемотехнического проектирования — позволяет описывать информацию о схеме и получать эскизы логических и принципиальных схем, автоматически формировать массивы связей и контроль содержащейся в них информации;
- программного обеспечения проектирования топологии и фотошаблонов — позволяет описывать совмещенный топологический чертеж, проектировать топологию в интерактивном режиме, проверять конструкторско-технологические ограничения, а также выводить фрагменты топологии на графопостроитель;
- программного обеспечения общего назначения — выполняет функции реализации директив пользователей и выводит конструкторско-технологическую документацию;
- средств развития программного обеспечения — позволяет совершенствовать программное обеспечение комплекса.

Состав ТС включает одну или несколько ЭВМ с возможностью создания иерархической структуры подключения технических средств нижнего уровня.

Организация ТС САПР предусматривает использование несколькими конструкторско-проектными организациями. В состав ТС входят высокопроизводительная аппаратура передачи данных (АПД) между частями комплекса и мощные вычислительные средства.

Для САПР крупных научно-производственных объединений предполагается применение сверхбольших систем ТС и сетей САПР коллективного пользования. Комплекс ТС таких САПР представляет собой единую вычислительную сеть, объединяющую территориально распределенные центры. Пользователь любого из функциональных уровней может увеличить вычислительную мощность своего уровня ТС за счет подключения его к одному из мощных вычислительных комплексов, содержащихся в этой сети, получить информацию, накопленную другими пользователями, использовать вычислительную технику сети для решения задач.

Развитие таких систем предъявляет повышенные требования к техническим средствам. Необходимо существенное увеличение емкости и уменьшение габаритов внешних накопителей, уменьшение времени выборки информации. Переход на оптические диски доведет емкость до 200 Гбайт на одну поверхность. Необходимо улучшать характеристики терминалов. Намечается переход на графические терминалы со встроенными функциями обработки изображений, имеющие достаточно большую буферную память, модули для подключения к сетям передачи данных.

Для получения возможности подсоединения к сети различных машинных комплексов на всех уровнях необходимо включить в состав технических средств сетей специальные коммутаторы, комплекующие разнотипные ЭВМ.

Режимы работы технических средств САПР. Состав технических средств базовых конфигураций САПР различных уровней в значительной степени определяется характером проектных задач. Существует взаимосвязь между классом решаемых задач и режимом использования ЭВМ. Рассмотрим задачи, решаемые в САПР, с целью выделения характеристик, определяющих выбор различных режимов работы ЭВМ.

По характеру вычислительного процесса решаемые задачи можно разделить на две основные группы: задачи, решаемые без участия пользователя, и задачи, в процессе решения которых необходимо участие пользователя.

По сложности вычисления задачи бывают:

- *задачи первой группы* — задачи, на решение которых требуется более нескольких минут; задачи, время счета которых измеряется секундами;
- *задачи второй группы* — время взаимодействия с пользователем соизмеримо с временем счета задачи: время решения велико по сравнению со временем диалога.

По объему информации задачи, решаемые в САПР, можно разделить на монопольно использующие основную память ЭВМ и частично использующие основную память ЭВМ.

Исходя из этой классификации решаемых задач САПР, можно выделить следующие необходимые режимы работы технических средств:

I — однопрограммный режим, при котором решаемой задаче доступны все ресурсы ЭВМ;

II — мультипрограммный режим с фиксированным количеством задач; при таком режиме ОП ЭВМ делится на фиксированное число разделов, которые определены для выполнения одной задачи в каждом; некоторые внешние устройства (ВУ) могут быть назначены для использования несколькими задачами;

III — мультипрограммный режим с переменным числом задач, все ресурсы ЭВМ общие.

Режим работы технических средств можно классифицировать по удалению проектировщика от основного компонента технических средств:

- местный режим, при котором пользователь работает непосредственно у ЭВМ;
- дистанционный режим, при котором часть периферийного оборудования связана с процессором канала связи.

Режим работы технических средств можно классифицировать по степени участия пользователя в процессе решения задач:

- *пакетный режим*, когда пользователь составляет задание на выполнение программы, которое в составе пакета заданий запускается для обработки на ЭВМ. Обработка задач производится по очереди. После решения пользователю требуется проанализировать результаты обработки своего задания и подготовить новый вариант задания, что замедляет отладку и увеличивает время получения окончательных результатов;
- *режим разделения времени (РРВ)*, при котором каждой решаемой задаче поочередно выделяется определенный квант времени работы процессора. Пользователь во время сеанса работы за абонентским пунктом, используя средс-

тва системы разделения времени (СРВ), может составить, протранслировать, отредактировать программу и приступить к ее выполнению, непосредственно контролируя происходящий процесс. Степень готовности программы зависит от подготовленности пользователя к работе с СРВ.

От выбора правильного режима использования технических средств САПР зависит эффективность эксплуатации технических средств. Поэтому при создании конкретной САПР определенного уровня необходимо провести четкий анализ решаемых задач.

Пакетный режим обработки информации предпочтительнее для задач с большим временем счета и задач, не требующих вмешательства в процесс решения пользователя.

Режим разделения времени удобнее для задач, время счета у которых соизмеримо с временем отклика пользователя на запрос ЭВМ, а также в том числе, когда необходимо вмешательство пользователя в процесс решения.

Технические средства машинной графики. Технические средства машинной графики можно классифицировать по следующим отличительным признакам: назначению, степени автоматизации, методу обработки информации, способу отсчета текущих координат. Указанная схема классификации показана на рис. 8.37.

Рассмотрим устройства графического ввода информации. Они позволяют увеличить производительность труда и повысить достоверность кодируемой информации. В полуавтоматических устройствах кодирование входной информации происходит при участии человека, который, перемещая регистрирующий орган, обходит последовательно все элементы графического документа. Перемещение регистрирующего органа фиксируется. Существуют дополнительные панели для ввода вспомогательной информации (клавиатура печатающего устройства). Представителями таких устройств являются устройства ЭМ-709, ГАРНИ, ПАСГИ, Calcomp, Venson, Aristo и др.

Полуавтоматические системы ввода графической информации могут сопрягаться с ЭВМ, которые выполняют автоматический контроль ошибок, допущенных оператором при считывании, позволяют выбирать фрагменты из библиотеки элементов и ранее введенных участков изображения, осуществляют оперативный вывод закодированной информации. Примерами таких систем служат системы QED и IED фирмы Quest Automation и системы Aristigrid фирмы Aristo.

Автоматические устройства ввода графической информации характеризуются значительным быстродействием и отсутствием

несистематических ошибок, но для их работы требуются графические документы высокого качества и точности.

По методу считывания различают устройства непрерывного, дискретного и непрерывно-дискретного действия. Для графической информации большого объема желательно использовать дискретные устройства, обладающие более жесткими требованиями к точности.

В зависимости от способа отсчета текущих координат устройства ввода графической информации подразделяют на электромеханические, оптико-механические, магнитные, контактные и звуковые.

Примеры устройств ввода:

1) устройство магнитного типа, полуавтомат кодирования графической информации (ПКГИ — 0), входящий в состав АРМ-М. Полуавтомат позволяет кодировать элементы чертежа с указанием типа линий, символы 3 алфавитов числом 256 и произвольные кривые, чертежи с учетом смены системы координат;

2) устройство графического ввода FRED фирмы Quest Automation, состоящего из планшета с матрицей проводов, располо-



Рис. 8.37. Технические средства машинной графики

женных по координатным осям, устройства для цифрового отображения информации, передвижного пульта и ЭВМ NOVA. Задание положения точки определяется при касании шупом поверхности планшета.

Представителями группы устройств вывода графической информации являются чертежные автоматы, позволяющие получать документацию в виде чертежей, графиков, схем, диаграмм.

Классификацию чертежных автоматов можно провести по следующим признакам: способу программного управления; методу обработки данных; принципу действия исполнительного механизма. Классификация чертежных автоматов по этим признакам показана на рис. 8.38. Чертежные автоматы работают как в автономном режиме от машинных носителей информации, так и непосредственно от ЭВМ.

Чертежным автоматам присущи недостатки: невысокое быстродействие, недостаточная надежность. Поэтому прямое сопряжение их с ЭВМ не приводит к желательным результатам. Чертежные автоматы входят в комплекс периферийных устройств со своей малой ЭВМ, которая сопрягается с решающими устройствами. Одними из лучших представителей группы чертежных автоматов являются автоматы «Итекан-2М», «Итекан-3»,

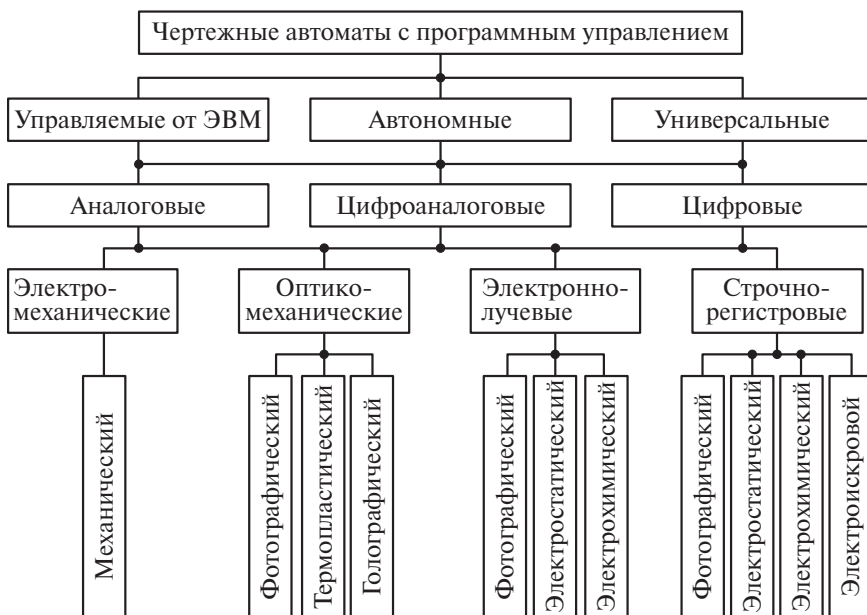


Рис. 8.38. Классификация чертежных автоматов

«Итекан-4», «Итекан-6», различные по назначению, программному управлению и конструктивному исполнению. Изображение вычерчивается цветными чернилами, тушью или карандашом на обычной бумаге или кальке линиями различной толщины. Размеры рабочего поля и максимальная скорость вычерчивания для автомата «Итекан-4» составляют 818×40000 мм и 40 мм/с.

К устройствам графического вывода информации можно отнести программно-управляемые координатографы, предназначенные для изготовления прецизионных фотооригиналов печатных плат, полосковых, микрополосковых линий и др. От чертежных автоматов координатографы отличаются повышенной точностью и способом нанесения изображений. Изображение наносится экспонированием, вырезанием, гравированием, скрайбированием и другими способами.

Программно-управляемые координатографы отличаются большими размерами рабочего поля. Управление осуществляется непосредственно через каналы связи или с помощью промежуточных носителей информации.

Координатограф предназначен для изготовления фотошаблонов микросхем и печатных плат. В состав координатографа входят: FS, пульт управления, координатный стол с размерами рабочего поля 1200×1200 мм, устройство управления с блоками ввода информации, операционное устройство, интерполятор, блоки задания скоростей, обработки информации, ориентации инструмента, технологических операций, управления приводом, цифровой индикации, а также центрального управления. Максимальная скорость перемещений на прямолинейных участках 90 мм/с, на дугах окружностей 25 мм/с.

Наиболее перспективный класс устройств ввода/вывода графической информации — алфавитно-цифровые и графические дисплеи. Скорость обмена информации этих устройств сравнима со скоростью обработки информации в ЭВМ, это обусловило их широкое применение.

Все дисплейные устройства можно классифицировать по следующим основным признакам: назначению и виду отображаемой информации; методу формирования изображения; физическим принципам создания информации. Один из возможных типов классификации дисплеев приведен на рис. 8.39.

Алфавитно-цифровые дисплеи предназначены для обработки символьной информации. Их отличают высокая надежность и экономичность. Рассмотрим основные характеристики наиболее известных алфавитно-цифровых дисплеев.

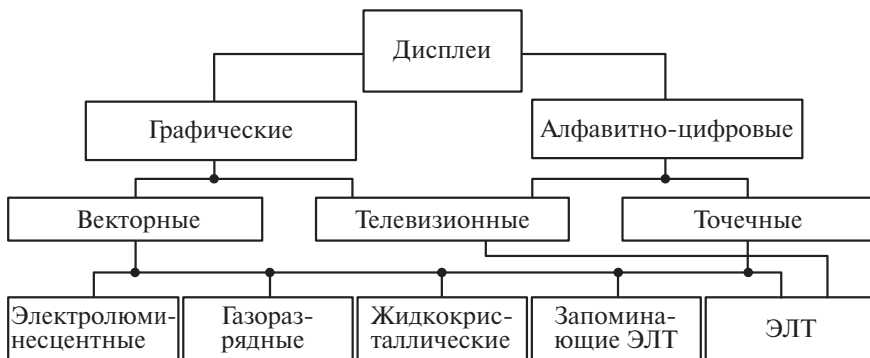


Рис. 8.39. Классификация дисплеев

Изображение знаков осуществляется векторным обходом луча по контуру знака. Графические дисплеи позволяют выводить на экран графические изображения, имеют широкий набор встроенных функций преобразования информации: сдвиг, масштабирование, поворот, редактирование и др.

8.3.9. Организация взаимодействия конструктора с ЭВМ

Основные тенденции развития САПР определяют преимущественное развитие интерактивных систем автоматизированного проектирования. Центральное место в таких системах занимает диалог «конструктор – ЭВМ». Организация диалога обеспечивается информационными, программными и техническими средствами САПР.

При выборе технических средств САПР, обеспечивающих интерактивное взаимодействие конструктора с процессом решения, среди возможных альтернатив следует руководствоваться следующим.

Используемая ЭВМ должна позволять организовать работу определенного числа пользователей (т. е. удовлетворять требованиям по производительности) в режиме разделения времени.

Используемое терминальное оборудование должно отвечать эргономическим, эстетическим требованиям и требованиям эффективности работы.

К терминальному оборудованию можно отнести следующие устройства: пультовую пишущую машинку, телетайп, алфавитно-цифровой и графический дисплей, акустические устройства.

Применение этих устройств обусловлено определенным классом решаемых задач в САПР. В настоящее время практически во всех применениях становится экономически невыгодным использование телетайпов и терминалов с посимвольным вводом/выводом информации. Дисплеи со страничной обработкой информации позволяют увеличить производительность работы проектировщика.

Применение графических дисплеев идет в направлении представления терминалов как автономных систем со специальными операционными системами управления работой отдельных аппаратных и программных компонентов терминала и взаимосвязью терминала с основной ЭВМ. Появление супермикроЭВМ, базиса автономного комплекса, приблизило по своим возможностям к мини-ЭВМ. Развитие комплексов идет по двум направлениям: разработки однопрограммных систем, рассчитанных только на одного пользователя, и мультипрограммных систем, рассчитанных на одновременную работу нескольких пользователей, с реализацией виртуальной памяти.

Для достижения высокой производительности систем, необходимой для одновременной работы многих пользователей и соединения функциональных блоков системы, используют различные каналы, объединяющие процессор, оперативную память, матричный процессор и различные источники информации.

Схема одного из автономных терминальных комплексов показана на рис 8.40. В графических терминалах для управления функционированием всех компонент применяют микропроцессоры; для обработки графической информации необходимы специализированные процессоры; к терминалу могут подключаться устройства памяти. В целом тенденции развития терминалов могут быть охарактеризованы как движение в сторону многофункциональных систем с распределенной обработкой информации, основанных на широком использовании БИС.

Диалоговые средства. Рассмотрим общие требования к диалоговым средствам (ДС), которые приобретают все больший вес в современных САПР. К *диалоговым средствам* САПР относят средства, обеспечивающие прямое (интерактивное) взаимодействие пользователя с КСАП, осуществляемое в реальном масштабе времени.

Диалоговый режим — это проектирование с применением ДС, при котором пользователь, взаимодействуя с КСАП, выдает инструкции управления проектированием. Диалоговые средства составляют часть КСАП и обеспечивают перевод САПР из пакетного режима в диалоговый. *Пакетный режим* — это проектирование

при отсутствии ДС и непосредственного воздействия пользователя на процесс проектирования. В зависимости от вида представления данных ДС подразделяют на символные, графические и смешанные. К символным относят ДС, обеспечивающие представление данных в символьном (алфавитно-цифровом) виде, к графическим — ДС в графическом виде, а к смешанным — ДС как в графическом, так и в символьном виде. В зависимости от режима применения ДС САПР подразделяют на средства автономного, неавтономного и смешанного применения.

В состав ДС САПР входят:

- МО и ЛО, содержащие инструкции и методики ведения диалогового режима, язык диалогового взаимодействия и язык внутреннего представления данных;
- ТО, в которое входят устройства ввода/вывода данных на электронных и других типах дисплеев, процессоры ЭВМ, устройства обмена данными;
- ПО, имеющие программы диалогового ввода и вывода данных, программы интерпретатора языка диалогового

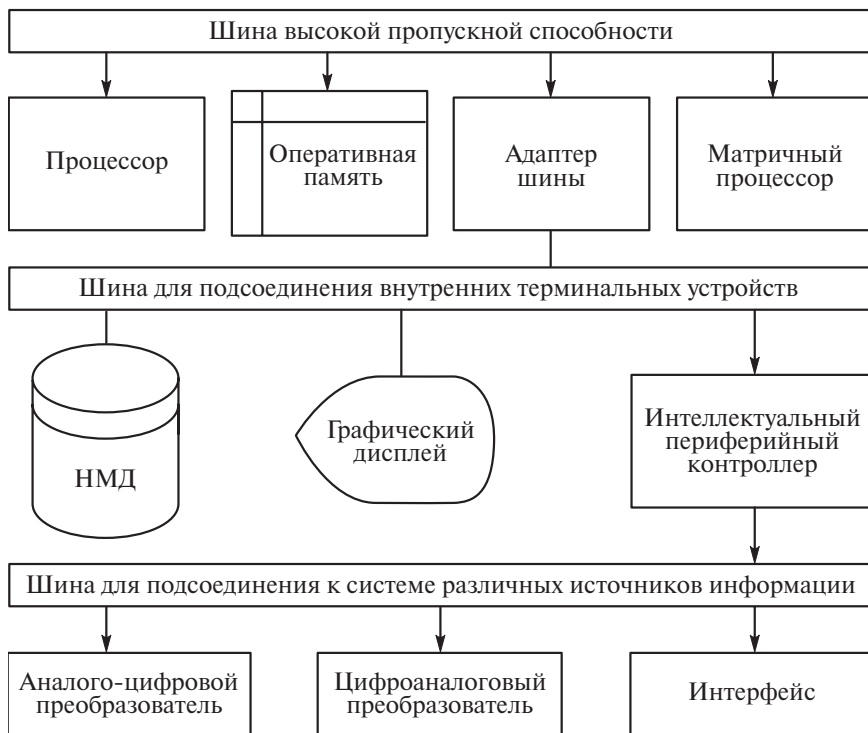


Рис. 8.40. Схема автономного терминального комплекса

взаимодействия, программы управления процессом диалога, программы формирования результатов проектирования, программы вывода результатов на дисплей и их редактирование, программы взаимодействия с другими обслуживающими подсистемами САПР, программы регистрации процесса диалога и выдачи пользователю справок, инструкций, сообщений об ошибках;

- ИО, в которое входят БД для процесса диалога, БД для регистрации и корректировки результатов проектирования, библиотека типовых графических элементов, технических данных и параметров типовых элементов проектирования.

Рекомендуется предусматривать в составе ДС средства обучения и самообучения пользователя, а также адаптацию ДС по отношению к возможностям пользователя.

Математическое обеспечение ДС САПР должно содержать оригинальные и типовые методы проектирования. Лингвистическое обеспечение ДС САПР базируется на естественных языках, общепринятых символьных и графических образах языков, оно должно быть инвариантным по отношению к ИО ДС САПР.

Технические ДС САПР обеспечивают ввод/вывод данных с использованием символьных и графических образов естественных языков. Технические ДС строят на основе серийно изготавливаемого периферийного оборудования ЭВМ, обеспечивающего ввод/вывод данных.

Программное обеспечение ДС САПР создают с использованием базового ПО средств вычислительной техники. Программное обеспечение ДС подразделяют на общее и специализированное.

Общее ПО ДС обеспечивает ввод и вывод информации на диалоговый терминал; выдачу пользователю справок, инструкций и сообщений об ошибках; регистрацию процесса диалога и управление им; адаптацию структуры диалога с учетом особенностей конкретного пользователя; управление базами данных и их защиту; редактирование и манипулирование данными; формально-логический контроль диалога; восстановление процесса диалога.

Специализированное ПО ДС обеспечивает трансляцию или интерпретацию входных сообщений; формирование информации пользователю о результатах проектирования; лексический контроль входных сообщений. Программное обеспечение ДС должно иметь модульную структуру и адаптироваться к изменению структуры и средств ведения диалога. Информационное обеспечение ДС САПР должно быть совместимым с ИО недиалоговых средств САПР. В него должны входить БД для управления процессом диалога и обеспечения режимов обучения и адаптации.

8.3.10. Стадии создания САПР

Различают внешнее и внутреннее проектирование (рис. 8.41). К внешнему проектированию относятся следующие стадии:

- *предпроектные исследования* — производится обследование проектной организации, оформление технического отчета, а также его согласование и утверждение;
- разработка, согласование и утверждение *технического задания* — выполняются совместно разработчиком и заказчиком.

К внутреннему проектированию относятся следующие стадии: разработка *технического предложения* — выбирают и обосновывают оптимальный вариант САПР. Производится разработка комплекта документации, а также согласование и утверждение технического предложения;

- *эскизный проект* — разрабатывают принципиальные решения по созданию САПР и документации, согласовывают и утверждают эскизный проект;
- *технический проект* — разрабатывают окончательные решения по созданию САПР и документации, согласовывают и утверждают технический проект;
- *рабочий проект* — разрабатывают рабочую документацию по САПР, согласовывают и утверждают рабочий проект;

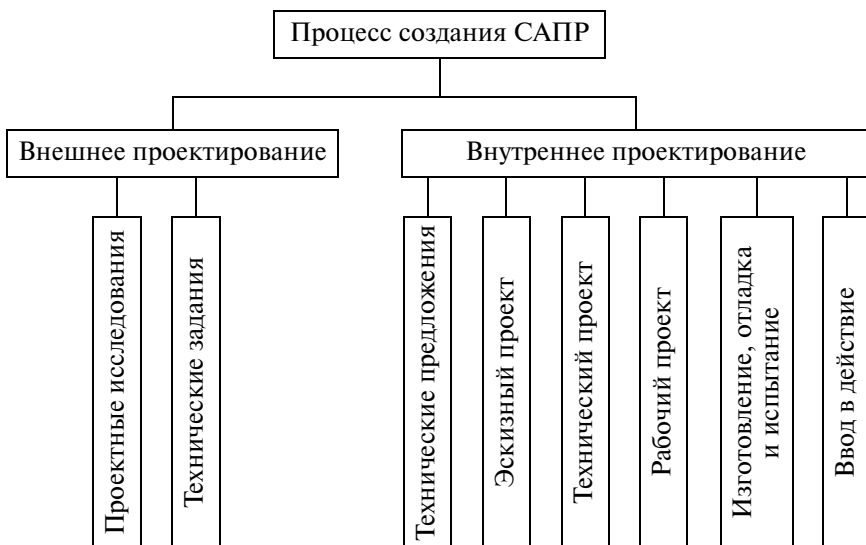


Рис. 8.41. Стадии создания САПР

- *изготовление, отладка и испытания* — изготавливают и отлаживают компоненты САПР, производят монтаж, наладку и испытание КСАП, осуществляют подготовку организации к вводу системы в действие;
- *ввод в действие* — производят опытное функционирование и приемочные испытания САПР.

Работы по подготовке организации к вводу в действие САПР, подсистем и компонентов должны производиться на всех стадиях ее создания. Перечисленные стадии САПР не всегда являются обязательными. Иногда они отсутствуют или объединяются в одну или несколько стадий в зависимости от конкретных требований на проектируемую САПР.

Рассмотрим более подробно основные стадии создания САПР. К работам на стадии предпроектных исследований (ПИ) относят обследование проектной организации, оценку возможности создания САПР, сбор данных, описание существующих САПР и их анализ, сбор предложений по созданию САПР, составу подсистем, разработке компонентов САПР, формированию технических требований к функциям и структуре САПР, видов обеспечения, общесистемных принципов создания САПР. Общее руководство ПИ осуществляет главный конструктор САПР. По окончании ПИ оформляется и утверждается научно-технический отчет.

Техническое задание является исходным и обязательным документом для создания, приемки или сдачи системы и должно содержать все исходные данные и требования, необходимые для создания САПР. Техническое задание на создание САПР разрабатывает организация — разработчик системы с учетом выполнения предпроектных исследований на основе требований заказчика. Оно согласовывается с организацией — пользователем САПР и по необходимости с другими заинтересованными организациями. Техническое задание на создание САПР утверждает заказчик. Изменяют и уточняют утвержденное ТЗ в процессе создания подсистемы с помощью дополнений в порядке, установленном в ТЗ. Полное ТЗ на создание САПР ЭВА должно содержать следующие разделы:

- наименование и область применения — указаны полное или условное наименование (индекс) САПР, краткая характеристика области ее применения;
- основание для создания — указаны полное наименование директивных документов, на основании которых создают САПР, организация, утвердившая эти документы, и дата утверждения, а также наименование или условное обозначение темы, в рамках которой создается САПР;

- характеристика объектов проектирования — даны сведения о назначении, составе, условиях применения разрабатываемого объекта;
- цель и назначение — указаны цель создания САПР, ее назначение и критерий эффективности ее функционирования;
- характеристика процесса проектирования — приведены общее описание процесса проектирования, требования к входным и выходным данным, а также требования по разделению проектных процедур, выполняемых при автоматизированном и неавтоматизированном проектировании;
- требования к САПР — приведены требования к САПР в целом, к составу ее подсистем, к включению в САПР ранее созданных подсистем и компонентов, требования к взаимосвязи САПР с другими системами (например, с АСУ, АСУТП), видам обеспечения и возможности развития САПР;
- технико-экономические показатели — оценены затраты на создание САПР, источники экономии, ожидаемая эффективность от применения САПР, требования к технико-экономическим показателям объекта, которые будут достигнуты в результате функционирования САПР;
- стадии и этапы — установлены необходимые стадии создания, очередность ввода в действие САПР, этапы по стадиям, сроки выполнения работ и исполнители, перечень документации, предъявляемой по окончании стадии, объемы работ в соответствии с действующими нормативно-техническими и методическими документами;
- порядок испытаний и ввода в действие — определены требования к проведению испытаний и вводу в действие САПР и их подсистем;
- источники разработки — даны перечень научно-исследовательских, опытно-конструкторских и экспериментальных работ, нормативных документов, методических материалов и т. п., используемых при создании САПР;
- приложения — включают таблицы, схемы, расчеты, библиографию, листинги программ, обоснования, доказательства и т. п.

Техническое задание на создание САПР оформляют в соответствии с общими требованиями к текстовым документам.

Целью технического предложения (ТП) является выбор рациональных вариантов САПР, учитывающих требования ТЗ. При разработке ТП выполняются следующие работы: анализ процессов автоматизированного проектирования; выявление возмож-

ных вариантов структуры для САПР на основе разработки модулей, реализующих процессы проектирования; выбор рациональных вариантов (варианта) структуры САПР; технико-экономическое обоснование выбранного варианта; составление дополнительных по сравнению с ТЗ требований к САПР; уточнение требований к содержанию работ на последующих стадиях САПР.

Техническое предложение разрабатывает организация, являющаяся главным разработчиком системы, его согласовывают с заказчиком и со всеми заинтересованными организациями. Техническое предложение на создание САПР должно содержать следующие разделы:

- общие положения — цель разработки ТП, наименование и дата утверждения ТЗ; назначение и область применения — назначение, характеристика, область и условия применения создаваемой САПР; описание объекта проектирования в САПР — основные составные элементы объектов проектирования, их взаимосвязь, схема деления; описание процесса автоматизированного проектирования — результаты анализа процесса АП; характеристика и анализ вариантов структуры САПР — результаты синтеза и анализа структуры САПР с выделением подсистем и компонентов и связей между ними, предложения по использованию существующих подсистем и компонентов САПР;
- обоснование и описание выбранного варианта — по результатам рассмотрения вариантов структуры САПР обосновывается и описывается выбранный вариант САПР с указанием взаимодействия подсистем и компонентов и выполняемых ими функций;
- технико-экономическое обоснование — основные технико-экономические показатели создаваемой САПР;
- предложения по содержанию и организации работ на последующих стадиях — уточненные по сравнению с ТЗ данные по очередности и содержанию работ.

Техническое предложение может иметь по необходимости дополнительные разделы и приложения. В последних приводят схемы, описания, расчеты, необходимые для качественного обоснования выбранного варианта САПР.

На стадии эскизного проекта выполняют:

- принятие основных решений по взаимодействию САПР с другими системами;
- принятие основных технических решений по структуре подсистем САПР, разработку вопросов функционирования подсистем на уровне компонентов;

- описание исходных данных и требований на разработку языков проектирования, алгоритмов, компонентов ИО по подсистемам;
- оформление и утверждение эскизного проекта.

На стадии технического проекта выполняют: принятие решений по новому процессу проектирования с обеспечением взаимодействия и совместимости автоматических и автоматизированных процедур, получение окончательной схемы функционирования САПР в целом; разработку структуры и состава подсистем САПР; получение окончательной структуры всех видов обеспечения САПР; выбор математических моделей объекта проектирования и его элементов; разработку алгоритмов проектных операций; разработку требований на создание программ реализации процедур проектирования; разработку алгоритмов, языков проектирования, компонентов ИО, формирование общесистемного программного обеспечения; расчет производительности и выбор режимов работы средств технического обеспечения; определение требований к представлению исходных данных, результатов проектирования и проектных документов; оформление и утверждение совокупности документов, составляющих технический проект.

Технический проект включает в себя следующие основные документы: ведомость; пояснительную записку; схемы процессов проектирования, подсистем и средств обеспечения, спецификации компонентов всех видов обеспечения; смету затрат на создание САПР; ТЗ на разработку соисполнителями по отдельным компонентам; расчет ожидаемых технико-экономических показателей.

Рабочий проект (РП) создается на основании технического проекта организацией — разработчиком САПР.

На стадии РП проводят: разработку детальной структуры САПР, ее подсистем, взаимосвязи с другими системами и ее уточнение; построение алгоритмов и структурных схем автоматизированных процессов проектирования; формирование МО, ПО, ИО, ОО; разработку документации для монтажа, настройки и эксплуатации КСАП; создание проектов программ и методик испытаний и опытной эксплуатации; оформление и утверждение.

Результатом работ на этой стадии являются документ «Рабочий проект» и комплект эксплуатационных документов.

Рабочий проект содержит:

- ведомость РП; пояснительную записку; спецификацию — перечислены подсистемы, спецификации видов обеспечения; документацию технического обеспечения - приведены спецификация, комплект конструкторских документов;

- документацию ИО — включены спецификация, описание базы данных, инструкция по ее заполнению, инструкция по введению массивов данных;
- документацию ПО — приведены спецификация, тексты программ, описание программ, порядок и методика испытаний;
- документацию МО, МТО, ЛО — содержит спецификацию, пояснительную записку, описание языка, подсистемы .управления САПР;
- документацию ОО - даны спецификация, программа подготовки специалистов пользователей САПР, положение о службе САПР;
- программу и методику испытаний КСАП — содержит технические данные, подлежащие проверке при испытании компонентов САПР, порядок испытаний и методы их контроля;
- программу и методику опытного функционирования системы и подсистем — приведены данные, обеспечивающие получение и проверку проектных решений, выявление причин сбоев, показателей качества функционирования системы и подсистем;
- комплект эксплуатационных документов — содержит МТО, ТО, ПО и ведомость эксплуатационных документов.

Изготовление, отладку и испытания САПР проводят на основании документации РП. Они должны обеспечивать бесперебойную работу подсистем действующей САПР.

На стадии изготовления, отладки и испытаний выполняют: подготовку организации к вводу в действие САПР, включая обучение пользователей; изготовление и отладку компонентов ТО, ПО и ИО на машинных носителях; монтаж, отладку и испытания КСАП для САПР и всех ее подсистем; организацию работ по оформлению и утверждению актов сдачи компонентов и КСАП.

Работы на стадии ввода в действие САПР предназначены для обеспечения возможности промышленного функционирования САПР и определения технико-экономических показателей системы. На этой стадии проводят опытное функционирование, приемочные испытания, корректировку, доработку и выпуск эксплуатационных документов, промышленную эксплуатацию и развитие САПР.

Опытное функционирование САПР предполагает проверку работоспособности и взаимодействия подразделений организации с подсистемами САПР, выявление неисправностей и отказов, определение фактических технико-экономических показате-

лей системы и корректировку документации, устранение дефектов, оформление протокола.

Приемочные испытания включают в себя оценку качества разработки КСАП, проверку готовности организации к промышленному функционированию САПР, оценку качества опытного функционирования, проверку документации на САПР. Работа оканчивается вводом в действие системы, оформлением и утверждением протокола опытного функционирования и акта приемки САПР в промышленное функционирование.

Промышленную эксплуатацию выполняет служба САПР. Проектная организация для определения соответствия научно-техническому уровню проводит периодические испытания САПР. При этом оформляют заключение о соответствии подсистем и компонентов действующей САПР современным требованиям, определяют оценку эффективности и качества продукции, полученной с помощью САПР, записывают рекомендации о разработке или модернизации подсистем и компонентов проверяемой САПР.

Развитие САПР ведут путем модернизации подсистем, компонентов и средств обеспечения, а также ввода в эксплуатацию новых подсистем и компонентов.

8.4. Автоматизированные банковские системы управления

8.4.1. Характеристика банковской деятельности в РФ как предметной области информатизации

Банковская деятельность неразрывно связана с политическим и экономическим состоянием страны в целом. Парадокс последних лет функционирования банковской системы России заключается в том, что чем ниже падали общеэкономические показатели страны, тем более бурно шло развитие банковского сектора. Банковская сфера приносила самые высокие доходы. Банковские капиталы создавались на сравнительно простых операциях. Выдача краткосрочных кредитов в валюте при активном курсе доллара обеспечивала беспроигрышный результат. Возможность для населения защитить свои средства от инфляции давала приток сбережений физических лиц. Выгодны были валютнообменные операции с населением. В этот период появилось много банков, увеличивалось число отделений и филиалов. Экстенсивное развитие банковской системы сопровождалось массовой за-

купкой компьютеров, сетевого оборудования, созданием автоматизированных информационных технологий.

В экономике страны и в сфере банковской деятельности переломным стал 1995 год. Снижение торговой активности, затоваривание привело к резкому спаду краткосрочных банковских кредитов. Стабилизация курса рубля уменьшила активность населения в отношении валютнообменных операций. Начал разваливаться рынок межбанковских кредитов, уменьшаться банковская маржа из-за снижения доходности государственных ценных бумаг. Частные вкладчики в своем большинстве переориентировались на рублевые вклады, в результате у банков стали исчезать легкие способы получения прибыли. Однако время «тяжелых» инвестиционных способов увеличения прибыли, основанных на вложении в производство и другие долгосрочные проекты, еще не наступило. Именно в этот период банки беднеют, разоряются новых возникает гораздо меньше. В банках ужесточается режим экономии, который отражается и на формировании систем автоматизации обработки банковской информации.

Поток привлекаемой в банки денежной массы существенно уменьшился. Это обстоятельство заставило банки искать новые формы привлечения средств и наиболее эффективные решения по размещению активов. Меняются подходы банков в решении кадровых проблем, возникает необходимость в специалистах-профессионалах с практическим опытом, организаторскими способностями, деловыми качествами, имеющих базовые знания в области экономики, финансово-кредитной деятельности, маркетинга, менеджмента, экономико-математических методов и моделирования, способных к аналитическому мышлению и системному подходу, умеющих работать в среде автоматизации банковских услуг.

В условиях возрастающей конкуренции в банковской сфере и постепенного отказа от экстенсивных способов получения прибыли особое значение для деятельности банков приобретает анализ рыночной ситуации, выбор альтернативных решений. В банковской сфере все активнее реализуются новые виды услуг, такие, как фондовые и карточные операции, учет вкладов или населения, работа с драгоценностями, трастовые, ипотечные, торговые операции, инвестиционная деятельность, международные платежи и т. д. Многие банки предоставляют свои услуги по сети Internet, которая становится частью банковских технологий. Однако новые возможности банков порождают и новые проблемы.

Банковская сфера подвергается постоянным изменениям и развитию со стороны органов государственного управления,

Центрального банка, законодательных структур. Следить за изменением на финансовых рынках и вовремя на них реагировать российским банкам становится все труднее. Это связано прежде всего с тем, что средства банковской автоматизации, как правило, отстают от последних требований, вызванных быстрым расширением сферы деятельности банков.

Уровень и сущность банковских технологий в России, сложившиеся традиции банковского дела пока не соответствуют международному уровню. На российском банковском рынке практически нет устоявшихся технологий, что является одной из причин отсутствия магистрального направления автоматизации. Освоение западных финансовых рынков, постепенное сближение с мировыми стандартами характерно для банковской деятельности России в настоящее время.

Переход на новый план счетов — это наиболее значительный шаг в банковской реформе, которая продолжается уже несколько лет. В 1991—1997 гг. в условиях постоянных изменений правил проведения и учета банковских операций создавались почти все действующие автоматизированные банковские системы (АБС).

Ряд особенностей развития российской банковской деятельности наряду с ужесточением контроля со стороны Центрального банка создает для банковского бизнеса совершенно новые условия. Банки осваивают работу с новыми финансовыми инструментами, расширяют спектр оказываемых услуг, выходят на мировые рынки, реформируют бухгалтерский учет (приказ ЦБ РФ от 18 июля 1997 г. № 02—263) и др. В результате этой деятельности отечественная банковская система становится ближе к мировым стандартам. В связи с переходом на новый план счетов наиболее важные изменения касаются правил ведения бухучета, плана и структуры счетов, разрядности счетов. По этим направлениям вносятся изменения в существующие АБС.

8.4.2. Функциональные задачи и модули автоматизированной банковской системы

Автоматизация банковских технологий в новых рыночных условиях стала складываться в начале 1990-х годов, когда появились коммерческие банки. На рынке программных средств появляется ряд фирм-разработчиков, ведущие из них: «Инверсия», «Диасофт», «Асофт», «Програмбанк», «R-Style» и др. Развитие процесса автоматизации привело к предложению разнообразных банковских систем, что обусловлено не столько жесткостью фирм-разработчиков, но и различием самих банков по

выполняемым функциям, структуре, размерам и технологии банковского дела.

Создание и функционирование автоматизированных банковских технологий основывается на системотехнических принципах, отражающих важнейшие положения теоретической базы, которая включает ряд смежных научных дисциплин и направлений. К ним относятся экономическая кибернетика, общая теория систем, теория информации, экономико-математическое моделирование банковских ситуаций и процессов, анализ и принятие решений.

Банковские технологии как инструмент поддержки и развития банковского бизнеса создаются на базе ряда основополагающих принципов:

- комплексный подход в охвате широкого спектра банковских функций с их полной интеграцией;
- модульный принцип построения, позволяющий легко конфигурировать системы под конкретный заказ с последующим наращиванием;
- открытость технологий, способных взаимодействовать с различными внешними системами (системы телекоммуникации, финансового анализа и др.), обеспечивать выбор программно-технической платформы и переносимость ее на другие аппаратные средства;
- гибкость настройки модулей банковской системы и адаптация их к потребностям и условиям конкретного банка;
- масштабируемость, предусматривающая расширение и усложнение функциональных модулей системы по мере развития бизнес-процессов (например, поддержка работы филиалов и отделений банка, углубление анализа и т. д.);
- многопользовательский доступ к данным в реальном времени и реализация функций в едином информационном пространстве;
- моделирование банка и его бизнес-процессов, возможность алгоритмических настроек бизнес-процессов;
- непрерывное развитие и совершенствование системы на основе ее реинжиниринга бизнес-процессов.

Создание или выбор автоматизированных банковских систем связаны с планированием всей системной инфраструктуры информационной технологии банка.

Под *инфраструктурой АБС* понимается совокупность, соотношение и содержательное наполнение отдельных составляющих процесса автоматизации банковских технологий. В инфраструктуре кроме концептуальных подходов выявляются обеспечиваю-

щие и функциональные подсистемы. К обеспечивающим относят: информационное обеспечение, техническое оснащение, системы связи и коммуникации, программные средства, системы безопасности, защиты и надежности и др. *Функциональные подсистемы* реализуют банковские услуги, бизнес-процессы и любые комплексы задач, отражающие содержательную или предметную направленность банковской деятельности.

Создание автоматизированных банковских технологий помимо общесистемных (системотехнических) принципов требует учета особенностей структуры, специфики и объемов банковской деятельности. Это относится к организационному взаимодействию всех подразделений банка, которое вызывает необходимость создания многоуровневых и многозвенных систем (головной банк, его отделы, филиалы, обменные пункты, внешние структуры) со сложными.

Другой характерной особенностью банковских технологий является многообразие и сложность видов обеспечения автоматизации деятельности банка.

Автоматизированные банковские системы создаются в соответствии с современными представлениями об архитектуре банковских приложений, которая предусматривает разделение функциональных возможностей на три уровня (рис. 8.42).

Верхний уровень (front-office) образуют модули, обеспечивающие быстрый и удобный ввод информации, ее первичную обра-

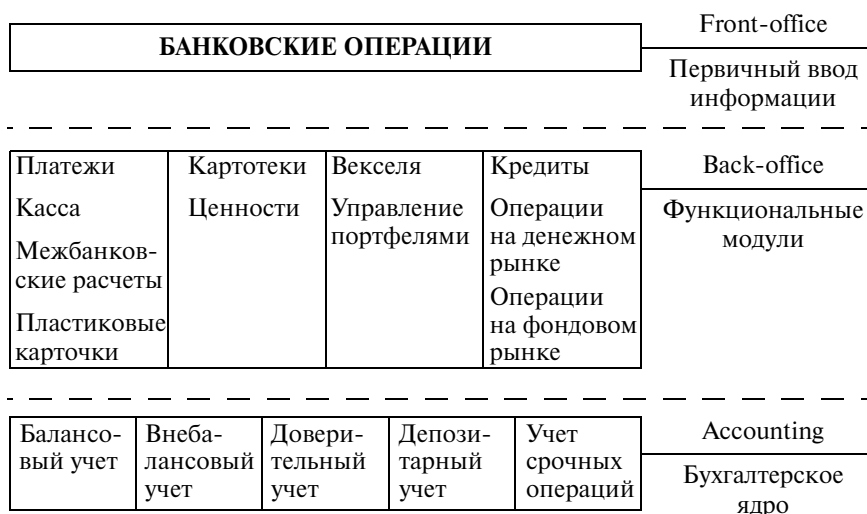


Рис. 8.42. Архитектура банковских учреждений

ботку к любое внешнее взаимодействие банка с клиентами, другими банками, ЦБ, информационными и торговыми агентствами и т. д.

Средний уровень (back-office) представляет собой приложение по разным направлениям внутрибанковской деятельности и внутренним расчетам (работу с кредитами, депозитами, ценными бумагами, пластиковыми карточками и т. д.).

Нижний уровень (accounting) — это базовые функции бухгалтерского учета, или бухгалтерское ядро. Именно здесь сосредоточены модули, обеспечивающие ведение бухгалтерского учета по всем пяти главам нового плана счетов.

Разделение банка на front-office и back-office основывается не столько на функциональной специфике обработки банковских операций (сделок) и принятия решений (обобщения и анализа), сколько на самой природе банка как системы, с одной стороны, фиксирующей, а с другой — активно влияющей на экономическое взаимодействие в финансово-кредитной сфере. Основные этапы создания АБС требуют:

- проведения функционального и информационного обследования деятельности банка;
- формирования требований к системе и их анализ;
- разработки структурно-функциональной модели банка;
- разработки информационной модели банка;
- детальной структуризации АБС, общесистемного проектирования, постановок задач;
- программирования, отладки, внедрения, эксплуатации, сопровождения.

Наработанный в России опыт создания АБС к настоящему времени имеет ряд особенностей. Охарактеризуем главные из них.

Прежде всего это ориентация системы на работу с проводками. Теперь явно прослеживается отход от чисто бухгалтерского построения АБС на основе операционного дня. Высокотехнологичная АБС — это эффективное средство контроля над настоящим и прогнозирование будущего развития финансово-кредитной деятельности банка. В любом западном банке такая система является жизненно необходимой составляющей в каждом звене банковской деятельности.

Сложился и реализуются при создании АБС и АИТ два концептуальных подхода к реализации банковских функций. Первый строится на узкой, ограниченной основе, которая воплощает ту или иную идею. Например, построение АБС ведется по принципу удовлетворения требований пользователей. Построение АБС на основе документооборота приводит к снижению эффективности банковской технологии в угоду этой идее. Без должного

уровня проработки и моделирования бизнес-процессов происходит механическое наращивание задач, функций, услуг. Противоположный подход основан на самостоятельном описании и построении пользователями с помощью средств АИТ банковских бизнес-процессов и документооборота. Этот подход отличается значительной трудоемкостью, усложняет настройку системы, нередко приводит к выхолащиванию банковской специфики. В результате система может по терять свою ценность для конечного пользователя.

Трудности выбора концептуальной основы заключаются в том, что на российском банковском рынке практически нет устоявшихся технологий. Автоматизация всегда отражает сложившийся технологический уровень банковского дела. Именно поэтому на выходе любых сколь угодно масштабных проектов комплексной АБС появляются системы на основе операционного дня и бухгалтерского учета.

Обращение к западным системам сопряжено с еще большими проблемами и трудностями: адаптация АБС к российской практике, неготовность банковского персонала к западным методам работы и т. д. Внедрение зарубежной АБС чревато полномасштабной внутрибанковской модернизацией. Напрашивается вывод о необходимости разумного сочетания разливных концептуальных подходов.

Расширение потребительских (функциональных) свойств АБС связано с отечественными особенностями развития банковского дела. Если возрастающие запросы банков на те или иные виды услуг носят массовый характер, то на рынке АБС появляются новые банковские приложения, новые классы банковских технологий.

Автоматизированная банковская система (базовый комплекс) позволяет организовать быстрое и качественное обслуживание клиентов по широкому спектру услуг. Основные функциональные модули системы реализуют:

- расчетно-кассовое обслуживание юридических лиц;
- обслуживание счетов банков-корреспондентов;
- кредитные, депозитные, валютные операции;
- любые виды вкладов частных лиц и операции по ним;
- фондовые операции;
- расчеты с помощью пластиковых карт;
- бухгалтерские функции;
- анализ, принятие решений, менеджмент, маркетинг и др.

АБС последнего (четвертого) поколения основаны на сетевой технологии в архитектуре «клиент—сервер», опираются на еди-

ные принципы построения и функционирования. Слабой стороной многих отечественных систем является недостаточная поддержка специфики банковского дела и его моделирования, недостаточное отражение предметной области. В последнее время стало уделяться больше внимания вопросам финансового анализа и целям управления бизнесом. Отсутствуют пока системы, позволяющие контролировать финансовые риски, управлять ресурсами, анализировать прибыльность операций, например, доходность банковской услуги (продукта), доходность клиента, доходность подразделения. Развитие банковского бизнеса приводит к необходимости использовать подобные инструменты в повседневной деятельности.

Остановимся на краткой характеристике основных функциональных подсистем АБС на примере разработок фирмы «Инверсия».

Операционный день банка как программно-технологический комплекс автоматизирует наиболее трудоемкие операции банковского учета (рис. 8.43). Все операции по лицевым счетам клиентов осуществляются по платежным документам, а выписка лицевого счета отражает каждую проводку. Комплекс реализует фактическое и планируемое движение средств по лицевым счетам (по неоформленным проводкам документов). Документы проходят операции последующего контроля, при совпадении всех параметров составляется опись документов и формируется файл для отправки в расчетно-кассовый центр (РКЦ). Документы, прошедшие через корреспондентский счет, разносятся по счетам.

Движение кассовых документов имеет свои особенности, главной из которых является связь с другими службами банка. Эта же особенность характерна и для внутренних проводок. При этом в системе ведется план счетов, каталог лицевых счетов банка, каталог клиентов банка, подводится баланс, выполняются служебные сервисные функции. Комплекс «Операционный день банка» имеет мультивалютные свойства.

Комплекс по учету деятельности филиалов предназначен для автоматизации рабочих мест в бухгалтерии и других подразделений филиала. На уровне «банк — филиалы» автоматизируется сбор, обработка и анализ информации, получаемой от филиалов, расчеты между всеми филиалами. Взаимодействие между филиалами может быть организовано напрямую либо через центральное отделение банка. В рамках филиала комплекс включает валютные, рублевые, кассовые операции, рассчитывает с учетом внутренних проводок баланс по филиалу, выполняет расчеты по корреспондентским отношениям.

Договорная подсистема работает с кредитными, депозитными, межбанковскими договорами, осуществляет валютный дилинг. Имеется возможность формировать тексты договоров, заводить условия договоров по выбранному шаблону. Условия договоров при изменении их статуса могут изменяться и проверяться на корректность. Реальное состояние договоров поддерживается путем ручного и автоматического исполнения режимов идентификации проводок по выдаче и возврату ссуд, гашению процентов и пеней. В журнале договоров отражаются данные по текущим, законченным, просроченным и тем договорам, у которых подошел срок платежей.

Программно-технологический комплекс-депозитарий фирмы «Инверсия» реализует следующие функции:

- формирование списка клиента с определением их типа (инвестор, дилер, эмитент, депозитарий, хранилище);

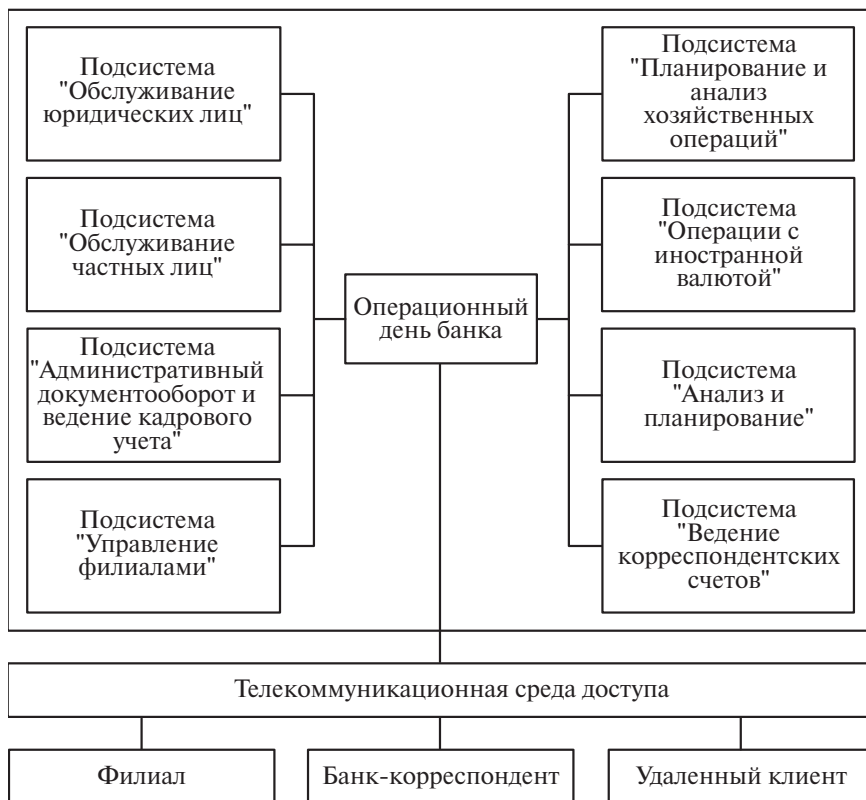


Рис. 8.43. Состав модулей операционного дня

- ввод типов и выпусков ценных бумаг;
- формирование депозитария, балансовых счетов и счетов депо, состоящих из кода клиента, кода ценной бумаги, кода места хранения;
- выполнение операций с ценными бумагами (прием и снятие с учета, смена владельца и места хранения и др.), подготовка выписок по счетам депо;
- ведение каталогов операций, размещение ценных бумаг, налогов и тарифов, подготовка сводной отчетности.

В качестве системы поддержки торгового процесса приведем программно-технологический комплекс «Биржевые операции» фирмы «Инверсия», который автоматизирует регистрацию контрактов и заявок на покупку-продажу валюты, контроль и анализ валютных платежей и показателей, статистику торгов валютой, генерирует отчетную документацию.

Комплекс «Операции на рынке ГКО» (государственные краткосрочные облигации) является самостоятельной системой регистрации. В нем реализуются следующие основные функции:

- заведение счетов депо для банка дилера, его клиентов, филиалов банка с их реквизитами, типом обслуживания, процентными ставками;
- учет операций купли-продажи ценных бумаг, перевода на другие счета, начисление комиссий и налогов по операциям с ГКО;
- переоценка портфелей ценных бумаг по результатам торгов, расчет доходов банка и его клиентов;
- формирование проводок, журнала учета операций, выписок по счетам, сводных отчетов;
- анализ доходности ценных бумаг по ряду показателей.

Уровень автоматизации фондовых технологий определяется потребностями и финансовыми возможностями участников рынка ценных бумаг, развитием рынка, его правовой основой, степенью риска, защищенностью и доверием инвесторов.

Наиболее распространенными подсистемами АБС по обслуживанию клиентов являются: клиент-банк, работа с пластиковыми картами, операции обменного пункта и др.

Программно-технологический комплекс «Клиент — банк» фирмы «Инверсия» состоит из модулей «банк» и «клиент», которые устанавливаются на коммуникационных ПЭВМ в банке и в организациях клиента. Клиенту предоставляется возможность проводить стандартные банковские операции, не покидая офиса. Комплекс выполняет обычно функции взаимодействия по отправке и получению платежных документов, получению выписок по сче-

там, заявкам на продажу-покупку валюты, операциям с ценными бумагами и для получения справочного материала.

Эффективной формой обслуживания клиентов является использование пластиковых карт. Пластиковые карты по виду обеспечения делятся на дебитные, кредитные, дебитно-кредитные. Наряду с этим существует деление пластиковых карт на основе технической реализации их функций: магнитные карты, смарт-карты, лазерные карты, микропроцессорные (smart-карты). Наиболее передовой и дорогой технологией является работа с микропроцессорными картами. Торговые точки оснащаются соответствующими программно-техническими средствами (торговыми терминалами).

Составной частью банковских услуг являются банкоматы (автоматы-кассиры). Они могут быть расположены как в помещении банка, так и вне банка, могут выдавать наличные деньги, производить перевод денег и другие операции. Использование таких автоматов делает обслуживание клиентов более гибким. Услуги приближаются к клиентам, расширяются временные и пространственные рамки, сокращается персонал.

Комплекс «Обменный пункт» автоматизирует выполнение операции по покупке, продаже валюты, дорожных чеков и других сопутствующих операций (например, неторговых). В рамках этой подсистемы ведется каталог валют, контролируется наличный состав валюты в кассе, оформляются отчетные документы. Заключительной процедурой операционного дня после сведения остатков по кассе является формирование данных для разности средств по счетам клиентов. Разноска осуществляется по технологии валютного операционного дня.

8.4.3. Информационное обеспечение АБС

Информационное обеспечение, его организация определяются составом объектов отражаемой предметной области, задач, данных и совокупностей информационных потребностей всех пользователей автоматизированной банковской системы.

Информационное обеспечение, немашинное и внутримашинное, включает полный набор показателей, документов, классификаторов, файлов, баз данных, баз знаний, методов их использования в банковской работе, а также способы представления, накопления, хранения, преобразования, передачи информации. Принятые в конкретной системе для удовлетворения любых информационных потребностей всех категорий пользователей в нужной форме и в требуемое время. Ведущим направлением организации

внутримашинного информационного обеспечения является технология баз и банков данных. К организации информационного обеспечения банковской деятельности предъявляется ряд требований. Наиболее важными из них являются: обеспечение для многих пользователей работы с данными в реальном времени; предоставление для обмена информацией возможности экспорта/импорта данных в разных форматах; безопасность хранения и передачи банковской информации; сохранение целостности информации при отказе аппаратуры.

Информационным фундаментом современных АБС для многопрофильных и многофилиальных банков с возможностями анализа и прогноза являются БНД со структурой данных в базе, обеспечивающей надежное хранение, а также быстрый доступ к различным временным показателям по множеству объектов (документам, счетам, клиентам, филиалам и их группам). Для этого требуются многомерные структуры данных. Появление новых структур и видов (мультимедийные и пр.) данных требует совершенствования технологии баз данных.

Развитие теории и практики создания и использования баз данных приводит к более широкому понятию — хранилище данных. Это может быть централизованная база данных, объединяющая информацию из разнородных источников и систем и предоставляющая собранные данные по приложениям конечных пользователей.

Единое информационное пространство, созданное на основе технологий информационных хранилищ, служит базой для реализации разнообразных аналитических и управленческих приложений. К ним можно отнести оценку кредитных и страховых рисков, прогноз тенденций на финансовых рынках, выявление махинаций с кредитными картами и многое другое.

Концепция хранилища данных означает построение такой информационной среды, которая позволит осуществлять сбор, трансформацию и управление данными из различных источников с целью выработки решений по управлению банком, создаст новые возможности по привлечению прибыли.

По мере того, как преимущества хранилищ данных становились все очевидней, увеличилось число их версий и объем содержащихся в них данных. Самым главным требованием клиента к хранилищу является возможность для конечных пользователей вести работу в диалоге по полному набору бизнес-данных и получать ответы в приемлемые временные промежутки. Объем данных должен быть таким, который необходим для поддержки бизнеса. По мере роста объема информации хранилище

должно отвечать требованиям устойчивой производительности. Для обеспечения производительности и управляемости хранилища могут использоваться различные программные продукты и инструменты, автоматизирующие обработку информации и предоставляющие возможность рассматривать банковский бизнес в разных аспектах, что оправдывает любые издержки, связанные с разработкой системы.

Центр тяжести информационного обеспечения современной АБС приходится на полноту отражения специфики. Предметной области банковского бизнеса. Степень развития этой специфики нагляднее всего проявляется в словаре информационной модели. Если пользовательский интерфейс в системе (меню, экранные формы, отчеты и т. д.) охватывает предметную область наиболее полно (по количеству и объему понятий, объектов, процессов), это свидетельствует о близости автоматизированных информационных технологий. К реальным задачам банка. Широкий словарь профессиональных терминов, отражающих данную область деятельности и выведенный для общения с пользователем (пользовательский интерфейс), характерен для АБС высокого уровня и помогает специалистам банка и управляющим принимать стратегические и тактические решения.

Закладываемая в основу АИТ информационная модель должна отражать разнообразие понятий, их назначение, взаимосвязи, давать описание характерных сущностей, применяемых в банковской предметной области. К ним относятся такие понятия, как документ, операция, клиент, финансовый инструмент, счет и план счетов, банковский продукт (услуга), пользователь (рис. 8.44). Для этих целей разрабатывается стандарт на описание базовых понятий (сущностей), который включает список реквизитов и операций (алгоритмов) по каждому понятию, а также форму описания бизнес-процессов как функциональных моделей банка. Стандарт содержит формализованное описание всех действий (алгорит-

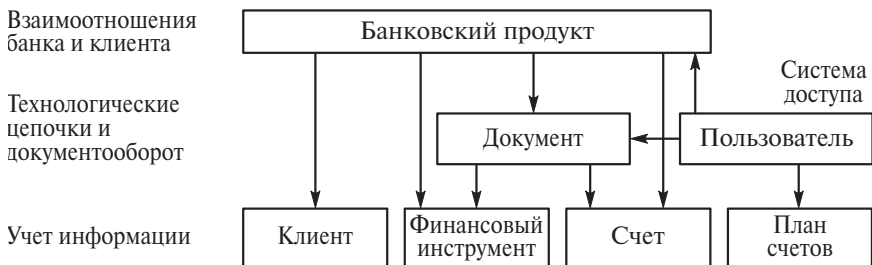


Рис. 8.44. Состав базовых понятий

мов), которые происходят при предоставлении клиентам услуг, при внешней и внутренней работе банка.

Одним из новых направлений в создании автоматизированных банковских систем является объектно-ориентированный подход в информационных технологиях.

Подход основан на объединении данных и процедур в рамках понятия «объект» в отличие от распространенного структурного подхода, при котором данные и процедуры определяются и реализуются отдельно.

Важным новшеством современных банковских систем являются подходы к проектированию информационного обеспечения, позволяющие специалистам банка самостоятельно модифицировать и дополнять словарь информационной модели и терминах банковских продуктов или услуг, предоставляемых клиентам. Среди множества понятий, например, первичным является понятие документ (договор, заявка, ордер, распоряжение и т. д.). Документ порождает последовательность действий, которые должны быть выполнены по его исполнению, т. е. операции. Наряду с простыми могут создаваться и сложные операции, формироваться новые услуги — действия, которые необходимо выполнять и отражать их последовательность.

Для расширения предметной области баз данных начато использование механизма гибких классификаторов. Они позволяют наделять объекты новыми признаками (атрибутами) и дополнять базы данных нестандартными свойствами объектов. Например, дополнительно могут быть введены адрес налоговой инспекции, данные о загранпаспорте, группах банков, клиентов, операций. Кроме того, гибкие классификаторы позволяют организовать нужные группировки объектов, удобно использовать их при формировании отчетов, поиске и переносе информации.

8.4.4. Программное обеспечение информационных технологий в банках

Банк, являясь развивающимся во времени объектом, порождает как количественные изменения информационной среды (увеличение объемов обрабатываемых данных, числа пользователей и пр.), так и качественные (расширение спектр; решаемых задач, изменение их характера). Адекватно построенная автоматизированная система должна на протяжении некоторого времени обеспечивать подобное развитие без проведения радикальных модификаций. Потенциальные возможности системы (или фундаментальные ограничения), как правило, сильно связаны с воз-

возможностями базовых программных средств, на основе которых она разработана. Поэтому первая группа требований к автоматизированной банковской системе (АБС — это требования к базовому программному обеспечению (операционной системе, СУБД, средствам автоматизации программирования и др.).

Вторая группа требований к АБС характеризует качество прикладной части как собственно банковской системы, ее потребительские свойства. К ним следует отнести широту функционального диапазона, удобство интерфейса. Подобное качество системы связано с профессионализмом конкретных разработчиков и может сильно варьироваться в рамках одних и тех же базовых программных средств.

В последние годы в банковской деятельности обострилась проблема обеспечения безопасности данных. Она включает в себя несколько аспектов. Во-первых, это гибкая, многоуровневая и надежная регламентация полномочий пользователей. Ценность банковской информации предъявляет особые требования к защите данных от несанкционированного доступа, в том числе к контролю управления процессами, изменяющими состояние данных. Во-вторых, наличие средств для поддержания целостности и непротиворечивости данных. Подобные средства подразумевают возможность осуществления контроля вводимых данных, поддержки и контроля связей между данными, а также ввода и модификации данных в режиме транзакций — набор операций, обеспечивающих поддержание согласованности данных. В-третьих, присутствие в системе многофункциональных процедур архивации, восстановления и мониторинга данных при программных и аппаратных сбоях.

Отличительной чертой функционирования автоматизированных банковских систем являются необходимость обработки больших объемов данных в весьма сжатые сроки, что предъявляет требования к производительности. При этом основная тяжесть падает на операции ввода, чтения, записи и передачи данных. Все это предъявляет весьма жесткие требования к производительности СУБД и средств, обеспечивающих передачу данных. Кроме того, значительные объемы информации должны быть доступны специалисту банка в оперативном режиме для обеспечения возможностей анализа, прогнозирования, контроля и пр. Поэтому базовые средства должны быть в состоянии поддерживать доступ к большим и постоянно возрастающим объемам данных без потери производительности.

Поддержание производительности в условиях возрастания и нагрузки на систему зачастую приводит к необходимости перехо-

да на более мощную аппаратную платформу (свойства «переносимости»). Поэтому базовые средства должны обеспечивать возможность переноса прикладной системы на новые аппаратные платформы без каких-либо глубоких изменений прикладного программного обеспечения.

Непременным атрибутом современной системы должно быть наличие в базовых средствах *сетевых функций*, обеспечивающих возможность объединения различных программных платформ (DOS, NetWare, Windows NT, UNIX и пр.) и как следствие возможность гибкого расширения и наращивания системы пополнения ее новыми рабочими местами и новыми серверами различных классов.

Основными свойствами АБС в отношении прикладных, потребительских свойств являются: достаточная широта функционального набора, интегрированность, конфигурируемость, открытость и настраиваемость системы.

Возможность приобретения систем различных конфигураций представляет заметный интерес для пользователей. Поэтому при рассмотрении системы важно учитывать такие ее характеристики, как набор модулей и реализуемых ими функций, степень автономности модулей, наличие межмодульного взаимодействия и формы его реализации (почта между модулями, пересылка управляющих сообщений и пр.), возможные конфигурации системы, ее минимальный состав, независимо функционирующие части, варианты расширения.

Открытость системы предполагает наличие в ней средств для развития и модификации: CASE-средства, SQL-средства (языков четвертого поколения), шлюзов для привязки внешних (написанных не разработчиком) процедур, генераторов запросов, процедур импорта/экспорта данных. Потребность во внесении изменений возникает в банке достаточно часто. Однако привлечение фирмы-разработчика для осуществления модификаций не всегда оказывается возможным. Препятствиями могут быть срочность внесения изменений, финансовые ограничения, соображения коммерческой тайны (если речь идет об отражении в системе элементов новой технологии), географическая удаленность от разработчика и пр. В этих условиях только открытость способна обеспечить развитие системы собственными силами и поддержание соответствия компьютерных систем изменяющимся условиям.

Настраиваемость системы необходима для адаптации к технологии конкретного банка. Необходимость настройки обычно возникает при установке АБС в банке, но может быть и следствием технологических изменений в операциях банков. В последнем слу-

чае настраиваемость непосредственно граничит с открытостью АБС. Настраиваемость предполагает наличие системе набора параметров и шаблонов — шаблонов операций, договоров и других текстовых документов. Кроме того, настраиваемость предполагает возможность процедурной настройки системы: регламентацию прав пользователей, конфигурирование рабочих мест, определение набора процедур при открытии и закрытии операционного дня.

Большинство эксплуатируемых в настоящее время банковских систем являются DOS-комплексами, функционирующими либо в локальном режиме, либо в сетевом (в конфигурации «файл-сервер»). В то же время на рынке АБС все отчетливее обозначается присутствие систем на основе многоплатформных СУБД (обычно не вполне правильно называемых банковскими UNIX-системами). И хотя общее количество установок подобных комплексов не превысило и сотни, UNIX-системы считаются весьма перспективной ветвью АБС.

Однако при сопоставлении возможных решений (DOS, UNIX, Windows NT и пр.) зачастую упускается из виду ценовой аспект проблемы, связанный с затратами потребителя на приобретение и сопровождение системы: стоимость базовых программных средств, вычислительной техники, цена собственно системы, оплата персонала. Считается, что по этим позициям DOS-системы существенно привлекательнее для значительной части малых и средних банков.

Следует учитывать и традицию: программно-технические системы DOS/IBM PC доминируют в нашей стране уже многие годы (и не только в банковской сфере), поэтому переход к иным решениям требует серьезной мотивации. Однако последняя в большинстве случаев как раз и отсутствует. Дело в том, что, несмотря на очевидные принципиальные недостатки, DOS-комплексы довольно успешно справляются с поставленными перед ними задачами в большом числе банков. Рассмотрим претензии, обычно предъявляемые DOS-системам.

1. *Недостаточная производительность*, невозможность поддержки больших объемов данных. Дело в том, что основная часть коммерческих банков весьма молода. Объем накопленных за это время данных и текущий документооборот обычно не настолько велики, чтобы не соответствовать системам DOS в отношении как поддержки, так и производительности. Однако в тех случаях, когда решается вопрос совершенствования АИТ, качество системы может быть улучшено путем замены сервера сети на более мощный, что обеспечивает системе запас прочности на достаточно долгий срок.

Ситуации, в которых персональные СУБД выходят на фундаментальные ограничения, характерны в настоящее время для относительно небольшого числа крупных банков. В частности, серьезные проблемы у DOS-систем возникают при необходимости поддержки таблиц, включающих миллионы записей (и именно это может потребоваться при обслуживании физических лиц - акционеров, вкладчиков), при увеличении числа рабочих мест в сети до ста и выше (архитектура «файл-сервер» порождает большой график).

2. *Невозможность обеспечения безопасности данных на должном уровне.* Эта проблема осознается большинством банков, но при этом зачастую она просто игнорируется. Решающим фактором служат стоимостные соображения, а проблема безопасности данных частично устраняется организационно-техническими мерами: установкой источника бесперебойного питания, соблюдением регламента системных работ, персональным контролем за использованием вычислительных средств и пр. Острота ситуации во многом ослабляется неразвитостью систем телекоммуникаций в нашей стране, в силу чего проблемы, связанные с «электронным взломом», пока не слишком актуальны.

3. *Ограниченность архитектурных средств.* Эта проблема возникает, когда для реализации тех или иных банковских операций необходимо наличие нескольких взаимодействующих и протяженных во времени процессов. В рамках DOS'а задача обычно решается выделением под каждый процесс станции локальной сети. Подобное решение имеет очевидные ограничения. Однако на начальном этапе развития банка сложные операции выполняются не слишком часто, поэтому данный недостаток не очень ощутим.

Таким образом, недостатки DOS-комплексов способны проявиться на этапе перехода банка в класс выше среднего. В то же время DOS-системы покрывают сегодняшние потребности многих малых и средних банков, являясь приемлемым компромиссом малой стоимости и ограниченных возможностей.

В качестве ступени, следующей за DOS-комплексами, можно рекомендовать системы, построенные в архитектуре «клиент-сервер» в рамках Novell NetWare. Предложенные фирмой Novell средства прикладных разработок (AppWare) позволяют в ближайшее время существенно повысить качество таких систем. Следует отметить также готовность отечественных потребителей к решению проблемы в рамках Novell NetWare, обусловленную как парком имеющейся вычислительной техники, так и опытом и специализацией работающих в банках программистов. Можно ожидать,

что автоматизированные компьютерные системы подобного типа станут достаточно массовыми в ближайшее время и сохранят это положение надолго.

Что касается АБС, разработанных на основе многоплатформенных СУБД, то они практически свободны от обсуждавшихся выше проблем, но сталкиваются с преградами иного, ценового, плана. Стоимость базовых программных средств и требуемой для их эффективного функционирования вычислительной техники может на порядок превышать аналогичные показатели для рассмотренных выше решений. Поэтому потребителями АБС данного класса в ближайшем будущем, несомненно, будут крупные и, возможно, некоторая часть средних банков.

От быстрой разработки пользовательских приложений (прикладных программ) выигрывает любая область бизнеса, но для банков это особенно актуально. В первую очередь это относится к фондовым операциям, при которых фактор времени имеет решающее значение. Средства работы с новыми финансовыми инструментами должны разрабатываться быстро. В области автоматизированной разработки программного обеспечения существуют разнообразные подходы и средства. Они способствуют развитию новых форм банковских услуг и бизнеса. Из новых языковых средств такого рода следует обратить внимание на средство Java. Java — это не зависящая от аппаратных платформ и операционных сред объектно-ориентированная языковая среда для программирования компактных сетевых приложений. Java позволяет работать на больших компьютерах, подключаться к сети простыми техническими средствами, предоставляет возможность работы Java-приложений как в сети Internet, так и в корпоративных сетях.

Если техническое обеспечение АБС в России, как правило, полностью зарубежное, то в программном обеспечении доля зарубежных систем значительно меньше. На отечественном рынке программных средств действуют несколько десятков поставщиков. Кроме того, ряд банков (около 50%) разрабатывают собственное программное обеспечение. Качественная эволюция деятельности банков, их возрастающие требования и финансовые возможности будут развивать и направлять подходы к организации программного обеспечения банковских технологий.

8.4.5. Техническое обеспечение

На уровне технического обеспечения банковские технологии должны строиться на современных требованиях к архитектуре аппаратных средств. К ним относятся: использование разнооб-

разных телекоммуникационных средств связи, многомашинных; комплексов, архитектуры «клиент — сервер», применение локальных, региональных и глобальных скоростных сетей, унификация аппаратных решений.

Количество и состав используемых технических средств определяется интенсивностью и объемами информационных потоков, режимами работы и временными особенностями реализации функций банковской системы. Рост состава и объемов банковских услуг, числа филиалов, клиентов и связей заставляет банки приобретать более мощные компьютеры и более развитое техническое обеспечение. Получили распространение сетевые банковские технологии. Сетевой парк становится все более разнообразным. Следует отметить и ускоренное развитие средств межбанковской телекоммуникации.

Основой современного подхода технических решений в построении информационных технологий банков является *архитектура «клиент—сервер»*. Она предполагает организацию технического обеспечения и разделения обработки информации между двумя компонентами, которые называются клиентом (рабочей станцией) и сервером. Обе части выполняются на разных по мощности компьютерах, объединенных сетью. При этом клиент посылает серверу запросы, а сервер их обслуживает. Такая технология реализуется в профессиональных СУБД, имеющих специальный язык структурированных запросов.

Одним из вариантов реализации технологии «клиент—сервер» является ее трехуровневая архитектура. В сети должны присутствовать как минимум три компьютера: клиентская часть (рабочая станция), сервер приложений и сервер базы данных. В *клиентской части* организуется взаимодействие с пользователем (пользовательский интерфейс). *Сервер приложений* реализует бизнес-процедуры для клиентской части. *Сервер базы данных* обслуживает бизнес-процедуры, которые выступают в роли клиентов. Гибкость такой архитектуры в независимом использовании и замене вычислительных и программных ресурсов.

Для повышения надежности, отказоустойчивости технических решений в банковских АИТ практикуется объединение серверов в группы (кластеры). При этом ресурсы и нагрузка разделяются между серверами (узлами системы) так, что пользователь не знает, с каким конкретным сервером он работает в данный момент, а использование технических средств оказывается более эффективным.

Телекоммуникационная архитектура в автоматизированных технологиях банка определяет набор и структуры подсистем тех-

нического обеспечения, которые должны обеспечивать разнообразные типы взаимодействия для всех приложений (модулей) АБС (рис.8.45). Возможности архитектуры в процессе создания АИТ согласуются с требованиями и условиями работы банка, определяемыми его бизнес-процессами. Предусматривается взаимодействие банка с внешними финансовыми и информационными структурами, с расчетно-клиринговыми палатами и центрами, биржами, РКЦ, с удаленными клиентами и другими банками и т.д. Телекоммуникационное обеспечение бизнес-процессов банка строится с учетом обслуживания свое и корпоративной сети и доступа в любые другие локальные и глобальные сети. Из-за отсутствия на текущем этапе стандартов на прикладные взаимодействия обычно связь банка с внешними организациями осуществляется через шлюзы, например, почту, телекс или АРМы других фирм.

Корпоративные сети того или иного банка выступают в качестве транспортной основы, на которой строится вся телекоммуникационная архитектура. В данной области существуют множество решений для линий любого качества, включая защиту транспортного уровня и управление им. Реализация функционально полного набора банковских телекоммуникации позволяет создавать единое информационное пространство. Возможно осуществлять интеграцию отделений филиалов во всех приложениях банковских услуг, обеспечивать доставку услуг в любое место востребования и в любое время, в онлайн-овом и офлайн-овом режимах (непосредственной и регламентируемой связи).



Рис. 8.45. Телекоммуникационная архитектура

Итак, телекоммуникационные системы позволяют банку решать Важнейшие задачи автоматизации — от чисто технических как например, обеспечение оптимальной производительности и прозрачности взаимодействия бизнес-процессов, до функциональных на наивысшем уровне банковского обслуживания.

Применение локальных, региональных и глобальных сетей в АБС предъявляет повышенные требования к их надежности, а также защите и целостности данных. Уровень готовности и отказоустойчивости сетевых средств должен быть высоким, чтобы исключить возможность нарушения работоспособности при выходе из строя одного из сетевых компонентов. Например, при организации взаимодействия с удаленными филиалами, пользователями надо предусматривать возможность перехода на дополнительные коммутируемые линии, дублирование основного канала связи или увеличение его пропускной способности.

Важным фактором, позволяющим сократить стоимость технической поддержки сети, является внедрение централизованной системы сетевого управления. Она предоставляет возможность дистанционного конфигурирования, контроля, устранения неисправностей и реализации ряда других функций. Интеграция технологий одного производителя сетевого оборудования, предоставляющего полный набор коммуникационных устройств (концентраторов, коммутаторов, маршрутизаторов) упрощает управление, администрирование, подготовку персонала, снижает суммарную стоимость оборудования, а также повышает эксплуатационную надежность системы в целом.

Модернизация сетевых инфраструктур играет существенную роль в процессе расширения банковских услуг, выхода банка на новые рынки. Весьма важным является внедрение мультипротокольных сетей межбанковского взаимодействия, которые позволяют организовать наиболее эффективный обмен информации.

Опыт показывает, что самым слабым звеном в вычислительных комплексах банков являются серверы. Наиболее перспективные из них — Unix-серверы. Серверы более низкого уровня, например IBM PC-серверы, требуют решений по расширению дисковой подсистемы, по предсказанию сбойных ситуаций и т. п.

Снижение стоимости компьютеров не приводит к сокращению расходов банка на поддержание информационной инфраструктуры. Увеличение числа автоматизированных рабочих мест, повышение их интеллектуального уровня, использование более сложного состава и более дорогих программных среде в требуют дополнительных затрат. Стоимость владения системой — это показатель затрат на установку и поддержку компьютерной систе-

мы, и он примерно в шесть раз больше начальной цены приобретения. Стоимость владения складывается *min* расходов на установку программного обеспечения на компьютеры, включение их в сетевую среду, обеспечение совместимости, настройку операционной системы и многое другое из обязательных и неотложных задач, из которых складывается рабочее время системных специалистов по поддержанию и сопровождению комплекса технических средств.

8.4.6. Учет межбанковского взаимодействия при создании АБС

Банковская система призвана обеспечить эффективность расчетов между хозяйствующими субъектами. Значительная часть расчетов носит межбанковский характер и служит для экономических связей финансово-кредитных органов. Межбанковские расчеты сопровождают различные виды внешнеэкономических связей. Банки между собой устанавливают корреспондентские отношения на договорной основе, когда для осуществления платежей и расчетов операции ведутся одним банком по поручению и за счет другого банка. К настоящему времени способы осуществления межбанковских платежей не удовлетворяют современным требованиям.

Одним из видов корреспондентских отношений являются расчеты со взаимным открытием корреспондентских счетов коммерческих банков, открытых главным образом в региональных учреждениях Центрального банка Российской Федерации (ЦБ РФ) — расчетно-кассовых центрах, региональных главных управлениях (РГУ) ЦБ РФ. По существу РКЦ — это отделения ЦБ РФ, являющиеся элементами платежной системы, главной функцией которых является перевод денежных средств. Иными словами, РКЦ служат посредниками в платежах и кредитах между коммерческими банками (КБ).

По способу организации работы банка через корсчет в РКЦ ЦБ все банки можно разделить на две большие группы. Первая группа — это банки, использующие технологию так называемых Прямых расчетов (региональных, ускоренных).

Технология прямых расчетов позволяет обеспечить прохождение платежей между банками—участниками этих расчетов в течение одного операционного дня. В наиболее полном виде эта технология реализована в системе электронных платежей, которая эксплуатируется в банках Украины. Все платежи, поступающие в банк в электронном виде, передаются несколько раз в день, что

позволяет более эффективно использовать финансовые ресурсы банка. В России такая технология реализована лишь на региональном уровне и, более того, не во всех регионах, поэтому российские банки могут вести прямые расчеты с банками только своего региона. Межрегиональные расчеты осуществляются проводками по корсчету аналогично работе банков второй группы. Вторая группа — это банки, не использующие технологию «прямых» расчетов и работающие только через корсчет. Выбор банком способа работы по корсчету определяется в основном возможностями, предоставляемыми банкам региональными вычислительными центрами. При этом банк учитывает все преимущества и недостатки того или иного способа.

Проблема комплексной автоматизации межбанковских расчетов в России стоит сегодня как никогда остро. Это связано, в первую очередь, с необходимостью ускорить прохождение платежных документов по инстанциям. Продуманная, а главное, реализованная концепция автоматизации позволит высвободить значительные суммы денег, никак не используемых во время нахождения в пути.

Центробанком России предпринимаются меры по формированию своей расчетной системы с применением новых алгоритмов учета и обработки информации на базе электронных систем перевода денежных средств.

Идеальным был бы вариант прямых корреспондентских отношений между банками, при которых благодаря АИТ каждый платежный документ отправляется от банка-отправителя к банку-получателю. Внедрение такой системы предполагает стыковку транспортного уровня с функционирующими в банках разнообразными автоматизированными банковскими системами.

В условиях отсутствия возможностей быстрой модернизации системы межбанковских расчетов Банка России создание банками альтернативных межбанковских систем — безусловно, правильный путь. В перспективе желательно объединение этих систем под контролем Центрального банка с участием как можно большего круга коммерческих банков по типу Федеральной банковской резервной системы США.

Фирмы — разработчики автоматизированных банковских расчетов прикладывают значительные усилия для ее решения. Однако немало зависит и от самих банков, ибо любая система разрабатывается под конкретного клиента. Автоматизация банковских расчетов — основа для деятельности банка и необходимый фактор его конкурентоспособности. Это понимает большинство руководителей банков. Но при существующем в данный

момент многообразии программного обеспечения для автоматизации всех видов деятельности банка возникает проблема унификации и выработки единых стандартов для новых программных продуктов и мер по стыковке уже созданных.

Фирмы-разработчики понимают, насколько важно обеспечить надежную и удобную стыковку между программами разных фирм. Ведущие фирмы—разработчики банковских систем предложили разработать единый формат для обмена данными между различными банковскими системами. В проекте разработки единого формата участвуют московские фирмы «Асофт», «Диасофт», «Инверсия», «R-Style», «Интербанксервис», «Програмбанк» и др. Однако сложность проблемы стыковки множества работающих разнородных банковских систем такова, что к внедрению предлагается несколько проектов.

Пока фирмы, тиражирующие свои системы, трудятся над решением проблемы их стыковки, в России появляются центры корреспондентских счетов, по логике работы весьма напоминающие клиринг. Это собственные клиринговые системы крупных коммерческих банков и межбанковские расчетные палаты. В частности, имеются примеры реализации безбумажной технологии обработки платежных документов за счет применения алгоритмов криптографической шифровки информации (в том числе электронной подписи), которые по сути являются клиринговыми центрами для отделений и филиалов. Все указанные банки работают на собственных автоматизированных системах.

Преимущество клиринговых центров и расчетных палат заключается в том, что они строятся на принципе зависимости от коммерческих банков (своих учредителей), которые готовы нести как юридическую, так и экономическую ответственность перед участниками расчетов. Кроме того, с апробированием клиринговых центров происходит формирование цивилизованного рынка свободных капиталов.

Процесс создания независимых расчетно-клиринговых палат происходит сложнее, чем создание таких же центров при банках. Из всех палат реально функционирует (имея необходимые лицензии Банка России — техническую и банковскую) только одна — Центральная расчетная палата (ЦРП). У палаты сейчас свыше 200 банков-корреспондентов. При этом доля прямых проводок «Банк — Банк» по электронной системе через ЦРП составляет 30—40%. Большая часть окончательных расчетов идет через РКЦ.

В перспективе с увеличением числа банков — участников палаты, прохождение через систему ЦБ будет сведено к минимуму, а расчеты будут осуществляться день в день.

Таким образом, все более актуальной становится необходимость выработки заинтересованными организациями (Центральным банком, банками, клиринговыми центрами, расчетными палатами, фирмами—разработчиками программного обеспечения) единой концепции автоматизированной системы расчетов в России с учетом международных стандартов, рекомендованных Банком международных расчетов (Швейцария).

В связи с имеющимися трудностями расчетов между банками страны важными в настоящее время представляются разработка и эксплуатация межбанковских электронных сетей и возможность их подключения к общей сети ЦБ РФ. Это позволит ускорить расчеты между коммерческими банками, повысить достоверность передаваемой информации.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Раскройте укрупненную схему организационной структуры среднего предприятия.
2. Дайте краткую характеристику функциональным подсистемам АСУП.
3. Как Вы себе представляете функционирование подсистемы оперативного управления основным производственным процессом промышленного предприятия?
4. Что такое технологический объект управления?
5. Перечислите характерные особенности АСУТП.
6. По каким признакам классифицируются АСУТП?
7. В чем заключается горизонтальная и вертикальная декомпозиция автоматизированного технологического комплекса?
8. Объясните термин «интегрированная АСУ».
9. Раскройте содержание понятия «гибкие производственные системы».
10. В чем заключается системный подход при проектировании АСУТП?
11. Перечислите порядок разработки АСУТП.
12. Возможно ли применение САПР при проектировании АСУТП?
13. Каким образом реализовано автоматизированное управление ходом технологического процесса в АСУТП вращающихся печей?
14. Какое программное обеспечение и какие технические средства применяются в современных АСУТП?
15. Расскажите, как Вы понимаете систему числового программного управления технологическим оборудованием в АСУТП.
16. Приведите пример АСУТП мониторинга и управления технологическими процессами сложного производства.
17. Раскройте особенности структуры, принципов построения и технических средств АСУТП высокоопасного ТОО.

18. Какие Вы знаете характерные этапы и стадии процесса проектирования новых объектов?
19. В чем заключается концепция блочно-иерархического подхода к проектированию новых объектов?
20. Дайте классификацию САПР.
21. Охарактеризуйте виды обеспечения САПР.
22. Как классифицируются математические модели технических объектов в САПР?
23. Что такое микро-, макро- и метауровни математических моделей объектов?
24. Расскажите о требованиях, предъявляемых к математическим моделям в САПР.
25. В чем состоит информационное обеспечение в САПР?
26. Объясните состав программного обеспечения САПР.
27. Каким образом организовано техническое обеспечение САПР?
28. Перечислите технические средства машинной графики САПР.
29. Раскройте содержание интерактивной системы автоматизированного проектирования диалог «конструктор – ЭВМ».
30. Расскажите о стадиях создания САПР.
31. Каковы функциональные задачи и модули автоматизированной банковской системы?
32. В чем состоит информационное обеспечение АБС?
33. Дайте характеристику программному обеспечению банковских информационных технологий.
34. Перечислите основные технические средства АБС.
35. Как учитывается межбанковское взаимодействие в АБС?

Глава 9

ОБЕСПЕЧИВАЮЩИЕ ПОДСИСТЕМЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ

Функционирование любого вида автоматизированного управления требует различных видов обеспечения, показанных на рис. 9.1. Техническую основу автоматизированного управления могут составлять большие, средние и малые ЭВМ. Для сферы организационно-экономического управления наиболее приемлемыми техническими средствами являются сети ЭВМ на базе мощных серверов и персональных ЭВМ.

Обязательным условием функционирования автоматизированного управления является *техническое обеспечение*. Это обосновано выбранным комплексом технических средств для их оснащения.

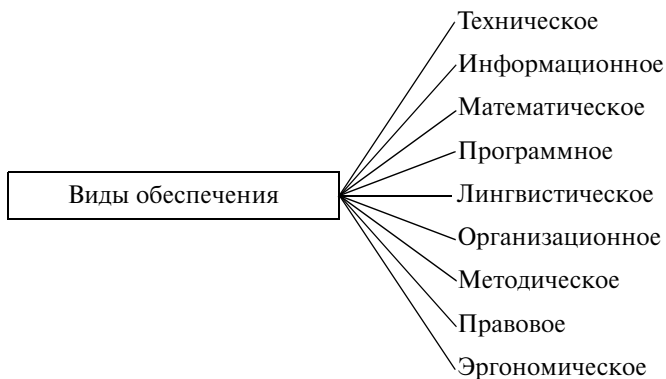


Рис.9.1. *Виды обеспечения автоматизированного управления*

Средства обработки информации — вычислительные машины разных мощностей и типов — составляют основу технического обеспечения вычислительных сетей. Характерной особенностью практического использования технических средств в организационно-экономическом управлении в настоящее время является переход к децентрализованной и сетевой обработке на базе ПЭВМ.

Если ПЭВМ используется в качестве АРМ небольшой локальной сети, на котором централизованно хранится вся информация необходимая для работы, объем обрабатываемой информации велик. Скорость работы при этом определяется не быстродействием компьютера, а скоростью диалога оператора и машины. Отсюда вытекает, что в данном случае вполне приемлема ПЭВМ с небольшим быстродействием и минимальным объемом ОЗУ.

При автоматизированном управлении выполняется децентрализованная одновременная обработка информации на рабочих местах исполнителей в составе распределенной БД. При этом они имеют выход через системное устройство и каналы связи в ПЭВМ и БД других пользователей, обеспечивая, таким образом, совместное функционирование ПЭВМ в процессе коллективной обработки.

Автоматизированное управление, созданное на базе персональных компьютеров, — наиболее простой и распространенный вариант автоматизированного рабочего места (АРМ) для работников сферы организационного управления. Такое автоматизированное управление рассматривается как система, которая в интерактивном режиме работы предоставляет кон-

кретному работнику (пользователю) все виды обеспечения монополю на весь сеанс работы. Этому отвечает подход к проектированию такого компонента АРМ, как внутреннее информационное обеспечение, согласно которому информационный фонд на магнитных носителях конкретного АРМ должен находиться в монополюльном распоряжении пользователя АРМ. Пользователь сам выполняет все функциональные обязанности по преобразованию информации.

Эффективным режимом работы автоматизированного управления является его функционирование в рамках локальной вычислительной сети в качестве рабочих станций. Особенно целесообразен такой вариант, когда требуется распределять информационно-вычислительные ресурсы между несколькими пользователями.

Более сложной формой автоматизированного управления является использование ПЭВМ в качестве *интеллектуального терминала*, а также с удаленным доступом к ресурсам центральной (главной) ЭВМ или внешней сети. В данном случае несколько ПЭВМ подключаются по каналам связи к главной ПЭВМ, при этом каждая ПЭВМ может работать и как самостоятельное терминальное устройство.

В наиболее сложных системах пользователи могут через специальное оборудование подключаться не только к ресурсам главной ЭВМ сети, но и различным информационным службам и системам общего назначения (службам новостей, национальным информационно — поисковым системам, базам данных и знаний, библиотечным системам и т. п.).

Возможности создаваемых АРМ в значительной степени зависят от технико-эксплуатационных характеристик ЭВМ, на которых они базируются. В связи с этим на стадии проектирования АРМ четко формулируются требования к базовым параметрам технических средств обработки и выдачи информации, набору комплектующих модулей, сетевым интерфейсам, эргономическим параметрам устройств и т. д.

Информационное наполнение автоматизированного управления при определении круга пользователей и выяснении сущности решаемых задач осуществляет *информационное обеспечение* (ИО). В сфере организационного управления пользователи могут быть условно разделены на три категории: руководители, персонал руководителей и обслуживающий персонал. Разрабатываемое ИО для разных категорий пользователей отличаются видами представления данных. К примеру, обслуживающий персонал обычно имеет дело с внутренними данными организации, решает

повторяющиеся задачи, пользуется, как правило, структурированной информацией.

Руководителям требуются как внутренние, так и внешние данные для реализации цели управления или принятия решения.

Математическое обеспечение автоматизированного управления представляет собой совокупность алгоритмов, обеспечивающих формирование результатной информации. Математическое обеспечение служит основой для разработки комплекса прикладных программ.

В составе *программного обеспечения* (ПО) автоматизированного управления можно выделить два основных вида обеспечения, различающихся по функциям: общее (системное), специальное (прикладное).

К общему программному обеспечению относится комплекс программ, обеспечивающий автоматизацию разработки программ и организацию экономического вычислительного процесса на ПЭВМ безотносительно к решаемым задачам. Специальное программное обеспечение представляет собой совокупность программ решения конкретных задач пользователя.

Основными приложениями *пакетов прикладных программ* (ППП), входящих в состав специального ПО автоматизированного управления, являются: обработка текстов, табличная обработка данных, управление базами данных, машинная и деловая графика, организация человеко-машинного диалога, поддержка коммуникаций и работа в сетях.

Наиболее эффективными пакетами в автоматизированном управлении являются многофункциональные интегрированные пакеты, реализующие несколько функций переработки информации: табличную, графическую, управление базами данных, текстовую обработку в рамках одной программной среды.

Интегрированные пакеты удобны для пользователей. Они имеют единый интерфейс, не требуют стыковки входящих в них программных средств, обладают достаточно высокой скоростью решения задач.

Лингвистическое обеспечение автоматизированного управления включает языки общения с пользователем, языки запросов, информационно-поисковые языки, языки-посредники в сетях. Языковые средства обеспечивают однозначное смысловое соответствие действий пользователя и аппаратной части в виде ПЭВМ.

Одновременно языки должны быть пользовательско-ориентированными, в том числе профессионально-ориентированными.

Основу языков автоматизированного управления составляют заранее определяемые термины, описания способов установления новых терминов, списки правил, на основе которых пользователь может строить формальные конструкции, соответствующие его информационной потребности.

Языковые средства автоматизированного управления можно разделить по видам диалога. Средства поддержки диалога определяют языковые конструкции, знание которых необходимо пользователю. В одном АРМ может быть реализовано несколько типов диалога: иницилируемый ЭВМ, с помощью заполнения шаблонов, с использованием меню, гибридный диалог и др.

Организационное обеспечение автоматизированного управления включает комплекс документов, регламентирующих деятельность специалистов при использовании компьютера или терминала другого вида на рабочем месте и определяющих функции и задачи каждого специалиста.

Методическое обеспечение автоматизированного управления состоит из методических указаний, рекомендаций и положений по внедрению, эксплуатации и оценке эффективности их функционирования. Оно включает в себя также организованную машинным способом справочную информацию, средства обеспечения работы автоматизированного управления и демонстрационные примеры.

Эргономическое обеспечение автоматизированного управления представляет собой комплекс мероприятий, обеспечивающих максимально комфортные условия использования АРМ специалистами. Это предполагает выбор специальной мебели для размещения аппаратуры, организацию картотек для хранения документации и магнитных носителей.

Правовое обеспечение автоматизированного управления — это система нормативно-правовых документов, определяющих права и обязанности специалистов в условиях функционирования АРМ. Эти документы строго увязаны с комплексом разработок, регламентирующих порядок хранения и защиты информации, правила ревизии данных, обеспечение юридической подлинности совершаемых на АРМ операций и т. д.

Эффективное функционирование АИС и АРМ базируется на комплексном использовании современных технических и программных средств обработки информации в совокупности с современными *организационными формами* размещения техники.

9.1. Математическое обеспечение автоматизированных систем

Мы рассмотрим математическое обеспечение применительно к системам управления производством, во-первых, потому, что в настоящее время этот класс систем является наиболее разработанным, и, во-вторых, наиболее типичным.

В соответствии с ГОСТом, под математическим обеспечением понимается совокупность математических моделей, методов и алгоритмов для решения задач обработки информации с применением средств вычислительной техники.

Если математическое обеспечение рассматривать с позиций используемого математического аппарата, то его можно разделить на три части:

Задачи логической обработки информации (типа сортировки, слияния, объединения и т. п.). Эти задачи, как правило, реализованы в стандартном программном обеспечении.

Задачи прямого счета, или, их еще называют, рутинные задачи. Это вычисления по известным соотношениям сумм, произведений, сумм произведений и т. д. Типичный пример — расчет заработной платы повременщика: необходимо его почасовую тарифную ставку умножить на число отработанных им часов, затем учесть надбавки, если они положены, затем произвести вычеты, которые положены всегда; в итоге получим величину заработной платы, подлежащую выдаче.

Задачи оптимизации. Процесс оптимизации предполагает выбор такого варианта решения, при котором достигается экстремальное (максимальное или минимальное) значение некоторой функции, характеризующей качество управления. Выбор этой функции осуществляется постановщиком задачи в зависимости от целей, стоящих перед системой управления.

Наиболее часто используются следующие основные экономико-математические модели управления производством (ЭММ):

1) модели, которые получили название *производственные функции*. Это простейший вид моделей. Они представляют собой алгебраические зависимости между различными факторами и показателями производства. Примеры факторов производства: численность основных рабочих, численность аппарата управления, размер и степень амортизации основных средств, виды и количество оборудования и т. п. Примеры показателей производства: прибыль, себестоимость, количество выпускаемой продукции и т. п.;

2) *балансовые модели*. Они представляют собой линейные зависимости между различными производственными факторами и

показателями производства. Типичный пример — модель межотраслевого баланса, которая будет рассматриваться ниже;

3) *модели объемного планирования*. Это, по сути, те же самые балансовые модели, но с указанием некоторых критериев оценки;

4) *модели календарного планирования*, или, как их еще называют, модели расписания. Они упорядочивают по времени выполнение тех или иных работ или событий. Проще говоря, это разработка расписания некоторой деятельности;

5) *потокосые модели*. Они оперируют с потоками продукции, финансов, энергии, услуг одного предприятия другим и т. п. Чаще всего эти модели отображаются с помощью графов;

6) *модели управления запасами*. С одной стороны, накопление запасов ведет к излишним издержкам, связанным с их хранением, запасы могут терять качество, морально устаревать и т. п. Все это приводит к финансовым потерям предприятия. С другой стороны, отсутствие запасов и вызванные этим случайные остановки производства вынуждают предприятие-поставщик платить штрафы своим потребителям за недопоставку. Указанные модели оптимизируют эту ситуацию;

7) *модели распределения*, которые осуществляют поиск решения в условиях взаимозаменяемости оборудования, технологий, процессов и т. п.;

8) *модели массового обслуживания*. В отличие от всех предыдущих моделей, они учитывают случайный характер различных факторов, воздействующих на производство. Например, моменты выхода оборудования из строя случайны во времени, продолжительность ремонта также величина случайная. Если ремонтников очень много, то они будут простаивать, и предприятие несет убыток; если их очень мало, то будет простаивать оборудование, и предприятие снова понесет убытки. Модели массового обслуживания позволяют оптимизировать эти ситуации;

9) *имитационные модели*. Это самый универсальный вид моделей, поскольку это есть не что иное, как эксперимент, проводимый на ЭВМ, и, стало быть, с помощью имитации можно решать любой из перечисленных выше классов задач.

Модели типа «производственные функции». Как отмечалось, это самый простейший вид моделей, они предназначены для выработки общей экономической политики.

Пример 1. *Определение размера партии выпускаемых деталей.*

Пусть цена партии деталей прямо пропорциональна количеству штук деталей в партии, а затраты на выпуск также прямо пропорциональны количеству деталей в партии и требуют некоторых первоначальных капиталовложений (рис 9.2).

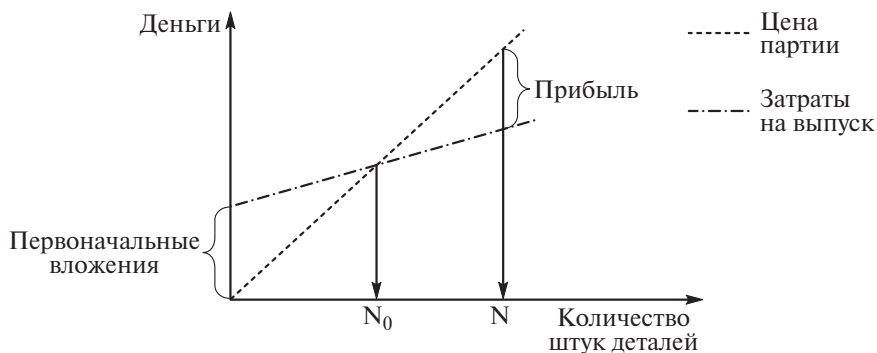


Рис. 9.2. *Определение размера партии деталей*

Из рис. 9.2 следует, что, только начиная с некоторого количества деталей N_0 , имеет смысл налаживать производство. Если задаться желаемой величиной прибыли, то можно определить, не менее какого числа N должна содержать партия деталей. В этом примере фактор производства — количество деталей, показатель производства — прибыль.

Пример 2. *Модель реализации продукции.*

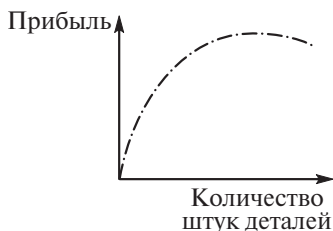


Рис.9.3. *Модель реализации продукции*

Из рис. 9.3 следует, что сначала прибыль растет, а затем, по мере насыщения рынка данной продукцией, начинает падать.

Аналитически производственные функции записываются в виде:

$$y = f(x),$$

здесь y — показатель производства, x — фактор производства.

Приращение показателя:

$$\Delta y = f'(x) \cdot \Delta x.$$

Поиск управляющих воздействий в таких моделях чрезвычайно прост:

- если $f'(x) > 0$, то рост фактора x целесообразно стимулировать;
- если $f'(x) < 0$, то целесообразно уменьшать значение x ;
- если $f'(x) = 0$, то ситуацию можно сохранить.

На практике очень часто производственные функции являются функциями нескольких аргументов (нескольких производственных факторов), т. е.

$$y = f(x_1, x_2, \dots, x_n).$$

В этом случае каждая частная производная показывает зависимость y от соответствующего аргумента:

$$\frac{dy}{dx_i} = f'_i, \quad i = \overline{1, n}$$

и полное приращение производственного показателя y :

$$\Delta y = \sum_{i=1}^n f'_i \Delta x_i.$$

Существуют различные способы определения вида функции f . Можно, например, используя эмпирические данные, методами регрессионного анализа найти вид функции $f(x)$ или, например, задавшись из каких-либо соображений видом функции f , опять же, используя статистику, найти все параметры выбранной функции.

Балансовые модели. В них используются свойства аддитивности, линейности и взаимной независимости различных факторов и показателей производства. В самом общем случае это можно записать в виде следующего баланса:

$$F\left(\sum_i \Phi_i\right) = \sum_i F(\Phi_i).$$

Читается эта запись так: влияние суммы равно сумме влияний. Здесь Φ_i — это i -тый фактор производственного процесса, F — это влияние факторов.

Балансовые модели количественно связывают потоки ресурсов и продукции, производственные мощности предприятия и загрузку оборудования и т.п.

Типичный пример: модель межотраслевого баланса. Эта модель задается системой линейных уравнений вида:

$$x_i \in \sum_{j=1}^n a_{ij} x_j = y_i, \quad i, j = \overline{1, n}.$$

В этом i -том уравнении:

x_i — это количество некоторой продукции, производимой i -той отраслью;

- y_i — это количество продукции, идущей на внутриотраслевое потребление, т. е. остающейся в этой отрасли;
- a_{ij} — это количество продукции i -той отрасли, идущей на производство единицы продукции в j -тую отрасль;
- $a_{ij}x_j$ — это полное количество продукции, поступающей из i -той в j -тую отрасль.

Аналогичной моделью можно описать связи между предприятиями по продукции, услуги одного производственного объекта, оказываемые другому производственному объекту, и т. п. задачи.

Балансовые модели дают точные прогнозы только для установившегося производственного процесса. Их построение связано с необходимостью определения специальных коэффициентов затрат или нормативов a_{ij} . Если эти коэффициенты меняются во времени, то необходимо уметь прогнозировать их будущие значения.

Модели объемного планирования. Это типичный пример задачи принятия решений с детерминированными параметрами.

Пусть имеется несколько производственных объектов, этими объектами могут быть участки цеха, цеха на предприятии, предприятия в отрасли и т. д. Обозначим эти производственные объекты через $A_1, A_2, \dots, A_m, A_r$ — текущий объект, $r = \overline{1, n}$.

Пусть все эти производственные объекты способны выпускать некоторые изделия, которые обозначим через B_1, B_2, \dots, B_m , некоторое текущее изделие $B_i, i = \overline{1, m}$.

Для выпуска этих изделий необходимо иметь k ресурсов (рабочие различной квалификации и профессий, инженеры, станки и т. п.). Обозначим эти производственные ресурсы через C_1, C_2, \dots, C_k , текущий ресурс — $C_j, j = \overline{1, k}$.

На каждый вид выпускаемой продукции имеется директивный план, т. е. то количество изделий данного вида, не менее которого все производственные объекты должны выпустить все вместе. Обозначим директивный план выпуска изделия B_i через b_i .

Известны нормативы: норматив использования ресурса C_j , который необходим для выпуска одной штуки изделия B_i , обозначим через c_{ij} .

Время использования каждого из ресурсов C_j есть величина ограниченная (не более 24 часов в сутки), кроме того, для каждого из производственных объектов эта величина различная (т. к. на одном предприятии, например, может быть трехсменная работа, на другом — двухсменная и т. п.). Обозначим допустимое время использования ресурса C_j на производственном объекте A_r через a_{rj} .

Известна также прибыль каждого производственного объекта от выпуска одного изделия каждого вида. Обозначим прибыль,

которую получает объект A_r , выпуская одну штуку изделий типа B_i , через P_{ri} .

Требуется так распределить план выпуска всех изделий между всеми объектами, чтобы суммарная прибыль была бы максимальной.

Это словесное описание задачи (проблемы). Теперь разработаем ее математическую модель.

Обозначим через X_{ri} план выпуска изделий типа B_i , $i = \overline{1, m}$, который мы поручим предприятию или производственному объекту A_r , $r = \overline{1, n}$. Следует отметить, что каждый из производственных объектов, выпуская одну штуку изделий типа B_i , получает различную прибыль в зависимости от технологии, организации труда и т.п.

Рассмотрим объект A_r : выпуская одну штуку изделий типа B_i , это предприятие получает прибыль P_{ri} , но по плану, который мы ищем, он должен выпускать X_{ri} штук, значит, прибыль этого объекта от выпуска данного изделия составит $P_{ri} X_{ri}$. Аналогичные рассуждения можно провести и для любого из перечисленных изделий, следовательно, суммарная прибыль объекта будет складываться от выпуска всех изделий, значит, указанную величину надо просуммировать по всем изделиям. В результате прибыль объекта A_r будет равна:

$$\sum_{i=1}^m P_{ri} X_{ri}.$$

Но нас интересует суммарная прибыль всех объектов, поэтому указанную величину необходимо просуммировать по всем объектам. Суммарная прибыль составит:

$$\sum_{r=1}^n \sum_{i=1}^m P_{ri} X_{ri} = F. \quad (9.1)$$

Итак, задача заключается в том, чтобы найти такие X_{ri} , которые бы обеспечивали максимум выражения (9.1), но при этом на все X_{ri} накладываются следующие ограничения: искомые плана X_{ri} не могут быть отрицательными

$$X_{ri} \geq 0, \quad r = \overline{1, n}, \quad i = \overline{1, m}, \quad (9.2)$$

второй вид ограничений вытекает из необходимости выполнения директивных планов. Одно предприятие выпускает X_{ri} штук изделий, тогда полный объем выпуска этих изделий получится в ре-

зультате суммирования X_{ri} по всем предприятиям. И эта величина не может быть меньше директивного плана b_i т. е.

$$\sum_{i=1}^m X_{ri} \leq b_i, \quad i = \overline{1, m}, \quad (9.3)$$

следующая группа ограничений вытекает из ограниченности ресурсов. Выпуская 1 шт. изделий вида B_j , мы занимаем ресурс C_j в течении времени c_{ij} (например, 1/2 часа изготавливается конкретная деталь на конкретном станке). Но выпускать мы будем X_{ri} штук изделий. Стало быть $c_{ij}X_{ri}$ — время занятости данного ресурса выпуском деталей только этого вида. Но этот же ресурс может быть занят выпуском и других изделий. Тогда $\sum_{i=1}^m c_{ij}X_{ri}$ — полное время занятости ресурса C_j . Но время использования каждого из ресурсов есть величина ограниченная, поэтому

$$\sum_{i=1}^m c_{ij}X_{ri} \leq a_{rj}, \quad r = \overline{1, n}, \quad j = \overline{1, k}. \quad (9.4)$$

И, окончательно, необходимо найти такие планы выпуска всех изделий на всех производственных объектах X_{ri} , которые бы обращали в максимум выражение (9.1) при соблюдении систем ограничений (9.2—9.4).

Если решать эту задачу в масштабах отрасли, то приближительными параметрами задачи будут: число предприятий $n = 20$, число видов продукции $m = 50$, число видов ресурсов $k = 10$. В результате число ограничений составит 1250.

Рассмотрим числовой пример модели объемного планирования.

Пусть, для простоты, мы имеем всего один производственный объект (скажем, цех). Стало быть, в выше указанных обозначениях, $n = 1$, A_1 — производственный объект.

Пусть также этот объект способен выпускать только два вида изделий. Тогда, в этих же обозначениях, $m = 2$, изделия — B_1 и B_2 .

Пусть также для выпуска этих двух изделий требуются всего два вида производственных ресурсов. В выше указанных обозначениях, $k = 2$, C_1 и C_2 — производственные ресурсы.

Известны нормативы: для выпуска одной штуки изделий вида B_1 ресурс C_1 необходимо использовать один час, а ресурс C_2 — два часа. Аналогично, для выпуска одной штуки изделий вида B_2 ресурс C_1 требуется использовать два часа, а ресурс C_2 — один час (рис. 9.4).



Рис. 9.4. Использование ресурсов для выпуска 1 шт. B_1 и 1 шт. B_2

Максимально возможное время использования ресурса C_1 — 8 часов в сутки, а ресурса C_2 — 10 часов.

Известна прибыль, которую получает производственный объект от выпуска этих изделий. Пусть, выпуская 1 штуку изделий B_1 , производственный объект получает прибыль 5 тыс. руб., а если он же выпускает 1 штуку изделий B_2 , то он получает прибыль 3 тыс.руб.

Для простоты будем считать, что директивные планы на выпуск этих изделий отсутствуют.

Требуется составить план выпуска этих изделий, который обеспечивал бы производственному объекту максимальную прибыль.

Мы разработали словесное описание задачи (словесное описание серьезной задачи может составлять несколько томов).

Разработаем теперь математическую модель, для этого обозначим план выпуска, или количество изделий, обеспечивающих максимальную прибыль, для изделия B_1 через X_1 ; для изделия B_2 — X_2 ; X_1 и X_2 — искомые управляющие воздействия, которые необходимо отыскать, это так называемый управляемый параметр задачи.

Сначала распишем выражение для прибыли. Выпуская 1 шт. B_1 , производственный объект получает 5 тыс.руб. прибыли, а штук этих будет X_1 ; $5X_1$ — прибыль, получаемая им от выпуска изделий B_1 , плюс к этому 3 тыс. руб. от выпуска 1 шт. B_2 , а штук этих X_2 ; $3X_2$ — прибыль от выпуска изделий B_2 , а сумма $(5X_1 + 3X_2)$ и будет составлять суммарную прибыль предприятия.

$$5X_1 + 3X_2 = F, \quad \text{требуется } F \rightarrow \max. \quad (9.5)$$

Выражение (9.5) есть критерий оптимальности принимаемого решения, его еще называют целевой функцией. Если бы не было ограничений, то, выпуская бесконечно много изделий одного вида, можно было бы получать бесконечно большую прибыль, но имеются ограничения, которые определяют область допустимых значений управляемых параметров задачи. Рассмотрим эти ограничения.

Первое ограничение — исходные планы не могут быть отрицательными, если мы поручим выпускать изделие B_1 , то $X_1 > 0$ если не поручим, то $X_1 = 0$. Это справедливо и для X_2 т. е.

$$X_1 \geq 0, \quad X_2 \geq 0. \quad (9.6)$$

Опять же, для простоты, мы снимаем требования целочисленности X_1 и X_2 .

Второе ограничение — ограничение по времени использования ресурсов. Рассмотрим ресурс C_1 — выпуская 1 шт. изделий B_1 , ресурс C_1 необходимо использовать 1 час, а всего будет выпускаться X_1 штук, следовательно, для выпуска всех изделий вида B_1 ресурс C_1 будет занят $1 \times X_1$ часов, но ресурс C_1 занят также и выпуском B_2 . Тогда полное время занятости ресурса C_1 будет $X_1 + 2X_2$, но, по условию, максимально возможное время использования ресурса C_1 не может превышать 8 часов, значит,

$$X_1 + 2X_2 \leq 8. \quad (9.7)$$

Совершенно аналогично расписывается ограничение и по использованию ресурса C_2 :

$$2X_1 + X_2 \leq 10. \quad (9.8)$$

Итак, требуется найти такие значения X_1 и X_2 (такие планы, такие управляющие воздействия), которые бы, удовлетворяя ограничениям (9.6—9.8) (которые определяют область допустимых значений управляющих воздействий), обращали бы в максимум выражение (9.5) (критерий эффективности). И, окончательно,

$$5X_1 + 3X_2 = F \rightarrow \max \quad \text{при} \quad \left. \begin{array}{l} X_1 + 2X_2 \leq 8, \\ 2X_1 + X_2 \leq 10, \\ X_1 \geq 0, \\ X_2 \geq 0 \end{array} \right\}.$$

Модели календарного планирования. Эти модели упорядочивают во времени выполнение каких-либо работ или событий. Математическим аппаратом моделей календарного планирования является теория расписаний. Рассмотрим типичный пример моделей календарного планирования.

Пусть имеется m деталей, подлежащих обработке; обработка идет на n станках: $i = 1, m, j = 1, n$. Маршрут обработки, то есть порядок прохождения каждой детали станков, у всех деталей оди-

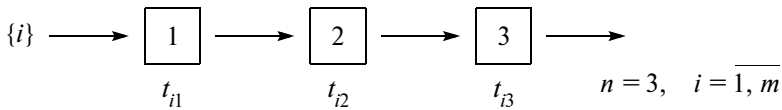


Рис. 9.5. Модель календарного планирования

наков, но время обработки естественно различное. Требуется упорядочить очередь деталей на входе, т.е. определить такую последовательность запуска деталей в производство, которая минимизировала бы суммарное время обработки всех деталей.

Эту задачу можно проиллюстрировать рисунком 9.5.

Указанная задача решается при следующих ограничениях: для каждой детали i ее обработка на станке j может начаться не ранее, чем она закончится на станке $(j-1)$. Если t_{ij} — время начала обработки i -той детали на j -том станке; $\bar{t}_{i(j-1)}$ — время окончания обработки i -той детали на j -том станке, то можно записать:

$$t_{ij} \geq \bar{t}_{i(j-1)},$$

на каждом станке одновременно может обрабатываться только одна деталь.

Процесс обработки детали не прерывается до полного его завершения

$$\bar{t}_{ij} = t_{ij} + t_{ij}$$

В настоящее время указанная задача строго решена только для случая двух станков. При $n > 2$ используются либо эвристические приемы, не гарантирующие отыскания глобального экстремума, либо перебор всех возможных ситуаций.

Пример. Пусть $n = 2$. В этом случае может быть использован алгоритм Джонсона. Обозначим для простоты $t_{i,1}$ через a_i , $t_{i,2}$ через b_i .

Первый шаг алгоритма. Среди чисел $(a_1, a_2, \dots, a_m, b_1, b_2, \dots, b_i, \dots, b_m)$ ищется наименьшее число. Если этим числом оказывается число a_i , то полагают $i = i_1$, т. е. помещают i -тую деталь на первое место в расписании. Если наименьшим числом оказалось число b_i , то полагают $i = i_m$, т. е. помещают i -тую деталь на последнее место в расписании. Если как в 1-м, так и во 2-м случае оказалось несколько равных чисел, то берут любое из них, затем деталь i вычеркивается из расписания (т. е. вычеркиваются числа (a_i, b_i)).

k-тый шаг алгоритма ($1 < k < m$): Среди оставшихся чисел, еще не вычеркнутых на предыдущих шагах, находим наименьшее

число. Если им оказалось число a_i , то помещаем деталь i на первое место из мест еще не занятых в списке. Если им оказалось число b_i , то помещаем на последнее еще не занятое место в списке; вычеркиваем деталь i из списка.

t-ный шаг алгоритма (последний): Помещаем единственную оставшуюся деталь на единственное свободное место.

Рассмотрим числовой пример: пусть время обработки деталей задано табл. 9.1.

Таблица 9.1

i	a_i	b_i	Номер шага	Место детали в расписании
1	8	9	V	$i_1 = 3$
2	3	5	I	$i_2 = 1$
3	4	6	II	$i_3 = 2$
4	10	9	VI	$i_4 = 4$
5	7	4	III	$i_5 = 6$
6	7	5	IV	$i_6 = 5$

Таким образом, оптимальное расписание $\text{opt} \langle i_1, i_2, i_3, i_4, i_5, i_6 \rangle = \langle 2, 3, 1, 4, 6, 5 \rangle$. Перебор потребовал бы просмотра $6! = 720$ вариантов.

Модели управления запасами. *Запас* — это всякое неиспользуемое в данный момент времени средство, имеющее экономическую ценность. Из этого определения следует, что на данный запас всегда присутствует спрос. Запас пополняется посредством поступлений (рис. 9.6).

Экономической единицей могут быть участок, цех, предприятие, объединение, отрасль. Дискретность поставок, свои произ-



Рис. 9.6. Структура модели управления запасами

водства, случайные колебания в спросе вынуждают экономическую единицу создавать запас своей продукции с тем, чтобы не подвергать потребителя риску простоя, поскольку в этой ситуации экономическая единица вынуждена платить неустойки или штраф. С другой стороны, накопление чрезмерных запасов приводит к излишним затратам на их хранение, запасы морально и физически устаревают. Все это приводит к омертвлению до (50%) капитала.

Таким образом, основной задачей управления запасами является определение минимального уровня запаса при удовлетворительном снабжении всех потребителей. В данной модели производство рассматривается как процесс поступления (от поставщика) и выхода (к потребителю). Поступление и выход описываются интенсивностями $a(t)$ и $b(t)$.

Если уровень запаса в момент времени $t = 0$ есть I_0 , то в момент времени t функция запаса $I(t)$:

$$I(t) = I_0 + \int_0^t [a(t) - b(t)] dt.$$

Интенсивность выхода очень часто отождествляется с интенсивностью спроса $r(t)$. Этой характеристикой характеризуется предприятие-потребитель. Функция запаса $I(t)$, как отмечалось выше, вызывает появление затрат, которые являются в общем случае некоторым функционалом от $a(t)$, $b(t)$, $r(t)$:

$$E = E\{a(t), b(t), r(t)\}.$$

Этот функционал характеризует эффективность системы. Требуется найти такие значения функций $a(t)$, $b(t)$, $r(t)$, которые бы оптимизировали его значение.

Рассмотрим частный случай этой задачи, введя следующие предположения.

Пусть спрос не зависит от времени и присутствует всегда. Уменьшение запаса от спроса происходит с постоянной скоростью r , т. е.

$$\Delta I(t) = -r\Delta t, \quad r = \text{const.}$$

Пополнение запаса происходит в дискретные моменты времени. Увеличение запаса от поступления происходит с некоторой постоянной скоростью p , т. е.

$$\Delta I(t) = +p\Delta t, \quad p = \text{const.}$$

Процесс движения запасов бесконечен.

Каждому поступлению соответствуют постоянные издержки s .

Расходы, связанные с хранением запаса, прямо пропорциональны интегралу от положительного уровня запаса, коэффициент пропорциональности h .

Потери от выплаты неустоек в случае невозможности удовлетворения потребителя прямо пропорциональны интегралу от отрицательного уровня запаса; коэффициент пропорциональности d .

Спрос не уменьшается от наличия дефицита.

Изобразим графически поведение функции запаса, словесное описание которой представлено выше (рис. 9.7).

На рис. 9.7 приняты следующие обозначения: $I(t)$ — функция запаса, t — текущее время, I_{max} — максимально возможный уровень запасов, i — нижний возможный уровень запасов, T — период обращения запаса. При $0 \leq t \leq t_1$ идет пополнение запаса с постоянной скоростью p и его расход с постоянной скоростью r . При $t = t_1$ пополнение запаса прекращается, и, начиная с этого момента, идет только его расход с постоянной скоростью r . При $t = t_1 + t_2$ уровень запаса равен нулю, и начинается отрицательный расход запаса. При $t = t_1 + t_2 + t_3$ вновь начинается процесс поступления запаса, который поступает со скоростью p и продолжает расходоваться при этом со скоростью r .

Исходя из словесного описания задачи и ее графического представления, разработаем теперь математическую модель этой задачи.

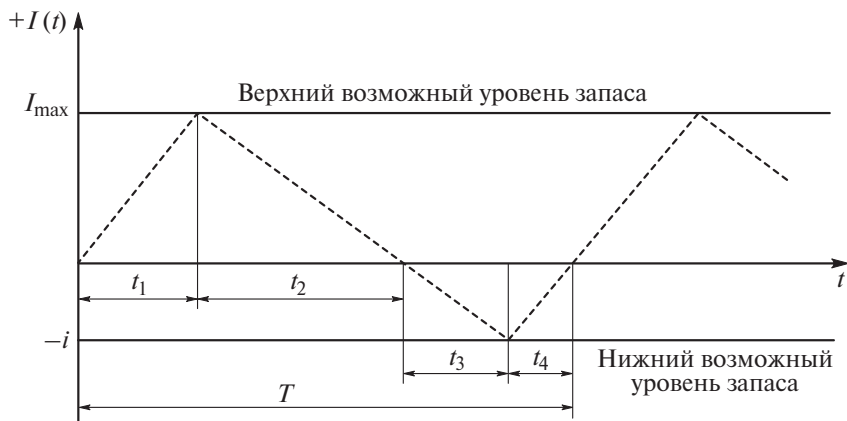


Рис. 9.7. Функция запаса

Поведение функции запаса $I(t)$ можно представить следующим образом:

$$\begin{cases} (p-r)t & \text{при } 0 \leq t \leq t_1, \\ I_{\max} - r(t-t_1) & \text{при } t_1 \leq t \leq t_1 + t_2 + t_3, \\ -i + (p-r)(t-t_1-t_2-t_3) & \text{при } t_1 + t_2 + t_3 \leq t \leq T \end{cases}$$

При $t = t_1$, $I(t) = I_{\max}$. Тогда из первой строчки выражения для $I(t)$ следует, что

$$t_1 = \frac{I_{\max}}{(p-r)}. \quad (9.9)$$

При $t = t_1 + t_2$, $I(t) = 0$. Учтя это, из второй строчки

$$t_2 = \frac{I_{\max}}{r}. \quad (9.10)$$

При $t = t_1 + t_2 + t_3$, $I(t) = -i$. Тогда из последней строчки

$$t_3 = \frac{i}{r}. \quad (9.11)$$

$$I(t) = 0. \quad (9.12)$$

Учтя полученные выражения (9.9—9.12), получим связь между максимумом запаса I_{\max} и максимумом дефицита i

$$i = \left(r(p-r) \frac{T}{p} \right) - I_{\max}. \quad (9.13)$$

Из словесного описания следует, что функция затрат за период C_T есть издержки, связанные с поступлением s и штрафом γ :

$$C_T = s + h \int_0^{t_1+t_2} I(t) dt + \gamma \int_{t_1+t_2}^T I(t) dt. \quad (9.14)$$

Затраты на единицу времени $C = \frac{C_T}{T}$. Учтя найденные ранее значения t_1 , t_2 , t_3 , t_4 , подставив выражения для $I(t)$ из формулы для функции запасов и выполнив интегрирование, можно получить аналитическое выражение для затрат на единицу времени C .

Оптимальное значение периода T и максимального уровня запаса I_{\max} можно получить, если продифференцировать выраже-

ние для C по T и I_{\max} и приравнять эти производные нулю. Полученное выражение доставит минимум затрат C . Задаваясь различными вариациями параметров можно исследовать некоторые важные частные случаи этого частного случая модели управления запасами. Например, очень высокий штраф, тогда надо положить

$$\frac{h}{\gamma} \approx 0, \text{ или мгновенное поступление запаса, тогда } \frac{r}{p} \approx 0.$$

Потоковые модели. Потоковые модели объединяют ряд различных задач, общей чертой которых является наличие потоков, циркулирующих между пунктами производства и потребления. Например, потоки продукции, потоки товаров, финансовые потоки, информационные потоки. Наиболее удобной формой представления этих моделей является графовая форма.

Рассмотрим типичный пример потоковой модели — задачу, получившую название в классике «Транспортная задача». Эта модель описывает задачу распределения однородной продукции между пунктами производства и потребления, минимизируя транспортные расходы. Пусть имеется n поставщиков, $i = 1, n$ и m потребителей, $j = 1, m$ этой продукции. Каждый i -тый поставщик характеризуется мощностью производства a_i . Каждый j -тый потребитель характеризуется потребностью в продукции b_j . Известна c_{ij} — себестоимость перевозки единицы продукции от i -того поставщика к j -тому потребителю.

Обозначим через x_{ij} объем перевозок от i -того поставщика к j -тому потребителю. Требуется найти все x_{ij} , которые бы обеспечивали минимум выражения:

$$\sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} \rightarrow \min,$$

т. е. минимум затрат на перевозку этой продукции. При этом должны соблюдаться следующие ограничения.

Перевозимые количества продукции не могут быть отрицательными, т. е.

$$x_{ij} \geq 0, \quad i = \overline{1, n}, \quad j = \overline{1, m}.$$

Потребность каждого потребителя должна быть удовлетворена, т. е.

$$\sum_{i=1}^n x_{ij} \geq b_j, \quad j = \overline{1, m}.$$

Нельзя перевозить больше того, что произведено, т. е.

$$\sum_{j=1}^m x_{ij} \leq a_i, \quad i = \overline{1, n}.$$

При этом, для того чтобы задача имела решение, суммарная потребность не должна превышать суммарной мощности производства, т. е.

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

Например, если пункты $\boxed{1}$, $\boxed{2}$, $\boxed{3}$ — поставщики, а пункты $\textcircled{4}$, $\textcircled{5}$, $\textcircled{6}$, $\textcircled{7}$, $\textcircled{8}$ — потребители и связь между пунктами задается графом, представленным на рис. 9.8, то задача заключается в отыскании X_{14} , X_{15} , X_{16} и т. д.

Эта задача относится к классу задач линейного программирования

Модели распределения. Общей чертой этих моделей является то, что решение ищется в виде распределения некоторых процессов по некоторым объектам. Например, производственной программы по группам оборудования, ресурсов по видам продукции, производства по видам технологии, решаемых задач по узлам АС и т. п. Причем все эти объекты предполагаются взаимозаменяемыми, то есть любой процесс может быть реализован на любом из объектов.

Рассмотрим пример. Пусть в распределенной автоматизированной системе, состоящей из m узлов $j = \overline{1, m}$, необходимо решать n $i = \overline{1, n}$ различных функциональных задач. Как отмечалось, любая задача может быть решена в любом узле. Известна матрица

$$a_{ij} = \begin{pmatrix} a_{11} & \dots & a_{1m} \\ \dots & \dots & \dots \\ a_{i1} & \dots & a_{im} \\ \dots & \dots & \dots \\ a_{n1} & \dots & a_{nm} \end{pmatrix}.$$

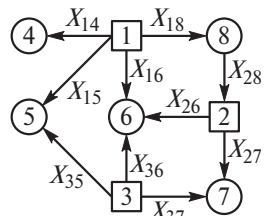


Рис. 9.8.
Граф транспортной задачи

В этой матрице $\|a_{ij}\|$ время, которое потребуется j -тому узлу для решения i -той задачи. Таким образом, $\|a_{ij}\|$ — матрица времен.

Известна также матрица $\|b_{ij}\|$, т. е. матрица затрат, в которой b_{ij} — затраты (например, деньги) j -того узла на решение i -той задачи. Требуется найти матрицу $\|x_{ij}\|$, где $x_{ij} = 1$, если i -тая задача решается в j -том узле, и $x_{ij} = 0$ в противном случае.

Взяв эту модель в качестве исходной, можно сформулировать ряд частных задач, например, так распределить задачи между узлами, чтобы затраты были минимальны, а суммарное время решения всех задач не превышало бы некоторой заданной величины. Или, например, обратная задача: требуется так распределить эти задачи, чтобы суммарное время решения всех задач было минимальным, а затраты не превышали бы некоторой величины. Это общая задача математического программирования.

Модели массового обслуживания. В отличие от всех выше рассмотренных моделей они учитывают случайный характер факторов, воздействующих на производство. Рассмотрим типичный пример — задача обслуживания станков. Пусть имеется некоторое количество станков. Они могут выходить из строя и требовать профилактики и поднастройки. Для этой цели создается бригада наладчиков. Моменты выхода станков из строя происходят в случайные моменты времени t , но известен закон распределения вероятностей, которому подчиняются эти моменты. Продолжительность ремонта также величина случайная с известным законом распределения. Когда станки не работают, предприятие несет убытки. Когда не работают наладчики, предприятие также несет убытки. Известна стоимость единицы времени простоя станка и наладчика. Требуется найти оптимальное число станков, закрепленных за одним наладчиком, которое бы минимизировало суммарные потери.

Учет важности для производства того или иного вышедшего из строя станка, отсутствует или нет возможность прерывания начатого ремонта, учет различной квалификации мастеров — все это, естественно, усложняет модель.

Имитационные модели. Это самый универсальный вид моделей. Имитация представляет собой эксперимент, выполняемый на ЭВМ над математической моделью, описывающей реальный производственный процесс. Этот машинный эксперимент состоит из следующих основных этапов:

1. Постановка задачи исследования.
2. Изучение объекта моделирования.
3. Построение его математической модели.
4. Разработка на основе этой модели моделирующих алгоритмов и их программирование.

5. Оценка точности и достоверности результатов.
6. Планирование машинных экспериментов.
7. Интерпретация результатов моделирования.

В качестве языков для машинного моделирования используются как универсальные языки высокого уровня, так и специализированные языки.

9.2. Информационное обеспечение автоматизированного управления

Совокупность больших сложных человеко-машинных информационных систем является важнейшей составляющей инфраструктуры общества, где информация выступает одним из главных ресурсов его жизнедеятельности. Являясь связующим звеном между разными видами интеллектуальной и материальной деятельности коллективов людей, между управлением и производством, информация, в отличие от других видов ресурсов, в частности природных, материальных, финансовых не убывает со временем, а наоборот, ее объем постоянно увеличивается, создавая условия для накопления опыта, способствуя выработке обоснованных управленческих решений.

Информация очень разнообразна по содержанию и подразделяется по виду обслуживаемой ею человеческой деятельности: научная, производственная, управленческая (социально-экономическая), медицинская, экологическая, правовая и т. п. Каждый из видов информации имеет свои особенные технологии обработки, смысловую ценность, формы представления и отображения на физическом носителе, требования к точности, достоверности, оперативности отражения фактов, явлений, процессов. Технология обработки информации с применением комплекса технических средств вызывает необходимость манипулировать с отдельными информационными элементами, обеспечивать их изучение и форматированное описание, идентификацию для удобства обработки, хранения, передачи информации. Информация, представленная в формализованном виде, получила название «данные».

Особенности управленческой информации. Информация, являясь сложным по структуре образованием, размещается на физических носителях (бумажных или магнитных документах, в виде сигналов, передаваемых по каналам связи) и может находиться в статическом и динамическом состояниях. Статическое состояние информации связано с ее более или менее длительным организо-

ванным хранением, накоплением в информационных фондах и базах данных. Под *базой данных* (БД) понимается вся необходимая для решения задач конкретной области совокупность данных, организованная по определенным правилам, позволяющим обеспечить независимость данных от прикладных программ, удобство хранения, поиска, манипулирования данными, которые записаны на машинных носителях. При этом каждый элемент строго идентифицируется для автоматизации процесса поиска, пополнения, обновления данных. *Динамическое состояние* — постоянное движение в виде потоков — присуще информации, реализующей в человеко-машинных, автоматизированных системах функций обмена сведениями с помощью знаковых символов. Приведенные особенности информации тщательно изучаются при создании систем автоматизированной обработки в процессе ее синтаксического, семантического и прагматического анализа.

Синтаксический анализ устанавливает важнейшие параметры информационных потоков, включая необходимые количественные характеристики, для выбора комплекса технических средств сбора, регистрации, передачи, обработки, накопления и хранения информации.

Семантический анализ позволяет изучить информацию с точки зрения смыслового содержания ее отдельных элементов, находить способы языкового соответствия (языка человека, языка ЭВМ) при однозначном распознавании вводимых в систему сообщений.

Прагматический анализ проводится с целью определения полезности информации, используемой для управления, выявления практической зависимости сообщений, применяемых для выработки управляющих воздействий. Учитывая, что полезность информации является функцией времени и что одна и та же информация в разное время может быть полезной либо бесполезной в зависимости от того, сколько новых сообщений об управляемом объекте она несет пользователю, принятые критерии оценки увязываются с достоверностью и современностью поступающих сообщений.

Информация, которая обслуживает процесс производства, распределения, обмена и потребления материальных благ, и обеспечивает решение задач организационно-экономического управления народным хозяйством и его звеньями, называется управленческой. Она представляет собой разнообразные сведения экономического, технологического, социального, юридического, демографического и другого содержания, используемого при этом. В информационном процессе, каким является управлен-

ческая деятельность, информация выступает как один из важнейших ресурсов наряду с энергетическими, материальными, трудовыми, финансовыми. В технологии обработки первичные сведения о производственных и хозяйственных операциях, людях, выпуске продукции, фактах приобретения и продажи товаров выполняют роль предметов труда, а получаемая результатная информация — продукта труда; она используется для анализа и принятия управленческих решений.

Особенности экономической информации. Важнейшей составляющей управленческой информации является информация экономическая, которая отражает социально-экономические процессы как в сфере производства, так и в непроизводственной сфере, во всех органах и на всех уровнях отраслевого регионального управления. Рассмотрим особенности управленческой информации, в частности экономической, оказывающие влияние на организацию ее автоматизированной обработки.

1. Экономическая информация отражает акты производственно-хозяйственной деятельности с помощью системы натуральных и стоимостных показателей. Во всех случаях при этом используются количественные величины, цифровые значения. Эта способность экономической информации предопределяет возможность широкого применения вычислительной техники в экономике.

2. Следующей отличительной чертой экономической информации является цикличность. Для большинства производственных и хозяйственных процессов характерна повторяемость составляющих их стадии и информации, отражающей эти процессы. Цикличность экономической информации позволяет, однажды создав программу машинного счета, многократно использовать ее. Это значительно упрощает проектирование автоматизированной обработки данных.

3. Важное значение для обработки имеет форма представления информации. Экономическая информация непременно отражается в материальных носителях: в первичных и сводных документах, в машинных носителях (магнитных лентах и дисках, перфолентах), передается по каналам связи. Для повышения достоверности ведется передача и обработка лишь юридически оформленной информации (при наличии подписи на документе или на электронном сообщении, указании кода, передающего сообщение, и т. п.).

4. Отличительной чертой экономической информации является ее объемность. Качественное управление экономическими процессами невозможно без детальной информации о них. Со-

вершенствование управления и возрастание объемов производства сопровождается увеличением сопутствующих ему информационных потоков.

Информационное обеспечение автоматизированного управления предусматривает организацию его информационной базы, регламентирует информационные связи и предопределяет состав и содержание всей системы информационного отображения. Решение об информационном наполнении АРМ специалиста может быть принято на основе предварительного определения круга пользователей для выяснения сущности решаемых ими задач.

В самом общем виде пользователи различаются по служебному положению (должностной иерархии), специальностям, уровню освоения и частоте работы с вычислительной техникой, виду потребляемых данных и т. д.

По служебному положению пользователи в сфере организационного управления могут быть условно разделены на три категории: руководители, персонал руководителей (начальники отделов, руководители групп и др.) и обслуживающий персонал (делопроизводители, секретари, референты и др.).

Как специалисты пользователи могут быть разбиты на группы, например бухгалтеров, финансистов, статистиков, кадровиков и др.

По степени подготовленности к работе с вычислительной техникой выделяются пользователи, имеющие навыки в программировании на алгоритмических языках; имеющие подготовку в используемом программном обеспечении автоматизированного управления; способные выполнять минимум операций на ПЭВМ (загрузка, ввод данных, корректировка, запуск программы, печать выходных таблиц и др.).

Пользователи АРМ могут быть разделены на две группы в зависимости от периода получения данных:

- кому данные нужны в процессе их обработки и формирования (динамическое потребление);
- кому нужны законченные сведения о состоянии объекта (статическое потребление).

В связи с этим для пользователей первой группы нужно обеспечить интерактивный режим работы, для пользователей второй группы он не обязателен.

В зависимости от категории пользователей ИО должно в той или иной мере обеспечивать не только обработку данных, но и выдачу комментариев, подсказок, разъяснений. Разрабатываемое ИО для разных категорий пользователей порой значительно отличаются видами представления данных. К примеру, обслужива-

ющий персонал обычно имеет дело с внутренними данными конкретной организации, решает повторяющиеся задачи, пользуется, как правило, структурированной информацией; получаемые в процессе обработки данные отличаются достаточно коротким активным периодом существования.

Руководителям требуются как внутренние, так и внешние данные. Для такого типа пользователей работа обычно сопутствует реализации какой-либо цели управления или принятия решения. Поэтому в получаемых данных может и не содержаться ответ в явном виде. С другой стороны, количество данных для визуального просмотра или альтернативных вариантов решений должно быть минимально, чтобы обеспечить аналитичность вывода.

Действия, выполняемые пользователем на АРМ, могут быть неточными, поэтому ИО любого уровня должно предусматривать (явным или неявным образом) возможность ошибок в действиях пользователя, например, при вводе данных, нажатии функциональных клавиш. ИО этом случае обеспечивают вывод на экран соответствующих инструкций или сообщений о том, что надо предпринять для корректировки или продолжения работы.

Информация, которую пользователь вынужден затребовать в процессе работы, должна выдаваться на экран достаточно быстро, в течение нескольких секунд. Исключение составляют случаи, когда пользователь сам сознательно инициирует задержки либо самостоятельно проектирует данные.

Приведенные соображения и лежат в основе разработки информационного обеспечения конкретного вида автоматизированного управления.

Первоочередной задачей при этом является организация *внутримашинной информационной базы*: выбор необходимого состава показателей, способа их организации и методов группировки и выборки необходимых данных, определение вида магнитных носителей и распределение между ними активной и пассивной информации, организация справочной и комментирующей информации, разработка макетов для упрощения ввода информации и выборе необходимых функций, возможности динамической адаптации базы данных и т. д.

В случае, когда АРМ пользователя служит элементом распределенной системы обработки информации, т. е. узлом сети, возникают иные, дополнительные требования к организации информационной базы:

- структура базы данных должна позволять легко расчленять ее на составные подбазы, размещаемые на отдельных АРМ, обеспечивать простоту доступа к любой подбазе, за-

- щиту от несанкционированного доступа к тем или иным данным и высокую производительность в работе с данными;
- структура информационной базы должна сохранять адекватность содержания внешней (документной) и внутренней (на магнитных носителях прямого доступа) форм хранения информации в разрезе тех объектов, с которыми работает исполнитель;
 - структура информационной базы и схема ее распределения по АРМ должны обеспечивать возможность единого или единовременного процесса корректировки нескольких одинаковых баз, хранящихся на разных АРМ, либо замену скорректированной базы данных одного АРМ на всех остальных, связанных с ним единой информационной взаимосвязью;
 - структура информационной базы должна быть минимально избыточна и одновременно удобна для архивирования данных.

9.2.1. Процедуры обработки информации

Создание и функционирование информационных систем в управлении тесно связаны с развитием информационных технологий. Как известно *информационная технология* — это система методов и способов сбора, накопления, хранения, поиска, обработки и выдачи информации. Она является предметом информатики как научной дисциплины и обеспечивает перевод практики управления, регулирования материального производства, научных исследований и других областей человеческой деятельности на индивидуальный уровень. В отличие от любой другой инженерной технологии, информационная технология позволяет интегрировать различные виды и технологии, а информация, которую она обрабатывает в различных сферах деятельности, синтезируется для накопления опыта и внедрения в практику в соответствии с общественными потребностями. Речь идет о технологическом применении ЭВМ и других технических средств управления. Как всякая технология, индустриальная информационная технология включает:

- 1) определенные наборы материальных средств (носители информации, технические средства измерения ее состояний, обработки, передачи и т. п.);
- 2) способы их взаимодействия;
- 3) совокупность определенных методов, организации работы и специалистов.

Информационная технология решения задач включает следующие важнейшие процедуры, которые могут быть сгруппированы по функционально-временным стадиям:

- 1) сбор и регистрация информации;
- 2) передача ее к месту обработки;
- 3) машинное кодирование данных;
- 4) хранение;
- 5) поиск;
- 6) вычислительная обработка;
- 7) тиражирование информации;
- 8) использование информации, т. е. принятие решений и выработка управляющих воздействий.

Как правило, информация подвергается всем процедурам преобразования, но в ряде случаев некоторые процедуры могут отсутствовать. Последовательность их выполнения также бывает различной, при этом некоторые процедуры могут повторяться. Состав процедур преобразования информации и особенности их выполнения во многом зависят от объекта управления и системы автоматизированной обработки информации.

Рассмотрим особенности выполнения основных процедур преобразования информации.

1. *Сбор и регистрация информации* происходят по-разному в различных экономических объектах. Наиболее сложна эта процедура в автоматизированных управленческих процессах промышленных предприятий, фирм и т. п., где производятся сбор и регистрация первичной учетной информации, отражающей производственно-хозяйственную деятельность объекта.

Особое значение при этом придается достоверности, полноте и своевременности первичной информации. На предприятии сбор и регистрация информации происходят при выполнении различных хозяйственных операций (прием готовой продукции, получение и отпуск материалов и т. п.). Сначала информацию собирают, затем ее фиксируют. Учетные данные могут возникать на рабочих местах в результате подсчета количества обработанных деталей, прошедших сборку узлов, изделий, выявление брака и т. п. Для сбора фактической информации производятся измерения, подсчет, взвешивание материальных объектов, получение временных и количественных характеристик работы отдельных исполнителей. Сбор информации, как правило, сопровождается регистрацией, т. е. фиксацией информации на материальном носителе (документе или машинном носителе).

Запись в первичные документы в основном осуществляется вручную, поэтому процедуры сбора и регистрации остаются пока

наиболее трудоемкими. В условиях автоматизации управления предприятием особое внимание придается использованию технических средств сбора и регистрации информации, совмещающих операции количественного измерения, регистрации, накопления и передачу информации по каналам связи в ЭВМ с целью формирования первичного документа.

Необходимость передачи информации для различных объектов управления обосновывается по-разному. Так, в автоматизированной системе управления предприятием она вызвана тем, что сбор и регистрация информации нередко территориально отделены от обработки. Процедуры сбора и регистрации информации, как правило, осуществляются на рабочих местах, а обработка — в вычислительном центре.

2. *Передача информации* осуществляется различными способами: с помощью курьера, пересылка по почте, доставка транспортными средствами, дистанционная передача по каналам связи. Дистанционная передача по каналам связи сокращает время передачи данных. Для ее осуществления необходимы специальные технические средства. Некоторые технические средства сбора и регистрации, собирая автоматически информацию с датчиков, установленных на рабочих местах, передают ее в ЭВМ.

Дистанционно может передаваться как первичная информация с мест ее возникновения, так и результатная в обратном направлении. В этом случае результатная информация отражается на различных устройствах: дисплеях, табло, печатающих устройствах. Поступление информации по каналам связи в центр обработки в основном осуществляется двумя способами: на машинном носителе или непосредственно в ЭВМ при помощи специальных программных и аппаратных средств.

Дистанционная передача постоянно развивается и совершенствуется. Особое значение этот способ передачи имеет в многоуровневых межотраслевых системах, где применение дистанционной передачи значительно ускоряет прохождение информации с одного уровня управления на другой и сокращает общее время обработки данных.

3. *Машинное кодирование* — процедура машинного представления (записи) информации на машинных носителях в кодах, принятых в ЭВМ.

Такое кодирование информации производится путем переноса данных первичных документов на магнитные диски, информация с которых затем вводится в ЭВМ для обработки.

Запись информации на машинные носители — трудоемкая операция, в процессе которой возникает наибольшее количество

ошибок. Поэтому обязательно выполняются операции контроля записи разными методами на специальных устройствах либо на ЭВМ.

4. *Хранение и накопление* информации вызвано многократным ее использованием, применением постоянной информации, необходимостью комплектации первичных данных до их обработки.

Хранение информации осуществляется на машинных носителях в виде информационных массивов, где данные располагаются по установленному в процессе проектирования группировочному признаку.

5. *Поиск данных* — выборка нужных данных из хранимой информации, включая поиск информации, подлежащей корректировке или замене. Процедура поиска информации выполняется на основе составленного запроса на нужную информацию.

6. *Обработка информации* на ЭВМ производится, как правило, централизованно, а на мини- и микроЭВМ — в местах возникновения первичной информации, где организуются автоматизированные рабочие места специалистов той или иной управленческой службы (отдела материально-технического снабжения и сбыта, отдела главного технолога, конструкторского отдела, бухгалтерии, планового отдела и т.п.).

Автоматизированное рабочее место специалиста включает персональную ЭВМ (ПЭВМ), работающую автономно или в вычислительной сети, набор программных средств и информационных массивов для решения функциональных задач.

Обработка информации на ПЭВМ начинается при полной готовности всех устройств машины. Оператор или пользователь при выполнении работы на ПЭВМ руководствуется специальной инструкцией по эксплуатации технических и программных средств.

В начале работы в машины загружаются программа и различные информационные массивы (условно-постоянные, переменные, справочные), каждый из которых сначала, как правило, обрабатывается для получения каких-либо результатных показателей, а затем массивы объединяются для получения сводных показателей.

При обработке информации на ЭВМ выполняются арифметические и логические операции. Арифметические операции обработки данных в ЭВМ включают все виды математических действий, обусловленных программой. Логические операции обеспечивают соответствующее упорядочивание данных в массивах (первичных, промежуточных, постоянных, переменных), подлежащих дальнейшей арифметической обработке. Значительное место в логических операциях занимают такие виды сортиро-

вальных работ, как упорядочение, распределение, подбор, выборка, объединение.

В ходе решения задач на ЭВМ, в соответствии с машинной программой, формируются результатные сводки, которые печатаются машиной.

7. *Печать сводок* может сопровождаться *процедурой тиражирования*, если документ с результатной информацией необходимо предоставить нескольким пользователям.

8. *Принятие решения* в автоматизированной системе организационного управления, как правило, осуществляется специалистом без применения технических средств, но на основе тщательного анализа результатной информации, полученной на ПЭВМ. Задача принятия решений осложняется тем, что специалисту приходится искать из множества допустимых решений наиболее приемлемое, сводящее к минимуму потери ресурсов (временных, трудовых, материальных и т.д.). Благодаря применению персональных ЭВМ и терминальных устройств повышается аналитичность обрабатываемых сведений, а также обеспечивается постепенный переход к автоматизации выработки оптимальных решений в процессе диалога пользователя с вычислительной системой. Математическая теория принятия решений (ТПР) строится на основе теории игр и исследования операций.

9.2.2. Организация информационных процессов в системах управления

Информационные процессы в автоматизированных системах организационного управления реализуются с помощью ЭВМ и других технических средств. По мере развития вычислительной техники совершенствуются и формы ее использования. Существуют различные способы доступа и общения с ЭВМ. Индивидуальный и коллективный доступ к вычислительным ресурсам зависит от степени их концентрации и организационных форм функционирования. Централизованные формы применения вычислительных средств, которые существовали до массового использования ПЭВМ, предполагали их сосредоточение в одном месте и организацию информационно-вычислительных центров (ИВЦ) индивидуального и коллективного пользования (ИВЦКП).

Деятельность ИВЦ и ИВЦКП характеризовалась обработкой больших объемов информации, использованием несколькими средними и большими ЭВМ, квалифицированным персоналом для обслуживания техники и разработки программного обеспечения.

Централизованное применение вычислительных и других технических средств позволяло организовать их надежную работу, планомерную загрузку и квалифицированное обслуживание.

Централизованная обработка информации наряду с рядом положительных сторон (высокая степень загрузки и высокопрофессиональное использование оборудования, квалифицированный кадровый состав операторов, программистов, инженеров, проектировщиков вычислительных систем и т.п.) имела ряд отрицательных черт, порожденных прежде всего отрывом конечного пользователя (экономиста, плановика, нормировщика и т.п.) от технологического процесса обработки информации.

Децентрализованные формы обработки информации стали применяться со второй половины 80-х годов, когда сфера экономики получила возможность перейти к массовому использованию персональных ЭВМ (ПЭВМ). Децентрализация предусматривает размещение ПЭВМ в местах возникновения и потребления информации, где создаются автономные пункты ее обработки. К ним относятся абонентские пункты (АП) и АРМ.

Назначение АП состоит в организации сбора, регистрации и передачи информации в центр обработки. АП позволяют обеспечить повышение достоверности информации на входе в систему, организовать ее надежное накопление, хранение, первичную обработку, формирование в массивы, а также передачу по каналам связи на верхний уровень управления для обобщения информации, получения необходимой отчетности, выработки управленческих решений.

АРМ оборудуются, как правило, ПЭВМ и предназначаются для автоматизации работ с информацией, относящейся к определенной предметной области (учету, нормированию, финансовой деятельности и т.п.). Организуются они на рабочих местах пользователей — специалистов управленческих служб — для их непосредственного доступа к вычислительным ресурсам.

Развитие организационных форм использования вычислительной техники строится на сочетании централизованной и децентрализованной — смешанной — форм. Предпосылкой появления смешанной формы явилось создание сетей ЭВМ на основе развития современных средств связи.

Сети ЭВМ предполагают объединение в систему с помощью каналов связи вычислительных средств, программных и информационных ресурсов (баз данных, баз знаний). Сетями могут охватываться различные формы использования ЭВМ, причем каждый абонент имеет возможность доступа не только к своим вычислительным, информационным и программным ресурсам, но

и к ресурсам всех остальных абонентов, что создает ряд преимуществ при эксплуатации вычислительной системы.

В последнее время организация применения компьютерной техники претерпевает значительные изменения, связанные с переходом к созданию *интегрированных информационных систем*. Интегрированные информационные системы создаются с учетом того, что они должны осуществлять согласованное управление данными в пределах предприятия (организации), координировать работу отдельных подразделений, автоматизировать операции по обмену информацией как в пределах отдельных групп пользователей, так и между несколькими организациями, отстоящими друг от друга на десятки и сотни километров. Основой для построения подобных систем служат *локальные вычислительные сети* (ЛВС). Характерной чертой ЛВС является предоставление пользователям возможности работать в универсальной информационной среде с функциями коллективного доступа к данным.

В последние годы компьютеризация вышла на новый уровень: активно создаются вычислительные системы различной конфигурации на базе персональных компьютеров (ПК) и более мощных машин. Состоящие из нескольких автономных компьютеров с общими совместно используемыми внешними устройствами (диски, ленты) и единым управлением, они позволяют обеспечить более надежную защиту компьютерных ресурсов (устройств, баз данных, программ), повысить отказоустойчивость, обеспечить простоту модернизации и наращивания мощности системы.

Все больше внимания уделяется развитию не только локальных, но и *распределенных вычислительных сетей* (РВС), без которых немислимо решение современных задач информатизации.

В зависимости от степени централизации вычислительных ресурсов роль пользователя и его функции меняются. При централизованных формах, когда у пользователя нет непосредственного контакта с ЭВМ, его роль сводится к передаче исходных данных на обработку, получению результатов, выявлению и устранению ошибок. При непосредственном общении пользователя с ЭВМ его функции в информационной технологии расширяются. Он сам вводит данные, формирует информационную базу, решает задачи, получает результаты, оценивает их качество. У пользователя открываются реальные возможности решать задачи с альтернативными вариантами, анализировать и выбирать с помощью системы в конкретных условиях наиболее приемлемый вариант. Все это реализуется в пределах одного рабочего места. От пользователя при этом требуется обязательное знание основ информатики и вычислительной техники.

9.2.3. Особенности новой информационной технологии управленческой деятельности

Современные информационные системы организационного управления предназначены оказывать помощь специалистам, руководителям, принимающим решения, в получении ими своевременной, достоверной, в необходимом количестве информации, создании условий для организации автоматизированных офисов, проведение с применением компьютеров и средств связи оперативных совещаний, сопровождаемых звуковым и видеорядом. Достигается это переходом на новую информационную технологию. *Новая информационная технология* — технология, которая основывается на:

- 1) применении компьютеров;
- 2) активном участии пользователей (непрофессионалов в области программирования) в информационном процессе;
- 3) высоком уровне дружественного пользовательского интерфейса;
- 4) широком использовании пакетов прикладных программ общего и проблемного назначения;
- 5) возможности для пользователя доступа к удаленным базам данных и программ благодаря сетям ЭВМ.

Постоянно расширяющиеся сферы применения персональных компьютеров, их массовое использование, в том числе и в экономической работе, привели к необходимости формирования наиболее эффективных организационных форм применения вычислительной и 1ругой организационной техники. В настоящее время на их основе создаются и успешно функционируют локальные и многоуровневые вычислительные сети, представляющие собой интегрированные компьютерные системы обработки данных.

Интегрированные компьютерные системы обработки данных проектируются как сложный информационно-технологический и программный комплекс, поддерживающий единый способ представления данных, единый способ взаимодействия пользователей с компонентами системы и обеспечивающий информационные и вычислительные потребности специалистов в их профессиональной работе. Особое значение в таких системах придается защите информации при ее передаче и обработке.

Наибольшее распространение при защите экономической информации получили аппаратно-программные способы: в частности, использование системы связи, выбранной по защитным свойствам и качеству обслуживания, гарантирующим сохранность информации в процессе передачи и доставки ее адресату;

шифрование и дешифрование данных абонентами сетей общего пользования (телефонных, телеграфных) при договоренности пользователей об общих технических средствах, алгоритмах шифрования и т.п.

Интегрированные системы обработки данных создаются на основе объединения и жесткой увязки всех входящих в систему элементов в информационном, программном и техническом аспектах. При этом должна быть построена максимально унифицированная технологическая схема функционирования системы с использованием общих четко спроектированных для разных задач структур и моделей данных.

Реализация принципа интеграции, накопления, хранения и систематического обновления данных для своевременного и надежного обслуживания многочисленных пользователей системы закладывается на стадии ее создания. Учитывается, что пользователями информации будут не только специалисты конкретной проблемной области управленческой деятельности (учета, планирования, менеджмента, маркетинга и т. п.), но и программисты, занимающиеся созданием и эксплуатацией программных средств.

Поэтому в процессе проектирования баз данных (БД) ведется тщательное разностороннее исследование предметной области, ее элементов, связи между ними, а также выявляются особенности циркулирующих в ней данных как особо важного ресурса. Создается общая структурная схема баз данных в виде многоуровневых моделей, формируются условия и осуществляется выбор системы управления базами данных (СУБД). При этом между пользователями устанавливаются соглашения по составу и структуре данных; разрабатываются способы фильтрации ошибочных данных, вводимых в систему; устанавливаются необходимые разграничения доступа к массивам конкретных пользователей; удовлетворяются требования независимости данных от программ и их физического расположения и т.п. Все перечисленное учитывается среди прочих факторов при выборе или создании СУБД.

В процессе реализации информационной технологии БД и СУБД нуждаются в систематическом обслуживании, поддержании в рабочем состоянии (сопровождении). Эти функции выполняет администратор БД, т.е. один или несколько специалистов, которые несут ответственность за функционирование интегрированной БД, отвечают как за целостность данных, так и за защиту их от несанкционированного доступа, надежность системы.

Функционирование баз данных под управлением СУБД и при систематическом контроле со стороны администратора представляет собой взаимодействие сложного организационно-техноло-

гического комплекса, который получил название автоматизированного банка данных.

Под *автоматизированным банком данных* (БНД) понимается организационно-технологический комплекс (система), включающий базы данных для решения функциональных задач управления, технические, программные и языковые средства, а также обслуживающий персонал.

9.2.4. Базы данных и их системы управления

Части программного обеспечения и методы, осуществляющие управление базой данных, составляют *систему управления базами данных*. На рис. 9.9 показан пример использования СУБД и БД для проектирования элементов ЭВМ.

Система управления БД позволяет получить доступ к интегрированным данным и допускает множество различных представлений о хранимых данных. Приведем основные определения СУБД, сформулируем требования, которым они должны удовлетворять, и опишем функции, выполняемые СУБД.

Программное обеспечение, которое позволяет прикладным программам работать с БД без знания конкретного способа размещения данных в памяти ЭВМ, называют СУБД. Система управления БД выступает как совокупность программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями.

Система управления БД должна обеспечивать простоту физической реализации БД; возможность централизованного и децентрализованного управления БД; минимизацию избыточности хранимых данных; предоставление пользователю по запросам не-

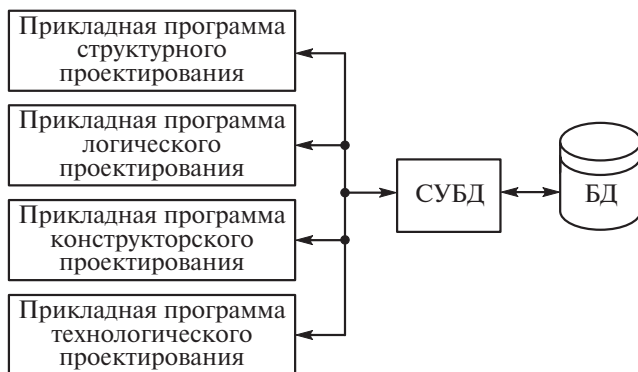


Рис. 9.9. Пример использования СУБД и БД для проектирования

противоречивой информации; простоту разработки, ведение и совершенствование прикладных программ; выполнение различных функций.

Система управления БД реализует два интерфейса: 1) между логическими структурами данных в программах и в БД; 2) между логической и физической структурами БД.

Опишем последовательность работы СУБД в одном из режимов: 1) программа запрашивает возможность чтения данных у СУБД, она передает необходимую информацию о программисте, типе записи и т.п.; 2) программа осуществляет поиск описания данных, на которые выдан запрос; 3) определяет, какого типа логические и физические записи необходимы; 4) выдает ОС запрос на чтение требуемой записи; 5) ОС взаимодействует с физической памятью; 6) записывает запрошенные данные в системные буферы; 7) выделяет требуемую логическую запись, выполняя необходимые преобразования; 8) передает данные из системных буферов в программу пользователя, а затем в программу пользователя — информацию о результатах выполнения запроса; 9) прикладная программа обрабатывает полученные данные.

Распределенная БД — это БД, в которой данные размещены по месту возникновения или наиболее эффективного использования. Она предполагает, что на каждой ЭВМ данные управляются локальными СУБД.

Для работы с распределенными БД создаются специальные системы управления распределенными БД (СУРБД). Основное требование к СУРБД такое же, как и к СУБД: обеспечение максимальной независимости прикладных программ САПР от локализации данных в сети. Основой СУРБД являются каталоги, в которых хранится информация о структуре сети, описание локальных СУРБД и т. п. Особенность управления распределенными БД определяется разнотипностью локальных СУРБД, которые желательно иметь одного типа. Если же разные части распределенной БД управляются разными СУБД, то используют некоторую промежуточную интерфейсную СУРБД, через которую на основе соответствующих отображений обеспечивается взаимодействие локальных СУРБД. Процесс разработки структуры БД на основании требований пользователя называют проектированием БД (ПБД). Результатом ПБД являются структура БД, состоящая из логических и физических компонент, и руководство для прикладных программистов.

Развитие системы БД во времени называют жизненным циклом. Последний делится на стадии анализа, проектирования и эксплуатации.

Первая стадия включает в себя этапы формулирования и анализа требований, концептуального проектирования, проектирования реализации, физического проектирования. Анализ требований является полностью неформализованным этапом. Его основная цель — обеспечить согласованность целей пользователей и представлений об информационных потоках.

Вторая стадия состоит из этапов реализации БД, анализа функционирования и поддержки, модификации и адаптации.

Концептуальное проектирование обеспечивает выбор структуры организации информации на основе объединения информационных требований пользователей.

Проектирование реализации (логическое проектирование) разделяют на две части: проектирование базы данных и проектирование программ. Результатом первой части является логическая структура БД. Результатом второй части считают функциональные описания программных модулей и наборы запросов к БД.

Физическое проектирование разделяют также на две части: выбор физической структуры БД и отладку программных модулей, полученных при проектировании программ. Результатом этапа является подготовка к эксплуатации БД.

На этапе реализации БД ставится задача разработки программ доступа к БД.

Этап анализа функционирования и поддержки обеспечивает статистическую обработку данных о функционировании системы. Восстановление БД и ее целостность после сбоев обеспечивает поддержка БД.

Этап модернизации и адаптации позволяет производить изменения, оптимизацию функционирования, модификацию программ.

Языки, используемые в БД, делят на языки описания данных (ЯОД) и языки манипулирования данными (ЯМД).

В общем случае ЯОД описывает различные типы записей, их имена и форматы, служит для определения: типов элементов данных, которые нужны в качестве ключей; отношений между записями или их частями и именования этих отношений; типа данных, которые используются в записях; диапазона их значений; числа элементов, их порядка и т. п.; секретности частей данных и режимов доступа к ним.

Различают три уровня абстракции для описания данных:

- концептуальный (с позиции администратора);
- реализации (с позиции прикладного программиста и пользователя);
- физический (с позиции системного программиста).

На концептуальном уровне описывают объекты, атрибуты и значения данных. На уровне реализации имеют дело с записями, элементами данных и связями между записями. На физическом уровне оперируют блоками, указателями, данными переполнения, группировкой данных.

Обычно ЯМД дают возможность манипулирования данными без знания несущественных для программиста подробностей. Они могут реализоваться как расширение языков программирования общего назначения путем введения в них специальных операторов или путем реализации специального языка (например, PL/I, ЯМД CODASYL).

При работе с БД используются несколько типов языков: манипулирования данными; программирования; описания физической организации данных.

Языки программирования, используемые в БД, представляют собой распространенные языки типа ФОРТРАН, КОБОЛ и др.

Языки описания логических схем пользователя реализуются средствами описания данных языка прикладного программирования, средствами СУБД, специальным языком.

Наиболее широко распространен первым способ описания. Он имеет в основе операторы объявления (например, DECLARE в языке PL/I, STRUCT в СИ, type в ADA).

Языки описания схем БД предназначены для администратора БД. С их помощью определяют глобальные описания данных.

Языки описания физической организации данных описывают физическую структуру размещения схемы на машинных носителях. С их помощью определяют методы доступа, предписывающие размещение данных на тех или иных носителях, и т. п.

Наибольшими преимуществами обладают специальные языки, так как не зависят от используемых языков программирования или технических средств. Следовательно, при переносе БД на другое ТО или смене языка программирования большинство описаний БД останется без изменения.

Процесс проектирования БД начинают с построения *концептуальной модели* (КМ). Концептуальная модель состоит из описания объектов и их взаимосвязей без указания способов физического хранения. Построение КМ начинается с анализа данных об объектах и связях между ними, сбора информации о данных в существующих и возможных прикладных программах. Другими словами, КМ — это модель предметной области. Версия КМ, обеспечиваемая СУБД, называется *логической моделью* (ЛМ). Подмножества ЛМ, которые выделяются для пользователей, называются внешними моделями (подсхемами). Логическая модель

отображается в физическую, которая отображает размещение данных и методы доступа. Физическую модель называют еще внутренней.

Внешние модели не связаны с используемыми ТС и методами доступа к БД. Они определяют первый уровень независимости данных. Второй уровень независимости данных связан с отсутствием изменений внешних моделей при изменении КМ. На рис. 9.10 показаны два уровня независимости данных в ИО САПР. В основном степень независимости данных определяется формализованностью проектирования БД.

Важным инструментом при разработке и проектировании БД является *словарь данных* (СД), предназначенный для хранения сведений об объектах, атрибутах, значениях данных, взаимосвязях между ними, их источниках, значениях, форматах представления. Словарь данных позволяет получить однообразную и формализованную информацию обо всех ресурсах данных. Главное назначение СД состоит в документировании данных. Он должен обеспечивать централизованное введение и управление данными, взаимодействие между разработчиками любого проекта, например, САПР. Существуют *интегрированные и независимые* СД. В первом случае СД — это часть пакета программ СУБД, а во втором — отдельный пакет программ в виде дополнения к СУБД.

В настоящее время СД рассматривают как связующее звено в системе ПО обработки данных, включающей в себя процессор, СУБД, языки запросов, монитор телеобработки. На рис. 9.11 показаны интерфейсы СД в гипотетической системе с БД. В полном объеме СД обязан: поддерживать КМ, логическую, внутреннюю и внешнюю модели; быть интегрированным с СУБД, подде-

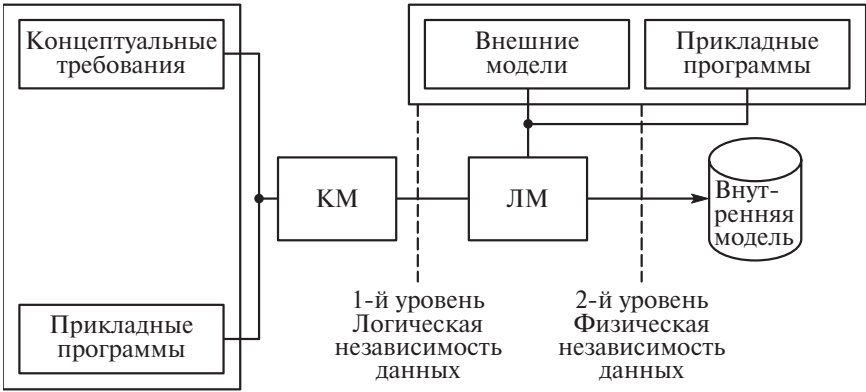


Рис. 9.10. Логическая и физическая независимость данных

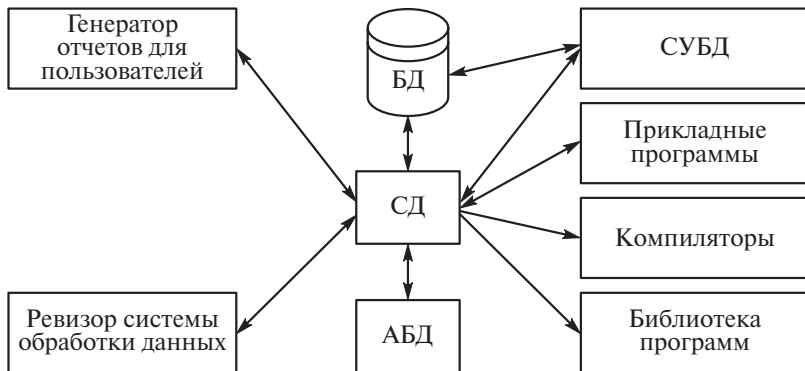


Рис. 9.11. Интерфейсные СД в гипотетической системе с БД

рживать тестовые и рабочие версии хранимых описаний; обеспечивать эффективный обмен информацией с СУБД и процесс изменения рабочей версии при изменении БД. Словарь данных должен иметь свою БД. Приведем основные составляющие БД словаря данных. Последний включает в себя: атрибут; объект; групповой элемент данных; выводимый объект данных; синонимы, т. е. атрибуты, имеющие одинаковое назначение, но различные идентификаторы; омонимы, т. е. атрибуты с различным назначением, но с одинаковыми идентификаторами; описание КМ, ЛМ, внешних и внутренних моделей; описание, позволяющее пользователям формально и однозначно выбирать атрибуты для решения задач.

По аналогии с СД создают информационный фонд. Отметим, что он может являться частью СД. Содержимое информационного фонда можно разделить на две основные части.

Первая часть содержит нормативно-справочную информацию, сведения о ГОСТ, ОСТ и т. п., а также различные модели компонентов (тепловые, электрические, коммутационные). Эта часть характеризуется многократным считыванием, сравнительно редкой обновляемостью. Иногда ее называют «постоянной» частью БД.

При проектировании БД необходимо провести идентификацию основных объектов предметной области и прикладных программ, подлежащих использованию, определить объекты и их взаимосвязи, построить СД, КМ, ЛМ, ФМ с проведением анализа и оценок (рис. 9.12).

Вторая часть содержит сведения об объектах, которые находятся в процессе проектирования. Они включают описания объ-

ектов после выполнения различных проектных операций (например, размещения, трассировки), описание конструкторской документации и т. п.

Информацию, используемую в процессе проектирования, делят условно на две группы. К первой группе относят информацию, описывающую концептуальное структурное представление. Ее называют ISP-информацией. Она не связана с конкретными способами обработки и приложениями и описывает концептуальные связи в БД.

Ко второй группе относят информацию, описывающую концептуальное представление. Ее называют UP-информацией. Она определяет требование организации к обработке данных и описывает данные и связи, используемые в приложениях.

Считается, что ISP-информация должна обеспечивать гибкость и адаптивность, UP-информация — эффективность проектирования. ISP-информацию в основном используют для построения начальной информационной структуры, UP-информацию — для ее улучшения и уточнения в конкретном приложении.

После определения физической структуры ее описание передается СУБД. До использования БД реализуют ее макет, или строят прототип, с помощью которых получают оценку характеристик БД и прогнозируют ее развитие. Если прототип не отвечает заданным требованиям, производят перепроектирование БД, в положительном случае — загрузку БД. Отметим, что построение прототипов, макетов и моделей реальной БД является сложной задачей из-за отсутствия математических методов оценки правильности модели.

При конвертировании и интеграции данных обеспечиваются возможность изменения физической структуры БД и поддержка разработанных прикладных программ управления БД.

При эксплуатации обеспечение безопасности, секретности, разграничения доступа к БД поддерживается определенными процедурами. Кроме того, вводят процедуры восстановления и повторного запуска.

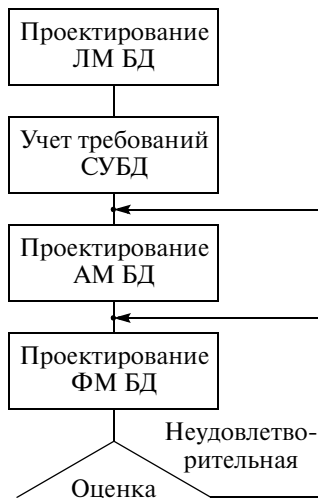


Рис. 9.12. Упрощенная структурная схема проектирования БД

9.2.5. Модели данных

Современные СУБД основываются на использовании *моделей данных* (МД), позволяющих описывать объекты предметных областей и взаимосвязи между ними. Существуют три основные МД и их комбинации, на которых основываются СУБД: *реляционная модель данных* (РМД), *сетевая модель данных* (СМД), *иерархическая модель данных* (ИМД).

Основное различие между этими МД состоит в способах описания взаимодействий между объектами и атрибутами. Взаимосвязь выражает отношение между множествами данных. Используют взаимосвязи «один к одному», «один ко многим» и «многие ко многим». «Один к одному» — это взаимно однозначное соответствие, которое устанавливается между одним объектом и одним атрибутом. Например, в определенный момент времени с одной ЭВМ используется один определенный процессор. Номеру выбранной ЭВМ соответствует номер выбранного процессора. «Один ко многим» — это соответствие между одним объектом и многими атрибутами. «Многие ко многим» — это соответствие между многими объектами и многими атрибутами. Например, на множество ЭВМ может одновременно работать множество пользователей. Взаимосвязи между объектами и атрибутами удобно представлять в виде графов и гиперграфов.

Реляционная модель данных. В РМД объекты и взаимосвязи между ними представляют в виде таблиц. Таблица, состоящая из строк и столбцов, называется отношением. Каждый столбец в таблице является атрибутом. Строки таблицы являются кортежами, т. е. упорядоченными множествами. Значения в столбце определяют из множества значений, которые принимает атрибут. Столбцы таблицы — это элементы данных, а строки — записи. Построим, например, таблицу для представления данных с помощью РМД (табл. 9.2).

Таблица 9.2

Номер	Этапы проектирования (имя)	Атрибуты	
		Сложность	Время реализации
1	Покрытие	$O(n^3)$	минуты
2	Типизация	$O(n^2)$	секунды
3	Компоновка	$O(n^3)$	минуты
4	Размещение	$O(n^3)$	минуты
5	Трассировка	$O(n^3 - n^4)$	часы
6	Контроль	$O(n^3)$	минуты

Первичным ключом в таблице является номер этапа проектирования. Таблица имеет два атрибута и шесть кортежей.

Приведем еще один пример представления данных с помощью таблицы ПОЛЬЗОВАТЕЛЬ (табл. 9.3).

Таблица 9.3

Идентификационный номер	Фамилия
236	Сидоров
185	Иванов
349	Петров
499	Кочетков
777	Харламов

Эта таблица представляет данные с помощью реляционной модели. Первичным ключом является идентификационный номер пользователя.

Основное достоинство реляционного подхода — его простота и доступность. Пользователи абстрагированы от физической структуры памяти. Это позволяет эксплуатировать БД без знания методов и способов ее построения. Основные достоинства РМД следующие: простота, независимость данных; гибкость; непроектурные запросы, теоретическое обоснование на основе теории отношений.

Основные недостатки РМД: низкая производительность по сравнению с НМД и СМД, сложность ПО, избыточность. Кроме того, проблематично обеспечение полного набора функциональных возможностей и необходимых операционных характеристик при обработке больших БД.

Иерархическая модель данных. Она основана на понятии деревьев, состоящих из вершин и ребер. Вершина дерева ставится в соответствие совокупности атрибутов данных, характеризующих некоторый объект. Вершины и ребра дерева как бы образуют иерархическую древовидную структуру (ИДС), состоящую из n уровней (рис. 9.13.).

Первую вершину в дереве называют корневой вершиной ИДС. Она удовлетворяет семи условиям: 1. Иерархия начинается с корневой вершины. 2. Каждая вершина соответствует одному или нескольким атрибутам. 3. На уровнях с большим номером находятся зависимые вершины. Вершина предшествующего уровня является начальной для новых зависимых вершин.

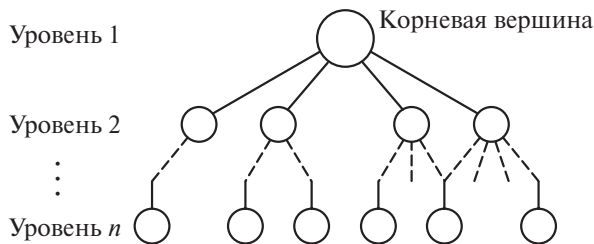


Рис. 9.13. Иерархическая древовидная структура

4. Каждая вершина, находящаяся на уровне i , соединена с одной и только одной вершиной уровня $i - 1$, за исключением корневой вершины. 5. Корневая вершина может быть связана с одной или несколькими зависимыми вершинами. 6. Доступ к каждой вершине происходит через корневую по единственному пути. 7. Существует произвольное количество вершин каждого уровня.

Иерархическая модель данных состоит из нескольких деревьев, т. е. является лесом. Каждая корневая вершина образует начало записи логической базы данных. В ИМД вершины, находящиеся на уровне i , называют порожденными вершинами на уровне $i - 1$. Ниже приведен пример представления информации в ИМД, реализующей отношение «один ко многим»:



Для каждого пользователя может иметься экземпляр корневой вершины. Иерархическая модель данных позволяет для каждого пользователя получать представление о нескольких опера-

циях и нескольких ЭВМ. ПОЛЬЗОВАТЕЛЬ соответствует корневой вершине и находится на более высоком уровне иерархии, чем ЭВМ, ОПЕРАЦИЯ и РЕЗУЛЬТАТ.

Отметим, что выбор ИМД осуществляет администратор БД на основе операционных характеристик. Введение двух ИМД, связанных между собой, позволяет решать вопросы включения и удаления данных. Основные достоинства ИМД — простота построения и использования, обеспечение определенного уровня независимости данных, наличие существующих СУБД, простота оценки операционных характеристик. Основные недостатки: отношение «многие ко многим» реализуется очень сложно, дает громоздкую структуру и требует хранения избыточных данных, что особенно нежелательно на физическом уровне; иерархическая упорядоченность усложняет операции удаления и включения; доступ к любой вершине возможен только через корневую, что увеличивает время доступа.

Сетевые модели данных. В СМД элементарные данные и отношения между ними представляются в виде ориентированной сети (вершины — данные, дуги — отношения). База данных, описываемая сетевой моделью, состоит из нескольких областей. Область содержит записи. Одна запись состоит из нескольких полей. Набор, состоящий из записей, может размещаться в одной или нескольких областях (рис. 9.14).

В СМД объекты предметной области объединяются в сеть. Графически сетевая модель описывается прямоугольниками и

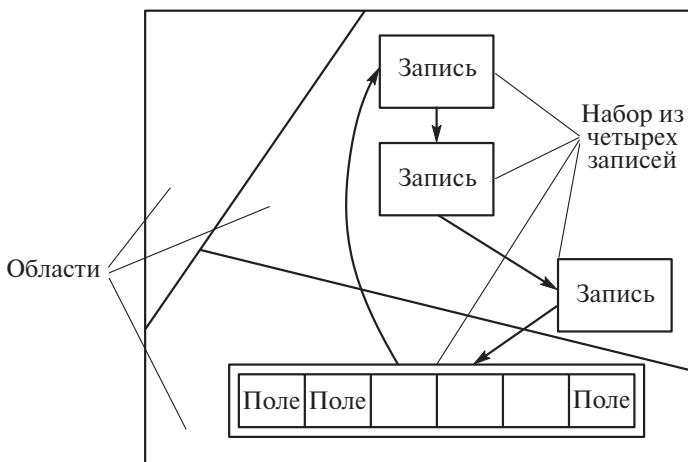


Рис. 9.14. База данных, описываемая сетевой моделью

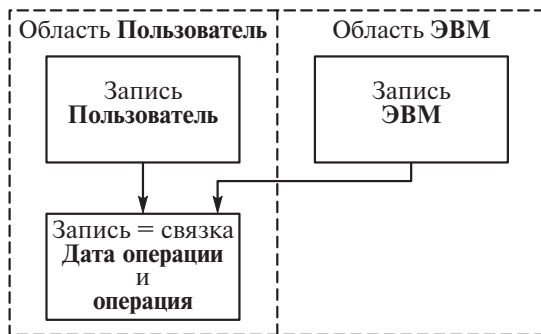


Рис. 9.15. Область СМД

стрелками. Каждый тип записи может содержать множество атрибутов. На рис. 9.15 показан пример представления области в СМД. Здесь область — это часть БД, в которой располагаются записи, наборы и части наборов. Стрелками соединены несколько типов записей, изображающих типы набора. Тип набора представляет логическую взаимосвязь «один ко многим».

9.3. Программное обеспечение автоматизированного управления

9.3.1. Общее и специальное программное обеспечение

Программное обеспечение (ПО) любого автоматизированного управления в основном определяет его интеллектуальные возможности, профессиональную направленность, широту и полноту управления. Комплекс программ автоматизированного управления должен охватывать множество функций: обеспечение организации диалога и решения функциональных задач пользователя, управление базами данных и трансляцию программ, выдачу справочной, диагностической информации и осуществление сервисных операций, облегчающих работу на ПЭВМ. Развитость программного обеспечения определяет функциональную полноту и разнообразие сервисной поддержки для пользователя. Поэтому если классифицировать АРМ с позиции программных средств, то признаками такой классификации могут быть:

- языки программирования;
- наличие СУБД;
- вид транслятора (интерпретатор или компилятор) с языков программирования;

- средства обнаружения и исправления ошибок;
- возможность достраивания программной системы и т.д.

Программное обеспечение любого вида автоматизированного управления подразделяется на *общее* и *специальное* (рис. 9.16).

Основные элементы общего программного обеспечения обычно поставляются вместе с ПЭВМ. К ним относятся операционные системы (ОС) и оболочки, программные средства ПС ведения баз данных, программные средства организации диалога, а также программы, расширяющие возможности ОС. Главное предназначение этой части ПО — управление работой процессора, организация интерфейса между пользователем и ПЭВМ, организация доступа к памяти, периферийным устройствам и сети, управление файлами, запуск прикладных программ и управление процессом их выполнения, трансляция и выполнение программ, подготовленных на алгоритмических языках (Паскаль, PL/1, Бейсик и др.).

Специальное программное обеспечение автоматизированного управления обычно состоит из уникальных программ и функциональных пакетов прикладных программ (ППП). Именно от функционального ПО зависят вид, содержание и конкретная специализация автоматизированного управления. Учитывая, что специальное ПО автоматизированного управления в конечном счете определяет область применения автоматизированных рабочих мест, состав решаемых пользователем задач, оно должно создаваться на основе инструментальных программных средств диалоговых систем, ориентированных на решение конкретного

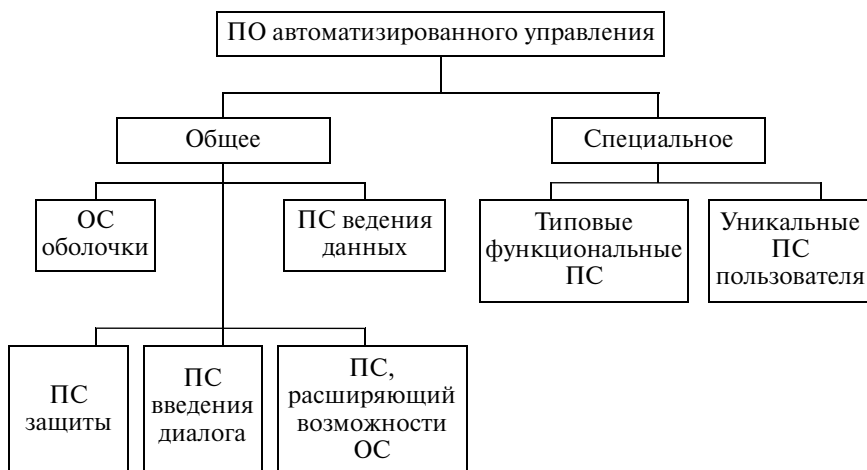


Рис. 9.16. Принципиальный состав программного обеспечения автоматизированного управления

класса задач со схожими функционально-технологическими особенностями обработки информации.

ПО автоматизированного управления должно обладать свойствами адаптивности, гибкости, модифицируемости и настраиваемое на конкретное применение в соответствии с требованиями пользователя. Непременным составным элементом программного обеспечения, передаваемого пользователю, является документация на него, которая облегчает выяснение сути решаемых задач, а также внедрение, эксплуатацию и сопровождение программного продукта.

Остановимся несколько подробнее на составных частях ПО автоматизированного управления, созданных на базе ПК. В качестве ОС автоматизированного управления 16-разрядных ПЭВМ обычно используется MS DOS, на базе 32-разрядных — OS/2 и UNIX. Это наиболее распространенные операционные системы, хотя имеются и другие.

9.3.2. Пакеты прикладных программ автоматизированного управления

Наиболее динамично развивающейся частью программного обеспечения являются пакеты прикладных программ. Круг решаемых с помощью ППП задач постоянно расширяется. Во многом внедрение компьютеров практически во все сферы деятельности стало возможным благодаря появлению новых и совершенствованию существующих ППП.

Все ППП могут быть разбиты на три группы:

1. Пакеты, расширяющие возможности операционных систем.
2. Пакеты общего назначения.
3. Пакеты, ориентированные на работу в автоматизированных системах управления.

Пакеты прикладных программ, расширяющие возможности операционных систем, обеспечивают функционирование ЭВМ различных конфигураций. К ним относятся пакеты, обеспечивающие работу многомашинных комплексов типовых конфигураций, диалоговые системы, системы для работы в реальном масштабе времени, удаленную пакетную обработку.

ППП общего назначения включают в себя набор программ для широкого круга применений для алфавитно-цифровых и графических дисплеев, графопостроителей, систем программирования, а также для научно-технических расчетов, математического программирования, обработки матриц, различного вида моделирования, решения задач теории массового обслуживания и т.д.

Пакеты, ориентированные на работу в АСУ, включают в себя набор программ для общецелевых систем обработки банков данных, информационных систем общего назначения, систем обработки документов.

Достижение в области микроэлектроники, приводящие к появлению более мощных по своим функциональным возможностям компьютерам, также является причиной создания новых ППП. В свою очередь, необходимость улучшения характеристик использования пакета при решении конкретных задач пользователя стимулирует совершенствование архитектуры и элементной базы компьютеров и периферийных устройств.

Структура и принципы построения ППП зависят от класса ЭВМ и операционной системы, в рамках которой этот пакет будет функционировать. Наибольшее количество разнообразных ППП создано для IBM PC-совместимых компьютеров с операционными системами MS DOS и WINDOWS. Классификация этих пакетов программ по функциональному признаку представлена на рис. 9.17.

Каждая группа пакетов имеет свои проблемы организации, трудности разработки и создания. Каждый пакет, в зависимости от ЭВМ и его назначения реализуется на конкретном языке программирования в соответствии с требованиями, предъявленными к пакету, и возможностями языка.

В приведенной классификации не указаны игровые программы — они не являются средством для автоматизации, профессиональной деятельности и предназначены для досуга. Отсутствие программ — переводчиков, орфографии, электронных словарей связано с тем, что эти программы являются функциональным дополнением ППП типа редактора текста, презентации и т.п. Наблюдается тенденция включения этих программ в состав прикладных пакетов.

Существующие ППП охватывают почти все сферы человеческой деятельности, связанные с обработкой информации. Развитие и совершенствование ППП — поступательный процесс, поэтому следует ожидать появления новых ППП, возможности которых превзойдут достижения настоящих пакетов.

Проблемно-ориентированные ППП — наиболее развиты в плане реализуемых функций и многочисленна по количеству созданных пакетов часть ППП. Она включает следующие проблемно-ориентированные программные продукты: текстовые процессоры, издательские системы, графические редакторы, демонстрационную графику, системы мультимедиа, ПО САПР, организаторы работ, электронные таблицы (табличные процессоры), системы управ-

ления базами данных, программы распознавания символов, финансовые и аналитико-статистические программы.

Текстовые процессоры — специальные программы, предназначенные для работы с документами (текстами), позволяющие компоновать, форматировать тексты при создании пользователем документа. Обычно они включают в себя дополнительные функции по работе с блоками текста и объектами. Признанными лидерами в части текстовых процессоров для ПЭВМ являются MS WORD, WordPerfect, AmiPro.

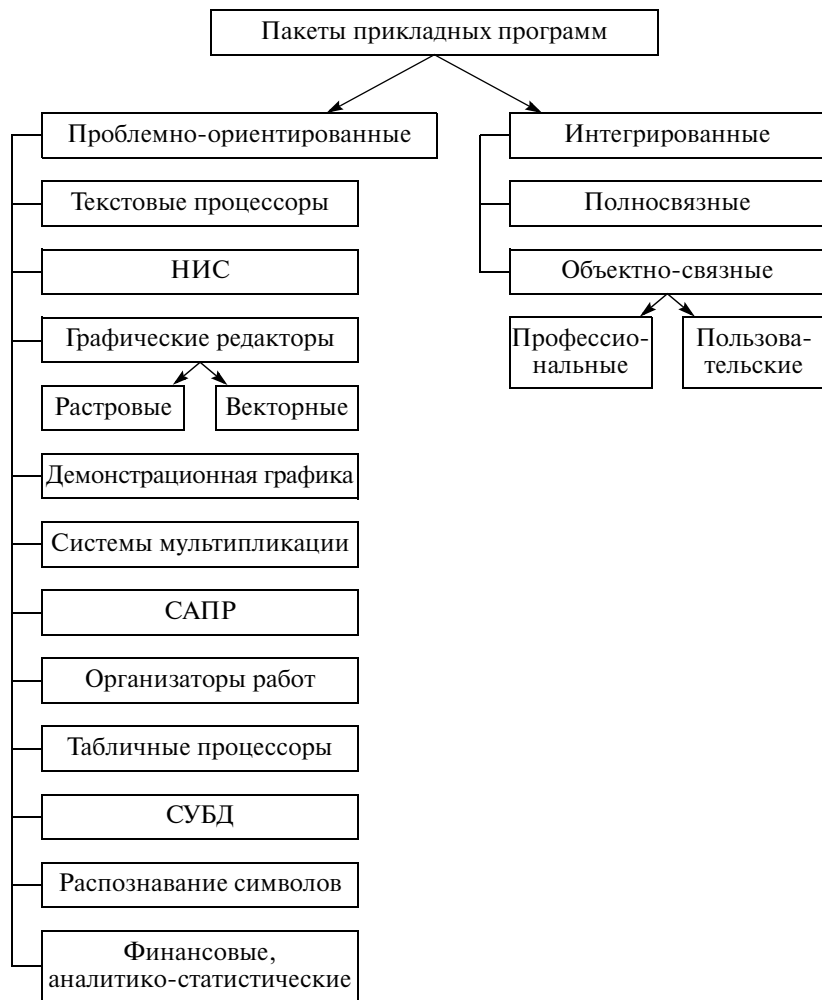


Рис. 9.17. Классификация ППП

Настольные издательские системы (НИС) — программы, предназначенные для профессиональной издательской деятельности и позволяющие осуществить электронную верстку широкого спектра основных типов документов, типа информационного бюллетеня, краткой цветной брошюры и объемного каталога или торговой заявки, справочника. Предусмотренные в пакетах данного типа средства позволяют:

- компоновать (верстать) текст;
- использовать всевозможные шрифты и осуществлять полиграфическое изображение;
- осуществлять редактирование текста на уровне лучших текстовых процессоров;
- обрабатывать графические изображения;
- обеспечивать вывод документов полиграфического качества;
- работать в сетях и на разных платформах.

Наилучшими пакетами в этой области для ПЭВМ являются Corel Ventura, PageMaker, QuarkXPress, FrameMaker, Microsoft Publisher, PagePlus, CompuWork Publisher.

Графические редакторы-пакеты, предназначенные для обработки графической информации, делятся на ППП обработки растровой графики и векторной графики.

ППП обработки растровой графики предназначены для работы с фотографиями и включают в себя набор средств по кодированию фотоизображений в цифровую форму. Признанный лидер среди пакетов данного класса — Adobe Photoshop. Известны также пакеты Picture Publisher, Photo Works Plus. Все программы ориентированы на работу в среде Windows.

Пакеты для работы с векторной графикой предназначены для профессиональной работы, связанной с художественной и технической иллюстрацией, с последующей цветной печатью (на рабочем месте дизайнеров, например). Они занимают промежуточное положение между пакетами для систем автоматизированного проектирования (САПР) и настольными издательскими системами. Пакеты данного класса в настоящее время обладают достаточно широким набором функциональных средств для осуществления сложной точной обработки графических изображений.

Своеобразным стандартом в этом классе является пакет CorelDraw. Можно также отметить Aldus Free Hand, Freelance Graphics.

Пакеты прикладных программ мультимедиа предназначены для использования ПЭВМ для отображения и обработки аудио- и видеоинформации. Помимо программных средств, компьютер при этом должен быть оборудован дополнительными платами,

позволяющими осуществлять ввод-вывод аналоговой информации, ее преобразование в цифровую форму.

Программы мультимедиа для ПЭВМ появились сравнительно недавно благодаря значительному росту вычислительных возможностей ПК и большим достижениям в области производства оптических дисков. Дело в том, что при представлении аналоговой информации в цифровом виде требуются огромные объемы памяти: несколько минут видеофильма занимают десятки мегабайт памяти. Естественно, что работа с таким большим файлом возможна лишь при наличии быстродействующего процессора (желательно использовать ПК с RISC-процессором и быстродействующей шиной данных). Кроме того, распространение таких мультимедиа-приложений невозможно на традиционных магнитных дискетах, для этого следует использовать оптические компакт-диски (CD-ROM).

Среди мультимедиа-программ можно выделить две небольшие группы.

Первая включает *пакеты для обучения и досуга*. Поставляемые на CD-ROM емкостью от 200 до 500 Мбайт каждый, они содержат аудиовизуальную информацию по определенной тематике. Разнообразия их огромно, и рынок этих программ постоянно расширяется при одновременном улучшении качества видеоматериала. Так, созданы и продаются электронные энциклопедии по отраслям знаний, электронные учителя в области иностранных языков, бизнеса, политики, деловые и авантюрные игры.

Вторая группа включает *программы для подготовки видеоматериалов* для создания мультимедиа-представлений, демонстрационных дисков и стендовых материалов.

К пакетам данного типа относятся Director for Windows, Multimedia Viewer Kit, NEC MultiSpin.

Другая разновидность пакетов программ, связанная с обработкой графических изображений, — *системы автоматизации проектирования*. Они предназначены для автоматизации проектно-конструкторских работ в машиностроении, автомобилестроении, промышленном строительстве и т.д. Пакеты САПР обладают набором инструментальных средств, обеспечивающих реализацию основных функций:

- коллективная работа в сети пользователей с пакетом;
- экспорт-импорт файлов различных форматов;
- масштабирование объектов;
- управление объектами в части их группировки, передвижения с растяжкой, поворота, разрезания, изменения размеров, работа со слоями;

- перерисовка (фоновая, ручная, с прерываниями);
- управление файлами в части библиотек и каталогов чертежей;
- использование разнообразных чертежных инструментов, позволяющих рисовать кривые, эллипсы, произвольной формы линии, многоугольники и т.п., использование библиотеки символов, выполнение надписей и т.д.;
- работа с цветом;
- автоматизация отдельных процедур с использованием встроенного макроязыка.

Своеобразным стандартом среди программ данного класса являются пакеты AutoCad, DesignCad, Grafic Cad Professional, DrawBase, Microstations, Ultimate Cad Base, Turbo Cad. Перечисленные пакеты отличаются богатством функциональных возможностей и предназначены для функционирования в среде Windows.

Организаторы работ — это пакеты программ, предназначенные для автоматизации процедур планирования использования различных ресурсов (времени, денег, материалов) как отдельного человека, так и всей фирмы или ее структурных подразделений. Целесообразно выделить две разновидности пакетов данного класса: управления проектами и организации деятельности отдельного человека.

Пакеты первого типа предназначены для сетевого планирования и управления проектами. Достаточно простые и удобные в использовании, эти программные средства позволят быстро спланировать проект любой величины и сложности, эффективно распределить людские, финансовые и материальные ресурсы, составить оптимальный график работ и проконтролировать его исполнение.

К пакетам данного типа относятся Time Line, Ms Project, CA-Super Project.

Пакеты второго типа представляют собой своего рода электронных помощников делового человека. Такие пакеты, как Lotus Organiser, ASTI, выполняют функции электронных секретарей и предназначены для эффективного управления деловыми контактами.

Электронные таблицы (табличные процессоры) — пакеты программ, предназначенные для обработки табличным образом организованных данных. Пользователь имеет возможность с помощью средств пакета осуществлять разнообразные вычисления, строить графики, управлять форматом ввода-вывода данных, компоновать данные, форматировать, проводить аналитические исследования и т.д.

В настоящее время наиболее популярными и эффективными являются пакеты Excel, Improv, Quattro Pro, Lotus 1-2-3.

Системы управления базами данных (СУБД) — предназначены для автоматизации процедур создания, хранения и извлечения электронных данных. Многие существующие экономические, информационно-справочные, банковские, программные комплексы реализованы с использованием инструментальных средств СУБД.

Для различных классов компьютеров и операционных средств разработано множество СУБД, отличающихся по способу организации данных, формату данных, языку формирования запросов. Наиболее распространенными пакетами для ПЭВМ типа IBM PC являются dBase, Paradox, FoxBase, FoxPro, Clipper, Microsoft Access и др.

Выбор СУБД определяется многими факторами, но главный из них — возможность работы с построенной моделью данных. Поэтому одной из важнейших характеристик является тип модели (иерархический, сетевой, реляционный), который поддерживается СУБД. Имеются системы для работы с разными моделями, однако большинство СУБД для персональных ЭВМ работают с реляционной моделью. Реляционные СУБД различаются набором реляционных операций, которые они могут выполнять. Перечисленные СУБД эффективны для создания небольших изолированных систем с несложной структурой данных, с относительно небольшими объемами данных (10—30 Мбайт) и несложными запросами. За пределами такого рода ограничений эффективность использования указанных СУБД существенно снижается.

Удобство и комфортность работы пользователя с СУБД во многом определяются пользовательским интерфейсом.

Пользовательский интерфейс — это средство и часть СУБД, ориентированные на взаимодействие пользователя с компьютерной системой. Благодаря разветвленному иерархическому меню, всевозможным подсказкам и разнообразной помощи, пользователю легко ориентироваться в выборе действий адекватных возникающей в работе ситуации.

Очень важна в интерфейсе минимизация действий пользователя, необходимых для подключения часто требуемых функций. Для этой цели применяют функциональные клавиши. Их нажатие вызывает исполнение программных модулей, которые реализуют требуемую функцию.

Пользовательский интерфейс может быть многоуровневым — рассчитанным на более широкий круг разнообразных пользователей. Благодаря дружественному характеру интерфейса пользователь избавляется от необходимости знать язык программирования системы, чем достигается более высокая его производительность.

Сочетанием простоты освоения и использования функциональных возможностей с помощью простого интерфейса обеспечивается широкая сфера применения таким массовым СУБД. При усложнении информационных потребностей пользователя возникает необходимость в более развитых СУБД и в знании языка программирования используемой СУБД.

Большинство информационных систем в нашей стране используют простейшие СУБД, которые функционируют на персональных компьютерах. Такие системы покрывают первоначальные потребности организаций, но они не перспективны.

Базой систем нового поколения являются профессиональные (многопользовательские, многоплатформные) СУБД и архитектура «клиент-сервер», реализуемая на их основе.

Профессиональные СУБД обеспечивают выполнение более сложных операций. Они позволяют разработчику расширять сервисные возможности — процедуры баз данных, которые вызываются клиентом и выполняются сервером более производительнее, чем компьютеры на рабочих местах пользователей.

К профессиональным СУБД относятся Oracle, SyBase, Informix, Ingres, Progress. Перечисленные системы имеют средства обработки информации, распределенной по нескольким узлам сети. Распределенная обработка данных позволяет разместить базу в различных узлах таким образом, чтобы отслеживать изменения на всех узлах и чтобы каждый компонент данных располагался на том узле, где он будет обрабатываться. Новейшей технологией управления распределенными базами данных является тиражирование. Профессиональные СУБД поддерживают те или иные механизмы тестирования.

Особенностью современных информационных систем, например, биржевых или банковских, является требование оперативного оповещения пользователей о происходящих событиях, например, все участники фондовой биржи должны немедленно получать информацию о совершенных сделках, изменениях котировок и т.д.

Другими словами, предполагается наличие некоторого количества процессов, которые должны использоваться параллельно и синхронизироваться во времени исполнения. Это приводит к необходимости обмена информацией между ними.

Профессиональные СУБД типа Oracle позволяют организовать эти процессы в виде отдельных приложений на одной базе данных. Например, при совершении сделки процесс, занимающийся их регистрацией, возбуждает событие «возбуждена сделка». Результаты ее включаются в общий поток информации о

сделках. Если же этот процесс не исполняется, то событие «совершена сделка» не приводит ни к каким дополнительным действиям. Механизмы событий, реализованные в современных профессиональных СУБД, являются готовым технологическим средством, которое позволяет разработчикам информационных систем экономить значительное количество времени и усилий.

По мере развития любой хозяйственной деятельности появляется потребность в наращивании информационной системы. Возникает вопрос, как встроить имеющееся локальное приложение в новую систему. Профессиональные СУБД предоставляют достаточно широкие возможности. Развитые системы шлюзов позволяют строить информационные системы, распределенные по узлам с различными аппаратными и программными платформами. Большой интерес представляет также использование локальными приложениями, так называемого, ODBC-стандарта (Open Database Connectivity стандарт, предложенный фирмой Microsoft), который дает возможность прозрачного доступа к данным СУБД различных типов. Таким образом, приложение, разработанное с учетом стандарта ODBC, имеет большую гибкость при интеграции в существующую информационную систему.

Потребность в гибких решениях для современных информационных систем диктуется жизнью. На практике чаще всего встречается потребность в объединении возможностей отдельных подсистем или программных модулей. Причем все это нужно иметь на одной базе данных. Через некоторое время соотношение потребностей может измениться. Поэтому для построения информационной системы важно иметь инструмент, который наиболее приспособлен для построения открытых и гибких систем. Таким инструментом в настоящее время являются профессиональные СУБД SQL, обеспечивающие работу в модели «клиент—сервер» и обладающие развитыми средствами разработки и сопровождения баз данных. Использование профессиональной СУБД позволяет иметь программное обеспечение, в большей степени отвечающее конкретным потребностям организации.

Защита данных от несанкционированного доступа к профессиональной СУБД обеспечивается на разных уровнях:

1. Операционная система поддерживает разграничение прав доступа пользователей.
2. СУБД предоставляет свое разграничение прав доступа.
3. Защита данных средствами приложения — еще один уровень, который может быть настолько развитым и многообразным, насколько хватит фантазии у программиста, разрабатывающего приложение.

СУБД поддерживает достаточно сложную структуру таблиц. Требования к непротиворечивости данных в этих таблицах довольно жестки. Рассмотрим типичный пример, включающий счета клиентов в журнал сделок. Информация о ценных бумагах, находящихся на счетах клиентов, должна соответствовать информации, находящейся в журнале сделок. Занесенные в журнал данные с начала функционирования системы до настоящего времени должны привести к текущему состоянию таблицы счетов. Проверка этого соответствия — операция длинная и трудоемкая. Кроме того, если проверка показала несоответствие, то возникает следующий вопрос: где произошло рассогласование? Какая информация правильна: о счетах или о сделках? Ответить на эти вопросы практически невозможно, если разработчики информационных систем не предприняли специальных усилий для поддержания *ссылочной целостности базы данных*.

Методы поддержания целостности данных известны. Это — ведение журналов изменения таблиц и обработка транзакций. Различие между персональными и профессиональными СУБД здесь в том, что в первом случае разработчик должен брать их реализацию на себя, а во втором они уже реализованы внутри СУБД.

Заметим также, что профессиональные СУБД предоставляют средства восстановления базы данных, если нарушение целостности все же произошло, например, при сбое питания.

Современные профессиональные СУБД поддерживают средства, значительно ускоряющие разработку программы. Это языки четвертого поколения, интегрирующие средства высокого уровня для создания интерфейса с элементами CASE-технологии, средства для организации сложных запросов к базе данных, возможности подключения фрагментов, написанных на языках низкого уровня, поддержка SQL-интерфейса. Все это ускоряет разработку приложений. Реализация интерфейса запросов к базе данных занимает минимум времени и усилий. Это позволяет разработчику сосредоточить усилия на предметной области.

Профессиональные СУБД, поддерживающие технологию «Клиент — сервер», позволяют наиболее эффективно использовать имеющийся парк персональных компьютеров за счет превращения их в рабочие места пользовательской системы. Таким образом, выигрыш получается по трем направлениям: во-первых, наиболее эффективно задействуется мощный процессор сервера; во-вторых, освобождается от ненужной нагрузки сеть; в-третьих, отпадает необходимость в высокопроизводительных компьютерах на рабочих местах пользователей.

Программы распознавания символов предназначены для перевода графического изображения букв и цифр в ASCII — коды этих символов. Используются, как правило, совместно со сканерами.

Пакеты данного типа обычно включают разнообразные средства, облегчающие работу пользователя и повышающие вероятность правильного распознавания.

Скорость сканирования современных ППП составляет примерно 1,5 мин на страницу. К пакетам данного типа относятся File Reader, CunieForm, Tiger, OmniPage.

Разнообразными пакетами представлена группа финансовых программ для ведения деловых записей типа записной книжки и расчета финансовых операций (баланс денежных средств, определение процентных выплат по займам и кредитам, временная структура денежных вложений и т.п.).

Для расчета величины налогов можно использовать программы Turbo Tax for Windows, Personal Tax Edge.

С помощью программ Quicken, DacEasy Accounting, Peachtree for Windows можно автоматизировать бухгалтерский учет. Эту же функцию выполняет ряд отечественных программ: «Турбобухгалтер», «1С Бухгалтерия», «Бухгалтер» фирмы «Атлант-Информ» и др.

Для аналитико-статистических исследований используются хорошо зарекомендовавшие себя зарубежные статистические пакеты, такие, как StatGraphics или Statistics, или отечественная разработка «Статистик-Консультант». Однако в коммерческих фирмах внедрение этих пакетов сдерживается отсутствием соответствующим образом подготовленных специалистов, высокой ценой пакетов и широким внедрением табличных процессоров, с помощью которых можно провести простейшие, но наиболее часто используемые статистические расчеты.

Интегрированные пакеты программ — по количеству наименований продуктов немногочисленная, но в вычислительном плане довольно мощная и активно развивающаяся часть ПО.

Идея создания интегрированных программных комплексов не нова и в той или иной мере была реализована на всех поколениях ЭВМ.

Внимание к их проблеме объясняется как расширением сферы применения вычислительной техники, так и стремлением фирм-разработчиков программного обеспечения не «потерять» своих клиентов с переходом на более совершенные системы обработки данных.

Традиционные, или полностью связанные, интегрированные комплексы представляют собой многофункциональный автономный пакет, в котором в одно целое соединены функции и возможнос-

ти различных специализированных (проблемно-ориентированных) пакетов, родственных в смысле технологии обработки данных на отдельном рабочем месте. Типичными представителями таких программ являются пакеты FrameWork, Symphony, а также пакеты нового поколения Microsoft Word, Lotus Works.

В этих программах происходит интеграция функций редактора текстов, системы управления базами данных и табличного процессора. В целом, стоимость такого пакета гораздо ниже суммарной стоимости аналогичных специализированных пакетов.

В рамках интегрированного пакета обеспечивается связь между данными, однако при этом сужаются возможности каждой компоненты по сравнению с аналогичными специализированными пакетами. Интерфейс более ранних программ был перегружен различными средствами обмена данными и описаниями средств работы, что требовало от пользователя определенных навыков и знаний в части переключения режимов пакета, форматов данных, принципов хранения и манипулирования различными типами данных, что в конечном счете снижало привлекательность пакетов. В современных пакетах (например, Microsoft Works) этот недостаток изжит: простота интерфейса позволяет применять его без предварительного обучения персонала.

В настоящее время активно реализуется другой подход интеграции программных средств: объединение специализированных пакетов в рамках единой ресурсной базы, обеспечения взаимодействия приложений (программ пакета) на уровне объектов и единого упрощенного центра переключения между приложениями. Интеграция в этом случае носит *объектно-связанный характер*.

Типичные и наиболее мощные пакеты данного типа: Borland Office for Windows, Lotus SmartSuite for Windows, Microsoft Office. В профессиональной редакции этих пакетов присутствуют четыре приложения: текстовый редактор, СУБД, табличный процессор, программы демонстрационной графики. Целесообразность создания таких пакетов, очевидно, связана с желанием получить дополнительный эффект от интеграции по отношению к простой сумме составляющих его компонент. Этот эффект должен достигаться за счет согласованного взаимодействия компонент в процессе работы пользователя. При традиционном подходе к интеграции программ этот выигрыш может быть легко сведен на нет отсутствием нужной пользователю функции, присутствующей в специализированном пакете, и необходимостью в пусть небольшом, но дополнительном обучении.

Особенностью нового типа интеграции пакетов является *использование общих ресурсов*. Здесь можно выделить три основных вида совместного доступа к ресурсам:

- использования утилит, общих для всех программ комплекса. Так, например, утилита проверки орфографии доступна из всех программ;
- реализация простого метода перехода (или запуска) из одного приложения к другому;
- реализация построенных на единых принципах средств автоматизации работ с приложением (макроязыка), что позволяет организовать комплексную обработку информации при минимальных затратах на программирование и обучение программированию на языке макроопределений.

Совместное использование объектов несколькими приложениями — краеугольный камень современной технологии интеграции программ и манипулирования данными. Разработаны два основных стандарта в этой области:

динамической компоновки и встраивания объектов Object Linking and Embedding 2.0 фирмы Microsoft (OLE);

OpenDoc (открытый документ) фирм Apple, Borland, IBM, Novell и WordPerfect.

Механизм динамической компоновки объектов дает возможность пользователю помещать информацию, созданную одной прикладной программой, в документ, формируемый в другой. При этом пользователь может редактировать информацию в новом документе средствами того продукта, с помощью которого этот объект был создан (при редактировании автоматически запускается соответствующее приложение). Запущенные приложения и программы обработки документа-контейнера выводит на экран согласованные меню, часть пунктов которых принадлежит одной программе, а другая часть — другой.

Кроме того, данный механизм позволяет переносить OLE-объекты из окна одной прикладной программы в окно другой.

В этой технологии предусмотрена также возможность общего использования функциональных ресурсов, например, модуль построения графиков табличного процессора может быть использован в текстовом редакторе.

Недостатком данной технологии является ограничение на размер вставляемого объекта размером одной страницы.

OpenDoc представляет собой объектно-ориентированную систему, базирующуюся на открытых стандартах фирм—участников разработки. В качестве модели объекта используется распределен-

ная модель системных объектов (DCOM-Distributed Component Object Model), разработанная фирмой IBM для OS/2. Предполагается совместимость между OLE и OpenDoc.

Самым популярным набором офисных приложений является интегрированный пакет Microsoft Office.

Microsoft Office объединяет удобные и простые в использовании интеллектуальные приложения, обеспечивающие автоматизацию работы и поддержку пользователя, помогающие сократить время выполнения регулярных повседневных задач.

Семейство Microsoft Office обладает широкой встроенной поддержкой технологий Интернета. В результате Microsoft Office является наилучшим набором инструментов для создания и управления интранет-документами, а также для осуществления быстрого и удобного доступа к данным в интранет-сетях.

Microsoft Office разработан с учетом необходимости обеспечить пользователю возможность простой установки, конфигурирования, а администраторам сетей — возможность эффективного управления процессом установки и эксплуатации информационных систем и решений Microsoft Office.

В последнее время в составе Microsoft Office появились новые программные элементы — ассистенты.

Office Assistant. При работе со сложным программным обеспечением серьезной проблемой является поиск информации о выполнении той или иной операции. Чтобы обеспечить пользователю Microsoft Office простой и удобной системой помощи, в новую версию Microsoft Office был включен уникальный элемент — Office Assistant.

Он помогает быстро найти ответы на большинство возникающих вопросов и в случае необходимости подсказывает, как выполняется та или иная операция, и даже предлагает помощь в организации более эффективной работы.

Office Assistant удачно вписывается в интерфейс Microsoft Office.

9.3.3. Организация разработки программных средств

Решение задачи на ЭВМ представляет собой процесс получения информации на основе обработки исходной с помощью программы, составленной из команд системы управления работой отдельных устройств вычислительной машины. Создание такой программы решения задачи предполагает выполнение ряда последовательных этапов технологического процесса, принципиальная схема которого представлена на рис. 9. 18.

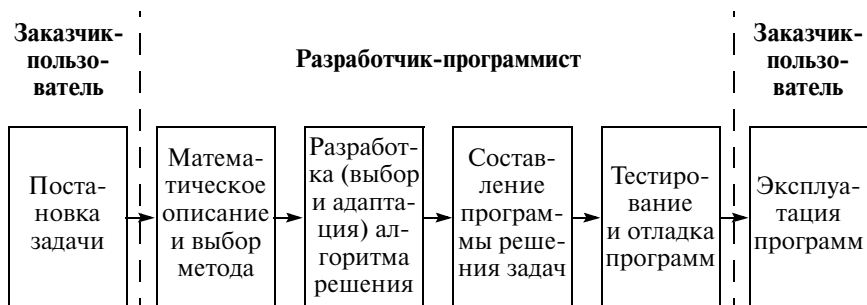


Рис. 9.18. Принципиальная схема технологического процесса разработки программных средств решения экономических задач

В зависимости от специфических особенностей конкретной задачи (ее вычислительной и логической сложности, состава и структуры исходной, промежуточной и результатной информации и т.п.), профессионального уровня подготовки специалистов и ряда других факторов некоторые этапы технологического процесса, представленные в общей схеме, могут быть объединены в более крупные этапы.

Первый этап технологического процесса представляет собой *постановку задачи*. На этом этапе раскрывается организационно-экономическая сущность задачи, т.е. формулируется цель ее решения; определяется взаимосвязь с другими задачами; указывается периодичность ее решения; раскрываются состав и форма представления входной, промежуточной и результатной информации; характеризуются формы и методы контроля достоверности информации на ключевых этапах решения задачи; специфицируются формы взаимодействия пользователя с ЭВМ в ходе решения задачи и т.п.

Особое внимание в процессе постановки задачи уделяется детальному описанию входной, выходной (результатной) и промежуточной информации. При этом характеризуются:

- форма представления отдельных реквизитов (цифровая, символьная и т.д.);
- количество знаков (разрядов), выделяемых для записи реквизитов исходя из их максимальной значности;
- вид реквизита в процессе решения (первичный, расчетный, нормативный, справочный и т.п.);
- источник (документ) возникновения реквизита.

Кроме того, для цифровой информации указываются: целочисленный или дробный характер реквизита (для дробных величин дополнительно указываются количество десятичных знаков

(разрядов, выделяемых для записи дробной части числа) и допустимый диапазон изменения величин (т.е. максимальное и минимальное значения реквизита).

Для расчетных реквизитов дается соответствующее описание расчетов, и особо выделяются те реквизиты, которые используются при последующих решениях задачи, так как эта информация должна сохраниться в памяти ЭВМ.

Важной особенностью экономических задач является использование в процессе их решения массивов условно-постоянной (постоянной) информации, отражающей многократно используемые справочные, нормативные, расценочные, плановые и другие сведения. Данная информация также детально специфицируется в соответствии с общими требованиями к описанию информации, а кроме того, указывается периодичность внесения изменений в эти массивы.

Завершается постановка задач описанием контрольного примера, демонстрирующего порядок решения задачи традиционным способом. Основное требование к контрольному примеру — отражение всего многообразия возможных форм существования исходных данных. Контрольный пример сопровождается перечислением нештатных ситуаций, которые могут возникнуть при решении задачи, и описанием действий пользователя в каждой конкретной ситуации.

Особенность реализации этого этапа технологического процесса заключается в том, что конечный пользователь разрабатываемой программы, хорошо знающий ее проблемную сторону, обычно слабо представляет специфику и возможности использования ЭВМ для ее решения. В свою очередь, предметная область пользователя бывает часто незнакома разработчику программ, хотя он знает возможности и ограничения на применение ЭВМ. Именно это противоречие является основной причиной возникновения ошибок при реализации данного этапа технологического процесса разработки программ. По данным экспертов, на этот этап приходится более 50% общего числа ошибок, обнаруживаемых в процессе разработки программ решения задач организационно-экономического характера, а затраты на исправление таких ошибок составляют в среднем 80% всех усилий разработчиков на поиск и устранение ошибок в программе.

Отсюда вся важность и ответственность этого этапа, необходимость осуществления корректной и полной постановки задачи, а также однозначность ее понимания как разработчиком программы, так и пользователем этой программы, в качестве которого обычно выступает постановщик задачи.

Математическое описание задачи и выбор метода ее решения представляет *второй этап* в технологии разработки программы. Выделение этого этапа обуславливается рядом причин, одна из которых вытекает из свойства неоднозначности естественного языка, на котором осуществляется описание постановки задачи. В связи с этим на втором этапе технологического процесса разработки программы выполняется формализованное описание задачи, т.е. устанавливаются и формулируются средствами языка математики логико-математические зависимости между исходными и результатными данными.

Математическое описание задачи обеспечивает ее однозначное понимание постановщиком (пользователем) задачи и разработчиком программы, реализующей эту задачу.

В процессе подготовки экономико-математического описания (модели) задачи могут использоваться различные разделы математики, особенно прикладной. При решении экономических задач наиболее часто используются следующие классы моделей для формализованного описания их постановок:

- аналитические (вычислительные);
- матричные (балансовые);
- графические (частным видом которых являются сетевые).

Выбор класса модели, а иногда и конкретной формы ее представления внутри одного и того же класса в ряде случаев позволяет не только облегчить и ускорить процесс решения задачи, но иногда и повысить точность получаемых результатов. Хотя математическая запись постановки задачи, как правило, отличается высокой точностью отображения ее сущности, лаконичностью записи, а главное, однозначностью понимания, далеко не для всех задач она может быть выполнена. Кроме того, математическое описание задачи в большинстве случаев трудно однозначно перевести на язык ЭВМ. Для задач, допускающих возможность экономико-математического описания, необходимо выбрать численный метод решения, а для нечисловых задач - принципиальную схему решения в виде однозначно понимаемой последовательности выполнения элементарных математических и логических функций (операций).

При выборе метода решения задачи предпочтение отдается методу, который наиболее полно удовлетворяет следующим основным требованиям:

- обеспечивает необходимую точность получаемых результатов;
- не обладает свойством вырождения, т.е. бесконечного закликивания на каком-либо участке решения задачи при определенных исходных данных;

- позволяет использовать уже готовые стандартные программы для решения задачи или ее отдельных фрагментов;
- ориентирован на минимальный объем исходной информации;
- обеспечивает наиболее быстрое получение искомых результатов решения.

Сложность и ответственность этапа подготовки экономико-математического описания задачи и выбора (разработки) соответствующего метода ее решения часто требует привлечения квалифицированных специалистов в области прикладной математики, обладающих знанием таких дисциплин, как исследование операций, математическая статистика, численный анализ, вычислительная математика и т.д.

Третий этап технологического процесса подготовки решения задачи на ЭВМ представляет собой *алгоритмизацию ее решения*, т.е. разработку оригинального или адаптацию (уточнение и корректировку уже известного) алгоритма.

Алгоритмизация — это сложный процесс, носящий в значительной степени творческий характер. По оценкам специалистов, постановка задачи и ее алгоритмизация нередко составляют 20—30% общего времени на разработку программных средств решения задачи. Сложность и ответственность реализации данного этапа объясняется тем, что для решения одной и той же задачи, как правило, существует множество различных алгоритмов, отличающихся друг от друга уровнем сложности, объемами вычислительных работ, составом необходимой исходной и промежуточной информации и другими факторами, которые оказывают существенное влияние на эффективность выбранного способа решения задачи.

Алгоритмизация решения задач на ЭВМ. Процесс алгоритмизации решения задачи в общем случае реализуется по следующей схеме:

- выделение автономных этапов процесса решения задачи (как правило, с одним входом и одним выходом);
- формализованное описание содержания работ, выполняемых на каждом выделенном этапе;
- проверка правильности реализации выбранного алгоритма на различных примерах решения задач.

В основу процесса алгоритмизации положено фундаментальное понятие математики и программирования — алгоритм. Название «алгоритм» происходит от латинизированного воспроизведения арабского имени узбекского математика Аль-Хорезми, жившего в конце VIII — начале IX в., который первым сформулировал правила, позволяющие систематически составлять и решать квадратные уравнения.

Развитие ЭВМ сделало понятие алгоритма одним из центральных в прикладной математике, так как возникла острая потребность в определении общих способов формирования и единообразного решения целых классов задач управления на основе разработки комплексов универсальных алгоритмов.

Наряду с трактовкой алгоритма в соответствии с ГОСТ 19.004—80 «*алгоритм* — это точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату», термин «алгоритм» может быть представлен более развернутым определением: как конечный набор правил, однозначно раскрывающих содержание и последовательность выполнения операций для систематического решения определенного класса задач за конечное число шагов.

Любой алгоритм обладает следующими свойствами: детерминированностью, массовостью, результативностью и дискретностью.

Детерминированность (определенность, обусловленность) означает, что набор указаний алгоритма должен быть однозначно и точно понят любым исполнителем. Это свойство определяет однозначность результата работы алгоритма при заданных исходных данных.

Массовость алгоритма предполагает возможность варьирования исходных данных в определенных пределах. Это свойство определяет пригодность использования алгоритма для решения множества задач данного класса. Свойство массовости алгоритма является определяющим фактором, обеспечивающим экономическую эффективность решения задач на ЭВМ. Из сказанного следует, что для задач, решение которых осуществляется один раз, целесообразность использования ЭВМ, как правило, диктуется внеэкономическими категориями.

Результативность алгоритма означает, что для любых допустимых исходных данных он должен через конечное число шагов (или итераций) завершить свою работу.

Дискретность алгоритма означает возможность разбиения определенного алгоритмического процесса на отдельные элементарные этапы, возможность реализации которых человеком или ЭВМ не вызывает сомнения, а результат выполнения каждого элементарного этапа вполне определен и понятен. Таким образом, алгоритм дает возможность чисто механически решать любую конкретную задачу из некоторого класса однотипных задач.

С понятием алгоритма тесно связано понятие данные. В алгоритмическом аспекте *данные* — это информация, несущая полезную смысловую нагрузку, представленная в формализованном виде, позволяющем собирать, передавать, вводить и обрабаты-

вать эту информацию с помощью заданных алгоритмов. Реализация алгоритма на конкретных исходных данных решаемой задачи называется *алгоритмическим процессом*.

Существует несколько способов описания алгоритмов: словесный, формульно-словесный, графический, средствами специального языка операторных схем, с помощью таблиц решений и др. Помимо требования обеспечения наглядности, выбор конкретного способа диктуется рядом факторов, из которых определяющими являются: степень необходимой детализации представления алгоритма, степень формализации алгоритма, уровень логической сложности задачи и т.п.

Словесный способ описания алгоритма отражает содержание выполняемых действий средствами естественного языка. К достоинствам этого способа описания следует отнести его общедоступность, а также возможность описывать алгоритм с любой степенью детализации. К главным недостаткам этого способа следует отнести достаточно громоздкое описание (и, как следствие, относительно низкую наглядность), отсутствие строгой формализации.

Для разработки алгоритмов решения многовариантных расчетов с большим количеством проверок условий, определяющих выбор той или иной ветви процесса обработки информации, целесообразно использовать изобразительные средства в виде *таблиц решений*. Позволяя четко описывать как саму задачу, так и необходимые для ее решения действия, таблицы решений в наглядной форме определяют, от каких условий зависит выбор того или иного действия.

Простота отражения задачи, хорошее восприятие логики ее решения, легкость модификации, компактность записи, а также простота формирования на их основе блок-схем алгоритмов стали основными факторами, обусловившими рост популярности расширения сферы применения таблиц решений. Кроме того, в настоящее время созданы средства автоматизации процесса получения программ на основе таблиц решений. Имеется возможность применения таблиц решений для описания параллельных процессов, а также описания логики построения моделируемых процессов.

Однако специфика процессов обработки экономической информации резко ограничивает эффективность применения таблиц решений. В связи с этим таблицы решений используются главным образом в качестве дополнений к основным способам описания алгоритмов (например, к графическим средствам) для описания участков с многовариантными расчетами.

Составление, тестирование и отладка программ. Составление программ (кодирование) является завершающим этапом технологического процесса разработки программных средств, предшествующим началу непосредственно машинной реализации алгоритма решения задачи. Процесс кодирования заключается в переводе описания алгоритма на один из доступных (понятных) для ЭВМ языков программирования. В процессе составления программы для ЭВМ конкретизируются тип и структура используемых данных, а последовательность действий, реализующих алгоритм, отражается посредством языка программирования.

Тестирование и отладка составляют заключительный этап разработки программы решения задач. Оба эти процесса функционально связаны между собой, хотя их цели несколько отличаются друг от друга. Тестирование представляет собой совокупность действий, предназначенных для демонстрации правильности работы программы в заданных диапазонах изменений внешних условий и режимов эксплуатации программы. Цель тестирования заключается в демонстрации отсутствия (или выявления) ошибок в разработанных программах на наборе заранее подготовленных контрольных примеров.

Процессу тестирования сопутствует понятие «отладка», которое подразумевает совокупность действий, направленных на устранение ошибок в программах, начинающихся с момента обнаружения фактов ошибочной работы программы и завершающихся устранением причин их возникновения.

По своему характеру (природе возникновения) ошибки в программах делятся на синтаксические и логические.

Синтаксические ошибки в программе представляют собой некорректную запись отдельных языковых конструкций с точки зрения правил их представления на выбранном языке программирования. Эти ошибки выявляются автоматически при трансляции исходной программы (т.е. в процессе перевода с исходного языка программирования во внутренние коды машины) для ее выполнения. После устранения синтаксических ошибок проверяется логика работы программы на заданных исходных данных. При этом возможны следующие основные виды проявления логических ошибок:

- в какой-то момент программа не может продолжать работу (возникает программное прерывание, обычно сопровождающееся указанием оператора, на котором оно произошло);
- программа работает, но не выдает всех запланированных результатов и не останавливается (происходит ее «зацикливание»);

— программа выдает результаты и завершает свою работу, но они полностью или частично не совпадают с контрольными. После выявления логических ошибок и установления причин их возникновения в программу вносятся соответствующие исправления и ее отладка продолжается.

Программа считается отлаженной, если она безошибочно выполняется на достаточно представительном наборе тестовых данных, обеспечивающих проверку всех ее ветвей.

Процесс тестирования и отладки программ носит итерационный характер и считается одним из наиболее трудоемких этапов процесса разработки программ. По оценкам специалистов, он может составлять от 30 до 50% (а иногда и больше) в общей структуре затрат времени на разработку проектов и зависит от объема и логической сложности разрабатываемых программных комплексов.

Для сокращения затрат на проведение тестирования и отладки в настоящее время широко используются специальные программные средства тестирования (например, генераторы тестовых данных) и приемы отладки (например, метод трассировки программ).

Эксплуатация программных средств. После завершения процесса тестирования и отладки программные средства вместе с сопроводительной документацией передаются пользователю для эксплуатации. Основное назначение сопроводительной документации — обеспечить пользователя необходимыми инструктивными материалами по работе с программными средствами. Как правило, это документы, регламентирующие работу пользователя при эксплуатации программных средств, а также содержащие информацию о программе, необходимую для изменений и дополнений в ней, которые могут потребоваться с целью дальнейшей ее модернизации. Сопроводительная документация призвана также облегчить процесс выявления причин возникновения тех или иных ошибок в работе программ, которые могут быть обнаружены уже в ходе эксплуатации переданных пользователям программ.

Состав сопроводительной документации обычно оговаривается заказчиком (пользователем) и разработчиком технического задания на программное средство.

Для передачи пользователю разработанных программных средств обычно создается специальная комиссия, включающая в свой состав, как представителей разработчиков, так и заказчиков (пользователей). Комиссия в соответствии с заранее составленным и утвержденным обеими сторонами планом проводит работы по приемке-передаче программных средств и сопроводитель-

ной документации. По завершении работы комиссии оформляется акт приемки-передачи.

В процессе внедрения и эксплуатации программных средств могут выявляться различного рода ошибки, не обнаруженные разработчиком при тестировании и отладке программных средств. Поэтому при реализации достаточно сложных и ответственных программных комплексов по согласованию пользователя (заказчика) с разработчиком этап эксплуатации программных средств может быть разбит на два пол-этапа: *экспериментальная (опытная)* и *промышленная эксплуатация*. Смысл экспериментальной эксплуатации заключается во внедрении разработанных программных средств на объекте заказчика (иногда параллельно с традиционными методами решения задач) с целью проверки их работоспособности при решении реальных задач в течение достаточно большого периода времени (обычно не менее года). Только после завершения периода экспериментальной эксплуатации и устранения выявленных при этом ошибок программное средство передается в промышленную эксплуатацию.

Для повышения качества работ, оперативности исправления ошибок, выявляемых в процессе эксплуатации программных средств, а также выполнения различного рода модификаций разработчик (поставщик) может по договоренности с заказчиком (пользователем) осуществлять сопровождение программных средств. Целесообразность привлечения высококвалифицированных специалистов для сопровождения программных средств у пользователей объясняется тем, что затраты на сопровождение программ в большинстве случаев значительно превосходят первоначальные затраты на их разработку (приобретение).

Следует принимать во внимание, что по своему характеру и последовательности выполняемых действий внесение различного рода изменений в уже функционирующие программы представляет в значительной мере повторение рассмотренных выше этапов, начиная с постановки задачи и кончая внесением изменений в сопроводительную документацию.

Описанная схема технологического процесса разработки эксплуатации программных средств отражает их «жизненный цикл».

Длительность и высокая стоимость разработки программных средств обуславливает целесообразность применения промышленных методов разработки, тиражирования и распространения программных средств. По мнению специалистов, в ближайшее время примерно 20—30% финансовых средств пользователей, выделяемых на программное обеспечение систем обработки информации, будет затрачиваться на приобретение готовых программных средств.

9.4. Техническое и технологическое обеспечение автоматизированного управления

Технологическое обеспечение автоматизированного управления представляет собой некоторую чётко установленную совокупность проектных решений, определяющих последовательность операций, процедур, этапов в соответствующей сфере деятельности пользователя. Различается *системное и прикладное* технологическое обеспечение автоматизированного управления, которое должно позволить решать задачи в регламентном и запросном режимах, в диалоге с пользователем, контролировать ход решения задачи, приостанавливать его, проверять результаты вычислений, вести повторные расчёты и т.п.

При автоматизированном управлении специалистом выполняются на АРМ следующие операции:

- ввод информации с документов при помощи клавиатуры (с визуальным контролем по экрану дисплея);
- ввод данных в ПЭВМ с магнитных носителей и других АРМ;
- приём данных в виде сообщений по каналам связи с других АРМ в условиях функционирования локальных вычислительных сетей;
- редактирование данных и манипулирование ими;
- накопление и хранение данных;
- поиск, обновление и защита данных;
- вывод на экран, печать, магнитный носитель резульатной информации, а также различных справочных и инструктивных сообщений пользователю;
- формирование и передача данных на другие АРМ в виде файлов на магнитных носителях или по каналам связи в вычислительных сетях;
- получение оперативных справок по запросам.

Внедрение АРМ в сферу деятельности управленческих работников приводит к определенному изменению характера их функций и изменению взаимоотношений с другими участниками управленческого процесса. Поэтому технологическое обеспечение должно быть скоординировано с другим видом обеспечения автоматизированного управления — организационным.

9.4.1. Сетевой режим автоматизированной обработки экономической информации

Напомним, что *информационно-вычислительная сеть* — это совокупность программных, технических и коммуникационных средств, обеспечивающих эффективное распределение вычислительных ресурсов.

Являясь одновременно и продуктом, и мощным стимулом развития интеллекта человека, сеть позволяет:

- построить распределенные хранилища информации (базы данных);
- расширить перечень решаемых задач по обработке информации;
- повысить надежность информационной системы за счет дублирования работы ПК;
- создать новые виды сервисного обслуживания, например, электронную почту;
- снизить стоимость обработки информации.

К сетям, как и к отдельным ПК, приложимо понятие «архитектура», под которой понимается конструирование сложных объединений ПК, предоставляющих пользователям широкий набор различных информационных ресурсов. Архитектура сетей имеет набор характеристик.

Открытость. Заключается в обеспечении подключения в контур сети любых типов современных ПК.

Ресурсы. Значимость и ценность сети определяется набором хранимых в ней знаний и данных и способности технических средств оперативно их представлять или обрабатывать.

Надежность. Трактуются как обеспечение высоких показателей надежности, таких как, вероятность безотказной работы, время восстановления работоспособности, коэффициент готовности и др.

Динамичность. Заключается в минимизации времени отклика сети на запрос пользователя.

Интерфейс. Предполагается, что сеть обеспечивает широкий набор сервисных функций по обслуживанию пользователя и предоставлению ему запрашиваемых информационных ресурсов.

Автономность. Понимается как возможность независимой работы сетей различных уровней.

Коммуникации. К ним предъявляются особые требования, связанные с обеспечением четкого взаимодействия ПК по любой принятой пользователем конфигурации сети. Сеть обеспечивает защиту данных от несанкционированного доступа, автоматическое восстановление работоспособности при аварийных сбоях, высокую достоверность передаваемой информации и вычислительных процедур.

Важнейшей характеристикой сети является *топология*, определяемая структурой соединения ПК в сети. Различают физический и логический типы топологии. Под физической топологией

понимается реальная схема соединения узлов сети каналами связи, а под логической — структура маршрутов потоков данных между узлами. Физическая и логическая структуры не всегда совпадают.

Для описания взаимодействия компонентов в сети используются протоколы и интерфейсы.

Протокол в информационной сети — это документ, однозначно определяющий правила взаимодействия одноименных уровней работающих друг с другом абонентов. Например, чтобы сеансовые программы абонентов 1 и 2 (когда сеансовый уровень каждого них представлен комплексом программ) понимали друг друга, они должны работать одинаковым образом, т.е. должны выполнять требования сеансового протокола — стандарта. Это требование определяет список команд, которыми могут обмениваться программы, порядок передачи команд, правила взаимной проверки, размеры передаваемых блоков данных и т.д.

Также протоколами описывается взаимодействие и других одноименных групп программ: транспортных, канальных и т.д. Обычно сеть представляется протоколами семи уровней.

Наиболее важными функциями протоколов на всех уровнях сети являются защита от ошибок, управление потоками данных в сети, защита ее от перегрузок; выполнение операций по маршрутизации сообщений и оптимизации использования ресурсов сети, обеспечивающее большую степень доступности услуг сети путем образования нескольких маршрутов между двумя абонентами.

При подключении компонентов сети друг к другу должны быть однозначно определены правила их стыковки. Их принято называть интерфейсами. *Интерфейс* — свод правил по взаимодействию между функциональными компонентами, расположенными в смежных уровнях и входящими в одну и ту же систему.

При разработке протоколов и интерфейса учитывается свойство открытости с целью их дальнейшего развития и обеспечения взаимодействия с другими средствами и абонентами. Эту работу проводит Международная организация по стандартизации ISO (International Standard Organisation) в содружестве с организациями различных стран.

Многообразие сетевых технологий вызывает необходимость их классификации по каким-либо ключевым признакам. Примерная классификация сетевых технологий дана в табл. 9.5.

Признаки классификации				
Специализация	Способ организации	Способ связи	Состав ПК	Охват территории
Универсальные	Одноранговые (одноуровневые)	Проводные Беспроводные	Однородные	Локальные
Специализированные	Двухуровневые	Спутниковые	Неоднородные	Территориальные Корпоративные Глобальные

По признаку специализации сетевые технологии подразделяются на универсальные, предназначенные для решения всех задач пользователей, и специализированные — для решения небольшого количества специальных задач. Примером специализированной АИС служит технология резервирования мест на авиационные рейсы «Сирена», поезда железной дороги АИС «Экспресс».

Классическим примером универсальной технологии является Академсеть РФ, предназначенная для решения большого количества разнообразных информационных задач.

Выделяемые по способу организации двухуровневые технологии имеют кроме ПК, с которыми непосредственно общаются пользователи, и которые называются рабочими станциями, специальные компьютеры, называемые серверами (serve — обслуживать). Задачей сервера и является обслуживание рабочих станций с предоставлением им своих ресурсов, которые обычно существенно выше, чем ресурсы рабочих станций. Их взаимодействие можно представить следующим образом. По мере необходимости рабочая станция отправляет серверу запрос на выполнение каких-либо действий: прочитать данные, напечатать документ, передать факс и т.д. Сервер выполняет требуемые действия и посылает отчет о «проделанной работе». В зависимости от вида работы, для которой предназначен сервер, он и называется по-разному:

- файловый сервер, если он выполняет простые операции чтения — записи данных из файла;
- принт-сервер, если он выполняет операции печати;
- SQL-сервер, если он выполняет сложные операции поиска и извлечения данных из баз данных (запросы к такому серверу формируются на специальном языке запросов Structured Query Languages — структурированном языке запросов).

В одно-ранговой технологии (одноуровневой, равноправной) функции рабочей станции и сервера совмещены — пользовательский ПК может быть одновременно и рабочей станцией и сервером. Каждый ПК в состоянии предоставлять другому ПК свои ресурсы или, наоборот, запрашивать их у другого.

Разумеется, в системе «клиент—сервер» за счет специализации работ достигается более высокая производительность сети, шире ее спектр и качество услуг. Однако одноранговые сети дешевле и проще в эксплуатации.

По способу связи осуществляется классификация коммуникаций (каналов передачи данных), обеспечивающих движение информации между элементами сети. В проводных технологиях в качестве физической среды в каналах используются:

- плоский двухжильный кабель;
- витая пара проводов;
- коаксиальный кабель;
- световод.

Беспроводные сетевые технологии, использующие частотные каналы передачи данных (средой является эфир), предоставляют в настоящее время разумную альтернативу обычным проводным сетям и становятся все более привлекательными. Самое большое преимущество беспроводных технологий — это возможности предоставляемые пользователям портативных компьютеров. Однако скорость передачи данных, достигаемая в беспроводных технологиях, не может пока сравниться с пропускной способностью кабеля, хотя она в последнее время и значительно выросла (табл. 9.6). Важно, что для перехода к беспроводной технологии не нужно менять уже имеющиеся сети. Аппаратное обеспечение локальных беспроводных сетей теперь может работать с NetWare и другими популярными сетевыми операционными системами, а беспроводные рабочие станции можно добавлять к обычной кабельной сети.

В спутниковых технологиях физической средой передачи данных также является эфир. Использование спутников оправдано в случае значительного удаления абонентов друг от друга при чрезмерном ослаблении посылаемых электромагнитных сигналов с большими посторонними шумами. Чтобы сигналы, направленные отправителем, не смешивались с аналогичными к получателю, при работе со спутником прокладываются два частотных канала — один для отправителя, другой для получателя. Это мероприятие позволяет избежать ошибок при передаче информации.

Характеристика беспроводных и кабельных технологий

Технология	Протокол	Способ передачи данных	Пропускная способность, Мбит/с	Радиус надежной связи, м
Различные	Ethernet	Кабель	10	—
Различные	Token Ring	Кабель	4 или 16	—
Altair Plus	Ethernet	СВЧ-излучение	3,3	До 43
ARLAN	Ethernet	СВЧ-излучение	1	До 40
Free Port	Ethernet	Инфракрасные лучи	5,7	До 25
InfalAn	Token Ring	Инфракрасные лучи	4	До 25
Raglan	Собственный	Широкополосные радиосигналы	0,23	До 250
WareLan	Собственный	Широкополосные радиосигналы	2	До 250

Сравнительно просто классифицируются сетевые технологии по составу ПК. Однородные сетевые технологии предполагают увязку в сети однотипных средств, разрабатываемых одной фирмой. Подключение к такой сети средств других пользователей возможно только при условии соблюдения в них стандартов, принятых в однородной архитектуре.

Другой подход состоит в разработке единой универсальной сетевой технологии независимо от типов применяемых в ней средств. Такие технологии называются неоднородными. Первым стандартом для таких сетей была базовая эталонная модель ВОС.

Наиболее обширно представлена классификация сетевых технологий по признаку «Охват территории».

При выборе информационно-вычислительной сети пользователь в первую очередь решает вопрос о прикладной системе, т. е. комплексе задач предметной области.

При выборе прикладной системы предполагается, что она будет функционировать в некоторой программной и технической среде.

Таким образом, работа по выбору сети предполагает:

- ознакомление с предметной областью;
- выбор сетевой операционной системы;
- предложения по аппаратным решениям:

- по компьютерам;
- по коммуникационному оборудованию.

Эту работу выполняет специализированная фирма-системный интерпретатор. При выборе сети фирма-интерпретатор несет ответственность за все принятые действия и предлагает фирме-заказчику только те решения, которые прошли апробацию на реальном оборудовании в постоянно действующей сетевой лаборатории.

9.4.2. Технология использования автоматизированных рабочих мест

Широкомасштабное оснащение вычислительной техникой всех отраслей человеческой деятельности остро ставит вопрос о технологическом обеспечении информационных систем и технологий.

Технологическое обеспечение реализует информационные процессы в автоматизированных системах организационного управления с помощью ЭВМ и других технических средств.

Разработка технологического обеспечения требует учета особенностей структуры экономических систем. Прежде всего — это сложность организационного взаимодействия, которая вызывает необходимость создания многоуровневых иерархических систем (головная фирма, филиалы), со сложными информационными связями прямого и обратного направления. В основу новой информационной технологии закладывается широкое применение компьютеров и формирование на их базе вычислительных сетей с взаимосвязанными специализированными АРМами.

Деятельность работников управления (бухгалтеров, специалистов кредитно-банковской системы, плановиков и т. д.) в настоящее время ориентирована на использование развитых технологий. Организация и реализация управленческих функций требует радикального изменения как самой технологии управления, так технических средств обработки информации, среди которых главное место занимают персональные компьютеры. Они все более превращаются из систем автоматической обработки входной информации в средства накопления опыта управленческих работников, анализа, оценки и выработки наиболее эффективных экономических решений.

Тенденция к усилению децентрализации управления влечет за собой распределенную обработку информации с децентрализацией применения средств вычислительной техники и совершенствованием организации непосредственно рабочих мест пользователей.

Напомним, что *автоматизированное рабочее место* можно определить как *совокупность* информационно-программно-технических средств, обеспечивающих конечному пользователю обработку данных и автоматизацию управленческих функций в конкретной предметной области.

Создание автоматизированных рабочих мест предполагает, что основные операции по накоплению, хранению и переработке информации возлагаются на вычислительную технику, а экономист выполняет часть ручных операций, требующих творческого подхода при подготовке управленческих решений. Персональная техника применяется пользователем для контроля производственно-хозяйственной деятельности, изменения значений отдельных параметров в ходе решения задачи, а также ввода исходных данных в АИС для решения текущих задач анализа функций управления.

АРМ как инструмент для рационализации и интенсификации управленческой деятельности создается для обеспечения выполнения некоторой группы функций. Наиболее простой функцией АРМ является информационно-справочное обслуживание. Хотя эта функция в той или иной степени присуща любому АРМ, особенности ее реализации существенно зависят от категории пользователя.

АРМ имеют проблемно-профессиональную ориентацию на конкретную предметную область. Профессиональные АРМ являются главным инструментом общения человека с вычислительными системами, играя роль автономных рабочих мест, интеллектуальных терминалов больших ЭВМ, рабочих станций в локальных сетях. АРМ имеют открытую архитектуру и легко адаптируются к проблемным областям.

Локализация АРМ позволяет осуществить оперативную обработку информации сразу же по ее поступлении, а результаты обработки хранить сколь угодно долго по требованию пользователя.

В условиях реализации управленческого процесса целью внедрения АРМ является усиление интеграции управленческих функций, и каждое более или менее «интеллектуальное» рабочее место должно обеспечивать работу в многофункциональном режиме.

АРМ выполняют децентрализованную одновременную обработку экономической информации на рабочих местах исполнителей в составе распределенной базы данных (БД). При этом они имеют выход через системное устройство и каналы связи в ПЭВМ и БД других пользователей, обеспечивая, таким образом, совместное функционирование ПЭВМ в процессе коллективной обработки.

АРМ, созданные на базе персональных компьютеров, — наиболее простой и распространенный вариант автоматизированного рабочего места для работников сферы организационного управления. Такой АРМ рассматривается как система, которая в интерактивном режиме работы предоставляет конкретному пользователю все виды обеспечения монополюльно на весь сеанс работы. Этому отвечает подход к проектированию такого компонента АРМ, как внутреннее информационное обеспечение, согласно которому информационный фонд на магнитных носителях конкретного АРМ должен находиться в монополюльном распоряжении пользователя АРМ. Пользователь сам выполняет все функциональные обязанности по преобразованию информации.

Создание АРМ на базе персональных компьютеров обеспечивает:

- простоту, удобство и дружелюбность по отношению к пользователю;
- простоту адаптации к конкретным функциям пользователя;
- компактность размещения и невысокие требования к эксплуатации;
- высокую надежность и живучесть;
- сравнительно простую организацию технического обслуживания.

Эффективным режимом работы АРМ является его функционирование в рамках локальной вычислительной сети в качестве *рабочей станции*. Особенно целесообразен такой вариант, когда требуется распределять информационно-вычислительные ресурсы между несколькими пользователями.

Более сложной формой является АРМ с использованием ПЭВМ в качестве интеллектуального терминала, а также с удаленным доступом к ресурсам центральной (главной) ЭВМ или внешней сети. В данном случае несколько ПЭВМ подключаются по каналам связи к главной ЭВМ, при этом каждая ПЭВМ может работать и как самостоятельное терминальное устройство.

В наиболее сложных системах АРМ могут через специальное оборудование подключаться не только к ресурсам главной ЭВМ сети, но и к различным информационным службам и системам общего назначения (службам новостей, национальным информационно-поисковым системам, базам данных и знаний, библиотечным системам и т. п.).

Возможности создаваемых АРМ в значительной степени зависят от технико-эксплуатационных характеристик ЭВМ, на которых они базируются. В связи с этим на стадии проектирования АРМ четко формулируются требования к базовым параметрам

технических средств обработки и выдачи информации, набору комплектующих модулей, сетевым интерфейсам, эргономическим параметрам устройств и т.д.

Синтез АРМ, выбор его конфигурации и оборудования для реальных видов экономической и управленческой работы носят конкретный характер, диктуемый специализацией, поставленными целями, объемами работы. Однако любая конфигурация АРМ должна отвечать общим требованиям в отношении организации информационного, технического, программного обеспечения.

Информационное обеспечение АРМ ориентируется на конкретную, привычную для пользователя, предметную область. Обработка документов должна предполагать такую структуризацию информации, которая позволяет осуществлять необходимое манипулирование различными структурами, удобную и быструю корректировку данных и массивов.

Техническое обеспечение АРМ должно гарантировать высокую надежность технических средств, организацию удобных для пользователей режимов работы (автономный, с распределенной БД, информационный, с техникой верхних уровней и т. д.), способность обрабатывать в заданное время необходимый объем данных. Поскольку АРМ является индивидуальным пользовательским средством, оно должно обеспечивать высокие эргономические свойства и комфортность обслуживания.

Программное обеспечение, прежде всего, ориентируется на профессиональный уровень пользователя, сочетается с его функциональными потребностями, квалификацией и специализацией. Пользователь со стороны программной среды должен ощущать постоянную поддержку своего желания работать в любом режиме активно либо пассивно. Приоритет пользователя при работе с техникой несомненен. Поэтому при их взаимодействии предусматривается максимальное обеспечение удобств работы человека за счет совершенствования программных средств.

В последнее время наметилась тенденция к созданию унифицированных АРМ, обслуживающих несколько предметных областей. Например, комплекс АРМ-аналитик, созданный на базе АРМ-статистика, значительно расширяет возможности последнего и в максимальной степени отвечает требованиям зарождающихся в условиях рынка производственных, научных и коммерческих структур. АРМ-аналитик позволяет осуществлять решение обширного комплекса функциональных задач.

9.4.3. Технология использования пакетов прикладных программ

Одним из условий эффективного внедрения вычислительной техники в практику является создание специализированных пакетов прикладных программ. Доступность и простота использования их создает предпосылки более широкого внедрения ЭВМ в инженерный труд, решение конкретных задач научной области, экономики, культуры, образования.

ППП обычно строятся на базе специальных систем и являются дальнейшим их развитием в конкретном направлении. Они поставляются отдельно от программного обеспечения вычислительных средств, имеют свою документацию и не входят в состав операционных систем. Многие пакеты имеют собственные средства генерации. Разработка пакета не должна требовать модификации операционных систем. Это относится к пакетам, влияющим на работу управляющих программ. Если пакет требует внесения изменений в управляющую программу, то это выполняется в процессе загрузки и инициализации пакета.

Пакеты прикладных программ являются наиболее динамично развивающейся частью программного обеспечения: круг решаемых с помощью ППП задач постоянно расширяется. Во многом внедрение компьютеров практически во все сферы деятельности стало возможным благодаря появлению новых и совершенствованию существующих ППП.

Достижения в области микроэлектроники, приводящие к появлению более мощных по своим функциональным возможностям компьютеров, также являются причиной создания новых ППП. В свою очередь, необходимость улучшения характеристик использования пакета при решении конкретных задач пользователя стимулирует совершенствование архитектуры и элементной базы компьютеров и периферийных устройств.

Структура и принципы построения ППП зависят от класса ЭВМ и операционной системы, в рамках которой этот пакет будет функционировать. Наибольшее количество разнообразных ППП создано для IBM PC-совместимых компьютеров с операционной системой MS DOS и WINDOWS.

Рассмотрим технологию использования современных ППП на примере СУБД Microsoft Access.

Технология создания баз данных с помощью типовых инструментальных средств, рассчитанных на массового пользователя — непрограммиста, предоставляется СУБД Microsoft Access. Несмотря на ориентированность на конечного пользователя, в Access присутствует язык программирования, имеется возможность интег-

рации с другими программными средствами Microsoft Office. Access — это популярная настольная система управления базой данных, рассчитанная на конечного пользователя. В то же время на небольшом предприятии (при объеме данных до 1 Гбайта) с количеством компьютеров не более 10 ресурсов Access вполне может хватить для обслуживания всего делопроизводства вместе со средствами Microsoft Office. Все пользователи могут обращаться к одной базе данных, установленной на одном компьютере, который может не быть сервером.

Проблемы сохранности и доступа к данным решаются с помощью использования средств защиты, которые предоставляет Access. Среди других технологий создания баз данных является направленность на конечного пользователя (непрограммиста), сохранение общего подхода, принятого в построении продуктов Microsoft для WINDOWS, массовость использования.

В Access для работы с данными используется процессор баз данных, средства быстрого построения интерфейса (*Конструктор форм и отчетов*), объекты доступа и манипулирования данными (таблицы, формы, запросы, макрокоманды, макросы, модули). Автоматизация рутинных типовых операций выполняется с помощью готовых визуальных средств или макрокоманд, объединяемых в макросы. Таким образом, пользователи Access могут обратиться к созданию процедур и функций для работы с данными. При этом, если недостает возможностей визуальных готовых средств, обращаются к макрокомандам, а если и их возможностей недостаточно, можно использовать язык программирования. Он позволяет создавать свои массивы, типы данных, функции, приложения. Имеется возможность целиком создать базу данных с помощью программирования, когда в этом появится необходимость.

Создание новой базы данных начинается с запуска Access и появления диалогового окна. Выбор опции *Запуск мастера* приводит в окно *Создание*. Далее для создания базы можно использовать шаблоны. Чтобы обратиться к списку шаблона, необходимо перейти на вкладку *Базы данных*. Создаются базы данных выбором из определенного списка. При этом возможен выбор таблиц, а в таблицах — нужных полей. После этого пользователь получает базу данных с таблицами, формами ввода и вывода. В табл. 9.7 приведен список мастеров (программных модулей), имеющихся в Access. Дополнительно к перечисленным возможностям все созданные формы можно редактировать с помощью вспомогательных диалоговых окон. При первом знакомстве с Access такой способ создания баз данных весьма эффективен.

Технология *ввода данных* в базу допускает использование таблицы и формы, через которые обеспечивается работа только с од-

ной строкой таблицы. Ввод с помощью формы позволяет располагать поля в нужном порядке, удобном для пользователя. Создание форм может выполнять сам пользователь или с помощью *Мастера*. Этапы создания формы включают выбор полей, внешнего вида, стиля и названия формы.

Работа с базой данных начинается с создания таблиц. Обращение к режиму *Создать* предоставляет возможность выбора одного из пяти вариантов технологии создания таблицы (табл. 9.8).

Технология запросов к данным базы в большинстве строится программно, а в Access она выполняется визуально за исключением сквозных запросов. Пользователь благодаря Access реализует разнообразные запросы выборки, при этом они могут модифицировать исходные данные. В этом заложены резервы ускорения работы с данными. Недостатком технологии Access является замедление скорости работы с данными при увеличении таблиц.

Пользователь непосредственно участвует в формировании запросов, не прибегая к услугам программистов.

Таблица 9.7

Мастера СУБД Access

Наименование	Назначение
Мастер баз данных	Создает базы данных из определенного списка, возможен выбор необходимых таблиц и полей, создает формы и отчеты
Мастер таблиц	Создает таблицы из списка уже готовых, которые можно изменять. Интересен только на начальном этапе использования таблиц, хотя определенные задачи можно решить, применяя только таблицы, предоставляемые мастером
Мастер простых форм	Создает простую форму, в которую выводятся выбранные пользователем поля из таблицы или запроса
Мастер форм с диаграммой	Создает форму с диаграммой, отражающей данные для полей из таблиц и запросов, которые служат источником данных для форм
Мастер форм со сводной таблицей Microsoft Excel	Создает форму, в которую включен объект «страница Excel» со сводной таблицей
Мастер построения кнопок	Создает кнопки в форме или отчете с выбранными Вами свойствами или функциональностью
Мастер построения групп	Создает группу переключателей, которая может содержать множество кнопок, флажков, выключателей

Наименование	Назначение
Мастер построения списков	Создает списки на основе полей из таблиц и запросов, SQL-выражений или предопределенного набора значений
Мастер построения комбинированных списков	Создает комбинированные списки на основе полей из таблиц и запросов, SQL-выражений или предопределенного набора значений
Мастер построения подчиненных форм	Создает подчиненную форму, которая может служить аналогом объектов Grid или Browse в других системах управления базами данных
Мастер создания отчета	Создает отчет, в который выводятся выбранные пользователем поля из таблицы или запрос, с возможностями установки сортировки и группировки
Мастер создания наклеек	Позволяет создавать наклейки, как стандартные, так и иных размеров
Мастер создания отчетов с диаграммой	Позволяет выводить на печать диаграммы, внешний вид которых зависит от данных в таблице или запросе, являющихся источником данных для отчета

Пользователь может направлять запросы в базу для добавления, удаления, обновления, создания таблиц. Запросы можно составить и программным путем. Одна из сильных сторон технологии Access — фильтры, которые позволяют выбирать информацию с помощью запросов или установкой критериев. Создание параметрических запросов дает возможность пользователю вводить значения для отбора данных.

Наряду с формами для каждой таблицы могут быть созданы *отчеты* с помощью средств СУБД или программным путем, что более трудоемко.

Для каждой таблицы можно создать *Автоотчет* с выводом данных в столбец.

При создании отчета с выбором полей, но без вывода всех имеющихся в таблице или запросе данных, Access позволяет обратиться к *Мастеру отчетов*. Мастер отчетов помимо выбора полей, группирует данные по какому — либо полю, устанавливает интервал группировки, порядок сортировки, диаграммы, макет отчета и его стиль. Для построения еще более сложных отчетов используется *Конструктор отчетов*.

Программное создание отчетов используется для построения собственных мастеров.

Способы создания таблиц в СУБД Access

Способ создания	Описание
Режим создания	Пользователю предоставляется таблица с тридцатью полями, куда необходимо ввести данные. После их сохранения Access решает, какой тип данных присвоить каждому полю. Как недостаток этого способа следует отметить невозможность создать таблицу с полями примечаний
Конструктор таблиц	После выбора этой операции открывается <i>Конструктор таблиц</i> , в котором пользователю необходимо самостоятельно создать поля, выбрать типы данных для полей, размеры полей и, если это необходимо, установить свойства полей
Мастер таблиц	Из определенного набора таблиц пользователь может создать таблицу по своему вкусу. Возможно, что некоторые таблицы целиком подойдут для данного приложения, следует их использовать, так как все средства хороши для того, чтобы побыстрее завершить процесс
Импорт таблиц	Позволяет импортировать данные из таблиц других приложений в базу данных. Новые таблицы теряют непосредственную связь с другими приложениями. В появившемся диалоговом окне необходимо выбрать тип файла и имя импортируемого файла. Тип файла ODBC позволяет импортировать данные практически любого формата
Связь с таблицами	Очень похоже на предыдущий пункт, но при этом таблица остается в своем формате, т.е. может использоваться несколькими приложениями

Технология выполнения разнообразных действий и функций с данными базы в среде Access осуществляется макрокомандами, которые объединяются в макросы. Задаваемые параметры придают этим действиям гибкость, которой иначе можно добиться только путем кропотливого программирования. Хотя сами макросы упрощают работу, их создание требует от пользователя затрат труда и времени. В Access имеется около пятидесяти макрокоманд.

Технологии создания баз данных для персональных компьютеров ориентированы на решение несложных задач с ограниченным объемом информации.

9.4.4. Технология обработки текстовой информации

Пользователь ПЭВМ часто встречается с необходимостью подготовки тех или иных документов — писем, статей, служебных записок, отчетов, рекламных материалов и т.д. Для подготовки документов текст редактируемого документа выводится на экран, и пользователь может в диалоговом режиме вносить в него свои изменения. Все внесенные изменения фиксируются. При распечатке выводится отформатированный текст, в котором учтены все исправления. Пользователь может переносить части текста из одного места документа в другое, использовать несколько видов шрифтов для выделения отдельных участков текста, печатать подготовленный документ на принтере в нужном количестве экземпляров.

Удобство и эффективность применения компьютера для подготовки текста привели к созданию множества программ для обработки документов. Такие программы называются текстовыми процессорами или редакторами.

Возможности этих программ различны — от программ, предназначенных для подготовки небольших документов простой структуры, до программ для набора, оформления и полной подготовки к типографскому изданию книг и журналов (*издательские системы*).

Редакторы текстов программ редактируют программы на том или ином языке программирования. Часто они встроены в систему программирования. Это редакторы Turbo (Borland), C++, Turbo Pascal, Multi-Edit, Brief и т.д.

Они выполняют следующие функции: диалоговый просмотр текста, редактирование строк программы, копирование и перенос блоков текста, копирование одной программы или ее части в указанное место другой программы, контекстный поиск и замена подстрок текста, автоматический поиск строки, содержащей ошибку, распечатка программы или ее части.

Часто редакторы текстов программ позволяют автоматически проверять синтаксическую правильность программ. Иногда эти редакторы объединены с отладчиками программ на уровне исходного текста. Редакторы текстов программ можно использовать для создания и корректирования небольших документов. Однако для серьезной работы с документами предпочтительнее использовать специальные редакторы, ориентированные на работу с текстами, имеющими структуру документа, т.е. состоящими из разделов, страниц, абзацев, предложений, слов и т.д. Такие редакторы обеспечивают следующие функции:

- возможность использования различных шрифтов символов;
- работу с пропорциональными шрифтами;
- задание произвольных межстрочных промежутков;
- автоматический перенос слов на новую строку;
- автоматическую нумерацию страниц;
- обработку и нумерацию сносок;
- печать верхних и нижних заголовков страниц;
- выравнивание краев абзацев;
- набор текста в несколько столбцов;
- проверку правописания и подбор синонимов;
- построение оглавлений индексов;
- сортировку текстов и данных и т.д.

Существует несколько сотен редакторов текстов — от самых простых до весьма мощных и сложных. Наиболее распространенные Microsoft Word, WordPerfect, WordStar. В США наиболее распространены Microsoft Word для WINDOWS и WordPerfect, в Европе и России — Microsoft Word.

Среди простых редакторов текста в России распространение получил Лексикон. Он имеет интерфейс на русском языке и позволяет готовить несложные документы с текстом на русском и английском языках. Лексикон вполне подходит тем, кому нужен простой инструмент для подготовки небольших и несложных документов, не требующих высокого полиграфического качества.

Пользователям, которым требуется обеспечить высокое качество напечатанных документов или подготовить сложные документы большого объема, рекламные буклеты или книги, возможностей Лексикона недостаточно. Им лучше воспользоваться Microsoft Word.

В Microsoft Word для WINDOWS реализована фоновая проверка орфографии. По мере введения текста редактор проверяет его и подчеркивает слова, содержащие ошибки, красной волнистой чертой. Ошибки проверяются не только в словах, но и в выражениях. Сомнительные и ошибочные слова, словосочетания и предложения подчеркиваются волнистой зеленой линией.

Microsoft Word — мощный интеллектуальный текстовый редактор, удобный в использовании инструмент создания профессионально оформленных документов. Он содержит инструмент рисования таблиц, путем обычного рисования линий в тех местах, где они должны быть в таблице. Эти линии автоматически превращаются в элементы таблицы. Выравнивание введенных линий по краям таблицы также происходит автоматически.

Кроме того, Microsoft Word работает с Мастером писем. Последний позволяет установить параметры письма, его оформле-

ние, вставить общий текст (например, обратный адрес и адрес получателя), а также отправить письмо.

Для подготовки рекламных буклетов, оформления журналов и книг используются специальные издательские системы. Они позволяют готовить и печатать на лазерных принтерах или выводить на фотонаборные автоматы сложные документы высокого качества.

Имеются два основных вида издательских систем. Издательские системы первого вида очень удобны для подготовки небольших материалов с иллюстрациями, графиками, диаграммами, различными шрифтами в тексте, например, газет, небольших журналов. Типичный пример такой системы — Aldus Page Maker.

Издательские системы второго вида более подходят для подготовки больших документов, например, книг. Одной из самых распространенных таких систем является система Ventura Publisher (Corel Ventura). Ventura управляется меню и может считывать тексты, подготовленные с помощью других текстовых редакторов (например, Microsoft Word), сохраняя при этом параметры форматирования, заданные этими редакторами.

Основная операция, для которой используется издательские системы — это верстка (размещение текста по страницам документа, вставка рисунков, оформление текста различными шрифтами и т.д.). В режиме ввода редактирования текста Ventura и Aldus Page Maker значительно уступают в скорости и удобстве редакторам текстов. Поэтому чаще всего документы подготавливаются в два этапа: набирают текст в редакторе типа Microsoft Word, а затем считывают его системой Aldus Page Maker или Ventura и осуществляют окончательную подготовку документа.

Основные функции издательских систем следующие: использование сотен видов шрифтов (начертаний и размеров символов текста), которые отображаются на экране также, как при печати, размещение фрагментов в документе, изменения и корректировка рисунков и диаграмм, растягивание букв в тексте (разрядка), сближение их друг с другом, подготовка таблиц, выравнивание нижнего края текста на странице на заданную границу (чтобы страницы документа имели единообразный вид), набор формул и т.д.

Многим пользователям для издательских работ оказывается вполне достаточно возможностей Microsoft Word для WINDOWS. В последнее время производители издательских систем стали встраивать в них элементы профессионального цветоделения, обеспечивающие подготовку высококачественных цветных изданий, а также средства графических редакторов.

9.4.5. Технология обработки табличной информации

Множество задач, которые предстоит решать фирмам и предприятиям, носят учетно-аналитический характер и требуют табличной компоновки данных с подведением итогов по различным группам и разделам данных, например при составлении баланса, справок для налоговых органов, возможных финансовых отчетов и т.п. Для хранения и обработки информации, представленной в табличной форме используют электронные таблицы (ЭТ).

Программные средства для проектирования таблиц называют также табличными процессорами. Они позволяют не только создавать таблицы, но и автоматизировать обработку табличных данных. Кроме того, с помощью ЭТ можно выполнять различные экономические, бухгалтерские и инженерные расчеты, а также строить разного рода диаграммы, проводить сложный экономический анализ, моделировать и оптимизировать решение хозяйственных различных операций и многое другое.

Функции табличных процессоров весьма разнообразны и включают:

- создание и редактирование ЭТ;
- оформление и печать ЭТ;
- создание многотабличных документов, объединенных формулами;
- построение диаграмм, их модификацию и решение экономических задач графическими методами;
- работу с электронными таблицами как с базами данных (сортировка таблиц, выборка данных по запросам);
- создание итоговых и сводных таблиц;
- использование при построении таблиц информации из внешних баз данных;
- решение экономических задач типа «что — если» путем подбора параметров;
- решение оптимизационных задач;
- статистическую обработку данных;
- создание слайд-шоу;
- разработку макрокоманд, настройку среды под потребности пользователя и т.д.

Табличные процессоры различаются в основном набором выполняемых функций и удобством интерфейса, поэтому целесообразно проанализировать лишь широко используемые программные продукты.

Перспективные направления в разработке ЭТ основными фирмами-разработчиками определяются по-разному.

Фирма Microsoft уделяет первостепенное внимание совершенствованию набора функциональных средств Excel. В Excel многие функции разработаны более тщательно, чем в других электронных таблицах. Кроме того, возможность использования массивов в Excel обеспечивает большую гибкость при работе с таблицами.

Фирма Lotus основные усилия сконцентрировала на разработке инструментов групповой работы.

Версия 4.0 пакета Lotus 1-2-3 дополнена Version Manager для моделирования по принципу «что — если», а версия 5.0 дополнена средствами маршрутизации и связи с Notes, что позволяет создать приложения в других пакетах. Lotus 1-2-3 имеет ряд сильных сторон, к которым можно отнести простоту создания и редактирования графиков, а также наиболее логичную структуру трехмерных таблиц. Предусматривается также совершенствование групповой работы с таблицами: использование Team Consolidate предоставляет возможность группе пользователей редактировать копии ЭТ, а затем их объединять. В версию пакета для WINDOWS включен язык программирования Lotus Script.

Пакет Quattro Pro в результате тестирования получил достаточно высокие оценки, но ни одна из особенностей пакета не вызвала к себе повышенного внимания. Наиболее привлекательными оказались возможности сортировки данных (которые хорошо реализованы в Excel), а также удобство эксплуатации. Вместе с тем отмечались сложности при освоении графических возможностей Quattro Pro и недостаточный объем справочной информации.

Ситуация, сложившаяся к настоящему времени на рынке ЭТ, характеризуется явным лидирующим положением фирмы Microsoft — 80% всех пользователей ЭТ предпочитают Excel. На втором месте по объему продаж — Lotus 1-2-3, а затем Quattro Pro.

Электронные таблицы состоят из столбцов и строк. Для обозначения строк используется цифровая нумерация, столбцов — буквенные индексы. Количество строк и столбцов в разных ЭТ различно. Например, в табличном процессоре Excel 256 столбцов и более 16 тысяч строк. Место пересечения строки и столбца называется ячейкой. Каждая ячейка имеет свой уникальный идентификатор (адрес), состоящий из имени столбца и номера строки, например, A28, B97 и т.п. ЭТ могут содержать несколько рабочих бланков, которые объединяются в один файл и носят название рабочей книги. В книгу можно поместить несколько различных типов документов, например, рабочий лист с ЭТ, лист диаграмм, лист макросов и т.п.

В ЭТ можно работать как с отдельными ячейками, так и с группой ячеек, которые образуют блок. Имена ячеек в блоках разделяются двоеточием (:), например, блок A1:B4 включает в себя ячейки A1, A2, A3, A4, B1, B2, B3, B4. С блоками ячеек, в основном, выполняются операции копирования, удаления, перемещения, вставки и т.д. Адреса используются в формулах как ссылки на определенные ячейки. Таким образом, введенные один раз значения можно многократно и в любом месте таблицы использовать без повторного набора. Соответственно, при изменении значения ячейки автоматически произойдут изменения в тех формулах, в которых содержатся ссылки на данную ячейку.

Технология работы с табличным документом аналогична процедурам подготовки текстовых документов: редактируемый отчет в виде таблицы выводится на экран, и пользователь может в диалоговом режиме вносить в него свои изменения (т.е. редактировать содержимое клеток ЭТ). Все внесенные изменения сразу же отображаются на экране компьютера.

В клетки ЭТ могут быть введены текст, цифры и формулы. Во всех табличных процессорах существует синтаксические соглашения, позволяющие отличить формально — цифровую информацию от текстовой, которых должен придерживаться пользователь, если хочет добиться правильных результатов. Обычно синтаксические правила интуитивно понятны и легко запоминаются (например, задание формулы начинается со знака «=» и т.п.). Формула — это выражение, состоящее из числовых величин и арифметических операций. Кроме числовых величин, в формулу могут входить в качестве документов адреса ячеек, функции и другие формулы. Пример формулы: = A5/H8*12. В ячейке, в которой находится формула, виден только результат вычислений. Саму формулу можно увидеть в строке ввода, когда данная ячейка станет активной.

Функции представляют собой запрограммированные формулы, позволяющие проводить часто встречающиеся последовательности вычислений. Например, функция автосуммирования может быть представлена следующим образом: = СУММ: (A1:A4).

В Microsoft Excel можно работать с четырьмя основными типами документов: электронной таблицей (в Excel ЭТ называется рабочим бланком), рабочей книгой, диаграммой, макротаблицей.

Рабочий бланк служит для организации и анализа данных. Одновременно на нескольких бланках данные можно вводить, править, производить с ними вычисления. В книгу можно вставить листы диаграмм для графического представления данных и модули для создания и хранения макросов, используемых при хранении специальных задач.

Рабочая книга представляет собой электронный эквивалент папки скоросшивателя. Книга состоит из листов имени, которых выводятся на ярлычках в нижней части экрана. По умолчанию книга открывается с 16 рабочими листами — Лист 1, Лист 2, ..., Лист 16, однако их число можно увеличить или уменьшить. В книгу можно поместить несколько различных типов документов, например рабочий лист с электронной таблицей, лист диаграмм, лист макросов и т.п.

Диаграмма представляет собой графическое изображение связей между числами ЭТ. Она позволяет показать количественное соотношение между сопоставляемыми величинами.

Макротаблица (макрос) — это последовательность команд, которую приходится постоянно выполнять пользователю в повседневной работе. Макросы позволяют автоматизировать часто встречающиеся операции.

Любая ЭТ состоит из следующих элементов: заголовка таблицы, заголовка столбцов, (шапки таблицы); информационной части (исходных и выходных данных, расположенных в соответствующих ячейках).

Процесс проектирования ЭТ состоит из следующих этапов:

- формирования заголовка ЭТ;
- ввода названий граф документа;
- ввода исходных данных;
- ввода расчетных формул;
- форматирования ЭТ с целью придания ей профессионального вида;
- подготовка к печати и ее печать.

При необходимости ЭТ могут сопровождаться различными пояснительными комментариями и диаграммами.

Excel предоставляет большой набор возможностей по графическому представлению данных. Имеется возможность выбора из 14 различных типов диаграмм, причем каждый тип диаграмм имеет несколько разновидностей (подтипов).

Диаграммы можно строить либо на рабочем бланке таблицы, либо на новом рабочем бланке. Создать диаграмму в Excel можно по шагам с помощью Мастера диаграмм, вызов которого осуществляется с панели инструментов диаграмм.

При использовании Мастера диаграмм можно просмотреть любой тип диаграммы и выбрать наиболее удачный для данной таблицы. Включенная в рабочий бланк диаграмма может находиться в одном из трех режимов:

- просмотра, когда диаграмма выделена по периметру прямоугольником;

- перемещения, изменения размера и удаления, когда диаграмма по периметру выделена прямоугольником с маленьким квадратиком;
- редактирования, когда диаграмма выделена по периметру синим цветом или выделен синим цветом заголовок.
- Представление данных в виде диаграмм позволяет наглядно представить числовые данные и осуществить их анализ по нескольким направлениям.

9.4.6. Технология использования автоматизированных банков данных

Автоматизированный банк данных (АБнД) — это автоматизированная система, представляющая совокупность информационных, программных, технических средств и персонала, обеспечивающих хранение, накопление, обновление, поиск и выдачу данных.

Главными составляющими банка данных являются база данных (БД), программные средства и администратор АБнД .

Использование принципов базы и банка данных предполагает организацию хранения информации в виде базы данных, где все данные собраны в едином интегрированном хранилище и к информации как важнейшему ресурсу обеспечен широкий доступ разнообразных пользователей. Такая организация данных решает целый ряд проблем:

- отпадает необходимость в каждой прикладной программе детально решать вопросы организации файлов.
- устраняется многократный ввод и дублирование тех же данных.
- не возникает проблемы изменения прикладных программ и связи с заменой физических устройств или изменение структуры данных.
- повышается уровень надежности и защищенности информации.
- уменьшается избыточность данных.

Перечисленные достоинства обеспечиваются способами логической и физической организации данных, закладываемыми на стадии проектирования внутримашинного информационного обеспечения.

Технология баз и банков данных является ведущим направлением организации внутримашинного информационного обеспечения. Развитие технологий баз и банков данных определяется рядом факторов: ростом информационных потребностей пользователей, требованиями эффективного доступа к информации,

появлением новых видов массовой памяти, увеличением ее объемов, новыми средствами и возможностями в области коммуникаций и многим другим.

В отличие от локально организованных информационных массивов, ориентированных на решение отдельных задач, база данных является интегрированной системой информации, удовлетворяющей ряду требований:

- сокращению избыточности в хранении данных;
- устранению противоречивости в них;
- совместному использованию для решения большого круга задач, в том числе и новых;
- удобства доступа к данным;
- безопасности хранения данных в базе, защиты данных;
- независимости данных от внешних условий в результате развития информационного обеспечения;
- снижение затрат не только на создание и хранение данных, но и на поддержание их в актуальном состоянии;
- наличие гибких организационных форм эксплуатации.

Реализация указанных требований дает высокую производительность и эффективность работы с данными для пользователей в больших объемах.

Напомним, что *база данных* (БД) — это специальным образом организованное хранение информационных ресурсов в виде интегрированной совокупности файлов, обеспечивающей удобное взаимодействие между ними и быстрый доступ к данным.

Наиболее общее представление о базе данных заключается в следующем.

База данных — это совокупность хранимых во внешней памяти ЭВМ большого объема данных; база данных является собой комплекс взаимосвязанных данных, предназначенных для обеспечения информационных нужд различных пользователей, каждый из которых имеет отношение к отдельным, возможно, совместно используемым частям данных; работа с базой данных может осуществляться либо в пакетном режиме, либо с удаленных терминалов в режиме реального времени.

База данных — это динамичный объект, меняющий значения при изменении состояния отражаемой предметной области (внешних условий по отношению к базе). Под предметной областью понимаются часть реального мира (объектов, процессов), которая должна быть адекватно, в полном информационном объеме представлена в базе данных. Данные в базе организуются в единую целостную систему, что обеспечивает более производительную работу пользователей с большими объемами данных.

Таким образом, база данных — это совокупность хранимых в памяти ЭВМ и специальным образом организованных взаимосвязанных данных, отображающих состояние предметной области. База данных также предназначена для обеспечения информационных нужд определенных пользователей.

Программные средства — это прикладное программное обеспечение, удобные пользовательские программы, написанные на языках СУБД.

Система управления базами данных является составной частью автоматизированного банка данных и обеспечивает работу прикладных программ с базой данных.

Одним из важнейших назначений СУБД является обеспечение независимости данных. Под этим термином понимается независимость данных и использующих их прикладных программ друг от друга, в том смысле, что изменение одних не ведет к изменению других. Необходимо также отметить такие возможности СУБД, как обеспечение защиты и секретности данных, восстановление баз данных после сбоев, ведение учета работы с базами данных. Однако, это является неполным перечнем того, что должна осуществлять СУБД для обеспечения интерфейса пользователей с базами данных и жизнеспособности всего автоматизированного банка данных.

Система управления данными имеет набор средств, которые обеспечивают вполне определенные способы доступа к данным. Наиболее общими операциями, которые выполняются средствами СУБД, являются операции поиска, исправления, добавления и удаления данных. Необходимо отметить, что операции поиска является главной среди указанных.

Степень реализации принципа независимости данных определяет гибкость СУБД. Учет особенностей обработки данных в какой-либо предметной области позволяет спроектировать специализированные СУБД, ориентированные на применение в соответствующей предметной области.

Существуют и универсальные СУБД, используемые для различных приложений. Процесс настройки СУБД на конкретную область применения называется генерацией системы. К универсальным относятся следующие системы: dBase, Paradox, Microsoft Access, Oracle.

Кроме важнейших составляющих БД и СУБД банк данных включает ряд других составляющих. Остановимся на их кратком рассмотрении.

Языковые средства включают языки программирования, языки запросов и ответов, языки описания данных.

Методические средства — это инструкции и рекомендации по созданию и функционированию БНД, выбору СУБД.

Технической основой БНД является ЭВМ, удовлетворяющая определенным требованиям по своим техническим характеристикам.

Обслуживающий персонал включает программистов, инженеров по техническому обслуживанию ЭВМ, административный аппарат, в том числе администратора БД. Их задача — контроль за работой БНД, обеспечение совместимости и взаимодействия всех составляющих, а также управление функционированием БНД, контроль за качеством информации и удовлетворение информационных потребностей. В минимальном варианте все эти функции для пользователя могут обеспечиваться одним лицом или выполняться организацией, поставляющей программные средства и выполняющей их поддержку и сопровождение.

Особую роль играет *администратор базы или банка данных* (АБД). Администратор управляет данными, персоналом, обслуживающими БНД. Важной задачей администратора БД является защита данных от разрушения, несанкционированного и некомпетентного доступа. Администратор предоставляет пользователям большие или меньшие полномочия на доступ ко всей или части базы. Для выполнения функций администратора в СУБД предусмотрены различные служебные программы. Администрирование базой данных предусматривает выполнение функций обеспечения надежной и эффективной работы БД, удовлетворение информационных потребностей пользователей, отображение в базе данных динамики предметной области.

Главными пользователями баз и банков данных являются конечные пользователи, т.е. специалисты, ведущие различные участки экономической работы. Их состав неоднороден, они различаются по квалификации, степени профессионализма, уровню в системе управления: главный бухгалтер, бухгалтер, операционист, начальник кредитного отдела и т.д. Удовлетворение их информационных потребностей — это решение большого числа проблем в организации внутримашинного информационного обеспечения.

Специальную группу пользователей БНД образуют прикладные программисты. Обычно они играют роль посредников между БД и конечными пользователями, так как создают удобные пользовательские программы на языках СУБД. Централизованный характер управления данными вызывает необходимость администрирования такой сложной системы, как банк данных.

Преимущества работы с БНД для пользователя окупают затраты и издержки на его создание, так как:

- повышается производительность работы пользователей, достигается эффективное удовлетворение информационных потребностей;
- централизованное управление данными освобождает прикладных программистов от организации данных, обеспечивает независимость прикладных программ от данных;
- развитая организация БД позволяет выполнять разнообразные нерегламентированные запросы, новые приложения;
- снижаются затраты не только на создание и хранение данных, но и на поддержание в актуальном и динамичном состоянии; уменьшаются потоки данных, циркулирующих в системе, сокращается их избыточность и дублирование.

Как банк данных, так и база данных могут быть сосредоточены на одном компьютере или распределены между несколькими компьютерами. Для того, чтобы данные одного исполнителя были доступны другим и наоборот, эти компьютеры должны быть соединены в единую вычислительную систему с помощью вычислительных сетей.

Банк и база данных, расположенные на одном компьютере, называются *локальными*, а на нескольких соединенных сетях ПЭВМ называются *распределенными*. Распределенные банки и базы данных более гибки и адаптивны, менее чувствительны к выходу из строя оборудования.

Назначение локальных баз и банков данных состоит в организации более простого и дешевого способа информационного обслуживания пользователей при работе с небольшими объемами данных в решении несложных задач.

Локальные базы данных *эффективны* при работе одного или нескольких пользователей, когда имеется возможность согласования их деятельности административным путем. Такие системы просты и надежны за счет своей локальности и организованной независимости.

Назначение распределенных баз и банков данных состоит в предоставлении более гибких форм обслуживания множеству удаленных пользователей при работе со значительными объемами информации в условиях географической или структурной разобщенности. Распределенные системы баз и банков данных обеспечивают широкие возможности по управлению сложных многоуровневых объектов и процессов.

Распределенная обработка данных позволяет разместить базы данных (или несколько баз) в различных узлах компьютерной сети. Таким образом, каждый компонент базы данных располагает-

ся по месту наличия техники и ее обработки. Например, при организации сети филиалов какой-либо организационной структуры удобно обрабатывать данные в месте расположения филиала. Распределение данных осуществляется по разным компьютерам в условиях реализации вертикальных и горизонтальных связей для организаций со сложной структурой.

В распределенных системах баз и банков данных, которые являются средством автоматизации крупных организаций, появляются новые проблемы. Увеличение числа пользователей, расширение географических размеров системы, увеличение физических узлов сети усложняет администрирование. Создается угроза рассогласования данных, хранящихся в различных частях системы. Возникает проблема целостности и безопасности данных, которая решается совокупностью средств, методов и мероприятий.

9.4.7. Интегрированные технологии в распределенных системах обработки данных

Многообразие компьютерных сетей и форм взаимодействия компьютеров порождает насущную проблему их интеграции или по крайней мере соединения на уровне обмена информацией.

В распределенных системах используются три интегрированные технологии:

1. Технология «клиент — сервер».
2. Технология совместного использования ресурсов в рамках глобальных сетей.
3. Технология универсального пользовательского общения в виде электронной почты

1. Основная форма взаимодействия ПК в сети — это «клиент—сервер». Обычно один ПК в сети располагает информационно-вычислительными ресурсами (такими, как процессоры, файловая система, почтовая служба, служба печати, базы данных), а другие ПК пользуются ими. Компьютер, управляющий тем или иным ресурсом, принято называть сервером этого ресурса, а компьютер, желающий им воспользоваться, — клиентом. Если ресурсом являются базы данных, то говорят о сервере баз данных, назначение которого обслуживать запросы клиентов, связанные с обработкой данных; если ресурс — файловая система, то говорят о файловом сервере или файл-сервере и т. д.

Технология «клиент — сервер» получает все большее распространение, но реализация технологии в конкретных программах существенно различается.

Один из основных принципов технологии «клиент — сервер» заключается в разделении операций обработки данных на три группы, имеющие различную природу.

Первая группа — это ввод и отображение данных.

Вторая группа объединяет прикладные операции обработки данных, характерные для решения задач данной предметной области. Наконец, к третьей группе относятся операции хранения и управления данными (базами данных или файловыми системами).

В соответствии с этим выделяют три модели реализации технологии «клиент—сервер»:

- модель доступа к удаленным данным (Remote Data Access — RDA);
- модель сервера базы данных (DataBase Server — DBS);
- модель сервера приложений (Application Server — AS).

В RDA — модели программы представления и прикладные программы объединены и выполняют на компьютере-клиенте, который поддерживает как операции ввода и отображения данных, так и прикладные операции. Доступ к информационным ресурсам обеспечивается или операциями языка SQL, если речь идет о базах данных, или вызовами функций специальной библиотеки. Запросы к информационным ресурсам направляются по сети удаленному компьютеру, например, серверу базы данных, который обрабатывает запросы и возвращает клиенту необходимые для обработки блоки данных (рис. 9.19).

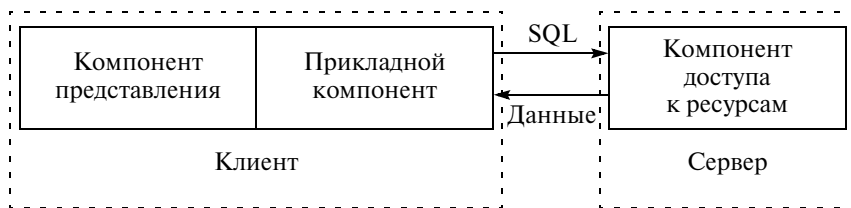


Рис. 9.19. Модель доступа к удаленным данным

DBS-модель строится в предположении, что программы, выполняемые на компьютере-клиенте, ограничиваются вводом и отображением, а прикладные программы реализованы в процедурах базы данных и хранятся непосредственно на компьютере — сервере базы данных вместе с программами, управляющими и доступом к данным — ядру СУБД (рис. 9.20).



Рис. 9.20. Модель сервера базы данных

На практике часто используются смешанные модели, когда поддержка целостности базы данных и простейшие операции обработки данных поддерживаются хранимыми процедурами (DBS-модель), а более сложные операции выполняются непосредственно прикладной программой, которая выполняется на компьютере-клиенте (RDA-модель).

В AS-модели программа, выполняемая на компьютере-клиенте, решает задачу ввода и отображения данных, т.е. реализует операции первой группы. Прикладные программы выполняются одним либо группой серверов приложений (удаленный компьютер или несколько компьютеров). Доступ к информационным ресурсам, необходимым для решения прикладных задач, обеспечивается так же, как и в RDA-модели. Прикладные программы обеспечивают доступ к ресурсам различных типов — базам данных, индексированным файлам, очередям и др. RDA-модели опираются на двухзвенную схему разделения операций, где прикладная программа выделена как важнейшая (рис. 9.21).

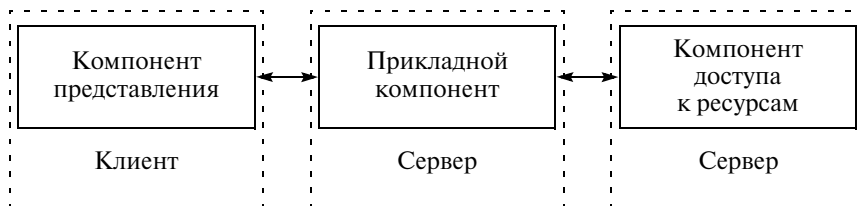


Рис. 9.21. Модель сервера приложений

Главное преимущество RDA-модели состоит в том, что она представляет множество инструментальных средств, которые обеспечивают быстрое создание приложений, работающих с SQL-ориентированными СУБД. Иными словами, основное достоинство

RDA-модели заключается в унификации и широком выборе средств разработки приложений. Подавляющее большинство этих средств разработки на языках четвертого поколения, включая и средства автоматизации программирования, обеспечивают разработку прикладных программ и операций представления.

Несмотря на широкое распространение, RDA-модель постепенно уступают место более технологичной DBS-модели. Последняя реализована в некоторых реляционных СУБД (Interbase, SyBase, Oracle).

1. В течение последнего десятилетия получают все более широкое развитие *глобальные вычислительные и информационные сети* — уникальный симбиоз компьютеров и коммуникаций. Идет активное включение всех стран во всемирные сетевые структуры. Мировой системой компьютерных коммуникаций ежедневно пользуются более 30 млн. человек. Возрастает потребность в средствах структурирования, накопления, хранения, поиска и передачи информации. Удовлетворению этих потребностей служат информационные сети и их ресурсы. Совместное использование ресурсов сетей (библиотек программ, баз данных, вычислительных мощностей) обеспечивается технологическим комплексом и средствами доступа.

2. *Глобальные сети* (Wide Area Network, WAN) — это телекоммуникационные структуры, объединяющие локальные информационные сети, имеющие протокол связи, методы подключения и протоколы обмена данными. Каждая из глобальных сетей (Internet, Bitnet, Decnet и др.) организовывалась для определенных целей, а в дальнейшем расширялась за счет подключения локальных сетей, использующих ее услуги и ресурсы.

Крупнейшей глобальной информационной сетью является Internet. Передача данных в этой сети организована на основе протокола Internet-IP (Internet Protocol), представляющего собой описание работы сети, которое включает правила налаживания и поддержания связи в сети, обращения с IP-пакетами и их обработки, описания сетевых пакетов семейства IP. Сеть спроектирована таким образом, что пользователь не имеет никакой информации о конкретной структуре сети. Чтобы послать сообщение по сети, компьютер размещает данные в некий «конверт», называемый, например, IP, с указанием конкретного адреса.

Процесс совершенствования сети идет непрерывно, большинство новаций происходит незаметно для пользователей. Любой желающий может получить доступ к сети.

В России подключение к Internet началось в начале 1990-х годов. Первыми пользователями Internet в России стали: ИАЭ

им. Курчатова, МГУ, Госкомвуз, МГТУ им. Баумана, НГУ и некоторые другие научные организации и вузы.

Архитектура сетевых протоколов TCP/IP, на основе которых построена Internet, предназначена специально для объединенной сети. Сеть может состоять из совершенно разнородных подсетей, соединенных друг с другом шлюзами. В качестве подсетей могут выступать локальные сети (Token Ring, Ethernet, пакетные радиосети и т.п.), национальные, региональные и специализированные сети, а также другие глобальные сети, например, Bitnet или Sprint. К этим сетям могут подключаться машины разных типов. Каждая из подсетей работает в соответствии со своими специфическими требованиями и имеет свою природу связи, сама разрешает свои внутренние проблемы.

Однако предполагается, что подсеть может принять пакет информации и доставить его по указанному в этой подсети адресу. Таким образом, две машины, подключенные к одной подсети, могут напрямую обмениваться пакетами, а если возникает необходимость передать сообщение машине другой подсети, то вступают в силу межсетевые соглашения, для чего подсети используют межсетевой язык — протокол IP. Сообщение передается по цепочке шлюзов и подсетей пока оно не достигнет нужной подсети, где доставляется непосредственно получателю. Аналогом Internet в России является сеть EU net / Relcom.

Основная задача Relcom — обеспечить не только доступ к компьютерным ресурсам, но и взаимодействие различных профессиональных групп, рассредоточенных на большой территории.

В настоящее время сеть акционерного общества Relcom является скорее средством общения разработчиков новых решений, чем частью устойчивых общественных структур.

Предполагается, что дальнейшее развитие глобальной сети приведет к появлению специализированных сетей, отражающих потребности конкретных групп общения (например, муниципальных, банковских, биржевых сетей для обмена информацией в области науки и образования).

Relcom объединяет пользователей почти 2500 организаций, расположенных в более чем 200 городах России и государствах СНГ.

Узловые машины сети осуществляют передачу почтовых сообщений и новостей между регионами и распространение сообщений на своей территории. Пользовательские персональные машины под управлением операционных систем UNIX или MS DOS используют для общения с региональными узлами протокол UUCP. Скорость обмена от 1200 до 9600 бит / с. Крупные региональные центры обмениваются сообщениями со скоростью 19,2 Кбит/с.

Используются коммутируемые телефонные линии специализированная телефонная сеть и выделенные линии, протоколы UUCP и TCP / IP (в зависимости от возможностей физических каналов).

Для дальнейшего развития услуг сети в 1995—2000 гг. планировалось расширить число информационных источников, организовать специализированные экспертные услуги, обеспечить возможность доставки «электронных» писем с использованием факсимильной связи. Техническое развитие сети прежде всего связывается с повышением пропускной способности каналов связи, широким переходом на протоколы более высокого уровня и расширением сервиса, предоставляемого пользователю.

Другим примером российской глобальной телекоммуникационной системы является сеть «Спринт».

«Спринт» — это система, созданная с целью обмена финансовой и деловой информацией между абонентами сети. Сеть обеспечивает интегрированные решения в области телекоммуникаций, высокую надежность, скорость и мировое качество услуг связи. Официальным поставщиком услуг связи является АО «Спринт-сеть». С 1995 г. «Спринт-сеть» предлагает широкий спектр услуг, таких, как электронная почта, факсимильная, телетайпная связь, предоставляет доступ к информационным ресурсам и финансовым базам данных абонентам сети, доступ в глобальную сеть Internet, возможность банковских платежей, услуги финансово-информационной системы Reuters, телекоммуникационные услуги в системах платежей на основе пластиковой карточки, осуществляет создание глобальных и частных клиентских сетей.

В России «Спринт-сеть» имеет свои центры доступа примерно в 150 городах. АО «Спринт-сеть» предоставляет услуги в области передачи данных в России и ежедневно передает несколько десятков гигабайт информации между своими клиентами.

3. Электронная почта является популярной услугой вычислительных сетей, и поставщики сетевых операционных систем комплектуют свои продукты средствами поддержки электронной почты.

Электронная почта в локальных сетях обеспечивает передачу документов, успешно используется при автоматизации конторских работ. При использовании для связи между сотрудниками всего офиса она оказывается удобнее телефона, так как позволяет передавать такую информацию, как отчеты, таблицы, диаграммы и рисунки, которые по телефону передать трудно.

Передача между терминалами сообщений, например, фототелеграмм, может также рассматриваться как разновидность элект-

ронной почты. Однако для большинства конкретных случаев использование электронной почты предполагает передачу сообщений через специальные «почтовые ящики», между которыми размещаются устройства обработки данных. «Почтовый ящик» — общая область памяти вычислительной сети, предназначенная для записи информации с помощью одной прикладной программы с целью ее дальнейшего использования другими прикладными программами, функционирующими в других узлах сети.

Электронная почта глобальных сетей передачи сообщений, где могут объединяться компьютеры самых различных конфигураций и со вместимостей, обеспечивает:

- работу в онлайн-режиме, когда не требуется постоянного присутствия на почтовом узле. Достаточно указать специальной программе-почтовику (Mailer) время системных событий и адреса, где следует забирать почту;
- доступ к телеконференциям (Echo Conference);
- доступ к файловым телеконференциям (File Conference).

Файловые телеконференции отличаются от обычных тем, что в качестве сообщений в них существует не письма, а файлы. Например, создается файловая телеконференция, посвященная экономике, где каждый участник конференции может поместить свой файл, а другие участники этот файл непременно получат.

Существуют и другие возможности, предоставляемые членам сети. Можно, например, послать заказ на посылку или прием факса. Составляется обычное электронное письмо, оформленное должным образом, и посылается на адрес компьютерного узла, занимающегося факсимильными операциями. Текст этого письма в виде факса будет доставлен на факсимильный аппарат адресата.

К преимуществам электронной почты относятся скорость и надежность доставки корреспонденции, относительно низкая стоимость услуг, возможность быстро ознакомиться с сообщением широкий круг пользователей.

Любая система электронной почты состоит из двух главных подсистем :

клиентского программного обеспечения, с которым непосредственно взаимодействует пользователь;

серверного программного обеспечения, которое управляет приемом сообщения от пользователя, передачей сообщения в почтовый ящик адресата и его хранение в этом ящике до тех пор, пока пользователь-получатель его оттуда не достанет.

Серверное программное обеспечение при совместимости протоколов передачи данных может обрабатывать почту, подго-

товленную различными клиентскими программами. Это программное обеспечение различается уровнями производительности, надежности, устойчивостью к ошибкам, возможностями расширения.

Клиентское программное обеспечение предоставляет пользователям удобные средства для работы с почтой.

Несмотря на их многообразие в различных системах электронной почты все они *имеют общие функции*: оповещение о прибытии новой почты, чтение входящей почты, создание исходящей почты, адресация сообщений, использование адресной книги, содержащей список абонентов, которым часто посылают почту, отправка сообщений, обработка сообщений и их сохранение.

К обработке сообщений относятся такие функции, как печать, удаление, переадресация письма, сортировка, архивирование сообщений, хранение связанных сообщений. Особо следует выделить программы, позволяющие работать с папками, создавать свои папки для хранения в них сообщений по различным темам. Это очень удобно и помогает быстрее и эффективнее обрабатывать почту.

9.4.8. Технология использования экспертных систем

Создание и использование экспертных систем являются одним из концептуальных этапов развития информационных технологий. В основе интеллектуального решения проблем в некоторой предметной области лежит принцип воспроизведения знаний опытных специалистов — экспертов.

Исходя из собственного опыта эксперт анализирует ситуацию и распознает наиболее полезную информацию, оптимизирует принятие решений, отсекая тупиковые пути.

Экспертная система — это совокупность методов с средств организации, накопления и применения знаний для решения сложных задач в некоторой предметной области. Экспертная система достигает более высокой эффективности за счет перебора большого числа альтернатив при выборе решения, опираясь на высококачественный опыт группы специалистов, анализируя влияние большого объема новых факторов, оценивая их при построении стратегий, добавляя возможности прогноза.

Основой экспертной системы является совокупность знаний (базы знаний), структурированных в целях формализации процесса принятия решений.

Экспертные системы разрабатываются с расчетом на обучение и способны обосновать логику выбора решения, т.е. облада-

ют свойствами адаптивности и ее аргументирования. У большинства экспертных систем имеется механизм объяснения. Этот механизм использует знания, необходимые для объяснения того, каким образом система пришла к данному решению. Очень важным является определение области применения экспертной системы, границ ее использования и действия.

В развитии информационного обеспечения автоматизированных информационных технологий управления экономической деятельностью наибольший интерес представляют применения в области искусственного интеллекта. Одной из форм реализации достижений в этой области является создание экспертных систем — специальных компьютерных систем, базирующихся на системном аккумулировании, обобщении, анализе и оценке знаний высококвалифицированных специалистов — экспертов. В экспертной системе используется база знаний, в которой представляются знания о конкретной предметной области.

База знаний — это совокупность моделей, правил и факторов (данных), порождающих анализ и выводы для нахождения решений сложных задач в некоторой предметной области.

Выделенные и организованные в виде отдельных, целостных структур информационного обеспечения знания о предметной области становятся явными и отделяются от других типов знаний, например, общих знаний. Базы знаний позволяют выполнять рассуждения не только и не столько на основе формальной (математической) логики, но и на основе опыта, фактов, эвристик, т.е. они приближены к человеческой логике.

Разработки в области искусственного интеллекта имеют целью использование больших объемов высококачественных специальных знаний о некоторой узкой предметной области для решения сложных, неординарных задач.

База знаний является основой экспертной системы, она накапливается в процессе ее построения. Знания выражаются в явном виде, позволяющем сделать явным способ мышления и решения задач, и организованы так, чтобы упростить принятие решений. База знаний, обуславливающая компетентность экспертной системы, воплощает в себе знания специалистов учреждения, отдела, опыт группы специалистов и представляет собой институциональные знания (свод квалифицированных, обновляющихся стратегий, методов, решений) (рис. 9.22).

Знания и правила работы рассматриваются в различных аспектах:

- глубинные и поверхностные;
- качественные и количественные;

- приближенные (неопределенные) и точные (определенные);
- конкретные и общие;
- описательные и предписывающие.

Содержание базы знаний может быть применено пользователем для получения эффективных управленческих решений. На рис. 9.23 показаны структура базы знаний и ее функционирование.

Эксперт — это специалист, умеющий находить эффективные решения в конкретной области.

Блок приобретения знаний отражает накопление базы знаний, этап модификации знаний и данных. База знаний отражает возможность использования высококачественного опыта на уровне мышления квалифицированных специалистов, что делает экспертную систему рентабельной в соответствии с нуждами бизнеса и заказчика.

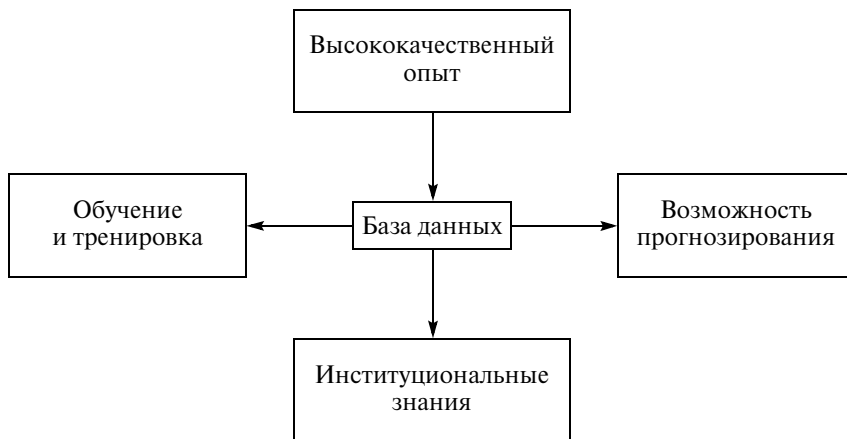


Рис. 9.22. Основные свойства базы знаний

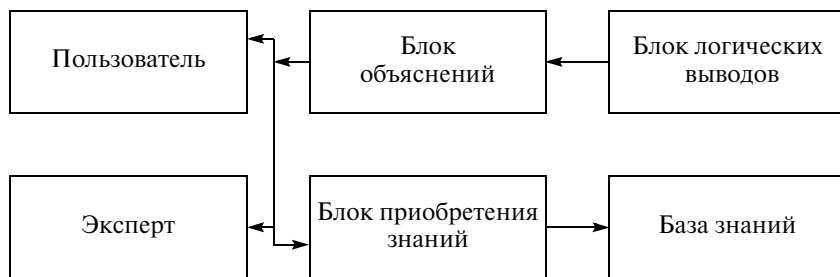


Рис. 9.23. Технология использования базы знаний

Блок логических выводов, осуществляя сопоставление правил с фактами, порождает цепочки выводов. При работе с ненадежными данными формируется нечеткая логика, слабые коэффициенты уверенности, низкая степень меры доверия и т.д.

Блок объяснений отражает в технологии использования базы знаний пользователем последовательность шагов, которые привели бы к тому или иному выводу с возможностью ответа на вопрос «почему».

К настоящему времени распространение баз знаний в значительной степени определяется темпами накопления профессиональных знаний.

Та область профессиональной деятельности, которая пока поддается формализации, а значит, и автоматизации на базе ЭВМ, — это небольшая часть накопленных человеком знаний.

В составе накопления знаний огромный слой составляют индивидуально накапливаемые неотчуждаемые знания. Меньший объем составляют знания, которые доступны для традиционной передачи. И, наконец, едва различимые в общем объеме всех остальных знаний — это формализованные знания.

Структуризация или формализация знаний основана на различных способах *представления знаний*. В современных системах самый популярный способ использует *факты и правила*. Они обеспечивают естественный способ описания процессов в некоторой предметной области.

Правила обеспечивают формальный способ представления рекомендаций, указаний, стратегий. Они подходят в тех случаях, когда предметные знания возникают из опытных (эмпирических) ассоциаций, накопленных за годы работы по решению задач в данной области. Правила чаще всего выражаются в виде утверждений типа «если — то».

Описание предметной области в базе знаний предполагает разработку способов представления и организации знаний, методов формулирования, переформирования и решения задач. Понятия (объекты) предметной области представляются с помощью символов. Например, для банковской системы это могут быть: клиент, фондовый инструмент, операции, задача и т.д. Между символическими понятиями определяются отношения, применяются различные стратегии (логические или полученные в результате опыта) для манипулирования понятиями. Представление знаний, их структуризация предполагает выбор понятий, сложных неординарных задач. Поэтому и правила в базе знаний бывают либо сложными, либо множественными и объемными.

Экспертные системы как инструмент в работе пользователей совершенствуют свои возможности решать сложные, неординарные задачи в ходе практической работы.

Экспертные системы создаются для решения разного рода проблем, типы которых можно сгруппировать в категории (табл. 7.5).

Таблица 9.9

Типичные категории применения экспертных систем

Категория	Решаемая проблема
Интерпретация	Описание ситуации по информации, поступающей от датчиков
Прогноз	Определение вероятных последствий заданных ситуаций
Диагностика	Выявление причин неправильного функционирования системы по результатам наблюдений
Проектирование	Построение конфигурации объектов при заданных ограничениях
Планирование	Определение последовательности действий
Наблюдение	Сравнение результатов наблюдений с ожидаемыми результатами
Отладка	Составление рецептов исправления неправильного функционирования системы
Ремонт	Выполнение последовательности предписанных исправлений
Обучение	Диагностика, отладка и исправление поведения обучаемого
Управление	Управление поведением системы как целого

Ниже перечислены некоторые из предметных областей, в которых применяются экспертные системы. Из них особенно популярна медицина.

Области применения экспертных систем:

- | | |
|----------------------|-----------------------|
| Военное дело | Метеорология |
| Геология | Промышленность |
| Инженерное дело | Сельское хозяйство |
| Информатика | Управление процессами |
| Компьютерные системы | Физика |
| Космическая техника | Химия |
| Математика | Электроника |
| Медицина | Юриспруденция |

Преимущества экспертных систем по сравнению с использованием опытных специалистов состоит в следующем:

- достигнутая компетентность не утрачивается, может документироваться, передаваться, воспроизводиться и наращиваться;
- имеют место более устойчивые результаты, отсутствуют эмоциональные и другие факторы человеческой ненадежности;
- высокая стоимость разработки уравнивается низкой стоимостью эксплуатации, возможностью копирования, а в совокупности она дешевле высококвалифицированных специалистов.

Недостатком экспертных систем, характерным для их современного состояния, является меньшая приспособляемость к обучению новым правилам и концепциям, к творчеству и изобретательству. Использование экспертных систем позволяет во многих случаях отказаться от высококвалифицированных специалистов, но предполагает оставить в системе место эксперту с более низкой квалификацией. Экспертные системы служат средством для расширения и усиления профессиональных возможностей конечного пользователя.

Наиболее уязвимы экспертные системы в распознавании границ своих возможностей, они также демонстрируют ненадежное функционирование вблизи границ их применения. Дальнейший прогресс в области искусственного интеллекта со временем предложит способы выявления границ своих возможностей. Другим недостатком экспертных систем являются значительные трудозатраты, необходимые для пополнения базы знаний. Получение знаний от экспертов и внесение их в базу знаний представляет собой сложный процесс, сопряженный с значительными затратами времени и средств. Проектирование экспертных систем также имеет определенные трудности и ограничения, которые влияют на их разработку.

Экспертная система должна демонстрировать компетентность, т.е. достигать в конкретной предметной области того же уровня, что и специалисты-эксперты. Недостаточно находить хорошие решения, это надо делать быстро. Системы должны иметь не только глубокое, но и достаточно широкое понимание предмета. Методы нахождения решений проблем достигаются на основе рассуждений, исходящих из фундаментальных принципов в случае некорректных данных или неполных наборов правил. Такие свойства наименее разработаны в компьютерных экспертных

системах, но именно присущи специалистам высокого профессионального уровня.

Отличиями экспертных систем от обычных компьютерных являются:

- экспертные системы манипулируют знаниями, тогда как любые другие системы — данными;
- экспертные системы, как правило, дают эффективные оптимальные решения и способны иногда ошибаться, но в отличие от традиционных компьютерных систем они имеют потенциальную способность учиться на своих ошибках.

Развитие концепции баз знаний связано с исследованиями и достижениями в области систем искусственного интеллекта. Область применения баз знаний и систем на их основе расширяются. Создается целый спектр баз знаний — от небольших по объему, для портативных систем, до мощных, предназначенных для профессионалов, эксплуатирующих сложные и дорогие АРМ. Очень большие базы знаний хранятся в централизованных хранилищах, доступ к которым осуществляется через сети пользователями различных систем, уровней, масштабов и т.д. Успехи в разработке баз знаний сделают их доступными для массового пользователя, что будет способствовать их появлению как актуального коммерческого продукта.

Зарубежный опыт показывает, что экспертные системы разрабатываются в основном в университетах, научно-исследовательских центрах и коммерческих организациях, в том числе и для финансовой индустрии. В сфере финансового обслуживания эти системы помогают страховым компаниям анализировать и оценивать коммерческий риск, устанавливать размеры ссуд при кредитовании организаций, составлять сметы проектов и т.д.

Область применения экспертных систем расширяется. Кроме охвата различных областей деятельности, одним из наиболее важных последствий разработки экспертных систем является модификация знаний. По мере того как разработчики будут строить большие, сложные базы знаний, появляется рынок знаний, независимых от компьютерных систем. Появятся средства обучения для изучающих определенную прикладную область. Коммерческим продуктом станут метазнания, т.е. знания об оптимальных стратегиях и процедурах использования предметных знаний. Развитие экспертных систем в интеллектуальные состоит в слиянии концепций оборудования, средств их создания (языков) и самих

экспертных систем. Объединение интеллектуальных систем особенно эффективно в сложных инфраструктурах. Интеллектуальные системы уже разрабатываются и внедряются за рубежом для коммерческого использования.

Так, например, экспертная система FOLIO (Стенфордский университет США) помогает консультантам по инвестициям определять цели клиентов и подбирать портфели ценных бумаг, наиболее соответствующие этим целям. Система определяет нужды клиента в ходе интервью и затем рекомендует, в каких пропорциях надо распределить капиталовложения между разными фондовыми инструментами, чтобы наилучшим образом удовлетворить запросы клиента.

Система различает небольшое число классов ценных бумаг (например, ориентированные на дивиденды акций с невысоким уровнем риска или ориентированные на дивиденды акций с высоким уровнем риска) и содержит знания о свойствах (например, годовых процентах на капитал) ценных бумаг каждого класса. В системе применена основанная на правилах схема представления знаний с прямой цепочкой рассуждений для вывода целей и схема линейного программирования для максимизации соответствия между целями и предлагаемым портфелем. Система доведена до уровня демонстрационного прототипа.

Искусственная компетентность экспертных систем не заменяет полностью человека. Эксперт-человек способен реорганизовать информацию и знания и использовать их для синтеза новых знаний. В области творческой деятельности люди обладают большими способностями по сравнению с самыми умными системами. Эксперты справляются с неожиданными поворотами событий и, используя новые подходы, способны проводить аналогии их других предметных областей. Эксперты адаптируют к изменяющимся условиям и приспособливают свои стратегии к новым обстоятельствам в более широком диапазоне проблем и задач. Экспертные системы менее приспособлены к обучению на уровне новых концепций и новых правил. Они оказываются не столь эффективны и мало пригодны в тех случаях, когда надо учитывать всю сложность реальных задач.

Эксперты могут непосредственно воспринимать весь комплекс входной информации: символьной, визуальной, графической, текстовой, звуковой, осязательной, обонятельной. У экспертной системы есть только символы, с помощью которых представлены базы знаний, воплощающие те или иные концепции. Преобразование сенсорной информации в символьную сопровождается потерей части информации.

Но главное, что огромный объем знаний, которым обладают эксперты-специалисты (профессиональные знания и знания о мире и действующих в нем законах), не удастся пока встроить в интеллектуальную систему, тем более столь специализированную, какой является любая экспертная система.

9.5. Лингвистическое, организационно-методическое, эргономическое и правовое обеспечение автоматизированного управления

Лингвистическое обеспечение автоматизированного управления включает языки общения с пользователем, языки запросов, информационно-поисковые языки, языки-посредники в сетях. Языковые средства автоматизированного управления необходимы для однозначного смыслового соответствия действий пользователя и аппаратной части в виде ПЭВМ. Без них практически невозможны процесс обучения, организация диалога, обнаружение и исправление ошибок. При этом языковые средства для конечного пользователя должны быть непроцедурными, т.е. указывать, что необходимо выполнить без подробной детализации, какие действия для этого требуются. Поскольку пользователям не нужно знать детали реализации своих информационных потребностей, то чем выше «интеллектуальность» автоматизированного управления, тем больше непроцедурных возможностей необходимо предусматривать в языковых средствах.

Одновременно языки в автоматизированном управлении должны быть пользовательско-ориентированными, в том числе профессионально-ориентированными. Использование естественного языка, несмотря на кажущуюся простоту и удобство, не приносит ощутимых преимуществ из-за необходимости введения через клавиатуру громоздких инструкций на естественном языке.

Как правило, основу языков автоматизированного управления должны составлять заранее определяемые термины, а также описание способов, с помощью которых могут устанавливаться новые термины, заменяться или дополняться существующие. Возможности языка во многом определяют также списки правил, на основе которых пользователь может строить формальные конструкции, соответствующие реализации информационной потребности. Например, в одних автоматизированных системах данные и конструкции представляются в виде таблиц, в других — в виде операторов специального вида.

Языковые средства автоматизированного управления можно разделить по видам диалога. Средства поддержки диалога в конечном счете и определяют те языковые конструкции, знание которых необходимо пользователю. При этом следует заметить, что в одной и той же автоматизированной системе может быть реализовано несколько типов диалога:

- диалог, инициируемый ЭВМ;
- диалог с помощью заполнения шаблонов;
- диалог с использованием меню;
- гибридный и др.

9.5.1. Особенности технологии взаимодействия пользователя с ЭВМ

Здесь несколько подробнее остановимся на проблеме эффективного взаимодействия человека и ЭВМ, которая возникла с появлением первых вычислительных машин. На начальном этапе внедрения ЭВМ непосредственно с ними работали только программисты и операторы, общение же специалистов в управленческой деятельности происходило опосредованно путем выдачи программистам заданий на разработку программ решения задач и последующего получения распечаток готовых результатов вычисления на ЭВМ.

Расширение сфер использования ЭВМ, совершенствование аппаратных и программных средств ЭВМ, а главное, массовое появление персональных компьютеров привело к тому, что в процесс непосредственного взаимодействия с ЭВМ был вовлечен массовый, или конечный, пользователь. Под *конечным пользователем* будем понимать специалистов сферы организационно-технического управления, в интересах которых создается система обработки экономической информации на ЭВМ.

С учетом принятого определения конечных пользователей можно разделить на две большие группы. Представители первой группы занимаются прежде всего вводом информации в систему и решением регламентных задач на ЭВМ, т. е. заранее predeterminedенных и решаемых, как правило, с заданной периодичностью. Задача пользователей второй группы — на основании получаемой информации готовить и принимать управленческие решения.

Специфика работы конечных пользователей, особенно пользователей второй группы, требует создания для них таких средств и методов общения с вычислительной системой, благодаря которым, не зная архитектуры и принципов функционирования ЭВМ, не владея профессиональными приемами программирова-

ния, они могли бы удовлетворять свои информационные потребности в ходе взаимодействия с машиной.

Человек и компьютер в процессе решения различных задач, связанных с обработкой информации, могут взаимодействовать в различных режимах.

Различают *следующие режимы* взаимодействия пользователя с ЭВМ: *пакетный* и *интерактивный* (запросный, диалоговый).

Сами же ЭВМ также могут работать в различных режимах: одно- и многопрограммном, разделения времени, реального времени, телеобработки и т.д. При этом предусматривается цель удовлетворения потребности пользователей в максимально возможной автоматизации решения разнообразных задач.

Пакетный режим был наиболее распространен в практике централизованного решения экономических задач, когда большой удельный вес занимали задачи отчетности о производственно-хозяйственной деятельности экономических объектов разного уровня управления.

Организация вычислительного процесса при пакетном режиме строилась без доступа пользователя к ЭВМ. Его функции ограничивались подготовкой исходных данных по комплексу организационно взаимосвязанных задач и передачей их в центр обработки, где формировался пакет, включающий задание ЭВМ на обработку программы, исходные, нормативно-расценочные и справочные данные. Пакет вводился в ЭВМ и реализовывался в автоматическом режиме без участия пользователя и оператора, что позволяло минимизировать время выполнения заданного набора задач. При этом работа ЭВМ могла проходить в однопрограммном или многопрограммном режиме, что предпочтительнее, так как обеспечивалась параллельная работа основных устройств машины. В настоящее время пакетный режим реализуется применительно к электронной печати.

Интерактивный режим предусматривает непосредственное взаимодействие пользователя с информационно-вычислительной системой, может носить характер запроса (как правило, регламентированного) или диалога с ЭВМ.

Запросный режим необходим пользователям для взаимодействия с системой через значительное число абонентских терминальных устройств, в том числе удаленных на значительное расстояние от центра обработки. Такая необходимость обусловлена решением оперативных задач справочно-информационного характера, такими являются, например, задачи резервирования билетов на транспорте, номеров в гостиничных комплексах, выдача справочных сведений и т. п., ЭВМ в подобных случаях реализует

систему массового обслуживания, работает в режиме разделения времени, при котором несколько независимых абонентов (пользователей) с помощью устройств ввода-вывода имеют в процессе решения своих задач непосредственный и практически одновременный доступ к ЭВМ. Этот режим позволяет дифференцированно в строго установленном порядке представлять каждому пользователю время для общения с ЭВМ, а после окончания сеанса отключать его.

Диалоговый режим открывает пользователю возможность непосредственно взаимодействовать с вычислительной системой в допустимом для него темпе работы, реализуя повторяющийся цикл выдачи задания, получения анализа ответа. При этом ЭВМ сама может инициировать диалог, сообщая пользователю последовательность шагов (представление меню) для получения искомого результата.

Обе разновидности интерактивного режима (запросный, диалоговый) основываются на работе ЭВМ в режимах реального времени и телеобработке, которые являются дальнейшим развитием режима разделения времени. Поэтому обязательными условиями функционирования системы в этих режимах являются: во-первых, постоянное хранение в запоминающих устройствах ЭВМ необходимой информации и программ и лишь в минимальном объеме поступление исходной информации от абонентов и, во-вторых, наличия у абонентов соответствующих средств связи с ЭВМ для обращения к ней в любой момент времени.

Рассмотренные технологические процессы и режим работы пользователей в системе «человек — машина» особенно четко проявляются при интегрированной обработке информации, которая характерна для современного автоматизированного решения экономических задач в многоуровневых информационных системах.

Диалоговый режим взаимодействия пользователя и ЭВМ возник как альтернатива пакетному, когда потребовалось обеспечить возможность оперативного вмешательства пользователя в процесс обработки информации на ЭВМ, хотя на практике весьма часто можно наблюдать совместное использование этих режимов, помогающие за счет их частных преимуществ организовать более эффективную технологию решения задач на ЭВМ.

Используя коллективный диалог с вычислительной системой, управленческий персонал организации (предприятия) может вовлечь в автоматизированный процесс формирования производственно-хозяйственных средств большой набор слабо формализуемых факторов и обстоятельств в соответствии со своим

опытом, интуицией и знаниями в реальной экономической ситуации. Особенно это важно в экспертных системах, реализуемых с помощью программно-технических средств интерактивной обработки данных на базе вычислительной техники.

Диалог, или режим диалога, — это активный обмен информационными сообщениями между участниками процесса, когда прием, обработка и выдача сообщений производится в реальном масштабе времени. В литературе диалоговый режим, нередко реализуемый в системах коллективного пользования на основе разделения времени, называются интерактивным режимом.

Диалог может быть парным, когда число его участников равно двум, и множественным при большем числе участников. При парном диалоге субъектами диалога могут выступать как люди, так и технические средства в виде ЭВМ. Причем возможны такие сочетания, как «человек—человек», «человек—ЭВМ» и «ЭВМ—ЭВМ». Однако общие принципы реализации диалога при различных комбинациях остаются неизменными.

В основе машинной диалоговой технологии обработки информации лежит взаимодействие человека и ЭВМ во время решений задачи посредством передачи и приема сообщений через терминальные устройства. При этом необходимо, чтобы у такого процесса была совокупность следующих качеств или хотя бы одно из них:

- наличие у общих партнеров общей цели взаимодействия;
- определенная степень равноценности деятельности в процессе реализации задачи;
- расширение и усовершенствование знаний (умений) одного партнера на основе знаний (умений) другого;
- обмен сообщениями, направленными на установление понимания одним партнером сообщений другого.

Остановимся на некоторых качествах диалогового режима более подробно. При диалоге типа «человек—ЭВМ» целью пользователя является получение системы результатных данных в процессе решения задачи, целью же использования ЭВМ (аппаратной и программной среды) — оказание помощи пользователю при выполнении рутинных операций. Выполнение этого в зависимости от степени «интеллектуализации» ЭВМ допускает либо прямое выполнение необходимых трудоемких рутинных расчетов, либо получение от человека дополнительной информации, хранение больших объемов данных и выдачу результатов решений, либо выбор путей решений задачи.

Взаимопонимание партнерами друг друга достигается при наличии у них единой системы языковых знаков или кодов, из ко-

торых формируются сообщения, и хотя бы частичного понимания каждым из них предметной области. Чем больше количество знаний и умений оказываются у обоих партнеров, тем легче достигается ими понимание друг друга и эффективнее происходит диалог.

В случае, когда роли участников диалога неизменны, однозначны и predeterminedены заранее, структура диалога называется *жесткой*. В простейшем случае такая структура диалога сводится к двум взаимосвязанным высказываниям партнеров («вопрос—ответ») с указанием того, кому из партнеров принадлежит инициатива. Развитием жесткой структуры является альтернативная структура. Она задает множество возможных, но заранее предписанных в структуре направлений течения диалога. Возможные направления течения диалога представляются пользователю в виде списков вариантов, т. е. меню, как правило, иерархической структуры, из которого он должен выбрать устраивающие его дальнейшие действия.

Структура диалога, позволяющая участникам общения изменять ее в ходе диалога некоторым заранее predeterminedенным способом, является *гибкой*. *Свободной* называется такая структура, которая позволяет участникам общения менять ее произвольным образом в ходе диалога.

9.5.2. Типы диалога и формы его реализации на ЭВМ

Анализ различных диалоговых систем, реализующих вычислительные, информационные, справочные и обучающие функции, а также комплексные экономические задачи, показал, что наиболее распространенными типами организации диалога являются меню, шаблоны, команды, естественный язык.

Меню — это тип диалога, который благодаря наглядности имеет широкое распространение при решении самых различных задач на разных классах и моделях ЭВМ. Этот тип диалога наиболее удобен для конечного пользователя, так как средством отображения сообщений между человеком и ЭВМ является экран видеотерминала. Шаг диалога начинается, как правило, с выдачи системой некоторого входного сообщения, альтернативный ответ на которое позволяет продолжить или завершить диалог (рис. 9.24). Завершается шаг диалога после ввода пользователем ответного сообщения и его обработки системой. Выход из диалоговой системы возможен в случае прохождения всей цепочки шагов, обеспечивающей полное решение задач, путем прерывания диалога по инициативе пользователя и выхода из системы.

Машинная реализация диалога типа «меню» возможна несколькими способами, однако, при всех вариантах в качестве входного сообщения на экран видеотерминала выводится некоторое подмножество функций системы, реализация которых возможна в текущем состоянии диалога.

Выбрать требуемую функцию пользователь в зависимости от реализации операции в программной среде диалоговой системы может различными способами:

- a — набором на клавиатуре требуемой директивы или ее сокращенного обозначения;
- b — набором на клавиатуре номера необходимой функции;
- c — подведением курсора в строку экрана с нужной пользователю функцией;
- d — нажатием функциональных клавиш, запрограммированных на реализацию данной функции.

Наиболее экономичными для пользователя являются способы c и d. Способы a и b, несмотря на некоторое увеличение трудоемкости, обеспечивают большую надежность, поскольку исключают возможность случайного набора соседних директив.

Схему диалога условно можно представить в виде графа типа «дерево», в котором вершины — это подмножества функций системы, действия или альтернативные выходные сообщения. В процессе реализации диалога возможны движения вниз по графу и возврат в начало графа или в какие-либо ранее прошедшие вершины.

Наиболее простым частным случаем диалога типа меню является режим ответа ДА / НЕТ, когда пользователю в качестве альтернативы предлагается всего два варианта ответа ДА или НЕТ.

Машинная реализация этого режима может быть осуществлена одним из следующих способов, когда:

- ответ набирается полностью ДА или НЕТ;
- ответ задается с помощью первых букв Д или Н (У или N);
- одна из альтернатив (ДА или НЕТ) предполагается по умолчанию и реализуется клавишей ввода, а другая клавишей отмены;

Второй тип диалога — шаблон. Это инициируемый системой режим взаимодействия конечного пользователя и ЭВМ, на каждом шаге которого система воспринимает только синтаксически ограниченное входное сообщение пользователя в соответствии с

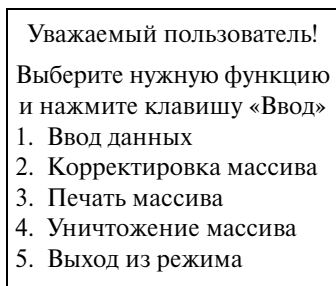


Рис. 9.24. Содержание кадра

заданным форматом. Как и при типе диалога меню, инициативная роль в начале диалога принадлежит ЭВМ. Возможные варианты ответа пользователя ограничиваются форматами, предъявляемые ему на экране видеотерминала. Поэтому гибкость пользователя такой системой относительно невысока, хотя и выше, чем при предыдущих схемах организации, но при этом резко снижается и операционная сложность реализации данного типа диалога на машине.

Данный тип диалога может быть реализован несколькими способами, из которых самые распространенные два:

- указание системой на экране дисплея формата вводимого пользователем сообщения;
- резервирование места для сообщения пользователя в тексте сообщения системы на экране терминала.

Тип диалога *шаблон* обычно используется для ввода, данных значения которых или понятны (например, поле для записи даты, фамилии, названии предприятия и т.д.) или являются профессиональными терминами, известными пользователю по его предметной области.

Следует различать *жесткий шаблон*, когда количество вводимых пользователем символов обязательно должно соответствовать числу разрядов, выделенных тем или иным способом на экране дисплея, и *свободный шаблон*. В последнем случае задается предельно допустимое поле, в которое вносится конкретное значение, например, фамилия работающего при формировании справочника.

Частным случаем данного типа диалога является режим, называемый *простым запросом*. Наибольшее распространение он получил в автоматизированных системах сброса и формирования массивов данных. При этом режиме пользователю представляется возможность вводить массив, состоящий более чем из одного сообщения, по формату, заданному системой. Диалог в этом случае сводится лишь к одному шагу, а в качестве сообщений на экране компьютера могут быть выведены анкетные данные работающих либо номенклатура материальных ценностей и т.п. В качестве примера на рис. 9.25 приведен информационный кадр для создания машинной лицевой карточки работника предприятия.

Диалог типа *команда* представляет собой инициированный пользователем тип диалога, при котором выполняется одна из допустимых на данном шаге диалога директив (команд) пользователя. Перечень допустимых команд, как правило, отсутствует на экране дисплея, однако его можно вызвать на экран с помощью специальной директивы или функциональной клавиши на

ПЭВМ (обычно F1) для ознакомления. Если пользователь ввел ошибочную команду, т.е. отсутствующую в списке или с нарушениями формата или синтаксиса, выдается предупредительное сообщение, и система остается в начале текущего шага диалога, ожидая его продолжения.

Взаимодействием на естественном языке называется такой тип диалога, когда и инициирование и ответ со стороны пользователя ведутся на языке, близком к естественному (как в плане используемого словаря, так и в плане построения сложных языковых конструкций предложений). Пользователю предоставляется возможность свободной формулировки нужной ему задачи, однако, с употреблением обусловленных программой средой слов и фраз, а также заранее установленным синтаксисом языка. В связи с неоднозначностью естественных языков система должна иметь возможность задавать вопросы, уточняющие формулировку пользователя и предметную область рассматриваемой проблемы. При подобном типе диалога пользователь должен быть готов к тому, что диалоговая система может не распознать его запрос с первого раза, а потому требуется повторение или уточнение запроса.

Одной из разновидностей данного диалога является *речевое общение с системой*.

На практике в чистом виде в системе только один тип диалога, как правило, не применяется, а используется сочетание нескольких типов. Это повышает гибкость системы, упрощает ее, делает диалог более разнообразным и интенсивным.

В «дружелюбной» диалоговой системе, ориентированной на непрофессионального пользователя, человек, работая за пультом ЭВМ, должен воспринимать не весь диалог, а лишь визуальное

<p><u>Введите данные в соответствии с форматом</u></p> <p>Фамилия</p> <p>Имя</p> <p>Отчество</p> <p>Год рождения</p> <p>Образование</p> <p>Должность</p> <p>Ученая степень</p>
--

Рис. 9.25. Информационный кадр прямого вопроса

(иногда и звуковое) его отображение. Большая часть программных средств управления диалогом должна быть скрыта от пользователя, а ему предоставляется минимум для поддержки внешнего диалога.

Аппаратные средства современных ЭВМ позволяют отображать на экране дисплея графические и цветные изображения движущихся объектов различной природы, сложные структуры данных и лишь простым нажатием клавиш управлять событиями, возникающими на экране. Кроме того, во многие профессионально — ориентированные ППП экономистов по рекомендации психологов введены игровые моменты с целью предоставления пользователю кратковременного отдыха, например, в случае переутомления и возникающих из-за этого ошибок.

Важным для всех категорий пользователей программных средств, работающих в режиме диалога, является включаемая в них в обязательном порядке система помощи и средств обучения (HELP), существенно ускоряющая как процесс освоения, так и процесс работы. Другое требование к системе — простота работы с пакетом. Освоение основных функций любого пакета диалогового типа не должно требовать специальных знаний в области языков программирования, архитектуры ЭВМ и пр.

Поскольку пользователю приходится работать с различными диалоговыми программными системами, целесообразно закладывать в них некоторое единообразие и определенные сложившиеся традиции, например, использование функциональных клавиш F1 и F10 соответственно для вызова помощи, выхода из системы, применение управляющих клавиш или их комбинаций для управления состоянием процесса вычислений и т.д.

Кроме того, диалоговые системы должны использовать достижения эргономики и современного дизайна. Внешнее привлекательное оформление диалога по гамме цветов, графическому оформлению, расположению различных объектов, многооконность и т.п. не только настраивает пользователя на работу с системой, но и делает ее приятной, менее утомительной и более привлекательной.

В настоящее время все более широкое распространение в качестве формы диалогового взаимодействия пользователя с ЭВМ приобретает его работа в мультимедийной среде, где под термином «мультимедиа» понимается объединение взаимодействия различных каналов передачи информации от машины к человеку (в первую очередь изобразительной и звуковой) и обратная связь: действия человека должны существенно влиять на ход событий, воспроизводимых компьютерной системой.

Возникшие в первую очередь как средство реализации компьютерных игр мультимедийные технологии находят благоприятную почву для применения в обучающих системах, в компьютерной мультипликации и рекламе, в системах автоматизации проектирования, в системах эмулирования военных ситуаций и экспертных системах.

Развитие мультимедийных технологий и возрастание роли, которую они начинают играть в деятельности человека, стимулирует дальнейшее развитие микроэлектронной технологии, совершенствование и разработку новых средств обмена информацией в системе «человек—ЭВМ» (например, обеспечение стереофонического звучания, объемного изображения, тактильных и других средств съема информации).

9.5.3. Организация диалоговой технологии обработки данных на ПЭВМ

Использование для диалоговой обработки информации персональных компьютеров, обладающих такими особенностями, как реактивность, возможность наращивания программных мощностей, портативность, высокая надежность, простота эксплуатации, низкие затраты на эксплуатацию системы, обеспечивает ряд преимуществ по сравнению с системами, реализуемыми на ЭВМ других классов:

1. Обеспечивается доступ пользователей различных категорий (специалистов управления, инженерно — технических работников и др.) к ресурсам вычислительной системы.

2. Увеличивается «интеллект» системы на более гибкой основе, расширяются ее вычислительные и информационные ресурсы, так как терминалы на базе персональных компьютеров могут использоваться любыми работниками сферы организационно-экономического управления.

3. Снижается общая стоимость диалоговой обработки данных из-за сравнительно низкой стоимости ПЭВМ.

4. Повышается общая надежность вычислительной системы, так как отказ в работе отдельной ПЭВМ не приводит к прекращению работы системы, созданной из ряда микрокомпьютеров. При наличии запасных машин или узлов устранение сбоев в работе вычислительной системы может занимать небольшой промежуток времени.

5. Легче осуществляется адаптация пользователя и системы к изменению объектов автоматизации и функций управления.

6. Обеспечивается благодаря активности ПЭВМ требуемое время реакции на нерегулярно поступающие запросы.

7. Происходит максимальное сближение систем обработки информации на базе различных классов ЭВМ с технологическими операциями, выполняемыми работниками аппарата управления с применением персональных компьютеров.

В процессе разработки диалоговых систем рекомендуется учитывать ряд требований, которые можно свести к двум группам:

- определяющие принципы построения систем;
- требования, предъявляемые к эксплуатационным характеристикам диалоговых систем.

В первой группе требований принципиально важными для создания гибкой, надежной и эффективной диалоговой системы являются следующие требования:

- до начала формулирования требований к системе целесообразно ознакомиться с какой-либо действующей системой;
- необходимо предусмотреть адаптивную обратную связь между пользователем и системой, позволяющую разработчику в интерактивном режиме приближать систему к реальным условиям решения задачи;
- предусмотреть ряд мер по контролю и защите вводимой и хранимой информации;

В частности, диалоговые системы решения экономических задач для пользователей — специалистов в области организационного управления должны обеспечивать относительно простые, но надежные сервисные функции по синтаксическому, логическому и численному контролю исходных данных; корректировке хранимой в памяти ЭВМ информации; прерыванию процедуры выполнения алгоритмического процесса с возвратом в ближайшую к прерванной в процедуре точку алгоритма с восстановлением соответствующих ей исходных состояний файлов; фиксировать действия системы для обеспечения безопасности ее путем последующего анализа зарегистрированных сообщений.

В числе требований, предъявляемых к эксплуатационным характеристикам диалоговых систем, выделим такие, как легкая адаптация пользователя к системе; единообразии вычислительных, опросных процедур и терминологии, предоставление возможности получения углубленной информации по мере необходимости в ходе работы с системой; снабжение пользователя справочной информацией и необходимыми инструкциями, выводимыми на экран видеотерминала или печатающее устройство, с указанием моментов получения помощи от ЭВМ или необходимости проведения ответных действий (реакций); обзорность состояния диа-

лога путем использования кратких форм диалога; наличие защитных средств информации в системе, реализуемых операционными системами и специальными программами.

Технология создания диалоговых систем может быть представлена в виде определенной последовательности шагов.

На первом шаге анализируется подлежащая решению задача, выявляются ее характерные признаки, формулируются методы и средства ее решения, уточняются источники информации, объемы информационных массивов, вычислительные ресурсы и т.д.

Второй шаг — отображение этих представлений в виде интерактивного взаимодействия, синтезированного из простейших компонентов процедур, т.е. составление укрепленной логической схемы.

Третий шаг — формирование полного подробного сценария диалогового решения задачи на основе построенной структуры человека — машинного диалога и его отображение в виде информационной базы, информационных кадров, обеспечивающее информационно — вычислительный процесс по заданному сценарию.

Четвертый шаг — обеспечение автоматизированного ведения гибкого диалога путем написания программы на алгоритмическом языке или на базе логико — лингвистического процессора.

Важной проблемой при создании диалоговых систем является обеспечение *реактивности*, т.е. достаточно быстрой циркуляции сообщений как между функциональными задачами (программами), так и между задачами и пользователем. Обмен сообщения должен осуществляться в режиме реального времени, без «зависаний» системы, которые психологически угнетающе действуют на партнеров по диалогу, а в системе «человек — ЭВМ» — на пользователя.

Для конечных пользователей — специалистов управления диалоговая система должна быть достаточно «прозрачной» и требовать от них лишь выполнения обычных действий в соответствии с их служебными обязанностями. При этом пользователь должен получать от системы необходимые разъяснения (подсказки) по содержанию требуемых от него действий.

Возможность обмена пользователя с системой с помощью языка диалога может быть обеспечена двумя способами.

Первый способ предполагает, что с системой общается пользователь-неспециалист. Потому ему не необходимо знать весь объем команд (директив, сообщений) системы и правил их исполнения. Он должен знать лишь свой пароль, свое «меню», т.е. обладать лишь достаточным объемом знаний, система же сама должна подсказать пользователю, что он может отвечать в той или иной ситуации.

При *втором способе* общение с диалоговой системой осуществляет пользователь-специалист (например, администратор системы). В этом случае диапазон выполняемых им функций более широк и может включать такие операции, как:

- изменение состава технологических цепочек, значений параметров, правил запуска и т.д.;
- вывод на экран текстов справочных сообщений, форм таких сообщений и внесение в них новой информации;
- преобразование данных из одной формы представления в другую;
- собственно операции над данными.

В числе функций второй группы:

- контроль состояний всей диалоговой системы и правильности взаимодействия процессов;
- контроль состояния процессов;
- обеспечения надежности функционирования системы в целом и отдельных процессов;
- адаптация и развитие системы: при разработке диалога управления процессами используются файловые системы организации обмена и преобразования данных в памяти ЭВМ, утилиты для работы с программными модулями, специальные языковые средства.

При переходе к массовому применению ПЭВМ в режиме диалога появляется возможность отказа от использования традиционных бумажных носителей информации, работа с которыми трудоемка, требует специальной подготовки операторов и нарушает индивидуальность автоматизированной обработки информации. Использование ПЭВМ, работающих в режиме диалога, в местах возникновения информации (на складах, в цехах, в функциональных управленческих отделах и т.д.) позволяет автоматизировать процесс изготовления и заполнения первичной документации.

Значительно ускоряется также общий процесс технологической обработки данных в распределенных (децентрализованных) системах обработки данных на базе ПЭВМ, работающих в режиме диалога, так как отпадает необходимость использования процедур комплектования документов в пачки. Вместо перфорации данных с документов производится их ручной ввод с клавиатуры или автоматический (например, с помощью сканера), исключается такая трудоемкая операция, как контроль перфорации. Замена перфорации вводом данных позволяет повысить достоверность вводимой информации за счет визуального контроля на экране, применения логико-синтаксического метода контроля и,

кроме всего прочего, снижаются общие затраты сил и средств на проведение операции по формированию отдельных массивов и баз данных.

Особенностью диалоговой технологии для систем обработки данных на базе персональных ЭВМ является применение автоматизированного сбора, регистрации и предварительной обработки данных непосредственно на рабочих местах специалистов управления.

Поскольку на ПЭВМ в режиме диалога может работать не только оператор, но и непосредственно сам конечный пользователь, который знает предметную область решаемой задачи, он может визуально обнаружить не только ошибки, возникшие при вводе, но и не выявленные по том или иным причинам ранее непосредственно в первичных документах.

В условиях сетевого использования ЭВМ автоматизируется доставка исходной информации к месту ее основной обработки и результатной к месту ее использования. Применение каналов связи обеспечивает ввод данных со скоростью в несколько раз более высокой, чем при вводе перфоносителей или простой передаче магнитных носителей в центр обработки. Одновременно за счет создания централизованных и распределенных баз данных достигается сокращение объема вводимой информации, а вместе с тем и общего времени на ее обработку.

Диалоговая технология способствует приближению средств вычислительной техники к пользователю, однако, требует применения повышенных мер по обеспечению защиты информации от несанкционированного доступа. Доступ к машине имеют пользователи, выполняющие различные функции, связанные с решением задач, за которые они несут ответственность, как юридические лица. Для исключения порчи данных и программ, располагающихся на жестких винчестерских дисках ПЭВМ, в структуру диалога включаются дополнительные модули по обеспечению защиты данных от несанкционированного доступа.

Другим аспектом диалоговой технологии на ПЭВМ является юридический. Автоматизация составления первичной документации приводит к ужесточению требований по контролю за работой пользователей, заполняющих первичный документ. Поскольку в заполнении документа могут принимать участие несколько лиц (в данном случае документ толкуется несколько шире и может подразумевать файл или фрагменты базы данных), каждый работник аппарата управления, имеющий доступ к машине, несет юридическую ответственность за корректность вносимой в документ (файл) информации. Это заверяется подписями в документах.

При автоматическом заполнении документа часто происходит разрыв во времени между заполнением в него данных и получением готовой сводки. Поэтому целесообразно проконтролировать, кто, когда и какую информацию вносил в заполняемый документ, какие операции и в какой последовательности выполнял, подтверждать выполнение процедуры «электронными подписями». С этой целью диалог должен поддерживаться соответствующими программными средствами.

Реализуемые на практике диалоговые системы различаются между собой по различным признакам: назначению и виду обрабатываемой информации, технической базе, автономности или включению в более сложную систему, затратам на реализацию и эксплуатацию и т. д.

Поэтому для оценки и выбора той или иной системы, ориентированной на решение однотипного класса задач, должна быть установлена система критериев или показателей оценки и выбора. Признавая ведущую роль человека в диалоге, целесообразно эффективность диалоговой системы определять с точки зрения пользователя. Поэтому основными показателями системы являются ее реактивность, то есть время ответа системы и время на ожидание обслуживания, время решения задачи, степень обеспечения устойчивости и степень «подсказываемости» пользователю возможных вариантов действий в случае непредвиденных ситуаций и ряд других.

9.5.4. Организационно-методическое, эргономическое и правовое обеспечение автоматизированного управления

Организационное обеспечение автоматизированного управления включает комплекс документов, регламентирующих деятельность специалистов при использовании ПЭВМ или терминала на их рабочем месте. При этом возникает необходимость определить функции и задачи каждого специалиста, регламентировать взаимодействие работников, обеспечить персонал конструктивными материалами на всех технологических операциях автоматизированной обработки информации.

Методическое обеспечение автоматизированного управления состоит из методических указаний, рекомендаций и положений по внедрению, эксплуатации и оценке эффективности их функционирования. Оно включает в себя также организованную машинным способом справочную информацию в целом и отдельных его функциях, средства обучения работе, демонстрационные и рекламные примеры.

Эргономическое обеспечение автоматизированного управления представляет собой комплекс мероприятий, выполнение которых должно создавать максимально комфортные условия для использования АРМ специалистами, быстрейшего освоения технологии и качественной работы на АРМ. Комфортные условия предполагают выбор специальной мебели для размещения технической базы АРМ, организацию картотек для хранения документации и магнитных носителей.

Общеизвестно, что внедрение ПЭВМ оказывает как положительные, так и отрицательные воздействия на организацию труда исполнителя. Положительный социально-экономический эффект выражается в росте производительности труда, снижении его рутинности и др. Отрицательное же воздействие ПЭВМ на человека выражено менее заметно, но оно проявляется в нарушении функции зрения, утомляемости и т.д. Поэтому одна из важнейших задач эргономического обеспечения АРМ — уменьшить отрицательные воздействия на человека со стороны ПЭВМ.

Правовое обеспечение автоматизированного управления включает систему нормативно-правовых документов, которые должны чётко определять права и обязанности специалистов в условиях функционирования АРМ. Должен быть разработан и применен на практике комплекс документов, регламентирующих порядок хранения и защиты информации, правила ревизии данных, обеспечение юридической подлинности совершаемых на АРМ операций и т.д.

Следует иметь в виду, что АРМ специалистов управления должны эксплуатироваться работниками, прошедшими в той или иной мере обучение по основам информатики и вычислительной техники, эксплуатации АРМ. Должностные инструкции этой категории пользователей должны предусматривать выполнение вполне определенных процедур на АРМ. Должно стать правилом: конкретные АРМ специалистов сферы организационно-экономического управления необходимо создавать на основе анализа организации труда исполнителей, паспортизации и аттестации рабочих мест.

9.6. Защита информации при автоматизированном управлении

В настоящее время ПЭВМ пользователей в автоматизированных системах, как правило, включены в информационно-вычислительные сети (ИВС). В них сосредотачивается информация, исключительное право на пользование которой принадлежит оп-

ределенным лицам или группе лиц, действующих в порядке личной инициативы или в соответствии с должностными обязанностями. Такая информация должна быть защищена от всех видов постороннего вмешательства: чтение информации лицами, не имеющими права доступа к ней, и преднамеренное изменение информации. К тому же в ИВС должны приниматься меры по защите вычислительных ресурсов сети от их несанкционированного использования, т.е. должен быть исключен доступ к сети лиц, не имеющих на это права.

Физическая система защиты системы и данных может осуществляться только в отношении рабочих ЭВМ и узлов связи и оказывается невозможной для средств передачи, имеющих большую протяженность. По этой причине в ИВС должны использоваться средства, исключаящие несанкционированный доступ к данным и обеспечивающие их секретность.

Неизбежным средством борьбы с этой опасностью стали постоянно увеличивающиеся расходы на защиту информации. Например, по оценке немецких экспертов лишь в 1987 году в промышленности и учебных заведениях Западной Европы потрачено 1,7 млрд марок на обеспечение безопасности компьютеров.

Исследования практики функционирования систем обработки информации и вычислительных систем доказали, что существует достаточно много возможных направлений утечки информации и путей несанкционированного доступа в системах и сетях. В их числе:

- чтение остаточной информации в памяти системы после выполнения санкционированных запросов;
- копирование носителей и файлов информации с преодолением мер защиты;
- маскировка под зарегистрированного пользователя;
- маскировка под запрос системы;
- использование программных ловушек;
- использование недостатков операционной системы;
- незаконное подключение к аппаратуре и линиям связи;
- злоумышленный вывод из строя механизмов защиты;
- внедрение и использование компьютерных вирусов.

Обеспечение безопасности информации в ИВС и в автономно работающих ПЭВМ достигается комплексом организационных, организационно-технических, технических и программных мер.

К организационным мерам защиты информации относятся:

- ограничение доступа в помещения, в которых происходит подготовка и обработка информации;

- доступ к обработке и передаче конфиденциальной информации только проверенных должностных лиц;
- хранение магнитных носителей и регистрационных журналов в закрытых для доступа посторонних лиц сейфах;
- исключение просмотра посторонними лицами содержания обрабатываемых материалов через дисплей, принтер и т.д.;
- использование криптографических кодов при передаче информации по каналам связи ценной информации;
- уничтожение красящих лент, бумаги и иных материалов, содержащих фрагменты ценной информации.

Организационно-технические меры защиты информации включают:

- осуществление питания оборудования, обрабатывающего ценную информацию от независимого источника или через специальные фильтры;
- установку на дверях помещений кодовых замков;
- использование для отображения информации при вводе-выводе жидкокристаллических или плазменных дисплеев, а для получения твердых копий — струйных принтеров и термопринтеров, поскольку дисплей дает такое высокочастотное электромагнитное излучение, что изображение с его экрана можно принимать на расстоянии нескольких сотен километров;
- уничтожение информации, хранящейся в ОЗУ и на «винчестере» при списании или отправке ПЭВМ в ремонт;
- установка клавиатуры и принтеров на мягкие прокладки с целью снижения возможности снятия информации акустическим способом;
- ограничение электромагнитного излучения путем экранирования помещений, где происходит обработка информации, листами из металла или специальной пластмассы.

Технические средства защиты информации — это системы охраны территорий и помещений с помощью экранирования машинных залов и организация контрольно-пропускных систем.

Защита информации в сетях и вычислительных средствах с помощью технических средств реализуется на основе организации доступа к памяти с помощью:

- контроля доступа к различным уровням памяти компьютера;
- блокировки данных и ввода ключей;
- выделения контрольных битов для записей с целью идентификации и др.

Архитектура *программных средств защиты информации* включает:

- контроль безопасности, в том числе и контроль регистрации входа в систему, фиксацию в системном журнале, контроль действий пользователя;
- реакцию, в том числе звуковую, на нарушение системы защиты контроля доступа к ресурсам сети;
- контроль мандатов доступа;
- формальный контроль защищенности операционных систем (базовой операционной и сетевой);
- контроль алгоритмов защиты;
- проверку и подтверждение правильности функционирования технического и программного обеспечения.

Для надежной защиты информации и выявления случаев неправомерных действий проводится регистрация работы системы: создаются специальные дневники и протоколы, в которых фиксируются все действия, имеющие отношение к защите информации в системе. Фиксируются время поступления заявки, ее тип, имя пользователя и терминала, с которого инициализируется заявка. При отборе событий, подлежащих регистрации, необходимо иметь в виду, что с ростом количества регистрируемых событий, затрудняется просмотр дневника и обнаружение попыток преодоления защиты. В этом случае можно применять программный анализ и фиксировать сомнительные события.

Используются также специальные программы для тестирования системы защиты. Периодически или в случайно выбранные моменты времени они проверяют работоспособность аппаратных и программных средств защиты.

К отдельной группе мер по обеспечению сохранности информации и выявлению несанкционированных запросов относятся программы обнаружения нарушений в режиме реального времени. Программы данной группы формируют специальный сигнал при регистрации действий, которые могут привести к неправомерным действиям по отношению к защищаемой информации. Сигнал может содержать информацию о характере нарушения, месте его возникновения и другие характеристики. Кроме того, программы могут запретить доступ к защищаемой информации или симулировать такой режим работы (например, моментальная загрузка устройств ввода-вывода), который позволит выявить нарушителя и задержать его соответствующей службой.

Один из распространенных способов защиты — явное указание секретности выводимой информации. В системах, поддерживающих несколько уровней секретности, вывод на экран терминала или печатающего устройства любой единицы информации (например,

файла, записи или таблицы) сопровождается специальным грифом с указанием уровня секретности. Это требование реализуется с помощью соответствующих программных средств.

В отдельную группу выделены средства защиты от несанкционированного использования программного обеспечения. Они приобретают особое значение вследствие широкого распространения персональных компьютеров. Исследования, проведенные зарубежными исследователями, свидетельствуют, что на одну проданную копию оригинальной программы приходится минимум одна нелегальная. А для особо популярных программ это соотношение достигает 1:7.

Особое внимание уделяется законодательным средствам, регулирующим использование программных продуктов. В соответствии с Законом Российской Федерации об информации, информатизации и защите информации от 25 января 1995 года предусматриваются санкции к физическим и юридическим лицам за нелегальное приобретение и использование программных продуктов.

Большую опасность представляют компьютерные вирусы.

Компьютерный вирус — это специально написанная небольшая по размерам программа, которая может приписывать себя к другим программам (т.е. «заражать» их), а также выполнять различные нежелательные действия. Программа, внутри которой находится компьютерный вирус, называется зараженной. Когда такая программа начинает работу, то сначала управление получает вирус, который находит и заражает другие программы, а также выполняет ряд вредных действий, в частности «засоряет» активную память, портит файлы и т. д.

Для маскировки вируса его действия по заражению других программ и нанесению вреда могут выполняться не всегда, а при выполнении каких-либо условий. После того, как вирус выполнит нужные ему действия, он передает управление той программе, в которой он находится, и она работает как обычно, т. е. внешне работа зараженной программы какое-то время не отличается от работы незараженной программы.

Действия вируса могут выполняться достаточно быстро и без выдачи сообщений, поэтому пользователь часто и не замечает, что компьютер работает несколько странно. Однако по прошествии некоторого времени, на компьютере может происходить следующее:

- некоторые программы перестают работать или работают неправильно;
- на экран выводятся посторонние сообщения, символы, рисунки и т. д.;

- работа на компьютере существенно замедляется;
- некоторые файлы оказываются испорченными и т.д.

Многие вирусы устроены так, что при запуске зараженной программы они остаются постоянно (точнее, до перезагрузки ОС) в памяти компьютера и время от времени заражают программы. Кроме того, зараженные программы с данного компьютера могут быть перенесены с помощью дискет или по локальной сети на другие компьютеры.

Если не принимать мер по защите от вируса, то последствия заражения вирусом компьютера могут быть серьезными. В число средств и методов защиты от компьютерных вирусов входят:

- общие средства защиты информации, которые полезны так же, как и страховка от физической порчи машинных дисков, неправильно работающих программ или ошибочных действий пользователя;
- профилактические меры, позволяющие уменьшить вероятность заражения вирусом;
- специализированные программы для защиты от вирусов.

Комплексное решение вопросов безопасности ИВС принято именовать *архитектурой безопасности*, где выделяются угрозы безопасности, службы безопасности и механизмы обеспечения безопасности.

Под *угрозой безопасности* понимается событие или действие, которое может привести к разрушению, искажению или несанкционированному использованию ресурсов сети, включая хранимую и обрабатываемую информацию, а также программные и аппаратные средства.

Угрозы распределяются на *случайные* (непреднамеренные) и *умышленные*. Источником первых могут быть ошибки в ПО, неправильные (ошибочные) действия пользователей, выход из строя аппаратных средств и др.

Умышленные угрозы преследуют цель нанесения ущерба пользователям (абонентам) вычислительной сети и подразделяются на *активные* и *пассивные*.

Пассивные угрозы не разрушают информационные ресурсы и не оказывают влияния на функционирование ИВС. Их задача — несанкционированно получить информацию. Активные угрозы преследуют цель нарушить процесс функционирования ИВС путем разрушения или радиоэлектронного подавления линий связи ИВС, вывода из строя ЭВМ или ее операционной системы, искажения баз данных и т. д. Источником активных угроз могут быть непосредственные действия людей — злоумышленников, компьютерные вирусы и т. д.

Служба безопасности вычислительной сети призвана обеспечить:

- подтверждение подлинности того, что объект, который предлагает себя в качестве отправителя информации в сети, действительно им является;
- целостность информации, выявляя искажения, вставки, повторы и уничтожение данных, передаваемых в сетях, а также последующее восстановление данных;
- секретность всех данных, передаваемых по сетям;
- нейтрализацию всех попыток несанкционированного использования ресурсов ЭВМ. При этом контроль доступа может быть избирательным, т.е. распространяться только на некоторые виды доступа к ресурсам, например, на обновление информации в базе данных, либо полным;
- получателя информации доказательствами, что информация получена от данного отправителя, несмотря на попытки отправителя отрицать факт отправления;

К механизмам обеспечения безопасности относятся:

- идентификация пользователей;
- шифрование данных;
- электронная подпись;
- управление маршрутизацией и др.

Идентификация пользователей позволяет устанавливать конкретного пользователя, работающего за терминалом и принимающего или отправляющего сообщения. Право доступа к определенным вычислительным и информационным ресурсам, программам и наборам данных, а также ВС в целом предоставляется ограниченному контингенту лиц, и система должна распознавать пользователей, работающих за терминалами. Идентификация пользователей чаще всего производится с помощью паролей.

Пароль — это совокупность символов, известных подключенному к сети абоненту, вводится в начале сеанса взаимодействия с сетью, а иногда и в конце сеанса (в особо ответственных случаях пароль выхода из сети может отличаться от входного). Система может предусматривать ввод пароля для подтверждения правомочия пользователя через определенные интервалы времени.

Для защиты средств идентификации пользователей от неправомерного использования, пароли передаются и сравниваются в зашифрованном виде, а таблицы паролей хранятся в зашифрованном виде, что исключает возможность прочтения паролей без знания ключа.

Для идентификации пользователей могут применять и физические методы: например, карточка с магнитным покрытием, на котором записывается персональный идентификатор пользователя или карточка со встроенным чипом.

Наиболее надежным, хотя и наиболее сложным является способ идентификации пользователя на основе анализа его индивидуальных параметров: отпечатков пальцев, рисунков линий руки, радужной оболочки глаз и др.

Шифрование данных — это обеспечение секретности методами *криптографии*, т. е. методами преобразования данных из общепринятой формы в *кодированную* (шифрование) и обратного преобразования (дешифрования) на основе правил, известных только взаимодействующим абонентам сети. Криптография применяется для защиты передаваемых данных, а также информации, хранимой в базах данных, на магнитных и оптических дисках и т. д.

К криптографическим средствам предъявляются требования сохранения секретности, даже когда известна сущность алгоритмов шифрования — дешифрования. Секретность обеспечивается введением в алгоритмы специальных ключей (кодов). Зашифрованный текст превращается в исходный только в том случае, когда в процессе шифрования и дешифрования используется один и тот же ключ. Область значений ключа выбирается столь большой, что практически исключается возможность его определения путем простого перебора.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Объясните необходимость обеспечивающих подсистем автоматизированного управления.
2. Перечислите виды обеспечивающих подсистем автоматизированного управления.
3. Дайте краткую характеристику математическому обеспечению автоматизированного управления.
4. Какие Вы знаете экономико-математические модели управления производством?
5. Охарактеризуйте модели управления запасами.
6. Какими математическими моделями описывается процесс планирования?
7. Как Вы себе представляете информационное обеспечение автоматизированного управления?
8. Каковы особенности экономической информации?
9. Расскажите о процедурах обработки информации.
10. В чем заключается организация информационных процессов в СУ?
11. Чем отличается новая информационная технология управленческой деятельности от традиционной?
12. Объясните понятия «база данных» и «СУБД».
13. Какие Вы знаете модели данных?
14. Что такое программное обеспечение автоматизированного управления?

15. Как подразделяется программное обеспечение автоматизированного управления?
16. Дайте характеристику общему и специальному ПО автоматизированного управления.
17. Расскажите о классификации пакетов прикладных программ.
18. Каково содержание проблемно-ориентированных ППП?
19. В чем суть интегрированных ППП?
20. Как организуется процесс разработки программных средств?
21. Что такое алгоритмизация решения задач на ЭВМ?
22. Каким образом осуществляется составление, тестирование и отладка программ?
23. В чем заключается эксплуатация ПО?
24. Какова роль технологического обеспечения автоматизированного управления?
25. Расскажите о сетевом режиме автоматизированной обработки информации.
26. Каковы особенности технологии использования автоматизированных рабочих мест?
27. Объясните технологию использования ППП.
28. В чем состоит технология обработки текстовой информации при автоматизированном управлении?
29. Раскройте технологию обработки табличной информации в автоматизированном управлении.
30. Какова технология использования автоматизированных банков данных?
31. Какие Вы знаете интегрированные технологии в распределенных системах обработки данных?
32. Приведите области применения экспертных систем.
33. В чем отличие технологии использования экспертных систем от традиционных подходов?
34. Что входит в лингвистическое обеспечение автоматизированного управления?
35. Каким образом строится организационно-методическое обеспечение автоматизированного управления?
36. Как Вы понимаете правовое обеспечение автоматизированного управления?
37. Раскройте особенности взаимодействия пользователя с ЭВМ.
38. Каковы типы диалога и формы его реализации на ЭВМ?
39. В чем состоит организация диалоговой технологии обработки данных на ПЭВМ?
40. Объясните суть методов защиты информации в автоматизированных системах?
41. Перечислите физические и технические меры защиты информации в автоматизированных системах.
42. Как строится архитектура программных средств защиты информации?
43. Какова роль службы защиты информации в ИВС?

Заключение

Этапы развития теории автоматизированного управления. Автоматизированное управление сложными динамическими объектами интенсивно развивается с середины XX в.

В процессе своего развития теория и системы автоматизированного управления прошли ряд этапов.

На первом этапе своего становления теория автоматизированного управления (1960—1970 гг.) обосновывала принципы построения систем «человек—машина». В результате исследований в кибернетической системе «черный ящик» (рис. 3.1.) про-

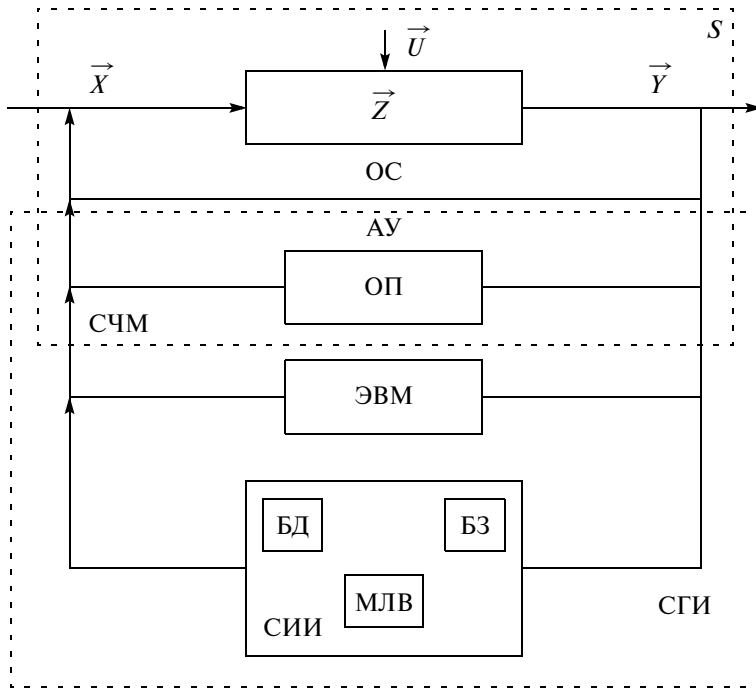


Рис 3.1. Схема кибернетического управления сложной системой S :

\vec{Z} — вектор внутренних состояний системы; \vec{X} — вектор входных воздействий; \vec{U} — вектор управляющих воздействий; \vec{Y} — вектор выходных эффектов; ОС — обратная связь; ОП — оперативный персонал; СЧМ — система «человек — машина»; БД — база данных; БЗ — база знаний; МЛВ — машина логического вывода (решатель); СИИ — система искусственного интеллекта; СГИ — система гибридного интеллекта

изошло четкое распределение функций между автоматическим и автоматизированным способами управления. Началось развитие теоретических основ эргономики.

Стремительное внедрение в практику результатов научно-технической революции последней трети XX в. привело к появлению таких сложных высокоответственных и технологически опасных комплексов, как сверхзвуковые и межконтинентальные самолеты, ядерные энергетические установки, космические станции, противовоздушные и противоракетные системы, атомные подводные лодки, территориальные и региональные промышленные и административные системы. При управлении такими динамическими системами на человека-оператора может поступать одновременно сотни, а иногда и тысячи сигналов, справиться с которыми оперативному персоналу очень трудно. Поэтому в контур управления системой была включена ЭВМ, на которую возлагаются функции сбора, обработки, хранения и представления в удобном виде оператору информации о функционировании системы. Получили большое развитие системы отображения информации (1970—1980 гг.): это второй этап развития теории автоматизированного управления.

Однако даже включение в контур управления сложными динамическими комплексами мощных ЭВМ не привело к кардинальному решению проблем управления. Поэтому с начала 80-х годов прошлого века в теории автоматизированного управления с использованием ЭВМ четвертого поколения стали разрабатываться теоретические основы применения систем поддержки принятия решений оператора. На третьем этапе (1980—1990 гг.) получили развитие разнообразные системы искусственного интеллекта в различных областях науки, техники, медицины, энергетики, транспорта и др.

Современный, четвертый, этап развития теории автоматизированного управления характерен широким использованием ЭВМ пятого поколения, информационных и вычислительных сетей, систем «советчик оператора», систем искусственного и гибридного интеллекта.

Направления дальнейших исследований в области автоматизированного управления.

Теоретической основой автоматизированного управления являются идеи и методы кибернетики. За последние пятьдесят лет своего становления и развития кибернетика прошла путь, глубоко отличный от путей обычных, классических наук. Ее идеи, формальный аппарат и технические решения вызревали и разви-

вались в рамках разных научных дисциплин, в каждой по-особому, которые в конечном итоге привели к новым концепциям и методологии. Симбиоз управления и информации, как и весь связанный с ними арсенал понятий и методов, был поднят до уровня общенаучных представлений. Вместе с другими научными дисциплинами, такими, как теория автоматов, теория графов, теория надежности, теория массового обслуживания и др., теория автоматизированного управления достигла высокого уровня своего развития.

Вместе с тем в теории автоматизированного управления существует и ряд проблем, требующих своего разрешения. К ним, в частности, относятся:

1. «Извечная» проблема размерности сложных систем. Структура и функции систем постоянно усложняются. Это порождает ряд проблем построения АСУ, иерархии их элементов и подсистем, взаимодействия между уровнями управления и внешней средой, оптимального целеполагания.

2. Усложнение объектов управления и АСУ постоянно требует развития новых методов анализа и синтеза их надежности, безопасности, живучести и эффективности.

3. С созданием автоматов, роботов, систем искусственного и гибридного интеллекта появилась проблема о предельных возможностях таких систем и о сравнении возможностей переработки информации и принятия решений машиной и человеком, а также о характере взаимоотношения человека и машины.

Термин «гибридный интеллект» был впервые введен В.Ф. Вендой еще в 1975 году, и после этого Валерием Федоровичем были разработаны основы теории гибридного интеллекта. 80-е годы XX века были годами бурного развития АСУ и систем искусственного интеллекта. Существовало даже мнение, что искусственный интеллект практически вытеснит человека из контура управления атомными и тепловыми электростанциями и другими энергетическими установками, транспортными объектами, технологическими процессами. Проектирование новой техники, сочинение музыкальных произведений и других видов творческой деятельности человека будто бы можно целиком возложить на ЭВМ. Вот эта концепция искусственного интеллекта естественно нуждалась в своем антипode — концепции гибридного интеллекта. Причем эти два типа интеллектуальных систем диалектически взаимосвязаны и переходят один в другой.

Главными принципами гибридного интеллекта являются:

1) *эволюционность* — как соответствие глубинным интересам и структурам человека, техники и биосферы;

2) *демократичность* — как выражение равенства, общих интересов и ответственности всех элементов иерархии системы управления;

3) *гибкость иерархии СУ*, при которой каждый ее элемент является лидером на тот период, когда он наиболее компетентен, полезен и дальновиден;

4) *взаимная адаптация* всех участников и компонентов СУ;

5) *трансформация стратегий* — как путь к творческому порождению новых стратегий (теорий, методов, прогнозов);

6) *интенсивность процессов коммуникации*.

Обострение проблемы безопасности сложных энергетических, транспортных и технологических систем, угрозы экологической катастрофы и будущее выживание человечества прямо зависит от роста его интеллектуальных возможностей. Аварии и катастрофы в промышленности и на транспорте подтвердили остроту проблемы надежности человека-оператора как центрального звена в СУ. Они выявили неуклонно возрастающую сложность и ответственность интеллектуальных задач в ходе научных исследований, проектирования, производства и эксплуатации систем. В настоящее время возрастает доля таких задач, решение которых непосильно одному человеку. Поэтому к их решению должны привлекаться коллективы разных специалистов, действующие как единый гибридный интеллект.

Особенно своевременна постановка проблемы создания систем гибридного интеллекта в связи с бурным развитием информатики, вычислительной техники, информационных технологий и систем связи. Действительно, достижения в области экспертных систем, систем поддержки принятия решений, телеконференций, адаптивной информационной техники уже теперь позволяют реализовывать системы гибридного интеллекта не только для решения научных проблем, проектирования сложнейших видов техники и сооружений, составления и оперативного согласования хозяйственных планов, но и для экстренного решения аварийных задач на АЭС и других объектах с привлечением разработчиков, опытных операторов, централизованных банков данных. Это чрезвычайно важно для решения следующей проблемы.

4. Проблема разработки теории безопасности системы «человек — машина — среда». Эта проблема тесно связана с реше-

нием задач математического, программного и инструментального обеспечения интеллектуальных информационных систем. В круг нерешенных до конца задач этой проблемы входят также разработка теории, принципов и методов оптимального отображения информации, а также разработка теории и методов организации труда метаоператора, координирующего процесс информационного взаимодействия участников системы гибридного интеллекта.

5. Проблема моделирования. Математическое и машинное моделирование стали эффективным инструментом моделирования сложных СУ. Математические модели вследствие их гибкости и адекватности реальным процессам и сравнительно невысокой стоимости реализации на базе современных ЭВМ позволяют решать многие задачи проектирования СУ. Особенно актуальна задача моделирования на ранних этапах проектирования АСУ. Однако в проблеме моделирования таких сложных систем, как системы «человек — машина — среда», системы гибридного интеллекта, системы поддержки и принятия решения и других информационных интеллектуальных систем, еще остается большое число нерешенных задач.

Список литературы

Основная

1. Антонов А.В. Системный анализ. Методология. Построение модели: Учеб. пособие. — Обнинск: ИАТЭ, 2001. — 272 с.
2. Богданов А.А. Всеобщая организационная наука (тектология): В 3 т. — М., 1905—1924.
3. Венда В.Ф. Системы гибридного интеллекта: эволюция, психология, информатика. — М.: Машиностроение, 1990. — 448 с.
4. Волкова В.Н. Основы теории систем и системного анализа/ В.Н. Волкова, А.А. Денисов. — СПб.: СПбГТУ, 1997. — 510 с.
5. Волкова В.Н. Методы формализованного представления систем/ В.Н. Волкова, А.А. Денисов, Ф.Е. Темников. — СПб.: СПбГТУ, 1993. — 108 с.
6. Гаскаров Д.В. Интеллектуальные информационные системы. — М.: Высш. шк., 2003. — 431 с.
7. Глушков В.М. Введение в АСУ. — Киев: Техника, 1974.
8. Дегтярев Ю.И. Системный анализ и исследования операций. — М.: Высш. шк., 1996. — 335 с.
9. Корячко В.П. Теоретические основы САПР: Учеб. для вузов/ В.П. Корячко, В.М. Курейчик, И.П. Норенков. — М.: Энергоатомиздат, 1987. — 400 с.
10. Мамиконов А.Г. Основы построения АСУ: Учеб. для вузов. — М.: Высш. шк., 1981. — 248 с.
11. Меньков А.В. Теоретические основы автоматизированного управления: Учеб. пособие. — М.: МГУП, 2002. — 176 с.
12. Острейковский В.А. Автоматизированные информационные системы в экономике: Учеб. пособие. — Сургут: СурГУ, 2000. — 165 с.
13. Острейковский В.А. Современные информационные технологии экономистам: Учеб. пособие. Ч. 1. Введение в автоматизированные информационные технологии. — Сургут: СурГУ, 2000. — 72 с.
14. Острейковский В.А. Теория систем: Учеб. для вузов. — М.: Высш. шк., 1997. — 240 с.
15. Острейковский В.А. Современные информационные технологии экономистам: Учеб. пособие. Ч.2. Технологическое обеспечение экономических информационных систем/ В.А. Острейковский, В.С. Микшина. — Сургут: СурГУ, 2002. — 123 с.

16. *Перегудов Ф.И.* Основы системного анализа/ *Ф.И. Перегудов, Ф.П. Тарасенко.* — Томск: Изд-во НТЛ, 1997.— 396 с.
17. *Советов Б.Я.* Моделирование систем: Учеб. для вузов/ *Б.Я. Советов, С.А. Яковлев.* — М.: Высш. шк., 1998. — 319 с.

Дополнительная

1. *Автоматизированные информационные технологии в экономике/* Под ред. проф. Г.А. Титоренко. — М.: Компьютер, ЮНИТИ, 1998.— 400 с.
2. *Автоматизированные информационные технологии в банковской деятельности /* Под ред. проф. Г.А. Титоренко. — М.: Финстатинформ, 1997.
3. *АСУ на промышленном предприятии: Методы создания: Справочник /* С.Б. Михалев, Р.С. Седегов, А.С. Гринберг и др. — М.: Энергоатомиздат, 1989. — 400 с.
4. *Искусственный интеллект: Справочник: В 3 т.* — М.: Радио и связь, 1990.
5. *Комплекс общеотраслевых руководящих методических материалов по созданию АСУ и САПР.* — М.: Статистика, 1980.
6. *Малин А.С.* Исследование систем управления: Учеб. для вузов/ *А.С. Малин, В.И. Мухин.* — М.: ГУ ВШЭ, 2002. — 400 с.
7. *Месарович М.* Теория иерархических многоуровневых систем/ *М. Месарович, Д. Мако, И. Такахага.* — М.: Мир, 1973. — 344 с.
8. *Моисеев Н.Н.* Математические задачи системного анализа. — М.: Наука, 1981. — 488 с.
9. *Острейковский В.А.* Информатика: Учеб. для вузов. — М.: Высш. шк., 1999. — 511 с.
10. *Острейковский В.А.* Математические модели макро- и микроэкономики: Учеб. пособие. — Сургут: СурГУ, 2000. — 72 с.
11. *Перегудов Ф.И.* Введение в системный анализ: Учеб. для вузов/ *Ф.И. Перегудов, Ф.П. Тарасенко.* — М.: Высш. шк., 1989. — 376 с.
12. *Пригожин И.* Порядок из хаоса/ *И. Пригожин, И. Стенгерс.* — М.: Прогресс, 1986.
13. *Система автоматизированного проектирования в радиоэлектронике: Справочник/* *Е.В. Авдеев, А.Т. Еремин, И.П. Норенков, М.И. Песков;* Под ред. *И.П. Норенкова.* — М.: Радио и связь, 1986.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению курсовых работ
по дисциплине
«Теоретические основы
автоматизированного управления»

Введение

Курсовые работы выполняются в том же семестре, в котором читается дисциплина. Их цель — закрепление и практическое усвоение наиболее сложных, по мнению профилирующей кафедры, разделов дисциплины «Теоретические основы автоматизированного управления».

Курсовые работы представляют собой набор тематически законченных задач. Количество задач в наборе, их содержание, степень сложности определяет профилирующая кафедра. Ниже приводятся примеры постановок задач и их решений.

Каждая задача оформляется индивидуально на листах формата А4 с помощью ПЭВМ. Она состоит из титульного листа и информационной части.

Титульный лист содержит следующую обязательную информацию: наименование вуза, наименование кафедры, номер и наименование задачи, наименование группы, фамилия и инициалы студента, защищающего задачу, фамилия и инициалы преподавателя, руководившего работой, название города, год.

Информационная часть каждой из задач состоит из формулировки задачи и ее решения.

Формулировка задачи содержит текстовой, цифровой и графический материал. Текстовая часть в ряде случаев может быть общей для группы студентов, а цифровой и графический материал выдается индивидуально.

Для получения зачета по курсу каждый студент помимо теоретической части, излагаемой на лекциях, обязан индивидуально защитить свой вариант курсовых работ.

Индивидуальная защита курсовых работ является необходимым условием допуска к сдаче экзамена по курсу «Теоретические основы автоматизированного управления».

На с. 612 приведен пример выполнения титульного листа курсовой работы.

Наименование вуза

Наименование кафедры

КУРСОВАЯ РАБОТА

по дисциплине
«Теоретические основы
автоматизированного управления»

Задача №... . «Наименование задачи»

Руководитель:

Исполнитель:

Студент группы

_____ 2005 г.

_____ 2005 г.

Город -----

Информационная часть курсовой работы

Задача №1. Формализация представления структуры

Формулировка задачи

Разработать формализованное представление структуры, изображенной на рис. 1.

Структуру отобразить рассмотренными способами. Выделить (можно не все, а только в качестве примера) цепи, пути, циклы, контуры, степени вершин, полустепени исходов и заходов.

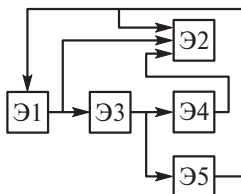


Рис. 1. Структура системы

Решение

См. 4.3.

Задача № 2. Введение порядковой функции

Формулировка задачи

В результате обследования некоторой организационной системы был получен неупорядоченный граф информационно-логической взаимосвязи между задачами, решаемыми в этой системе. Необходимо определить:

1. В какой последовательности следует решать указанные задачи.
2. Решение каких задач можно начинать одновременно.
3. На протяжении скольких тактов следует хранить результаты решений.
4. Убедиться в том, что матрица смежности упорядоченного графа оказалась треугольной.

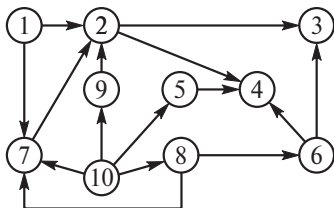


Рис. 1. *Граф информационно-логической взаимосвязи между задачами*

Решение

См. 4.3.2.

Задача № 3. Введение числовой функции на графе

Формулировка задачи

В задаче календарного планирования возникла необходимость определения максимального пути из вершины a_1 в вершину a_7 для графа, представленного на рис.1.

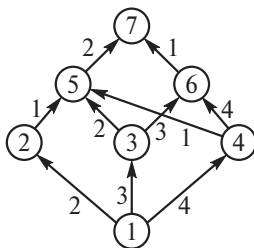


Рис. 1. *К нахождению пути максимальной длины*

Решение

См. 4.3.3.

Задача № 4. Топологическая декомпозиция структур

Формулировка задачи

В распределенной АСУ пункты обработки информации обмениваются данными так, как это изображено с помощью графа, представленного на рис. 1. Возникла необходимость в сокращении числа этих пунктов, исходя только из структурных свойств анализируемой системы. (Объединение производится без учета

производительности, надежности и т. п., учитывая только структурные свойства схемы.)

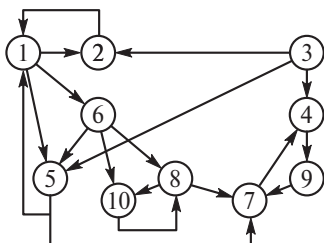


Рис. 1. Граф структуры распределенной АСУ

Решение

См. 4.4.

Задача № 5. Формализованный анализ информационных потоков

Формулировка задачи

Схеме движения оперативной отчетности в подсистеме оперативного управления производством соответствует информационный граф, представленный на рис.1. Необходимо формально выявить все свойства данного информационного графа.

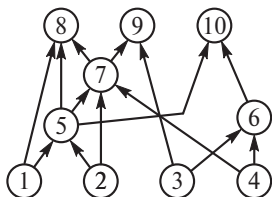


Рис. 1. Исходный граф

Решение

См. 4.5.

Задача № 6. Расчет топологических характеристик

Формулировка задачи

Для структуры, представленной на рис. 1, вычислить все ее структурно-топологические характеристики. По полученным результатам охарактеризовать структуру.

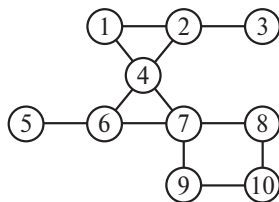


Рис. 1. Исходный граф

Решение

См. 4.6.

Задача № 7. Разработка модели функционирования

Формулировка задачи

Необходимо разработать модель функционирования простейшей организационной системы (например, продление срока действия карты метрополитена). Система должна содержать два или более операторов системы и некоторый документооборот. Разработка должна содержать следующие обязательные разделы: словесное описание процесса функционирования системы; организационная схема системы; описание документов системы; алгоритм обработки информации; принципиальная схема документооборота; оператор сопряжения; оператор преобразования; моделирующий алгоритм.

Решение

См. 4.7. и 4.8.

Задача № 8. Многокритериальная задача принятия решения

Формулировка задачи

Для отдела ЦКБ необходимо устройство для вывода на печать конструкторских чертежей (плоттер). Имеются плоттеры трех моделей. Каждая модель характеризуется тремя локальными критериями: максимально возможный формат отпечатанного чертежа F (мм), разрешение чертежа R (dpi) и объем буфера V (КБайт).

Конкретные значения указанных локальных критериев для каждого из вариантов представлены в таблице.

Требуется, используя известные схемы компромисса, определить лучший вариант плоттера:

- а) без учета приоритета локальных критериев;
- б) с учетом приоритета локальных критериев.

Решение

См. 6.8.

Задача № 9. Первая частная задача синтеза оптимальной структуры

Формулировка задачи

Автоматизированная система включает в себя $j, j = \overline{1, m}$, пунктов обработки информации. Она предназначена для решения $i, i = \overline{1, n}$, функциональных задач. Каждая из этих задач может быть решена в любом из пунктов обработки информации. Известна матрица $\|a_{ij}\|$, в которой a_{ij} — время решения i -той задачи в j -том пункте. Требуется так распределить задачи между пунктами обработки, чтобы обеспечить минимум времени решения всех задач при соблюдении ограничений на затраты каждого из пунктов. Затраты задаются вектором $|b_j|, j = \overline{1, m}$, где b_j — предельно допустимые затраты j -того пункта.

Решение

См. 5.2.3. и 5.3.

Задача № 10. Вторая частная задача синтеза оптимальной структуры

Формулировка задачи

Решить задачу № 9 в условиях, когда вместо ограничений на затраты каждого из пунктов заданы ограничения на общие затраты.

Решение

См. 5.2.4. и 5.3.

ПРАВИЛА ОФОРМЛЕНИЯ СХЕМ АЛГОРИТМОВ, ПРОГРАММ, ДАННЫХ И СИСТЕМ

Составлены на основе требований ГОСТ 19.701-90 (ИСО 5807-85) Единая система программной документации СХЕМЫ АЛГОРИТМОВ, ПРОГРАММ, ДАННЫХ И СИСТЕМ Unified system for program documentation. Data, program and system flowcharts, program network charts and system resources charts. Documentation symbols and conventions for flowcharting.

Настоящие правила распространяются на условные обозначения (символы) в схемах алгоритмов, программ, данных и систем и устанавливают требования и правила выполнения этих схем, используемых для отображения различных видов задач обработки данных и средств их решения.

Изложенные ниже правила составлены на основе требований ГОСТ 19.701-90.

1. Общие положения

1.1. Схемы алгоритмов, программ, данных и систем (далее — схемы) состоят из имеющих заданное значение символов, краткого пояснительного текста и соединяющих линий.

1.2. Схемы могут использоваться на различных уровнях детализации, причем число уровней зависит от размеров и сложности задачи обработки данных. Уровень детализации должен быть таким, чтобы различные части и взаимосвязь между ними были понятны в целом.

1.3. В настоящих правилах определены символы, предназначенные для использования в документации по обработке данных, и приведено руководство по условным обозначениям для применения их в:

- 1) схемах данных;
- 2) схемах программ;
- 3) схемах работы системы;
- 4) схемах взаимодействия программ;
- 5) схемах ресурсов системы.

1.4. В правилах используются следующие понятия:

1) основной символ — символ, используемый в тех случаях, когда точный тип (вид) процесса или носителя данных не известен или отсутствует необходимость в описании фактического носителя данных;

2) специфический символ – символ, используемый в тех случаях, когда известен точный тип (вид) процесса или носителя данных или когда необходимо описать фактический носитель данных;

3) схема – графическое представление определения, анализа или метода решения задачи, в котором используются символы для отображения операций, данных, потока, оборудования и т. д.

2. Описание схем

2.1. Схема данных.

2.1.1. Схемы данных отображают путь данных при решении задач и определяют этапы обработки, а также различные применяемые носители данных.

2.1.2. Схема данных состоит из:

1) символов данных (символы данных могут также указывать вид носителя данных);

2) символов процесса, который следует выполнить над данными (символы процесса могут также указывать функции, выполняемые вычислительной машиной);

3) символов линий, указывающих потоки данных между процессами и (или) носителями данных;

4) специальных символов, используемых для облегчения написания и чтения схемы.

2.1.3. Символы данных предшествуют и следуют за символами процесса. Схема данных начинается и заканчивается символами данных (за исключением специальных символов, указанных в п. 3.4).

2.2. Схема программы.

2.2.1. Схемы программ отображают последовательность операций в программе.

2.2.2. Схема программы состоит из:

1) символов процесса, указывающих фактические операции обработки данных (включая символы, определяющие путь, которого следует придерживаться с учетом логических условий);

2) линейных символов, указывающих поток управления;

3) специальных символов, используемых для облегчения написания и чтения схемы.

2.3. Схема работы системы.

2.3.1. Схемы работы системы отображают управление операциями и поток данных в системе.

2.3.2. Схема работы системы состоит из:

1) символов данных, указывающих на наличие данных (символы данных могут также указывать вид носителя данных);

2) символов процесса, указывающих операции, которые следует выполнить над данными, а также определяющих логический путь, которого следует придерживаться;

3) линейных символов, указывающих потоки данных между процессами и (или) носителями данных, а также поток управления между процессами;

4) специальных символов, используемых для облегчения написания и чтения блок-схемы.

2.4. Схема взаимодействия программ.

2.4.1. Схемы взаимодействия программ отображают путь активации программ и взаимодействий с соответствующими данными. Каждая программа в схеме взаимодействия программ показывается только один раз (в схеме работы системы программа может изображаться более чем в одном потоке управления).

2.4.2. Схема взаимодействия программ состоит из:

1) символов данных, указывающих на наличие данных;

2) символов процесса, указывающих на операции, которые следует выполнить над данными;

3) линейных символов, отображающих поток между процессами и данными, а также инициации процессов;

4) специальных символов, используемых для облегчения написания и чтения схемы.

2.5. Схема ресурсов системы.

2.5.1. Схемы ресурсов системы отображают конфигурацию блоков данных и обрабатывающих блоков, которая требуется для решения задачи или набора задач.

2.5.2. Схема ресурсов системы состоит из:

1) символов данных, отображающих входные, выходные и запоминающие устройства вычислительной машины;

2) символов процесса, отображающих процессоры (центральные процессоры, каналы и т. д.);

3) линейных символов, отображающих передачу данных между устройствами ввода-вывода и процессорами, а также передачу управления между процессорами;





4) специальных символов, используемых для облегчения написания и чтения схемы.





Примеры выполнения схем приведены в Приложении.





3. Условные обозначения символов и их применяемость в схемах




В табл. П 2.1 представлены условные обозначения различных символов и их применяемость в схемах данных, программ, работы системы, ресурсов системы.



Таблица П 2.1




Наименование символа	Условное обозначение символа	Описание символа и его применения	Схема применения
1	2	3	4
Данные	Символы данных Основные 	Символ отображает данные, носитель данных не определен	Данных, программ, работы системы, взаимодействия, ресурсов системы
Запоминаемые данные		Символ отображает хранящиеся данные в виде, пригодном для обработки, носитель данных не определен	Данных, работы системы, взаимодействия, ресурсов системы
Оперативное запоминающее устройство	Специфические 	Символ отображает данные, хранящиеся в оперативном запоминающем устройстве	Данных, работы системы, взаимодействия, ресурсов системы
Запоминающее устройство с последовательным доступом		Символ отображает данные, хранящиеся в запоминающем устройстве с последовательным доступом (магнитная лента, кассета с магнитной лентой, магнитофонная кассета)	Данных, работы системы, взаимодействия, ресурсов системы




1	2	3	4
<p>Запоминающее устройство с прямым доступом</p>		<p>Символ отображает данные, хранящиеся в запоминающем устройстве с прямым доступом (магнитный диск, магнитный барабан, гибкий магнитный диск)</p>	<p>Данных, работы системы, взаимодействия, ресурсов системы</p>
<p>Ручной ввод</p>		<p>Символ отображает данные, вводимые вручную во время обработки с устройств любого типа (клавиатура, переносчик, кнопки, световое перо, полоски со штриховым кодом)</p>	<p>Данных, работы системы, взаимодействия, ресурсов системы</p>
<p>Карта</p>		<p>Символ отображает данные, представленные на носителе в виде карты (перфокарты, магнитные карты, карты со считываемыми метками, карты с отрывным ярлыком, карты со сканируемыми метками)</p>	<p>Данных, работы системы, взаимодействия, ресурсов системы</p>
<p>Бумажная лента</p>		<p>Символ отображает данные, представленные на носителе в виде бумажной ленты</p>	<p>Данных, работы системы, взаимодействия, ресурсов системы</p>

Дисплей		Символ отображает данные, представленные в человеко-читаемой форме на носителе в виде отображающего устройства (экран для визуального наблюдения, индикаторы ввода информации)	Данных, работы системы, взаимодействия, ресурсов системы
Процесс	<p>Символы процесса</p> <p>Основные</p> 	Символ отображает функцию обработки данных любого вида (выполнение определенной операции или группы операций, приводящих к изменению значения, формы или размещения информации, или к опделению, по которому из нескольких направлений потока следует двигаться)	Данных, программ, работы системы, взаимодействия, ресурсов системы
Предопределенный процесс		Символ отражает предопределенный процесс, состоящий из одной или нескольких операций или шагов программы, которые определены в другом месте (в программе, модуле)	Программ, работы системы, взаимодействия
Ручная операция		Символ отображает любой процесс, выполняемый человеком.	Данных, работы системы, взаимодействия

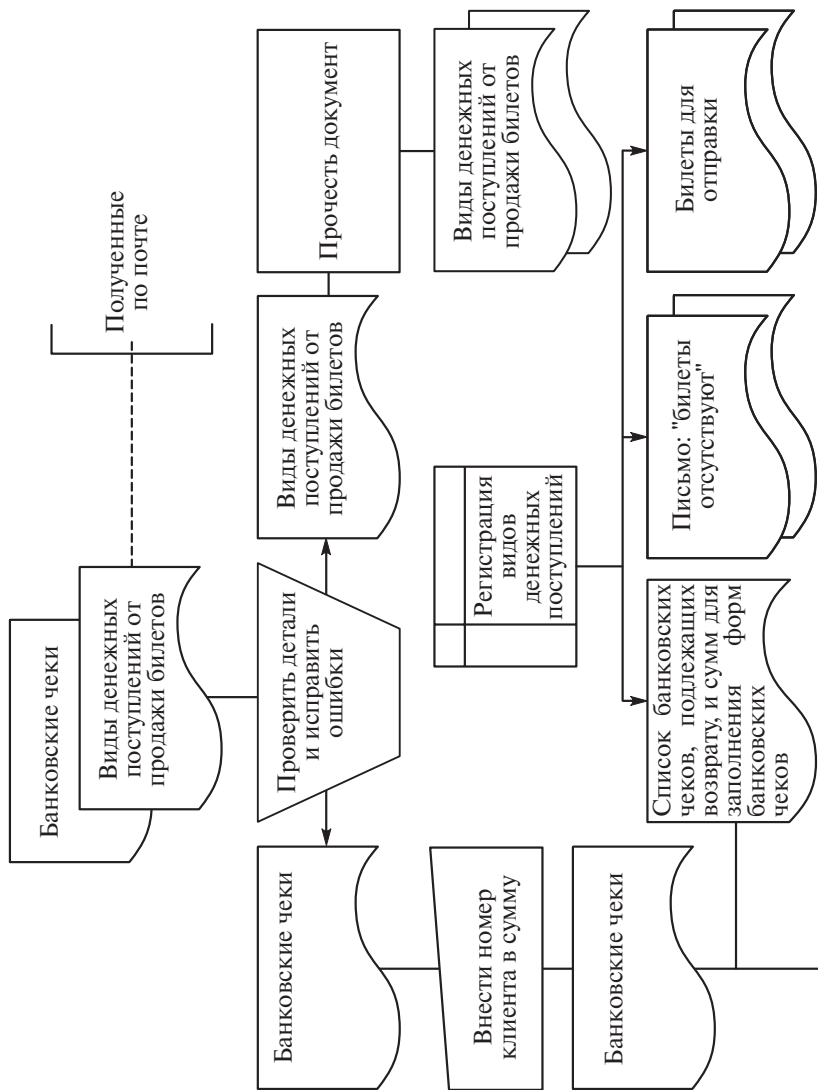
1	2	3	4
<p>Подготовка</p>		<p>Символ отображает модификацию команды или группы команд с целью воздействия на некоторую последующую функцию (установка переключателя, модификация индексного регистра или инициализация программы)</p>	<p>Данных, программ, работы системы, взаимодействия</p>
<p>Решение</p>		<p>Символ отображает решение или функцию переключаемого типа, имеющую один вход и ряд альтернативных выходов, один и только один из которых может быть активизирован после вычисления условий, определенных внутри этого символа. Соответствующие результаты вычисления могут быть записаны по соседству с линиями, отображающими эти пути</p>	<p>Программ, работы системы</p>
<p>Параллельные действия</p>		<p>Символ отображает синхронизацию двух или более параллельных операций</p>	<p>Программ, работы системы, взаимодействия</p>

Граница цикла		Символ, состоящий из двух частей, отображает начало и конец цикла. Обе части символа имеют один и тот же идентификатор. Условия для инициализации, приращения, завершения и т.д. помещаются внутри символа (в начале или в конце, в зависимости от расположения операции, проверяющей условие)	Программ, работы системы Программ, работы системы
Линия	Символы линий Основные —	Символ отображает поток данных или управления	Данных, программ, работы системы, взаимодействия, ресурсов системы
Передача управления	Специфические 	Символ отображает непосредственную передачу управления от одного процесса к другому, иногда с возможностью прямого возвращения к иницирующему процессу после того, как иницированный процесс завершит свои функции. Тип передачи управления должен быть назван внутри символа (например, запрос, вызов, событие)	Взаимодействия

1	2	3	4
Канал связи		Символ отображает передачу данных по каналу связи	Данных, работы системы, взаимодействия, ресурсов системы
Пунктирная линия		Символ отображает альтернативную связь между двумя или более символами. Кроме того, символ используют для обведения аннотированного участка. Если один из ряда альтернативных выходов используют в качестве входа в процесс либо когда выход используется в качестве входа в альтернативные процессы, эти символы соединяют пунктирными линиями	данных, программ, работы системы, взаимодействия, ресурсов системы
Соединитель	Специальные 	Символ отображает выход в часть схемы и вход из другой части этой схемы и используется для обрыва линии и продолжения ее в другом месте. Соответствующие символы-соединители должны содержать одно и то же уникальное обозначение	Данных, программ, работы системы, взаимодействия, ресурсов системы

Терминатор		Символ отображает выход во внешнюю среду и вход из внешней среды (начало или конец схемы программы, внешнее использование и источник или пункт назначения данных)	Данных, программ, работы системы
Комментарий		Символ используют для добавления описательных комментариев или поясительных записей в целях объяснения или замечаний	Данных, программ, работы системы, взаимодействия, ресурсов системы
Пропуск		Символ используют главным образом в схемах, изображающих общие решения с неизвестным числом повторений	Данных, программ, работы системы, взаимодействия, ресурсов системы

4. Примеры выполнения схем



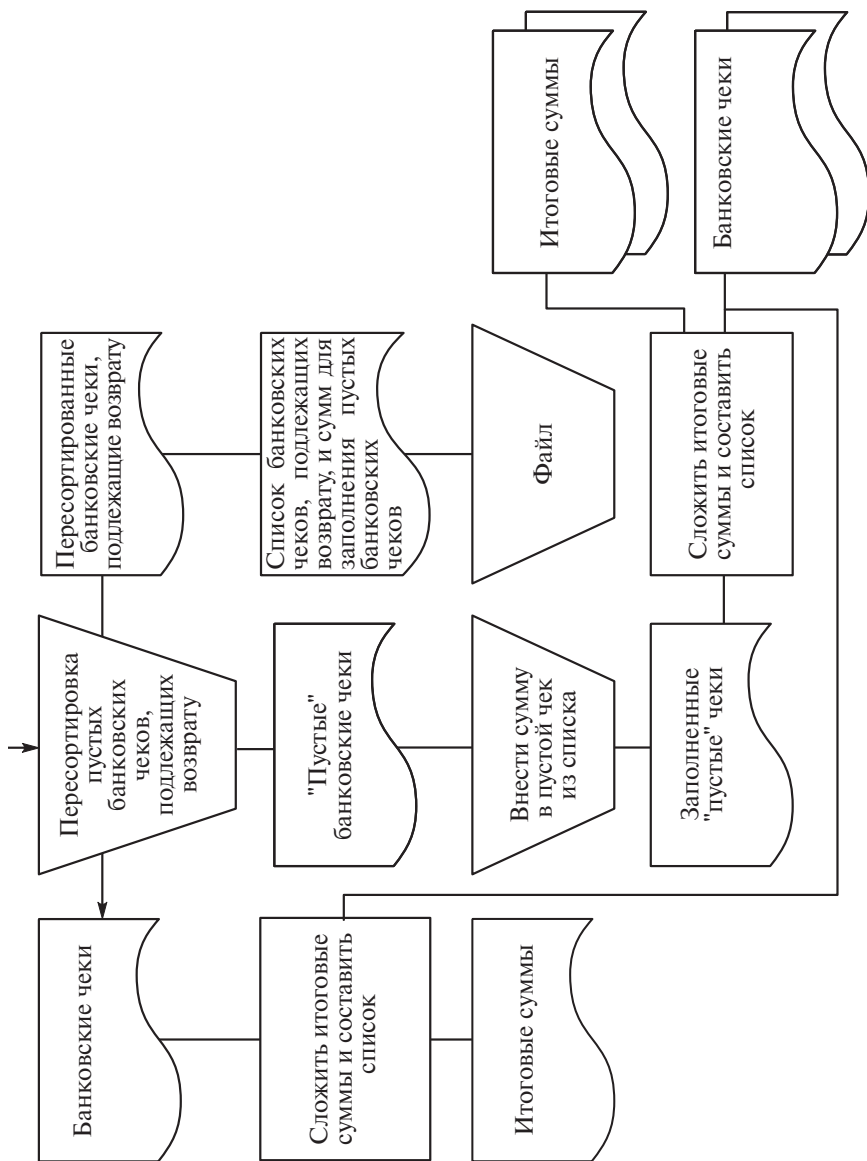


Рис. П 2.1.
Схема данных

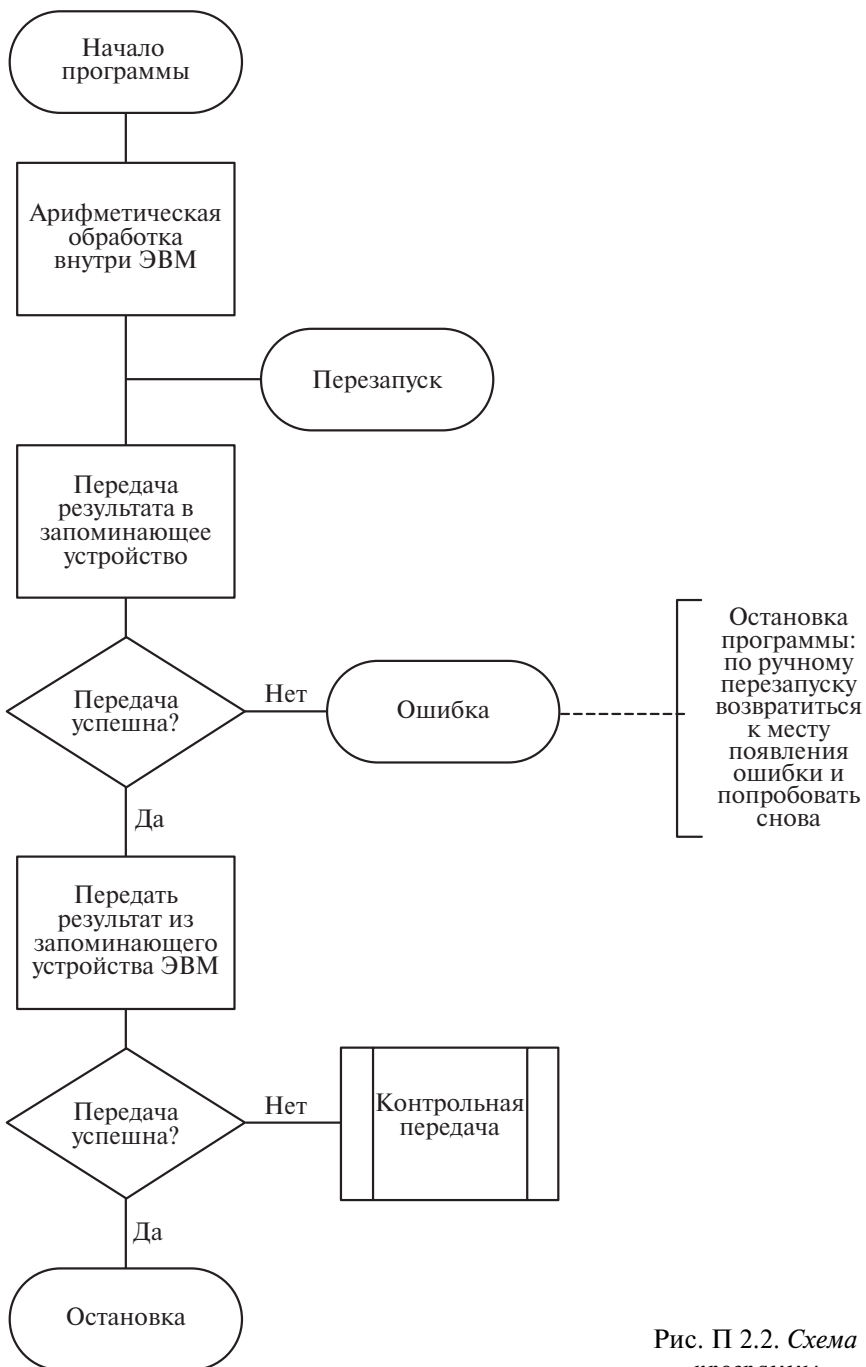


Рис. П 2.2. Схема программы

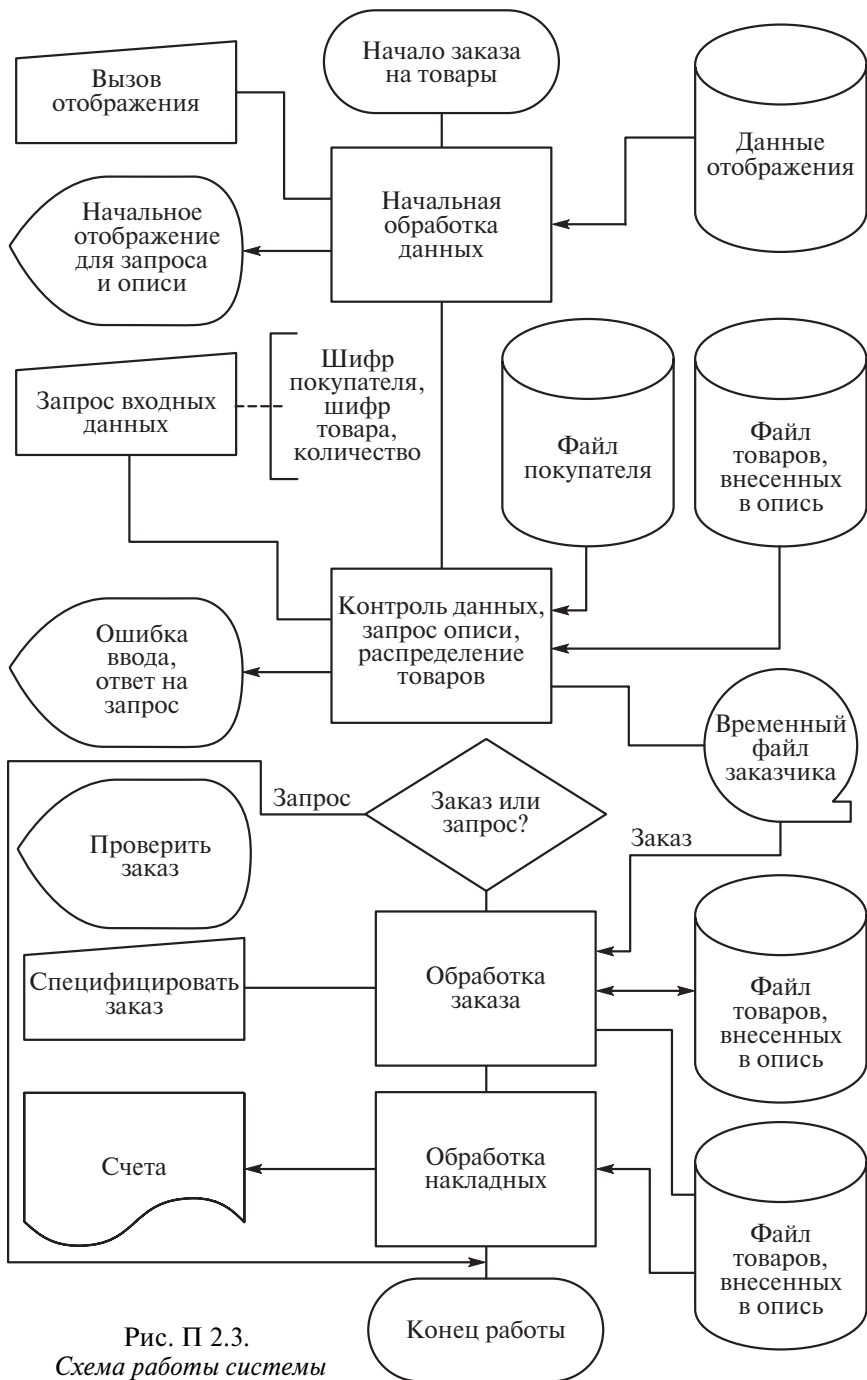
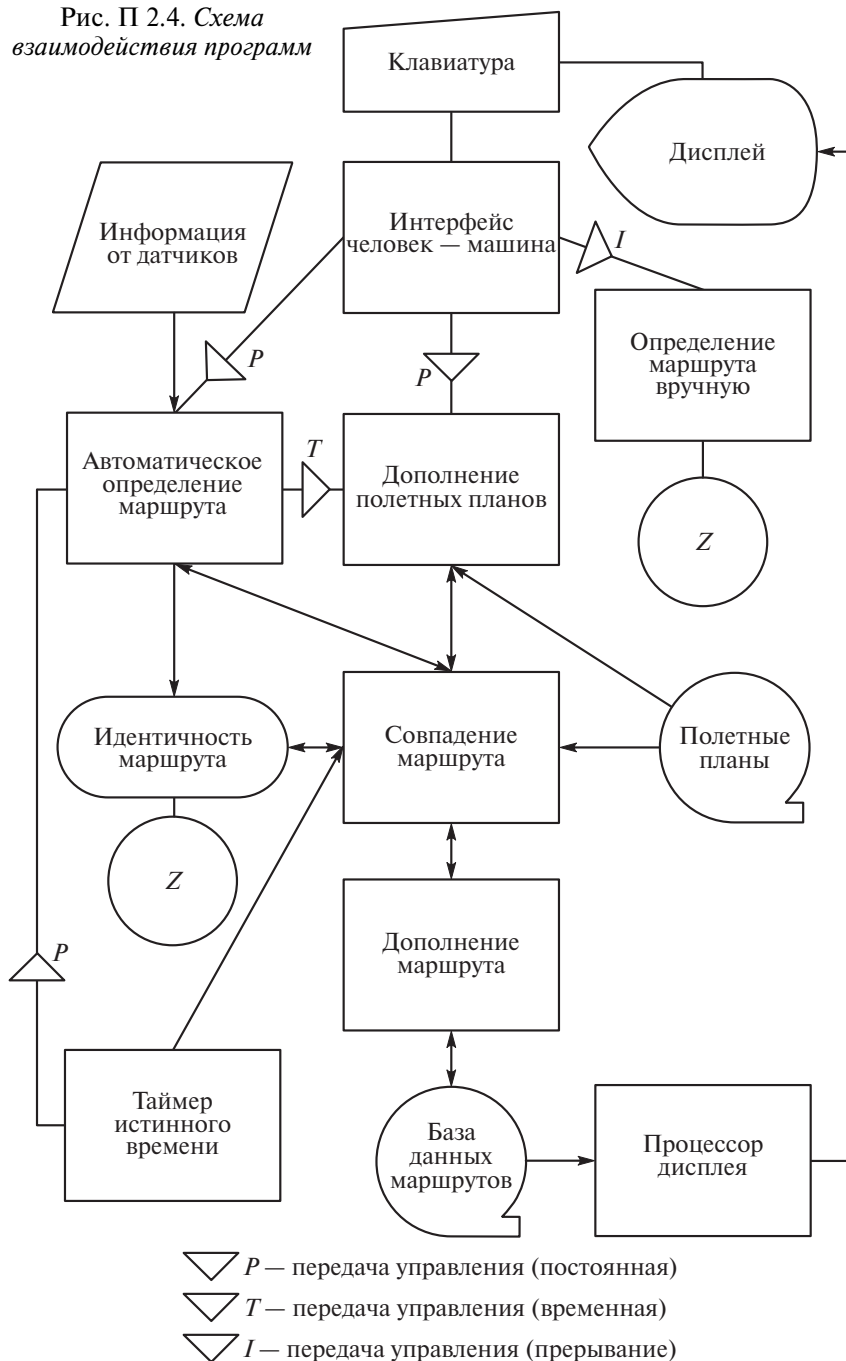


Рис. П 2.3.
Схема работы системы

Рис. П 2.4. Схема взаимодействия программ



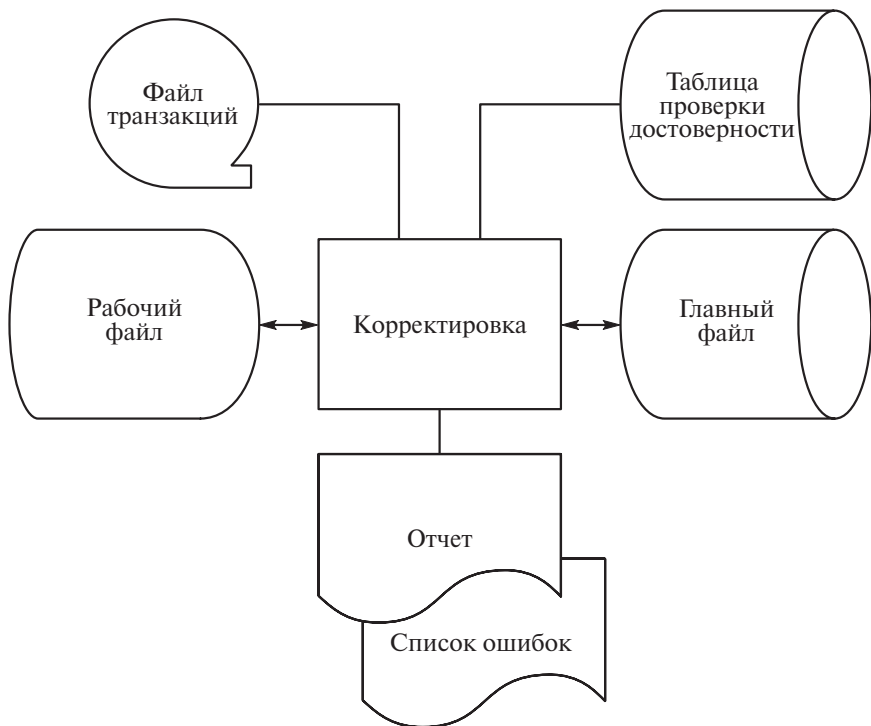


Рис. П 2.5. Схема ресурсов системы

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ	3
Список основных сокращений	7
Введение в теорию автоматизированного управления.....	9
<i>В.1. Объективная необходимость автоматизации обработки информации и управления.</i>	9
<i>В.2. Краткая историческая справка о развитии теории автоматизированного управления</i>	10
<i>В.3. Цель и задачи дисциплины «Теоретические основы автоматизированного управления»</i>	13
РАЗДЕЛ 1. МАТЕМАТИЧЕСКИЕ МЕТОДЫ ТЕОРИИ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ.	15
ГЛАВА 1. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ	15
1.1. Понятия «управление» и «система управления»	15
1.2. Терминология теории автоматизированного управления.	22
1.3. Этапы управления	28
1.4. Объект и предмет теории автоматизированного управления	32
<i>Контрольные вопросы</i>	43
ГЛАВА 2. МЕТОДОЛОГИЯ ПОСТРОЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ.	44
2.1. Классификация автоматизированных систем	44
2.2. Основные принципы построения автоматизированных систем	51
2.3. Этапы разработки АС	54
2.4. Задачи, решаемые на стадиях проектирования АС. ..	57
<i>Контрольные вопросы</i>	60
ГЛАВА 3. КАТЕГОРИАЛЬНЫЕ ПОНЯТИЯ СИСТЕМНОГО АНАЛИЗА АВТОМАТИЗИРОВАННЫХ СИСТЕМ.	60
3.1. Системность как общее свойство материи	61
3.2. Место системного анализа в системных представлениях.	65
3.3. Развитие системного анализа	68
3.4. Методики и процедуры системного анализа	72
3.4.1. Принципы, этапы и процедуры системного анализа	72

3.4.2.	Определение целей системного анализа.	77
3.4.3.	Анализ структуры системы.	80
3.4.4.	Сбор данных о функционировании системы	84
3.4.5.	Исследование информационных потоков . . .	87
3.4.6.	Построение моделей системы	89
3.4.7.	Проверка адекватности моделей. Анализ неопределенности и чувствительности	92
3.4.8.	Исследование ресурсных возможностей.	97
3.4.9.	Формирование критериев.	99
3.4.10.	Генерирование альтернатив	101
3.4.11.	Реализация выбора и принятия решений . . .	106
3.4.12.	Внедрение результатов анализа	110
	<i>Контрольные вопросы</i>	112
ГЛАВА 4. МОДЕЛИ АНАЛИЗА СТРУКТУРЫ АСУ		112
4.1.	Цели и задачи структурного анализа АСУ	113
4.2.	Уровни описания структуры АСУ	115
4.3.	Формализация описания структуры методами теории графов	117
4.3.1.	Способы формализованного задания графа. .	117
4.3.2.	Порядковая функция на графе.	122
4.3.3.	Числовая функция на графе	127
4.4.	Топологическая декомпозиция структур АСУ	129
4.5.	Модели описания и анализа потоков информации в АСУ.	133
4.6.	Структурно-топологические характеристики систем и их применение	144
4.7.	Модели функционирования организационной системы	163
4.8.	Пример разработки модели функционирования организационной системы	175
	<i>Контрольные вопросы</i>	181
ГЛАВА 5. МОДЕЛИ СИНТЕЗА СТРУКТУРЫ АСУ		182
5.1.	Формализация общей задачи синтеза структуры АСУ	183
5.2.	Частные задачи синтеза оптимальной структуры АСУ.	185
5.2.1.	Частные критерии оптимизации	185
5.2.2.	Ограничения в частных задачах синтеза.	186
5.2.3.	Первая частная задача синтеза оптимальной структуры АСУ	186

5.2.4. Вторая частная задача синтеза оптимальной структуры АСУ	188
5.2.5. Третья частная задача синтеза оптимальной структуры АСУ	190
5.3. Примеры частных задач синтеза оптимальной структуры АСУ.	190
<i>Контрольные вопросы</i>	<i>196</i>
ГЛАВА 6. МОДЕЛИ И ПРОЦЕСС ПРИНЯТИЯ РЕШЕНИЙ В АСУ	197
6.1. Проблема принятия решений в больших системах.	198
6.2. Процесс принятия решений.	200
6.3. Общая постановка задачи принятия решений.	203
6.4. Классификация задач принятия решений	204
6.5. Однокритериальные задачи принятия решений.	206
6.6. Принятие решений в условиях риска	209
6.7. Принятие решений в условиях неопределенности.	211
6.8. Многокритериальные задачи принятия решений.	213
<i>Контрольные вопросы</i>	<i>230</i>
РАЗДЕЛ 2. ПРИКЛАДНЫЕ ВОПРОСЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ	231
ГЛАВА 7. ВИДЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ	231
7.1. Централизованное и децентрализованное управление.	231
7.2. Иерархическое управление	234
7.3. Основные типы иерархий	238
7.4. Формализация иерархических понятий	246
7.5. Принципы управления сложными системами	257
7.6. Эргатические системы управления	277
7.6.1. Особенности человеко-машинных (эргатических) систем управления	277
7.6.2. Инженерно-психологические проблемы создания и эксплуатации эргатических систем управления.	279
7.6.3. Специфика анализа и синтеза эргатических систем управления.	282
7.6.4. Типовые противоречия в процессе создания новых эргатических систем управления.	286
7.7. Типовые организационные структуры управления производством	290
<i>Контрольные вопросы</i>	<i>292</i>

ГЛАВА 8. АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ	293
8.1. Автоматизированные системы управления предприятием.	294
8.1.1. Организационная структура предприятия ...	294
8.1.2. Пример документооборота предприятия	296
8.1.3. Функциональная структура АСУП	297
8.1.4. Подсистема оперативного управления основным производственным процессом. ...	299
8.2. Автоматизированные системы управления технологическим процессом.	302
8.2.1. Классификация и характерные особенности АСУТП	305
8.2.2. Структура АСУТП	309
8.2.3. Гибкие производственные системы (ГПС)...	314
8.2.4. Особенности проектирования АСУТП	320
8.2.5. Порядок разработки АСУТП	322
8.2.6. Особенности проектирования АСУТП различного исполнения.	325
8.2.7. Автоматизированная система контроля и регулирования вращающихся печей	329
8.2.8. Система числового программного управления технологическим оборудованием NC-2000	344
8.2.9. АСУТП для резервуаров с реагентами цеха редких металлов.	354
8.2.10. Подсистема АСУ 3-03Р системы управления и защиты исследовательского ядерного реактора ПИК	363
8.3. Системы автоматизированного проектирования	381
8.3.1. Общие сведения о проектировании	382
8.3.2. Классификация САПР	387
8.3.3. Системный подход к автоматизации проектирования и принципы организации САПР	391
8.3.4. Виды обеспечения САПР	393
8.3.5. Математические модели проектируемых объектов в САПР	399
8.3.6. Информационное обеспечение САПР	407
8.3.7. Программное обеспечение САПР	410
8.3.8. Техническое обеспечение САПР	419
8.3.9. Организация взаимодействия конструктора с ЭВМ	429
8.3.10. Стадии создания САПР	433

8.4. Автоматизированные	
банковские системы управления	439
8.4.1. Характеристика банковской деятельности в РФ как предметной области информатизации	439
8.4.2. Функциональные задачи и модули автоматизированной банковской системы. . .	441
8.4.3. Информационное обеспечение АБС.	449
8.4.4. Программное обеспечение информационных технологий в банках.	452
8.4.5. Техническое обеспечение	457
8.4.6. Учет межбанковского взаимодействия при создании АБС	461
<i>Контрольные вопросы</i>	464
ГЛАВА 9. ОБЕСПЕЧИВАЮЩИЕ ПОДСИСТЕМЫ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ	465
9.1. Математическое обеспечение автоматизированных систем	470
9.2. Информационное обеспечение автоматизированного управления	487
9.2.1. Процедуры обработки информации	492
9.2.2. Организация информационных процессов в системах управления.	496
9.2.3. Особенности новой информационной технологии управленческой деятельности . . .	499
9.2.4. Базы данных и их системы управления	501
9.2.5. Модели данных.	508
9.3. Программное обеспечение автоматизированного управления	512
9.3.1. Общее и специальное программное обеспечение	512
9.3.2. Пакеты прикладных программ автоматизированного управления	514
9.3.3. Организация разработки программных средств	527
9.4. Техническое и технологическое обеспечение автоматизированного управления	537
9.4.1. Сетевой режим автоматизированной обработки экономической информации	537
9.4.2. Технология использования автоматизированных рабочих мест	543
9.4.3. Технология использования пакетов прикладных программ	547

9.4.4. Технология обработки текстовой информации	552
9.4.5. Технология обработки табличной информации	555
9.4.6. Технология использования автоматизированных банков данных.	559
9.4.7. Интегрированные технологии в распределенных системах обработки данных	564
9.4.8. Технология использования экспертных систем.	571
9.5. Лингвистическое, организационно-методическое, эргономическое и правовое обеспечение автоматизированного управления	579
9.5.1. Особенности технологии взаимодействия пользователя с ЭВМ	580
9.5.2. Типы диалога и формы его реализации на ЭВМ	584
9.5.3. Организация диалоговой технологии обработки данных на ПЭВМ.	589
9.5.4. Организационно-методическое, эргономическое и правовое обеспечение автоматизированного управления	594
9.6. Защита информации при автоматизированном управлении	595
<i>Контрольные вопросы</i>	<i>602</i>
Заключение	604
Список литературы	609
Приложения.	611