

Министерство образования и науки Российской Федерации  
Сибирский государственный аэрокосмический университет  
имени академика М. Ф. Решетнева

# **АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ**

*Утверждено редакционно-издательским советом университета  
в качестве учебного пособия для студентов бакалавриата  
по направлению подготовки 15.03.04 «Автоматизация  
технологических процессов и производств»,  
профиль «Автоматизация технологических процессов  
и производств химической отрасли»,  
очной формы обучения*

Красноярск 2016

УДК 681.5(075.8)  
ББК 32.966я73  
А22

Авторы:

И. В. Ковалев, В. В. Лосев, В. В. Брезицкая,  
Д. И. Ковалев, А. В. Перанцева, Е. В. Туева

Рецензенты:

доктор технических наук, профессор С. А. БРОНОВ  
(Красноярский государственный аграрный университет);  
доктор технических наук, профессор А. Н. АНТАМОШКИН  
(Сибирский государственный аэрокосмический университет  
имени академика М. Ф. Решетнева)

**Автоматизированные системы управления** : учеб. пособие /  
А22 И. В. Ковалев, В. В. Лосев, В. В. Брезицкая и др. ; Сиб. гос. аэрокос-  
мич. ун-т. – Красноярск, 2016. – 240 с.

ISBN 978-5-86433-703-5

Рассмотрены вопросы автоматизации систем управления предприятий. Представлены математические модели и методы анализа процессов параллельной обработки информации АСУ, включая детерминированные и стохастические интерпретации модели. Уделено внимание алгоритмической GERT-процедуре определения нормативного времени выполнения параллельных процессов в условиях неопределенности. Предложена методика расчета экономической эффективности автоматизации управления организационно-технологическими процессами предприятий.

Предназначено для студентов бакалавриата по направлению подготовки 15.03.04 «Автоматизация технологических процессов и производств», профиль «Автоматизация технологических процессов и производств химической отрасли», очной формы обучения. Также может быть полезно студентам других направлений при изучении вопросов, связанных с автоматизированными системами управления.

**УДК 681.5(075.8)**  
**ББК 32.966я73**

ISBN 978-5-86433-703-5

© Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева, 2016  
© Ковалев И. В., Лосев В. В., Брезицкая В. В.,  
Ковалев Д. И., Перанцева А. В., Туева Е. В., 2016

## ОГЛАВЛЕНИЕ

<b>Список принятых сокращений .....</b>	<b>5</b>
<b>Введение .....</b>	<b>7</b>
<b>Глава 1. Основные характеристики и принципы построения автоматизированных систем управления .....</b>	<b>11</b>
1.1. Система «технологический процесс – автоматизированная система управления технологическим процессом» .....	11
1.2. Принципы построения автоматизированных систем управления .....	22
Контрольные вопросы и задания .....	31
<b>Глава 2. Системный анализ технологических объектов управления .....</b>	<b>32</b>
2.1. Информационное обследование и описание технологических объектов управления .....	32
2.2. Анализ качества функционирования технологических объектов управления .....	39
Контрольные вопросы и задания .....	63
<b>Глава 3. Математическое обеспечение проектирования автоматизированных систем управления технологическим процессом .....</b>	<b>64</b>
3.1. Алгоритмизация технологических процессов и математическое обеспечение .....	64
3.2. Методы построения математических моделей технологических процессов .....	66
3.3. Методы оценки и оптимизации состояния объекта управления .....	89
3.4. Процессы параллельной обработки информации.....	99
3.5. Детерминированные модели формирования параллельных процессов .....	106
3.5.1. Классификация методов планирования параллельных процессов .....	106
3.5.2. Однопроцессорные модели .....	115
3.5.3. Конвейерные модели .....	121
3.5.4. Мультипроцессорные модели .....	127
Контрольные вопросы и задания .....	161

<b>Глава 4. Алгоритмическая реализация процессов обработки данных автоматизированных систем управления технологическими процессами</b> .....	162
4.1. Алгоритм оптимизации сетевых моделей методом случайного поиска с пересчетом с переменной величиной шага .....	162
4.2. Алгоритмическая GERT-процедура определения нормативных времен выполнения параллельных процессов в условиях неопределенности .....	170
4.3. Алгоритм Дейкстры в решении задачи определения временного пути в детерминированных моделях.....	182
Контрольные вопросы и задания .....	190
<b>Глава 5. Основные этапы проектирования и внедрения автоматизированных систем управления</b> .....	191
5.1. Процессы жизненного цикла автоматизированных систем управления .....	191
5.2. Этапы проектирования и внедрения автоматизированных систем управления .....	193
Контрольные вопросы и задания .....	216
<b>Глава 6. Методы анализа и расчета экономической эффективности применения автоматизированных систем управления</b> .....	217
6.1. Системный анализ эффективности автоматизации технологических процессов .....	217
6.2. Методика детерминированного расчета экономической эффективности .....	225
Контрольные вопросы и задания .....	231
<b>Заключение</b> .....	232
<b>Библиографические ссылки</b> .....	234
<b>Библиографический список</b> .....	238
<b>Ключевые понятия</b> .....	239

## СПИСОК ПРИНЯТЫХ СОКРАЩЕНИЙ

АСУ	автоматизированная система управления
АСУП	автоматизированная система управления предприятием
АСУ ТП	автоматизированная система управления технологическим процессом
АСУ ОТП	автоматизированная систем управления организационно-технологическими процессами
ВС	вычислительная система
ВВР	вектор временной развертки
ВР	вектор реализации
ВТ	вычислительная техника
ИУС	информационно-управляющая система
КВГ	верхняя контрольная граница
КГН	нижняя контрольная граница
КГТ	теоретическая контрольная граница
КГП	практическая контрольная граница
КСУП	корпоративная система управления производством
КТС	комплекс технических средств
МГУА	методы группового учета аргументов
ТЗ	техническое задание
ТОУ	технологический объект управления
ТЦУ	технологический цикл управления
УВК	управляющий вычислительный комплекс
у. е. в.	условная единица времени
BS	неприоритетные (простые) планы (Basic Schedules)
GS	общие планы (General Schedules)

IFP	фиксированные приоритеты (Intelligent Fixed Priority)
JO	упорядочивание Джонсона (the Johnson Ordering)
LPT	наидлиннейшее время обработки (Longest Processing Time)
ME	стратегия планирования «больше и раньше» (More And Earlier)
MJO	модифицированное упорядочение Джонсона (Modified Johnson Ordering)
MTS	минимальное временное квантование (Minimal Time Slicing)
PS	планы с приоритетными прерываниями (Preemptive Schedules)
RMP	назначения с монотонными приоритетами (Rate Monotonic Priority)
RPT	оставшееся время обработки (Remaining Processing Time)
SPT	наикратчайшее время обработки (Shortest Processing Time)
SLA	простой уровневый алгоритм (Simple Level Algorithm)
TCP	критичный по времени процесс (Time Critical Process)
QAD	«быстрый и приближенный» алгоритм (Quick And Dirty)

## ВВЕДЕНИЕ

Развитие наукоемких производств и многостадийных технологических процессов в химической промышленности требует использования как более совершенных автоматизированных систем управления на базе вычислительной техники и математических методов описания технологических процессов, так и более квалифицированного обслуживающего персонала.

Сложность создания современных автоматизированных систем управления и подготовки необходимых специалистов объясняется тем, что технологические процессы протекают, как правило, в высокотемпературных и химически агрессивных средах, в агрегатах большой единичной мощности. Кроме того, исследование технологических процессов и обучение оперативного персонала на действующих технологических агрегатах требует значительных затрат времени и капитальных средств и может привести к возникновению аварийных ситуаций в случае применения недопустимых управляющих воздействий.

Свободными от этих недостатков являются новые информационные технологии исследования технологических процессов и производств в целом, а также обучение рабочих, построенное на базе автоматизированных информационно-советующих систем управления, реализованных на персональных ЭВМ (компьютерных тренажеров).

Использование вычислительной техники (ВТ) в информационно-управляющих системах (ИУС) всегда сопровождалось возникновением многих научных проблем, связанных с эффективным применением ВТ. Эти проблемы исключительно разнообразны и охватывают такие вопросы как создание алгоритмических языков и трансляторов, разработка операционных и инструментальных систем и средств взаимодействия с ЭВМ, построение библиотек и пакетов программ и т. п.

Понятие эффективности затрагивает не только сферу обеспечения пользователей ИУС различными информационными услугами, упрощающими и облегчающими общение с ВТ. Существенным образом с этим понятием связано и достижение максимальной производительности. Последний фактор имеет особое значение для суперЭВМ и больших параллельных систем, предназначенных для сложных и трудоемких задач, решаемых на грани возможностей ВТ. Это исключительно важно для критичных приложений ИУС в аэрокосмической и ядерной отраслях. Для указанных приложений обеспечение максимальной эффективности использования играет не меньшую роль, чем увеличение технических возможностей ИУС.

Эффективность использования ВТ в ИУС всегда была и остается критическим вопросом. Но если среди разработчиков технических устройств, операционных систем и программного обеспечения эта проблема существует, прежде всего, на этапе проектирования аппаратно-программных средств, то для пользователей, в первую очередь алгоритмистов и системных аналитиков, с созданием ВТ она только появляется. Лишь в процессе решения задач обработки информации и управления, реализуемых в ИУС, эти пользователи во всей полноте ощущают на себе достоинства и недостатки принятых технических и системных решений.

В условиях появления больших параллельных систем и перспектив создания сверхмощных новых систем для использования в ИУС открывается обширнейшая и малоосвоенная область исследований, связанная с проблемами эффективного формирования параллельных процессов в ИУС, а также совместным изучением этих структур и вычислительных систем, на которых они реализуются. Поэтому эффективное формирование параллельных процессов в действительности должно начинаться с построения математической модели работы большого числа устройств с изменяемыми в общем случае связями и с анализа на основе данной модели особенностей информационных потоков, проходящих через всю совокупность устройств. Это позволит разработать методы и алгоритмы формирования параллельных процессов в ИУС, а также предложить подходы к их оптимизации.

Наиболее сложным моментом в решении поставленных задач является выбор средств математического описания как вычислительных систем, так и процессов в ИУС. Сложность выбора определяется не столько тем, что с помощью этих средств требуется фиксировать описание каких-либо объектов, сколько необходимостью задавать и распознавать различные их свойства, а также сравнивать и преобразовывать описания отдельных объектов.

В качестве средств описания широко используются графы (в том числе потоковые), а также сети (детерминированные и стохастические GERT-сети). В той или иной мере указанные модели применялись и ранее, тем не менее их использование было лишь фрагментарным. Это можно объяснить, в частности, тем, что при сравнении параллельных алгоритмов, заданных произвольными графами, для осуществления этой операции (сравнение в этом случае является *NP*-сложной задачей) в общем случае требуется полный перебор всех вариантов. Отсюда нередко делаются пессимистические выводы. В подобных ситуациях предлагаются модифицированные алгоритмы оптимизации

сетевых моделей методом случайного поиска. Например, при формировании оптимального расписания выполняемых операций в ИУС может использоваться многоэтапная оптимизационная процедура случайного поиска с пересчетом с переменной величиной шага.

В данном учебном пособии рассмотрены вопросы автоматизации систем управления предприятий. Издание составлено в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 15.03.04 «Автоматизация технологических процессов и производств» (уровень бакалавриата) № 200, утвержденным приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г.

Целью изучения дисциплины «Автоматизированные системы управления», которая тесно связана с изученными ранее дисциплинами «Математика», «Прикладной системный анализ», «Вычислительная математика» «Вычислительные машины, системы и сети», является формирование у студентов способностей (компетенций):

– участвовать в постановке целей проекта (программы), его задач при заданных критериях, целевых функций, ограничений, разработке структуры его взаимосвязей, определении приоритетов решения задач с учетом правовых и нравственных аспектов профессиональной деятельности, разработке проектов изделий с учетом технологических, конструкторских, эксплуатационных, эстетических, экономических и управленческих параметров, в разработке проектов модернизации действующих производств и создании новых, разработке средств и систем автоматизации, контроля, диагностики, испытаний, управления процессами, жизненным циклом продукции и ее качеством в соответствии с техническими заданиями и использованием стандартных средств автоматизации расчетов и проектирования (ПК-4);

– участвовать в разработке проектов по автоматизации производственных и технологических процессов, технических средств и систем автоматизации, контроля, диагностики, испытаний, управления процессами, жизненным циклом продукции и ее качеством, в практическом освоении и совершенствовании данных процессов, средств и систем (ПК-7);

– участвовать в разработке мероприятий по проектированию процессов разработки и изготовления продукции, средств и систем автоматизации, контроля, диагностики, испытаний, управления производством, жизненным циклом продукции и ее качеством, их внедрения (ПК-14);

– участвовать в работах по моделированию продукции, технологических процессов, производств, средств и систем автоматизации,

контроля, диагностики, испытаний и управления процессами, жизненным циклом продукции и ее качеством с использованием современных средств автоматизированного проектирования, а также в разработке алгоритмического и программного обеспечения средств и систем автоматизации и управления процессами (ПК-19);

– разрабатывать практические мероприятия по совершенствованию систем и средств автоматизации и управления изготовлением продукции, ее жизненным циклом и качеством, а также по улучшению качества выпускаемой продукции, технического обеспечения ее изготовления, практическому внедрению мероприятий на производстве; осуществлять производственный контроль их выполнения (ПК-29);

– участвовать в разработке обобщенных вариантов решения проблем, связанных с автоматизацией производств, выборе на основе анализа вариантов оптимального прогнозирования последствий решения (ОПК-4).

Овладение данной дисциплиной способствует изучению таких дисциплин как «Инструментальные средства программирования в системах управления», «Автоматизация технологических процессов», «Проектирование автоматизированных систем», «Автоматизация управления жизненным циклом продукции», а также повышает подготовленность студентов к прохождению производственной практики.

## Глава 1

# ОСНОВНЫЕ ХАРАКТЕРИСТИКИ И ПРИНЦИПЫ ПОСТРОЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

### 1.1. СИСТЕМА «ТЕХНОЛОГИЧЕСКИЙ ПРОЦЕСС – АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ»

Материальной базой любого производства служат технологические процессы, поэтому для повышения таких характеристик производства как производительность, качество (надежность) выпускаемой продукции, рентабельность производства необходимо обеспечить управляемость процессов, в частности, за счет введения автоматизированных систем управления технологическим процессом (АСУ ТП).

В понятие «технологический процесс как объект управления» включается технологическое оборудование, кроме датчиков и исполнительных органов, которые являются конструктивными элементами оборудования, но входят в состав технических средств АСУ ТП, поэтому управление технологическим процессом означает управление режимами работы технологического оборудования.

Под термином «управляемый технологический процесс» понимается такой процесс, для которого определены входные контролируемые воздействия (управляющие и управляемые), установлены детерминированные или вероятностные зависимости между входными воздействиями и выходными параметрами выпускаемого изделия (продукта), разработаны методы автоматического измерения входных воздействий и выходных параметров (всех или их части), а также методы управления.

Задачу, выполняемую в системе «технологический процесс – АСУ ТП», можно сформулировать следующим образом: по полученным данным о технологическом процессе составить прогноз его протекания, а также составить и реализовать такой план управляющих воздействий (в том числе изменение режимов работы оборудования), чтобы в определенный момент времени состояние технологического процесса отвечало некоторому экстремальному значению обобщенного критерия качества. Для решения этой задачи необходимо иметь математическую модель процесса, которая определяет основное содержание алгоритма управления, реализуемого АСУ ТП.

В общем виде математическая модель системы «технологический процесс – АСУ ТП» представляет собой зависимость вида

$$y(t + \Delta t) = A \{ u(t) + By(t); F[\xi(\tau), \eta(\tau)] \}, \quad (1.1)$$

где  $y = \{y_1, y_2, \dots, y_n\}$  – выходная переменная;  $\Delta t$  – время от начального цикла действия АСУ ТП до получения контрольной информации о его результатах;  $A$  – оператор действия АСУ ТП в целом;  $u(t) = \{u_1(t), u_2(t), \dots, u_n(t)\}$  – входные контролируемые воздействия;  $B$  и  $F$  – операторы управляющих и неуправляющих воздействий соответственно;  $\xi(t) = \{\xi_1(\tau), \xi_2(\tau), \dots, \xi_n(\tau)\}$  – контролируемые, но неуправляемые воздействия (например, измеряемые параметры исходных материалов, используемых в процессе);  $\eta(\tau) = \{\eta_1(\tau), \eta_2(\tau), \dots, \eta_n(\tau)\}$  – неконтролируемые воздействия.

В выражении (1.1) интервалы изменения временных параметров  $t$  и  $\tau$  находятся в следующей зависимости:

$$t_0 \leq t \leq t_0 + T; \quad t \leq \tau \leq t + \Delta t.$$

Здесь  $t_0$  – начало отсчета времени;  $T$  – длительность интервала наблюдения за поведением процесса.

В начальный момент времени выполняется равенство  $y(t_0) = 0$ , т. е. результат действия системы управления равен нулю. Следовательно,  $y(t)$  для АСУ ТП есть кусочно-гладкая монотонная неубывающая функция, поскольку отрицательное значение выпуска продукции не имеет смысла. За  $t_0$  в функции  $y(t)$  можно принять любой момент времени.

С учетом ограниченных ресурсов системы управления и процесса вид математической модели, в первую очередь, зависит от соотношения между временем реализации управляющего воздействия и длительностью цикла технологического процесса. В общем случае время запаздывания управляющего воздействия относительно изменения состояния технологического процесса имеет вид

$$\tau_{\text{зап}} = n\tau_{\text{пр}} \quad (1.2)$$

где  $n$  – некоторая константа,  $0 < n < \infty$ ;  $\tau_{\text{пр}}$  – время, прошедшее от изменения состояния входных параметров процесса до изменения выходных параметров (время процесса).

Для АСУ ТП с управляющими ЭВМ всегда  $n > 0$ . Если  $0 < n \leq 1$ , то возможно синхронное управление в реальном времени. В этом случае

$$\tau_{\text{зап}} = \tau_{\text{вв}} + \tau_{\text{об}} + \tau_{\text{выб}} + \tau'_{\text{зап}} \leq \tau_{\text{пр}}, \quad (1.3)$$

где  $\tau_{\text{вв}}$  и  $\tau_{\text{об}}$  – время ввода и обработки информации о процессе в ЭВМ;  $\tau_{\text{выб}}$  – время отработки управляющего воздействия;  $\tau'_{\text{зап}}$  – время «чистого» запаздывания, т. е. время от начала действия новых управляющих воздействий  $By(t)$  до получения контрольной информации о новом значении выходной переменной.

Одним из простых практических примеров реализации математической модели с временным соотношением  $0 < n \leq 1$  служит многоконтурная стабилизация параметров процесса по одному из известных законов регулирования (П, ПИ, ПИД) в режиме прямого цифрового управления от ЭВМ.

При  $n > 1$  можно управлять не текущим, а только последующим состоянием стационарных процессов. В этом случае управляющая ЭВМ реализует циклический алгоритм управления в масштабе времени, кратном реальному ( $n = 1, 2, 3, \dots$ ).

Практическими примерами также могут служить различные алгоритмы адаптивного управления. Следует отметить, что предельный случай  $n \rightarrow \infty$  имеет ясный физический смысл: он соответствует состоянию системы управления без обратной связи (с нарушенной обратной связью).

Весьма часто при проектировании АСУ ТП, реализующей синхронный алгоритм управления, приходится учитывать, что существующие в процессе неуправляемые воздействия  $\xi(\tau)$  и  $\eta(\tau)$  могут быть определены и учтены не в текущем, а в последующем состоянии процесса (например, после предварительной статистической обработки результатов управления процессом). Поскольку  $\eta(\tau)$  – вектор случайных воздействий, характер которых в общем случае неизвестен, выражение (1.1) принимает вид

$$My(t + \Delta t) = MA \left\{ u(t) + B[My(t)], \xi(\tau) \right\}, \quad (1.4)$$

где  $M$  – символ математического ожидания.

В этом случае целесообразно говорить о синхронно-циклическом алгоритме управления.

Приведенный общий анализ алгоритма работы системы технологический процесс – АСУ ТП» можно распространить на процессы как непрерывные и полунепрерывные, так и на дискретные с учетом иерархичности производства.

Рассмотрим состав функций, выполняемых по уровням управления в двухуровневых АСУ ТП. Такие системы имеют блочную структуру (рис. 1.1).

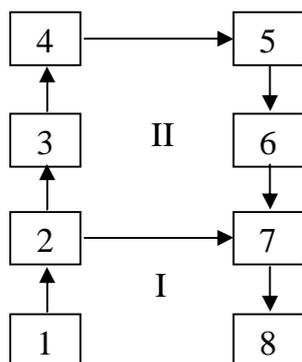


Рис. 1.1. Пример распределения функций управления в двухуровневых АСУ ТП

На *уровне управления технологической операцией* (область I на рис. 1.1) происходит получение информации о выпуске изделий, технологических потерях и браке, параметрах технологических операторов, контрольных измерений, а также осуществляются первичная фильтрация информации (выделение полезного сигнала) и контроль соответствия режима функционирования технологической операции контрольным границам. Обычно выполнения этих функций достаточно, чтобы управлять относительно простыми технологическими операциями, имея в виду процессы обработки информации об их ходе (блоки 1 и 2 на рис. 1.1).

Для технологических процессов в состав основных функций управления может быть включен анализ соответствия результатов, полученных на технологической операции, прогнозу хода технологической операции, выполненному на базе математической модели операции. Эта функция реализуется для определения соответствия математической модели реальному ходу технологической операции.

На *уровне управления технологическим процессом* (область II на рис. 1.1) осуществляется управление процессом сбора информации с технологических операций и ее интеграции. По полученной информации проверяют соответствие математической модели реальному ходу технологического процесса (блоки 3 и 4 на рис. 1.1).

При управлении одной технологической операцией после выполнения функций анализа системы управления переходят к решению задачи оптимизации технологического процесса. Для этого можно либо по известным алгоритмам откорректировать математическую модель технологической операции, либо ввести в работу адаптационные алгоритмы. Для относительно простых технологических операций управление технологическим процессом можно оптимизировать в соответствии с заложенным в систему законом управления (например, ПИД-законом). Решения, полученные в системе управления, в последующем преобразуются в управляющие задания на регуляторы (если такие имеются в АСУ ТП) и управляющие органы (блоки 5 и 7 на рис. 1.1). Таким образом завершается цикл управления технологической операцией.

Если после анализа математической модели в ней выявляются несоответствия требованиям, предъявляемым к технологическому процессу, для двухуровневой системы управления в работу вводят адаптационный алгоритм. Результаты его применения распределяются в виде заданий на системы управления технологическими операциями (блоки 5 и 6 на рис. 1.1).

Помимо функций, связанных с управлением технологическими процессами, в иерархических интегрированных АСУ ТП реализуются и функции управления организационными процессами. По уровням управления они распределяются следующим образом. На уровне технологической операции это, прежде всего, отчет о выпуске годной продукции, технологических потерях и браке, состоянии оборудования, ресурсного обеспечения (блок 1 на рис. 1.1). Последующий логический счетный и формативный контроль необходимы, если информация используется как при автономном управлении технологической операцией, так и при управлении технологическим процессом в целом (блок 2). Следующими по порядку реализации являются интеграция отчетных данных, получаемых с технологических операций, анализ качества организационного управления технологическим процессом, а также необходимый технико-экономический анализ производства (блоки 3 и 4). В блоках 5 и 6 реализуются задачи организации работы технологического процесса – как правило, посредством функционирования системы оперативного планирования производства и его ресурсного обеспечения, которая выдает плановые задания (на сутки, смену) на технологические операции, а также задания службам обеспечения. Если технологическая операция не находится в жестком режиме конвейера, то на ее уровне можно составлять оперативный план выполнения суточного или сменного задания (блок 7). Реализация организационного плана работы технологической операции показана блоком 8.

Известны несколько критериев классификации автоматизированных систем управления. Так, например, уже порядка 20 лет в научной литературе и практической работе используют классификацию технических средств АСУ ТП, за критерий которой принят уровень решаемых задач по управлению объектом (централизованный контроль, управление в режиме советчика, управление в замкнутом контуре). Однако существует и ряд других классификаций. Рассмотрим их.

*Классификация АСУ ТП по критерию сложности объектов управления.* В 1970 г. предложена классификация АСУ ТП применительно к предприятиям с непрерывным и непрерывно-дискретным характером производства, за критерий которой взято число контролируемых параметров и управляющих воздействий в соответствии с ростом сложности объекта управления (табл. 1.1).

Таблица 1.1

**Классификация АСУ ТП по сложности объектов управления**

Класс АСУ ТП	Функциональные признаки	Типовые примеры объектов управления
1–0. Автоматизированная система программного управления	Управление по жесткой программе с предварительно запрограммированными воздействиями	Станки, смесеприготовители, полиграфические машины
1–1. АСУ технологическими установками с малым числом контролируемых и регулируемых параметров (до 20)	Измерение, индикация, регистрация и регулирование параметров	Топки паровых котлов, весовые дозаторы, установки автоматического пожаротушения
1–2. АСУ технологическими установками или агрегатами с малым числом контролируемых и регулируемых параметров (около 40)	То же, что для класса 1–1, и логические операции	Технологические котельные, методические печи, нагревательные колодцы и фурмы доменных печей, ректификационные колонки
1–3. АСУ технологическими установками, агрегатами или процессами со средним числом контролируемых и регулируемых и оптимизируемых параметров (около 100)	То же, что для класса 1–2, и многоконтурное регулирование	Конверторы, комплексы шихтоподготовки обогатительных и агломерационных фабрик, химические реакторы, установки первичной переработки нефти, секционные печи

Класс АСУ ТП	Функциональные признаки	Типовые примеры объектов управления
1–4. АСУ технологическими агрегатами или процессами с большим числом регулируемых и оптимизируемых параметров (около 800)	То же, что для класса 1–3, и вычисление технико-экономических показателей	Энергоблоки, прокатные станы, доменные печи, атомные реакторы, производство этилен-бензола и печной сажи
1–5. АСУ технологическими переделами и производствами с агрегатами и установками, для местного управления которыми средства вычислительной техники не используются	То же, что для класса 1–4, и диспетчеризация при одноступенчатом уровне	Электролизные цеха производства серной кислоты, искусственного волокна, агломерационные фабрики, обогатительные фабрики
1–6. АСУ технологическими переделами и производствами с агрегатами и установками, оснащенными средствами вычислительной техники	То же, что для класса 1–5, но при двухступенчатом управлении	Конверторные цеха, доменные печи, цементные заводы, сернокислотные производства, обогатительные комбинаты

Эта классификация позволяет ориентировочно определять номенклатурную базу АСУ ТП и может служить основой для планирования разработок, однако не отражает функции, которые может выполнять та или иная система управления [1].

*Классификация АСУ ТП по функционально-алгоритмическому признаку.* По функционально-алгоритмическому признаку системы управления технологическими процессами на базе управляющих ЭВМ целесообразно делить на три класса (табл. 1.2). Эта классификация в определенной мере условна, поскольку функции, выполняемые системами указанных классов, в ряде случаев могут перекрываться. Однако такое разделение АСУ ТП в настоящее время имеет принципиальное практическое значение для развертывания работ по автоматизации технологических процессов, поскольку, осуществляя управление технологическим процессом, ЭВМ получает информацию о ходе процесса и выдает регулирующие воздействия (в частном случае – советы оператору) в соответствии с алгоритмом управления, заложенным в виде программ в запоминающие устройства.

**Классификация АСУ ТП  
по функционально-алгоритмическому признаку**

Класс АСУ ТП	Функциональные признаки	Типовые примеры объектов управления
1. Системы логико-программного управления (группой однотипных технологических установок)	Прямое цифровое управление по жесткой или полужесткой программе в режиме разделения времени между управляемыми установками	Группы автоматизированных постов контроля или испытаний изделий электронной техники, прецизионных механообрабатывающих станков, откачных вакуумных постов, термического оборудования
2. Системы оптимального управления (технологическим процессом или режимами технологической установки)	Решение задачи оптимизации, на основании получаемой от управляемого объекта информации и принятых математических моделей, выработка регулирующих воздействий или советов оператору в реальном времени	Химические реакторы, трубопрокатные станы, группа диффузионных печей, установки первичной переработки нефти
3. Системы комплексного управления (технологической линией, участком, цехом) – АСУ ОТП	Автоматический или полуавтоматический сбор, обработка, наглядное отображение технологической и организационно производственной информации, управление через оперативный персонал ходом технологических процессов	Технологические линии производства интегральных схем, кинескопов, энергоблок атомной электростанции, сернокислотное производство, доменная печь, тепловая электростанция

К 1-му классу АСУ ТП относятся системы с наиболее простой формой алгоритма управления – полностью запрограммированным ходом процесса (ранее его вел оператор). Основная функция центрального процессора АСУ ТП – логические операции по выполнению нескольких программ (в частном случае – одной) с автоматическим распределением времени. Типовым алгоритмом управления служит заранее установленная последовательность логических операций с условным или безусловным переходом от одной позиции к другой. В общем случае при управлении с помощью ЭВМ рядом

технологических установок в запоминающем устройстве хранится число программ, реализующих типовой алгоритм, соответствующее числу объектов управления. При этом мультипрограммный режим работы машины организуется с помощью специальной программы-диспетчера.

Примерами систем данного класса могут служить системы прямого многоканального цифрового регулирования (стабилизации параметров) или системы прямого цифрового управления металлорежущими станками.

АСУ ТП 2-го класса достаточно широко применяются в непрерывных и непрерывно-дискретных производственных процессах. Главными функциями центрального процессора в таких системах являются выполнение на основании входных данных, получаемых от объекта управления, математических операций и выработка по результатам вычислений регулирующих воздействий. Алгоритм управления процессом (объектом), как правило, разрабатывается на основе его детерминированной или статистической модели, что позволяет оптимизировать процесс, т. е. управлять им с целью удовлетворения некоторого критерия.

К системам 2-го класса относятся, в частности, системы прямого многосвязного цифрового управления с оптимизацией, системы управления последовательными технологическими операциями, связанными по качеству, системы адаптивного управления технологическими комплексами.

АСУ ТП 3-го класса в основном охватывают среднюю ступень иерархических систем управления производством [2]. Это класс организационно-технологических АСУ – АСУ ОТП. Главной функцией технологического характера в данном случае является управление через оперативный персонал (операторов, технологов и т. д.) ходом технологических процессов на основании статистической обработки технологической информации и текущего планового задания.

Поскольку 3-й класс систем охватывает группу технологических процессов, а следовательно, ряд различных технологических установок и целые производственные подразделения, то в функции этих систем включают также обработку планово-производственной информации и управление (по результатам этой обработки) оперативным персоналом, работой участка, цеха. Из вышесказанного следует, что алгоритмы отдельных задач, решаемых АСУ ТП 3-го класса, весьма разнообразны, носят, в первую очередь, информационно-вычислительный характер, и каждый алгоритм в отдельности прост

для программирования. Однако в целом задача анализа и прогноза хода производственного процесса (например, реализация алгоритма управления технологическим процессом в целях оптимального номенклатурного распределения выпускаемых изделий в зависимости от планового задания) может быть весьма сложной.

Следует отметить, что системы 3-го класса могут вырастать из систем 1-го и 2-го классов, когда ЭВМ осуществляет централизованное логико-программное или оптимальное управление группой технологических установок на уровне производственного участка, линии или цеха и на нее возлагаются дополнительные функции оперативно-диспетчерского управления с анализом работы производственного подразделения и прогнозом его дальнейшего хода.

АСУ ТП, представляющие собой, как правило, сложные технические системы, характеризуются рядом системных показателей эффективности, с помощью которых оценивают качество их функционирования. Понятие «эффективность» ( $E_0$ ) отражает степень соответствия системы своему назначению, ее техническое совершенство и экономическую целесообразность, а количественные показатели эффективности в общем случае представляют собой функции множества параметров реализуемых алгоритмов (управление), множества параметров комплекса технических средств (КТС) системы, множества параметров ее элементов и множества параметров (факторов) внешней среды, влияющих на функционирование системы [3].

К количественным (обобщенным) показателям эффективности АСУ ТП как системы относится в первую очередь *время решения задач управления* конкретным технологическим объектом управления (ТОУ)  $T_0$ , которое определяется множеством параметров, характеризующих АСУ ТП, ее элементы, условия эксплуатации и реализуемые алгоритмы. Данный показатель с учетом особенностей его функционирования эквивалентен стандартизованному показателю систем обработки данных любого назначения – времени работоспособности системы  $T_{cp}$ :

$$T_{cp} = T_{c.э} - T_{п.т},$$

где  $T_{c.э}$  – эксплуатационное время системы;  $T_{п.т}$  – время технических простоев системы, обусловленных необходимостью поддерживать и восстанавливать ее работоспособность (плановые профилактические работы и ремонты, неплановые контроль технического состояния и ремонты для устранения отказов и причин сбоев, включая восстановление работоспособности программных средств).

По ГОСТ 26525–85 более строго показатель  $T_0$  соответствует другому, «вспомогательному», показателю  $T_{р.в}$  – времени решения задач системой, определяемого как часть суммарного времени выполнения заданий системой, в течение которого обработку данных выполняют по отлаженным программам. Для такого класса систем обработки данных как АСУ ТП (АСУ ОТП)  $T_{р.в}$  должно в общем случае соответствовать  $T_{ср}$  (решение задач управления только по отлаженным программам).

Показатель  $T_0$  непосредственно связан с системным показателем КТС (УВК) АСУ ТП, называемым *эффективной производительностью УВК*  $W$ . Эффективную производительность для иерархических УВК в первом приближении можно оценивать суммой эффективных производительностей (числом решаемых задач в единицу времени), входящих в него как составные части отдельных УВК (ЭВМ):

$$W = \sum_{i=1}^N W_i,$$

где  $W_i$  – эффективная производительность  $i$ -го УВК;  $N$  – общее число УВК в иерархическом УВК.

Производительность как системный показатель зависит не только от быстродействия ЭВМ (числа выполняемых операций обработки данных в единицу времени), но и от времени выполнения процессов ввода, вывода, обмена информацией, времени обнаружения и устранения отказов и сбоев, степени параллельности всех выполняемых в УВК процессов.

На стадии эксплуатации важнейшими системными показателями АСУ ТП являются *показатели надежности*. Как известно, для относительно простых систем, к которым можно отнести локальные АСУ ТП, приняты следующие показатели:  $P_c(\tau)$  – вероятность безотказной работы за время  $\tau$  при заданных условиях эксплуатации;  $T_0$  – наработка на отказ;  $T_{в.с}$  – среднее время восстановления системы. Для сложных систем, к которым можно отнести иерархические АСУ ТП (АСУ ОТП), отказы отдельных элементов приводят не к отказу всей системы, а к определенному снижению общей эффективности, поэтому в качестве показателей надежности таких систем целесообразно использовать коэффициент снижения эффективности  $K_E = E / E_0$ , показывающий, какую часть эффективность реального УВК ( $E$ ) составляет от идеальной эффективности ( $E_0$ ). Также для оценки этого параметра можно использовать разность эффективностей  $E_0 = E_0 - E$ .

К группе эксплуатационных системных показателей относятся также *показатели помехозащищенности и готовности*. Количественно помехозащищенность иерархических УВК можно оценить с помощью коэффициента  $K_{пз} = E_{п} / E_{о}$ , показывающего, какую часть составляет показатель эффективности АСУ ТП, функционирующего в условиях помех с заданными параметрами ( $E_{п}$ ), от показателя эффективности для идеальных условий. Аналогично соответствующему показателю надежности для оценки помехозащищенности можно использовать абсолютную величину разности  $E_{пз} = E_{о} - E_{п}$ .

Готовность системы, отражающая качество и быстроту выполнения операций по подготовке к применению, может характеризоваться вероятностью того, что АСУ ТП приведена в готовность к выполнению своих функций за время, не превышающее допустимое или за среднее время подготовки к работе.

Наконец, важнейшим обобщенным системным показателем служит *экономичность АСУ ТП*, оцениваемая затратами на ее разработку, создание и эксплуатацию.

Следует также отметить, что в практике проектирования часто используют обобщенный показатель эффективности, выражающийся дробью – в числителе находится взвешенная сумма выбранных технических показателей, а в знаменателе – приведенные затраты. При этом применяют метод последовательных уступок по отношению к показателям с меньшим весом.

## 1.2. ПРИНЦИПЫ ПОСТРОЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

На систему и объект управления постоянно воздействуют возмущения. Физически природа и источники возмущений могут быть различными, а сами возмущения – приложены как к системе «объект управления – система управления» в целом, так и к различным ее элементам. Чтобы повысить качество систем управления, эти возмущения необходимо знать.

Общим источником возмущений как для объекта управления, так и для всей системы управления служит среда – физическая и информационная.

К физической среде относятся тепловые, электрические и магнитные поля, окружающая атмосфера (например, температура, влажность, запыленность, наличие агрессивных примесей), а также источники

энергии. Чтобы компенсировать возмущения, вызванные изменением параметров среды, желательно вводить в систему управления дополнительные контуры управления по данным возмущениям.

Под информационной средой подразумевают внешнюю информацию, поступающую в систему управления. Отсутствие управляющей информации в определенное время или ее ошибочность также можно рассматривать как возмущения, нарушающие процессы управления. Чтобы устранить такого рода возмущения, в системе управления нужно предусмотреть соответствующие программы контроля качества (достоверности) поступающей информации. Если управляющей информации нет, должны быть предусмотрены либо алгоритмы, которые обеспечивают получение требуемой информации от внешних источников методами краткосрочного прогнозирования хода в самой системе управления, либо возможность аварийной остановки системы.

Возмущения в системах управления и объектах, с которыми они работают, могут быть внешними и внутренними.

*Внешними возмущениями* называются изменения параметров материалов, комплектующих изделий, сырья, полуфабрикатов и т. п. относительно параметров, на которые был настроен технологический процесс до появления этих изменений. Кроме того, к ним можно отнести и изменение уставок в системе управления, а также сами управляющие воздействия на объект управления. Дело в том, что их отработка, которая изменяет одни контролируемые технологические операторы, приводит, в свою очередь, к большему или меньшему изменению других контролируемых технологических операторов, поскольку в объекте управления все операторы так или иначе связаны. Если эти связи существенны, то систему управления необходимо проектировать как многосвязанную, с учетом перекрестных связей. Отработка контролируемых технологических операторов по тем же причинам ведет к изменению неконтролируемых технологических операторов и, следовательно, к изменению статистических характеристик технологической операции и технологического процесса, что тоже стоит иметь в виду при проектировании системы управления.

*Внутренние возмущения* в системе и объекте управления обычно связаны с неконтролируемыми изменениями параметров оборудования, датчиков и исполнительных органов. К ним же относятся и шумы разной природы, возникающие в контуре управления. Уровень этих возмущающих воздействий снижают схемотехническими решениями либо применением в системе управления устройств более

высокого качества. Для повышения отношения «сигнал–шум» можно использовать различные фильтры (как схемотехнические, так и программные) [4].

На качество управления, если под ним понимать соответствие параметров получаемых изделий и продуктов требуемым значениям, большое влияние оказывает качество математических моделей, на основе которых ведется управление. Неточности, а тем более ошибки в этих моделях приводят к несоответствию параметров выходного продукта требуемым. Этот эффект может интерпретироваться как результат воздействия на объект управления возмущений. Снизить влияние этих возмущений на процесс управления можно только повышением соответствия математической модели объекту управления, в предельном случае – построением адаптивных управляющих систем.

Известны два способа устранения воздействия возмущений: управление (регулирование) по отклонению (принцип Ползунова–Уатта) и управление (регулирование) по возмущению (принцип Понселе).

*Принцип регулирования по отклонению* заключается в том, что система управления оказывает на объект действие, направленное на устранение отклонения контролируемой переменной от заданного значения (рис. 1.2). Отклонение значения контролируемой величины от заданного под действием любых возмущений (они вырабатываются в блоке сравнения) преобразуется в управляющее воздействие на объект управления, дополнительное к воздействию на этот же объект начальной уставки, обеспечивающей получение заданных значений контролируемых выходных параметров при отсутствии возмущений.

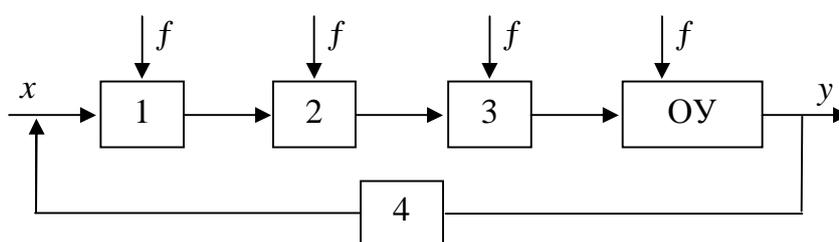


Рис. 1.2. Функциональная схема системного регулирования по отклонению:

- 1 – элемент сравнения, вырабатывающий сигнал отклонения фактического значения выходной величины от заданного;
- 2 – элемент, обеспечивающий усилие сигнала рассогласования и его преобразование в вид, необходимый для работы исполнительного органа;
- 3 – исполнительный орган;
- 4 – датчик выходной контролируемой величины;
- ОУ – объект управления;  $f$  – действующее возмущение

Однако, чтобы даже при очень больших коэффициентах усиления в цепи обратной связи на выходе ее появлялся сигнал, позволяющий компенсировать действие возмущений, необходимо, чтобы он был на входе блока обратной связи. А так как этот сигнал, по сути, является разностью между заданным значением контролируемого выходного параметра и полученным при воздействии на объект управления возмущения, то эта разность должна иметь конечное значение и, следовательно, в системах регулирования по отклонению невозможно полностью устранить воздействие возмущения на регулируемую величину. Исключение составляет лишь случай установившегося режима в астатических системах, когда блок обратной связи вырабатывает регулирующее воздействие пропорционально интегралу отклонения фактического значения контролируемой величины от заданного.

Из сказанного следует вывод, что при регулировании по отклонению влияние возмущений на объект управления хотя и можно существенно уменьшить, но нельзя устранить полностью.

АСУ ТП, построенные на принципе *регулирования по возмущению*, относительно просты в проектировании и эксплуатации, а сам принцип регулирования достаточно хорошо изучен и освоен конструкторами. В настоящее время распространено создание инвариантных АСУ ТП на основе теории построения инвариантных систем управления [5; 6].

Режимы ряда технологических процессов оказываются близкими к границам их устойчивости (например, к таковым относятся крупнотоннажные установки первичной обработки нефти). Использование принципа инвариантности позволило существенно улучшить динамические характеристики технологического процесса. С помощью теории инвариантности разработаны структура и алгоритмы управления системы стабилизации толщины проката, инвариантной относительно изменения параметров и структуры каналов [7].

К классу систем с компенсацией возмущений относятся инвариантные системы управления. Регулирование по возмущению (принцип Понселе) заключается в том, что в систему вводится специальное устройство, которое компенсирует воздействие возмущений (компенсатор). В данном же случае система обеспечивает компенсацию только одного выбранного возмущения за счет выбора структуры системы управления (структурную компенсацию), поэтому системы управления структурной компенсацией распространены весьма ограниченно.

Теория инвариантности позволяет при определенной структуре системы управления и свойствах ее элементов обеспечивать абсолютную или частичную независимость (инвариантность) одной или нескольких выходных переменных (координат) от одного или нескольких возмущающих воздействий [8–10].

В теории различают три вида инвариантности:

- абсолютную – получение нулевых вынужденной (установившейся) и переходной составляющих от действия возмущений при ненулевых начальных условиях;
- полную – компенсацию только вынужденной составляющей;
- частичную («инвариантность до  $\varepsilon$ ») компенсацию вынужденной составляющей с точностью до некоторой малой величины  $\varepsilon$ .

Одним из основных требований теории инвариантности к инвариантным системам управления является наличие двух каналов передачи возмущений (принцип «двухканальности», предложенный Б. Н. Петровым). Смысл этого требования нетрудно понять, рассмотрев простейшую функциональную схему системы регулирования по возмущению (рис. 1.3).

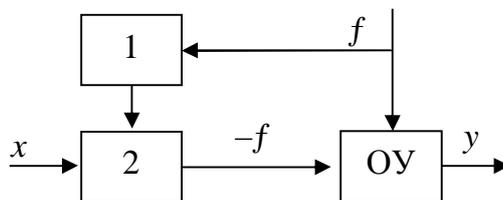


Рис. 1.3. Функциональная схема системы регулирования по возмущению:

1 – датчик возмущений; 2 – усилительно-исполнительный блок;  
ОУ – объект управления;  $f$  – действующее возмущение

Пусть некоторое возмущение  $f$  воздействует на объект управления (ОУ). Если бы нам удалось создать некоторую дополнительную цепь, проходя через которую возмущение  $f$  преобразовывалось бы в  $-f$ , то, воздействуя этим возмущением на ОУ, мы бы получили полную компенсацию, так как понятно, что  $f - f = 0$ .

Чтобы реализовать такую цепь на практике, нужно иметь датчик 1, измеряющий возмущение, и обеспечить такую передаточную функцию усилительно-исполнительного блока 2, чтобы на его выходе получить значение возмущения  $-f$ . Если передаточная функция блока 2 остается неизменной в широком диапазоне амплитуд и частот, то инвариантность системы управления будет существовать практически для всех возмущений (при условии, что они не очень велики). Это

является одним из достоинств инвариантных систем, так как на возмущающие воздействия может не накладываться никаких ограничений, за исключением конечности их величины. В ряде случаев по тем или иным причинам добиться абсолютной инвариантности не удастся.

Особой эффективности от применения методов теории инвариантности можно достичь применительно к сложным многомерным нелинейным объектам управления. Дело в том, что в многосвязанных объектах управления изменение одного из регулируемых параметров ведет к изменению других так же, как изменение управляющего воздействия для управления одной из координат приводит к изменению других. В системах управления по отклонению задача учета многосвязности управляемых переменных решается с помощью математических моделей, которые, как известно, всегда приближенные. Поэтому точность управления с ростом сложности объекта управления начинает снижаться, а в некоторых случаях может привести к неустойчивости управляемого процесса [11]. Методы теории инвариантности позволяют в определенных случаях добиться автономности регулируемых координат и исключить влияние перекрестных связей на качество управления [12].

По сравнению с ручным управлением *автоматизированные (автоматические) системы логико-программного управления* имеют следующие преимущества: во-первых, исключаются ошибочные действия человека; во-вторых, можно использовать объем информации, соответствующей знаниям специалистов многих смежных областей, который значительно превосходит объем знаний отдельного оператора. Поскольку к этому классу АСУ ТП относятся системы с алгоритмом управления, отражающим точное логико-программное ведение процесса, ранее выполняемое оператором, практически можно определить управляемый процесс как детерминированный.

Комплекс технических средств автоматизированных систем включает в себя управляющую ЭВМ, УВВИ (преобразователи типа «код–аналог» и «аналог–код», релейные выходные и входные каналы), датчики информации, исполнительные органы и (или) регуляторы. В зависимости от характера решаемых задач и числа программно управляемых технологических установок могут применяться накопители большой емкости, печатающие устройства и другие типы устройств связи ЭВМ с оперативным персоналом.

Желание повысить качество управления технологическими процессами приводит к необходимости создавать *оптимальные системы управления*. В зависимости от уровня знаний о технологическом про-

цессе, сложности динамических характеристик объекта управления, степени воздействия на технологический процесс неконтролируемых возмущений можно использовать либо жесткие, заранее запрограммированные методы достижения экстремальных параметров технологических процессов без адаптации, либо системы управления с самонастройкой – адаптационные системы [13]. Оптимальные системы управления без адаптации удобно использовать тогда, когда технологические процессы имеют точно описывающие их математические модели, а неконтролируемые возмущающие воздействия малы, либо отсутствуют вовсе. Перенастройка оптимальной системы управления без адаптации требует участия в ней либо оператора, либо управляющей ЭВМ, обеспечивающих изменение параметров оптимизации и (при необходимости) рабочих программ.

Более эффективными адаптивными АСУ ТП являются системы, в которых те или иные виды настройки выполняются автоматически. АСУ ТП с самонастройкой предназначены обычно для определения с помощью ЭВМ некоторых параметров математических моделей технологических процессов (при использовании алгоритмов МГУА, о которых будет сказано далее (см. гл. 3), можно автоматически формировать структуру математических моделей). Обычно же объектами настройки выступают коэффициенты корреляции, регрессии, пропорциональности в математических моделях законов регулирования, параметры физических моделей и процессов, используемых в системе управления и т. п.

В большинстве случаев информация о случайных переменных в технологическом процессе либо неполная, либо может отсутствовать вовсе, поэтому оптимальное управление технологическим процессом требует создания такой адаптивной системы, чтобы она в процессе работы могла самостоятельно распознавать природу и характер случайных переменных и по полученным результатам вырабатывать ту или иную оптимальную стратегию управления. Таким образом, адаптивная система в соответствии с изменяющейся обстановкой должна выполнять самонастройку как без изменения, так и, в случае необходимости, с изменением своей структуры. Другими словами, система с адаптацией осуществляет оптимальное управление процессом с помощью оптимизации структуры алгоритма управления (в некоторых случаях – структуры технических средств).

Следует отметить, что при решении проблемы построения оптимальной адаптивной системы управления во многих случаях встречаются настолько серьезные трудности, что приходится пользоваться не оптимальными, а субоптимальными стратегиями.

Основное отличие систем оптимального управления от систем логико-программного управления заключается в структуре не технических средств, а алгоритмов управления. Для технологических операций и процессов с высоким уровнем шумов оптимальная система управления должна быть построена с учетом этих обстоятельств.

Еще одним видом систем управления являются *адаптивные системы*. Условно их можно подразделить на системы для незашумленных технологических процессов и для «шумящих» объектов управления. Адаптивные АСУ ТП для «нешумящих» или «малошумящих» объектов могут быть спроектированы как полностью с автоматизированными алгоритмами адаптации, так и с алгоритмами, требующими участия оператора в адаптационном процессе. Для «шумящих» технологических процессов использование алгоритмов адаптации с оператором ведет к возникновению большого числа ошибок из-за малой эффективности работы оператора в условиях больших шумов. Поэтому в оптимальных АСУ ТП адаптационные алгоритмы полностью реализуются в ЭВМ, а в систему управления включаются специальные технические или программные блоки, обеспечивающие фильтрацию шумов.

*Адаптивные АСУ ТП для «нешумящих» объектов.* Одним из относительно простых вариантов адаптивных систем управления с оператором являются системы, в которых некоторому набору параметров технологического процесса поставлена в соответствие определенная совокупность его режимов, сформированная по правилам «если..., то...». Эта система позволяет оператору по конкретным исходным данным о технологическом процессе выбрать оптимальный план его ведения и таким образом реализовать процесс адаптации системы управления к изменяющимся условиям производства.

Примером такой адаптивной АСУ ТП может служить интегральная автоматизированная система управления кислородно-конвертерным цехом Днепропетровского металлургического комбината [14]. В этой системе на базе исходной информации на текущую плавку оператор, используя ЭВМ, может выбрать из данных о пятистах последних плавках на конвертере наиболее подходящий режим продувки и подачи материалов в конвертер. Информацию об аналогичных плавках выводят на экран дисплея в виде таблиц и графиков, а принятые оператором решения оценивает и контролирует вычислительная машина. Несмотря на относительную простоту адаптационной системы управления такого типа, она достаточно эффективна. В рассмотренном случае в результате оптимизации режимов конвертерных плавков отклю-

нения фактических содержания углерода и температуры стали от заданных для разных марок стали уменьшились на 10–15 %.

В настоящее время достаточно широко распространены адаптивные ПИД-регуляторы, используемые либо как автономные цифровые устройства, либо как устройства, входящие в состав АСУ ТП. Смысл такой адаптации заключается в автоматическом выборе коэффициентов пропорциональности, определяющих закон регулирования. Алгоритм анализирует реакцию объекта на специально вводимое в систему управления или естественно возникающее возмущение, что соответствует режиму ведения активного или пассивного эксперимента.

Общий принцип адаптации заключается в следующем. Требования к качеству регулирования формулируются в виде требований к переходной характеристике. Система анализирует расхождение между заданной и реально полученной на объекте переходной характеристикой и рассчитывает такие коэффициенты настройки, при которых это расхождение будет минимальным. Таким образом оптимизируются переходные процессы в управляемом объекте, что, в свою очередь, ведет к повышению стабильности качества продукции и минимизации энергетических затрат.

Использование адаптивных алгоритмов в ПИД-регуляторах для управления технологическими процессами позволяет улучшить реакцию системы управления на возмущение и уменьшить перерегулирование, снизить расход энергии в производстве и сократить сроки освоения технологического оборудования. Их использование в 5 % случаев дает существенный прямой эффект, в 80 % – косвенный эффект за счет вторичных факторов.

*Системы комплексного (организационно-технологического) управления.* Объектом при комплексном управлении является организационно связанная группа технологических процессов (операций) с комплексом основного и вспомогательного оборудования, а также с потоками материалов, полуфабрикатов и энергоносителей (участок, линия, цех, замкнутое производство одного продукта). Таким образом, эти системы можно определить как промежуточный организационно-технологический класс АСУ.

В большинстве систем рассматриваемого класса оперативный персонал на основании анализа результатов обработки предыдущего цикла управления, выполненной ЭВМ, принимает решение и выдает управляющие воздействия на объекты управления, т. е. во многих случаях правомерно говорить о синхронно-циклическом характере комплексного управления.

В таких системах часть управляющих воздействий может автоматически передаваться из ЭВМ на исполнительные органы, регуляторы или АСУ ТП низшего уровня. Задачи, решаемые АСУ, имеют смешанный организационно-технологический характер, поскольку объект управления представляет собой группу организационно связанных технологических подсистем. Оперативный персонал является основным звеном АСУ для принятия решения и выдачи управляющих воздействий, поскольку сложность структуры объекта управления с учетом его неупорядоченности в общем случае пока не позволяет создавать с практически удовлетворительной степенью приближения детерминированные или статистические модели для автоматического управления объектом в целом. АСУ комплексного управления должны иметь связь с нижним и верхним уровнями АСУП, имеющей иерархическую структуру, поскольку они занимают в ней среднее положение.

Основная форма передачи информации наряду с электрическими сигналами – ее наглядное отображение. В отличие от АСУ организационного характера в АСУ ТП документ не является основной формой передачи информации, а принцип автоматизации документооборота реализуется лишь при составлении сводных форм о работе объекта управления (участка, линии, цеха, замкнутого производства одного продукта) и не определяет сути системы. К этому следует добавить, что внедрение АСУ в производственные процессы должно приводить к сокращению документооборота.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Каковы основные принципы устранения действия возмущений на объект управления?
2. Приведите пример распределения функции управления в двухуровневых АСУ ТП.
3. Дайте классификацию АСУ по критерию сложности объектов управления.
4. Приведите классификацию АСУ по функционально-алгоритмическому признаку.
5. Какие существуют основные системные показатели АСУ?
6. Какими особенностями обладают системы логико-программного управления?
7. Где используются системы оптимального управления?
8. Охарактеризуйте адаптивные АСУ и системы комплексного (организационно технологического) управления.

## Глава 2

# СИСТЕМНЫЙ АНАЛИЗ ТЕХНОЛОГИЧЕСКИХ ОБЪЕКТОВ УПРАВЛЕНИЯ

### 2.1. ИНФОРМАЦИОННОЕ ОБСЛЕДОВАНИЕ И ОПИСАНИЕ ТЕХНОЛОГИЧЕСКИХ ОБЪЕКТОВ УПРАВЛЕНИЯ

Детальное исследование технологии и организации производства представляет собой один из основных этапов системного анализа и включает исследование структуры технологического процесса, технологического оборудования и его реальных характеристик, а также установление наличия и определение достаточности информации о технологических факторах и параметрах предмета труда в процессе производства, выявление связей между ними. В процессе обследования должна быть по возможности собрана вся априорная информация о технологии производства и обобщены сведения, имеющиеся о ней у эксплуатационного персонала.

Началом обследования технологического процесса является построение пооперационной схемы технологического процесса в двух вариантах: так, как он был спроектирован, и так, как он реально функционирует. Анализ этих двух схем позволяет определить, можно ли усовершенствовать технологический процесс, а также с чем связаны появившиеся в нем изменения: прогрессом в технологии и оборудовании, отсутствием требуемого оборудования или попытками разрешить возникающие трудности в реализации технологического процесса заданного качества. Эти вопросы заслуживают серьезного внимания при дальнейшем исследовании технологического процесса.

Особое же внимание необходимо уделить наличию в технологическом процессе специальных подстроечных и компенсирующих операций, операций селективной сборки. На схеме технологического процесса необходимо привести контрольные операции и указать, что на них выполняется: отбраковка, классификация по группам или измерение контролируемых параметров.

Следующий вопрос, подлежащий исследованию – изучение технологического оборудования. Прежде всего, устанавливаются паспортные реально достигнутые допуски на управляемые и технологические факторы и дают предварительную оценку их достаточности для достижения требуемых показателей технологического процесса. Кроме того, в составе технологического оборудования определяют различные

системы регулирования и управления – как встроенные в него, так и функционирующие самостоятельно. Оценка принципов регулирования и управления, положенных в основу функционирования данных систем, позволит сделать выводы о возможности повышения качества управления при замене используемых способов и методов более эффективными.

В составе технологического оборудования указывают аппаратуру регистрации информации и ее отображения (это также относится к контрольно-измерительному оборудованию) как возможные автоматизированные источники информации для будущей АСУ ТП. Для технологических процессов, особенно вероятностных, очень важно исследовать и проанализировать систему контроля технологических факторов и параметров. Поэтому анализу использованного в технологическом процессе измерительного оборудования необходимо уделить особое внимание: изучить его технические точностные характеристики, определить и оценить способы измерений технологических факторов и контролируемых параметров, проверить правильность установления контрольных точек по технологическому процессу и получить ответ на вопрос, установлены ли контрольные точки там, где нужно, или там, где можно.

Система контроля в технологическом процессе – это не только контрольные операции и измерительное оборудование. Для сложных вероятностных технологических процессов в них, как правило, предусмотрена система сбора информации о ходе процесса с использованием оперативной документации. Поэтому при исследовании определяют графики прохождения этой документации, порядок ее заполнения, обработки и хранения. Кроме того, очень важный вопрос – по каким алгоритмам получают соответствующие выводы о качестве технологического процесса и необходимых мерах по его управлению, а также как оценивается эффективность использованных в процессе методов управления. Для исследования управляемости технологического процесса можно использовать метод опроса производственного персонала, однако нужно помнить, что полученные данным способом сведения будут субъективными.

Если при ознакомлении с технологическим процессом возможно провести по нему контрольные партии (это могут быть обычные изделия, идущие на технологическом процессе в данный момент, но имеющие сопроводительные листки), то можно считать, что предварительная информация о нем получена в достаточном объеме. Стоит отметить, что проведение контрольных партий, возможно, ответит

на два важных вопроса: как влияет разброс параметров входных материалов и полуфабрикатов на параметры производимых изделий и устойчив ли технологический процесс относительно допусков на изменение технологических факторов. Анализ первых данных покажет, необходимо ли вводить систематически функционирующие входные контрольные операции, анализ вторых даст сведения об устойчивости технологического процесса. При изменении технологических факторов и качества входных материалов и полуфабрикатов в пределах установленных допусков параметры готовых изделий должны укладываться в рамки, установленные для технологического процесса.

В результате проведенного информационного обследования технологического процесса, кроме схем и описаний, должны быть получены следующие сведения:

- соответствует ли функционирующий технологический процесс спроектированному, и каковы причины несоответствий, если они обнаружены;

- соответствуют ли характеристики технологического оборудования требованиям, предъявляемым к технологическому процессу, и можно ли улучшить характеристики оборудования за счет введения или замены действующих регуляторов и систем управления новыми;

- соответствуют ли характеристики контрольного оборудования требованиям, предъявляемым к нему технологическим процессом;

- эффективна ли существующая система управления технологическим процессом и избыточна ли (или недостаточна) информация для управления им (предварительное заключение);

- предварительная оценка необходимости и целесообразности работ по дальнейшему исследованию технологического процесса.

Данные, полученные в результате информационного обследования, служат основой для построения информационного описания технологических объектов. Оно в определенной мере представляет собой информационную модель исследуемого объекта, которая может иметь три уровня иерархии: технологического перехода, технологической операции и технологического процесса.

*Уровень технологического перехода* используется в основном для описания наиболее важных элементов, обычно существенно влияющих на качество всего технологического процесса. Такие ключевые технологические переходы детально исследуют с точки зрения построения математической модели процесса (технологического перехода) с последующей оптимизацией как технологических операторов, так и построения автоматизированной системы управления.

Технологический переход можно рассматривать как технологическую микрооперацию. Для нее в зависимости от содержания процесса (обработка, сборка и т. п.) строят структурную схему, указывают число входов и выходов, определяют коэффициенты передачи. В соответствии с моделью технологической операции описываются выходные параметры предмета труда, если их можно измерить непосредственно после завершения технологического перехода. Если результаты технологического перехода можно получить после проведения одной или нескольких технологических операций в полном объеме, то это необходимо указать, дав примечание о возможности изменения этих параметров с момента завершения технологического перехода до момента измерения. Если такие изменения существуют, то желательно дать им предварительную оценку и сделать пересчет на их непосредственную величину в момент завершения технологического перехода.

Особое внимание следует обратить на статистические свойства выходных параметров и, если неизвестна функция распределения, дать интервалы возможного их изменения. В качестве входных параметров на технологическом переходе выступают параметры полуфабрикатов, основных и вспомогательных материалов, поступающих в работу. Для них тоже желательно указать область разброса контролируемых и неконтролируемых параметров. Описание технологических операторов дается в форме физических или химических воздействий, которым подвергается предмет труда на технологическом переходе (например, нагрев, травление и т. п.). Должны быть указаны номинальные значения технологических операторов и допуски на их изменения, при которых разброс входных параметров и технологических операторов не выводит технологический процесс за установленные рамки качества. Разброс входных контролируемых и неконтролируемых параметров, а также контролируемых и неконтролируемых технологических операторов – это переменные вектора возмущающих воздействий.

Информационный анализ технологического перехода включает в себя также описание технологического оборудования, инструмента и оснастки, с помощью которых совершается этот технологический переход с указанием регуляторов и систем управления, связанных с ним. Кроме того, указываются принципы и законы регулирования, использованные для управления технологическими операторами.

Информационное описание технологического перехода завершается составлением графа взаимосвязи входных параметров и тех-

нологических операторов с выходными параметрами изделия или полуфабриката, прошедшего технологический переход. На этом графе указывается перечень входных параметров предмета труда, основных и вспомогательных материалов, технологических операторов, а также выходные параметры предмета труда, прошедшего технологический переход (рис. 2.1). Стрелками обозначаются связи входных параметров с выходными (если они известны или их можно установить в соответствии с физико-химическими свойствами процесса). При этом может оказаться, что для некоторых входных (выходных) параметров нет связей. На это стоит обратить особое внимание: во-первых, эти связи могут существовать, хотя о них и неизвестно; во-вторых, они могут замыкаться в целом на технологической операции или технологическом процессе.

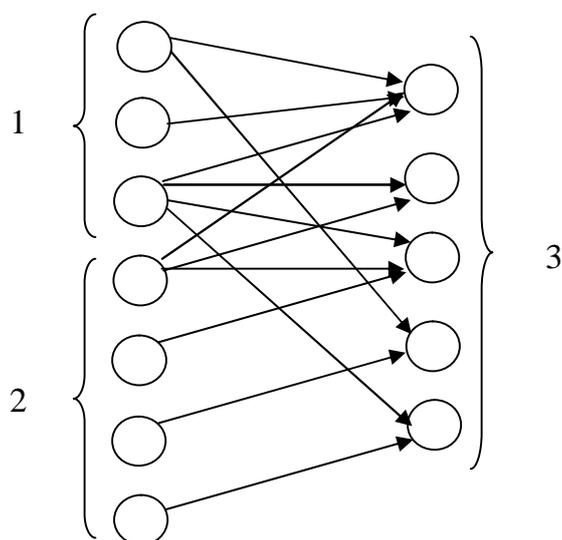


Рис. 2.1. Граф связи входных и выходных параметров:  
 1 – технологические операторы; 2 – материалы,  
 комплектующие изделия и полуфабрикаты;  
 3 – выходные параметры изделий

*Информационное описание технологической операции* включает в себя все то, что было сказано об информационном описании технологического перехода. Исключение составляют технологические операции сборки и контроля, имеющие много входов (выходов). Однако необходимо помнить, что такую форму отображения технологической операции желательно использовать лишь тогда, когда коэффициент передачи на каждом последующем технологическом переходе зависит от изменения коэффициента передачи предыдущей. Если этой зависимости нет, то структурная модель технологической операции сбор-

ки (контроля) или разборки (распределения, разагрегации) может быть представлена с указанием коэффициента передачи по каждому входу (выходу) (рис. 2.2).

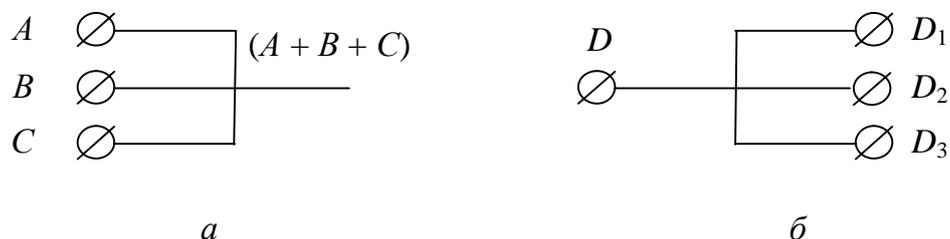


Рис. 2.2. Структурные модели для технологических операций с независимыми коэффициентами передачи:  
*a* – сборка; *б* – контроль; *A, B, C* – детали сборки;  
*D* – изделия, поступающие на контрольную операцию;  
*D<sub>1</sub>, D<sub>2</sub>, D<sub>3</sub>* – отсортированные изделия

Дополнительно к описанию технологического перехода при информационном описании технологической операции указывается:

- описание ресурсов, накапливаемых из операции в виде страховых запасов или незавершенного производства, а также причины их образования (технологические или организационные), принципы расхода и пополнения (в соответствии со структурными моделями технологических операций и регуляторов к ним);
- виды отходов (возвратные и безвозвратные) и, если известны, их параметры (контроль за отходами может оказаться эффективным методом контроля за качеством технологической операции);
- подробное описание информационных связей технологической операции с собственными системами управления;
- способы управления технологической операцией (ручное, при помощи регуляторов, микропроцессорных систем, управляющих машин) и их характеристики;
- перечень входных установок на системы управления технологической операции;
- описание входных и выходных потоков информации на технологической операции, непосредственно связанных как с управлением технологическим процессом, так и с организацией его функционирования;
- объемно-временные характеристики информационных потоков;
- способы представления и получения информации (по техническим каналам) на магнитных или бумажных носителях;

– адреса источников и приемников информации.

*Информационное описание технологического процесса* делается, как правило, в блочном разрезе. В качестве блоков могут выступать основные технологические операции, а также группы объединенных технологических операций, если среди них нет таких, которые представляют особый интерес при исследовании технологического процесса или при автоматизации управления им. Разделить технологический процесс на блоки можно разными способами, наиболее часто – по технологическому оборудованию или его группам, представляющим так называемые технологические участки (модули), на которых совершается определенный законченный комплекс (совокупность) технологических операций, начиная с контроля предметов труда и материалов на входе технологического участка и заканчивая измерением входных параметров предметов труда, прошедших обработку.

Основное внимание при построении информационной модели технологического процесса необходимо уделять описанию взаимосвязи технологических операций и, в частности, выявлению возникающих изменений в параметрах предмета труда по ходу технологического процесса из-за косвенного влияния технологических процессов на параметры изделий, формирование которых было завершено в предшествующих технологических операциях. Эти данные представляют собой вектор  $A$  (см. формулу (1.1)) для технологического процесса в целом. По этим данным так же, как для технологического перехода и технологической операции, строится граф взаимосвязи изменения параметров предмета труда по мере движения их в технологическом процессе.

В управлении исследуемым технологическим процессом может существовать система связанных между собой технологических регуляторов. В этом случае описываются их взаимосвязи, законы регулирования, порядок изменения уставок. Если взаимосвязанное управление технологическими операциями осуществляется вручную, то указываются способы получения информации и методы регулирования процесса, параметры предметов труда, материалов и комплектующих изделий, а также значения технологических операторов, при которых в действие вступает система управления.

При описании технологического процесса может выделяться группа технологических операций, в основном определяющих качество готовой продукции с целью ввести на этих операциях контрольные экспресс-партии, необходимые для определения настроенности операций перед запуском изделий в производство. Контрольные экспресс-

партии могут проводиться по определяющим технологическим операциям либо во время функционирования технологического процесса в целом для его оперативной подстройки.

При составлении информационного описания технологического процесса большое внимание необходимо уделять организации его работы и, в частности, описанию системы управления организационными процессами. Особое значение при этом придается системам оперативной отчетности и оперативного планирования. Описание системы организационного управления можно либо заимствовать из соответствующих проектных документов по АСУП или АСУ цехов, если такие системы функционируют на производстве, либо составлять по описанию соответствующих неавтоматизированных систем управления организационными процессами. Особое внимание стоит обратить на наличие в действующих организационных системах управления связей принимаемых организационных решений с параметрами технологического процесса, входных материалов и комплектующих изделий, контролируемые параметрами предметов труда.

Таким образом, завершив построение информационного описания технологического процесса, исследователь будет располагать информационными моделями трех различных уровней описания с детальными характеристиками всех входящих в него элементов и систем управления: технологических и организационных. Полезность информационных моделей будет тем больше, чем меньше в них пропущено сведений, казавшихся при обследовании несущественными или второстепенными.

## **2.2. АНАЛИЗ КАЧЕСТВА ФУНКЦИОНИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ ОБЪЕКТОВ УПРАВЛЕНИЯ**

Информационное описание технологического объекта управления позволяет получить определенные сведения о качестве технологической операции и технологического процесса с позиций совершенствования управления ими. Для этого проводится:

- оценка точности и настроенности технологических операций;
- установление взаимосвязей параметров изделий по ходу технологического процесса;
- выявление ключевых технологических операций, больше всего влияющих на весь технологический процесс;
- определение управляемости отдельных технологических операций и технологического процесса в целом.

Одним из относительно простых методов исследования качества технологических процессов в целом является *анализ доли (процента) выхода годных по всему процессу*. Такой анализ позволяет построить график изменения выхода годной продукции по данным тех контрольных операции, которые имеются на технологическом процессе. Понятно, что если контрольные операции следуют через некоторое число других операций, то свойства технологических процессов по таким данным можно изучать только для всей группы в целом. Наличие контрольных операций используется для разбиения технологического процесса на блоки.

В качестве примера приведем график для технологического процесса изготовления мезопланарных транзисторов (рис. 2.3). Стоит отметить, что он охватывает лишь часть технологического процесса, который имеет семь крупных блоков технологических операций, причем пять из них являются непосредственно операциями изготовления прибора, а две – автономными технологическими операциями, на которых обеспечивается выполнение контрольных функций (операции контроля вольтамперных характеристик смонтированных арматур и классификации изготовленных транзисторов).

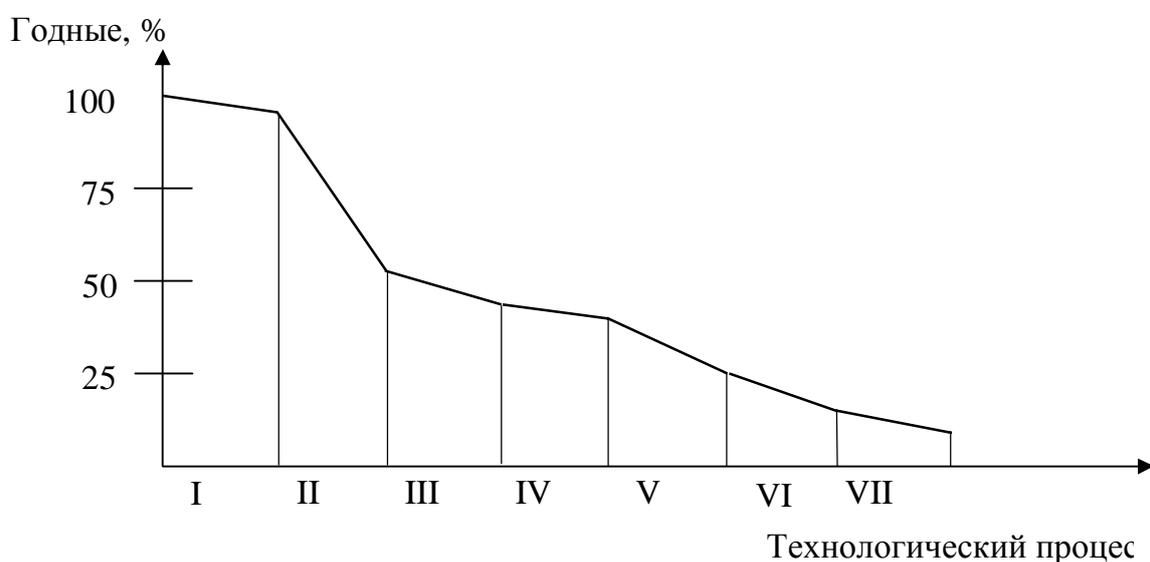


Рис. 2.3. Изменение доли выхода годных по технологическому процессу:  
*I* – диффузия; *II* – фотолитография; *III* – впавление электродов, резка пластин на блоки арматуры; *IV* – сборка блока арматуры на ножке; *V* – проверка электрических характеристик; *VI* – настройка; *VII* – классификация

Понятно, что кроме этих контрольных операций, каждая группа имеет свои собственные, на основании данных которых и определены

доли выхода годных. Однако особенность V и VII операций состоит в том, что они контролируют параметры изделий, формирующихся из многих предшествующих технологических операций и зависящих от качества материалов, используемых в технологическом процессе.

Анализ технологического процесса показывает, что наименьшая доля выхода годных приходится на технологические операции фотолитографии (II), сплавления электродов и резки пластины на блоки арматур (III). Казалось бы, в качестве технологической операции с относительно малой долей выхода годных должна быть указана и операция V. Однако она является контрольной и определяет качество изделия, зависящее от предшествующих технологических операций, а потому причину такой относительно большой отбраковки изделий нужно искать не в ней. Такие потери продукции нельзя отнести и к операции VI (монтаж арматуры на ножке), которая предшествует контрольной, поскольку на ней предусмотрен собственный контроль. Это же относится и к контрольной операции VII (классификация), так как ее результаты определяются всем предшествующим технологическим процессом.

Таким образом, анализ технологического процесса по выходу годных с технологических операций производства данного типа транзистора указывает, что существенно влияет на долю выхода годных транзисторов группа операций I–IV.

Другой метод исследования качества технологического процесса – *анализ по изменению характера распределения контролируемых параметров*. Если технологическое и измерительное оборудование позволяет при необходимости контролировать изменение определенных параметров по ходу технологического процесса, то, анализируя изменение характера распределения контролируемого параметра, можно установить технологические операции, на которых необходимо ввести дополнительный контроль и управление.

Поскольку в реальном технологическом процессе не каждая технологическая операция сопровождается контрольной (даже если это в принципе возможно), выявленное в процессе контроля отклонение от установленного характера распределения может быть следствием отклонений, возникших в технологическом процессе на предыдущих операциях. Определение качества процесса и технологической операции, повлекшей за собой выявленное отклонение, в данном случае ведется от контрольной операции, где установлено это отклонение, по направлению к началу технологического процесса. При проведении этих исследований может оказаться, что в проверяемой техноло-

гической цепочке будут встречаться операции, априори не влияющие на изменение контролируемого параметра (например, операции промежуточного хранения полуфабрикатов). Тем не менее их не следует оставлять без внимания, так как может оказаться, что именно в этих местах из-за различных неконтролируемых нарушений выполнения технологических операций или воздействия на изделие неучтенных неконтролируемых факторов как раз и возникают ненормальные отклонения контролируемых параметров.

Рассмотрим порядок анализа качества технологического процесса по изменению характера распределения контролируемого параметра на конкретном примере технологии производства полупроводниковых приборов (рис. 2.4).

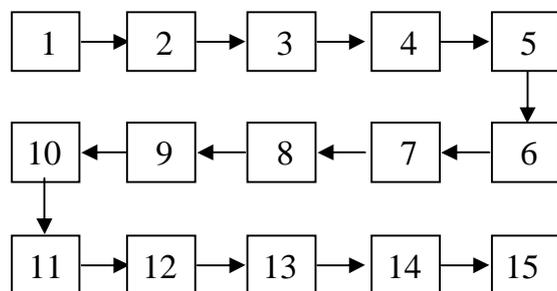


Рис. 2.4. Укрупненная блок-схема технологического процесса производства одного из типов транзисторов:

- 1 – резка слитка на пластины; 2 – шлифование пластин;
- 3 – травление пластин; 4 – диффузия; 5 – фотолитография;
- 6 – вплавление электродов; 7 – резка пластин на блоки арматур;
- 8 – приплавление блока арматур к ножке; 9 – настройка эмиттера;
- 10 – настройка базы; 11 – нанесение защитного покрытия;
- 12 – монтаж в корпусе; 13 – герметизация; 14 – окраска; 15 – испытания

По данным, полученным на участке технологических операций изготовления полупроводниковых пластин, можно построить гистограмму плоскопараллельности этих пластин после травления, при изучении которой становится очевидно, что существуют пластины с достаточно большой неплоскопараллельностью (от 9 до 13 мкм). Кроме того, в процессе построения аналогичной гистограммы на предшествующей технологической операции становится очевидно, что тенденция снижения неплоскопараллельности в левой стороне гистограммы нарушается, а потому (по сопоставлению с гистограммой после травления) следует полагать, что причина достаточно большой неплоскопараллельности пластин после травления лежит в некачественности либо выполнения шлифования, либо предшествующей

ему технологической операции. Выполненные измерения неплоскопараллельности пластин показывают, что на гистограмме имеется сильный правосторонний подъем, связанный с плохой регулировкой станка для резки слитка на пластины, откуда следует, что при автоматизации управления технологическим процессом резки пластин желательно иметь после операции резки кристалла на пластины контрольную операцию плоскопараллельности пластин.

Выборки, сделанные на трех указанных выше операциях, позволяют сделать вывод о настроенности технологических операций и точности всего технологического процесса. Настроенность технологического процесса определяется по средней  $\bar{x}$ . Для операции резки настроенность  $\bar{x}_p = 33$  мкм, для операции шлифования  $\bar{x}_ш = 7$  мкм, для операции травления  $\bar{x}_т = 5,7$  мкм. Из этих данных следует, что настроенность операции резки существенно смещена в правую сторону и более чем в пять раз превышает  $\bar{x}_т$  (последующая настройка станка для резки пластин существенно снизит  $\bar{x}_p$  и, следовательно, повысит производительность труда на участке изготовления пластин, так как длительность операций шлифования и травления существенно зависит от соотношения значений  $\bar{x}_p$ ,  $\bar{x}_ш$  и  $\bar{x}_т$ ).

По данным, полученным на участке изготовления пластин, можно оценить точность операций, входящих в технологический процесс, а также изменение точности всего процесса по мере движения пластины по операциям. В качестве оценки точности технологического процесса используется дисперсия выборки  $\sigma$ . В рассматриваемом примере она имеет следующие значения:  $\sigma_p = 25$ ,  $\sigma_ш = 4,4$  и  $\sigma_т = 3,6$  (для операций резки, шлифования и травления соответственно).

Таким образом, точность технологического процесса возрастает по мере движения пластин от начала процесса к его концу. На операцию диффузии, следующую по технологическому процессу, пластины поступают с относительно большой неплоскопараллельностью, хотя и в пределах технических норм.

Качество технологических процессов также можно оценить посредством *анализа по основным статистическим параметрам*. При проведении информационного обследования вероятностных технологических процессов могут быть построены по имеющимся данным либо получены в действующей системе управления статистические характеристики контролируемых процессов (так называемые статистические характеристики генеральной совокупности). Понятно, что они отражают некоторые интегральные характеристики технологичес-

кого процесса в прошлом. Возникают вопросы, что представляет собой исследуемый технологический процесс, установился ли он, имеет тенденцию к ухудшению или улучшению? Ответ может дать статистический анализ информации, полученной из технологического процесса, – так называемые статистические выборки [15]. Они могут быть взяты на технологическом процессе по истечении времени, за которое набиралась генеральная совокупность (для ориентира укажем, что размер выборки должен быть получен на базе измерения параметров 25–50 изделий, размер генеральной совокупности при этом должен быть больше выборки в 8–10 раз).

Для оценки качества технологического процесса берутся несколько последовательных выборок через оговоренный интервал времени (например, интервал, кратный длительности исследуемого процесса). При оценке можно использовать как изменения среднего значения контролируемого параметра в выборке  $\bar{x}_v$  по отношению к средней  $\bar{x}$  генеральной совокупности (для оценки настроенности технологического процесса), так и изменения дисперсии выборки  $\sigma_v$  по отношению к дисперсии генеральной совокупности (для оценки точности технологического процесса). Также можно использовать и другие статистические характеристики, в том числе производные от указанных. Например, в качестве производной характеристики часто используется коэффициент точности, определяемый отношением

$$\mu = 6\sigma / \delta,$$

где  $\delta$  – практическое значение зоны рассеивания.

Методику определения оценки состояния технологического процесса по результатам анализа основных статистических параметров рассмотрим на примере оценки процесса диффузии. В качестве контролируемого параметра выберем значение поверхностного сопротивления полупроводниковой пластины  $\rho$  (рис. 2.5).

Исследование диффузии проводилось на основании трех случайных выборок (табл. 2.1). При расчете коэффициента точности технологического процесса  $\mu$  зона рассеяния  $\delta = 2,64 \cdot 10^{-3}$  Ом·см (верхняя контрольная граница). Данные всех трех выборок показали, что точность технологического процесса достаточно высока: коэффициент точности  $\mu < 1$  (плохая точность технологического процесса при  $\mu > 1$ ), а  $\bar{x}$  и  $\sigma$  в данном случае существенно лучше, чем для генеральной совокупности.

**Результаты обработки параметра  $\rho$   
полупроводниковых пластин**

Параметр	Номер партии			Генеральная совокупность
	1	2	3	
$\sigma, 10^{-6} \text{ Ом}^2 \cdot \text{см}$	0,213	0,0958	0,185	0,394
$\bar{x}, 10^{-3} \text{ Ом} \cdot \text{см}$	1,73	1,29	1,97	2,125
$\mu, 10^{-3} \text{ Ом} \cdot \text{см}$	0,484	0,218	0,420	—

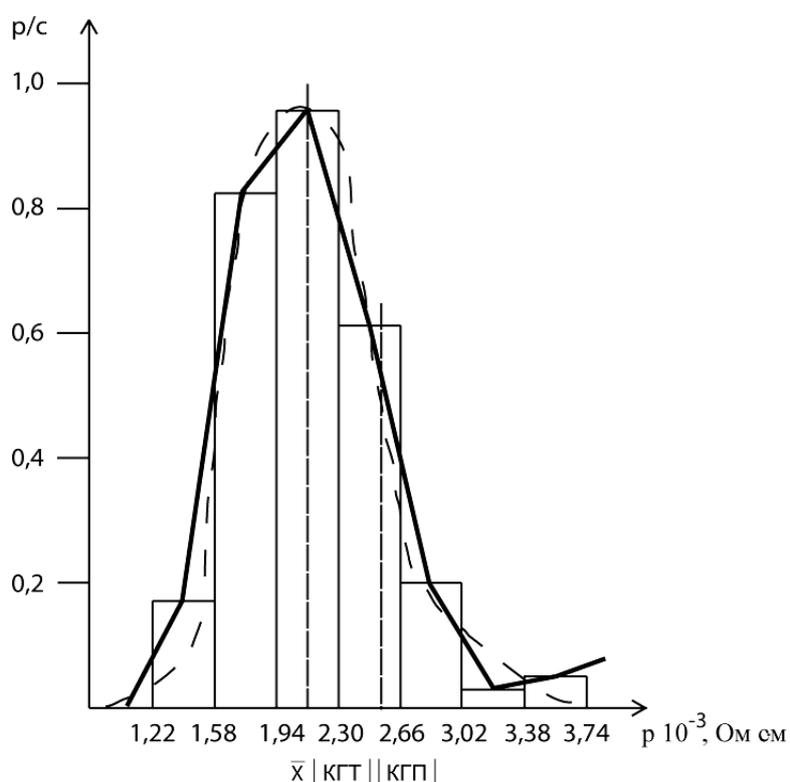


Рис. 2.5. Распределение параметра  $\rho$  на операции диффузии:  
 КГТ – теоретическая контрольная граница; КГП – практическая контрольная граница;  $\rho/c$  – плотность распределения ( $\rho$  – частость,  $\rho = n_i / N$ );  
 $c$  – размер интервала;  – эмпирическая кривая;  
 – теоретическая кривая

Казалось бы, особенного беспокойства при исследовании этого технологического процесса быть не должно: он достаточно качествен. Однако стоит обратить внимание, что уже только в трех выборках разброс  $\sigma$  превышает 2 раза (а по сравнению с генеральной совокупностью это число возрастает до 4 и более раз), тогда как уменьшение

среднего значения сопротивления  $\rho$  в выборке составляет всего 1,5 раза, а по отношению к генеральной совокупности – 1,7 раза.

Можно продолжить поиск другого параметра для контроля результатов процесса диффузии, который был бы более чувствителен к изменению режима. Например, таким параметром может быть глубина диффузии.

Рассмотрим график изменения глубины диффузионного слоя по девяти партиям пластин одного и того же слитка (рис. 2.6). Он показывает, что размах колебаний составляет 2 раза, а это больше, чем размах колебаний среднего значения  $\rho$ . Поэтому для контроля процесса этот параметр использовать можно. Следует помнить, что при выборе параметров контроля хода технологического процесса необходимо учитывать доступность автоматизации их измерения. В данном случае выбранный параметр этому требованию соответствует.

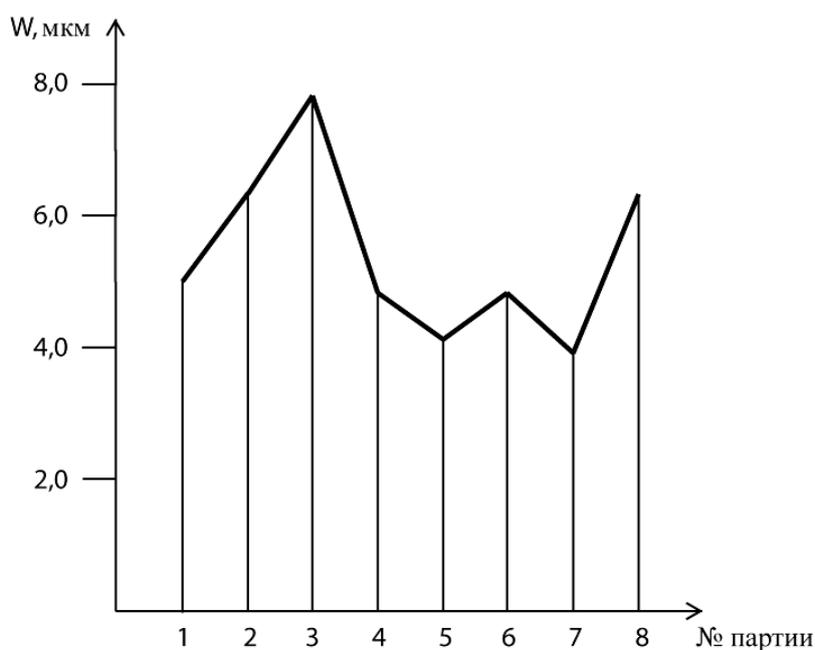


Рис. 2.6. График глубины диффузионного слоя

Несмотря на то, что при информационном обследовании невозможно получить исчерпывающую информацию о технологическом процессе (она постоянно дополняется на этапах технического и рабочего проектирования), ряд полезных заключений практически всегда можно получить, выполнив *анализ управляемости исследуемого технологического процесса*. Этот анализ, позволяет сделать выводы о соответствии рабочей области технологического процесса заданным кон-

трольным границам; возможности управления номенклатурным распределением выпускаемой продукции за счет изменения технологических факторов в пределах, допускаемых технологическим процессом; некоторых технологических факторах, которые можно использовать в дальнейшем в качестве управляющих.

Важным при анализе качества является также *определение ответственности рабочей области технологического процесса контрольным границам*. Для контролируемых параметров материалов, полуфабрикатов и изделий могут быть установлены контрольные границы, при выходе за которые состояние технологического процесса считается аварийным, т. е. в таком случае процесс требует немедленного вмешательства или даже остановки.

Контрольные границы на контролируемые показатели можно устанавливать как экспериментально, так и исходя из определенных теоретических соображений. Если ограничиваются минимальные или максимальные значения параметра, устанавливается одна граница, если ограничивается диапазон изменения контролируемой величины – две. Анализ управляемости технологического процесса по контрольным границам заключается в установлении реального положения рабочей области технологического процесса относительно контрольных границ и определении достаточности или недостаточности «запаса» в изменении контролируемых показателей.

Как известно, контролируемые границы устанавливаются исходя из определенной достигнутой доли (процента) выхода годных. Если требуемая доля определена, то можно найти и допуск на контролируемые выходные параметры изделий, а найдя их – сравнить с реальной долей выхода годных.

Для нормального закона распределения выход годных  $\beta$  равен

$$\beta = 1 - 1/\alpha^2,$$

откуда

$$\alpha = \sqrt{1/(1-\beta)}.$$

Зная значение  $\alpha$ , по статистическим таблицам можно найти коэффициент  $\beta$ , а с помощью него, в свою очередь, определить допуски на отклонения контролируемого параметра от среднего значения этого параметра для генеральной совокупности:

$$\varepsilon = \beta\sigma.$$

Так, например, если допускается выход годных 70 % (т. е. 0,7), то коэффициент  $\varepsilon$ , определяющий границы годных изделий, будет равен 1,03, а сами границы определяются формулой  $\varepsilon = \pm 1,03\sigma_n$ , где  $\sigma_n$  – стандартное отклонение для генеральной совокупности.

Таким образом, при управлении технологической операцией или технологическим процессом при выходе контролируемого параметра за эти границы процесс следует считать расстроенным.

Для рассмотренного процесса диффузии при среднем значении  $\bar{x}_n = 2,125 \cdot 10^{-3}$  Ом·см,  $\sigma = 0,394 \cdot 10^{-3}$  Ом·см и нормальном законе распределения установлена доля выхода годных 0,7 ( $\beta = 1,03$ ). Тогда контрольные границы будут определяться формулой

$$x_{в,н} = \bar{x} \pm 1,03\sigma,$$

где  $x_{в,н}$  – верхняя и нижняя контрольные границы. В данном случае  $x_{в,н} = (2,125 \cdot 10^{-3} \pm 0,046 \cdot 10^{-3})$  Ом·см.

Для реального технологического процесса установлена лишь верхняя контрольная граница (КГП) для  $\rho \geq 2,75$  Ом·см, что выше фактически установленной границы (КГТ) для  $\rho \geq 2,531$  Ом·см. Отсюда следует, во-первых, что на процессе диффузии установлена только верхняя контрольная граница, и это обеспечивает выход годных уже равным или большим 85 % (распределение симметричное, и снятие любой из границ дает увеличение площади распределения на 15 %, обеих – на 30 %). Во-вторых, так как установленная контрольная граница выше теоретической для 15 % выхода годных, то следует, что запланированный выход годных на операции более 85 % и для технологического процесса установлены достаточно жесткие рамки, хотя и существенно меньшие, чем реальный выход годных – 5–10 % (см. рис. 2.5).

Описанный выше метод установления контрольных границ для нормальных распределений применим и для несимметричных распределений, если они существенно не отличаются от нормального. Чтобы установить факт существенности или несущественности отличия характера исследуемого распределения от нормального, рассчитывают нормированные центральные моменты, а с их помощью – симметрию и эксцесс (заостренность или уплощенность) распределения. По соответствующим статическим таблицам устанавливается существенность отличия исследуемого закона распределения от нормального [4; 16; 17].

Кроме стандартных распределений типа нормального, биномиального (см., например, рис. 2.5) и других, в вероятностных технологических процессах часто встречаются асимметричные распределения (рис. 2.7).

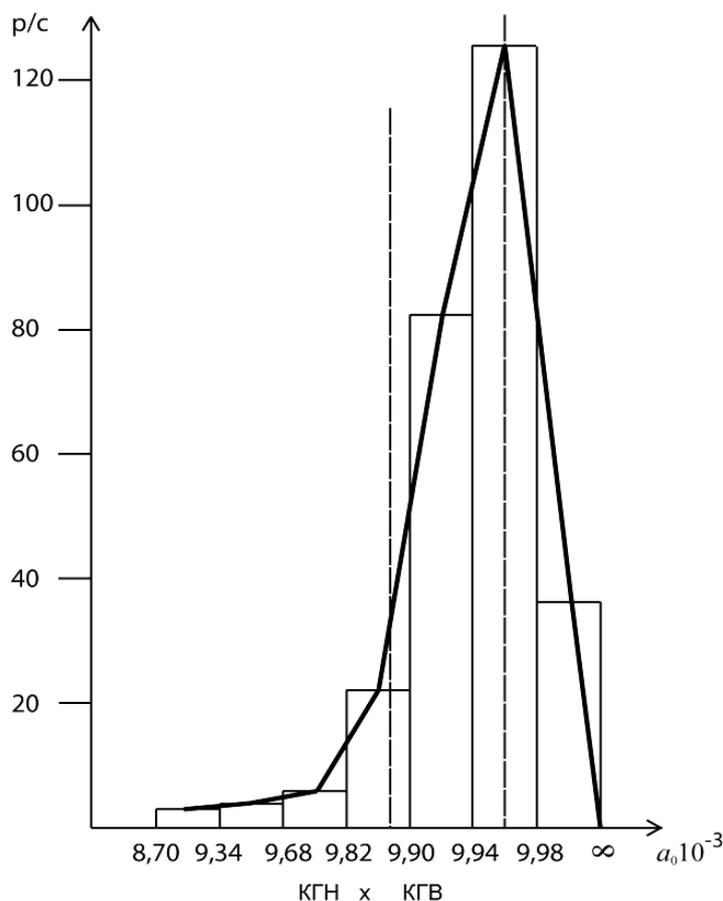


Рис. 2.7. Распределение параметра  $\alpha_0$  (интервалы не равны)  
 КГН, КГВ – нижняя и верхняя контрольные границы  
 (расчетные для 70 % выхода годных);  
 $p/c$  – плотность распределения

В ряде случаев (в том числе в последнем) асимметричные распределения могут быть приведены к нормальному подбором некоторой функции от случайной величины (например, случайная величина  $x$  заменяется преобразующей функцией  $\lg x$ ). Плотность такого распределения рассчитывается по формуле

$$f(x) = M / (x\sigma\sqrt{2\pi}) \exp\left[-(\lg x - \lg x_0) / 2\sigma^2\right],$$

где  $\lg x = y$  и  $\lg x_0 = x_0$  – коэффициенты перехода от натуральных логарифмов к десятичным;  $M = 0,4343$ .

Преобразовав все значения  $x$  в  $y$ , группируют ряд по интервалам по  $y$  и рассчитывают  $\bar{y}$  и  $D_{lg}$ :

$$\bar{y} = y_0 + \frac{C}{N} \sum_1^m n_i \frac{y - y_0}{C};$$

$$D_{lg} = \frac{C^2}{N^2} \left[ \sum_1^m n_i \left( \frac{y - y_0}{C} \right)^2 N - \sum_1^m n_i \left( \frac{y - y_0}{C} \right)^2 \right],$$

где  $C$  – константа.

После того как распределение приведено к логарифмически нормальному, рассматриваемый технологический процесс контролируется аналогично процессу, имеющему нормальное распределение контролируемого параметра путем сравнения  $\bar{y}$ ,  $D_{lg}$ ,  $\sigma_{lg}$  выборок и генеральной совокупности.

Контрольные границы для преобразованного распределения определяют исходя из заданного выхода годных так же, как это было показано выше. Экспериментально установленные границы сравнивают с теоретическими в системе преобразованных координат, т. е. экспериментальные граничные значения для контролируемых переменных пересчитывают так же, как и остальные данные с использованием уже установленной функции преобразования. Такой функцией для распределения, приведенного выше (см. рис. 2.7), является

$$y = \lg(a - x) - \lg(x - b),$$

где  $x$  – исходное значение переменной;  $y$  – преобразованное значение переменной;  $a, b$  – константы (в данном случае  $a = 1, b = 0,8$ ).

Для использования результатов анализа (например, расчетных значений контрольных границ в управлении технологическим процессом) необходим обратный пересчет полученных значений в исходные.

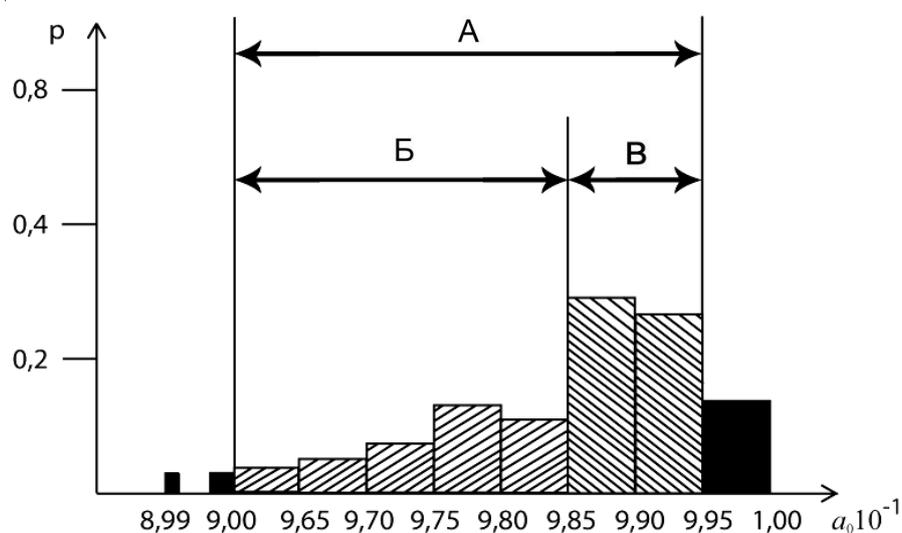
При анализе качества технологического процесса также проводится *определение возможности управления номенклатурным распределением выпускаемой продукции*. Для вероятностных технологических процессов характерен не только относительно небольшой выход годных, но и достаточно большой диапазон изменения выходных параметров изделий. Последнее приводит к необходимости классифицировать изделия по группам. Для стабильного технологического

процесса распределение годных изделий по группам сохраняется на относительно длительном отрезке времени функционирования. При исследовании технологического процесса может быть поставлен вопрос о возможности (или невозможности) управления номенклатурным распределением годных изделий по группам в связи с тем, что, во-первых, изделия более высоких квалификационных групп имеют, как правило, большую потребительную стоимость и потому желательно иметь больший их выход, а во-вторых, заказчика может интересовать в разное время разное распределение изделий по группам (например, в связи со сдвигами в структуре номенклатуры выпускаемых изделий). Таким образом, возникает вопрос о том, можно ли в исследуемом технологическом процессе целенаправленно изменять распределение годных изделий по группам. Для этих целей необходимо исследовать границы классификационных показателей и установить непротиворечивость требований относительно изменения реального технологического процесса. Последнее достигается исследованием корреляционных зависимостей между классификационными параметрами изделий.

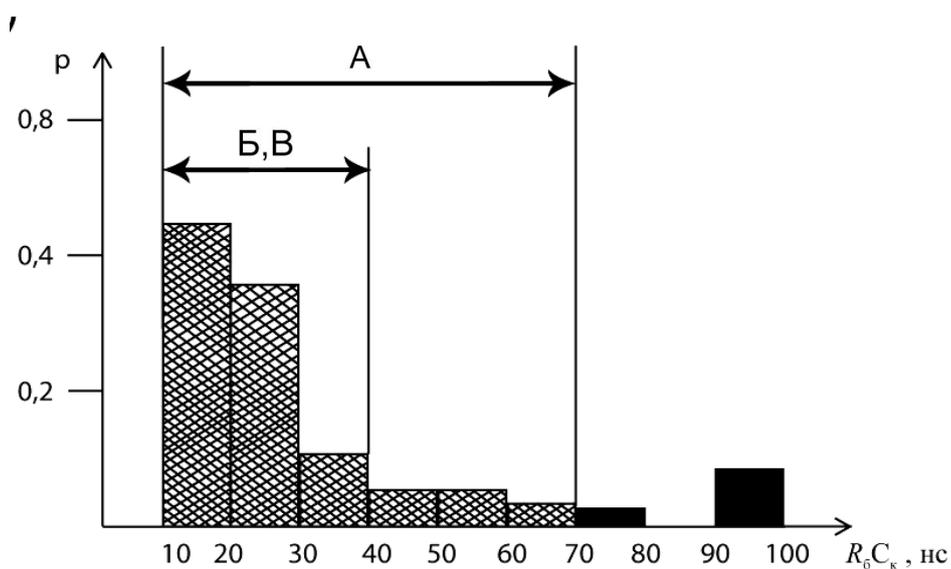
Порядок определения возможности управления номенклатурным распределением изделий рассмотрим на примере исследования принципиальной возможности повышения доли (процента) выхода высшей классификационной группы В одного из типов транзисторов. Дан график, на котором показаны границы квалификационных групп распределения транзисторов по параметрам  $\alpha_0$  и  $R_6C_k$  (рис. 2.8). Приборы группы В, пользующиеся у потребителей большим спросом, имеют достаточно большие значения коэффициента передачи по току  $\alpha_0$  и относительно небольшую постоянную времени  $R_6C_k$ . Возникает вопрос: если, предположим, удастся настроить технологический процесс так, чтобы увеличилось число приборов, попадающих в группу В по параметру  $\alpha_0$ , не приведет ли это автоматически к эквивалентному росту числа приборов, не попадающих в группу В в связи с увеличением  $R_6C_k$ ? Чтобы ответить на этот вопрос, необходимо определить, какова реальная связь в данном технологическом процессе между параметрами  $\alpha_0$  и  $R_6C_k$ .

Посмотрим, как распределяются параметры  $\alpha_0$  и  $R_6C_k$  для партии транзисторов на корреляционном поле (рис. 2.9), после чего трансформируем его, построив корреляционную зависимость  $\alpha_0$  и  $R_6C_k$  (рис. 2.10). Из нее следует, что интересующая нас зависимость положительна, т. е. в принципе рост  $\alpha_0$  должен вести к росту  $R_6C_k$ . Однако

для установленной группам Б и В верхней классификационной границы значение  $R_6C_k = 40$  не должно увеличить верхнюю классификационную границу по  $\alpha_0 = 0,992$ , что выше установленной классификационной границы для группы В  $\alpha_0 = 0,991$ . Таким образом, на данном технологическом процессе возможно некоторое увеличение выпуска транзистора группы В за счет смещения центра распределения транзисторов по  $\alpha_0$  в сторону больших значений  $\alpha_0$ .



*a*



*б*

Рис. 2.8. Распределение приборов по группам А, Б, В ( $p$  – частость):  
*a* – распределение по  $\alpha_0$ ; *б* – распределение по  $R_6C_k$

Если известны аналитические или экспериментально определенные зависимости параметров изделий от технологических факторов, то, исследуя их, можно выявить наиболее эффективные управляющие технологические операторы. Это называется *определением управляющих факторов по математическим моделям технологических процессов*.

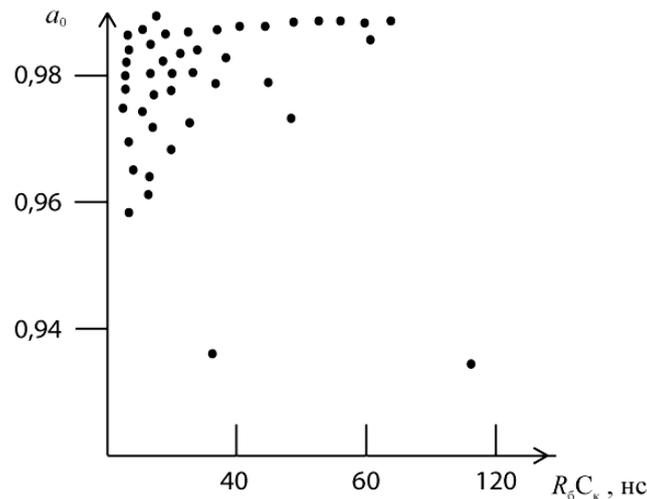


Рис. 2.9. Корреляционное поле распределения параметров  $\alpha_0$  и  $R_6 C_k$

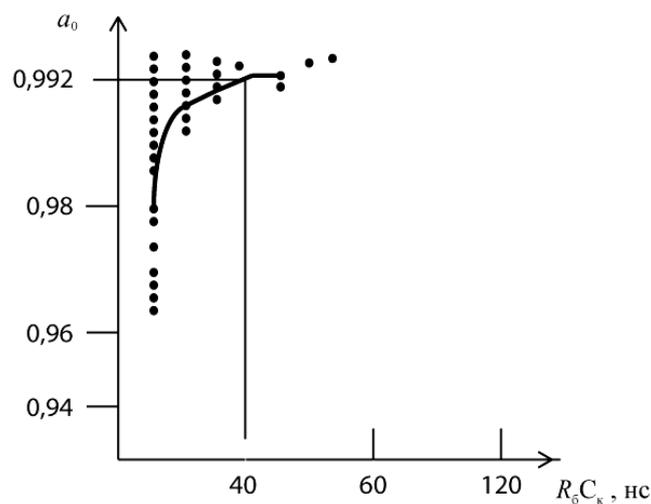


Рис. 2.10. Трансформированное корреляционное поле с построенной корреляционной зависимостью  $\alpha_0$  и  $R_6 C_k$

Пусть интересующий нас показатель (параметр) является функцией многих переменных  $y_i = f(x_1, x_2, \dots, x_j)$ . Изменение любого аргумента ведет к изменению всей величины:

$$\Delta y_i = f(x_1 + \Delta x_1; x_2 + \Delta x_2; \dots, x_j + \Delta x_j) - f(x_1, x_2, \dots, x_j).$$

Переходя к производным и полагая, что членами, содержащими  $\Delta x_j$  второго и более порядка, можно пренебречь, получаем

$$\Delta y_i = \sum_{j=1}^n \frac{\partial y_i}{\partial x_j} \Delta x_j. \quad (2.1)$$

Чтобы можно было сравнить влияние различных технологических операторов на контролируемый технологический фактор, удобно перейти к безразмерной оценке:

$$\Delta y_i / y_j = \sum_{j=1}^n \left( \frac{\partial y_i}{\partial x_j} \right) (x_j / y_i) (\Delta x_j / x_j). \quad (2.2)$$

Выражение (2.1) дает максимальное изменение функции  $y_i$  относительно изменения отдельных технологических операторов  $x_j$ . Обычно функции  $y$  – это функции вида

$$y = f \left[ z_p (x_j) \right]. \quad (2.3)$$

С учетом выражения (2.3) уравнение (2.2) можно записать следующим образом:

$$\begin{aligned} \Delta y_i / y = \sum_{j=1}^n \left[ \left( \frac{\partial y_i}{\partial z_p} \right) (z_p / y_i) \right] \times \\ \times \left[ \left( \frac{\partial z_p}{\partial x_j} \right) (x_j / z_p) \right] (\Delta x_j / x_j) \end{aligned} \quad (2.4)$$

Формулы (2.2) и (2.4) определяют влияние отдельных технологических операторов на контролируемый показатель, а также значимость вклада изменения данного технологического оператора в изменение контролируемого, а поэтому могут быть названы коэффициентами значимости.

Покажем процесс вычисления коэффициента значимости для коэффициента  $b_{\varpi}$ ,  $N_g$ , определяющего влияние поверхностной концентрации на толщину пассивной базы  $\varpi$ .

Для данного процесса диффузии имеем исходное уравнение

$$N_a = N_g \operatorname{erfc} \left[ \varpi / \left( 2\sqrt{D_n t_{\partial n}} \right) \right], \quad (2.5)$$

где  $N_g$  – поверхностная концентрация;  $t_{\partial n}$  – время диффузии.

В формулу (2.5) температура диффузии  $T_{\partial n}$  входит в неявном виде:

$$D_n = D_n(T_{\partial n}).$$

Чтобы определить  $\partial \varpi / \partial N_g$ , введем в уравнение (2.5) обозначения  $z = \varpi / (2\sqrt{D_n t_{\partial n}})$ , запишем  $N_g = N_a / \operatorname{erfc} z$ . Отсюда после дифференцирования получаем:

$$\partial z / \partial N_g = (\sqrt{\pi} / 2) (N_a / N_g^2) e^{z^2}.$$

Учитывая, что  $\Delta z / z = \Delta \varpi / \varpi$ , найдем коэффициент значимости влияния поверхностной концентрации на толщину пассивной базы:

$$b_{\varpi, N_g} = (\partial z / \partial N_g) (N_g / z) = \left( \sqrt{\pi} N_a e^{z^2} \right) / (2 N_g z). \quad (2.6)$$

Можно определить и другие коэффициенты значимости для технологических операторов процесса диффузии и, исследуя их совокупность, проранжировать технологические операторы по степени их влияния на технологический процесс диффузии, а именно на толщину пассивной базы. Кроме того, выражения вида (2.6) могут быть использованы непосредственно для управления технологическим процессом.

В результате анализа качества технологических процессов, может быть обнаружено множество недостатков. Чтобы повысить качество, а также управляемость процессов, можно усовершенствовать их структуру. В соответствии со свойствами технологических операций формируемые из них технологические процессы в большей или меньшей степени отражают эти операции, что особенно заметно в последовательных технологических цепочках, в которых содержатся технологические процессы вероятностного характера, приводящие к появлению значительных разбросов выходных параметров изделий. Учитывая, что не существует абсолютно детерминированных технологических процессов, при построении любых из них достаточной длины крайне важно уменьшить влияние вероятностных изменений промежуточных процессов на изменение выходных параметров изделия. Один из способов сделать это – *введение компенсационных технологических операций*.

В качестве примера рассмотрим график изменения  $y(t)$  для технологического процесса из двух технологических операций, из которых управляемая – только вторая (рис. 2.11).

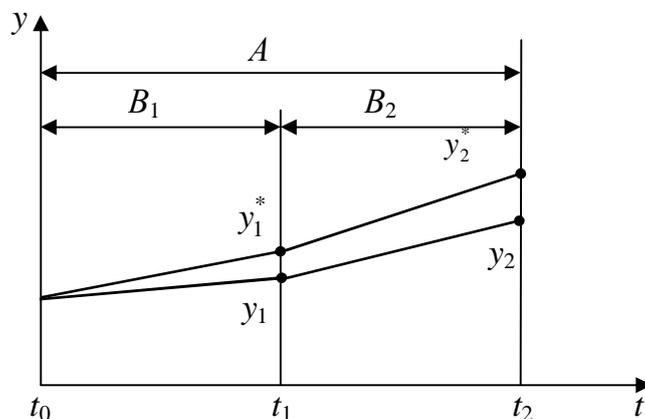


Рис. 2.11. Изменение контролируемого параметра  $y$  по ходу технологического процесса, состоящего из двух технологических операций

Управление технологическим процессом в случае, если обе операции абсолютно детерминированные, имеет вид

$$y(t_1 + t_2) = y(t_1) + y(t_2) = Ay(t_0) + Au(t_0) + By(t_1) + Bu(t_1). \quad (2.7)$$

В случае, если переменные  $u$  и  $y$  не связаны между собой в технологическом процессе (т. е.  $Au(t_0)$  и  $Bu(t_1)$  не действуют на  $y(t)$ ), выражение (2.7) можно упростить:

$$y = (t_1 + t_2) = Ay(t_0) + By(t_1).$$

Однако реально действующие операторы  $A$  и  $B$  отличаются от планируемых:

$$y^* = (t_1 + t_2) = A^* y(t_0) + B^* y^*(t_1),$$

и из-за неточности действующего оператора  $A$  разброс выходных параметров растет не только за счет неточности выдерживания самих операторов  $A$  и  $B$ , но и за счет того, что объект воздействия уже имеет отклонение реальных значений выходных параметров от заданных и возникших на предшествующих технологических операциях (на это указывает второй член уравнения). Полученная в результате действия технологического процесса разность  $y(t_1 + t_2) - y^*(t_1 + t_2) = \Delta y(t_1 + t_2)$

может оказаться относительно большой и не удовлетворять производство. Из достаточно простых соображений можно показать, что если в технологическом процессе за этими двумя операциями ввести третью (компенсирующую), такую, что

$$B_{\kappa}y(t_2) = y(t_3) = -\Delta y(t_1 + t_2),$$

то получим следующее уравнение:

$$y(t_1 + t_2 + t_3) = y(t_1 + t_2) + \Delta y(t_1 + t_2) - \Delta y(t_1 + t_2) = y(t_1 + t_2).$$

Из сказанного следует, что даже для абсолютно детерминированных технологических процессов при отсутствии по каким-либо причинам управления хотя бы на одной технологической операции может оказаться желательным введение дополнительной компенсирующей (и, конечно, управляемой) операции. При выборе такой операции важно, какой знак имеет отклонение  $\Delta y(t_1 + t_2)$ . Если оно меньше нуля, то в качестве компенсирующей можно повторно использовать предшествующую технологическую операцию. Если же отклонение больше нуля, то предшествующую технологическую операцию нельзя использовать в качестве компенсирующей, так как ее повторение ведет к еще большему возрастанию  $\Delta y$ ; в этом случае должна быть предложена специально сконструированная технологическая операция, имеющая оператор  $B_{\kappa}$ , отрицательный по отношению к действию оператора  $B_2$  (т. е. оказывающий на предмет труда обратное действие). Так, например, если бы оператор  $B_2$  отвечал за операцию осаждения какого-либо материала, то оператор  $B_{\kappa}$  представлял бы собой операцию травления осажденного материала. Таким образом, за управляемой технологической операцией, результаты действия которой, а также действия находящейся перед ней частью технологического процесса, известны, можно при необходимости ввести компенсирующую операцию (рис. 2.12): либо одну (рис. 2.12, а), либо несколько (например, при наличии в технологическом процессе как отрицательных, так и положительных отклонений в качестве компенсирующих необходимы две технологические операции (рис. 2.12, б)).

Для чисто вероятностных технологических операций их результат является случайной функцией, и выходные параметры предмета труда оказываются распределенными по некоторому закону. Поэтому в технологическом процессе нельзя достичь полной компенсации

отклонений  $\Delta y(t_1 + t_2)$  при использовании детерминированных технологических операций, хотя и можно увеличить число изделий, укладываемых в допуски.

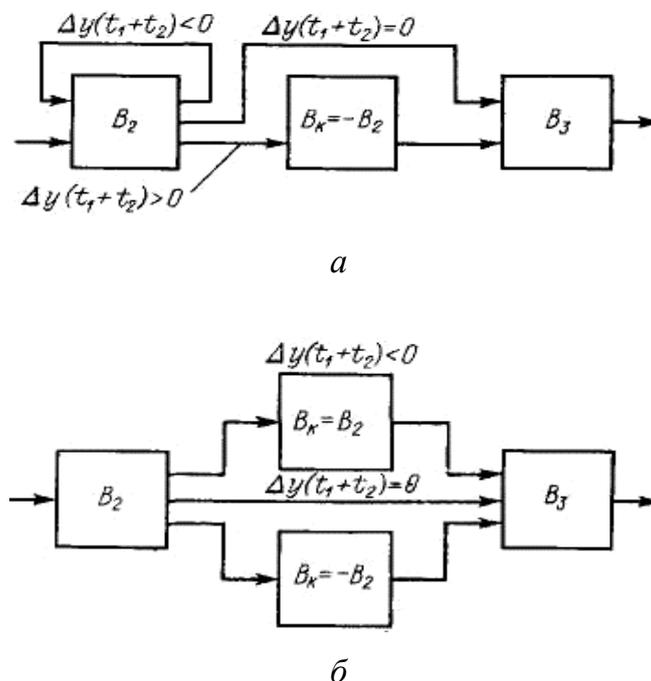


Рис. 2.12. Структурная схема технологического процесса с компенсирующими технологическими операциями:  
*a* – при использовании предшествующей технологической операции в качестве одной из компенсирующих;  
*б* – при использовании дополнительных технологических операций в качестве компенсирующих

Примером управляемой компенсирующей технологической операции с  $B_k = -B_2$  может быть уже рассмотренная нами в технологическом процессе изготовления полупроводниковых пластин операция травления (см. рис. 2.3). Операцию шлифования можно использовать не только для придания поверхностям шлифуемых пластин соответствующего качества, но и для уменьшения толщины самих пластин. И тогда ее можно рассматривать как компенсационную с  $B_k = -B_2$ .

В качестве компенсационной технологической операции, аналогичной основной ( $B_k = B_2$ ), можно рассматривать также, например, операцию гальванопокрытий. В случае, если толщина полученного гальванопокрытия оказывается меньше необходимой, то ее можно увеличить за счет повторения операции с использованием основного или аналогичного ему дополнительного оборудования.

Повысить эффективность вероятностных технологических процессов можно также посредством *введения селективных сборочных операций*. Сущность селективной сборки заключается в том, что если изделие состоит из некоторого числа взаимосвязанных деталей (элементов), а их параметры распределены по одинаковым (или близким друг к другу) законам, то эти детали можно подобрать так, что параметры готового изделия будут укладываться в требуемый допуск. Например, если

$$y(t) = y_1^*(t) + y_2^*(t);$$

$$\left. \begin{aligned} y_1^*(t) &= y_1(t) + \Delta y_1(t); \\ y_2^*(t) &= y_2(t) - \Delta y_2(t); \\ \Delta y_1(t) &= \Delta y_2(t), \end{aligned} \right\}$$

то в совокупности получим:

$$y(t) = y_1(t) + y_2(t).$$

В данном случае выходные характеристики изделия в точности соответствуют заданным.

Операция селективной сборки может быть введена в технологический процесс искусственно, если это, конечно, допускается характером технологического процесса и свойствами обрабатываемого предмета. Так, одно сопротивление можно заменить двумя последовательно включенными сопротивлениями, изготовленными по одной и той же технологии. Понятно, что, подбирая сопротивления по парам, можно добиться меньшего разброса номинала пары, чем при изготовлении одного сопротивления.

В качестве селективных могут выступать и операции подборки предметов труда в определенные группы для дальнейшего прохождения технологического процесса – в случае, когда все множество предметов имеет слишком большие разбросы параметров. Если для каждой такой группы можно установить собственные значения технологических операторов, повышающих выход годных изделий, то введение технологической операции селективной подборки в группы оправданно. В качестве примера можно назвать подборки полупроводниковых пластин в группы по значениям поверхностного сопротивления или по глубине диффузии.

Производительность операции селективной сборки определяют следующим образом. Пусть даны две группы деталей, из которых

необходимо собрать изделие. В каждое изделие входят две детали: одна – из первой группы, вторая – из второй. Введем условие, что к каждой детали из первой группы может подойти только одна деталь из второй группы. Число деталей в первой и во второй группах одинаково. Распределение вероятностей выбора детали из первой и второй групп – равномерное по интервалу значений параметров.

Чтобы собрать изделие, к детали из первой группы подбирают наугад деталь из второй группы. При этом проверяются параметры случайно выбранной детали. Если они по условиям сборки соответствуют параметрам первой детали, то изделие собирается. В противном случае из второй группы выбирают другую деталь (не подошедшую деталь на место не возвращают), и все операции повторяются. При сборке годного изделия все ранее извлеченные из второй группы детали, не использованные в сборке, возвращаются в нее. Функционирование технологической операции заканчивается, когда для каждой детали из первой группы будет подобрана деталь из второй, т. е. все изделия будут собраны.

Поскольку вероятность сборки за один шаг обратно пропорциональна числу изделий второй группы, а вероятность извлечения любого из них вычисляется по выражению

$$p_i = 1 / N_i, \quad (2.8)$$

где  $N_i$  – число изделий, находящихся в группе, в общем случае имеем:

$$p_n = \frac{1}{[N - (n - 1)]}.$$

Здесь  $N$  – начальное число изделий во второй группе (в предположении по условию, что число деталей в первой группе  $N_1$  равно числу деталей во второй группе  $N_2$ );  $n$  – число извлеченных из второй группы деталей.

Формула (2.8) определяет вероятность сборки изделия в первом цикле на каждом шаге. Видно, что вероятность растет с ростом числа выбранных деталей. В зависимости от цикла сборки изделия (сборка первого изделия, второго изделия и т. д.) ее вероятность на каждом шаге каждого цикла выражается зависимостью

$$p_{n\gamma} = \frac{1}{[N - (\gamma - 1)] - (n - 1)},$$

где  $\gamma$  – порядковый номер цикла или собираемого изделия.

Полная сборка всех изделий будет произведена за  $I$  шагов:

$$I = N^2 - N(N-1)/2 = N(N+1)/2. \quad (2.9)$$

Если изделие состоит из нескольких (больше одной) деталей, подлежащих селективной сборке, то формула (2.9) принимает вид

$$I = m \left[ N(N+1)/2 \right], \quad (2.10)$$

где  $m$  – число деталей, подлежащих подборке для изделия.

Из выражения (2.10) следует, что число элементарных операций подбора растет пропорционально числу деталей, требующих подбора (для равновероятностного характера распределения параметров подбираемых деталей). Если селективная сборка осуществляется наиболее удачно, с первого шага, то получим:

$$I = mN. \quad (2.11)$$

Формулы (2.10) и (2.11) дают крайние оценки затрат на подборку деталей: оптимистическую (формула (2.10)) и пессимистическую (формула (2.11)). Средняя из этих двух оценок равна

$$I = m \left[ N(N+3)/4 \right]. \quad (2.12)$$

Если в технологическом процессе не принято соответствующих мер по уменьшению разброса параметров или они оказываются недостаточными, то на «длинных» технологических процессах могут находиться изделия, параметры которых заведомо выйдут за допуски, и их движение по технологическому процессу лишь «засорит» его и приведет к росту непроизводительных затрат. Исключить движение по технологическому процессу изделий, которые выйдут заведомо негодными, можно посредством *введения дополнительных отбраковочных операций*.

Принцип определения местоположения операции отбраковки рассмотрим на примере технологического процесса, у которого для каждой технологической операции действие случайных факторов приводит к распределению параметров по нормальному закону. Для аддитивных технологических процессов с нормальным распределением параметров имеем:

$$\sigma_{\Sigma}^2 = \sum_i \sigma_i^2,$$

где  $\sigma_{\Sigma}^2$  – дисперсия контролируемого параметра по завершению технологического процесса;  $\sigma_i^2$  – дисперсия контролируемого параметра из  $i$ -й технологической операции.

Условием необходимости введения отбраковочной операции является отношение

$$\sigma_{\Sigma}^2 > \sigma_{\text{пр}}^2,$$

где  $\sigma_{\text{пр}}^2$  – предельно допустимое значение дисперсии контролируемого параметра по завершении технологического процесса.

Конкретное положение отбраковочной операции в технологическом процессе определяется (в зависимости от удобства ее реализации и наличия контрольно-измерительной аппаратуры) возможностью доступа к местам изделия, в котором должны осуществляться измерения. Естественно, что надо учитывать экономические соображения целесообразности введения и функционирования отбраковочной операции.

Введение отбраковочной операции в технологический процесс эквивалентно замене суммы дисперсий предшествующих технологических операций на некоторую новую такую, что выполняется неравенство

$$\sigma_1^2 + \sigma_2^2 > \sigma_{1,2}^2,$$

и тогда получим следующее:

$$\sigma_{\Sigma}^2 \leq \sigma_{\text{пр}}^2, \quad (2.13)$$

что и требовалось. Может, однако, оказаться, что в целом для технологического процесса введения одной отбраковочной операции окажется недостаточным, и неравенство (2.13) удовлетворяться не будет. В таком случае в технологический процесс надо ввести еще одну или несколько отбраковочных операций.

Одним из вариантов отбраковочных операций, искусственно вводимых в технологический процесс производства изделий электронной техники, являются отбраковочные операции по внешнему виду. Благодаря им снимаются с технологического процесса изделия, которые, дойдя до конца технологического процесса, были бы квалифицированы как брак.

Чтобы определить число контрольных операций, необходимо учесть такие основные экономические факторы, как затраты на создание

и функционирование отбраковочных операций. Поэтому их число в технологическом процессе ограничивается не только самим процессом, но и допустимой разностью между убытками, понесенными предприятием в результате обработки изделий, которые уйдут в брак, и затратами на дополнительные контрольные операции [18].

### **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Перечислите этапы информационного обследования технологических операций и процессов.
2. Какие существуют методы информационного описания технологических операций и процессов?
3. Приведите примеры структурных моделей технологических операций и процессов.
4. Как классифицируются методы анализа качества технологических процессов?
5. Каким образом можно определить управляющие факторы по математическим моделям технологических процессов?

## **МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОЕКТИРОВАНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ**

### **3.1. АЛГОРИТМИЗАЦИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ**

Алгоритмизация технологических процессов представляет собой разработку математического описания поведения системы «технологический процесс – автоматизированная система управления технологическим процессом (АСУ ТП)» на нескольких уровнях формализации. Она является одной из основных составляющих процесса проектирования АСУ ТП на всех этапах. Автоматизация включает:

- изучение технологического процесса и факторов, определяющих его поведение;
- постановку задачи автоматизированного управления процессом;
- разработку математической модели, алгоритма управления процессом и программ применительно к конкретной управляющей ЭВМ [19].

Результаты выполнения этих работ представляют в виде технических документов, основными из которых служат: математическое описание (математическая модель) процесса (объекта) управления; блок-схема алгоритма управления; алгоритм в общем виде, определяющий последовательность математических и логических операций решения задачи; машинный алгоритм, учитывающий особенности конкретной управляющей ЭВМ; программы на алгоритмическом языке в автокоде или условных адресах; рабочие программы в кодах машины с реальными адресами и описание программ.

Для каждого из документов характерна определенная степень полноты математического описания системы «технологический процесс – АСУ ТП» с соответствующим уровнем формализации. Назовем комплект всех перечисленных документов математическим обеспечением АСУ ТП. В нем отражается математический аспект функционирования системы. Его использование помогает разработчикам создавать и эффективно эксплуатировать систему.

Трудоемкость и минимальные сроки создания математического описания, а следовательно, и создание системы в целом, главным образом зависят от степени разработки методов прикладной математики

применительно к управлению технологическими процессами, а также развитости программного обеспечения управляющих ЭВМ, на базе которых строятся АСУ ТП.

Введем термин «математическое обеспечение проектирования АСУ ТП», включив в него все методы, приемы и стандартные алгоритмы, облегчающие и ускоряющие проектирование АСУ ТП (методы исследования поведения технологических процессов и построения математических моделей; приемы структурной теории алгоритмов; программное обеспечение для автоматизации программирования, стандартное программное обеспечение, используемое как составная часть программ управления).

В составе математического обеспечения АСУ ТП большое место занимают алгоритмы и программы, связанные с непосредственным управлением объектом. Используемый при этом математический аппарат существенно зависит от класса АСУ ТП. Для систем логико-программного управления он в основном ограничивается логико-арифметическими и алгебраическими операциями, а в системах оптимального управления технологическими процессами и комплексных системах управления используется практически весь аппарат прикладной математики.

Среди алгоритмов первичной обработки информации, получаемой с датчиков технологического процесса, как типовую часть следует выделить *алгоритмы реализации программных фильтров*. Среди них наиболее распространены статистические программные фильтры для линейно статических и динамических объектов с гауссовским шумом.

Благодаря наличию *алгоритмов блока анализатора входа* в системе управления осуществляется статистический анализ стационарной и нестационарных частей сигнала. При этом используются в основном методы корреляционного анализа.

В *алгоритмах реализации модели процесса*, как правило, выделяются программные блоки реализации отдельно детерминированной и стохастической частей модели. При построении математических моделей используются основные разделы прикладной математики, причем наиболее широко – корреляционный и регрессионный анализ и методы математического планирования эксперимента.

На основании результатов реализации *алгоритмов блока перспективы* в системе управления формируется план управления технологическим процессом. Алгоритмы данного блока подразделяются на две группы: 1) алгоритмы, на основании которых вырабатывается

прогноз поведения управляемого объекта; 2) алгоритмы, реализующие оптимальный регламент ведения процесса управления.

Определение совместимости прогнозируемого состояния технологического процесса и оптимального регламента его ведения является основным результатом реализации *алгоритмов оценки состояния* системы «технологический процесс – АСУ ТП». Математический аппарат, на котором базируются алгоритмы получения обобщенной оценки состояния системы, как правило, ограничивается логико-арифметическими операциями.

*Алгоритмы выработки управляющих воздействий* обеспечивают формирование процесса оптимального управления, выработку управляющих команд, преобразование их в сигналы управления и выдачу в определенной временной последовательности. Математический аппарат, на котором базируются алгоритмы выработки управляющих воздействий, может включать в себя методы динамического программирования и вариационного исчисления или ограничиваться логико-арифметическими операциями.

*Алгоритмы блока самообучения* предназначены для ведения программы эксперимента, анализа полученных результатов и определения соответствующих параметров математической модели технологического процесса. Для реализации данных алгоритмов ведения эксперимента применяются логико-арифметические операции, алгоритмов анализа результатов эксперимента – корреляционный анализ, математический аппарат методов оценки состояния (методы наименьших квадратов, максимального правдоподобия и др.), алгоритмов коррекции модели процесса – логико-арифметические операции.

В связи с тем, что основой для построения алгоритмов различных блоков могут служить одни и те же математические методы, при дальнейшем изложении материала они объединены в две группы: методы построения математических моделей технологических процессов; методы оценки и оптимизации состояния объекта управления.

### **3.2. МЕТОДЫ ПОСТРОЕНИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ**

Методы построения математических моделей весьма разнообразны: от использования физических закономерностей протекания технологических процессов, экспертно модифицированных с помощью эмпирических оценок, до универсальных методов группового учета

аргументов (МГУА), полностью ориентированных на использование опытных данных по технологическому процессу и применение ЭВМ как интеллектуального инструмента.

В настоящее время большинство математических моделей действующих АСУ ТП построено с помощью методов, лежащих между этими двумя границами. Они характеризуются тем, что структуру математической модели определяет исследователь, а коэффициенты модели находят методами корреляционного и регрессионного анализа, математического эксперимента, теории чувствительности. Для этой же цели могут быть использованы и алгоритмы МГУА.

Причинно-следственные связи в новых моделях отображаются в виде полиномов различной степени сложности и не могут быть изменены без вмешательства проектировщика. Использование МГУА позволяет при необходимости передать ЭВМ и функции автоматического определения вида и структуры математической модели. Значимость технологических операторов в математических моделях определяется непосредственно из самой модели (разумеется, при определенных коэффициентах), экспертно (методами ранжирования по результатам обследования технологического процесса) или априорно. Для относительно простых объектов управления математическая модель технологического процесса реализуется косвенно, в выборе соответствующего закона управления (регулирования).

Коэффициенты настройки такой системы управления отображают и характеристики как объекта управления, так и самой системы. Область их нахождения обычно определяется теоретически на основании известных динамических характеристик объекта и системы управления и уточняется в процессе ввода АСУ ТП в эксплуатацию и при самой эксплуатации.

Кроме распространенных оценок, выполняемых определением разности или отношений заданных и достигнутых показателей, могут быть использованы оценки, полученные методом наименьших квадратов и его разновидностями (интегральные и дифференциальные оценки), а также методом максимального правдоподобия (в качестве интегральной оценки для объектов вероятностного характера, для сильно зашумленных объектов). Для оценки состояния вероятностных процессов, в основном с целью прогнозировать будущее состояние, можно использовать байесовые оценки.

Полученные различными методами математические модели предназначены для оптимизации технологических процессов. Они могут осуществляться с помощью хорошо известных методов: вариаци-

ционного исчисления и динамического программирования (для управления объектами со сложными динамическими характеристиками), линейного программирования и различных градиентных методов (для технологических процессов с большими постоянными времени). Комплекс алгоритмов МГУА реализует задачи оптимизации в широком спектре критериев МГУА. В системах управления, использующих стандартные задачи управления, для определения коэффициентов оптимальной настройки в процессе ввода АСУ ТП и ее эксплуатации применяются различные адаптационные алгоритмы.

Анализ технологических процессов показывает, что выходная переменная имеет сложную зависимость от параметров и времени [20]. При этом детерминированные факторы определяют значение и характер изменения математического ожидания выходной переменной, тогда как неуправляемые и неконтролируемые факторы – значения и характер случайных отклонений выходной переменной от значения математического ожидания.

Если говорить о стохастических моделях технологических и организационных процессов, то здесь важно иметь в виду, что неупорядоченность производства приводит его математическую модель

$$p(t) = p_{\text{оп}}(t) + p_{\text{т}}(t) \quad (3.1)$$

к вероятностному виду

$$p(t) = Mp_{\text{оп}}(t) + Mp_{\text{т}}(t) + p_{\text{оп}}(t) + p_{\text{т}}(t), \quad (3.2)$$

где  $p_{\text{оп}}(t)$  и  $p_{\text{т}}(t)$  – компоненты характеристики, значения которых определяются организационными и технологическими процессами;  $t$  – некоторая независимая переменная;  $Mp_{\text{оп}}(t)$  и  $Mp_{\text{т}}(t)$  – компоненты математического ожидания значения данного параметра, определяемые организационными и технологическими процессами.

В зависимости от выбранной модели объекта управления системы управления подразделяются на три класса:

1) технологические:

$$p_{\text{т}}(t) = Mp_{\text{т}}(t) + p_{\text{т}}(t);$$

2) организационные:

$$p_{\text{оп}}(t) = Mp_{\text{оп}}(t) + p_{\text{т}}(t);$$

3) комплексные.

При создании системы управления любого класса предварительно необходимо найти хотя бы одну из функций  $Mp_T(t)$  или  $Mp_{op}(t)$ . Для этого можно использовать два способа: априорный, когда вид зависимости  $Mp_T(t)$  или  $Mp_{op}(t)$  выбирают исходя из некоторых теоретических предпосылок об исследуемом объекте, или апостериорный, когда  $Mp_T(t)$  или  $Mp_{op}(t)$  определяют экспериментально. В любом случае предполагается, что для исследуемого объекта выполняется соотношение

$$Mp(t) = Mp_{op}(t) + Mp_T(t). \quad (3.3)$$

Для технологических процессов, не имеющих точного детерминированного математического описания, задача построения математической модели технологического процесса в основном сводится к нахождению функции  $Mp_T(t)$  и реже  $p_T(t)$ .

**Методы корреляционного анализа.** Одним из наиболее простых и в ряде случаев дающих хорошие результаты методов определения влияния технологических или организационных факторов на характеристики выпускаемой продукции является выявление их статистической взаимосвязанности посредством нахождения эмпирических коэффициентов корреляции и корреляционных отношений (далее для краткости описания слово «эмпирический» будем опускать).

Экспериментальные исследования зависимостей технологических процессов и параметров изделий в силу их вероятностного характера дают определенные разбросы экспериментальных данных, характеризующих эти связи. При их построении получают так называемые корреляционные поля, отражающие статистический характер исследуемых зависимостей (рис. 3.1). Конечно, они дают определенное представление о взаимосвязи переменных, но малопригодны для статистических исследований и построения математических моделей. Тем не менее, перестроив эти данные (т. е. разбив ось абсцисс или ординат на интервалы, сгруппировав экспериментальные данные по интервалам, подсчитав и соединив середины распределений по интервалам), можно построить корреляционные зависимости (рис. 3.2). Полученные таким образом связи уже могут быть аппроксимированы аналитическими функциями. Однако для определения корреляционных зависимостей более эффективно (хотя иногда и менее наглядно) использовать аналитические методы.

Заметим, что при расчетах корреляционных связей используются только два параметра распределения: среднее и дисперсия, которые полностью определяют только нормальный и логарифмический законы распределения. Поэтому при исследовании корреляционных отношений предварительно необходимо установить характер изучаемых распределений. Если они несущественно отличаются от указанных выше или могут быть приведены к ним с помощью различных преобразователей, то, естественно, к ним или к трансформированным распределениям полностью можно применить изложенные ниже методы корреляционного анализа.

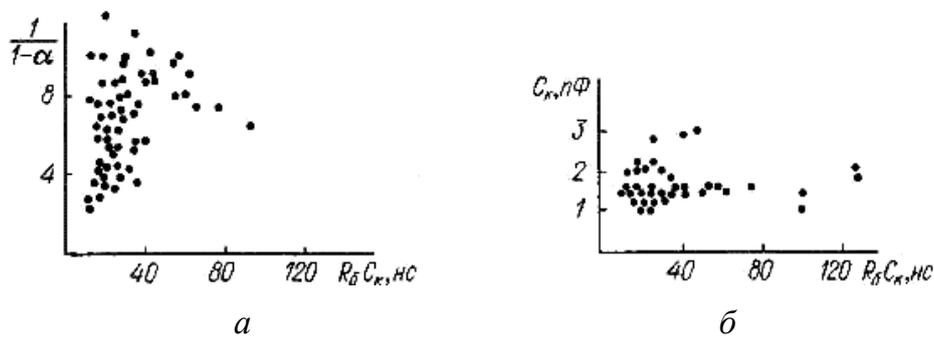


Рис. 3.1. Корреляционное поле распределения параметров:  
 $a - 1/(1-\alpha)$  по  $R_\beta C_k$ ;  $b - C_k$  по  $R_\beta C_k$

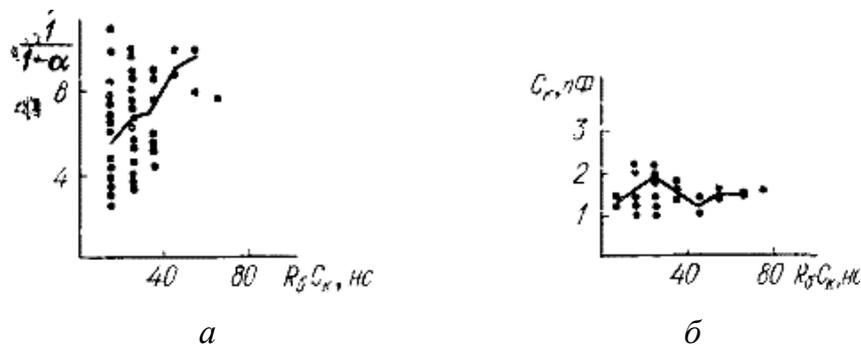


Рис. 3.2. Трансформированное корреляционное поле распределения параметров:  
 $a - 1/(1-\alpha)$  по  $R_\beta C_k$ ;  $b - C_k$  по  $R_\beta C_k$

Коэффициент линейной корреляции  $r$  статически характеризует линейную связанность переменных  $x$  и  $y$ , т. е. определяет силу связи типа

$$\left. \begin{aligned} y &= ax + b; \\ x &= cy + d. \end{aligned} \right\} \quad (3.4)$$

Значение коэффициента лежит в пределах  $-1 \leq r \leq 1$ . Если  $r = 0$ , то линейной корреляционной связи между  $x$  и  $y$  нет, хотя нелинейная зависимость между ними может существовать; если  $|r| = 1$ , то между  $x$  и  $y$  имеется жесткая функциональная зависимость.

Коэффициент линейной корреляции для двух переменных  $x$  и  $y$  определяется следующим образом:

$$r = \frac{1}{S_x S_y (N-1)} \sum_{i=1}^N (x_i - \bar{X})(y_i - \bar{Y}); \quad (3.5)$$

$$S_x = \frac{1}{N-1} \left[ \sum_{i=1}^N (x_i - \bar{X})^2 \right]^{1/2};$$

$$S_y = \frac{1}{N-1} \left[ \sum_{i=1}^N (y_i - \bar{Y})^2 \right]^{1/2};$$

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N x_i; \quad \bar{Y} = \frac{1}{N} \sum_{i=1}^N y_i.$$

где  $S_x, S_y$  – несмещенные и состоятельные оценки среднеквадратических отклонений  $\sigma_x, \sigma_y$ ;  $\bar{X}, \bar{Y}$  – несмещенные и состоятельные оценки средних значений  $a$  и  $b$ ;  $N$  – число результатов измерений.

Вычисленный по формуле (3.4) коэффициент корреляции дает состоятельную, но смещенную оценку статической взаимосвязанности переменных, однако смещение ее убывает с ростом числа испытаний и при  $N = 50$  становится менее 1 % [16].

Из-за простоты вычисления коэффициента линейной корреляции его часто используют для установления факта существования статистической зависимости при нелинейной взаимосвязи переменных. Однако необходимо проверить, можно ли применять к исследуемому объекту гипотезу о его линейности.

Значимость линейной корреляционной связи определяется соотношением  $r \geq r_{кр}$ , где  $r_{кр} = t_g / (t_g^2 + N - 2)^{1/2}$ ;  $t_g$  – критерий Стьюдента для  $g$ -процентной доверительной вероятности.

Понятие линейной корреляции между двумя переменными может быть распространено и на тот случай, когда переменных больше двух. Вычисляемый для множества переменных коэффициент линейной корреляции имеет вид

$$R_{x/y, z, \dots, t} = \left(1 - \Delta / \Delta_{11}\right)^{1/2},$$

где  $\Delta$  – определитель  $(n \times n)$ -мерной корреляционной матрицы:

$$\Delta = \begin{vmatrix} 1 & r_{x/y} & r_{x/z} & r_{x/t} \\ r_{y/x} & 1 & r_{y/z} & r_{y/t} \\ r_{t/x} & r_{t/y} & r_{t/z} & 1 \end{vmatrix}.$$

Здесь  $\Delta_{11}$  – определитель, получаемый из  $\Delta$  вычеркиванием первой строки и первого столбца.

Например, для трех переменных  $x, y, z$  коэффициент множественной линейной корреляции находится по формуле

$$r_{x/y, z} = \left( \frac{r_{xy}^2 - r_{x/y}r_{x/z} + r_{x/z}^2}{1 - r_{y/z}^2} \right)^{1/2}.$$

Значимость множественного коэффициента корреляции определяется  $F$ -отношением Фишера:

$$\left[ R^2 / (1 - R^2) \right] \left[ (N - n') / (n' - 1) \right] < F_{n'-1; N-n'}(\alpha),$$

где  $n' = n + 1$ ;  $F_{n'-1; N-n'}(\alpha)$  – верхняя точка значимости при  $n' - 1$  степенях свободы числителя и  $N - n'$  степенях свободы знаменателя, соответствующая уровню значимости  $\alpha$ .

Критическое значение коэффициента множественной линейной корреляции находят по формуле

$$R_{кр} = \left[ \frac{F(\alpha)(n' - 1)}{(N - n') - F(\alpha)(n' - 1)} \right]^{1/2}.$$

Коэффициент множественной линейной корреляции  $R$  считается значимым, если его значение удовлетворяет неравенству  $R \geq R_{кр}$ . Коэффициент линейной корреляции  $r$  используют в основном для

определения статистической взаимосвязи линейно зависимых переменных. Для оценки же статистической взаимосвязи нелинейно зависимых переменных следует использовать корреляционные отношения Пирсона:  $\eta_{x/y}$  (для зависимости  $x$  от  $y$ ) и  $\eta_{y/x}$  (для зависимости  $y$  от  $x$ ):

$$\eta_{x/y} = \left[ \frac{\sum_{j=1}^{l'} m'_j (x / y_j - \bar{X})^2}{(N-1)S_x^2} \right]; \quad \eta_{y/x} = \left[ \frac{\sum_{j=1}^l m_j (y / x_j - \bar{Y})^2}{(N-1)S_y^2} \right].$$

Здесь  $l, l'$  – число интервалов, на которые разбиваются ординаты  $x$  и  $y$ ;  $m_j, m'_j$  – число наблюдений  $(x_{ij}, y_{ij})$ , абсциссы которых попали в  $j$ -й интервал;  $\bar{X} / y_j, \bar{Y} / x_j$  – условные средние:

$$\bar{X} / y_j = \frac{1}{m'_j} \sum_{i=1}^{m'_j} x_{ij}; \quad \bar{Y} / x_j = \frac{1}{m_j} \sum_{i=1}^{m_j} y_{ij}.$$

Для корреляционного отношения всегда справедливо неравенство  $0 \leq \eta \leq 1$ . При  $\eta = 1$  между переменными  $x$  и  $y$  имеется детерминированная функциональная зависимость; при  $\eta = 0$  зависимости между переменными нет (при этом необходимо иметь в виду, что из равенства  $\eta_{x/y} = 0$  не следует равенство  $\eta_{y/x} = 0$ , и наоборот). Всегда выполняется неравенство  $\eta \geq |r|$ , причем в случае линейной корреляционной зависимости  $\eta = |r|$  и  $\eta_{x/y} = \eta_{y/x}$ . Чем хуже выполняются указанные соотношения, тем больше нелинейная зависимость  $x$  и  $y$ .

В качестве примера приведем значения  $r, \eta_{x/y}$  и  $\eta_{y/x}$  для двух пар электрических параметров одного из типов транзисторов: для малосигнального коэффициента усиления в схеме с общим эмиттером – Би-статистическим коэффициентом усиления в схеме с общим эмиттером  $B_{СТ}$  ( $r = 0,970; \eta_{x/y} = 0,979; \eta_{y/x} = 0,973$ ) постоянной цепи обратной связи  $r'_6 C_k$  и для емкости коллекторного перехода  $C_k$  ( $r = -0,006; \eta_{x/y} = 0,191; \eta_{y/x} = 0,252$ ). В первом случае имеется практически линейная зависимость между  $B$  и  $B_{СТ}$  ( $\eta_{x/y} \approx \eta_{y/x} \approx r$ ).

Во втором случае коэффициент корреляции  $r = -0,006 \approx 0,0$  говорит о крайней незначительности связи  $C_k$  и  $r'_0 C_k$ , а неравенство  $\eta_{x/y} \neq \eta_{y/x}$  — о том, что если и есть корреляционная зависимость между  $C_k$  и  $r'_0 C_k$ , то она нелинейна.

**Методы регрессионного анализа.** Для аппроксимации функциональной зависимости между переменными  $x$  и  $y$ , значения которых заданы в виде статистических распределений, можно использовать любую функцию  $f_y(x)$ , называемую регрессией  $x$  на  $y$ . Переменную  $y$  при аппроксимации статистической зависимости  $y$  от  $x$  функций  $f_y(x)$  находят из уравнения

$$y = f_y(x) + h_y(x, y),$$

где  $h_y(x, y)$  — поправочный член, определяемый погрешностью аппроксимации.

При построении математических моделей технологических процессов используют в основном два вида аппроксимации статистической зависимости  $y$  от  $x$ : линейную (см. формулу (3.4)) и параболическую

$$y = ax^2 + bx + c.$$

Для линейной регрессии выполняются следующие равенства:

$$a = \rho_{yx} = r_{xy} (S_y / S_x), \quad (3.6)$$

$$b = \bar{y} - \rho_{yx} \bar{x}, \quad (3.7)$$

где  $\rho_{yx}$  — коэффициент линейной регрессии  $y$  на  $x$ .

Полученные по формулам (3.6) и (3.7) коэффициенты уравнений минимизируют среднеквадратическое отклонение:

$$M [y - (ax + b)]^2 = \min.$$

Математическое ожидание квадрата отклонения реальной зависимости  $y(x)$  от ее аппроксимации  $M [y - (ax + b)]^2$  служит мерой точности аппроксимации реальной статистической взаимосвязанности выбранной регрессионной функции.

При вычислении регрессий на ЭВМ для корректировки математических моделей технологических процессов в системах управления уравнение нелинейной регрессии  $f_y(x)$  удобнее искать по ортогональным полиномам Чебышева  $p_j(x)$ :

$$y = b_0 p_0(x) + b_1 p_1(x) + \dots + b_n p_n(x). \quad (3.8)$$

Рекуррентность соотношений, используемых для определения полиномов  $p_j(x)$ , позволяет облегчить программирование вычислений регрессивных полиномов высоких степеней на ЭВМ.

В уравнении (3.8) коэффициенты при ортогональных полиномах  $b_j$  определяют по формуле

$$b_j = \frac{\sum_{k=1}^n y_k p_j(x_k) \varpi_k}{\sum_{k=1}^n p_j^2(x_k) \varpi_k}, \quad j = 0, 1, \dots, n,$$

где  $\varpi_k$  – вес измерения  $x_i$ , который определяют из соотношения  $\varpi_1 : \varpi_2 : \dots : \varpi_k = 1 | S_1^2 : 1 | S_2^2 : \dots : 1 | S_k^2$ .

Истинные значения параметров  $b_j$  с надежностью  $P$  лежат в доверительных границах:

$$b_j \pm t_j \left[ \frac{S}{(h-j-1)H_j} \right]^{1/2},$$

при нахождении которых значение  $t_j$  при заданном числе степеней свободы  $k = h - j - 1$  определяют по таблицам [16; 17; 21]. Значения  $S$  и  $H_j$  вычисляют по формулам:

$$S = \sum_{i=1}^n [y_i - b_0 p_0(x_i) - b_1 p_1(x_i) - \dots - b_n p_n(x_i)]^2;$$

$$H_j = \sum_{i=1}^n p_j^2(x_i).$$

Кроме рассмотренных выше регрессионных полиномов, достаточно часто используют регрессионные полиномы вида

$$y = a_0 + \sum_k a_i x_i + \sum_k a_{ij} x_i x_j + \sum_k a_{ii} x_i^2 + \dots, \quad (3.9)$$

где  $x_i$  – нормированные факторы;  $a_0, a_i, a_{ij}, a_{ii}$  – коэффициенты регрессии;  $i, j = 1, 2, 3, \dots, k$  – номер фактора  $j > k$ .

Коэффициенты регрессионных полиномов вида (3.8) и аналогичных ему можно определить методами математического планирования эксперимента.

**Методы математического планирования эксперимента.** Для многофакторных технологических процессов, математическая модель которых соответствует зависимости  $Mp_T(t) = Mp_T(t_1, t_2, \dots, t_n)$ , статическую взаимосвязанность факторов можно определить методами математического планирования эксперимента [20]. Различают три типа эксперимента:

1) активный – экспериментатор преднамеренно изменяет ход технологического процесса с целью получить его математическую модель;

2) пассивный – экспериментатор устанавливает связь между технологическими факторами по результатам наблюдений за самопроизвольными изменениями технологического процесса;

3) смешанный (активно-пассивный) – часть управляемых факторов изменяет экспериментатор по своему желанию, а остальные изменяются самопроизвольно.

В связи с тем, что при разработке математических моделей многофакторных технологических процессов применение методов активного эксперимента более эффективно, рассмотрим их подробнее [20].

Основными требованиями, предъявляемыми к технологическим факторам при ведении активного эксперимента, являются управляемость, наблюдаемость и совместная непротиворечивость. Требование управляемости технологических факторов вытекает из базового принципа ведения активного эксперимента, в котором ход технологического процесса надо изменять целенаправленно. Из этого требования вытекает требование наблюдаемости, т. е. возможности измерения значений технологических факторов с помощью имеющихся приборов, причем погрешности измерения должны быть существенно

меньше наблюдаемых изменений управляемых факторов. Требование совместной непротиворечивости измененных в процессе эксперимента факторов означает, что область выбираемых их значений не должна лежать в закритических режимах работы технологического оборудования, приводящих к его выходу из строя.

Основным инструментом метода активного планирования эксперимента является план опытов, во многом определяемых свойствами (характеристиками) подлежащего оптимизации технологического фактора, поэтому от правильности выбора оптимизируемого фактора зависит успех решения поставленной задачи – разработки математической модели процесса. Оптимальное значение технологического фактора должно соответствовать не только и не столько оптимальному режиму данной технологической операции, но оптимальному состоянию технологического процесса в целом («глобальный» оптимум). Это условие необходимо выполнять потому, что технологические операции в процессе производства взаимосвязаны, а значит, несмотря на вывод одной технологической операции в область оптимальных режимов работы, качество всего технологического процесса может снизиться.

Важное требование к оптимизируемому фактору – возможность получения его количественной оценки. При этом если оптимизируемый технологический фактор до проведения эксперимента обычно оценивали качественно (цвет, вкус, запах и т. п.), то количественную оценку ему необходимо присвоить. Операция присвоения технологическому фактору количественной оценки называется *ранжированием*, а присвоенное ему значение – *рангом фактора*. Ранг оптимизируемого фактора присваивают по заранее выбранной шкале целых или дробных значений методом экспертных оценок.

Основное отличие метода математического планирования эксперимента от классического метода планирования опытов заключается в том, что при проведении эксперимента варьируется не один, а одновременно все технологические факторы. Это повышает производительность труда (сокращает число опытов) и снижает вероятность ошибок.

При построении математических моделей технологического процесса, если из теоретических соображений о процессе нельзя сделать заключение о характере взаимосвязи между переменными, математическую модель их взаимозависимости выбирают априори. Аналогично задачам корреляционного и регрессионного анализа в качестве

таких моделей используют полиномы первой и второй степеней. В частности, для двух переменных факторов эти полиномы могут иметь вид

$$\left. \begin{aligned} y &= a_0 + a_1x_1 + a_2x_2; \\ y &= a_0 + a_1x_1 + a_2x_2 + a_{12}x_1x_2 + a_{11}x_1^2 + a_{22}x_2^2. \end{aligned} \right\} \quad (3.10)$$

Нелинейность взаимосвязи технологических факторов приводит к тому, что нелинейная модель (3.10) обычно имеет малую адекватность для всей области изменения технологических факторов и применяется в основном для описания малых областей факторов пространства, в которых нелинейными зависимостями между переменными можно пренебречь.

При математическом планировании активного эксперимента необходимо выбрать основные уровни (начальные значения) факторов и интервал их изменения (интервал варьирования). На первом шаге эксперимента в качестве начального уровня выбирают значения технологических факторов, соответствующие нормальному режиму технологического процесса. Интервал варьирования выбирают таким образом, чтобы он, с одной стороны, несколько раз укладывался в диапазон изменения варьируемого фактора, а с другой – позволял бы пренебречь ошибками исполнительных и измерительных приборов и механизмов.

Чтобы упростить запись и обработку результатов, полученных при проведении активного эксперимента, масштабы переменных выбирают так, чтобы верхний уровень фактора соответствовал значению +1, нижний – значению –1, а основной – нулю. Для натуральных непрерывных переменных это требование нормировки выполняется путем преобразования абсолютных значений технологических факторов  $x_i$  в относительные  $x_j$ :

$$x_j = (x_i - x_{i0}) / I_i,$$

где  $x_{i0}$  – основной уровень переменной, относительно которого ведется ее варьирование;  $I_i$  – интервал варьирования.

Различают два вида активного эксперимента – полный факторный и дробный факторный эксперименты.

При полном факторном эксперименте реализуются все возможные сочетания уровней. Число опытов  $N$  для получения оценки со-

стояния технологического процесса определяется выражением  $N = 2^k$ , где  $k$  – число варьируемых переменных. В частности, для трех переменных необходимо провести восемь опытов по матрице планирования (табл. 3.1).

Таблица 3.1

**Матрица планирования для трех переменных (8 опытов)**

Номер опыта	$x_1$	$x_2$	$x_3$	$y$
1	+1	-1	-1	1
2	-1	+1	-1	2
3	-1	-1	+1	3
4	+1	+1	-1	4
5	+1	-1	+1	5
6	-1	+1	+1	6
7	+1	+1	+1	7
8	-1	-1	-1	8

Значения  $y_1 - y_8$  оптимизируемого технологического фактора  $y$  при трехфакторном активном эксперименте соответствуют его значениям в вершинах куба факторного пространства (области существования технологического процесса). По результатам полного факторного эксперимента находят коэффициенты аппроксимирующего полинома (3.10):

$$a_i = \sum_{j=1}^N y_j x_{ij} / N, \quad (3.11)$$

где  $i$  – номер фактора ( $i = 0, 1, 2, \dots$ ).

Фактор  $x_i$  при  $i = 0$  называют фиктивным фактором и вводят для удобства вычисления коэффициента  $a_0$ , входящего в математическую модель технологического процесса (3.10). Фиктивному фактору во всех опытах приписывают постоянное значение (например, +1). При введении такого фактора в математическую модель технологического процесса последняя принимает вид

$$y = f(x_0, x_1, x_2, \dots, x_n). \quad (3.12)$$

Полный факторный эксперимент с  $N = 2^4$  был использован при определении коэффициентов линейного регрессионного полинома для построения математической модели формирования толсто пленочных резисторов [22]:

$$H_p = 13,55 + 0,41x_1 + 0,45x_2 - 0,32x_3 + 0,19x_4,$$

где  $H_p$  – толщина резистивного слоя, мкм;  $x_1$  – расстояние «сетка–трафарет–подложка», мм;  $x_2$  – вязкость резистивной пасты, мм;  $x_3$  – давление ракеля, Н;  $x_4$  – скорость движения ракеля, мм/с.

Использование модели позволяет предсказать толщину резистивного слоя с точностью не ниже  $\pm 10\%$ .

При полном факторном эксперименте появляется возможность учесть без увеличения числа опытов взаимодействие технологических факторов. В этом случае математическая модель технологического процесса (3.12) будет иметь вид

$$y = a_0x_0 + a_1x_1 + a_2x_2 + a_{12}x_1x_2, \quad (3.13)$$

а матрица планирования – соответствовать табл. 3.2.

Таблица 3.2

Матрица планирования (4 опыта)

Номер опыта	$x_0$	$x_1$	$x_2$	$x_1x_2$	$y$
1	+1	+1	+1	+1	1
2	+1	-1	+1	-1	2
3	+1	-1	-1	+1	3
4	+1	+1	-1	-1	4

Коэффициенты  $a_i$  в выражении (3.13) вычисляются по формуле (3.10), а  $a_{mi}$  (в приведенном примере  $a_{12}$ ) – по выражению

$$a_{mi} = \sum_{j=1}^N y_j x_{mj} x_{ij} / N,$$

где  $m$  и  $i$  – номера факторов ( $m, i = 1, 2, \dots, k$  при  $m \neq i$ ).

Исследуя технологический процесс методом полного факторного эксперимента с нелинейными зависимостями факторов, из полученных результатов нельзя извлечь информацию о квадратических

членах математической модели [5]. Если форма математической модели технологического процесса линейная, то число опытов, проводимых для определения ее коэффициентов, по сравнению с числом опытов при полном факторном анализе, можно сократить, поскольку при построении линейной модели принимается гипотеза о незначимости эффектов взаимодействия факторов. Поэтому столбцы взаимодействия в матрицах планирования эксперимента можно заменить новой переменной. Так, в нашем примере столбец  $x_1x_2$  можно заменить столбцом  $x_3$  (табл. 3.3).

Таблица 3.3

**Матрица планирования без столбца взаимодействия**

Номер опыта	$x_0$	$x_1$	$x_2$	$x_3$	$y$
1	+1	+1	+1	+1	1
2	+1	-1	+1	-1	2
3	+1	-1	-1	+1	3
4	+1	+1	-1	-1	4

Из приведенной матрицы планирования следует, что при замене столбца  $x_1x_2$  столбцом  $x_3$  (введение третьего переменного фактора) число опытов остается таким же, как и для двух переменных факторов (фактор  $x_0$  – фиктивный), т. е. равно четырем. Полученные таким способом значения коэффициентов  $a_1, a_2$  и  $a_3$  будут смешанными оценками, включающими эффект истинного действия фактора  $\alpha_i$  и эффекта парных взаимодействий  $a_{ij} : a_1 = \alpha_1 + \alpha_{23}; a_2 = \alpha_2 + \alpha_{13}; a_3 = \alpha_3 + \alpha_{12}$ .

В отличие от матрицы планирования полного факторного эксперимента, называемой полной репликой, матрица планирования, составленная заменой столбцов взаимодействий на новые переменные факторы, называется дробной репликой (в рассмотренном случае – полурепликой). Использование дробных реплик при планировании активного эксперимента позволяет резко сократить число опытов, которые необходимо провести для определения коэффициентов математической модели технологического процесса. Так, в частности, при анализе влияния 15-ти технологических факторов число опытов можно уменьшить с 32 768 до 16. По этой причине в адаптивных системах управления технологическими процессами алгоритмы «обучения»

системы управления строятся на базе дробного факторного анализа. Справедливость гипотезы незначимости взаимодействия технологических факторов для автоматизируемого технологического процесса проверяют на этапе проектирования, а систему смешения линейных эффектов с нелинейными, необходимую при замене полного факторного эксперимента дробным, выбирают и проверяют еще на этапе исследования технологического процесса.

Математическую модель технологического процесса, полученную методом активного планирования эксперимента, проверяют на адекватность и значимость ее коэффициентов  $a_i$  и  $a_{ij}$ . Дисперсию адекватности модели для числа степеней свободы определяют по формуле

$$S_{\text{ад}}^2 = \sum_{j=1}^N \Delta y_j^2 / f,$$

в которой

$$f = N - (k + 1),$$

где  $N$  – число опытов;  $k$  – число варьируемых факторов.

По найденной дисперсии адекватности  $S_{\text{ад}}$  и известной дисперсии воспроизводимости опытов  $S_y^2$  по критерию Фишера определяют адекватность модели технологического процесса:

$$F = S_{\text{ад}}^2 / S_y^2.$$

Значимость коэффициентов математической модели проверяют аналогично значимости коэффициентов регрессионных полиномов.

Рассмотрим примеры использования метода активного эксперимента для решения конкретной задачи построения математической модели процесса ультразвуковой приварки проволочных выводов к контактными площадкам кремниевой интегральной схемы. Положим, что прочность сварки  $y$  нелинейно зависит от давления, прижимающего проволоку к контактной площадке  $P$ , и от мощности ультразвуковых колебаний  $\varpi$ , а также соответствует зависимости

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_{11} x_1^2 + a_{22} x_2^2 + a_{12} x_1 x_2, \quad (3.14)$$

где  $x_1$  и  $x_2$  – нормированные переменные:

$$x_1 = (P - 2,5) / 3,5; \quad x_2 = (\varpi - 4) / 1,4.$$

Чтобы определить коэффициенты регрессии  $a_0, a_1, a_2, a_{11}, a_{22}$ , в уравнении (3.14) используем схему рототабельного планирования второго порядка для двух факторов с равномерным дублированием опытов (например, десять опытов в каждой серии опытов). В соответствии с правилами построения рототабельных планов второго порядка для двух переменных нужно иметь 13 серий экспериментов ( $N = 13$ ). В итоге получим матрицу планирования с результатами усреднения по дублированным опытам, а также данными по усилиям отрыва проволоки от контактной площадки  $y_p$ , имеющими довольно хорошее совпадение с опытными данными  $y_{\Pi}$  для случаев, когда  $y_{\Pi} > 5$  (табл. 3.4).

Таблица 3.4

**Матрица планирования при дублировании опытов**

Номер опыта	$x_0$	$x_1$	$x_2$	$x_1^2$	$x_2^2$	$x_1x_2$	$y_{\Pi}$	$y_p$
1	+1	-1	-1	+1	+1	+1	5,935	4,843
2	+1	+1	-1	+1	+1	-1	9,945	9,295
3	+1	-1	+1	+1	+1	-1	11,96	12,16
4	+1	+1	+1	+1	+1	+1	5,225	5,869
5	+1	-2	0	+2	0	0	10,58	11,12
6	+1	+2	0	+2	0	0	9,905	9,816
7	+1	0	-2	0	+2	0	3,105	4,242
8	+1	0	+2	0	+2	0	4,685	6,994
9	+1	0	0	0	0	0	10,80	10,83
10	+1	0	0	0	0	0	10,86	10,83
11	+1	0	0	0	0	0	10,80	10,83
12	+1	0	0	0	0	0	10,08	10,83
13	+1	0	0	0	0	0	11,61	10,83

*Примечание.* Коэффициенты регрессии в данном случае следующие: для  $x_0 = 10,83$ ; для  $x_1 = -0,181$ ; для  $x_2 = 0,973$ ; для  $x_1^2 = -0,181$ ; для  $x_2^2 = -2,606$ ; для  $x_1x_2 = -2,686$ .

С учетом рассчитанных коэффициентов регрессии имеем следующую модель технологического процесса приварки выводов:

$$y = 10,83 - 0,460x_1 + 0,937x_2 - 0,181x_1^2 - 2,605x_2^2 - 2,686x_1x_2.$$

**Методы группового учета аргументов (МГУА).** Необходимость оптимизации и управления технологическими процессами служит постоянным стимулом развития различных методов их математического анализа, что в настоящее время особенно актуально, поскольку технология производства постоянно развивается, появляются новые, сложные и мало изученные технологические процессы, которыми, тем не менее, необходимо управлять. Кроме того, все увеличивается число технологических процессов, нормально функционирующих только в довольно ограниченных областях технологических параметров, выход за которые может привести не только к серьезному снижению производительности, но и вообще к выходу объекта управления в аварийные состояния с непредсказуемыми последствиями.

Современное развитие средств вычислительной техники, доступность их практически каждому исследователю в качестве пользователя как персональных, так и суперЭВМ, определяют дальнейшее развитие математических методов анализа технологических объектов и процессов: оно идет по пути разработки и реализации универсальных вычислительных методов, алгоритмов и программ, в которых ЭВМ отводится центральное место в качестве интеллектуального субъекта [5; 23–26].

Метод и алгоритмы МГУА предназначены для машинной идентификации, моделирования и прогнозирования сложных процессов и систем. МГУА не требует от пользователя квалифицированных знаний об исследуемом объекте, достаточно лишь, чтобы он ввел данные выборки наблюдений, предварительно определил классы моделей, среди которых следует искать желаемую, а также критерий решения задачи идентификации (соответствие построенной ЭВМ математической модели реальному объекту). Сама задача идентификации (определение оптимальной структуры математической модели и расчет ее параметров) решается автоматически [25].

Алгоритмы МГУА работают и при наличии «шума» в системе. В зависимости от класса моделей это могут быть «шумы» измерений на выходе объекта либо входные, в качестве которых обычно рассматривается так называемый «белый шум». Впрочем алгоритмы МГУА успешно применяются также при сильно зашумленных факторах и коррелированных шумах [26; 27].

МГУА включает в себя следующие алгоритмы: комбинаторные, селекционно-комбинаторные, итерационные, моделирования многомерных объектов (выходы заданы), объектный системный анализ (входы и выходы не заданы).

Структурно-параметрическая идентификация представляет собой совокупность процедур, позволяющих выбрать из множества структур моделей некоторую функцию  $f$  и рассчитать такие параметры функции, чтобы полученное решение обеспечивало минимум некоторого критерия.

Принцип работы МГУА можно показать на следующем примере [27]. Пусть задан некоторый набор данных, о которых известно, что они содержат как входные переменные (переменные состояния  $x_1, x_2, \dots, x_{10}$ ), так и выходные ( $y_1, y_2, \dots, y_4$ ). В принципе в данном случае мы можем получить четыре полных полинома (линейных или нелинейных) и четыре уравнения:

$$\begin{aligned} y_1 &= f_1(x_1, \dots, x_{10}; y_2, y_3, y_4); \\ y_2 &= f_2(x_1, \dots, x_{10}; y_1, y_3, y_4); \\ y_3 &= f_3(x_1, \dots, x_{10}; y_1, y_2, y_4); \\ y_4 &= f_4(x_1, \dots, x_{10}; y_1, y_2, y_3). \end{aligned}$$

Применяя итерационный алгоритм МГУА, получим две системы:

$$\begin{aligned} &\left\{ \begin{aligned} y_1 &= f_1(x_1, x_2, y_2) \\ y_2 &= f_2(x_2, x_3, y_1) \end{aligned} \right\}, \\ &\left\{ \begin{aligned} y_3 &= f_3(x_4, x_5, y_4) \\ y_4 &= f_4(x_5, x_6, y_3) \end{aligned} \right\}. \end{aligned} \tag{3.15}$$

Из систем (3.15) следует, что для дальнейшего изучения объекта необходимо организовать четыре плана экспериментов:

$$\begin{aligned} &\langle x_1, x_2, y_1, y_2 \rangle, \langle x_4, x_5, y_3, y_4 \rangle, \\ &\langle x_2, x_3, y_1, y_2 \rangle, \langle x_5, x_6, y_3, y_4 \rangle. \end{aligned}$$

Заметим, что длина выборки измерений для полученной системы алгебраических уравнений равна числу уравнений в системе (алгебраический минимум), однако для качественного решения задачи структурно-параметрической идентификации желательно брать число точек, в 5–10 раз большее алгебраического минимума.

Задачу структурно-параметрической идентификации методами МГУА решают в следующей последовательности [25; 28; 29]:

- получение экспериментальных данных;
- выбор на основе анализа экспериментальных данных класса базисных функций и соответствующего преобразования данных;
- генерация различных моделей в выбранном классе функций;
- оценка параметров структур и формирование множества функций, из которого далее предстоит выбрать наилучшую модель;
- выбор лучшей модели  $f$  в соответствии с критерием оптимальности;
- проверка адекватности полученной оптимальной модели.

Целями моделирования в МГУА являются:

- построение моделей объекта исследования, отражающих закономерности «вход–выход» («физическая», или несмещенная, модель);
- создание модели для наилучшей фильтрации (сглаживания) зашумленных выходных переменных («фильтрующая» модель);
- получение наилучшей модели для прогнозирования (экстраполяции) выходных переменных («прогнозирующая» модель).

Существуют следующие классы моделей [25]:

- регрессионные модели статических объектов управления;
- модели временных рядов или процессов;
- модели динамических систем в виде линейно-разностных уравнений.

Регрессионные модели статических моделей имеют вид, аналогичный уравнению

$$y_i = f(x_i) + \xi_i = B\varphi(x_i) + \xi_i,$$

где  $i$  – индекс точки наблюдения;  $\varphi(x)$  – известная из эксперимента  $M$ -мерная вектор-функция от  $x$  входных переменных;  $B$  – неизвестная  $M_y \times M$  матрица параметров;  $\xi_i$  – независимый некоррелируемый случайный вектор с нулевым средним и конечной неизвестной дисперсией.

Слагаемое  $B\varphi(x_1)$  можно представить полиномами различных степеней вида

$$b^T \varphi_j(x) = \sum_{j=1}^M b_j \prod_{\gamma=1}^{MX} x_{\gamma}^{C_{j\gamma}}, \quad (3.16)$$

где  $C_{j\gamma}$  принимает только целые значения: 0, 1, ...,  $ST$ .

Полиномы вида (3.16) широко используют в регрессионном анализе и, в частности, в системах математического планирования эксперимента. Из моделей временных рядов в МГУА достаточно широко используется также известная тригонометрическая модель тренда процесса:

$$b^T \varphi(t) = \sum_{j=1}^M (a_j \sin \omega_j t + b_j \cos \omega_j t) = \sum_{j=1}^M C_j \sin(\omega_j t) + \psi_j.$$

При моделировании динамических систем используют линейно-разностные уравнения вида

$$y_k = \sum_{\alpha=1}^{LY} A_{\alpha} y_{k-\alpha} + \sum_{v=1}^{LX} B_v x_{k-v+1} + \xi_k,$$

где  $LY, LX$  – число учитываемых прошлых значений (запаздываний для выходных и входных переменных);  $A_1, \dots, A_{LY}, B_1, \dots, B_{LX}$  – матрицы неизвестных параметров;  $\xi_k$  имеет смысл входного процесса типа «белого шума».

Добавим, оптимальную структуру математических моделей формируют одним из двух способов: либо перебором конечного числа различных структур заданного класса, либо итерационным наращиванием сложности структур с использованием селекционных или релексционных методов.

В МГУА предусмотрена возможность использования целого комплекса критериев: регулярности, непротиворечивости, баланса прогноза, баланса переменных, интегрального и др. Основными системными критериями являются критерии регулярности  $ARS$  и непротиворечивости  $CBS$ , вычисленные по принципу среднего:

$$ARS = 1 / MY \sum_{l=1}^{MY} AR_l; CBS = 1 / MY \sum_{l=1}^{MY} CB_l,$$

где  $AR_l$  и  $CB_l$  – значения соответствующих критериев для  $l$ -го уравнения системы.

Критерии МГУА формируют разделением выборки  $D$  на две части –  $A$  и  $B$ :

$$X = \left( X_A^T / X_B^T \right); Y = \left( Y_A^T | Y_B^T \right);$$

$$\theta = \left( X_G^T X_G \right)^{-1} X_G^T Y_G; G = A, B, W, A \cup B = D; A \cap B = 0.$$

Критерий регулярности в данном случае можно сформировать в виде

$$\left\| Y_B - X_B \hat{\theta}_A \right\|^2,$$

а критерий непротиворечивости:

$$\left\| X_B \hat{\theta}_A - X_W \hat{\theta}_B \right\|^2.$$

Методы МГУА достаточно широко применяют для построения математических моделей различных процессов и объектов. Приведем пример использования МГУА для построения математической модели прогноза хода технологического процесса в алюминиевом электролизере [30].

Основным нарушением хода технологического процесса электролиза является анодный эффект (искровой разряд на границе между электролитом и поверхностью погруженного в него анода). При возникновении анодного эффекта снижаются производительность и выход металла, увеличивается расход энергетических и материальных ресурсов, ухудшаются санитарно-гигиенические условия. Чтобы предсказать наступление анодного эффекта, необходимо определить разделяющую функцию между нормальным ходом технологического процесса и его нарушением. Предварительные исследования показали, что по мере приближения к анодному эффекту в расплаве нарастают низкочастотные колебания электрического тока. Модель процесса искали в виде соотношения (3.16), используя критерий  $CR$ , определяемый через рассогласование наблюдаемых  $Y_B$  и модельных  $\hat{Y}_B$  значений выхода:

$$CR = \left( Y_B - \hat{Y}_B \right)^T \left( Y_B - \hat{Y}_B \right).$$

Итерационная процедура построения модели заканчивается, как только значения критерия  $CR$  для лучших моделей двух последовательных итераций совпадают. При значении критерия  $CR = 0,34$  полученная разделяющая функция имеет вид

$$R^*(x^*) = x_5^* + 33,9x_1^* + 46,1x_3^*x_6^* - 67,3x_3^*x_{13}^*, \quad (3.17)$$

где  $x^*$  – нормированные значения, определяемые выражением

$$x_{jt}^* = (x_{jt} - \bar{x}_j) / \sigma_{xj}^2,$$

в котором  $\bar{x}_j$  – среднее;  $\sigma_{xj}^2$  – дисперсия;  $j$  – порядковый номер низкочастотных колебаний (гармоник);  $t$  – время, в течение которого в эксперименте после проведения измерений наступал анодный эффект. В свою очередь,  $x_j$  рассчитывается по формуле

$$x_j = \lg \left( 10^5 S_{yy} (f_{j-1}) \right),$$

где  $S_{yy}$  – оценка плотности спектра мощности.

Полученная формула в подавляющем большинстве случаев позволяет предсказывать анодный эффект более чем за минуту до его наступления. Этого времени вполне достаточно, чтобы принять меры для предотвращения нарушения рабочего режима электролизера.

### 3.3. МЕТОДЫ ОЦЕНКИ И ОПТИМИЗАЦИИ СОСТОЯНИЯ ОБЪЕКТА УПРАВЛЕНИЯ

Одной из основных задач, решаемых системой управления, является получение оценки состояния объекта управления. В алгоритме управления эту оценку может вырабатывать отдельный блок (в ряде случаев блок оценки состояния объекта объединяется с блоком выработки управляющих воздействий). Для оценки состояния управляемого объекта используют методы наименьших квадратов, максимального правдоподобия, байесовых оценок и ранжирования.

*Метод наименьших квадратов* заключается в таком определении оценки параметров состояния объекта (функция  $y$ ), чтобы они минимизировали квадратичное отклонение  $S$  измеренных значений  $y$  от прогнозируемых значений  $f(x_k, a_0, a_1, \dots, a_n)$ :

$$S = \sum_{k=1}^N \left[ y_k - f(x_k, a_0, a_1, \dots, a_n) \right]^2 \omega_k = \min,$$

где  $a_0, a_1, \dots, a_n$  – оцениваемые параметры состояния объекта;  $\omega_k$  – веса измерений, обратно пропорциональные дисперсиям (в качестве веса измерения  $\omega_k$  может быть использовано число измерений  $m_k$  в точке  $k$ ).

Отыскание значений параметров состояния объекта  $a_0, a_1, \dots, a_n$  сводится к решению систем уравнений, которые имеют вид  $\partial S / \partial a_i = 0$ .

Преимущество метода наименьших квадратов заключается в том, что он не требует каких-либо дополнительных сведений о статистических свойствах случайных переменных (за исключением конечности величин их дисперсий). Это позволяет относительно широко использовать его для построения алгоритмов блока оценки состояний объекта в адаптивных АСУ ТП. Кроме того, этот метод дает возможность получить несмещенные и состоятельные оценки параметров  $a_i$ . Если прогнозируемое состояние объекта описывается линейной зависимостью, то оценки, полученные методом наименьших квадратов, являются эффективными [4].

*Метод максимального правдоподобия* является одним из основных методов теории точечных оценок. С его помощью можно получить такую оценку параметра или переменной, которая максимизирует функцию правдоподобия

$$L(x_1, x_2, \dots, x_n) = \varphi(x_1), \varphi(x_2), \dots, \varphi(x_n), \quad (3.18)$$

где  $x_i$  – случайная величина с подлежащей определению плотностью вероятности  $\varphi(x_i)$ .

Из выражения (3.18) следует, что функцией правдоподобия служит плотность распределения вероятностей, рассматриваемая как функция неизвестного параметра. По методу максимального правдоподобия в качестве оценки каждого параметра  $\eta_k$  выбирается такая функция  $y_k(x_1, x_2, \dots, x_n)$ , которая придает возможно большее значение величине  $L(x_1, x_2, \dots, x_n; y_1, y_2, \dots, y_m)$  для всех выборок  $(x_1, x_2, \dots, x_n)$ . Чтобы получить совокупность  $m$  наиболее правдоподобных оценок  $y_1(x_1, x_2, \dots, x_n), y_2(x_1, x_2, \dots, x_n), \dots, y_m(x_1, x_2, \dots, x_n)$ , решают систему  $m$  уравнений максимального правдоподобия:

$$\frac{\partial}{\partial y_k} \ln L(x_1, x_2, \dots, x_n; y_1, y_2, \dots, y_m) = 0, \quad k = 1, 2, \dots, m.$$

Данный метод требует более сложных вычислений, чем метод наименьших квадратов, однако полученные оценки могут оказаться предпочтительнее (особенно в случаях малых выборок). Метод применим к многомерным совокупностям, при этом получаются состоятельные, асимптотически не смещенные и асимптотически эф-

фективные оценки. Также данный метод широко применяют для оценки состояния статических и динамических объектов при наличии шума [31–33].

*Метод байесовых оценок* основан на применении формулы Байеса:

$$p(x_i / y^i) = \frac{p(x_i) p(x_i / y^i)}{\sum_{i=1}^n p(x_i) p(x_i / y^i)}, \quad (3.19)$$

из которой следует, что вероятность состояния объекта после испытаний равна произведению вероятности состояния до испытания на соответствующую ей условную вероятность события, которое произошло при испытании, деленному на полную вероятность этого события. Данная формула позволяет при известных состояниях объекта до ( $x_i$ ) и после испытаний ( $y^i$ ) прогнозировать его состояние [31]. Качество управления объектом при этом можно оценить определением математического ожидания условной вероятности  $E(x_i / y^i)$ . В некоторых случаях в качестве оценки вероятности  $p(x_i / y^i)$  могут использоваться мода, медиана и т. п.

Байесовы оценки наиболее широко применяют в системах управления дискретными технологическими процессами, например, по оценке выхода годных изделий, точности изготовления механических деталей и т. п. Однако их можно использовать и в управлении непрерывными технологическими процессами. Особенно часто формулу Байеса применяют для оценок состояния статистических и динамических объектов, имеющих большие шумы, что позволяет получить весьма важные результаты, в частности, уравнения фильтра Винера–Калмана и фильтра Калмана [3; 31; 33].

*Метод ранжирования* используют для оценки состояния объектов управления в том случае, когда необходимо контролировать параметры, ранее измерявшиеся качественно. Им приписывают некоторые количественные оценки, т. е. вводят ранжирование параметров [20]. Допускаемая при этом производительность выбора количественных оценок затрудняет управление объектом в адаптивных системах.

Чтобы унифицировать процесс ранжирования различных качественных параметров, можно использовать принцип нормированного ранжирования по какому-либо априорно принятому закону.

Проиллюстрируем принцип введения априорного нормированного ранжирования на примере одномерной задачи. Распределим все известные качественные оценки состояния объекта управления по уровням таким образом, чтобы более значимые оценки находились на верхних уровнях, а менее значимые – на нижних. В итоге получим  $N$  уровней качественных оценок. Присвоим каждому из них порядковый номер  $n_N$  (для верхнего уровня качества  $n=1$ ), после чего найдем такой натуральный ряд, в котором независимо от числа уровней качественной оценки параметра сумма его членов равна единице, т. е. введем нормировку. Натуральный ряд, соответствующий данным условиям, существует (он был предложен Л. В. Добролюбовым) и определяется выражением

$$S_N = 2 \sum_{j=0}^N \frac{N-j}{N(N+1)}, \quad (3.20)$$

где  $j = n_N - 1$ .

Нетрудно убедиться, что для любых  $N$  всегда выполняется равенство  $S_N = 1$ . В соответствии с соотношением (3.20) вес каждого качественного уровня находят по формуле

$$p_n = 2 \frac{N - n_i + 1}{N(n_i + 1)}, \quad i = 1, 2, \dots, N.$$

Если при ранжировании несколько значений качественных параметров  $k$  отнесены к одному и тому же уровню, то вес каждого качественного значения определяют следующим образом:

$$p_{nk} = 2 \frac{N - n_i + 1}{N(n_i + 1)k}.$$

Принцип нормированного ранжирования относительно легко распространяется и на многомерные случаи (например, сборочные операции, при которых качество сборки определяется качеством изделий, поступающих на сборку).

Теперь обратимся к оптимизации состояния объекта управления. Она также может быть проведена различными методами: вариационного исчисления, основанными на принципе максимума, динамического программирования, градиентными, а также посредством GERT-моделирования и др.

Методы вариационного исчисления применяют в тех случаях, когда задачи оптимального управления математически сводятся к определению неизвестных функций  $y(x)$ , обеспечивающих экстремум определенных интегралов вида

$$I = \int_{x_1}^{x_2} F \left[ y_1(x), y_2(x), \dots, y_n(x); y_1'(x), y_2'(x), \dots, y_n'(x), x \right] dx, \quad (3.21)$$

$$n = 1, 2, \dots, N.$$

Как известно, вариация  $\delta y$  функции  $y(x)$  переменного  $x$  есть функция от  $x$ , определяемая при каждом значении  $x$  как разность новой функции  $v(x)$  и функции  $y(x)$ :  $\delta y = v(x) - y(x)$ . Вариация  $\delta y$ , вызывая изменение функциональных связей между  $x$  и  $y$ , приводит к изменению функции  $F \left[ y_1(x), y_2(x), \dots, y_n(x); x \right]$ . Приращение  $\Delta F$ , соответствующее вариациям  $\delta_1, \delta_2, \dots, \delta_n$ , определяется следующим образом:

$$\Delta F = F(y_1 + \delta y_1; y_2 + \delta y_2; \dots, y_n + \delta y_n; x) - F(y_1, y_2, \dots, y_n; x).$$

Если функции  $y(x)$  и  $\delta(x)$  дифференцируемые, то вариация производной  $y'(x)$  имеет вид

$$\delta y' \equiv \delta \frac{dy}{dx} \equiv \frac{d}{dx}(\delta y) = v'(x) - y'(x).$$

Необходимым, но недостаточным условием экстремума функционала (3.20) является условие

$$\delta I = \int_{x_1}^{x_2} \delta F dx = \frac{\partial F}{\partial y'} \delta y \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \left[ \frac{d}{dx} \left( \frac{\partial F}{\partial y'} \right) - \frac{\partial F}{\partial y} \right] \delta y dx = 0$$

для произвольно малой вариации  $\delta y$ .

Основная задача оптимальных систем, решаемая методами вариационного исчисления, состоит в нахождении такой функции управления, которая минимизировала бы функционал (3.21) при следующих условиях: закрепленных граничных точках, подвижных граничных точках и минимуме отклонений выходной переменной

от заданного значения [31]. Вариационные задачи при перечисленных условиях в большинстве случаев решают численными методами решения дифференциальных уравнений. Тем не менее ряд практически важных результатов можно получить и аналитически.

Задача нахождения минимума функционала при закрепленных конечных точках имеет вид

$$I = \int_{t_0}^{t_k} F(y, y', t) dt \quad (3.22)$$

и интерпретируется как определение оптимальной траектории движения объекта, обеспечивающей минимум отклонения скорости  $y'$  от заданной. Экстремум функционала  $I$  находят из решения дифференциального уравнения, соответствующего обращению в нуль вариации функционала:

$$\partial E / \partial y - (d / dt)(\partial F / \partial y') = 0. \quad (3.23)$$

Уравнение (3.23), определяющее оптимальную траекторию  $y(x)$ , является дифференциальным уравнением Эйлера–Лагранжа и в ряде случаев может быть решено аналитически.

Задача нахождения минимума функционала при подвижных конечных точках (одна или обе точки перемещаются) обязательно имеет решение в случаях, если функционал (3.23) минимизируется при неподвижных конечных точках. Класс функций, удовлетворяющих минимуму функционала, расширяется, так как положение конечных точек не фиксировано [31].

Задача нахождения минимума функционала при условии минимума отклонения выходной переменной от заданного значения сводится к решению уравнения

$$(F'_y g'_x - F'_x g'_y) / g'_x - (d / dx)(F'_x g'_x) = 0, \quad (3.24)$$

где  $F'_y$ ,  $g'_y$ ,  $g'_x$  – частные производные  $\partial F / \partial y'$ ,  $\partial g / \partial x$  и  $\partial g / \partial y$ .

Задачу решают для объектов управления, которые описываются функцией  $y'(t) = g(y, x)$  при начальных условиях  $y(t_0) = y_0$ . Оптимальную функцию управления  $y(t)$  находят из уравнения (3.24) численными методами.

Использовать методы вариационного исчисления для оптимизации процессов управления не всегда эффективно из-за сложности учета ограничений, налагаемых на изменения переменных, а также из-за необходимости решать на управляющей ЭВМ двухточечные краевые задачи, требующие значительных затрат машинного времени.

*Методы, основанные на принципе максимума*, в отличие от классических методов вариационного исчисления наиболее широко применяются в случае наложения ограничений на управляющие воздействия за счет принципа максимума Понтрягина [31], при использовании которого еще на этапе проектирования оптимальных систем управления можно определить основные особенности системы управления и сделать некоторые заключения относительно ее структуры.

Принцип максимума дает необходимое и достаточное условие оптимальности линейных систем, описываемых уравнением вида

$$y'_i(t) = \sum_{k=1}^n A_{ik}(t) y_k(t) + x_i(m_1, m_2, \dots, m_n), \quad i = 1, 2, \dots, n,$$

или в векторной форме:

$$y'_i(t) = A(t)y(t) + x(m).$$

В общем случае (например, для нелинейных систем) принцип максимума дает только необходимое условие экстремума. Управление объектом будет оптимальным, если обеспечивается возможность получения экстремума функции:

$$\rho = b'y(t),$$

где  $b'$  – вектор-столбец, зависящий от оптимизированных координат;  $y$  – вектор состояния рассматриваемого процесса  $n$ -го порядка.

Упрощенная геометрическая интерпретация принципа максимума заключается в определении такого вектора управления  $x$ , при котором вектор состояния  $y(t_k)$  будет как можно дальше перемещаться в направлении  $b$  (в этом случае  $\rho = \rho_{\min}$ ).

Принцип максимума при реализации оптимальных процессов управления дает возможность учесть ограничения, налагаемые на переменные, но имеет тот же недостаток, что и классические вариаци-

онные методы оптимизации – требует решения на управляющий ЭВМ двухточечных краевых задач, т. е. выполнения большого числа вычислительных операций.

Предложенный Беллманом *метод динамического программирования* основан на одном из фундаментальных принципов теории оптимальных процессов: независимо от начального или конечного решения, решение, принимаемое на любом последующем шаге управления системой, должно быть оптимальным по отношению к состоянию, для которого оно определяется. Применение данного метода строго обосновано для дискретных систем, с некоторыми ограничениями его можно использовать для систем непрерывных.

Основное достоинство данного метода состоит в том, что он позволяет заменить поиск закона оптимального управления на всем интервале оптимизации поиском оптимальных решений на  $i$  последовательных шагах управления. Следует, однако, иметь в виду, что полученный методом динамического программирования оптимальный закон управления в целом на всем интервале может быть неоптимальным на его промежуточных участках [31].

Достаточно эффективный, хотя и приближенный способ, реализующий метод динамического программирования для определения оптимального закона управления  $S(y, t)$  объекта, описываемого уравнением  $y = F_0(y, x, t)$  на одном шаге оптимизации, заключается в следующем. Полагая, что

$$S(y, t) = \min_{x(t)} \int_{t_0}^{t_k} F(y, x, t) dt,$$

запишем приближенное равенство:

$$S(y, t) \approx \min_{x(t)} \left\{ F[y, x(t), t] \Delta t + S(t + \Delta t, y + \Delta y) \right\}, \quad (3.25)$$

где  $F$  – вектор-функция состояния системы в точке  $t$ ;  $y$  – траектория движения;  $x$  – вектор управляющих воздействий;  $\Delta t$  – шаг изменения независимой переменной.

Значение  $y + \Delta y$  можно приближенно записать в виде

$$y + \Delta y = y(t) + \Delta t F_0[y(t), x(t), t] = y + \Delta t F_0[y, x(t)], \quad (3.26)$$

где  $F_0$  – вектор-функция состояния системы в точке  $t_0$ .

Вычисленная по формулам на каждом шаге траектория движения оптимальна на заданном интервале управления  $T = t_k - t_0$ . Поскольку для конечного момента времени  $t_k$  и любых  $y$  всегда выполняется равенство  $S(t_k, y) = 0$ , определение оптимальной стратегии можно начинать с конечного момента времени  $t_k$  (первый шаг  $t = t_k - \Delta t$ , второй –  $t = t_k - 2\Delta t$  и т. д.).

Недостатком метода динамического программирования при его использовании в оптимальных системах управления является необходимость выполнения большого числа шагов ( $\Delta t$  желательно иметь небольшим) и наличия относительно большого объема ОЗУ для хранения промежуточных вычислений.

*Градиентные методы* используют для формирования оптимальных стратегий управления в адаптивных АСУ ТП, если управление ведут на основе анализа поверхности отклика объекта. Эти методы заключаются в том, что если область эксперимента, ведущегося адаптивной системой управления, охватывает небольшую часть поверхности отклика, то для практических целей на каждом шаге область управления можно рассматривать как плоскость. Определив уравнение плоскости, можно найти направление, в котором следует двигаться к экстремальной точке. Так как система управления на каждом шаге должна вести эксперимент в направлении максимальной крутизны поверхности отклика, то такое управление будет оптимальным (для данного момента оптимизации).

Направление движения с наибольшей крутизной восхождения определяется градиентом

$$\nabla y = (\partial y / \partial x_1) i + (\partial y / \partial x_2) j + \dots + (\partial y / \partial x_k) k,$$

для которого оценками частных производных  $\partial y / \partial x_i$  служат коэффициенты регрессии. При изменении независимых переменных  $x_i$  пропорционально коэффициентам регрессии движение в направлении градиента функции осуществляется по самому крутому пути. Указанный метод получил название «крутого восхождения» [20].

Кроме того, к градиентным относятся методы случайного поиска, случайного поиска с пересчетом, различные варианты метода «Опкон» и др. Однако метод «крутого восхождения» оказывается лучшим как в смысле быстродействия, так и в смысле минимума потерь на рысканье [20].

Градиентные методы построения оптимальных стратегий различаются в основном лишь правилом выбора пробных шагов в начальной точке управления и в тех промежуточных точках, в которых реализованное на  $(i + 1)$ -м шаге состояние существенно отличается от состояния объекта управления, прогнозируемого системой на  $i$ -м шаге для  $(i + 1)$ -го шага.

Помимо очевидных преимуществ (простоты реализации, сравнительно малого объема вычислений), данные методы имеют и ряд недостатков. Самый существенный из них – трудность установления глобального экстремума при наличии на поверхности отклика частных экстремумов. При существующих способах поиска глобального экстремума, базирующихся в основном на случайном выборе исходных точек и направления шагов, требуются большие затраты на поиск и обработку полученных результатов.

*Использование GERT-моделирования.* Для обеспечения автоматизации отображения пространственно-временной картины производственного процесса или хода технологического процесса формирование технологии управления можно реализовать путем динамической интерпретации сетевых управляющих моделей. Сетевая модель может быть основным электронным документом системы формирования циклограмм управления, а результирующая модель разрабатывается на алгоритмическом уровне воспроизведения процессов.

Динамическая интерпретация модели управления позволяет автоматизировать построение пооперационных графиков-циклограмм (включая циклограммы выполнения последовательности технологических операций), формализовать ряд постановок задач оптимизации циклограмм и технологии управления в целом в виде задач оптимизации параметров и структуры эталонной управляющей модели.

Также рассматривается возможность использования стохастического GERT-анализа при формировании технологии управления. Эта возможность связана с использованием сетевых моделей со стохастической структурой, так как нередко на практике именно они оказываются наиболее гибкими и полезными. В нашем случае при анализе реализуемости циклограмм управления стохастическую сеть будем понимать как сеть, которая может быть реализована только при выполнении некоторого подмножества дуг, при этом время выполнения каждой дуги (задачи управления) выбирается в соответствии с вероятностным распределением. В такой стохастической сети для выполнения узла необязательно выполнять все входящие в него дуги, а потому в GERT-модели допускается существование циклов и петель.

### 3.4. ПРОЦЕССЫ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ИНФОРМАЦИИ

Развитие фундаментальных и прикладных наук, технологий требует применения все более мощных и эффективных методов и средств обработки информации. В качестве примера можно привести разработку реалистических математических моделей, которые часто оказываются настолько сложными, что не допускают точного аналитического исследования. Единственная возможность исследования таких моделей, их верификации (подтверждения правильности) и использования для прогноза – компьютерное моделирование, применение методов численного анализа. Кроме того, важной задачей является обработка больших объемов информации в режиме реального времени. Все эти проблемы могут быть решены лишь на достаточно мощной аппаратной базе, с применением эффективных методов программирования.

Мы являемся свидетелями быстрого прогресса вычислительной техники. Производительность современных компьютеров на много порядков превосходит производительность первых ЭВМ и продолжает возрастать заметными темпами. Увеличиваются и другие параметры – объем и быстродействие оперативной и постоянной памяти, скорость передачи данных между компонентами компьютера и т. д., совершенствуется архитектура ЭВМ.

Вместе с тем следует заметить, что уже сейчас прогресс в области микроэлектронных компонент сталкивается с ограничениями, связанными с фундаментальными законами природы. Вряд ли можно надеяться на то, что в ближайшее время основной прогресс в быстродействии электронно-вычислительных машин будет достигнут лишь за счет совершенствования их элементной базы. Переход на качественно новый уровень производительности потребовал от разработчиков ЭВМ и новых архитектурных решений.

Традиционная архитектура ЭВМ была последовательной. Это означало, что в любой момент времени выполнялась только одна операция и только над одним операндом. Совокупность приемов программирования, структур данных, отвечающих последовательной архитектуре компьютера, называется *моделью последовательного программирования*. Ее основными чертами являются применение стандартных языков программирования, достаточно простая переносимость программ с одного компьютера на другой и невысокая производительность.

Появление в середине шестидесятых первого компьютера класса суперЭВМ, разработанного в фирме CDC знаменитым Сеймуром Крэем, ознаменовало рождение новой, векторной, архитектуры. Начиная с этого момента суперкомпьютером принято называть высокопроизводительный векторный компьютер. Главная идея, положенная в основу новой архитектуры, заключалась в распараллеливании процесса обработки данных, когда одна и та же операция применяется одновременно к массиву (вектору) значений. Идея параллелизма оказалась плодотворной и нашла воплощение на разных уровнях функционирования компьютера.

Основными особенностями *модели параллельного программирования* являются высокая эффективность программ, применение специальных приемов программирования и, как следствие, более высокая его трудоемкость, а также проблемы с переносимостью программ.

В настоящее время существуют два основных подхода к распараллеливанию вычислений: параллелизм данных (Data Parallel) и параллелизм задач (Message Passing). В основе обоих подходов лежит распределение вычислительной работы по доступным пользователю процессорам параллельного компьютера. При этом приходится решать разнообразные проблемы. Прежде всего, необходимо обеспечить достаточно равномерную загрузку (сбалансированную работу) процессоров, поскольку если основная вычислительная работа будет ложиться на один из процессоров, мы придем к случаю обычных последовательных вычислений, и никакого выигрыша за счет распараллеливания задачи не будет.

Другая не менее важная проблема – скорость обмена информацией между процессорами. Если вычисления выполняются на высокопроизводительных процессорах, загрузка которых достаточно равномерна, но скорость обмена данными низкая, основная часть времени будет тратиться впустую на ожидание информации, необходимой для дальнейшей работы данного процессора. Рассматриваемые парадигмы программирования различаются методами решения этих двух основных проблем.

Основная идея подхода, основанного на *параллелизме данных*, заключается в том, что одна операция выполняется сразу над всеми элементами массива данных. Различные фрагменты такого массива обрабатываются на векторном процессоре или на разных процессорах параллельной машины. Распределением данных между процессорами занимается программа. Векторизация или распараллеливание при этом чаще всего выполняется уже на этапе компиляции – перевода исход-

ного текста программы в машинные команды. Роль программиста в этом случае обычно сводится к заданию опций векторной или параллельной оптимизации компилятору, директив параллельной компиляции, использованию специализированных языков для параллельных вычислений. Отметим, что наиболее распространенными языками для параллельных вычислений являются высокопроизводительный ФОРТРАН (High Performance FORTRAN) и параллельные версии языка С (например, С\*).

Распараллеливание вычислений посредством параллелизма данных обладает следующими основными особенностями:

- обработкой данных управляет одна программа;
- пространство имен является глобальным, т. е. для программиста существует одна-единственная память, а детали структуры данных, доступа к памяти и межпроцессорного обмена данными от него скрыты.

- синхронизация вычислений на параллельных процессорах слабая, т. е. выполнение команд на разных процессорах происходит, как правило, независимо друг от друга, и только лишь иногда производится согласование выполнения циклов или других программных конструкций – их синхронизация; каждый процессор в данном случае выполняет один и тот же фрагмент программы, но нет гарантии, что в заданный момент времени на всех процессорах выполняется одна и та же машинная команда;

- параллельные операции над элементами массива выполняются одновременно на всех доступных данной программе процессорах.

Из перечисленных особенностей можно увидеть, что в рамках данного подхода от программиста не требуется больших усилий по векторизации или распараллеливанию вычислений. Даже при программировании сложных вычислительных алгоритмов можно использовать библиотеки подпрограмм, специально разработанных с учетом конкретной архитектуры компьютера и оптимизированных для нее.

Применяя при разработке программ подход, основанный на параллелизме данных, используют базовый набор операций, в который входят операции:

- управления данными;
- над массивами в целом и их фрагментами;
- условные;
- приведения;
- сдвига;

- сканирования;
- связанные с пересылкой данных.

Операции управления данными (распределения данных между процессорами) используются в том случае, когда, например, требуется обеспечить равномерную загрузку процессоров. Чем более равномерна эта загрузка, тем более эффективной будет работа компьютера.

Операции над массивами и их фрагментами в качестве аргументов используют массивы в целом или их фрагменты (сечения), при этом данные операции применяются одновременно (параллельно) ко всем элементам массива. Примерами операций такого типа являются вычисление поэлементной суммы массивов, умножение элементов массива на скалярный или векторный множитель и т. д. Операции данного вида могут быть и более сложными – например, вычисление функций от массива.

Условные операции выполняются лишь над теми элементами массива, которые удовлетворяют какому-то определенному условию. В сеточных методах это может быть четный или нечетный номер строки (столбца) сетки или неравенство нулю элементов матрицы.

Операции приведения применяются ко всем элементам массива (или его сечения). Их результатом является одно-единственное значение, например, сумма элементов массива или максимальное значение его элементов.

Операции сдвига требуются для эффективной реализации некоторых параллельных алгоритмов. Примерами таких операций служат алгоритмы обработки изображений, конечно-разностные алгоритмы и некоторые другие.

Операции сканирования также называются префиксными/суффиксными операциями. Префиксная операция, например, суммирование, выполняется следующим образом: элементы массива суммируются последовательно, а результат очередного суммирования заносится в очередную ячейку нового, результирующего массива, причем номер этой ячейки совпадает с числом просуммированных элементов исходного массива.

К операциям пересылки данных относятся, например, операции перемещения данных между массивами разной формы (т. е. имеющими разную размерность и разную протяженность по каждому измерению) и некоторые другие.

При программировании на основе параллелизма данных часто используются специализированные языки – CM FORTRAN, C\*,

FORTRAN+, MPP FORTRAN, Vienna FORTRAN, а также HIGH PERFORMANCE FORTRAN (HPF), основанный на ФОРТРАН-90, что связано с наличием в последнем удобных операций над массивами.

Стиль программирования, основанный на *параллелизме задач*, подразумевает, что вычислительная задача разбивается на несколько относительно самостоятельных подзадач, и каждый процессор загружается своей собственной подзадачей. Компьютер при этом представляет собой MIMD-машину. Аббревиатура MIMD обозначает в известной классификации архитектур ЭВМ-компьютер, выполняющий одновременно большое число различных операций над множеством разнотипных данных. Для каждой подзадачи пишется своя собственная программа на обычном языке программирования (чаще всего это ФОРТРАН или С). Чем больше подзадач, тем большее число процессоров можно использовать, и тем большей эффективности можно добиться. Важно то, что все эти программы должны обмениваться результатами своей работы, практически такой обмен осуществляется вызовом процедур специализированной библиотеки. Программист при этом может контролировать распределение данных между процессорами и подзадачами, а также обмен данными. Очевидно, что в этом случае требуется определенная работа, чтобы обеспечить эффективное совместное выполнение различных программ.

По сравнению с подходом, основанным на параллелизме данных, программирование на базе параллелизма задач является более трудоемким, с ним связан ряд проблем, среди которых можно отметить следующие:

- повышенная трудоемкость разработки программы и ее отладки;
- на программиста ложится вся ответственность за равномерную загрузку процессоров параллельного компьютера;
- программисту приходится минимизировать обмен данными между задачами, так как пересылка данных – процесс, требующий наибольшего количества времени;
- повышенная опасность возникновения тупиковых ситуаций, когда отправленная одной программой посылка с данными не приходит к месту назначения.

Привлекательными особенностями данного подхода являются большая гибкость и большая свобода, предоставляемая программисту в разработке программы, эффективно использующей ресурсы параллельного компьютера и, как следствие, возможность достижения максимального быстродействия. В качестве примера использования подхода, основанного на параллелизме задач, можно привести специали-

зированные библиотеки MPI (Message Passing Interface), разработанная в Аргоннской национальной лаборатории (США), и PVM (Parallel Virtual Machines) – продукт Окриджской национальной лаборатории, университетов штата Теннесси и Эмори (Атланта). Обе библиотеки свободно распространяются в исходных кодах.

Проблема распараллеливания как способа повышения производительности вычислительной системы может быть рассмотрена в двух аспектах, определяющих две конкретные задачи распараллеливания, которые являются взаимообратными [34].

*Задача 1.* В этом случае известен граф предшествования задач и заданы ограничения на время выполнения задач. Необходимо выбрать комплектацию ВС минимальной стоимости. Под стоимостью в данном случае понимают вес каждого типа процессоров, который учитывается при оптимальной комплектации. Как частный случай, весом может быть действительная стоимость, цена ЭВМ. В однородной ВС, т. е. когда все процессоры однотипны, эта задача сводится к задаче нахождения минимального числа процессоров, необходимого для решения требуемого комплекса задач за время, не превышающее заданное.

*Задача 2.* На практике разработчик чаще встречается с задачей именно этого типа. Здесь имеется уже укомплектованная вычислительная система и необходимо сформировать план решения заданного комплекса взаимосвязанных задач за минимальное время. При решении данной задачи могут использоваться такие термины как статическое и динамическое распараллеливание.

При статическом распараллеливании задача загрузки процессоров решается перед началом вычислительного процесса. В этом случае составляется план реализации комплекса задач, который сохраняется при каждом выполнении программы (при изменении состава исполнительных устройств план требует коррекции). Возможность составления плана распараллеливания приводит к желанию использовать те методы, которые дают точное значение минимума времени вычислительного процесса, а они довольно сложны. Знание точных методов необходимо при проектировании управляющих систем, в которых определен набор решаемых задач, а также при оценке выбираемых диспетчеров динамического распараллеливания и обосновании эвристических методов распараллеливания. Методы решения задач распараллеливания представляют отдельный класс задач оптимального планирования.

Динамическое распараллеливание лежит в основе диспетчирования и заключается в оперативном принятии решения о назначении работ и использовании ресурсов ВС в ходе вычислительного процесса. Алгоритмы диспетчирования базируются, как правило, на эвристических методах распараллеливания, в общем случае не обеспечивающих точный результат для всего множества распределяемых задач. Диспетчер, производящий динамическое распараллеливание, должен вырабатывать достаточно близкий к оптимальному план загрузки ресурсов ВС и в то же время быть достаточно быстродействующим. Работа диспетчера должна совершаться на фоне выполнения уже назначенных работ. Эти противоречивые требования налагают условия на выбор решающих правил назначения, лежащих в основе любого диспетчера, и простоту представления данных для него.

Существуют два способа диспетчирования: централизованный и децентрализованный.

Централизованное диспетчирование реализуется управляющим процессором или периодически включаемой в состав очереди заданий задачей наивысшего приоритета. В данном случае существует возможность более полного охвата состояния всех средств ВС и тщательного выбора дальнейшей загрузки процессоров.

Децентрализованное диспетчирование предусматривает возможность самостоятельного обращения каждого процессора к общей очереди для выбора задания. Обращение производится по мере освобождения или по прерыванию. Учет ресурсов всей ВС и связанная с этим оценка различных вариантов назначения отсутствуют, что сокращает затраты времени на диспетчирование, но не позволяет использовать методы оптимального планирования совместной работы процессоров. Децентрализованное диспетчирование обеспечивает высокую надежность ВС. Это следует из независимости состояния каждого процессора от состояния других процессоров, из простоты и неизменности его функции анализа очереди и выбора из нее заданий. Единственной проблемой здесь является возвращение заданий в очередь при выходе из строя или отключении процессоров, которое не позволяет применять децентрализованное диспетчирование в чистом виде.

Итак, для исследования поставленных ранее задач используются модели детерминированного характера. Под этим подразумевается, что вся информация, необходимая для описания характеристик комплекса информационно и по управлению взаимосвязанными задачами, известна еще до решения этих задач в составе ИУС.

Целью результирующего плана исполнения параллельных процессов является оптимизация одного или более оценочных критериев. Основная причина этого в том, что зачастую неудовлетворительная реализация процессов может привести к неприемлемой по отношению к временным ограничениям реакции объекта управления или к недопустимому использованию аппаратно-программных ресурсов ИУС. Показано, что очень часто невозможно или слишком дорого получать наилучшее из возможных решений методами полного перебора. В таких ситуациях должны быть использованы эвристические методы решения. Ниже мы рассмотрим некоторые из этих приближительных решений, а также предложим алгоритм оптимизации сетевой модели методом случайного поиска с пересчетом с переменной величиной шага.

### **3.5. ДЕТЕРМИНИРОВАННЫЕ МОДЕЛИ ФОРМИРОВАНИЯ ПАРАЛЛЕЛЬНЫХ ПРОЦЕССОВ**

#### **3.5.1. Классификация методов планирования параллельных процессов**

Методы планирования параллельных процессов тесно связаны с такой организацией параллельного процесса, которая отражает иерархическую структуру алгоритмов и программно-модульный принцип их отображения. Структура параллельного процесса должна, с одной стороны, показывать структуру задач и алгоритмов, а с другой – обеспечивать простые механизмы управления распараллеливанием.

Хотя проблемы формирования параллельных процессов изучаются уже десятки лет, значительный прорыв в этой области наблюдается только в последние годы. Многие методики формирования процессов, используемые сегодня, были заимствованы из более старых, хорошо известных результатов, полученных в рамках теории управления проектами и исследования операций. Эти исследования были связаны с использованием людей, оборудования и сырья. Если сырье приравнять к компьютерным программам, а людей и оборудование представить в роли процессоров в компьютерных системах, то использование именно теории управления проектами и исследования операций выглядит вполне очевидным. В этом смысле рассматриваемая нами терминология будет основываться на компонентах компьютерных систем, используемых в ИУС или АСУ, и лишь изредка мы будем делать ссылки на аналоги, например, сборочных линий для этих компонентов.

Следует подчеркнуть, что вклад в развитие теории планирования параллельных процессов, сделанный исследователями, не имевшими в своем арсенале компьютерной техники, играет особенно важную роль. В качестве доказательства этого можно отметить, что многие из рассматриваемых результатов были предложены Р. В. Конвэем, В. Л. Максвеллом и Л. В. Миллером в их публикациях по теории планирования, которые в основном базируются на изучении задач календарного планирования производства, т. е. тех задач, которые используют терминологию производства: работа, машина, операция, маршрутизация и время исполнения [35]. А вот книга Б. В. Головкина [36] ориентирована уже на компьютерные системы, и целью ее написания было полное представление последних результатов в компьютерной и операционно-технологической теориях планирования.

Обратимся теперь непосредственно к планированию параллельных процессов. Оно подразумевает, что задания или задачи (сегменты кода, если использовать терминологию ВС и ИУС) должны быть назначены конкретному процессору для исполнения в конкретное время. Поскольку для выполнения может рассматриваться множество задач или заданий (термины взаимозаменяемы), необходимо представить набор этих задач в их взаимосвязи друг с другом. Наиболее популярным в литературе по планированию является представление наборов заданий с использованием ориентированного графа или графа предшествования (другие представления см. в источниках [37, 38]).

Рассмотрим одно из возможных эквивалентных представлений в виде графа множества задач и заданий (рис. 3.3). Узлы в этом графе могут представлять собой независимые операции или части одной программы, которые связаны между собой во времени.

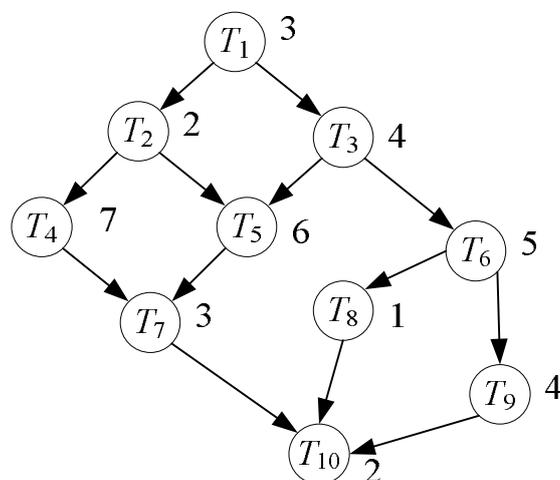


Рис. 3.3. Представление набора задач процесса

На основании данного графа можно сделать несколько важных выводов. Во-первых, множество вершин представляет набор задач  $T = (T_1, \dots, T_R)$ . Направленные дуги между узлами означают, что существует частичное упорядочивание или отношение предшествования «<» между задачами. Так, если  $T_i < T_j$ , задача  $T_i$  должна быть завершена до того, как начнется выполнение  $T_j$ . В рассматриваемом случае  $T_1 < T_2$ ,  $T_1 < T_3$ ,  $T_4 < T_7$  и  $T_5 < T_7$ .

Во-вторых, гипотетическому процессору требуется время на исполнение кода, представленного каждой вершиной. Таким образом, мы говорим о функции  $M: T \rightarrow (0, \infty)$ . Граф программы может быть представлен тройкой  $(T, M, <)$ . Если процессоры идентичны, то любая задача может выполняться на любом процессоре, который удовлетворяет ее условиям предшествования. В рассматриваемом случае никакой информации о количестве процессоров, доступных для выполнения набора задач  $T$ , не дается.

Конечно, число процессоров непосредственно определяет количество времени, требуемое для выполнения задач  $T$ . Однако не совсем верно, что время выполнения обратно пропорционально количеству процессоров. Тем не менее среди многих критериев классификации количество процессоров – единственный наиболее важный фактор в построении оптимальных или частично оптимальных планов реализации параллельных процессов в ИУС.

Возвращаясь к рассмотренному выше графу (см. рис. 3.3), заметим, что он является ациклическим, т. е. в нем нет петель или циклов. Циклы в графе могут нарушить статичность планирования графа (т. е. планирования, сделанного до времени выполнения), так как условие, ограничивающее число итераций, не может быть разрешено до времени выполнения. Большинство опубликованных работ по планированию процессов явно или косвенно игнорируют сложности, представляемые циклами через предположение о том, что полная петля может содержаться внутри единичного узла в графе.

Также обратим внимание на то, что данный граф не содержит условных или решающих узлов. Решающим узлом называется узел, время выполнения которого может влиять на управляющий поток в программе (например, ответвление, зависящее от данных). Если количество исходящих из узла граней  $n$  больше единицы ( $n > 1$ ), то  $m$  узлов, которые непосредственно исходят из рассматриваемого узла, не могут быть инициированы, пока вычисления, представленные в нем, не будут завершены. Аналогично узел со степенью сходимости узла (количеством граней инцидентных узлу) большей, чем 1, должен ждать завершения всех непосредственных предшественников, чтобы быть

иницированным. Методики планирования, рассматриваемые далее, будут основаны на двух условиях, изложенных выше: отсутствии петель и узлов решения.

Добавим, на тему моделирования вычислительных последовательностей, не основывающихся на этих предположениях, было проведено множество исследований. Для более детального изучения данного материала стоит обратиться к работе А. Б. Барского «Параллельные процессы в вычислительных системах. Планирование и организация» [37].

Граф в форме, изображенной на рис. 3.3, относится к связному графу с узлами, содержащими один вход и один выход, или SEC, однако возможен и вариант представления графа с множественным начальным состоянием (рис. 3.4).

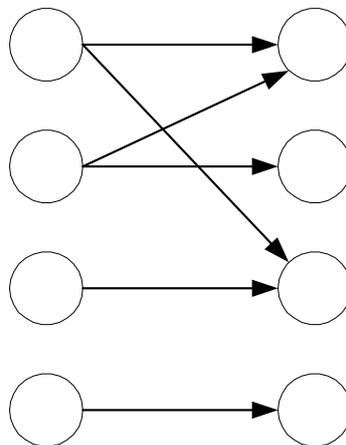


Рис. 3.4. Множество задач с множественным начальным состоянием

Формирование параллельных процессов может осуществляться по разнообразным моделям, основанным на определенных видах классификации, при рассмотрении которых основным для нас является вопрос, должен ли граф программы, реализующей процесс, обрабатываться одним процессором или системой в рамках ИУС, содержащей более одного процессора. Решение разграничить планы реализации процессов именно таким образом не является очевидным ввиду большого количества факторов, которые могут использоваться для классификации. К таким факторам относятся следующие.

*1. Количество процессоров.* Традиционно среди компьютерных систем, управляющих технологическими процессами и системами, доминировали однопроцессорные системы. Тем не менее параллельно шел поиск более высоких вычислительных возможностей, в частно-

сти, за счет использования нескольких процессоров, так как однопоточная архитектура управляющей ЭВМ постепенно себя изживала. Внедрение интеграции высокого уровня и необходимость повышения надежности вычислительных систем явились стимулом к использованию многопроцессорной организации. Однако переборные оптимальные планы были созданы только в ограниченном количестве случаев.

2. *Продолжительность задачи.* Выше уже говорилось о том, что узлы графа программы могут представлять задачи равной и неравной продолжительности. В первом случае можно сказать, что на все модули накладывается ограничение единичной продолжительности выполнения задания. Термин «единичный» используется здесь как время, необходимое для выполнения некоторого количества инструкций. В практике обработки комплекса информационно и по управлению взаимосвязанных задач разной продолжительности обычным является условие, что все задачи могут быть разделены на целые кратные числа наименьших базовых заданий. Тем не менее нередко программные графы, для которых деление задач неприемлемо, рассматриваются как отдельная категория.

3. *Структура графа предшествования.* Отдельные узлы в графе связаны друг с другом различными способами. Например, все задачи могут быть независимы друг от друга, и в этом случае мы говорим, что между ними не существует предшествования или частичной предопределенности. В других ситуациях структуру программного графа необходимо строить таким образом, чтобы каждый его узел имел хотя бы одного предшественника или последователя. Также существует возможность построения общей структуры предшествования, для которой не применяются предыдущие ограничения.

4. *Прерывание задач.* Если допустимо прерывание (и, следовательно, возобновление) задачи до ее завершения, то речь идет о плане с приоритетами (вытесняющий план реализации процесса), если же прерывание до завершения задачи недопустимо – о процессе без приоритетов или простом плане. В общем случае вытесняющие методы формируют планы, лучшие по сравнению с планами, разработанными с использованием невытесняющих алгоритмов, но тем не менее имеющие и некоторые недостатки, связанные с дополнительными расходами на «переключатели» задачи, состоящие из системы прерывания вычислений и дополнительной памяти, необходимой для хранения состояния прерванной задачи. Эти накладные расходы могут быть незначительны, если прерывания происходят не очень часто; в условиях же, когда вытеснения проявляются чаще, может стать заметным снижение производительности вычислительного комплекса ИУС.

5. *Простой процессора.* Величина производительности может быть увеличена за счет использования холостого хода процессора. Определение момента отключения процессора может привести к значительным усложнениям алгоритма планирования реального времени в ИУС. В «прожорливой» системе места в плане для холостого хода не выделяются, и отложенная задача начинается, как только освобождается процессор.

6. *Периодичность работы.* Большая часть рассматриваемых ранее исследований связана только с однократным выполнением задач, которые тем не менее через неопределенные интервалы могут выполняться повторно на протяжении большого периода времени. Многократные выполнения заданий в составе оптимальных или почти оптимальных планов могут в результате дать выигрыш по времени даже с учетом затрат на анализ задачи формирования этих планов. В течение этого времени код заданий не изменяется, хотя изменения в данных, обрабатываемых кодом, допустимы и скорее представляют правило, чем исключение. Несмотря на это, оценивание производительности комплекса информационно и по управлению взаимосвязанных задач периодического процесса рассматривается только для однократного их выполнения.

В настоящее время в вычислительных комплексах ИУС используются как одно-, так и двухпроцессорные системы. ВЕС могут быть охарактеризованы множеством задач, для каждой из которых известны частота и время выполнения. Процесс планирования в таких системах особенно затруднителен по двум причинам: временные и частотные ограничения могут быть различны для каждой задачи на множестве периодических задач, а допустимые отклонения от запланированного времени запуска (и, следовательно, времени завершения) должны либо быть очень малыми, либо вообще отсутствовать на каждой итерации каждой задачи.

7. *Наличие и отсутствие пределов.* Для оценки поведения планов был разработан ряд показателей производительности. В большинстве случаев рассматривается поведение только полного плана или полного множества задач. В других случаях пределы или запланированные времена завершения (далее используются оба термина) устанавливаются для индивидуальных компонентов множества взаимосвязанных задач. Если есть некоторый дополнительный временной показатель, связанный со временем завершения отдельных задач (время простоя) и имеющий ограничение, то речь идет о жестком пределе или о жестком плане реального времени. Если же время простоя –

статистически распределенная величина, можно говорить о мягком пределе или о мягком плане реального времени. Планы с ограничениями появляются в связи с периодическими планами реализации задач в ИУС и определением нормативных времен процессов.

8. *Планы с ограниченными ресурсами.* В большинстве работ по планированию предполагается наличие неограниченных ресурсов для поддержки мультипроцессорных систем на стадии их выполнения. Хотя обычно это не упоминается, но сами процессоры были косвенно приняты в состав некоторого (обычно неограниченного) класса ресурсов.

В последнее время для каждой из задач, входящих в состав системы, принято определять требования на определенные классы ресурсов [38]. Если не принимать во внимание процессор, то первоочередной тип ресурсов, который важен и актуален – память. Во многих работах предполагается наличие достаточного объема памяти для хранения кода и данных, необходимых для выполнения каждой из задач на конкретном процессоре. В системах, где между задачами разделяется единственная главная память, подразумевается, что объем общей требуемой памяти для множества задач не превышает размера главной. С появлением распределенных систем, в которых процессору может быть доступна как локальная, так и распределенная память, вопрос о распределении ресурсов становится особенно важным. Конечно, память не единственный системный ресурс, который может быть доступен в ограниченных количествах. Теория моделей с ограниченными ресурсами может быть расширена за счет включения множества других ресурсов.

9. *Гомогенные и гетерогенные процессоры.* Наряду с исследованиями планов с ограниченными ресурсами рассмотрение неоднородных или гетерогенных многопроцессорных систем представляет собой новейшую область исследований. Вопрос внедрения неоднородных процессоров становится особенно актуальным, поскольку многопроцессорные системы занимают все большую нишу в индустрии обработки данных и управления в ИУС. Наличие различных процессоров в вычислительном комплексе ИУС означает, что возможны усовершенствования систем управления с использованием современных компонент, т. е. процессор (например, неисправный) может быть заменен или добавлен в систему без всяких ограничений на замену и без добавления точной копии изначального оборудования. Это особенно важно, если меньшие по размерам, более дешевые и подходя-

щие элементы доступны для монтирования на исходном оборудовании, что обычно и имеет место. Это способствует развитию принципов реконфигурации ИУС.

*10. Показатели эффективности.* Как уже говорилось ранее, для оценивания эффективности сформированных планов реализации процессов был предложен ряд показателей. Наиболее часто используются следующие пять (перечислены в порядке популярности):

- время окончания или завершения;
- количество используемых процессоров;
- среднее время потока;
- загрузка процессора;
- время простоя процессора.

Чтобы проиллюстрировать использование показателей эффективности, рассмотрим план с временными диаграммами, известными как диаграммы Ганта (рис. 3.5). Для реализации этого плана требуются три процессора. Задачи, назначенные каждому процессору, порядок их выполнения и временные ограничения представлены горизонтальными линиями и каждому процессору соответствуют свои идентификаторы задач.

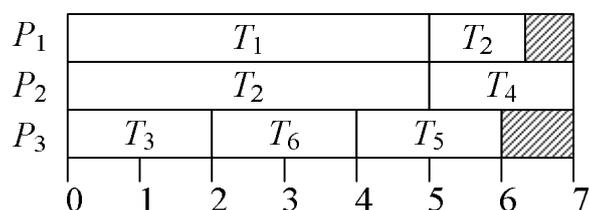


Рис. 3.5. План задач в виде диаграммы Ганта

Время завершения, обозначаемое здесь как  $w$ , для данного плана равно 7, время потока задачи – времени, за которое ее выполнение завершается, а время потока плана определяется как сумма времен потоков всех задач в плане. Например, времена потоков задач  $T_1$  и  $T_4$  равны 7 и 4 соответственно, в то время как время потока плана равно 25,5. Среднее время потока получается делением времени потока на количество задач в плане. Загрузка (критерий использования)  $P_1$ ,  $P_2$  и  $P_3$  имеет значения 0,93, 1,00 и 0,86 соответственно (они получены делением времени, в течение которого процессор был занят, на общее время, в течение которого он был доступен для выполнения). Время простоя  $P_1$ ,  $P_2$  и  $P_3$  равно 0,5, 0,0 и 1,0 соответственно.

Минимизации времени выполнения способствует повышению пропускной способности системы, которая определяется как число

наборов задач, обработанных в единицу времени (например, за час), и, следовательно, обратно пропорциональна сумме времен вычислений отдельных наборов задач.

Минимизация количества необходимых процессоров имеет смысл, по крайней мере, по двум причинам. Первой, наиболее очевидной, является цена. Вторая же заключается в следующем: если количество процессоров, необходимых для выполнения задач в данное время меньше, чем количество всех доступных, то оставшиеся процессоры могут использоваться как резервные для увеличения надежности системы или как фоновые для некритичных вычислений.

Минимизация среднего времени потока связана с временным диапазоном, в течение которого задачи используют системные ресурсы, отличные от процессора, например, память. Чем короче время, в течение которого некоторые задачи работают с памятью, тем дольше память доступна другим задачам. Аналогией в производстве является количество складского пространства, занятого сырьем, которое должно быть переработано в конечный продукт. Время потока является косвенным показателем производительности системы.

*11. Эффективность алгоритмов.* Ключевым вопросом в изучении планирования процессов является количество вычислительного времени, необходимого для формирования соответствующего плана. В нашем случае будем говорить, что алгоритм является эффективным, если число перебираемых альтернатив может быть выражено в виде полиномиальной зависимости. Неэффективным же алгоритмом станет тот, который прежде чем отобрать лучшее решение требует перечисления практически всех возможных альтернатив. Сложность алгоритмов данного типа выражается экспоненциальной зависимостью.

Необходимо отметить, что универсальных алгоритмов для большинства задач, представляющих интерес в теории планирования и формирования процессов в ИУС, не разработано. Фактически известно, что если удастся построить эффективный алгоритм для некоторых задач, то он, скорее всего, будет эффективен на некотором классе однотипных *NP*-сложных задач. Характеризуя задачу как *NP*-сложную, мы подразумеваем, что она может быть сложна в вычислении, по крайней мере, как сложнейшая задача *NP*-семейства, которое является семейством задач, решаемых недетерминированными алгоритмами в полиномиальное время. Это множество включает, например, такие задачи как определение, будет ли предложенная формула выполнимой, будет ли граф обладать глубиной заданного размера,

а также задачи коммивояжера. Детерминированный алгоритм, который решал бы любую из этих задач за полиномиальное время, до сих пор не найден.

### 3.5.2. Однопроцессорные модели

Важнейшей особенностью однопроцессорных планов можно назвать следующую: в таких моделях все задачи – кандидаты в формируемый процесс – доступны для выполнения одновременно. Для каждой из задач известны ее точные характеристики, которые остаются постоянными на протяжении всего жизненного цикла задачи, и указан частный критерий качества работы, например, минимизация максимального времени завершения. Эти планы не включают разновидность задач типа мультипрограммных и компьютерных систем с разделением времени, поскольку точные характеристики задач, обрабатываемых такими системами, заранее неизвестны. Представленные результаты могут использоваться в одной из двух областей применения ИУС: технологических процессах в серийном производстве или циклических системах управления оборудованием. В обоих случаях задачи обычно рассматриваются как независимые. При этом решения, получаемые в среде управления циклическими технологическими процессами, имеют периодичный характер.

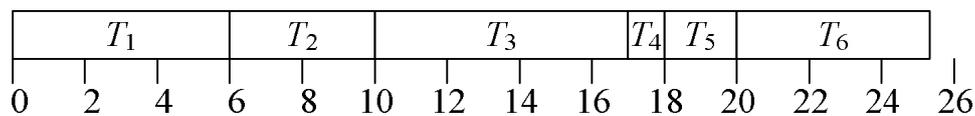
**Решения, полученные для производственных систем.** Среда, состоящая из  $n$  одновременно доступных задач с известными характеристиками и одной машины (процессора), рассматривается Конвэем, Максвеллом и Миллером как простейшая задача планирования. В своих работах они предлагают несколько интересных результатов для формирования планов этого типа процессов.

В задаче формирования процесса из  $n$  независимых задач на одном процессоре нет необходимости рассматривать планы с приоритетным прерыванием или с включением времени холостого хода. Показатель производительности такой системы не может быть улучшен ни приоритетным прерыванием (и впоследствии возобновлением) выполнения задачи, ни включением холостого хода процессора до завершения  $n$  задач. Такой критерий качества – время выполнения всех заданий – должен быть как можно меньше по своей абсолютной величине, однако при увеличении времени обслуживания любой из задач он, напротив, возрастает.

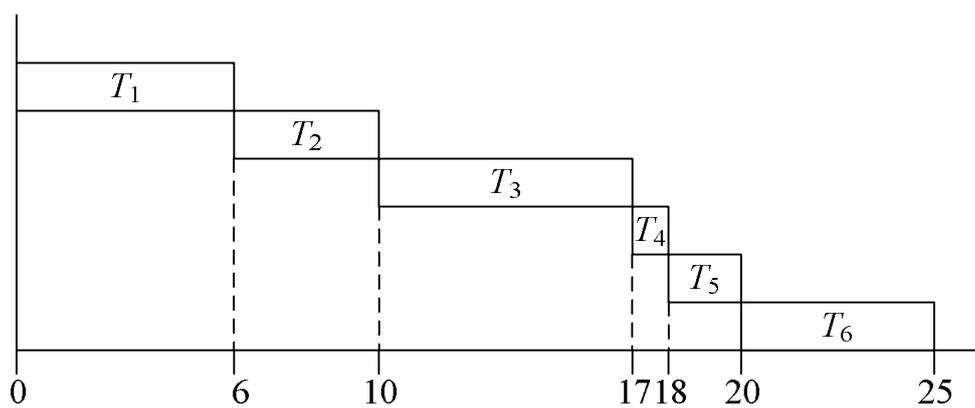
Величина максимального времени потока для таких планов вычисляется как сумма  $n$  времен завершения и одинакова для каждого

из  $n!$  возможных расписаний. Среднее время потока плана данного вида минимизируется чередованием работ в порядке неубывающего времени обслуживания. Такой способ планирования назван упорядочивание наикратчайшего времени обработки (Shortest Processing Time Sequencing, SPT [35]) и характеризуется как наиболее важная концепция в вопросе планирования процессов.

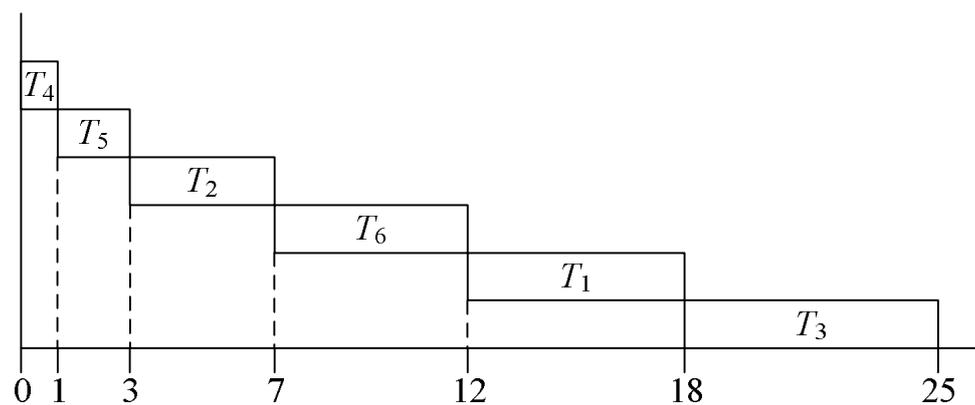
Чтобы продемонстрировать идею подхода, рассмотрим задачу формирования однопроцессорного плана реализации шести независимых заданий (рис. 3.6), представленную диаграммой Ганта для одного из 6 возможных планов для этого набора заданий (рис. 3.6).



*a*



*б*



*в*

Рис. 3.6. Определение времени потока  
(по Ганту, альтернативное представление SPT):  
*a* – представление по Ганту; *б* – альтернативное представление;  
*в* – планировщик кратчайшего времени обработки

Из графика видно, что максимальное время потока равно 25, как и общее время обслуживания. Для определения среднего времени потока  $F$  наиболее удобно представить план в форме, показанной на рис. 3.6, б. Общая площадь этого графика, включая отмеченные блоки и площадь под блоками, представляет собой сумму времен потоков задач.

Время потока  $i$ -й задачи  $F_i$  в последовательности определяется в виде

$$F_i = \sum_{j=1}^n p(f),$$

где  $p(f)$  – время обработки задачи, занимающей  $j$ -ю позицию в плане.

Другими словами, время потока задачи – это просто время завершения выполнения задачи.

Среднее время потока можно рассчитать по формуле

$$\bar{F} = \left( \sum_{i=1}^n F_j \right) / n.$$

Например, на рис. 3.6, б,  $F = 16$ . Если представленные там задачи перегруппировать в форму SPT-плана (рис. 3.6, в), то  $F = 11$ .

Отсюда можно сделать следующий вывод: упорядочивание наибольшего времени обработки (Longest Processing Time Sequencing, LPT) максимизирует то, что SPT минимизирует. Процедуры планирования, формирующие противоположные последовательности, называются противоположными.

В соответствующих работах Максвелл, Конвэй и Миллер также приводят результаты для случаев, когда 1) дается только среднее, ожидаемое или оценочное время обработки; 2) приоритеты назначены отдельным задачам; 3) все задачи доступны не одновременно, а прибывают периодически, с перерывами.

**Мультипрограммирование с жесткими ограничениями.** Говоря о мультипрограммировании с жесткими ограничениями, мы подразумеваем так называемое критичное по времени управление циклическими/периодическими процессами в ИУС. Процесс называется критичным по времени, если для задач, входящих в него, определены частоты и времена выполнения. Каждый запуск задачи должен происходить в границах временного интервала, определенного частотой, либо совершаться по сигналу внешнего прерывания. Набор этих про-

цессов может представлять вычисления, которые необходимо выполнить для управления конкретной системой реального времени. Примером такой реальной среды является среда автоматизированной системы технической диагностики электронных устройств.

С точки зрения вычислительной загрузки, отличие систем такого рода от обычной мультипрограммной системы заключается в том, что заранее известен характер необходимых вычислений. Кроме того, реакция системы обязательно должна происходить внутри строго установленных пределов. Также недопустимо говорить (в отличие от типичной системы), что «большинство откликов ИУС должны произойти в течение  $x$  секунд». Два независимых исследования, проведенных в этой области (О. Серлин, Ц. Л. Лью и Дж. В. Лайленд), дали очень похожие результаты.

Рассмотрим критичный по времени процесс (Time Critical Process, TSP) (рис. 3.7). В данной модели  $E$  представляет назначенное время выполнения одной итерации процесса, время  $T$ , иногда называемое структурным временем, – период повторения, т. е. период между последовательными вхождениями сигнала, связанного с процессом, а  $d$  – предельное время вычисления. На практике  $d$  обычно равно  $T$ .

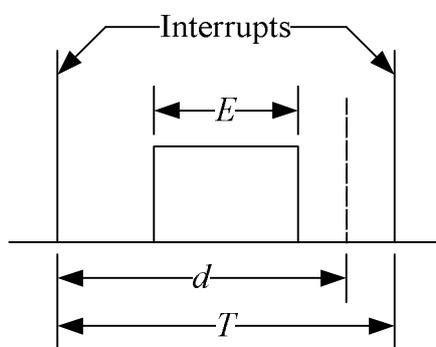


Рис. 3.7. Модель процесса, критичного по времени

Состояние, когда сигнал прерывания инициирует итерацию TSP до того, как заканчивается предыдущая, называется перегрузкой. Рассматриваемая здесь проблема заключается в планировании последовательности TSP на одном процессоре с гарантированным отсутствием перегрузок. Если перегрузок никогда не происходит, план называется допустимым.

В рассматриваемой модели предполагается, что все задачи независимы и не выполняют никакого ввода–вывода. Поскольку некоторые

из них имеют более высокую частоту выполнения, иногда будет необходимо прерывать и соответственно восстанавливать выполнение задач с низшей частотой (т. е. задачи с низкой частотой можно назвать вытесняемыми), чтобы гарантировать своевременное завершение задач с большей частотой выполнения. В качестве примера приведем две задачи:  $T_1(T_1 = 2, E_1 = 1)$  и  $T_2(T_2 = 0,5, E_2 = 1)$ , с  $T_1$ , имеющим больший приоритет (рис. 3.8). На рис. 3.8, *а* показано допустимое назначение, а из рис. 3.8, *б* видно, что значение  $E_2$  может быть увеличено до 2. Если  $T_2$  имеет больший приоритет, то ни  $E_1$ , ни  $E_2$  не могут принимать значение, большее 1 (рис. 3.8, *в*).

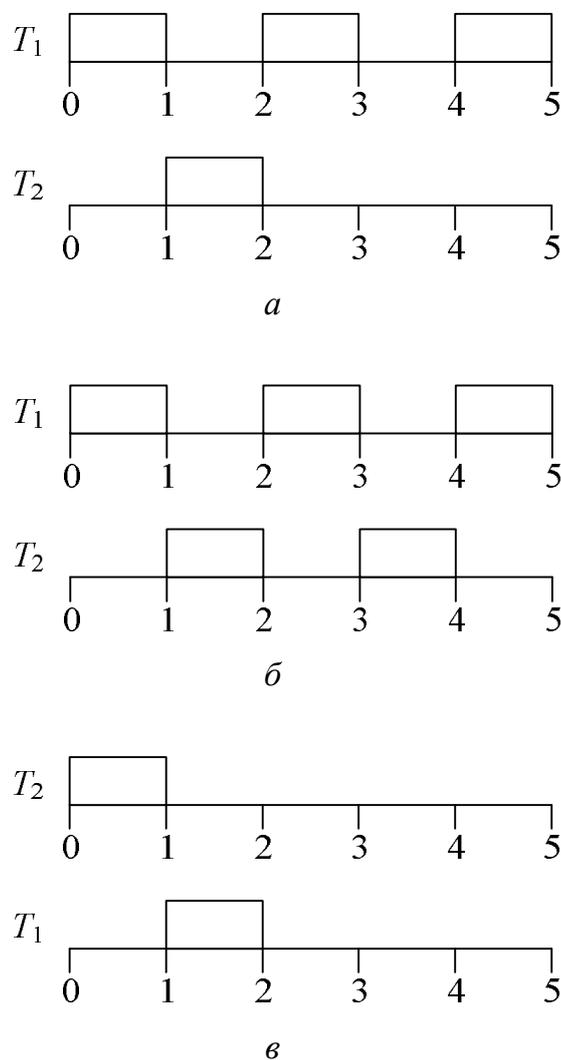


Рис. 3.8. Планы для двух ТСП-процессов:  
*а* – возможное значение, когда  $T_1$  имеет более высокий приоритет чем  $T_2$ ; *б* – другое возможное значение, когда  $T_1$  имеет более высокий приоритет чем  $T_2$ ; *в* – единственное возможное значение, если  $T_2$  имеет более высокий приоритет чем  $T_1$

Согласно Серлину, эффективным алгоритмом распределения является тот, который предоставляет достаточно процессорного времени ТСП для своевременного выполнения при уменьшении длительности вынужденных простоев, т. е. времени, в течение которого работа процессора должна быть приостановлена для разрешения внутренних конфликтов. Лью и Лайленд пытались найти наибольший возможный коэффициент использования при условии завершения всех задач в рамках временных ограничений. Коэффициент использования  $U$ , названный Серлином коэффициентом загрузки, для  $n$  ТСП процессов определяется в виде

$$U = \sum_{i=1}^n E_i / t_i .$$

Серлин, Лью и Лайленд получили одинаковые оптимальные решения для схемы с фиксированными приоритетами, в которой задание с частотой  $f_i$  имеет более высокий приоритет, чем задание с частотой  $f_j$ , если  $f_i > f_j$ . Лью и Лайленд назвали такую схему с фиксированными приоритетами назначениями с монотонными приоритетами (Rate Monotonic Priority, RMP), а Серлин – интеллектуальным алгоритмом с фиксированными приоритетами (Intelligent Fixed Priority (IFP) Algorithm). Несмотря на различное название схемы, все три исследователя показали, что верхняя граница коэффициента использования определяется в виде

$$U = n(2^{1/n} - 1),$$

где  $n$  – количество ТСП-процессов.

Этот результат означает, что допустимая сумма факторов индивидуальной загрузки должна быть значительно меньше 1, чтобы гарантировать, что каждый процесс будет завершен вовремя. При большом количестве задач процессор должен быть недозагружен более чем на 30 %. Эта схема является оптимальной с условием, при котором для некоторого набора задач план не может быть сформирован никаким правилом назначения с фиксированными приоритетами, если он не может быть сформирован IFP- или RMP-алгоритмами.

Рассмотренное правило назначения является фиксированным, или статичным, относительные приоритеты задач в нем основаны на частотах задач и не меняются во время выполнения. Однако существуют и динамические алгоритмы, позволяющие менять приоритеты и допускающие 100%-ную загрузку процессора. Так, Лью и Лайленд сформулировали правило, назвав его алгоритмом планирования с уче-

том пределов (Deadline-Driven Scheduling Algorithm), а Серлин рассмотрел похожий алгоритм, разработанный Файнбергом. В обоих алгоритмах приоритеты переоцениваются каждый раз, когда в систему поступает инициирующее прерывание. При этом наивысший приоритет отдается задаче, чей предел ближе всего, а низший – задаче, чей предел является самым дальним на данный момент времени. Это применимо только к тем задачам, чьи вычисления в данном фрейме еще не завершились. Другой исследователь, Э. Г. Кофман, рассматривает алгоритм относительной срочности, в котором приоритеты переоцениваются в каждый момент времени.

Серлин также говорит об алгоритме минимального временного квантования (Minimal Time Slicing (MTS) Algorithm), основанного на понятии интервалов планирования. *Интервал планирования* – это время между прерыванием и появлением первого предела после него. На протяжении этого интервала каждое незавершенное задание может использовать процессор в монопольном режиме; продолжительность режима для некоторой задачи прямо пропорциональна относительной части загрузки процессора, приходящейся на долю этой задачи. Такой подход гарантирует своевременное выполнение всех задач, но его успех зависит от незначительного времени контекстного переключения.

Кроме того, Лью и Лайленд рассматривают смешанный алгоритм планирования, который является комбинацией алгоритмов с фиксированными и динамическими приоритетами. Для  $n$  задач ( $n > k$ )  $k$  задач, имеющих наикратчайшие периоды, планируется в соответствии с монотонным алгоритмом планирования с фиксированными уровнями приоритетов, а оставшиеся  $n - k$  задач – с использованием алгоритма планирования с учетом пределов, когда процессор не занят первыми  $k$  задачами. По мнению авторов, данная методика не всегда позволяет достигать 100%-ной загрузки процессора, но тем не менее обладает большинством преимуществ алгоритма планирования с учетом пределов. В то же время она достаточно легко реализуема, поскольку статическое планирование  $k$  задач совместимо с механизмом прерываний, который действует как планировщик с фиксированными приоритетами.

### 5.3.3. Конвейерные модели

После рассмотрения однопроцессорных планов, казалось бы, логично обратиться к многопроцессорным. Однако существует класс планов, в которых более чем один процессор включен в совместное выполнение ряда задач, и существует последовательная взаимосвязь

между этими процессорами. Это так называемые конвейерные планы, случаем многопроцессорного планирования не являющиеся. В таких планах задача, которую необходимо выполнить, обслуживается каждым из процессоров по очереди. Это чередование должно соблюдаться для всех задач, входящих в план, но никакого требования идентичности процессоров не вводится.

Своим происхождением конвейерные планы обязаны производственной среде, где работа построена на использовании целого ряда машин, каждая из которых выполняет уникальные, специфические операции. В компьютерной среде аналогом этого процесса является требование для задач монопольного использования процессора и каналов ввода–вывода. Чередование этих режимов монопольного использования соответствует прохождению задачи через ряд машин, количество которых невелико.

По мнению Конвэя, Максвелла и Миллера, наиболее часто цитируемым является решение С. М. Джонсоном задачи о двухмашинном конвейере. Алгоритм Джонсона упорядочивает  $n$  задач, одновременно доступных на двухмашинном конвейере, таким образом, чтобы минимизировать максимальное время потока. Используя терминологию Конвэя, Максвелла, и Миллера, а также систему обозначений, принятую у Джонсона, можно сказать, что каждая задача состоит из пары операций  $(A_i, B_i)$ , где  $A_i$  и  $B_i$  – работы, которые необходимо выполнить на первой и второй машинах конвейера соответственно. Именно в таком порядке задачи будут выполняться на двух машинах, хотя для некоторых  $A_i$  и  $B_i$  допустимо принимать нулевые значения, так как некоторые задания могут выполняться за одну операцию. Предполагается, что каждая машина может обслуживать за раз только одну задачу, и действие  $A_i$  должно быть закончено прежде, чем может быть начато действие  $B_i$ .

Учитывая, что дано  $n$  пар в форме  $(A_i, B_i)$ , задачей можно поставить упорядочение  $n$  заданий с целью минимизации максимального времени потока. Джонсон показал, что задача  $J_k$  должна предшествовать задаче  $J_l$ , если  $\min(A_k, B_l) < \min(A_l, B_k)$ .

Рассмотрим следующий пример. Пусть даны характеристики задач, которые должны быть спланированы (табл. 3.5). По табл. 3.6 мы можем увидеть, что  $T_2 < T_3$  и  $T_4 < T_5$ , поскольку выполняются следующие условия:

$$\min(A_2, B_3) = \min(0, 4) = 0 < \min(A_3, B_2) = \min(5, 2) = 2,$$

$$\min(A_4, B_5) = \min(8, 1) = 1 < \min(A_5, B_4) = \min(2, 6) = 2.$$

Также  $T_2 < T_4$  и  $T_2 < T_5$ , так как имеем:

$$\min(A_2, B_4) = \min(0, 6) = 0 < \min(A_4, B_2) = \min(8, 2) = 2,$$

$$\min(A_2, B_5) = \min(0, 1) = 0 < \min(A_5, B_2) = \min(2, 2) = 2.$$

Кроме того,  $T_4 < T_3$  и  $T_3 < T_5$ .

Таблица 3.5

### Характеристики планируемых задач

Номер задачи, $i$	$A_i$	$B_i$
1	6	3
2	0	2
3	5	4
4	8	6
5	2	1

Таблица 3.6

### Минимальные времена потока для разных пар задач из таблицы 3.5

$j$	$(A_j, B_{j+1})$	Min	$(A_{j+1}, B_j)$	Min
1	$(A_1, B_2) = (6, 2)$	2	$(A_2, B_1) = (0, 3)$	0
2	$(A_2, B_3) = (0, 4)$	0	$(A_3, B_2) = (5, 2)$	2
3	$(A_3, B_4) = (5, 8)$	5	$(A_4, B_3) = (8, 4)$	4
4	$(A_4, B_5) = (8, 1)$	1	$(A_5, B_4) = (2, 8)$	2

В итоге находим, что должна быть установлена следующая очередность:  $T_2, T_4, T_3, T_5$ . Единственная не включенная в план задача –  $T_1$ . Из результатов работы алгоритма Джонсона следует, что  $T_2 < T_1$ ,  $T_3 < T_1$ ,  $T_4 < T_1$  и  $T_1 < T_5$ . То есть единственная позиция для  $T_1$ , которая удовлетворяет этим отношениям предшествования и чередования  $T_2, T_4, T_3, T_5$  – та, в которой  $T_3 < T_1$  и  $T_1 < T_5$ , и конечное упорядочивание будет иметь вид  $T_2, T_4, T_3, T_1, T_5$  (рис. 3.9). Такой план выдает минимальное время потока, равное 23 (рис. 3.9, а). Кроме того, по приведенным данным можно составить SPT-план на основе  $A_i$  (рис. 3.9, б) и план, основанный на очередности, в которой задачи появляются в начальной таблице (рис. 3.9, в).

Конвей, Максвелл и Миллер показали, что если не принимать во внимание методы математического программирования, то не существует эффективных алгоритмов минимизации среднего времени потока для задачи двухмашинного конвейера, аналогичных алгоритму Джонсона (т. е. задача является *NP*-трудной). Использование метода ветвей и границ для решения этой задачи удваивает число необходимых вычислений при каждом добавлении в систему одного задания. Авторы отмечают, что  $2^n$  уровень роста числа вычислений все-таки лучше, чем  $n!$  вычислений, которые бы потребовались для полного перебора. С целью минимизации максимального времени потока для трехмашинного конвейера с различной степенью успеха использовались методы математического программирования.

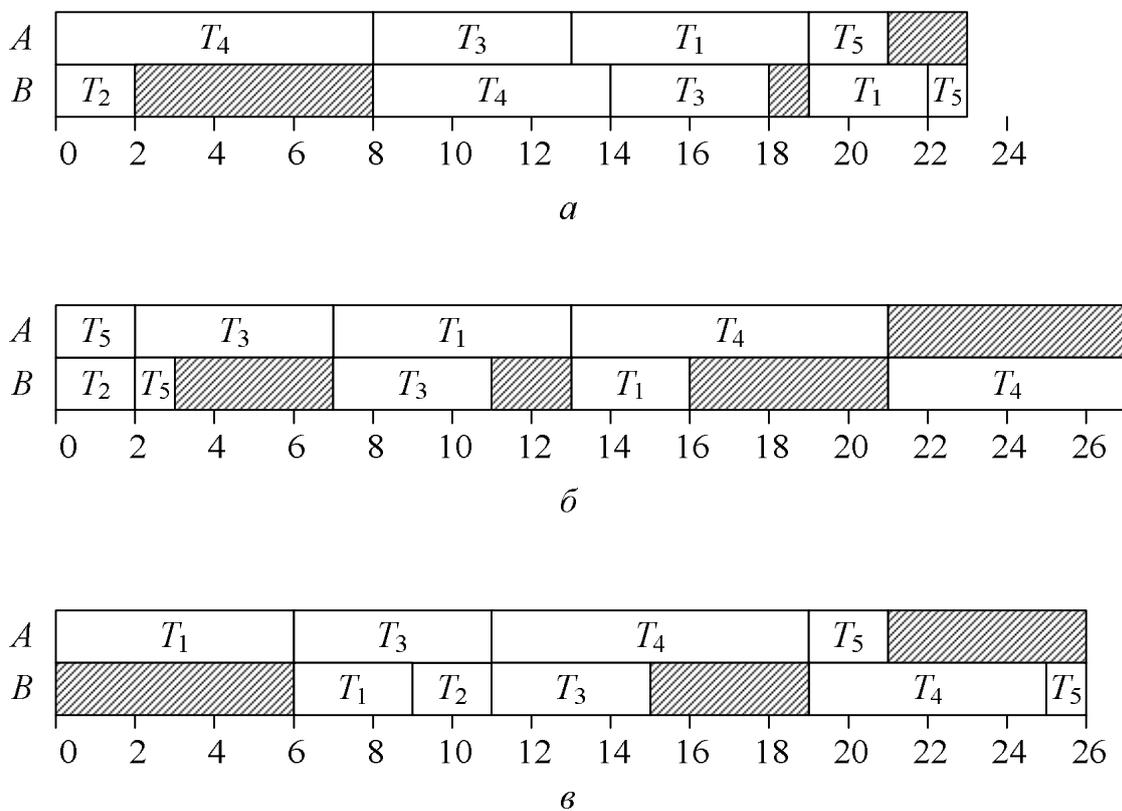


Рис. 3.9. Алгоритм Джонсона для задач из табл. 3.5:  
*a* – оптимальная последовательность  $F_{\max} = 23$ ;  
*б* – STP-последовательность на  $A_i$ ;  $F_{\max} = 27$ ;  
*в* – последовательность в порядке  $(T_1, T_2, T_3, T_4, T_5)$ ;  $F_{\max} = 26$

Предыдущие результаты были обобщены для ситуации, когда более чем один процессор может существовать в каждом из двух классов – классе *A* и классе *B*. В стратегии планирования «больше и раньше» (More And Earlier, ME) В. Ю. Шен и Ю. Е. Чен рассмотрели

систему с  $m$  процессорами класса  $A$  и  $n$  процессорами класса  $B$  с целью минимизации максимального времени выполнения. Авторы показали, что, хотя стратегия МЕ не оптимальна, она проста и достаточно хорошо работает. В МЕ частичное упорядочивание определено таким образом, что задача  $T_i$  предшествует  $T_j$ , если  $A_i + B_i \geq A_j + B_j$  и  $A_i \leq A_j$ , где  $A_i$  и  $B_i$  представляют требования  $T_i$  для процессора класса  $A$  и класса  $B$  соответственно.

В более поздней работе Р. Е. Бутен и В. Ю. Шен отбросили ограничение, что задача  $T_i$  должна предшествовать задаче  $T_j$ , если  $A_i + B_i \geq A_j + B_j$  и  $A_i \leq A_j$ . Вместо этого они предположили так называемое модифицированное упорядочение Джонсона (a Modified Johnson Ordering, MJO). Следовательно, упорядочивание Джонсона (the Johnson Ordering, JO) основано на вышеупомянутом алгоритме Джонсона. В конвейерной среде с  $m$  процессорами типа  $A$  и  $n$  процессорами типа  $B$  в соответствии с MJO  $T_i < T_j$ , если  $\min(A_i/m, B_j/n) < \min(A_j/m, B_i/n)$ . Авторы разработали две теоремы, которые описывают верхние и нижние границы для MJO подхода [39].

В задачах конвейерного типа используется предположение, что в случае, когда задача ожидает обслуживания по причине занятости машины, выделяется необходимое количество памяти для хранения промежуточных данных. Для компьютерных систем это предположение может не иметь силы вследствие того, что ресурсы промежуточной памяти состоят из различного рода буферов, поскольку задание по ходу своего выполнения может прогрессировать из главной памяти к процессору и к устройствам ввода–вывода. С. С. Редди и Ц. В. Рамамурти исследовали конвейерные планы, которые не основывались на предположении о бесконечности промежуточной памяти. Предполагалось, что такая среда имеет ограниченную промежуточную память (FSFIS) в отличие от среды с бесконечной промежуточной памятью (FSIIS). В качестве первого шага к решению FSFIS-задачи Редди и Рамамурти нашли решение для задачи без промежуточной памяти (FSNIS).

Поясним эффект промежуточной памяти посредством графического изображения планов (рис. 3.10, а) В FSIIS-среде минимальное общее время завершения соответствует плану  $(J_1, J_3, J_2)$  (рис. 3.10, б). Однако если предположить, что промежуточной памяти нет, упорядочивание  $(J_1, J_3, J_2)$  дает план, требующий 52 единицы времени (рис. 3.11, в) Оптимальным FSNIS-планом является упорядочение  $(J_1, J_2, J_3)$ , для выполнения которого требуется 45 единиц времени (рис. 3.10, г).

Рассмотренная здесь задача фактически ослабляет действие нескольких ограничений, используемых в алгоритме Джонсона. Во-первых, допускается наличие более двух машин, а во-вторых, предполагается, что объем доступной промежуточной памяти равен нулю. Как уже говорилось, Конвэй, Максвелл и Миллер показали, что не существует эффективного алгоритма для решения конвейерной задачи, ослабляющей ограничения Джонсона. FSNIS-задача, исследованная Редди и Рамамурти, не является исключением. Тем не менее П. Ц. Гилмор и Р. Е. Конрой разработали алгоритм с полиномиальным временем для этой среды «без ожидания», когда количество машин ограничено двумя.

Processor	Job		
	3	30	2
	10	4	5
	3	5	4

*a*

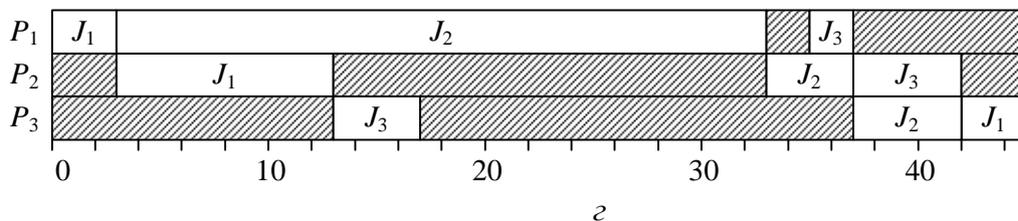
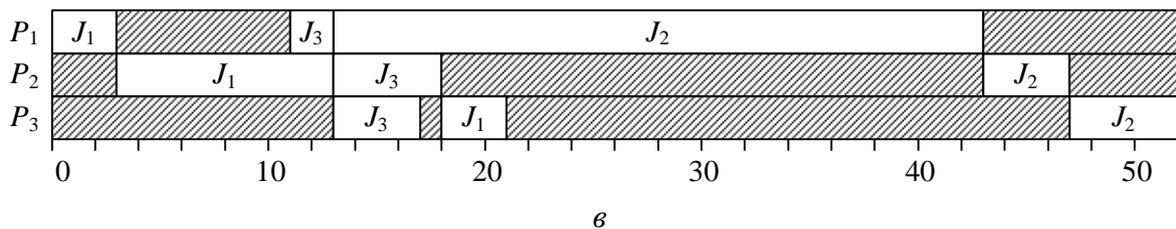
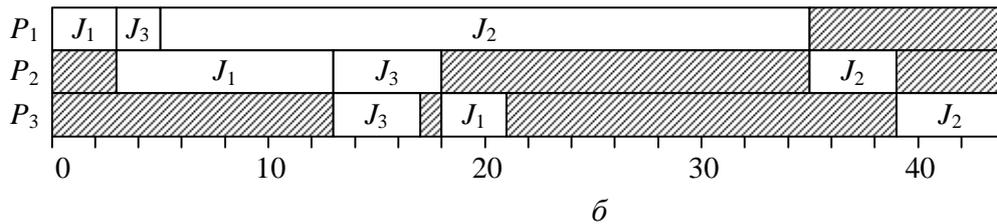


Рис. 3.10. Сравнение планов для сред с промежуточной памятью и без нее:  
*a* – время обслуживания задач; *б* – оптимальный FSIIS-план;  
*в, г* – оптимальные FSNIS-планы

С. С. Редди и Е. Фьюстел рассмотрели дополнительные аспекты планирования в конвейерной среде. В основном исследовался вопрос накладных расходов для создания оптимального плана на базе компьютерной системы с двумя машинами – центральным процессором и процессором ввода–вывода. Был получен следующий результат: поскольку накладные вычислительные расходы отличны от нуля, при некоторых обстоятельствах лучше оптимально планировать только подмножество общего множества задач, а оставшиеся задачи планировать случайным образом. Добавим, что большинство приведенных выше результатов были всесторонне изучены с более формальной, математической точки зрения Кофманом и П. Дж. Денингом.

Следует также отметить, что некоторые авторы обращаются к мультипроцессорным планам, когда рассматриваются конвейерные планы, потому что в них включается более одного процессора. Мы же для разграничения конвейерных и мультипроцессорных планов используем условие о том, что задача должна обслуживаться сначала одной машиной или процессором, а затем другой.

#### 3.5.4. Мультипроцессорные модели

Мультипроцессорными называются такие планы, в которых для оптимизации показателей производительности используется более одного процессора. Рассматривая их, следует обратиться к описанию двух типов сред планирования – обычным и особым. В первом случае исследуется ряд идентичных процессоров, набор задач с равными или неравными временами обслуживания и порядок предшествования (возможно пустой), а также схемы – с приоритетами и без. Во втором случае вводятся дополнительные требования и ограничения различного вида: определенное число ресурсов для каждого из класса, периодически выполняемые работы с указанием времени инициирования и завершения, промежуточные ограничения внутри планов.

**Обычные среды планирования.** Говоря об обычных средах планирования, необходимо остановиться на таких важных вопросах, связанных с оптимизируемыми критериями производительности, как минимизация максимального времени выполнения, количества процессоров и среднего времени потока.

*Минимизация времени выполнения процессов и количества процессоров.* Получение оптимальных решений посредством минимизации времени выполнения процессов и количества процессоров применимо, прежде всего, к планам с наличием приоритетного прерывания и без

его допущения. Сначала мы рассмотрим получаемое оптимальное решение для случая, когда любой граф со взаимно соизмеримыми весами узлов выполняется на двух процессорах (множество узлов считают взаимно соизмеримым, если существует такое  $w$ , что каждый вес узла является кратным целому числу  $w$ ), после чего обратимся к оптимальным решениям для корневых деревьев с взаимно соизмеримыми весами узлов и любым числом процессоров.

Планы с приоритетными прерываниями (Preemptive Schedules, PS) обычно противопоставляются неприоритетным (простым) планам (Basic Schedules, BS). В первом случае обслуживание некоторой задачи процессором может быть приостановлено, если это приведет к улучшению показателя производительности, во втором же задача обслуживается процессором до тех пор, пока не завершится ее выполнение.

Наиболее важный вклад в область изучения планов с приоритетными прерываниями внесли Р. Р. Мюнц и Кофман. В своих исследованиях они полагаются на результаты Р. Макнаутона, которые устанавливают нижнюю границу на оптимальном PS для множества из  $n$  независимых задач с весами (продолжительностью задачи или времени выполнения)  $\{w_1, w_2, \dots, w_N\}$  и  $k$  процессорами. Эта оптимальная длительность определяется выражением

$$\max \left\{ \max_{1 \leq i \leq n} \{w_i\}, \left( \sum_{i=1}^n w_i \right) / k \right\}.$$

Другими словами, оптимальная длина PS не может быть меньше, чем наибольшая по длительности задача или сумма времен выполнения, деленная на количество процессоров.

В оптимальном алгоритме множество узлов в графе с узлами удельного веса необходимо разбить в последовательность непересекающихся подмножеств так, чтобы все узлы в подмножестве были независимы. При этом все узлы из одного подмножества или на одном уровне являются кандидатами на одновременное выполнение. В  $N$ -уровневом графе заключительный узел занимает исключительно первый уровень; узлы, которые могут быть выполнены в течение единичного периода времени, предшествующего выполнению заключительного узла, – второй; и так далее вплоть до начального или входного узла в графе, который будет занимать  $N$ -й уровень.

Подобное назначение уровней соответствует методам разбиения по старшинству (предшествования), рассмотренным Ц. В. Рамамурти и М. Дж. Гонзалесом. В частности, описанная выше процедура назначения соответствует разбиению по последнему предшествованию, т. е. назначению узлов уровням с отнесением момента инициирования задачи к самому последнему возможному времени без увеличения минимального времени выполнения, с условием, что число процессоров не меньше максимального числа задач среди уровней. Для произвольного графа  $G$  отношение предшествования будет существовать между подмножеством благодаря предшествованию, которое существует между узлами в изначальном графе. PS-план для  $G$  может быть создан путем планирования сначала подмножества с наибольшим номером, затем следующего, нижележащего, уровня и т. д. Когда подмножество состоит только из одного узла, узел из нижележащего подмножества перемещается вверх, при условии, что это не нарушает ограничений предшествования. Если каждое из подмножеств спланировано оптимально, то в результате мы получим план подмножеств. Мюнц и Кофман показали, что для двух процессоров и равных узловых весов оптимальный план подмножеств для  $G$  является оптимальным PS-планом для  $G$ .

Этот результат мы обобщим для случая графов со взаимно соизмеримыми весами узлов. Чтобы получить оптимальный результат, необходимо преобразовать граф  $G$  в граф  $G_w$ , в котором все узлы имеют равные веса. Это достигается преобразованием веса узла  $w_i$  в последовательность  $n$  узлов  $w_i = nw$  (рис. 3.11). Следует отметить, что целостность изначального графа должна сохраняться неизменной – дуги, входящие или выходящие из узла в  $G$ , должны входить или выходить из узлов последовательности, представляющей изначальный узел. Показано, что оптимальный план подмножеств для  $G_w$  является оптимальным PS-планом для  $G$  с  $k = 2$ .

В этом подходе нужно определить, каким является число задач на уровне – четным или нечетным. Если число четное, то все задачи могут быть выполнены за минимальный промежуток времени без простоя на любом из двух процессоров. Если количество процессоров не кратно двум, то последние три планируемые задачи выполняются не менее чем за 1,5 единичных интервала времени, так как все задачи в  $G_w$  имеют длительность 1.

В качестве примера приведем план, когда три задачи одного уровня могут быть обслужены в минимальное время без простоя про-

цессора (рис. 3.12). Поскольку такое планирование не использует холостого хода процессора, последовательность подмножества может рассматриваться для создания PS-плана минимальной длины. Что касается оптимального PS-алгоритма (рис. 3.13), то оптимальной последовательностью подмножества для  $G$  является:  $\{T_1\}$ ,  $\{T_2, T_3\}$ ,  $\{T_5, T_6, T_7\}$ ,  $\{T_4, T_8\}$ ,  $\{T_9, T_{10}\}$ ,  $\{T_{11}\}$ .

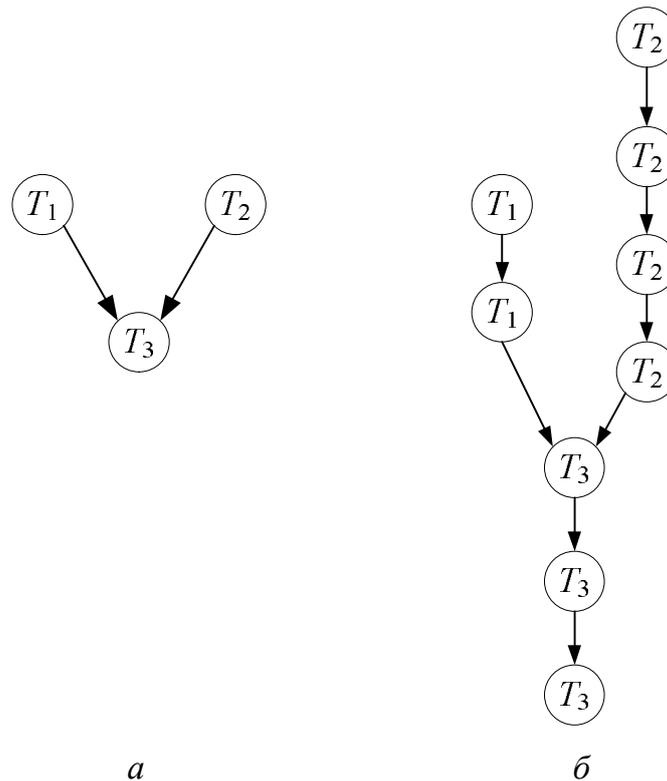


Рис. 3.11. Граф с обычными соизмеримыми весами узлов и соответствующий ему граф, имеющий узлы с равными весами:  
 $a$  – узлы графа  $G$ :  $w_1 = 7$ ,  $w_2 = 14$ ,  $w_3 = 10,5$ ;  $б$  –  $w = 3,5$

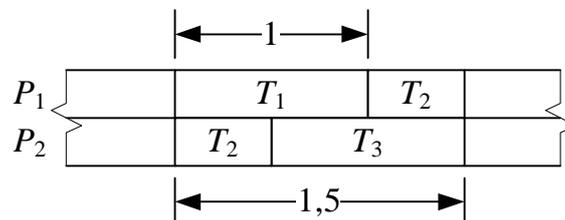


Рис. 3.12. Двухпроцессорный план с минимальным временем выполнения трех единичных задач

Полученные оптимальные решения были обобщены для случая, когда допускается любое количество процессоров и вычислительный граф является корневым деревом (т. е. деревом, в котором каждый

узел, за исключением входного и заключительного, имеет, по крайней мере, одного приемника) с взаимно соизмеримыми весами узлов  $w_i$ . Вместе с тем рассматриваются понятия общего плана (General Schedules, GS) и разделения процессорного времени.

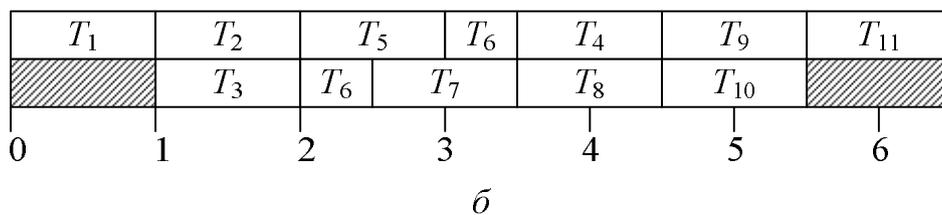
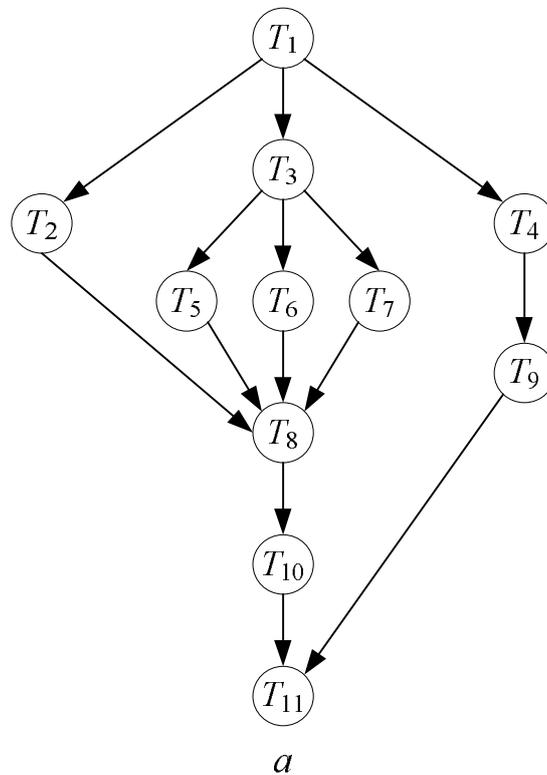


Рис. 3.13. Оптимальная последовательность подмножества:  
*a* – граф  $G$  для множества задач со всеми узлами, имеющими единичные веса;  
*б* – оптимальный план с приоритетными прерываниями.

Обычно  $k$  процессоров в системе воспринимаются как объединение  $k$  дискретных вычислительных устройств. Задача может обслуживаться процессором на приоритетной или неприоритетной основе и в течение времени обслуживания все ресурсы процессора подчиняются этой задаче. Допустим, что ресурсы процессора могут быть доступны задачам в дробных частях  $a$ , которые меняются от 0 до 1. Так, например, задача, требующая  $t$  единиц времени, при назначении

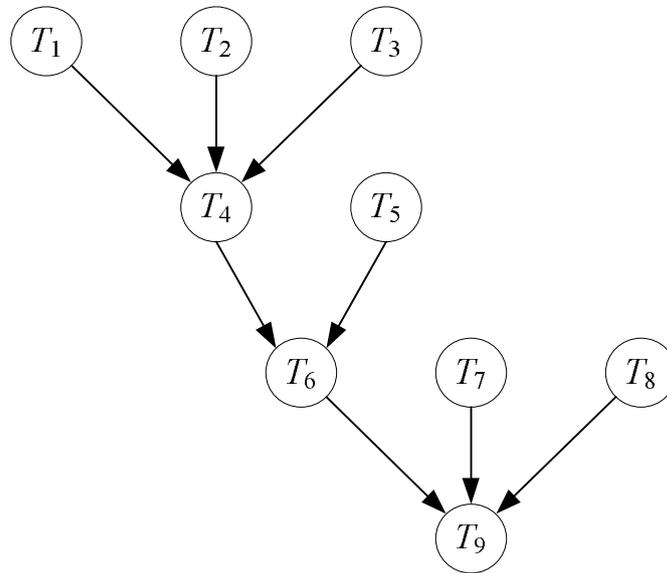
полному процессору может потребовать  $2t$  единиц при использовании половины процессорного времени ( $a = 0,5$ ). Если распределение процессорных ресурсов допускается изменять до завершения задачи, то можно говорить об общих планах, ставших возможными благодаря технике разделения процессорного времени.

Мюнц и Кофман впервые показали, что для заданного графа с заданным количеством процессоров при критерии качества работы в виде минимального времени выполнения планы, полученные с использованием GS-подхода, эквивалентны планам, сформированным PS-методом. Имеется в виду, что разделение процессорного времени не является необходимостью для формирования оптимального плана, если разрешено приоритетное прерывание.

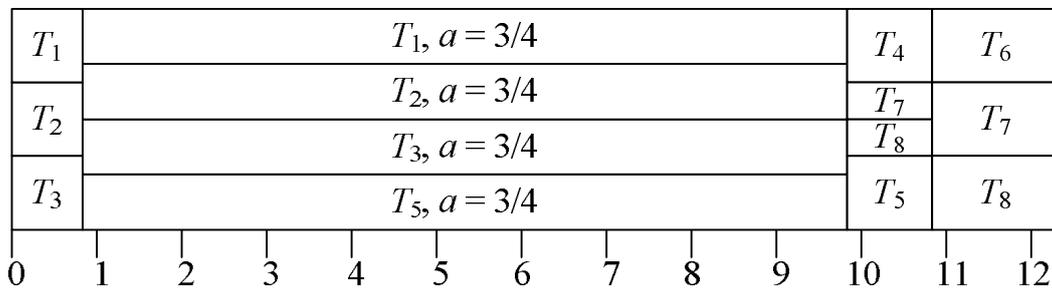
Эти же авторы используют определение уровней и формулируют алгоритм генерации оптимальных приоритетных планов для вычислений с древовидной структурой при произвольном числе процессоров  $k$  и взаимно соизмеримых весах узлов. Алгоритм начинается с назначения отдельных процессоров (т. е.  $a = 1$ ) каждой из  $k$  самых дальних от корня дерева задач. Две задачи  $T_i$  и  $T_j$  называются равноудаленными от заключительной задачи (находятся на одном уровне), если сумма весов задач от  $T_i$  до заключительной (включая  $T_i$ ) равна сумме весов задач от  $T_j$  до заключительной. Если в любой момент времени число задач  $n$ , требующих обслуживания, больше чем  $k$ , то каждой из задач на одном и том же уровне назначается дробная часть  $a$  ресурсов процессора такая, что  $a = k / n$ .

Задачи обслуживаются назначенными им процессорами с  $0 < a \leq 1$ , пока не закончится обслуживание задачи в дереве или пока не отменится текущее назначение процессора; некоторые задачи, находящиеся на равном расстоянии от заключительного узла, будут пропускаться, пока не будет произведено переназначение. Когда происходит одно из этих событий, процессоры переназначаются в соответствии с начальной процедурой. План, сформированный согласно этим правилам, называется М-планом (рис. 3.14). Доказано, что М-план, построенный таким образом, является оптимальным. Поскольку М-план является общим, т. е. GS-планом, а GS, в свою очередь, эквивалентен PS, алгоритм формирует оптимальный план с приоритетами. Приоритетный план (рис. 3.14, в) получен при условии, что все задачи, выполняемые внутри каждого из временных интервалов (рис. 3.14, б), независимы. Следовательно, внутри каждого интервала задачи могут быть спланированы оптимальным образом с использованием приоритетных методов. Это осуществляется назначением задачи процессору, пока

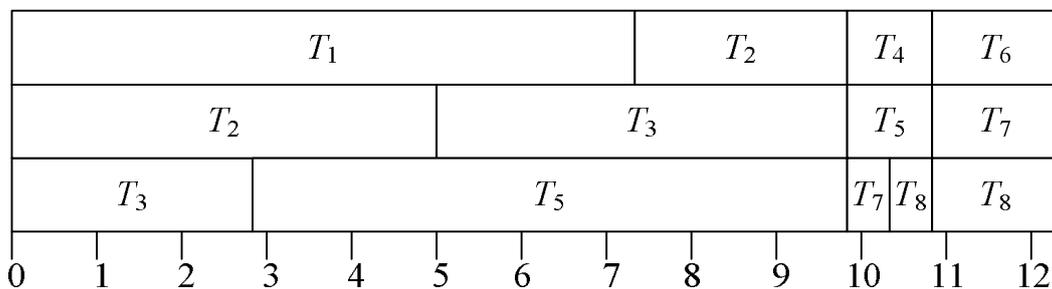
не завершится ее выполнение или не превысится интервал времени выполнения. В первом случае в точке завершения инициируется новая задача, во втором – задача переназначается следующему в последовательности процессору.



*a*



*b*



*в*

Рис. 3.14. Алгоритм Мюнца и Кофмана для  $k = 3$ :

*a* – корневое дерево с взаимно соизмеримыми весами отдельных узлов  $w$  ( $w_1 = 7,5, w_2 = 7,5, w_3 = 7,5, w_4 = 1, w_5 = 8, w_6 = 1,5, w_7 = 2, w_8 = 2, w_9 = 7,5$ );

*b* – *M*-планировщик; *в* – оптимальный преимущественный планировщик

Описанный выше алгоритм можно считать обобщенным алгоритмом «критического пути» [35], так как в нем на основании удаленности заданий от заключительных задач определена система приоритетов.

С. Лан и Р. Сети [40] применили уровневые алгоритмы, используемые Мюнцем и Кофманом, для исследования систем с неидентичными процессорами. Они показали, что такие алгоритмы формируют наикратчайшие приоритетные планы на двух процессорах для произвольных наборов задач. Когда количество процессоров возрастает до трех и более, уровневый алгоритм не позволяет получить оптимальный план, даже если структурой предшествования является дерево. Вместо этого он, в отличие от оптимального плана, используется для обеспечения выполнения условий о времени выполнения как для идентичных, так и для неидентичных процессоров. Было показано, что для  $m$  процессоров эти ограничения являются более жесткими, когда процессоры обладают разной производительностью ( $\sqrt{1,5m}$ ), чем когда процессоры являются подобными ( $2 - 2/m$ ).

В работе [40] формируются планы для независимых задач в гетерогенной системе, где относительно ресурсов процессоры рассматриваются как «стандартные». Говорят, что процессор имеет скорость  $b$ , если он в  $b$  раз быстрее стандартного процессора. Ц. Л. Лью и А. Т. Янг рассматривают мультипроцессорную систему, которая содержит  $n_1$  процессоров со скоростью  $b_1$ ,  $n_2$  процессоров со скоростью  $b_2$  и  $n_k$  процессоров со скоростью  $b_k$ . С помощью оптимального приоритетного алгоритма планирования получено выражение минимума времени выполнения для такой структуры.

В неприоритетных (простых) планах процессор, назначенный задаче, выделяется этой задаче, пока она не выполнится. Начальные результаты позволяют получать оптимальные неприоритетные двухпроцессорные планы для произвольного упорядочения задач, имеющих единичную продолжительность.

По мнению М. Фуджи, Т. Касами и К. Ниномия, ключом к решению этой проблемы является разбиение общего множества задач на пары совместимых и несовместимых задач. Говорят, что пара задач  $T_i$  и  $T_j$  является совместимой, если  $T_i \neq T_j$  и  $T_j \neq T_i$ . Пусть для множества из  $n$  задач  $m$  представляет максимальное число непересекающихся совместимых пар задач. Тогда  $n - m$  является нижней границей по времени, необходимому для выполнения всех задач. Данный подход приводит к обнаружению максимального числа совместимых пар задач и затем – к обнаружению оптимального упорядочения от задач этого множества к остальным задачам.

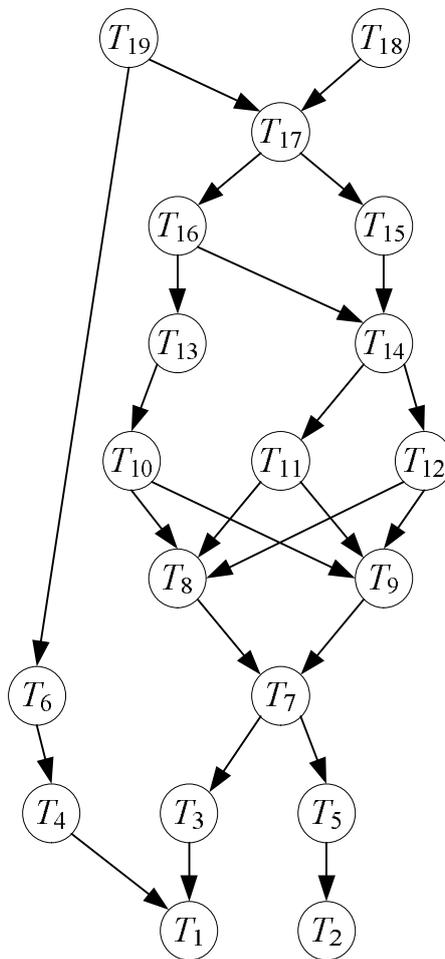
Кофман и Грахам предложили алгоритм для генерации списка задач и показали, что план, сгенерированный с использованием этого списка, не хуже любого плана, сформированного по любому другому списку. Списочный план (список, список задач)  $L$  для графа  $G$  из  $n$  задач, обозначенный как  $L = (T_1, T_2, \dots, T_n)$ , представляет собой некоторую перестановку  $n$  задач. Говорят, что задача готова, если все были выполнены все предшествующие ей задачи. При использовании списка для создания плана бездействующий процессор начинает обслуживать первую найденную в списке готовую задачу. Отсюда следует, что если список должен быть использован для создания оптимального плана, упорядочение в нем задач имеет первостепенное значение. Таким образом, ключевым моментом в подходе Кофмана и Грахама является поиск списка, из которого может быть произведен выбор оптимального плана.

Алгоритм, используемый для генерации оптимального списка, является рекурсивной процедурой, которая начинается назначением задаче или задачам, которые выполняются последними вследствие ограничений предшествования в графе задач, списка индексов в порядке возрастания. Заметим, что множество последователей этих задач является пустым. Назначение производится «вверх по графу», т. е. в качестве кандидатов на присваивание индексов рассматриваются задачи, последователям которых уже был назначен индекс.

Рассмотрение задач, таким образом, равнозначно разбиению их по признаку последнего предшествования, хотя они и не выполняются к моменту времени, который соответствует этому разбиению. Фактически изначально индексы могут присваиваться задачам в графе произвольно, а затем алгоритм Кофмана и Грахама переназначает индексы и вносит задачи в порядке уменьшения индексов, начиная с последнего назначенного. В итоге создается список. Оптимальный план формируется назначением готовых задач в этом списке свободным процессорам. Данный алгоритм можно представить в виде графа задач с переназначенными индексами, результирующего списка  $L^*$  и оптимального плана (рис. 3.15).

Используя контрпримеры, Кофман и Грахам показали, что их алгоритм не всегда дает оптимальный результат, особенно в том случае, когда количество процессоров возрастает до трех и более, или когда количество процессоров равно двум, но задачи могут иметь произвольную длительность. Это справедливо даже в случае, когда длительности задач имеют один или два единичных интервала. В свою очередь, Фуджи и ряд других исследователей показали, что в двух-процессорном случае задачи неединичной длины могут быть разделены

на серии задач единичной длины, и их алгоритм дает нижнюю границу времени обработки первоначальной задачи.



*a*

$T_{19}$	$T_{17}$	$T_{16}$	$T_{14}$	$T_{12}$	$T_{10}$	$T_9$	$T_7$	$T_5$	$T_2$	
$T_{18}$	$T_6$	$T_{15}$	$T_{13}$	$T_{11}$	$T_4$	$T_8$		$T_3$	$T_1$	
0	1	2	3	4	5	6	7	8	9	10

*b*

Рис. 3.15. Алгоритм Кофмана:  
*a* – граф задач с переназначенными нижними индексами  $L^* = (T_{19}, T_{18}, \dots, T_1)$ ; *b* – оптимальный план

Р. Сети и Ю. Мураока разработали оптимальный алгоритм для среды, представляющей собой систему задач, в которой для всех задач набора  $T$  максимальная длина пути от начального узла до  $T$  и уровень  $T$  являются постоянными. Затем этот алгоритм обобщается для случая общих систем задач.

Оптимальные результаты для неприоритетных планов также были предложены Хью [41]. Он рассматривает две проблемы, связанные с задачами единичной длины: определение минимума времени, необходимого для обработки графа задач при фиксированном числе процессоров и определение количества процессоров, необходимых для обработки графа за заданное время.

Первым шагом в решении этой задачи является разметка узлов произвольного графа. Узлу  $N_i$  присваивается метка  $a_i = X_i + 1$ , где  $X_i$  – длина наидлиннейшего пути от  $N_i$  до конечной вершины в графе. Разметка начинается с конечной вершины, которой присваивается метка  $a_1 = 1$ . Узлы, на единицу отдаленные от конечного, получают метку 2, и т. д. Смысл этой схемы меток в том, что минимальное время  $T_{\min}$ , необходимое для обработки графа, связано с  $a(\max)$  узлами с наивысшими индексами через соотношение  $T_{\min} \geq a(\max)$ .

Рассмотрим оптимальные решения задач, описанных ранее, для корневых деревьев. Используя процедуру разметки, изложенную выше, можно получить оптимальный план для  $m$  процессоров, организуя дерево задач единичной длины по следующему алгоритму.

1. Включаем в план первые  $m$  начальных узлов с наивысшими индексами меток (термин «начальный узел» применяется к узлу без предшественников). Если количество таких узлов больше  $m$ , то необходимо выбрать  $m$  узлов с наибольшими значениями  $a_i$ . В случае нескольких одинаковых величин выбор производится случайным образом.

2. Удаляем  $m$  обслуженных узлов из графа.

3. Повторяем шаги 1 и 2 для оставшегося графа.

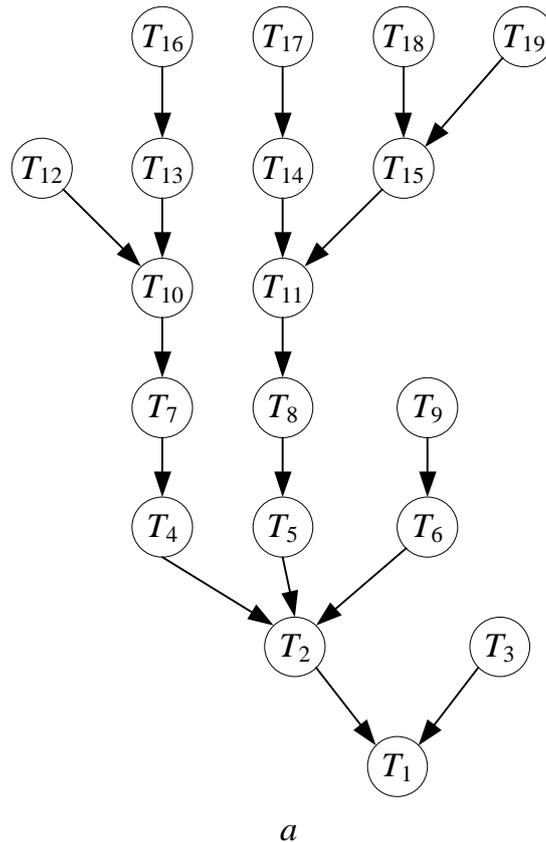
Полученные таким образом планы являются оптимальными при объявленных ограничениях, а процедуры разметки и планирования достаточно просты в исполнении.

Как было показано выше, минимальное время обработки графа, размеченного в соответствии с процедурой Хью, равно  $a(\max)$ . Предположим, требуется обработать граф за предписанное время  $t$ , где  $t = a(\max) + C$  и  $C$  – положительное целое число. Минимальное число  $m$  процессоров, необходимое для обработки графа за время  $t$ , вычисляется по формуле

$$m - 1 < [1 / (\gamma^* + C)] \sum_{j=1}^{\gamma^*} p(a_{\max} + 1 - j) < m,$$

где  $p(i)$  – количество узлов в графе с меткой  $a_i$ ;  $\gamma^*$  – величина константы  $\gamma$ , максимизирующая данное выражение.

Для объяснения этого результата рассмотрим оптимальный алгоритм Хью (рис. 3.16). Для  $C = 0$  величина  $\gamma^*$  имеет место, когда  $y = 1$  или  $y = 2$ , т. е. для обработки графа за минимальное время необходимы четыре процессора. Для  $C = 1$ ,  $t = 8$  и  $\gamma^*$ , имеющем место при  $y = 2$  или  $y = 5$ , требуются три процессора. Меняя  $C$ , определим, что необходимы три процессора, когда задачи должны быть выполнены за 9 единиц, и только два процессора – для максимального времени вычисления в 10 единиц.



*a*

$T_{19}$	$T_{16}$	$T_{15}$	$T_{11}$	$T_8$	$T_5$	$T_2$	$T_1$
$T_{18}$	$T_{14}$	$T_{13}$	$T_{10}$	$T_7$	$T_4$		
$T_{17}$	$T_{12}$	$T_9$	$T_6$	$T_3$			
0	1	2	3	4	5	6	7

*б*

Рис. 3.16. Оптимальный алгоритм Хью:  
*a* – корневое дерево, размеченное в соответствии с процедурой Хью;  
*б* – оптимальный план для трех процессоров

Ченом и Лью определено, что вышеописанная среда имеет дело с уровнем алгоритмом. Алгоритм, разработанный Кофманом и Гра-

хамом, также является уровневый. Определив множество задач, частичное упорядочение, временные ограничения для задач и неприоритетный алгоритм, Чен и Лью обнаружим, что уровень задачи, которая не имеет последователей, равен единице, а задачи, имеющей одного или более последователей, – единице плюс максимальное значение уровня из уровней последующих элементов задачи.

Уровневый алгоритм, в котором планируемые внутри одного уровня задачи полностью произвольны, называется простым (Simple Level Algorithm, SLA). Пусть  $\omega_{SLA}$  – общее время выполнения плана, полученного с помощью SLA, а  $\omega_o$  – общее время выполнения оптимального плана. Тогда можно показать, что для двухпроцессорной системы имеет место следующее выражение:

$$\omega_{SLA} / \omega_o \leq 4 / 3.$$

Для трехпроцессорной системы это отношение имеет вид

$$\omega_{SLA} / \omega_o \leq 3 / 2.$$

В отличие от предыдущих, большинство результатов, приведенных ниже, описаны в терминах ограничений, т. е. план, полученный с помощью эвристического или приближенного метода, выражается в виде коэффициента, сравнивающего почти оптимальный план с оптимальным. Эта величина больше или равна единице и показывает, насколько производительность упрощенного подхода сравнима с оптимальным решением.

Здесь следует сделать два замечания. Во-первых, часто случается, что эвристические планы выдают решение, которое не хуже оптимального. В результате появляется желание приписать такому подходу незаслуженное качество, до того как определены его границы производительности в наихудшем случае. Во-вторых, при сравнении эвристического плана с оптимальным необходимо помнить, что последний не обязательно является детерминированным, так как, за исключением уже рассмотренных специальных случаев, неперечисляемые решения для определения оптимального результата недоступны. Другими словами, оптимальное решение не может быть найдено простым перебором всех возможных решений и затем выбором лучшего. Для задач с малым количеством альтернатив перебор может быть не очень сложен. Однако когда количество альтернатив достаточно велико и для их обработки требуется экспоненциально растущее время, эвристические подходы выглядят более привлекательно по сравнению с переборными.

Приведенные далее результаты обсуждаются в терминах списочных планов. Списочный планировщик выполняет задачу, только в том случае, когда все ее предшественники закончили свое выполнение, и нет предшествующих задач в приоритетном списке, готовых к выполнению.

Наиболее значительный вклад в создание граничных оценок для почти оптимальных мультипроцессорных планов был сделан Грахамом. Его модели имеют дело с так называемыми мультипроцессорными аномалиями. Эти аномалии появляются при наблюдении и существовании одного из следующих условий, которые могут привести к увеличению времени выполнения:

1) замещении списка задач  $L$  другим списком  $L'$  с неизменными набором временных параметров задач  $m$ , порядком предшествования  $<$  и количеством сопроцессоров  $n$ ;

2) ослаблении некоторых ограничений частичного упорядочивания;

3) уменьшении некоторых времен выполнения;

4) увеличении количества процессоров.

Грахам разработал общую граничную оценку с двойным исполнением множества задач. Во время первого выполнения задачи характеризуются параметрами  $m$ ,  $<$ ,  $L$ ,  $n$  и  $w$  (длина плана), во время второго:  $m'$ ,  $<'$ ,  $L'$ ,  $n'$  и  $w'$ , так что  $m' \leq m$ , а каждое ограничение  $<'$  содержится в  $<$ . Результатом данной оценки является выражение

$$\frac{w'}{w} \leq 1 + [(n-1)/n'].$$

Грахам показал, что эта оценка является лучшей из возможных, и для  $n = n'$  отношение  $2 - 1/n$  может быть достигнуто изменением любой из величин ( $L$ ,  $m$  или  $<$ ).

Аномалия, возникающая после уменьшения времени выполнения задач, названа Г. К. Маначером «аномалией Ричардса», так как аномалии этого типа впервые были рассмотрены П. Ричардсом. Результаты моделирования, полученные Маначером, показали, что приблизительно 80 % всех тестов проявили аномалию Ричардса.

Также Маначер разработал алгоритм обеспечения «устойчивости в сильном смысле» такой, что время выполнения всех задач в списке не увеличивается с уменьшением времени выполнения любой из них. Это достигается добавлением «умеренного числа» ограничений предшествования к первоначальным ограничениям частич-

ного порядка. Манагер также рассмотрел проблему стабильности для случая, когда в многозадачной среде задания имеют различные времена пуска.

Ранее было сделано предположение о неизменности (статичности) списка задач до момента завершения их выполнения. Разновидностью таких списков являются динамически формируемые списки [41], переопределяемые каждый раз, когда освобождается процессор. Когда это случается, задача, возглавляющая «наидлиннейшую цепь невыполненных задач» (максимальная сумма времен задач в цепи), выполняется первой.

Пусть  $\omega_L$  – время завершения для множества задач, выполняемых таким образом, а  $\omega_0$  – минимальное время завершения. Используя описанную выше оценку, находим  $L/\omega_0 \leq 2 - \frac{1}{n}$ , так как динамически формируемый список определяет замену  $L$  на  $L'$ . Грахам разработал несколько лучшую оценку, которая определяется формулой

$$\omega_L/\omega_0 \leq 2 - [2/(n+1)].$$

Альтернативой этому подходу является назначение незанятому процессору задачи, чье время выполнения плюс время выполнения всех следующих за ней элементов является максимальным. Если множество задач, выполняемых таким образом, имеет время завершения  $\omega_m$ , то  $\omega_m/\omega_0$  также ограничено предшествующим ограничением.

Особый случай подхода наидлиннейшей динамической цепочки – ситуация, когда  $<$  является пустым, т. е. задачи независимы. Для этого случая Грахам определил наилучшую оценку в виде

$$\omega_L/\omega_0 \leq 4/3 - 1/3n.$$

Как говорилось ранее, первоочередной причиной для определения этих ограничений является обеспечение хороших субоптимальных планов при лишь частичных вычислительных затратах, необходимых для получения оптимального решения. Предположим, что во время изучения множества из  $r$  планируемых задач определено, что размер множества слишком велик для использования методов перебора. Тогда наиболее логично использовать следующую альтернативу (для случая  $< = \emptyset$ ): оптимальным образом спланировать  $k$  наидлиннейших задач ( $k \geq 0$ ), а оставшиеся  $r - k$  задач –

произвольным образом. Ограничение, разработанное Грахамом для этого подхода, дает следующее неравенство:

$$\frac{\omega(k)}{\omega_0} \leq 1 + \frac{1 - 1/n}{1 + [k/n]},$$

где  $n$  – число используемых процессоров.

Для этого результата имеют место два особых случая:

– при  $k = 0$

$$\omega(0)/\omega_0 \leq 2 - 1/n,$$

это ограничение, разработанное для общей начальной границы при  $n = n'$ .

– при  $k = 2n$ :

$$\omega(2n)/\omega_0 \leq 4/3 - 1/3n.$$

Таким образом, предыдущие два результата являются частными случаями для результатов более общих.

Позже Грахам рассмотрел обратную задачу, когда дан фиксированный предел минимального числа процессоров. Напомним, что Хью рассматривал этот вопрос как частный случай построения корневого дерева. Если принять, что  $<$  не определяет никакого отношения (т. е. задачи независимы), то данная задача сводится к одномерной задаче с ограничением запасов или частному случаю задачи балансировки сборочной линии и может быть рассмотрена следующим образом.

Предположим, что несколько объектов должны быть помещены в несколько одинаковых коробок и имеют одинаковые длину и ширину (но не высоту), причем оба эти размера точно совпадают с соответствующими размерами коробок. Тогда задача принимает форму задачи минимизации количества коробок для хранения объектов. Если объекты сопоставить задачам, а коробки – процессорам, то эвристики, разработанные Грахамом, могут быть использованы для определения верхних границ минимального количества процессоров. Таким образом, видно, что в частном случае без предшествования для достижения субоптимальных результатов со значительным уменьшением затрат на вычисления могут быть использованы решения из других областей.

Алгоритмы наидлиннейшего пути также были исследованы Кауфманом для графов с древовидной структурой. Считается, что в вычислительной среде ИУС допускаются неравные длительности задач,

но не разрешены приоритетные прерывания. В методе, схожем с методом Мюнца и Кофмана, задачи с весами больше единицы представляются в виде строки задач с единичным весом, чья сумма равна весу изначальной задачи. Такое представление графа позволяет определять оптимальное неприоритетное решение, используя алгоритм Хью, так как граф является деревом, хотя алгоритм наидлиннейшего пути Кауфмана или  $G$ -алгоритм не позволяет прерывать процессор до завершения задачи, если она является членом строки задач, представляющим задачу неединичного веса.

Пусть  $W_p$  – оптимальный приоритетный план,  $W_n$  – оптимальный неприоритетный план, а  $W_g$  – план, полученный с помощью  $G$ -алгоритма. Тогда ограничения Кауфмана могут быть определены в виде

$$W_p \leq W_n \leq W_g \leq W_p + k - [k/n],$$

где  $k$  – вес самой большой задачи в первоначальном графе;  $n$  – количество процессоров, доступных любому из алгоритмов.

Используя моделирование, Т. Л. Адам, К. М. Чанди и Дж Р. Диксон сравнили производительность нескольких списков планов, сформированных в среде без ограничений. Среда Адама рассматривается для общих графовых структур с двумя или более процессорами, неравными длительностями задач и без назначения приоритетов задачам. Эвристики могут быть следующими:

1) HLFET (Highest Levels First with Estimated Times, приоритет наивысших уровней с оценкой продолжительности) – термин «уровень», используемый здесь, относится к сумме весов всех вершин в наидлиннейшем пути от задачи до заключительного узла; так как не предполагается независимость задач, предшествующие задачи должны быть завершены до того, как иницируется новая;

2) HLFNET (Highest Levels First with No Estimated Times, сначала приводятся наивысшие уровни без оценки продолжительности) – фактически предполагается, что все задачи имеют равные времена;

3) RANDOM (произвольный) – приоритеты задачам назначаются произвольно;

4) SCFET (Smallest Co-levels First with Estimated Times, сначала приводятся наименьшие соуровни с оценкой продолжительности) – соуровень задачи определяется так же, как и уровень, за исключением того, что длина пути вычисляется от входного узла, а не от заключительного; приоритет назначается в соответствии с соуровнями (т. е. чем меньше соуровень, тем выше приоритет);

5) SCFNET (Smallest Colevels First with No Estimated Times, сначала приводятся наименьшие соуровни без оценки продолжительности) – определяется как SCFET, но все задачи имеют равную длительность, что равнозначно самому раннему разбиению по старшинству, если не принимается во внимание время выполнения.

Моделирование, основанное на реальных и случайно созданных графах, показывает, что порядок соответствия среди графов следующий: HLPET, HLFNET, SCFNET, RANDOM и SCFET. Почти оптимальная производительность HLPET также подтверждает полезность планов с наидлиннейшим путем, когда показателем производительности выбирается минимальное время выполнения.

Уровень производительности, полученный Адамом (в пределах 4,4 % от оптимального) для планирования с наидлиннейшим путем сравним с уровнем производительности, полученным Маначером, и равен 15 %. Почти оптимальное планирование наидлиннейшего пути или планирование с критическим путем также было одобрено В. Х. Кохлером, который продемонстрировал, что производительность этих эвристик возрастает с увеличением числа процессоров.

Напомним, что ранее целью являлась минимизация среднего времени потока для одного процессора. Для  $m$  процессоров ( $m \geq 2$ ) и множества независимых задач, если задачи планируются в соответствии с методом наикратчайшего времени обработки, результирующий план гарантированно даст минимум среднего времени завершения (табл. 3.7).

Таблица 3.7

**Время обработки задач**

$T$	$P_t$
1	1
2	2
3	4
4	4
5	5
6	8

Тем не менее SPT не обязательно минимизирует максимум времени завершения. Задача минимизации максимума времени завершения ряда независимых задач на двух процессорах (а следовательно, и многопроцессорный вариант) как известно, является *NP*-трудной, т. е. не может

быть решена никаким методом, кроме переборного. Доказательство того, что два SPT-плана с одинаковым средним временем окончания не дают одинакового времени окончания, показано на рис. 3.17, где SPT\* обозначает минимум времени завершения SPT-плана для тех же самых трех процессоров и множества задач.

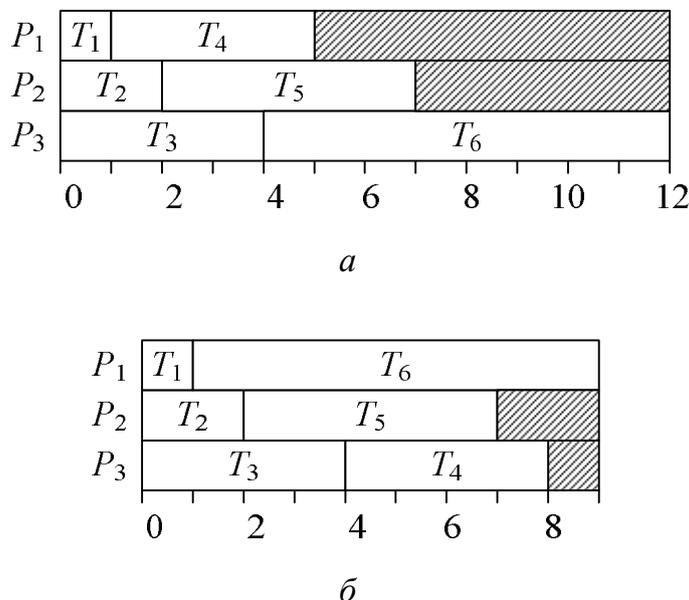


Рис. 3.17. Сравнение SPT- и SPT\*-планов для трех процессоров и задач, определенных в табл. 3.7:  
*a* – SPT-план; *б* – SPT\*-план

Дж. Бруно, Кофман и Сети сравнили параметр времени завершения SPT-плана с оптимальным планом и получили следующую границу:

$$W_{\text{SPT}^*} / W_{\text{ОПТ}} \leq 1 - 1/m .$$

Кофман и Сети показали, что наидлиннейший SPT-план почти на 50 % длиннее, чем SPT\*. Если список неубывающих задач назначен в обслуживание  $m$  процессорам, то эта цифра может быть уменьшена до 33 % при условии, что наидлиннейшие  $m$  задач назначаются первыми на  $m$  процессорах. Назначение всех множеств из  $m$  задач как первых наидлиннейших изменяет границу почти на 25 % в худшую сторону.

Формируя планы с наидлиннейшим временем обработки, имеющие тенденцию к максимизации среднего и минимизации максимального времени завершения, и затем подготавливая задачи, назначенные на процессор SPT-способом, так называемый RPT-способ формирует

планы с хорошим максимальным временем завершения и почти оптимальными свойствами среднего времени завершения (рис. 3.18).

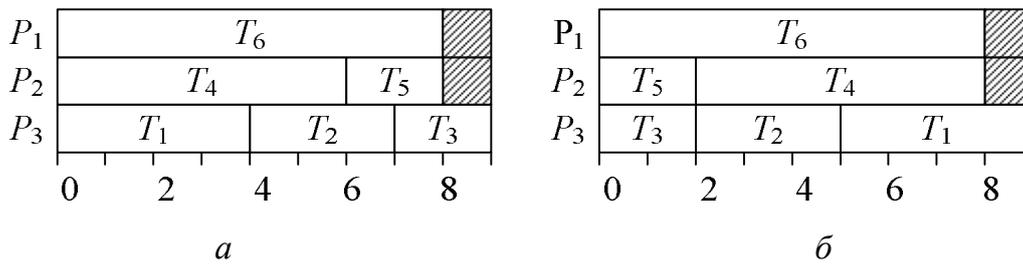


Рис. 3.18. Сравнение LPT- и RPT-планов для трех процессоров и задач, определенных в табл. 3.7:  
*a* – LPT-план; *б* – RPT-план

Рамамурти, Чанди и Гонзалез используют концепцию разбиения по старшинству для создания ограничений времени обработки и количества процессоров для структур графов, чьи узлы требуют единичного времени выполнения. Как показано выше, группы задач разбиваются на подмножества в порядке предшествования, указывая на самые ранние и самые поздние моменты времени, в течение которых задачи могут быть запущены, а также гарантируют минимум времени выполнения для графа. Это время определяется числом разбиений и является мерой наидлиннейшего пути в графе. Для графа из  $l$  уровней минимальное время выполнения равно  $l$  единицам. Чтобы выполнить граф за это минимальное время, абсолютный минимум необходимого числа процессоров вычисляется по формуле

$$\max\{|L_j \cap E_j|, 1 \leq j \leq L,$$

где  $L_i$  и  $E_i$  – самое последнее и самое раннее по старшинству  $j$ -е разбиения соответственно;  $|x|$  – мощность множества  $x$ .

Рамамурти и другие исследователи определяют задачи, содержащиеся в  $L_j \cap E_j$ , как существенные задачи (Essential Tasks). Задачи, содержащиеся в  $j$ -м подмножестве, определяемом через  $L_j \cap E_j$ , должны быть инициированы через  $j - 1$  временных единиц после старта начальной задачи графа, что гарантирует минимум времени выполнения. Используя метод, схожий с методом Хью, авторы получили нижнюю границу для минимального количества процессоров при времени выполнения, превышающем  $L$ , и для минимального времени выполнения при фиксированном количестве процессоров. Причем не обязательна графовая структура вида ветвящегося дерева.  $L$ -разбиение также используется авторами для получения нижней

и верхней границ минимального количества процессоров, необходимых для обработки графа за наименьшее время.

Кроме того, были разработаны алгоритмы определения необходимого количества процессоров для обработки графа за наименьшее возможное время, а также минимального времени обработки графа задач  $k$  процессорами. Второй из этих алгоритмов модифицирован для планирования графов с неравными длительностями задач. Сложностью в этом случае является то, что часто желательно сохранять процессор незанятым, даже когда имеются задания, требующие обслуживания. Так, например, оптимальный план (рис. 3.19, *а*) для графа (рис. 3.19, *б*) может требовать 15 единиц. Однако если процессору 2 назначается задача 6 при завершении выполнения задачи 3, необходимое время равно 17 единицам (рис. 3.19, *в*).

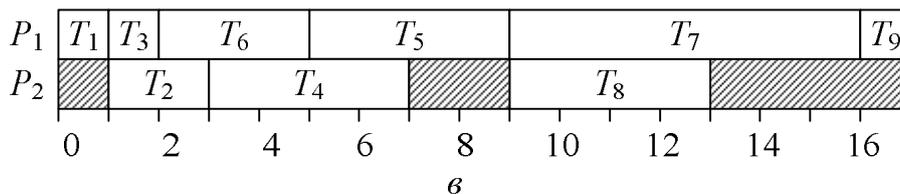
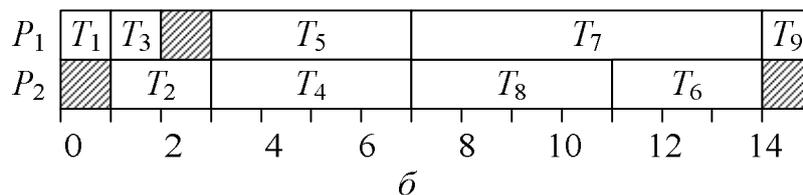
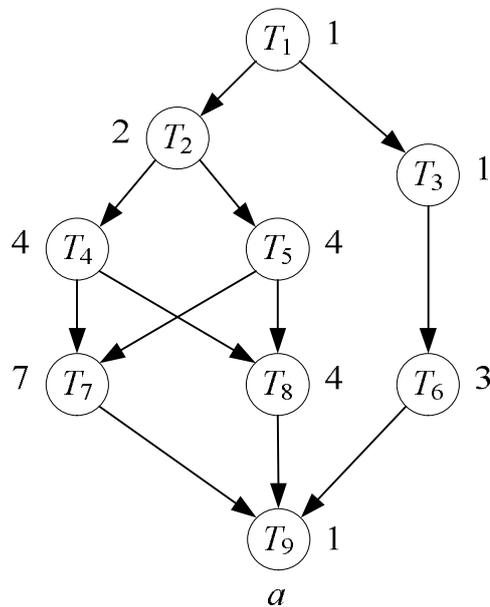


Рис. 3.19. Эффект намеренного включения простоя процессора:  
*а* – граф задач для множества задач; *б* – оптимальный план;  
*в* – план, когда процессоры активируются, как только это возможно

Время вычисления, необходимое для получения оптимального решения при помощи этих алгоритмов, достаточно велико. Оно было значительно уменьшено посредством двух эвристических подходов, которые дают оптимальный результат в большинстве случаев. В подходе  $A$  желательно не включать холостой режим работы процессора, а задачи выбирать в соответствии с их позицией в  $L$  разбиениях. В подходе  $B$  первыми выбираются существенные задачи. Из этих двух подходов эвристика  $A$  была быстрее, и во всех тестируемых случаях эвристический подход выдавал оптимальное решение.

Ограничения, рассмотренные ранее, были улучшены Е. Фернандезом и Б. Бусселом с помощью подхода критического пути. Для некоторого графа существует путь от входного узла до выходного, называемый критическим путем, который определяет минимальное время выполнения для графа. Этот подход не требует равенства времен задач. При заданном значении времени выполнения критического пути  $T_{\text{ср}}$  существует «интервал завершения» (определяемый самым ранним и самым поздним временем начала выполнения) для каждой задачи в графе, в течение которого эта задача должна быть завершена, и время ее выполнения не должно превысить  $T_{\text{ср}}$ .

В приближении к нижней границе количества процессоров Фернандез и Буссел рассматривали целочисленные интервалы между 0 и  $T_{\text{ср}}$ , внутри каждого из которых задачи перемещены так, чтобы дать минимальное перекрытие. Среднее число процессоров, требуемых в интервале, определяет минимальное число процессоров, необходимых для этого интервала. Если все интервалы исследованы, то максимальное среднее значение, округляемое в большую сторону к ближайшему целому, представляет минимальное количество процессоров, необходимых для обработки графа за минимальное время.

Похожие идеи использованы для определения минимального времени выполнения с фиксированным числом процессоров. В течение каждого интервала должно иметь место некоторое количество вычислений, чтобы убедиться, что общее время выполнения не превышает  $T_{\text{ср}}$ . Если количество используемых процессоров меньше некоторого минимума, то каждый интервал будет вносить количество времени сверх того, что могло бы быть внесено, если ограничение по  $T_{\text{ср}}$  выполнено. Максимальная нехватка времени, вносимая всеми интервалами, определяет минимальное количество времени свыше  $T_{\text{ср}}$ , необходимого для обработки графа.

Позднее Буссел, Фернандез и О. Леви обращались к проблеме разработки алгоритмов для минимизации числа процессоров, необходимых для выполнения плана за определенное время, и для минимизации времени выполнения с фиксированным числом процессоров. Полученные в результате планы были названы процессорно-оптимальными и оптимальными по времени соответственно. Рассмотренная среда состояла из множества задач частичного упорядочивания, неравных временам беспriorитетных прерываний активных задач. Иначе говоря, их алгоритм состоит в добавлении условий старшинства (добавочных дуг) в те узлы первоначального графа, где число задач-кандидатов превышает число процессоров. Результатом является распределение требований процессора по всей длине графа без неблагоприятного воздействия на времена выполнения.

Как и для алгоритмов Рамамурти, Чанди и Гонзалеза, описанных выше, фактическое описание и реализация данного алгоритма и его составных частей оказываются значительно более трудными, чем основная идея (предпосылка), лежащая в его основе. В обоих случаях это проявляется в виде значительного увеличения времени вычислений даже на больших компьютерных системах. Альтернативный метод – полный перебор – все равно является гораздо менее желательным. Вместо этого Рамамурти предложил субоптимальные эвристические методы, а Буссел – альтернативные субоптимальные методы, которые могут выполняться в интерактивном режиме. В обоих случаях упоминается о трудности получения оптимальных планов в общей среде планирования. В свою очередь, Лью и Янг разработали ограничения минимального времени завершения для произвольного множества задач, когда не все задачи независимы, а процессоры не обязательно являются идентичными.

Методы минимизации времени выполнения для некоторых особых случаев были рассмотрены Кофманом и Грахамом в работе об оптимальных непriorитетных двухпроцессорных планах. Точкой опоры в них является возможность преобразования графа  $G$  с произвольными взаимно соизмеримыми весами задач в граф  $G_w$  с временами выполнения, равными  $w$ . Как и раньше,  $w$  – самый большой вес, кратный всем весам задач в  $G$ . Если приоритетные прерывания допускаются только в моменты времени, кратные  $w$ , то оптимальный непriorитетный план для  $G$  может рассматриваться как оптимальный приоритетный. В качестве примера можно привести случай увеличения производительности разрешением приоритетных прерываний в конце каждого единичного интервала, т. е.  $w = 1$  (рис. 3.20). Вклю-

чение приоритетных прерываний в конец каждого единичного интервала приводит к увеличению производительности на 117 %.

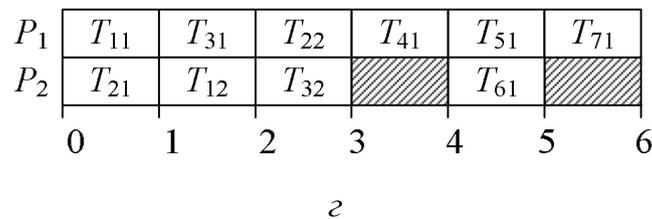
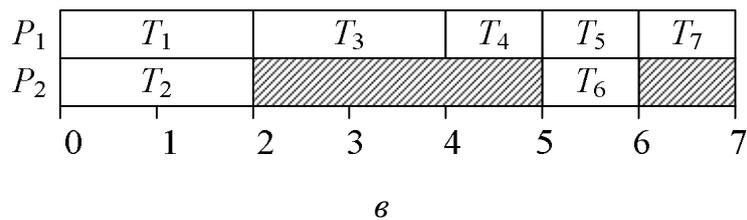
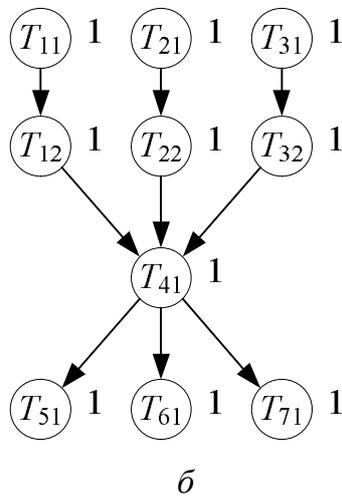
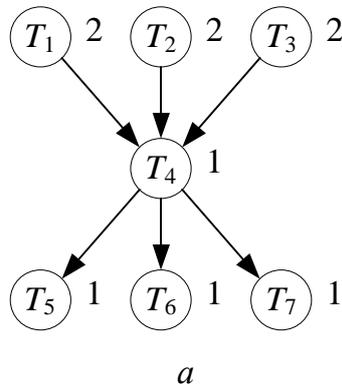


Рис. 3.20. Сравнение оптимальных приоритетных и неприоритетных планов:  
*a* – граф задач  $G$ ; *б* – граф задач  $G_w$  ( $w = 1$ );  
*в* – оптимальный неприоритетный план для графа  $G$ ;  
*з* – оптимальный план для графа  $G_w$

Кофман и Грахам заметили, что разрешение прерываний чаще, чем каждые  $w / 2$  единиц, не дает никакой выгоды, и показали, что для произвольного числа процессоров  $m$  ( $m \geq 1$ ) длина  $W_n$  неприоритетного плана относится к длине приоритетного плана следующим образом:

$$W_n / W_p \leq 2 - 1 / m.$$

*Минимизация среднего времени потока.* Рассмотрим задачу построения планов для множества независимых задач с целью минимизации среднего времени потока. Ранее было продемонстрировано, что показатель производительности может быть минимизирован для набора идентичных процессоров планированием задач в соответствии с SPT-подходом. Однако мы обратимся к неидентичным, или гетерогенным, процессорам. Приведенные результаты говорят об усложнении процесса планирования. Эти соображения отражают возрастающую необходимость многопроцессорных и распределенных процессорных систем и полезность увеличения пропускной способности систем либо замены неисправных или устаревших компонентов.

Бруно, Кофман и Сети приводят эффективный алгоритм планирования независимых задач, уменьшающий среднее время выполнения (среднее время потока). Поскольку речь идет о неидентичных процессорах, использование отдельной величины для представления времени выполнения задач больше не имеет смысла. Будем рассматривать время выполнения задачи на каждом из процессоров. Удобным способом сделать это для случая  $m$  процессоров и  $n$  задач является использование матрицы  $[T_{ij}]$  размером  $m \times n$ , в которой целые неотрицательные величины  $T_{ij}$  обозначают время выполнения задачи  $T_j$  на процессоре  $P_i$ . Такой матрицы для пяти задач и трех процессоров имеет вид

$$[T_{ij}] = \begin{bmatrix} 2 & 3 & 1 & 4 & 6 \\ 1 & 4 & 1 & 5 & 5 \\ 3 & 2 & 3 & 2 & 3 \end{bmatrix},$$

для которого можно составить соответствующий оптимальный план (рис. 3.21).

Из матрицы  $[T_{ij}]$  формируется матрица  $Q$  размером  $nm \times n$ , где множество  $n$  элементов является допустимым, если в любой его строке нет двух одинаковых элементов. Ценой этого множества называется сумма весов  $n$  элементов. Тогда целью станет поиск допустимого множества с наименьшей возможной ценой. Можно сформулиро-

вать данную задачу в виде транспортной задачи и прийти к непрерывному оптимальному решению:

$$Q = \begin{bmatrix} [T_{ij}] \\ 2[T_{ij}] \\ \vdots \\ n[T_{ij}] \end{bmatrix}.$$

Если каждой задаче назначены приоритет или величина срочности  $w$  и принято, что процессоры идентичны, то эта задача является  $NP$ -трудной. Однако для  $m$  процессоров и  $n$  задач показано, что нижняя граница взвешенного среднего времени потока  $F'_w(m)$  может быть определена в виде

$$F'_w(m) \geq [(m+n)/m(n+1)]F'_w(1),$$

где  $F'_w(1)$  – взвешенное среднее время потока с одним процессором.

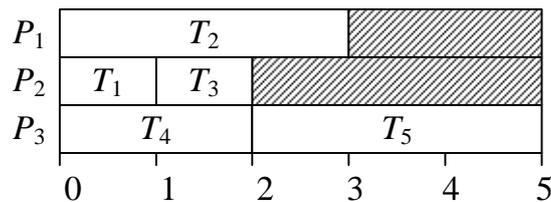


Рис. 3.21. Оптимальный план, использующий алгоритм Бруно, Кофмана и Сети

Оптимальное решение было использовано для создания простых эвристических методов, обеспечивающих почти оптимальные планы со средним временем завершения. Рассматривается  $n$  независимых задач,  $m$  неидентичных процессоров и матрица  $P$  для описания времени обработки задачи  $i$  на процессоре  $j$  –  $P_{ij}$ . Кларк, проводивший это исследование, в первую очередь, сделал вывод, что каждый элемент  $P_{ij}$  может быть выражен как «произведение времени, связанного с работой  $i$ , на показатель эффективности, связанный с процессором  $j$ , т. е.  $P_{ij} = p_i w_j$ ». В разработанном алгоритме формируется матрица времен обработки и процессорных коэффициентов (рис. 3.22, а). Причем времена обработки расположены в следующем порядке:

$$p_1 \geq p_2 \geq \dots \geq p_5.$$

Начиная с верхней строки матрицы, наименьшие коэффициенты обозначаются кружком. Это говорит о том, что задача, находящаяся

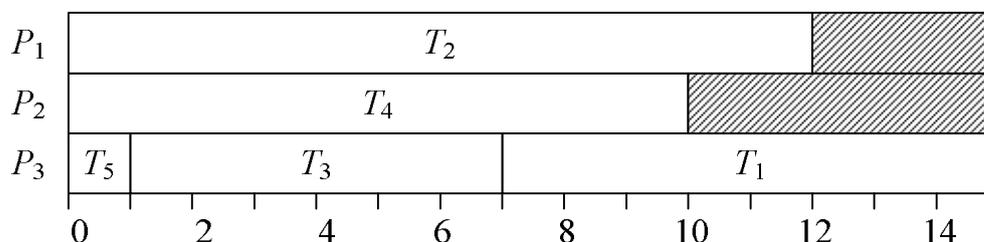
в первой строке матрицы (задача 1 в рассматриваемом примере), назначается процессору, который соответствует отмеченному коэффициенту (процессор 3). Вторая строка матрицы формируется копированием неотмеченных кружком коэффициентов из предыдущей строки и увеличением значения отмеченных коэффициентов на соответствующее значение  $w_j$ . Повторив эту процедуру  $n - 1$  раз (с произвольным выбором в случае совпадений), получаем план с оптимальным средним временем завершения (рис. 3.22, б).

$I$	$P_i$	potential coefficients			machine chosen
		1	2	3	
1	8	2	2.5	①	3
2	6	②	2.5	2	1
3	6	4	2.5	②	3
4	4	4	②.5	3	2
5	1	4	5	③	3

*a*

$i$	1	2	3	$h_1$	$h_2$	$h_3$
1	16	20	⑧	1	1	1
2	⑫	15	6	1	1	2
3	12	15	⑥	2	1	2
4	8	⑩	4	2	1	3
5	2	2.5	①	2	2	3

*б*



*в*

Рис. 3.22. Алгоритм Кларка:

*a* – матрица времени обработки, эффективности работы и выбора машин;  
*б* – альтернативный метод выбора машин; *в* – оптимальный план

В другом варианте этого подхода вся матрица первоначально заполняется явными временами обработки, а матрица коэффициентов формируется способом, похожим на описанный выше. Пусть коэффициенты  $h_1, h_2, \dots, h_m$  в некоторой строке  $i$  представляют возможные позиции соответствующих задач в последовательности из трех процессоров, считая от конца плана. Тогда для минимального  $h_i$  выбирается  $P_{ij}$  (отмечается кружком), а  $h_i$  в следующей строке увеличивается на единицу (рис. 3.22, в). Величина  $F$  для этого примера определяется следующим образом:

$$F = w_1 p_2 + w_2 p_4 + 3(w_3 p_5) + 2(w_3 p_3) + w_3 p_1 = \\ = 2 \cdot 6 + 2,5 \cdot 4 + 3 \cdot 1 \cdot 1 + 2 \cdot 1 \cdot 6 + 1 \cdot 1 \cdot 8 = 45.$$

Поскольку характеристики процессора не используются, последний подход является более общим, но это не гарантирует его оптимальности. Из-за простоты данного подхода Кларк использовал его в одной из своих работ как «быстрый и приближенный» (Quick And Dirty, QAD) алгоритм. Модификация этого алгоритма QAD\* сортирует задачи на каждой машине с помощью SPT-упорядочивания, поскольку это гарантирует минимизацию времени потока на каждой машине. Кларком доказано, что существует способ перенумерации задач (перестановка строк в массиве времени обработки) такой, что QAD дает план с минимальным временем потока и, соответственно, минимальным средним временем потока.

Лью и Янг разработали алгоритм минимизации среднего времени потока на множестве независимых задач для частного случая с одним процессором со скоростью  $b$  или  $n$  стандартными процессорами. Например, этот алгоритм позволяет сравнить производительность, учитывая соотношение среднего времени потока на гомогенной системе из  $n + 1$  стандартных процессоров, с негомогенной системой, содержащей  $n$  стандартных процессоров и один процессор, который в 6 раз мощнее стандартного процессора.

**Особые среды планирования (периодичные задачи при формировании процессов).** Введем новые ограничения в дополнение к ранее принятым – на классы ресурсов, периодичные задачи с бинарным жестким распределением и независимым распределением частоты, а также рассмотрим временные пределы реализации задач при формировании параллельных процессов.

*Ограничения на классы ресурсов.* Решения, рассматриваемые до сих пор, были связаны, прежде всего, с распределением процессоров. Вычислительные ограничения выражались в терминах времени

выполнения и отношений предшествования. Следовательно, предполагалось, что процессор является единственным ресурсом, необходимым для выполнения работ.

Признание факта, что задаче, кроме процессора, могут потребоваться дополнительные ресурсы, привело к исследованиям «систем с ограниченными ресурсами», в которых предполагается потребность в различных ресурсах, количество которых ограничено. Указанная модель расширяет понятие стандартной модели, состоящей из множества  $r$  задач неравной длительности, связанных отношением предшествования и выполняемых на неприоритетной основе набором из  $n$  идентичных процессоров. Дополнительно предполагается наличие множества ресурсов  $R = \{R_1, \dots, R_S\}$ . Если задаче  $T_i$  необходим ресурс  $R_j$ , то это требование принимается во внимание в течение всего периода выполнения задачи. Потребность задачи  $T_j$  в ресурсе  $R_i$  обозначается через  $p_{ij}$  ( $0 \leq p_{ij} \leq 1$ ).

Пусть  $r_i(t)$  обозначает общее количество ресурсов  $R_i$ , которое используется в момент времени  $t$ . Тогда  $r_i(t) = \text{Sum}(p_{ij})$  для всех  $T_j$ , выполняемых в момент времени  $t$  и  $r_i(t) \leq 1$ . Основная проблема заключается в определении того, в какой степени использование различных списков планов для этой модели влияет на время завершения  $w$ .

Допустим, что для двух произвольных списков  $L$  и  $L'$  расширенная система из  $n$  процессоров выполняет набор из  $r$  задач с результирующими временами завершения  $w$  и  $w'$  соответственно. Для такой среды предлагается следующее решение:

- при  $R = \{R_1\}$ , когда в системе существует только один вид ресурсов, отличных от процессора,  $w/w' \leq n$ ;
- при  $R = \{R_1\}$  и независимости всех задач  $w/w' \leq 3 - 1/n$ ;
- при  $R = (R_1, R_2, \dots, R_S)$ , независимости задач и  $n \geq r$  выполняется неравенство  $w/w' \leq S + 1$ .

Общий смысл этих результатов заключается в том, что добавление ресурсов в стандартную модель является причиной усиления ограничений на поведение в наихудших случаях. По существу, используется прежняя модель за исключением того, что все задачи для завершения требуют единичный интервал времени. Часто, используя эту модель, получаем ограничения на количество задач, число процессоров и правила формирования списка использованных планов. Показано, что эти алгоритмы ведут себя хуже, когда устраняются ограничения на ресурсы.

Можно предположить, что отдельные задачи требуют минимального количества памяти в дополнение к некоторому количеству

времени обработки. Тогда рассматривается система из  $m$  идентичных процессоров и  $n$  независимых задач, причем каждый процессор связан с определенным устройством памяти некоторой емкости. Когда процессор завершает выполнение задачи, в списке задач выбирается первая задача, чьи требования памяти не превышают его собственного объема. Для неприоритетной среды предлагаются ограничения и эвристические стратегии выбора задач на основе требований времени и памяти одновременно.

Теперь обратимся к периодичным задачам при формировании параллельных процессов. Ранее они рассматривались нами в связи с однопроцессорными планами. Тогда для выполнения временных ограничений для задач с более высокими приоритетами разрешалось приоритетное прерывание. Сейчас же рассматриваются неприоритетные многопроцессорные планы для набора независимых периодичных задач. Предполагается, что все задачи доступны одновременно. Их целью является минимизация числа процессоров, требуемых для выполнения ряда задач при временных ограничениях на начало/конец выполнения заданий.

Пусть  $E_i$  – максимальное время выполнения одной итерации задачи  $J_i$ , а  $f_i$  – частота выполнения. Каждой задаче  $J_i$  соответствуют два параметра  $J_i$ :  $(f_i, E_i)$ ,  $1 \leq i \leq n$ , где  $n$  – количество включаемых в план задач. Период повторения равен  $T_i$  – величине, обратной  $f_i$ .

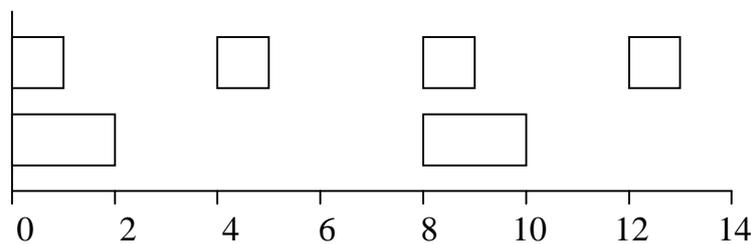
Рассмотрим два класса задач:

- 1) когда  $n$  задач с  $J_1$  по  $J_n$  распределены так, что  $f_i > f_{i+1}$ , то предполагается, что  $f_i = 2f_{i+1}$ ;
- 2) когда допускаются задачи с любой частотой.

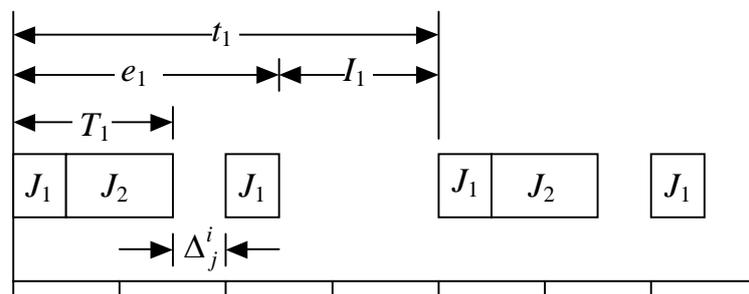
*Периодичные задачи с бинарным частотным распределением.* Множество задач, удовлетворяющих бинарному частотному распределению, приведено в табл. 3.8. Рассмотрим задачи  $J_1$  и  $J_2$ , спланированные на разных процессорах (рис. 3.23, а) и план, уменьшающий количество процессоров с двух до одного (рис. 3.23, б). Проблемой в данном случае является определение минимального количества процессоров без перебора всех возможных альтернатив. Заметим, что слияние двух задач (рис. 3.23, б) создает новую периодичную задачу с периодом  $t_1$ , равным  $2T_1$ , и временем выполнения  $e_1$ , равным  $T_1 + E_1$ . Кроме того, имеются два промежутка с простоями:  $I_1$  – периодичный простой с длительностью  $t_1 - e_1$ , и  $\Delta_2^1$  принудительное время простоя длиной  $I_1 - E_2$  ( $\Delta_j^i$  указывает, что принудительное время простоя получается, когда  $J_j$  объединяется с  $J_i$ ).

**Характеристики для множества задач  
с бинарным частотным распределением**

Задача	Частота	Период	Время выполнения
$J_1$	1/4	4	1
$J_2$	1/8	8	2
$J_3$	1/16	16	1,5
$J_4$	1/32	32	5
$J_5$	1/64	64	3



а



б

Рис. 3.23. Планирование периодических задач с бинарным частотным распределением:  
а – временная диаграмма для первых двух задач из табл. 3.8; б – ослабление ограничений на число процессоров путем слияния задач

В процессе объединения остальных задач плана нет необходимости рассматривать размещение задач в интервале принудительного простоя. Вместо этого для такой среды план с минимальным числом процессоров формируется в соответствии со следующим алгоритмом:

1) пусть  $J_1^*, J_2^*, \dots$  – подмножество задач, назначенных процессорам  $P_1, P_2, \dots$ . Сначала  $J_1^* = J_2^* = \dots = \emptyset$ , а  $I_1 = I_2 = \dots = \infty$ . Всякий раз, когда задача  $J_j$  назначается пустой  $J_i^*$ ,  $I_i = T_j - E_j$ ;

2) чтобы определить  $J_1$ , необходимо найти наименьшее  $I$  такое, что  $E_i \leq I_i$ , и назначить  $J_i$  к  $J_j^*$ .

Для набора задач из табл. 3.8 можно составить оптимальный план (рис. 3.24). Этот результат обобщается для случая, когда  $f_i = k(f_{i+1})$ , где  $k$  – положительное целое число больше 1.

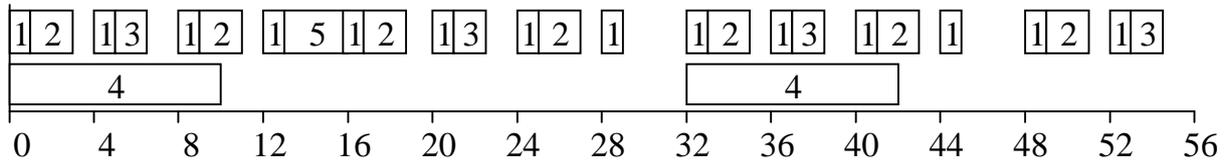


Рис. 3.24. Оптимальный план для задач из табл. 3.8

*Периодические задачи с независимым распределением частоты.* Рассмотрим случай, когда устранена частотная связь между задачами, принятая ранее. Как можно было бы ожидать, в таком виде задача становится более сложной, а оптимальное решение найти невозможно. Однако были разработаны эвристические подходы, и проведено их относительное сравнение с применением моделирования. Эти подходы можно разделить на три группы:

1) задачи располагаются в порядке уменьшения частоты, и их назначение также должно проходить в таком порядке;

2) задачи располагаются в порядке уменьшения критерия загрузки; критерий загрузки задачи  $J_i$ , обозначаемый  $L_i$ , определяется в виде  $L_i = E_i / T_i$ ;

3) сохраняется минимальная длина критического интервала: критический интервал между двумя задачами определяется как минимальный интервал между временем завершения первой задачи и временем начала выполнения второй в некоторой точке плана; нахождение этого интервала не включает первую итерацию обеих задач, где по определению начало выполнения второй задачи немедленно следует за завершением первой.

При тестировании данных методов, задачи разделялись на два класса. В первом классе частоты задач кратны более чем двум базовым частотам, а во втором – не более чем двум. Ни один из алгоритмов не показал значительного превосходства над другими. Однако подход 2 исключительно хорошо показал себя на задачах первого класса, подход 3 лучше решает некоторые задачи второго класса, а подходы 1 и 2 – задачи, которые оказались трудными для подхода 3.

В этом нет ничего необычного, так как число процессоров, необходимых для задач второго класса, было значительно меньше, чем для задач первого класса.

Также были выявлены некоторые интересные аномалии. Так, во многих случаях обнаружилось, что уменьшение частоты задач или времени их выполнения может привести к увеличению количества требуемых процессоров. И наоборот, требуемое количество процессоров может быть уменьшено увеличением частоты задач или времени выполнения, т. е. за счет увеличения загруженности процессора.

*Учет пределов при формировании параллельных процессов.* Ранее уже упоминался термин «планирование с учетом пределов» для среды, состоящей из одного процессора и набора периодических задач известной частоты и периода. Теперь рассмотрим многопроцессорную систему, в которой задачи с неравными временами выполнения связаны некоторыми отношениями предшествования и выполняются без учета приоритетов, в частности, случай когда терминальные и нетерминальные задачи требуют различного времени завершения. Изобразим граф, удовлетворяющий этим требованиям (рис. 3.25). На нем пара чисел вида  $A/B$  рядом с узлом обозначает задачу, чье время начала выполнения равно  $A$ , и которая должна быть выполнена за  $B$  единиц времени после начала.

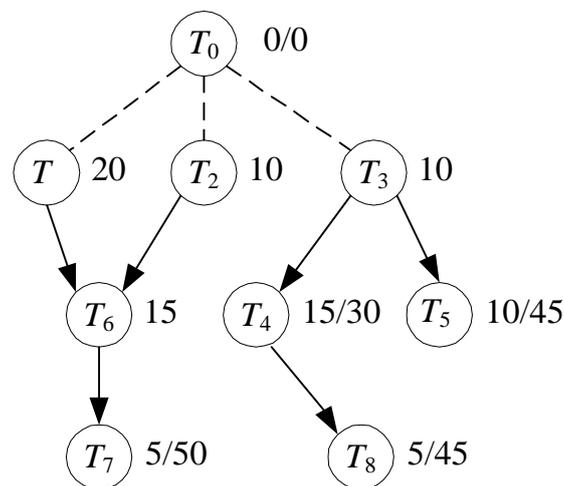


Рис. 3.25. Граф с многочисленными временными ограничениями

Эвристическое данной этой задачи является разновидностью планов с наидлиннейшим путем, рассмотренных ранее. Однако в этом случае большое число наидлиннейших путей может быть определено

для задач с пределами. Фактически процедура является вариацией разбиения по последнему предшествованию для случая с неравными временами задач и множественными пределами.

В своей работе по планированию с ограничениями Макнаутон рассмотрел случай, когда в качестве пределов определены ограничения на каждую задачу и потери в случае отказа. При задании так называемых абсолютных пределов задача не имеет никакой цены, если она не завершена до заданного предела. В случае с относительными пределами потери равны нулю до некоторой точки и монотонно возрастают (как функция времени завершения) вне (выше) этой точки. Основной вывод для многопроцессорного случая состоит в том, что все множество независимых задач имеет ограничение в нулевой момент времени, т. е. все задачи одновременно доступны и имеют равные приоритеты. Макнаутон показал, что для этого случая нет необходимости определять приоритетные прерывания, чтобы минимизировать функцию потерь.

Итак, мы рассмотрели некоторые из наиболее важных моделей и алгоритмов формирования детерминированных наборов задач ИУС. Предполагалось, что графы задач являются ациклическими, без ответвлений и что времена выполнения задач точно известны. Тем не менее следует упомянуть, что во многих компьютерных системах эти предположения не имеют смысла. Жд. Л. Баер рассматривал некоторые из реализаций циклов и ветвей в графах, Чанди, наряду с другими исследователями, – планы, в которых времена выполнения задач точно не известны, а большинство работ Конвэя, Максвелла и Миллера посвящено так называемой общей  $n$ -задачной  $m$ -машинной конвейерной проблеме.

Приведенные результаты показали, что эффективные оптимальные алгоритмы могут существовать только в некоторых частных случаях, и предполагают возможность активного исследования эвристических методов. Таким образом, задачу построения детерминированных графов для использования в реальных вычислительных средах ИУС можно считать выполненной. Результаты могут быть также интересны и в рамках теории исследования операций с учетом предположения о детерминированности процессов (что не всегда применимо на практике). Единственным способом решения практических задач формирования параллельных процессов может оказаться либо хорошо проверенный эвристический алгоритм, либо некоторый статистический метод. Последний представляет собой новый подход к решению проблемы.

## КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Приведите основные положения алгоритмизации технологических процессов и математического обеспечения АСУ.
2. В чем сущность методов математического планирования эксперимента?
3. Назовите основные положения метода группового учета аргументов.
4. Каковы основные методы оценки и оптимизации состояния объекта управления в АСУ?
5. Охарактеризуйте процессы параллельной обработки информации в АСУ.
6. Опишите взаимнообратные задачи формирования параллельных процессов в АСУ.
7. Каким образом классифицируют детерминированные модели формирования параллельных процессов в АСУ?
8. Сравните конвейерные и мультипроцессорные модели.
9. Приведите пример оптимального алгоритма Хью для многопроцессорной системы.

## Глава 4

# АЛГОРИТМИЧЕСКАЯ РЕАЛИЗАЦИЯ ПРОЦЕССОВ ОБРАБОТКИ ДАННЫХ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ

### 4.1. АЛГОРИТМ ОПТИМИЗАЦИИ СЕТЕВЫХ МОДЕЛЕЙ МЕТОДОМ СЛУЧАЙНОГО ПОИСКА С ПЕРЕСЧЕТОМ С ПЕРЕМЕННОЙ ВЕЛИЧИНОЙ ШАГА

Задача оптимизации на множестве сетевых моделей заключается в формировании модели оптимального расписания выполненных операций в ИУС, объединенных общими ограничениями на ресурсы. Рассмотрим эту модель.

Пусть имеется  $N$  сетевых моделей процессов (планов), причем  $q$ -й план ( $q = \overline{1, N}$ ) содержит множество  $V_q$  логически взаимосвязанных между собой операций. При этом сами возможные планы между собой не связаны. Для выполнения плана операций в ИУС используются ресурсы  $L$  видов (либо невзаимозаменяемые между собой группы исполнителей). Предполагается, что каждая  $(i, j)_q$  операция (работа)  $q$ -го плана, где  $i$  – начальное, а  $j$  – конечное событие работы, выполняется только одним видом ресурса, причем известно, каким. Множество работ  $q$ -го плана, выполняемых одним видом ресурса обозначим выражением

$$V_l^q \left( \bigcup_{l=1}^L V_q^l = V_q \right).$$

Каждой операции  $(i, j)_q$  ставится в соответствие продолжительность выполнения  $t(i, j)_q$ , наиболее ранний момент начала  $t_{\text{рн}}(i, j)_q$ , наиболее поздний момент окончания  $t_{\text{по}}(i, j)_q$  и удельные денежные затраты  $a(i, j)_q$  (затраты на единицу времени выполнения работы). Для каждого  $q$ -го проекта плана известны директивные сроки начала  $D_{\text{н}}^q$  и окончания  $D_{\text{о}}^q$ . Чтобы задача была корректной, необходимо выполнение следующих условий:

$$K_q \leq D_{\text{о}}^q - D_{\text{н}}^q,$$

$$K_q(j) \leq D_o^q(j) - D_H^q,$$

где  $K_q$  – длина критического пути в  $q$ -м проекте плана;  $K_q(j)$  – максимальный путь из начального события в событие  $j$ .

Планируемый период  $[T_H, T_o]$ ,  $T_H = \min_q D_H^q$ ,  $T_o = \max_q D_o^q$  разбивается на  $f$  интервалов  $\tau_k$  каждый ( $k = \overline{1, f}$ ). На  $k$ -й интервал для каждого вида используемых ресурсов заданы наличное их количество  $S_l^k$  и суммарная трудоемкость  $W_l^k = S_l^k \tau_k$ . Кроме того, считаются известными следующие функциональные зависимости:

$$w(i, j)_q^k = w(i, j)_q(t_H(i, j)_q, t_o(i, j)_q),$$

$$n(i, j)_q^k = n(i, j)_q(t_H(i, j)_q, t_o(i, j)_q),$$

$$c(i, j)_q^k = c(i, j)_q(t_H(i, j)_q, t_o(i, j)_q),$$

где  $w(i, j)_q^k$ ,  $n(i, j)_q^k$ ,  $c(i, j)_q^k$  – соответственно: трудоемкость, среднее количество используемых ресурсов и полезные трудозатраты в стоимостном выражении, относящиеся к той части продолжительности работы  $t(i, j)_q$ , которая покрывается результатом  $\tau_k$ ;  $t_H(i, j)_q$  и  $t_o(i, j)_q$  – моменты начала и окончания работ соответственно.

Поскольку состояние процесса формирования в произвольный момент времени определяется значениями  $t_H(i, j)_q$  и  $t_o(i, j)_q \forall (i, j)_q$ , то последние можно рассматривать как фазовые координаты объекта управления. Требуется найти такую точку фазового пространства, т. е. значения  $t_H(i, j)_q$  и  $t_o(i, j)_q \forall (i, j)_q$ , чтобы с учетом всех ограничений достигались номинальная и, по возможности, равномерная загрузка всех видов ресурсов на протяжении планируемого цикла.

Сформулированную выше задачу формализуем в следующем виде. Движение объекта в фазовом пространстве должно удовлетворять ограничениям на сроки выполнения процессов:

$$\max_{(i, j)_q \in V_q} (t_H(i, j)_q + t(i, j)_q) \leq D_o^q, \quad (4.1)$$

$$\min_{(i, j)_q \in V_q} t_H(i, j)_q \geq D_H^q, \quad q = \overline{1, N}$$

и ограничениям на порядок выполнения задач, задаваемых топологией сетевых моделей (комплексом информационно и по управлению взаимосвязанных задач ИУС):

$$G_q(J_q, V_q), \quad q = \overline{1, N}, \quad (4.2)$$

где  $J_q$  – множество вершин (событий) сетевой модели.

Пусть  $R^k$  – подмножество тех задач, которые покрываются  $k$ -м интервалом времени длительностью  $\tau_k$ ,  $k = \overline{1, f}$ ,  $R^k = \bigcup_{l=1}^L R_q^l$ . На подмножестве  $R_q^l$  так называемая задача многотемного сетевого планирования с учетом условий (4.1) и (4.2) представляется в виде целочисленной модели:

$$\sum_{q=1}^N \sum_{(i,j)_q \in V_q^l} \varphi_k(l) c(i, j)_q^k x(i, j)_q^k = \max, \quad (4.3)$$

$$\sum_{q=1}^N \sum_{(i,j)_q \in V_q^l} w(i, j)_q^k x(i, j)_q^k \leq w_l^k, \quad l = \overline{1, L}, \quad (4.4)$$

где

$$x(i, j)_q^k = \begin{cases} 1, & \text{если } (i, j)_q \text{ выполняется на } k\text{-м интервале,} \\ 0, & \text{в противном случае;} \end{cases}$$

$\varphi_k(l)$  – функция штрафа.

Для оптимального управления на всем планируемом периоде  $[T_n, T_o]$  требуется оптимальность этого управления на любом интервале  $\tau_k$ . Следовательно, эффективность управления на всем планируемом периоде будет определяться целевой функцией вида

$$F = \sum_{k=1}^f \sum_{l=1}^L \sum_{q=1}^N \sum_{(i,j)_q^k \in V_q^l} \varphi_k(l) c(i, j)_q^k x(i, j)_q^k. \quad (4.5)$$

Задача (4.1)–(4.5) наиболее полно учитывает специфические особенности процессов в ИУС. Максимизация функционала (4.5) эквивалентна минимизации себестоимости выполнения всех операций.

В настоящее время для решения задач оптимизации сетевых моделей применяются два метода: старый испытанный алгоритм Д. И. Голенко [44], использующий метод статистического моделирования в сочетании с методом направленного случайного поиска, и сравнительно недавно предложенный алгоритм, основанный на комбинации метода статистического моделирования и точного метода неявного перебора – алгоритм «ветвей и границ».

Алгоритм Д. И. Голенко имеет три основных недостатка:

- применение метода статистических испытаний при наличии большого числа сложных ограничений (см. выражения (4.1), (4.2) и (4.4)) требует значительных затрат машинного времени;

- при нарушении в процессе поиска ограничений (при выходе из допустимой области поиска) возврат осуществляется также методом статистических испытаний;

- направленный случайный поиск ведется с постоянным либо монотонно убывающим шагом, что малоэффективно при наличии сильной овражности (использование штрафных функций необходимо влечет за собой появление овражности) и многоэкстремальности оптимизируемой функции.

В связи с этим для решения сформулированной выше задачи используем алгоритм случайного поиска с пересчетом с переменной величиной шага, который осуществляет направленный поиск как при достижении допустимой области поиска, определяемой заданными ограничениями, так и непосредственно в ней.

Опишем процедуру поэтапно.

*Этап I.* В соответствии с топологией сети (4.2) и известными  $t_{\text{рн}}(i, j)_q$  и  $t_{\text{по}}(i, j)_q$  рассчитываем длину критического пути в каждом проекте плана операций  $k_q$ , а также резервы времени всех работ:

$$P(i)_q = t_{\text{по}}(i, j)_q - t_{\text{рн}}(i, j)_q + D_o^q - D_n^q - k_q, \quad q = \overline{1, N}.$$

Вычисляем величину шага  $\Pi_0 = \frac{1}{2} \max P(i)_q$ . Максимум здесь определяется по всем работам, входящим в  $N$  рассматриваемых проектов.

*Этап II.* Определяем начальную точку поиска  $T_0$  (положение начал операций проектов) по правилу

$$t_n^o(i, j)_q = t_{\text{рн}}(i, j)_q + \frac{1}{2} P(i)_q, \quad \forall (i, j)_q, \quad q = \overline{1, N}.$$

*Этап III.* Проверяем выполнения условия (4.4) в найденной точке для каждого  $k$ -го интервала  $k = \overline{1, f}$ , т. е. требования о том, чтобы ожидаемая трудоемкость не превышала наличную по каждому виду ресурса ( $l = \overline{1, L}$ ):

$$\delta_k = \begin{cases} 0, & \text{если } \sum_{q=1}^N \sum_{(i,j)_q \in V_q^l} w(i,j)_q^k x(i,j)_q^k, \forall (i,j)_q, q = \overline{1, N}, \\ 1, & \text{в противном случае.} \end{cases}$$

*Этап IV.* На данном этапе производится подсчет интервалов, на которых условие (4.4) не выполняется. Определяется

$$\Delta = \sum_{k=1}^f \delta_k.$$

Целочисленная функция  $\Delta$  является функцией качества при достижении допустимой области. Шаг считается удачным, если значение  $\Delta$  в новой точке меньше, чем в предыдущей. Если  $\Delta > 0$ , т. е. допустимая область еще не достигнута, то можно переходить к этапу VIII, если  $\Delta = 0$ , т. е. найденная точка находится внутри допустимой области – к вычислению функции качества (4.5).

*Этап V.* В функционале (4.5)  $\varphi_k(l)$  – функция штрафа за невыполнение условия

$$s_l^k - \Delta_2 \leq \sum_{q=1}^N \sum_{(i,j)_q \in V_q^l} n(i,j)_q^k x(i,j)_q^k \leq s_l^k + \Delta_1, \quad (4.6)$$

где  $l = \overline{1, L}$  и  $k = \overline{1, f}$ , обозначающие, что наличное количество  $l$ -го вида ресурса в произвольный момент времени устанавливается с некоторыми допусками  $\Delta_1$  и  $\Delta_2$ .

Функция  $\varphi_k(l)$  определяется по формуле

$$\varphi_k(l) = 1 - \left[ b_l' \sum'_{t_i \in \tau_k} \{s_l(t_i) - \tilde{s}(t_i) + \tau_k\} + b_l'' \sum_{t_i \in \tau_k} \{\tilde{s}_l(t_i) - s(t_i) - \tau_k\} \right] / I_l,$$

где  $t_i$  – момент времени, в который происходит изменение потребного количества ресурса ( $t_i = t_{\text{н}}(i, j)_q$  или  $t_i = t_{\text{н}}(i, j)_q + t(i, j)_q$ ,  $q = \overline{1, N}$ );  $\tilde{s}_l(t_i)$  – суммарное количество  $l$ -го вида ресурса, требуемого в момент времени  $t_i$ ;  $\sum'$  – сумма по тем  $t_i$ , в которых происходит нарушение условия (4.6) справа (перегрузка);  $b'_l$  и  $b''_l$  – коэффициенты штрафа за недогрузку и перегрузку соответственно;  $I_l$  – число моментов  $t_i$  на  $k$ -м интервале  $\tau_k$ .

Запоминаем значение  $F_0 = F(T_0)$  и переходим этапу VI; если  $\eta \geq I$  ( $\eta$  – номер шага поиска) – к этапу X.

*Этап VI.* Моделируем  $M$ -мерный единичный случайный вектор  $\Xi$  ( $M = \sum_{q=1}^N M_q$ ), равномерно распределенный по всем направлениям фазового пространства. Обозначим через  $\xi(i, j)_q$  координату этого вектора, соответствующую операции  $(i, j)_q$  в  $q$ -м плане проекта.

*Этап VII.* Переходим из точки  $T_\eta$  в точку  $T_{\eta+1}$  по следующим формулам:

$$\hat{t}_{\text{н}}^{\eta+1}(i, j)_q = \begin{cases} t_{\text{н}}^{\eta}(i, j)_q + \text{Ш}_{\eta} \xi(i, j)_q, & \text{если } t_{\text{рн}}(i, j)_q \leq t_{\text{н}}^{\eta}(i, j)_q + \\ & + \text{Ш}_{\eta} \xi(i, j)_q \leq t_{\text{по}}(i, j)_q; \\ t_{\text{н}}^{\eta}(i, j)_q, & \text{если } t_{\text{н}}^{\eta}(i, j)_q + \text{Ш}_{\eta} \xi(i, j)_q < t_{\text{рн}}(i, j)_q \\ & \text{или } t_{\text{н}}^{\eta}(i, j)_q + \text{Ш}_{\eta} \xi(i, j)_q < t_{\text{по}}(i, j)_q; \end{cases}$$

$$t_{\text{н}}^{\eta+1}(i, j)_q = \begin{cases} \hat{t}_{\text{н}}^{\eta+1}(i, j)_q, & \text{если } \hat{t}_{\text{н}}^{\eta+1}(i, j)_q \leq \max_{(i, j)_q \in V_q} (t_{\text{рн}}(i, j)_q + t(i, j)_q); \\ \max_{(i, j)_q \in V_q} (t_{\text{рн}}(i, j)_q + t(i, j)_q), & \forall (i, j)_q \in V_q, q = \overline{1, N}, \\ & \text{в противном случае.} \end{cases}$$

Чтобы можно было осуществить возврат в точку  $T_\eta$ , вычисляются величины

$$\delta(i, j)_q = t_{\text{н}}^{\eta+1}(i, j)_q - t_{\text{н}}^{\eta}(i, j)_q, \quad \forall (i, j)_q, q = \overline{1, N},$$

которые хранятся в памяти до следующего шага.

*Этап VIII.* Если функция  $\Delta$  вычисляется в первый раз, что случается либо в начале поиска, либо после того, как вычислялась функция качества (4.5) (алгоритм работал в допустимой зоне), то присваиваем  $\Delta_0$  значение  $\Delta$  ( $\Delta_0 = \Delta$ ) и возвращаемся к этапу VI. В противном случае формируем новую величину шага по формуле

$$\text{Ш}_\eta = \begin{cases} \text{Ш}_{\eta-1}, \text{ если } \Delta < \Delta_0 \text{ и } Y(\Delta) < \delta(\Delta) \\ \quad \text{или } \Delta \geq \Delta_0 \text{ и } H(\Delta) < \chi(\delta); \\ \alpha(\Delta)\text{Ш}_{\eta-1}, \text{ если } \Delta < \Delta_0 \text{ и } Y(\Delta) = \delta(\Delta); \\ \beta(\Delta)\text{Ш}_{\eta-1}, \text{ если } \Delta \geq \Delta_0, H(\Delta) = \chi(\delta), \\ \quad \beta\text{Ш}_{\eta-1} \geq \varepsilon(\Delta); \\ \text{Ш}_0, \text{ если } \Delta \geq \Delta_0, H(\Delta) = \chi(\delta), \\ \quad \beta\text{Ш}_{\eta-1} < \varepsilon(\Delta), \rho(\Delta) < \Lambda(\Delta), \end{cases} \quad (4.7)$$

где  $Y(\Delta)$  – число удачных подряд шагов;  $\delta(\Delta)$  – целое, наперед заданное число, которое определяет, сколько необходимо сделать подряд удачных шагов, чтобы величина шага увеличивалась;  $\alpha(\Delta) > 1$  – коэффициент увеличения шага;  $H(\Delta)$  – число неудачных подряд шагов;  $\chi(\delta)$  – целое, наперед заданное число, определяющее количество неудач, после которого следует уменьшение шага;  $\beta(\Delta) > 1$  – коэффициент уменьшения шага;  $\varepsilon(\Delta)$  – заданная точность, делать шаг меньше которой не имеет смысла;  $\rho(\Delta)$  – целое число, определяющее, сколько раз шаг уменьшался до заданной точности (если  $\rho(\Delta) = \lambda(\Delta)$ , которое задается заранее, то поиск прекращается, и точка  $T_\eta$  принимается за оптимальную).

Если заранее известно, что оптимизируемая функция является унимодальной, поиск можно прекращать сразу после того, как величина шага станет меньше заданной точности  $\varepsilon(\Delta)$ . В противном случае многократное ( $\lambda(\Delta)$  раз) увеличение шага до  $\text{Ш}_0$  при уменьшении его до  $\varepsilon(\Delta)$  позволяет избежать окончания поиска в локальном экстремуме. При удачном шаге ( $\Delta < \Delta_0$ ) значение  $H(\Delta)$  обнуляется, и происходит переход к этапу VI; при неудачном ( $\Delta \geq \Delta_0$ ) – обнуляется  $Y(\Delta)$ , и происходит переход к этапу IX. При попадании в допустимую область поиска (этап V) обнуляется как  $Y(\Delta)$ , так и  $H(\Delta)$ . Для достижения большей надежности поиска можно при этом обнулять и  $\rho(\Delta)$ .

*Этап IX.* На данном этапе происходит возврат в исходную точку по формуле

$$t_{\text{н}}^{\eta+2}(i, j)_q = t_{\text{н}}^{\eta+1}(i, j)_q - \delta(i, j)_q, \forall (i, j)_q, q = \overline{1, N}$$

и переход к этапу VI.

*Этап X.* Вновь полученное значение функции качества  $F(T_{\eta})$  сравнивается со значением, хранимым в памяти в качестве наилучшего  $F_0$ , и формируется величина шага поиска в допустимой зоне. Если  $F(T_{\eta}) > F_0$ , шаг считается удачным, и  $F_0$  присваивается значение  $F(T_{\eta})$ , в противном случае шаг считается неудачным, и значение  $F_0$  остается без изменения. Таким образом, в памяти постоянно хранится значение  $F_0$  – наилучшее из всех полученных в процессе поиска.

Величина шага формируется по правилу, аналогичному (4.7):

$$\text{Ш}_{\eta} = \begin{cases} \text{Ш}_{\eta-1}, & \text{если } F(T_{\eta}) > F_0 \text{ и } Y(F) < \chi(F) \\ & \text{или } F(T_{\eta}) \leq F_0 \text{ и } H(F) < \delta(F); \\ \alpha(F)\text{Ш}_{\eta-1}, & \text{если } F(T_{\eta}) > F_0 \text{ и } Y(F) = \chi(F); \\ \beta(F)\text{Ш}_{\eta-1}, & \text{если } F(T_{\eta}) \leq F_0, H(F) = \delta(F), \beta(F)\text{Ш}_{\eta-1} \geq \varepsilon(F); \\ \text{Ш}_0, & \text{если } F(T_{\eta}) \leq F_0, H(F) = \delta(F), \beta(F)\text{Ш}_{\eta-1} < \varepsilon(F), \rho(F) < \Lambda(F), \end{cases}$$

где все параметры имеют тот же смысл, что и в (4.7). В зависимости от специфики конкретной задачи значения параметров при поиске вне допустимой области и при поиске в допустимой области могут задаваться различными.

Если шаг был удачным, то переходим к этапу VI, в противном случае – к этапу IX. При  $\rho(F) = \Lambda(F)$  в качестве оптимальной принимается точка (совокупность моментов начал всех операций в ИУС)  $t_{\text{н}}(i, j)_q, q = \overline{1, N}$ , соответствующая  $F_0$ , которая хранится в памяти вместе со значением  $T_0$ .

Таким образом, применение многоэтапной процедуры оптимизации позволяет для имеющихся наборов величин  $t_{\text{н}}(i, j)_q$ , соответствующих оптимальному плану процесса с учетом продолжительности всех операций  $t(i, j)_q$ , рассчитать сроки окончания операций  $t_0(i, j)_q$ . В результате полностью определяется оптимальный план, т. е. формируется оптимальное расписание выполняемых в ИУС операций, объединенных общими ограничениями на ресурсы.

## 4.2. АЛГОРИТМИЧЕСКАЯ GERT-ПРОЦЕДУРА ОПРЕДЕЛЕНИЯ НОРМАТИВНЫХ ВРЕМЕН ВЫПОЛНЕНИЯ ПАРАЛЛЕЛЬНЫХ ПРОЦЕССОВ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

В качестве формальной базы алгоритмической GERT-процедуры используется аппарат стохастических сетей и графического метода оценки и пересмотра планов. До сих пор мы рассматривали лишь те модели реализации параллельных процессов, которые описываются детерминированными сетями. Для полного выполнения типичной сети такого класса необходимо выполнение всех дуг. Из этого условия следует, что в такую модель не могут быть включены операции с обратной связью, поскольку они представляются петлями, существование которых, в свою очередь, означает, что конечный узел операции должен быть выполнен раньше начального. В области детерминированных сетей наиболее полно были изучены две модели. В первой из них – модели критического пути – время выполнения каждой дуги фиксировано, во второй – модели PERT – для каждой дуги существует несколько возможных времен выполнения.

При моделировании работы промышленных автоматизированных комплексов и информационно-управляющих систем нередко наиболее гибкими и полезными оказываются сетевые модели со стохастической структурой. Стохастическую сеть определим как сеть, которая может быть выполнена только при выполнении некоторого подмножества дуг; при этом время выполнения каждой дуги выбирается в соответствии с вероятностным распределением. В стохастических сетях для выполнения узла не является необходимым выполнение всех дуг, входящих в него. Поэтому в таких моделях допускается существование циклов и петель.

Узлы стохастической сети можно интерпретировать как состояния системы, а дуги – как переходы из одного состояния в другое. Такие переходы можно рассматривать как выполнение обобщенных операций, характеризующихся плотностью распределения, или функцией массы, и вероятностью выполнения.

Каждый внутренний узел стохастической сети выполняет две функции, одна из которых касается входа в узел, а другая – выхода из него. Обычно эти функции называют входной и выходной. Входная функция определяет условие, при котором узел может быть выполнен, а выходная – совокупность условий, связанных с результатом выполнения узла (т. е. с помощью нее указывается, должны ли выпол-

няться все операции, которым данный узел непосредственно предшествует, или только одна из них). Отметим, что начальный узел сети выполняет только выходную функцию, в то время как конечный – только входную.

Существуют три типа входных функций:

- 1) узел выполняется, если выполнены все дуги, входящие в него;
- 2) узел выполняется, если выполнена любая дуга, входящая в него;
- 3) узел выполняется, если выполнена любая дуга, входящая в него, при условии, что в заданный момент времени может выполняться только одна дуга.

Выходные функции делятся на два типа:

- 1) детерминированные – все дуги, выходящие из узла, выполняются, если этот узел выполнен;
- 2) вероятностной – если узел выполнен, выполняется ровно одна дуга, выходящая из него; выбор такой дуги может быть описан с помощью вероятности.

Мы будем рассматривать два типа узлов: узлы с третьим типом входной функции и детерминированной выходной функцией и узлы с третьим типом входной функции и вероятностной выходной функцией. Сети, содержащие только эти два типа узлов, называются GERT-сетями, а сами узлы – GERT-узлами (рис. 4.1).

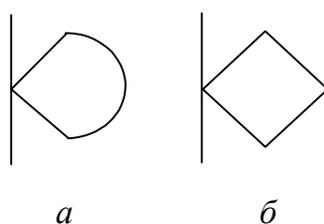


Рис. 4.1. Типы GERT-узлов:  
*a* – детерминированный выход;  
*б* – вероятностный выход

В качестве примера рассмотрим GERT-сеть простой ИУС технологического контроля и диагностики качества продукции (рис. 4.2). В результате выполнения операции проверки система контроля и диагностики определяет, какие детали следует сдать в металлолом, какие – исправить, а какие – отправить на линию сборки. После исправления детали могут быть проданы в розницу, сданы в металлолом или отправлены непосредственно на линию сборки.

В данной сети  $p_1 + p_2 + p_3 = 1$  и  $p_4 + p_5 + p_6 = 1$ . Кроме того, дуги (4, 2), (1, 4), (1, 2), (1, 3), (4, 3) и (4, 5) соответствуют физическим процессам, которые можно описать плотностями распределения. Например, случайные величины, соответствующие операциям (1, 4) и (4, 5), могут быть нормально распределенными, операциям (1, 2) и (4, 3) – экспоненциально распределенными, а операциям (1, 3) и (4, 2) – равномерно распределенными.

Методы GERT-системы позволяют ответить на следующие вопросы.

1. Какова вероятность того, что деталь будет сдана в металлолом?

2. Какова вероятность того, что деталь будет использована на линии сборки?

3. Какова вероятность того, что деталь поступит в розничную продажу?

4. Чему равны математическое ожидание и дисперсия времени, необходимого для изготовления детали, которая поступает на линию сборки?

5. Сколько времени будет потеряно, если деталь сдается в металлолом?

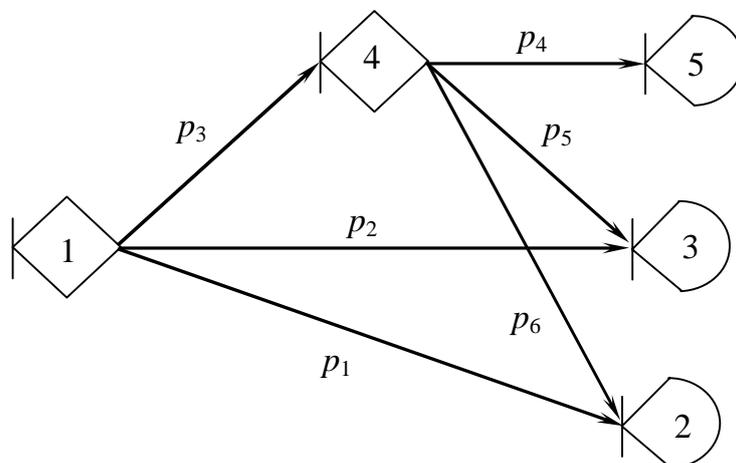


Рис. 4.2. Пример GERT-сети

Опишем аналитический подход, позволяющий ответить на эти вопросы. Для его реализации необходимо рассмотреть основные процедуры системы GERT. В качестве модели процесса возьмем сеть  $G = (N, A)$ , содержащую только GERT-узлы, которые образуют множество  $N$ . Пусть время выполнения операции  $(i, j)$  – случайная вели-

чина  $Y_{ij}$ . По определению операция  $(i, j)$  может быть выполнена только в том случае, если выполнен узел  $i$ . Поэтому для изучения вопросов, связанных с выполнением этой операции, необходимо знать условную вероятность (в дискретном случае) или плотность распределения (в непрерывном случае) случайной величины  $Y_{ij}$  при условии, что узел  $i$  выполнен. Это, в свою очередь, позволит провести исследования, связанные с выполнением всей сети. В частности, возможно будет найти моменты распределения времени выполнения сети, с помощью которых вычисляются математическое ожидание и дисперсия времени выполнения сети и соответствующего ей процесса. Некоторые наиболее важные функции распределения, соответствующие им производящие функции моментов, первые (математические ожидания) и вторые моменты относительно начала координат приведены в табл. 4.1.

Таблица 4.1

**Моменты и производящие функции моментов**

Тип распределения	$M_E(s)$	Математическое ожидание	Второй момент
Биномиальное ( $B$ )	$(pe^s + 1 - p)^n$	$np$	$Np(np + 1 - p)$
Дискретное ( $D$ )	$\frac{p_1 e^{sT_1} + p_2 e^{sT_2} + \dots}{p_1 + p_2 + \dots}$	$\frac{p_1 T_1 + p_2 T_2 + \dots}{p_1 + p_2 + \dots}$	$\frac{p_1 T_1^2 + p_2 T_2^2 + \dots}{p_1 + p_2 + \dots}$
Экспоненциальное ( $E$ )	$\left(1 - \frac{s}{a}\right)^{-1}$	$\frac{1}{a}$	$\frac{2}{a^2}$
Гамма ( $GA$ )	$\left(1 - \frac{s}{a}\right)^{-b}$	$\frac{b}{a}$	$\frac{b(b+1)}{a^2}$
Геометрическое ( $GE$ )	$\frac{pe^s}{1 - e^s + pe^s}$	$\frac{1}{p}$	$\frac{2-p}{p^2}$
Отрицательное биномиальное ( $NB$ )	$\left(\frac{p}{1 - e^s + pe^s}\right)^r$	$\frac{r(1-p)}{p}$	$\frac{r(1-p)(1+r-rp)}{p^2}$
Нормальное ( $NO$ )	$e^{sm + (1/2)s^2\sigma^2}$	$m$	$m^2 + \sigma^2$
Пуассона ( $P$ )	$e^{\lambda(e^s - 1)}$	$\lambda$	$\lambda(1 + \lambda)$
Равномерное ( $U$ )	$\frac{e^{sa} - e^{sb}}{(a-b)s}$	$\frac{a+b}{2}$	$\frac{a^2 + ab + b^2}{3}$

Положим, что  $f_{ij}$  – условная вероятность или плотность распределения времени выполнения операции  $(i, j)$ . Условная производящая функция моментов случайной величины  $Y_{ij}$  определяется выражением  $M_{ij}(s) = E[e^{sy_{ij}}]$ , или в виде

$$M_{ij}(s) = \begin{cases} \int e^{sy_{ij}} f(y_{ij}) dy_{ij} & \text{(для непрерывной случайной величины);} \\ \sum e^{sy_{ij}} f(y_{ij}) & \text{(для дискретной случайной величины).} \end{cases}$$

В частности,  $M_{ij}(s) = E[e^{sa}] = e^{sa}$  при  $y_{ij} = a = \text{const}$ . Если  $a = 0$ , то  $M_{ij}(s) = 1$ .

Пусть  $p_{ij}$  – вероятность того, что операция  $(i, j)$  будет выполнена при условии выполнения узла  $i$ . Для случайной величины  $Y_{ij}$  определим  $W$ -функцию следующим образом:

$$W_{ij}(s) = p_{ij}M_{ij}(s). \quad (4.8)$$

С помощью преобразования (4.8) всегда можно определить сеть  $G'$ , структура которой идентична структуре сети  $G$ , только вместо двух параметров дуг  $p_{ij}$  и  $y_{ij}$  присутствует один параметр  $W_{ij}$  (рис. 4.3).

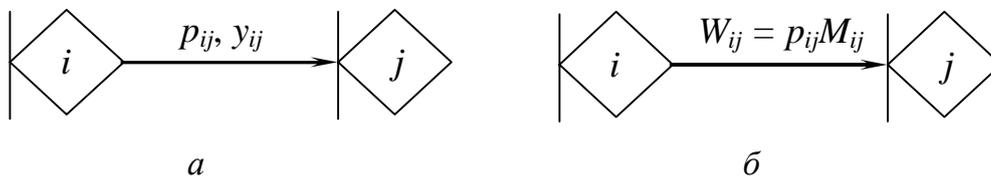


Рис. 4.3. Сети  $G$  и  $G'$ :  
 $a$  – элемент сети  $G$ ;  $b$  – элемент сети  $G'$

В описание GERT-системы в качестве параметра дуги мы включили время выполнения соответствующей операции. В действительности можно рассматривать также любой характерный параметр процесса, который обладает аддитивностью по дугам любого пути.

Если времена выполнения операций сети  $G$  – независимые случайные величины, то  $G'$  обладает рядом свойств, представляющих интерес с вычислительной точки зрения. Для изучения этих свойств мы рассмотрим три частных случая, когда  $G'$  состоит:

- 1) из двух последовательных дуг;
- 2) двух параллельных ветвей,
- 3) одной ветви и одной петли.

*Последовательные дуги.* Рассмотрим простую сеть, состоящую из двух последовательных ветвей. Эти две ветви могут быть заменены одной эквивалентной им ветвью, как будет показано ниже. Исходные ветви содержат  $W$ -преобразования:

$$W_{ij}(s) = p_{ij}M_{ij}(s),$$

$$W_{ik}(s) = p_{ik}M_{ik}(s).$$

$W$ -функция для эквивалентной ветви  $(i, k)$  имеет вид  $W_{ik}(s) = p_{ik}M_{ik}(s)$ . Напомним, что производящая функция моментов суммы двух независимых случайных величин равна произведению производящих функций моментов этих величин. В этом случае, поскольку  $p_{ik} = p_{ij}p_{jk}$  и  $M_{ik}(s) = [M_{ij}(s)][M_{jk}(s)]$ , получим:

$$W_{ik}(s) = [p_{ij}M_{ij}(s)][p_{jk}M_{jk}(s)] = W_{ij}(s)W_{jk}(s). \quad (4.9)$$

Основной результат (4.9) может быть обобщен на случай трех и более ветвей:  $W$ -функция для эквивалентной ветви равна произведению  $W$ -функций для последовательных ветвей.

*Параллельные ветви.* Рассмотрим простую сеть, состоящую из двух параллельных ветвей. Доказано, что эти ветви могут быть заменены эквивалентной ветвью. Пусть  $(i, j)$  – такая ветвь. По определению  $W_{ij}(s) = p_{ij}M_{ij}(s)$ . В этом случае  $p_{ij} = p_a + p_b$  и  $M_{ij}(s) = [p_a M_a(s) + p_b M_b(s)] / (p_a + p_b)$ . Отсюда имеем:

$$W_{ij}(s) = [p_a + p_b] \left[ \frac{p_a M_a(s) + p_b M_b(s)}{p_a + p_b} \right] = W_a(s) + W_b(s). \quad (4.10)$$

Формула (4.10) также может быть обобщена на случай сетей, состоящих из трех и более параллельных ветвей:  $W$ -функция для эквивалентной ветви равна сумме  $W$ -функций для параллельных дуг.

*Петли.* Рассмотрим простую сеть, состоящую из одной петли и одной дуги. Она может быть преобразована в эквивалентную сеть, содержащую только одну дугу, которая, в свою очередь, может быть преобразована в другую сеть. Новая сеть при этом будет состоять из бесконечной последовательности параллельных цепей, каждая из которых представляет собой совокупность последовательных ветвей. Поэтому можно вначале свести каждую цепь новой сети к эквивалентной дуге, а затем преобразовать эти дуги в эквивалентную исходной системе сеть, состоящую из одной ветви.

Пусть  $(i, j)$  – ветвь, эквивалентная сети. Из выражений (4.9) и (4.10) следует, что вес ветви  $(i, j)$  определяется как  $W_{ij} = W_b + W_a W_b + W_a^2 W_b + \dots = W_b \left[ 1 + \sum_{m=1}^{\infty} W_a^m \right]$ . Данное выражение, в котором мы временно опустили аргументы  $W$ -функций, можно упростить, зная, что биномиальный ряд  $(1 - W_a)^{-1}$  раскладывается следующим образом:

$$(1 - W_a)^{-1} = 1 + W_a + W_a^2 + W_a^3 + \dots = 1 + \sum_{m=1}^{\infty} W_a^m.$$

Окончательно получим:

$$W_{ij}(s) = W_b(s) [1 - W_a(s)]^{-1} = W_b(s) / [1 - W_a(s)]. \quad (4.11)$$

Следовательно, сеть сводится к одной-единственной эквивалентной ей ветви, для которой  $W_{ij}(s) = W_b(s) / [1 - W_a(s)]$ . Отметим, что описанная процедура может быть использована и для контуров, поскольку с помощью формулы (4.9) контур сводится к петле.

Таким образом, если GERT-сеть состоит из параллельных и последовательных цепей и (или) петель, то она может быть преобразована в эквивалентную сеть, состоящую из одной-единственной ветви. Данный результат обобщается на любую GERT-сеть, поскольку можно комбинировать базисные преобразования.

Работа системы GERT-моделей состоит из следующих основных шагов:

- 1) представление системы в виде стохастической сети с GERT-узлами;
- 2) определение для каждой дуги сети условной вероятности и производящей функции моментов;
- 3) вычисление для каждой дуги сети  $W$ -функции;
- 4) преобразование сети в эквивалентную сеть, состоящую из одной ветви.

При использовании GERT-системы необходимо учитывать основные положения и аппарат теории потоковых графов. Для нас существенно, что процесс может быть определен как совокупность активных взаимодействующих между собой элементов, которые выполняют некоторые функции. В дальнейшем мы будем рассматривать только такие процессы, в которых элементы и взаимосвязь между ними описываются линейными уравнениями. Наиболее широко для графического описания подобных систем используются потоковые графы,

в которых элементы системы представляются узлами, а взаимосвязь между ними (функции перехода) – дугами.

Основным элементом потокового графа является ориентированная ветвь, направленная из узла  $i$  в узел  $j$ , с параметром  $t_{ij}$ . Направление ветви указывает связь по входу и выходу между двумя переменными, представленными узлами этой ветви: узел  $i$  соответствует независимой переменной  $x_i$ , узел  $j$  – независимой переменной  $x_j$ . Параметр  $t_{ij}$  обычно называют коэффициентом пропускания дуги. Он равен множителю, используемому для преобразования величины  $x_i$  до рассмотрения ее как части величины  $x_j$ . Основное свойство потоковых графов заключается в том, что величина узла равна сумме преобразованных величин узлов, соединенных с рассматриваемым узлом входящими в него дугами.

Основные свойства потоковых графов рассмотрим на примере элементарного потокового графа (рис. 4.4). Прежде всего отметим, что эти свойства могут быть использованы для разработки методов, позволяющих обращаться непосредственно с элементами графа, в результате чего он будет преобразован в эквивалентный граф с более простой структурой, и, следовательно, решение задачи упростится. Наиболее известным результатом в данной области является топологическое уравнение Мейсона, которое может быть использовано для потоковых графов с произвольной структурой.

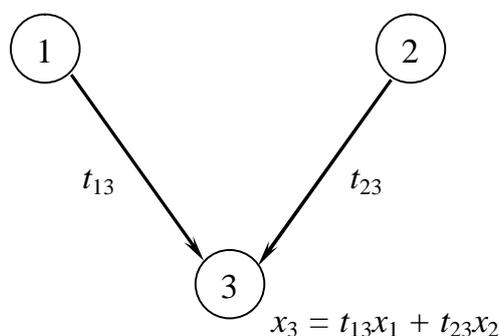


Рис. 4.4. Основное свойство потоковых графов

Для рассмотрения уравнения Мейсона введем определения петли, петли порядка  $n$  и замкнутого потокового графа. *Петля* – связная последовательность ориентированных ветвей, каждый узел которых является общим ровно для двух ветвей. Петлю обычно называют петлей первого порядка, чтобы указать на то, что она не содержит других петель, а каждого узла можно достичь из любого другого. Собственную петлю можно рассматривать как вырожденную петлю первого порядка. Множество  $n$  не связанных между собой петель первого по-

рядка называется *петлей порядка  $n$* , а граф, в котором каждая ветвь принадлежит, по крайней мере, одной петле, – *замкнутым потоковым графом*.

Далее будем рассматривать пример потокового графа, для которого могут быть найдены все петли и определен их порядок. Отметим, что потоковый граф является замкнутым, поскольку он целиком состоит из петель. Как было определено выше, петля второго (третьего) порядка – это множество, состоящее из двух (трех) не связанных между собой петель первого порядка. Например, петли  $L_1$  и  $L_2$  образуют петлю второго порядка, а петли  $L_1$ ,  $L_2$  и  $L_3$  – третьего. Петлю первого порядка можно рассматривать как цепь, состоящую из последовательно соединенных ориентированных ветвей, концевые узлы которой совпадают, поэтому коэффициент пропускания эквивалентного ей потокового графа равен произведению коэффициентов пропускания каждой ветви.

Как уже говорилось, петля порядка  $n$  состоит из  $n$  не связанных между собой петель первого порядка. Если каждую из петель первого порядка преобразовать в эквивалентный потоковый граф, состоящий из одной ветви, то исходную петлю порядка  $n$  можно будет рассматривать как связную последовательность таких ветвей. Отсюда следует, что коэффициент пропускания петли порядка  $n$  равен произведению коэффициентов пропускания  $n$  петель первого порядка.

Рассмотрим общий случай. Пусть  $L_{11}, L_{21}, \dots, L_{n1}$  –  $n$  не связанных между собой петель первого порядка заданной петли порядка  $n$ . Для каждой петли  $L_{k1}$  первого порядка эквивалентный коэффициент пропускания  $T_k$  равен произведению коэффициентов пропускания ветвей, принадлежащих этой петле:

$$T_k = \prod_{(i,j) \in L_{k1}} t_{ij}. \quad (4.12)$$

Следовательно, для петли порядка  $n$  эквивалентный коэффициент пропускания  $T(L_n)$  можно определить в виде

$$T(L_n) = \prod_{k=1}^n T_k = \prod_{k=1}^n \left[ \prod_{(i,j) \in L_{k1}} t_{ij} \right]. \quad (4.13)$$

Основной результат (4.13) будет использован для определения поведения процессов, которые могут быть описаны GERT-сетями.

Цель использования GERT-системы в стохастическом сетевом анализе процессов состоит в вычислении математического ожидания и дисперсии времени выполнения сети (которые рассматриваются здесь как общий параметр сети) и вероятности выполнения стока (или стоков). Очевидно, что коэффициент пропускания дуги GERT-сети есть соответствующая  $W$ -функция. Напомним, что  $W$ -функция дуги определяется как произведение вероятности выполнения этой дуги и производящей функции моментов для времени выполнения операции, представленной этой дугой.

Ранее мы показали, как определить все петли замкнутого потокового графа. Чтобы применить эти результаты к открытой сети, необходимо ввести дополнительную дугу с  $W$ -функцией  $W_a(s)$ , соединяющую сток  $t$  с источником  $s$ . Затем для модифицированной сети нужно найти все петли (вплоть до максимально возможного порядка). Функция  $W_A(s)$  необходима для того, чтобы определить эквивалентную  $W$ -функцию для исходной сети. Исходная сеть может быть изображена в виде «черного ящика» с  $W$ -функцией  $W_E(s)$  (рис. 4.5).

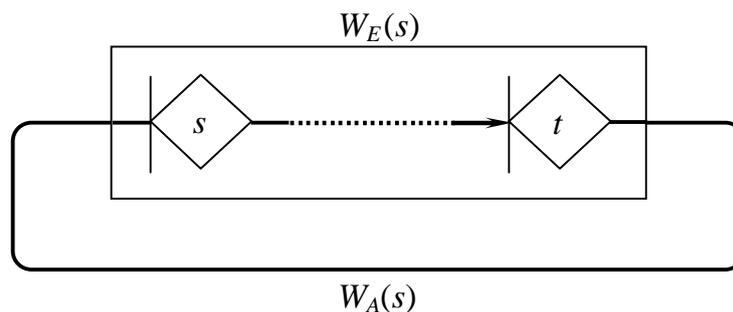


Рис. 4.5. Замкнутая стохастическая сеть

Напомним, что эквивалентный коэффициент пропускания для петли порядка  $n$  равен произведению коэффициентов пропускания  $n$  не связанных между собой петель первого порядка:

$$T(L_n) = \prod_{k=1}^n T_k. \quad (4.14)$$

Топологическое уравнение для замкнутых графов, известное также как правило Мейсона, имеет следующий вид:

$$H = 1 - \sum T(L_1) + \sum T(L_2) - \sum T(L_3) + \dots + (-1)^m \sum T(L_m) + \dots = 0, \quad (4.15)$$

где  $\sum T(L_i)$  – сумма эквивалентных коэффициентов пропускания для всех возможных петель  $i$ -го порядка.

Другими словами, при использовании топологического уравнения необходимо выполнить следующие шаги.

1. Вычислить эквивалентный коэффициент пропускания для всех петель порядка  $m$ .

2. Просуммировать полученные величины по всем петлям порядка  $m$  и результат умножить на  $(-1)^m$ . Отметим, что для петель с четным порядком данная величина положительна, а для петель с нечетным порядком – отрицательна. В топологическом уравнении (4.15) перед каждым слагаемым необходимо поставить знак «+» или «-».

3. К выражению, полученному на шаге 2, прибавить 1 и результат приравнять к нулю.

Рассматриваемый замкнутый потоковый граф (см. рис. 4.5) включает одну петлю первого порядка с эквивалентным коэффициентом пропускания, равным  $W_A(s)$ ,  $W_E(s)$ . По правилу Мейсона получаем, что  $1 - W_A(s) = 0$  или  $W_A(s) = 1 / W_E(s)$ . Отметим, что функция  $W_A(s)$  содержится в топологическом уравнении, поскольку является элементом, по крайней мере, одной петли первого порядка. Важность результата, полученного при рассмотрении данного примера, состоит в том, что если в топологическом уравнении  $W_A(s)$  заменить на  $1 / W_E(s)$  и решить его относительно  $W_E(s)$ , будет получена эквивалентная  $W$ -функция для исходной стохастической сети.

В качестве другого примера рассмотрим открытую сеть (рис. 4.6). Чтобы определить эквивалентную  $W$ -функцию для нее, нужно выполнить следующие шаги.

1. Замкнуть данную сеть дугой, ведущей из узла 4 в узел 1.
2. Найти все петли порядка  $n$ .
3. С помощью топологического уравнения (4.15) получить выражение для  $W_E(s)$ .

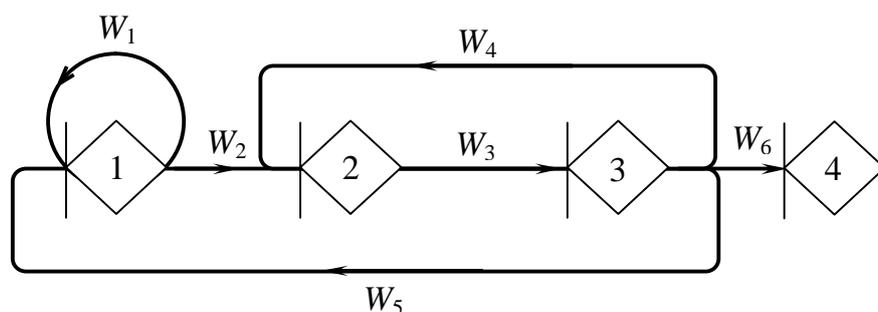


Рис. 4.6. Открытая стохастическая сеть

Данную сеть можно модифицировать, для удобства записи опустив аргумент  $s$  у  $W$ -функций (рис. 4.7).

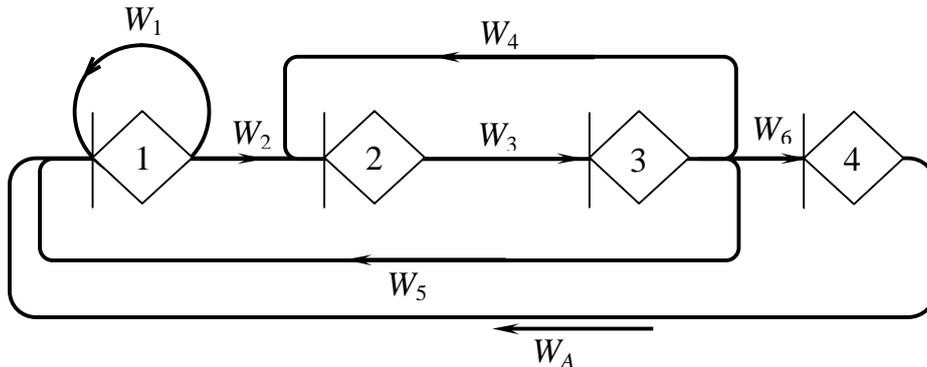


Рис. 4.7. Модифицированная замкнутая стохастическая сеть

Если  $W_A(s)$  заменить на  $1 / W_E(s)$ , то из выражения (4.13) для петли сети будут получены следующие коэффициенты пропускания:

– петли первого порядка:  $W_1, W_3W_4, W_2W_3W_5, W_2W_3W_6(1 / W_E)$ ;

– петля второго порядка:  $W_1W_3W_4$ .

Использував формулу (4.15), получим:

$$H = 1 - W_1 - W_3W_4 - W_2W_3W_5 - W_2W_3W_6(1/W_E) + W_1W_3W_4 = 0,$$

откуда:

$$W_E(s) = W_2W_3W_6 / (1 - W_1 - W_3W_4 - W_2W_3W_5 + W_1W_3W_4) = 0,$$

что является эквивалентной  $W$ -функцией для представленной выше сети (см. рис. 4.6).

С помощью этого равенства можно вычислить математическое ожидание и дисперсию времени выполнения сети (или подсети). Напомним, что  $M_E(s) = I$  при  $s = 0$ . Поскольку  $W_E(s) = p_E M_E(s)$ , то  $p_E = W_E(0)$ , откуда следует:

$$M_E(s) = W_E(s)/p_E = W_E(s) / W_E(0). \quad (4.16)$$

Отметим, что  $W_E(s)$  можно выразить через  $W$ -функции всех или некоторых ветвей исходной сети. Нетрудно также вычислить значение  $W_E(0)$ : для этого в выражении для  $W_E(s)$ , получаемом из формулы (4.15), просто необходимо принять  $s = 0$ .

Вычисляя  $j$ -ю частную производную по  $s$  функции  $M_E(s)$  и полагая  $s = 0$ , находим  $j$ -й момент  $\mu_{jE}$  относительно начала координат:

$$\mu_{jE} = \frac{\partial^j}{\partial s^j} [M_E(s)] \Big|_{s=0}. \quad (4.17)$$

В частности, первый момент  $\mu_{1E}$  относительно начала координат есть математическое ожидание времени выполнения сети, а дисперсия времени выполнения сети равна разности между  $\mu_{2E}$  и квадратом величины  $\mu_{1E}$ :

$$\sigma^2 = \mu_{2E} - (\mu_{1E})^2. \quad (4.18)$$

Рассмотренная алгоритмическая процедура метода графической оценки и анализа находит применение в случаях организации работ (задач), когда последующие задачи могут начинаться после завершения только некоторого числа предшествующих. При этом не все задачи, представленные на сетевой модели, должны быть выполнены для завершения проекта.

### **4.3. АЛГОРИТМ ДЕЙКСТРЫ В РЕШЕНИИ ЗАДАЧИ ОПРЕДЕЛЕНИЯ ВРЕМЕННОГО ПУТИ В ДЕТЕРМИНИРОВАННЫХ МОДЕЛЯХ**

Существующая аналитико-оптимизационная процедура [43], реализуемая в рамках детерминированной модели, направлена на решение задачи минимизации (сокращения) времени информационного взаимодействия структурных компонентов сети (информационной карты) путем выявления и сокращения холостых временных «окон». Также в рамках данной процедуры существует возможность оценить полное время реализации технологического цикла управления (ТЦУ) в рамках границ информационной карты. Однако не менее критичным с точки зрения управления становится задача расчета временных затрат маршрута управления на том или ином участке информационной карты, поскольку полное время реализации ТЦУ не позволяет делать вывод о том, где, а также за счет чего достигнуто сокращение холостых временных «окон».

С целью реализации подобной возможности рассмотрим алгоритм Дейкстры как один из методов сетевого анализа в решении задачи определения кратчайшего временного пути графа ТЦУ. Данный алгоритм применим по следующей причине. Современные автоматизированные системы управления можно рассматривать как совокупность сетей, предназначенных для транспортирования (передачи) и распределения потоков, в том числе энергетических и информационных. Поскольку каждая из подсистем АСУ имеет весьма сложную структуру, возникает необходимость в эффективном использовании уже существующих технических средств и в рациональном проекти-

ровании новых. При проектировании и усовершенствовании больших и сложных систем, а также при поиске путей их наиболее рационального использования существенное значение могут иметь методы сетевого анализа. Вводимый термин «временной путь» ( $t_p$ ) интерпретирован как отрезок времени между заданным исходным и любым другим узлом сети (графа ТЦУ), определяемый компонентами вектора временной развертки (ВВР) и исчисляемый условными единицами времени (у. е. в.).

Рассмотрим подробно модели и методы анализа и оптимизации процессов в АСУ, которые используются в рамках аналитико-оптимизационной процедуры и направлены на получение аналитических характеристик процессов управления АСУ.

*Анализ реализуемости детерминированных моделей.* Поскольку описание ТЦУ не зависит от типа комплекса управления, имеется начальное значение ВВР для графа ТЦУ  $G$ :

$$t = (t_1, t_2, \dots, t_n)',$$

где  $t_i$  соответствует времени задействования компонента структуры вычислительной системы для решения задачи находящейся в  $i$ -й вершине графа  $G$ , а ВВР полностью определяет информационное взаимодействие между структурными компонентами сети.

Граф ТЦУ  $G$ , построенный на этой структуре, описывается матрицей инцидентности  $A = a[i, j]$ . Это прямоугольная матрица размерности  $n \times m$ , в которой

$$a_{ji} = \begin{cases} 1, & \text{если } j\text{-я дуга выходит из вершины } i; \\ -1, & \text{если } j\text{-я дуга входит в вершину } i; \\ 0 & \text{— в остальных случаях.} \end{cases}$$

Также используется вектор реализации (ВР):

$$h = (h_1, h_2, \dots, h_m)',$$

где  $h_j$  — время выполнения задачи обработки информации и управления ТЦУ, находящейся в начале  $j$ -й дуги, которое задается структурой ВС.

Итак, при анализе реализуемости ТЦУ необходимо установить возможность реализации ВВР на ВС с заданной структурой при заданном векторе  $h$ . С этой целью сформулируем следующее утверждение. Для реализации ТЦУ на заданной структуре ВС с заданным ВВР  $t$  необходимо и достаточно выполнение следующего условия: если

из  $i$ -й вершины графа ТЦУ выходит  $j$ -я дуга, входящая в  $v$ -ю вершину, то разница  $t_v - t_i$  должна быть не меньше, чем время выполнения задачи в  $i$ -й вершине:

$$t_v - t_i \geq h_j a_{ij}. \quad (4.19)$$

Следовательно, благодаря матрице  $A$  и векторам  $t$  и  $h$  становится возможным определение многих характеристик процесса выполнения ТЦУ. Например, время реализации рассчитаем по формуле

$$T = \max_i (t_i + z_i) - \min_i t_i, \quad (4.20)$$

где  $z_i$  – время выполнения задачи, реализация которой начинается в момент  $t_i$ .

Отметим, что последовательность выполнения задач ТЦУ соответствует упорядочению координат ВВР по возрастанию. Если некоторые координаты равны, то это указывает на их параллельное выполнение.

Для случая, когда неравенство (4.19) не выполняется, предлагается следующий алгоритм коррекции ТЦУ, который позволяет производить замену компонентов ВВР на удовлетворительные. Чтобы этот корректирующий алгоритм имел смысл, необходимо выполнить три условия:

1) неотрицательности:

$$t_i \geq 0, \quad (i \in I), \quad (4.21)$$

где  $I = \{i\}$  – совокупность задач ТЦУ;

2) завершения предыдущих задач ТЦУ:

$$\sum_{t_j < t_i} z_j \leq t_i, \quad (i, j \in I); \quad (4.22)$$

3) логической последовательности:

$$t_i \geq \alpha_{ij} t_j, \quad (i, j \in I), \quad (4.23)$$

где  $\alpha_{ij}$  – коэффициент связности задач,

$$\alpha_{ij} = \begin{cases} 1, & \text{если задача } j \text{ предшествует задаче } i; \\ 0 & \text{– в противном случае.} \end{cases}$$

Здесь  $t_i$  и  $h_i$  – компоненты ВВР и ВР соответственно.

Для каждой задачи ТЦУ введем момент времени, до истечения которого ценность  $i$ -й задачи остается максимальной ( $t_i^f$ ) и момент времени, до истечения которого выполнение  $i$ -й задачи невозможно ( $t_i^s$ ). Данные параметры находятся в следующем соотношении:

$$t_i^s \leq t_i \leq t_i^f, \quad (4.24)$$

Очевидно, что

$$0 \leq t_i^s < t_i^f < T, \quad t_i^f - t_i^s \geq z_i,$$

где  $T$  – длительность реализации ТЦУ.

Если условия (4.21)–(4.23) выполняются, то выполняются и условия  $t_i^s < t_j^s, t_i^f < t_j^f$  для связанных задач  $i$  и  $j$ . С другой стороны, если ТЦУ существует, то имеют место зависимости

$$i < j \Rightarrow t_j^f \geq t_i^e; \quad t_j^b \geq t_i^s + z_i,$$

где  $t_j^b$  – момент начала  $j$ -й задачи;  $t_i^e$  – момент завершения  $i$ -й задачи.

Поэтому для каждой связанной пары  $i$  и  $j$  замена при коррекции ВВР «старых» значений  $t_i^{cr}$  на «новые»  $\max(t_i^s, t_j + z_i)$  и, соответственно,  $t_j^{cr}$  на  $\min(t_j^f, t_i + z_i)$  не влияет на факт допустимости существования ТЦУ. Таким образом, в результате работы алгоритма коррекции «старые» координаты ВВР меняются на «новые», удовлетворяющие следующим условиям:

- 1)  $i < j \Rightarrow t_i^{cr} \geq t_i^s, \quad t_j^{cr} \leq t_j^f;$
- 2)  $i < j \Rightarrow t_j^{cr} \geq t_i + z_i;$
- 3)  $i < j \Rightarrow t_i^{cr} \leq t_j - z_i.$

Алгоритм Дейкстры разработан для поиска кратчайшего пути (в том числе временного) между заданным исходным и любым другим узлом сети (графа). В процессе выполнения этого алгоритма при переходе от узла  $i$  к следующему узлу  $j$  используется специальная процедура пометки ребер. Обозначим через  $u_i$  кратчайшее расстояние от исходного узла 1 до узла  $i$ , а через  $d_{ij}$  – длину ребра  $(i, j)$ . Тогда для узла  $j$  метку  $[u_j, i]$  определим в виде

$$[u_j, i] = [u_i + d_{ij}, i], \quad d_{ij} \geq 0. \quad (4.25)$$

Метки узлов в алгоритме Дейкстры бывают двух типов: временные и постоянные. Одна временная метка в последствии может быть заменена на другую, если будет найден более короткий путь к данному узлу. Когда же станет очевидным, что более короткого пути не существует, статус временной метки изменится на постоянный.

Вычислительная схема алгоритма Дейкстры состоит из следующих этапов.

*Этап 0.* Исходному узлу (узел 1) присваивается постоянная метка  $[0, -]$ . Примем  $i = 1$ .

*Этап 1.* Вычисляются временные метки  $[u_i + d_{ij}, i]$  для всех узлов  $j$ , которые можно достичь непосредственно из узла  $i$  и которые не имеют постоянных меток. Если узел  $j$  уже имеет метку  $[u_j, k]$ , полученную от другого узла  $k$ , и если  $u_i + d_{ij} < u_j$ , тогда метка  $[u_j, k]$  заменяется на метку  $[u_i + d_{ij}, i]$ . Если все узлы имеют постоянные метки, процесс вычислений заканчивается. В противном случае выбирается метка  $[u_r, s]$  с наименьшим среди всех временных меток значением расстояния  $u_r$  (если таких меток несколько, то выбор произволен). Принимаем  $i = r$  и повторяем этап.

Опишем реализацию алгоритма Дейкстры. Для этого определим кратчайший временной путь графа ТЦУ, после корректировки ВВР методом анализа и коррекции детерминированной модели процесса (рис. 4.8).

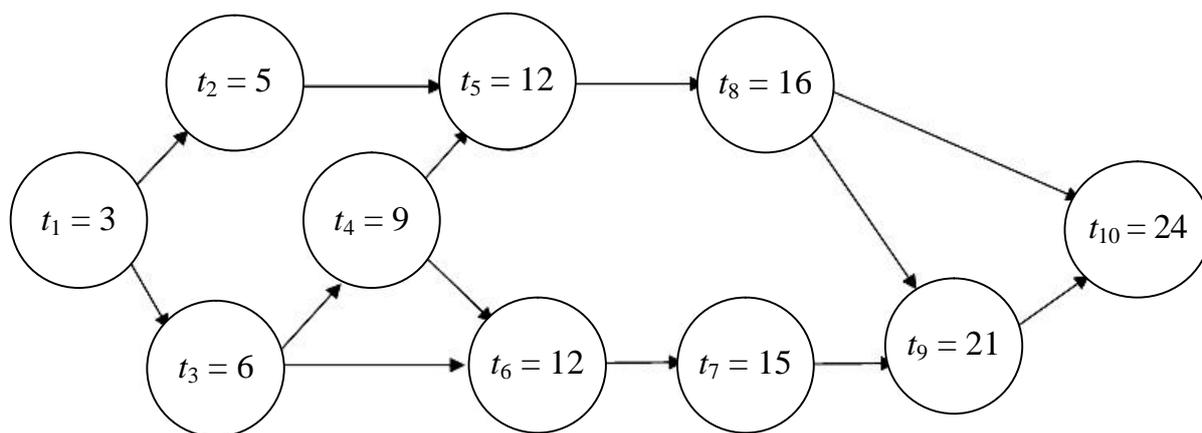


Рис. 4.8. Граф ТЦУ и откорректированные значения компонентов ВВР

Откорректированный ВВР  $\{t_i^{cr}\} = \{t_1 = 3; t_2 = 5; t_3 = 6; t_4 = 9; t_5 = 12; t_6 = 12; t_7 = 15; t_8 = 16; t_9 = 21; t_{10} = 24\}$  дает представление о времени выполнения задачи, реализация которой начинается

в момент  $t_i$ . Таким образом, временной отрезок между вершинами, выраженный в условных единицах времени, есть разность  $t_j - t_i$ .

Определим временные отрезки между узлами каждой пары взаимосвязанных вершин графа ТЦУ и представим их в виде таблицы (табл. 4.2).

Таблица 4.2

**Временные отрезки между узлами каждой пары взаимосвязанных вершин графа ТЦУ**

Номера взаимосвязанных вершин	Отрезок времени между взаимосвязанными вершинами, у.е.в.
$t_2 - t_1$	2
$t_3 - t_1$	3
$t_5 - t_2$	7
...	...
$t_{10} - t_8$	8
$t_{10} - t_9$	3

Нахождение кратчайшего временного пути от узла 1 графа ТЦУ до всех остальных узлов состоит из следующих этапов.

*Этап 0.* Назначим узлу 1 постоянную метку  $[0, -]$ .

*Этап 1.* Из узла 1 можно достичь узлов 2 и 3. Вычислим метки для этих узлов и сведем их в таблицу (табл. 4.3). Среди узлов 2 и 3 узел 2 имеет наименьшее значение ( $u_2 = 2$ ), поэтому статус метки этого узла изменяется на «постоянная».

Таблица 4.3

**Вычислительная схема алгоритма этапа 1**

Узел	Метка	Статус метки
1	$[0, -]$	Постоянная
2	$[0 + 2, 1] = [2, 1]$	Временная
3	$[0 + 3, 1] = [3, 1]$	Временная

*Этап 2.* Из узла 2 можно достичь узла 5. Вычислим метку для этого узла и также сведем результаты в таблицу (табл. 4.4). Поскольку не все узлы имеют постоянные метки, то для дальнейшей реализации алгоритма выбираем метку с наименьшим временным путем ( $u_3 = 3$ ), при этом статус метки узла 3 изменяется на «постоянная».

Таблица 4.4

## Вычислительная схема алгоритма этапа 2

Узел	Метка	Статус метки
1	[0, –]	Постоянная
2	[2, 1]	Постоянная
3	$[0 + 3, 1] = [3, 1]$	Временная
5	$[2 + 7, 2] = [9, 2]$	Временная

Этапы 3–8 описывают вычислительную схему алгоритма с последовательным присвоением статуса метки «постоянная» всем узлам сети. Приводить их мы не будем. Рассмотрим лишь завершающие этапы.

Этап 9. Из узла 9 можно достичь узла 10. Вычислим метку для этого узла, в результате получим таблицу меток (табл. 4.5). На 9-м этапе узел 10 получает две метки с одинаковым временным путем ( $u_{10} = 21$ ). Поскольку не все узлы имеют постоянные метки, то для завершения алгоритма выбираем метку с наименьшим временным путем, а именно единственную метку  $u_{10} = 21$ , при этом статус метки узла 10 изменяется на «постоянная».

Таблица 4.5

## Вычислительная схема алгоритма этапа 9

Узел	Метка	Статус метки
1	[0, –]	Постоянная
2	[2, 1]	Постоянная
3	[3, 1]	Постоянная
4	[6, 3]	Постоянная
5	[9, 2]	Постоянная
6	[9, 3]	Постоянная
7	[12, 6]	Постоянная
8	[13, 5]	Постоянная
9	[18, 7]	Постоянная
10	$[13 + 8, 8] = [21, 8]$ , $[18 + 3, 9] = [21, 9]$	Временная

*Этап 10.* Так как из узла 10 нельзя попасть ни в один другой узел, процесс вычислений заканчивается (табл. 4.6).

Таблица 4.6

**Вычислительная схема алгоритма этапа 10**

Узел	Метка	Статус метки
1	[0, –]	Постоянная
2	[2, 1]	Постоянная
3	[3, 1]	Постоянная
4	[6, 3]	Постоянная
5	[9, 2]	Постоянная
6	[9, 3]	Постоянная
7	[12, 6]	Постоянная
8	[13, 5]	Постоянная
9	[18, 7]	Постоянная
10	[21, 8]	Постоянная

В завершение опишем нахождение кратчайшего временного пути между узлом 1 и любым другим узлом графа ТЦУ. Этот путь определяется начиная с узла назначения путем прохождения узлов в обратном направлении с помощью информации, представленной в постоянных метках. Например, для определения кратчайшего временного пути между узлами 1 и 5 получаем следующую обратную последовательность узлов:

$$(5) \rightarrow [9, 2] \rightarrow [2, 1] \rightarrow (1)$$

Таким образом, имеем путь  $1 \rightarrow 2 \rightarrow 5$  общей продолжительностью  $t_p = 9$  у.е.в.

Аналитико-оптимизационная процедура, применимая к детерминированной модели в рамках данного пособия дополнена алгоритмической конструкцией. Вычислительная схема алгоритма Дейкстры показывает его реализуемость в детерминированных моделях (граф ТЦУ) и позволяет определить такие характеристики как кратчайший временной путь на любом участке информационной карты, что с точки зрения управления позволяет анализировать временные затраты маршрута управления и делать вывод о том, где, а также за счет чего достигнуто сокращение холостых временных «окон». Данный резуль-

тат значим, с точки зрения возможности применения методов сетевого анализа в решении задачи параметрической оптимизации детерминированных моделей технологических циклов автоматизированных систем управления.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Что называют GERT-сетевой моделью процессов в АСУ?
2. Какие существуют типы входных и выходных функций в GERT-модели?
3. Перечислите основные эквивалентные преобразования GERT-сети.
4. Охарактеризуйте свойства потоковых графов.
5. Приведите топологическое уравнение для замкнутых графов.

## Глава 5

# ОСНОВНЫЕ ЭТАПЫ ПРОЕКТИРОВАНИЯ И ВНЕДРЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

### 5.1. ПРОЦЕССЫ ЖИЗНЕННОГО ЦИКЛА АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

Совокупность взаимосвязанных процессов создания и последовательного изменения состояния АСУ от формирования исходных требований до окончания эксплуатации и утилизации комплекса средств автоматизации называется *жизненным циклом АСУ*. Рекомендуемые для описания жизненного цикла АСУ процессы делятся на основные, вспомогательные и организационные.

*Основные процессы* реализуются заказчиком, поставщиком, разработчиком или какой-то другой стороной, вовлеченной в деятельность. Всего их пять:

- процесс заказа: определяет работы заказчика – организации, которая приобретает систему;
- процесс поставки: определяет работы поставщика – организации, которая поставляет систему или ее часть;
- процесс разработки: определяет работы разработчика – организации, которая проектирует и разрабатывает систему или ее часть;
- процесс эксплуатации: определяет работы организации, которая обеспечивает эксплуатационное обслуживание системы в заданных условиях в интересах пользователей;
- процесс сопровождения: определяет работы персонала – организации, которая предоставляет услуги по сопровождению технических и программных средств, состоящих в контролируемом изменении с целью сохранения их исходного состояния и функциональных возможностей.

*Вспомогательный процесс* является целенаправленной составной частью другого процесса, обеспечивающей его успешную реализацию и качество выполнения. При необходимости он инициируется и используется другим процессом. Вспомогательными являются:

- процесс документирования: определяет работы по описанию сформированных требований, полученных результатов и т. п.;

- процесс управления конфигурацией: определяет работы по управлению конфигурацией технических и программных средств;
- процесс обеспечения качества: определяет работы по объективному обеспечению того, чтобы продукты проектирования и процессы соответствовали установленным для них требованиям и реализовывались в рамках утвержденных планов;
- процесс верификации: определяет работы заказчика, поставщика или независимой стороны по верификации разработанных продуктов и процессов, т. е. их соответствие предъявляемым требованиям, по мере реализации этапов проекта;
- процесс аттестации: определяет работы заказчика, поставщика или независимой стороны по окончательному утверждению соответствия продуктов проектирования предъявляемым к ним требованиям;
- процесс совместного анализа: определяет работы по оценке состояния и результатов какой-либо деятельности; может использоваться, когда одна из сторон проверяет другую;
- процесс аудита: определяет работы по установлению соответствия требованиям, планам и договору; может использоваться, когда одна из сторон контролирует продукты и работы другой;
- процесс решения проблемы: определяет работы по анализу и устранению проблем (включая несоответствия), которые были обнаружены во время разработки, эксплуатации или других процессов независимо от их характера и источника [44; 45].

*Организационные процессы* служат для создания и реализации в организации основной структуры, охватывающей взаимосвязанные процессы жизненного цикла и соответствующий персонал, а также для постоянного совершенствования данной структуры и процессов. К организационными относятся:

- процесс управления: определяет основные работы по управлению (в том числе проектом) при реализации процессов жизненного цикла;
- процесс создания инфраструктуры: определяет основные работы по созданию инфраструктуры процессов жизненного цикла;
- процесс усовершенствования: определяет основные работы, которые организация заказчика, поставщика, разработчика, персонала сопровождения или администратора другого процесса выполняет при создании, оценке, контроле и усовершенствовании выбранных процессов жизненного цикла;

– процесс обучения: определяет работы по соответствующему обучению персонала заказчика, поставщика, разработчика или кого-либо другого.

Следует отметить, что деление процессов жизненного цикла АСУ на основные и вспомогательные достаточно условное и в большей степени определяет взаимосвязи между ними, чем их важность для достижения целей проектирования. Так, на начальных этапах различных стадий проектирования усиливается роль организационных процессов и даже в основных процессах превалирует их организующая составляющая. Именно поэтому при планировании работ важно применение процессов всех вышеназванных типов для полного задействования их механизмов управления качеством.

## **5.2. ЭТАПЫ ПРОЕКТИРОВАНИЯ И ВНЕДРЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ**

Работы по созданию АСУ рекомендуется планировать как последовательность стадий и этапов, необходимых и достаточных для достижения поставленных целей. В зависимости от сложности объекта автоматизации и набора задач, требующих решения при создании конкретной АСУ, стадии и этапы работ могут иметь различную трудоемкость. Допускается объединять последовательные этапы и даже исключать некоторые из них на любой стадии проекта. Можно также начинать выполнение работ следующей стадии до окончания предыдущей. Стадии и этапы создания АСУ, выполняемые организациями-участниками, прописываются в договорах и технических заданиях на выполнение работ.

Выделяют следующие стадии проектирования и внедрения АСУ:

### **1. Формирование требований к АСУ:**

– обследование объекта и обоснование необходимости создания АСУ;

– формирование требований пользователей к АСУ;

– оформление отчета о выполненной работе и тактико-технического задания на разработку.

### **2. Разработка концепции АСУ:**

– изучение объекта автоматизации;

– проведение необходимых научно-исследовательских работ;

– разработка вариантов концепции АСУ, удовлетворяющих требованиям пользователей;

– оформление отчета и утверждение концепции.

3. Техническое задание (разработка и утверждение технического задания на создание АСУ).

4. Эскизный проект:

– разработка предварительных проектных решений по системе и ее частям;

– разработка эскизной документации на АСУ и ее части.

5. Технический проект:

– разработка проектных решений по системе и ее частям;

– разработка документации на АСУ и ее части;

– разработка и оформление документации на поставку комплектующих изделий;

– разработка заданий на проектирование в смежных частях проекта.

6. Рабочая документация:

– разработка рабочей документации на АСУ и ее части;

– разработка и адаптация программ.

7. Ввод в действие:

– подготовка объекта автоматизации;

– подготовка персонала;

– комплектация АСУ поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями);

– строительно-монтажные работы;

– пусконаладочные работы;

– проведение предварительных испытаний;

– проведение опытной эксплуатации;

– проведение приемочных испытаний.

8. Сопровождение АСУ:

– выполнение работ в соответствии с гарантийными обязательствами;

– послегарантийное обслуживание.

Рассмотрим эти стадии подробнее.

**Формирование требований к АСУ.** Особенностью данной стадии работ являются, как правило, нечетко сформулированные и недостаточно формализованные пожелания к будущей автоматизированной системе. Поэтому особенно важным и ответственным для принятия решения о возможности и целесообразности создания конкретной АСУ при имеющихся ресурсах и приемлемых сроках разработки, а также для планировании всей последующей деятельности становится

этап обследования объекта автоматизации. А при проектировании автоматизированной системы для достаточно узкой сферы деятельности рекомендуется даже проводить комплексное обследование объекта. При этом желательно:

- обновить или подтвердить данные о структуре предприятия;
- выяснить политику высшего руководства по тактике и стратегии развития предприятия;
- составить карту или паспорт технических и программных средств предприятия, в том числе эксплуатирующихся АСУ;
- описать и провести структурный анализ существующих бизнес-процессов, документооборота и информационных потоков.

Эти сведения будут необходимы как при разработке концепции АСУ, так и в начале проектирования – при предварительной оценке стоимости работ, решении вопросов интеграции новых и существующих программных средств.

Особое внимание следует уделить структурному анализу деятельности предприятия. В качестве инструмента исследования возможно применение методологий стандартов IDEF0, IDEF3, DFD. Безусловно, использование именно этих стандартов не является обязательным и единственно возможным вариантом – для графического представления бизнес-процессов и информационных потоков можно воспользоваться любыми другими средствами, не обязательно специализированными. Тем не менее в пользу указанных средств говорит то, что они оформлены как стандарты и, следовательно, понятны многим группам разработчиков; поддерживаются CASE-средствами (BPWin и др.), в которых решены те вопросы, которые будут периодически возникать при использовании подручных средств; в них наработан ряд формальных признаков для оценки качества описанных с их помощью бизнес-процессов. И главное – с их помощью можно качественно и количественно оценить влияние автоматизации на деятельность предприятия и возможность достижения с созданием АСУ поставленных целей (модели AS IS и TO BE). При выборе разумной степени детализации процессов такая работа не занимает много времени.

На этапе формирования требований пользователей к АСУ рекомендуется определить группы и отдельных сотрудников, для которых создаваемая автоматизированная система представляется как набор пересекающихся или непересекающихся функциональностей. В качестве инструмента для этого этапа можно применить диаграммы вариантов использования (Use Case Diagram) унифицированного языка моделирования UML (Unified Modeling Language). Язык UML является

простым и мощным средством моделирования, который может быть эффективно использован для построения концептуальных, логических и графических моделей сложных систем самого различного целевого назначения.

Визуальное моделирование в UML представляется как некоторый процесс поуровневого спуска от наиболее общей и абстрактной концептуальной модели исходной системы к логической, а затем и к физической модели соответствующей программной системы. Для достижения этих целей вначале строится модель в форме диаграммы вариантов использования, которая описывает функциональное назначение системы или то, что система будет делать в процессе своего функционирования. Данная диаграмма является исходным концептуальным представлением или концептуальной моделью системы в процессе ее проектирования и разработки. Разработка диаграммы вариантов использования преследует следующие цели:

- определение общих границ и контекста моделируемой предметной области на начальных этапах проектирования системы;
- формулирование общих требований к функциональному поведению проектируемой системы;
- разработка исходной концептуальной модели системы для ее последующей детализации в форме логических и физических моделей;
- подготовка исходной документации для взаимодействия разработчиков системы с ее заказчиками и пользователями.

В завершение этапа формирования требований к АСУ формируется отчет о выполненной работе, состоящий из следующих разделов:

- характеристика объекта и результатов его функционирования;
- описание существующей информационной системы и ее недостатков;
- обоснование необходимости совершенствования информационной системы объекта;
- цели, критерии и ограничения создания АСУ;
- функции и задачи создаваемой АСУ;
- ожидаемые технико-экономические результаты создания АСУ;
- выводы и предложения.

В разделе *«Характеристика объекта и результатов его функционирования»* описывают тенденции развития, требования к объему, номенклатуре и качеству результатов функционирования, характер взаимодействия объекта с внешней средой, а также определяют существующие показатели функционирования и тенденции их изменения во времени.

Раздел «*Описание существующей информационной системы и ее недостатков*» содержит описание функциональной и информационной структуры системы, качественных и количественных характеристик, раскрывающих взаимодействие ее компонентов. При описании недостатков системы приводят результаты оценки качества функционирования и организационно-технологического уровня системы, выявляют недостатки в организации и технологии функционирования информационных процессов и определяют степень их влияния на качество функционирования системы.

В разделе «*Обоснование необходимости совершенствования информационной системы объекта*» оценивают степень соответствия прогнозируемых показателей функционирования объекта предъявляемым требованиям и выявляют необходимость совершенствования информационной системы путем создания АСУ.

Раздел «*Цели, ограничения и критерии создания АСУ*» содержит:

- формулировку производственно-хозяйственных, научно-технических и экономических целей и критериев создания АСУ;
- характеристику ограничений по созданию АСУ.

В раздел «*Функции и задачи создаваемой АСУ*» включают:

- обоснование выбора перечня автоматизированных функций и комплексов задач с указанием очередности внедрения;
- требования к характеристикам реализации функций и задач в соответствии с действующими документами, определяющими общие технические требования к АСУ конкретного вида;
- дополнительные требования к АСУ в целом и ее частям, учитывающие специфику создаваемой системы.

Раздел «*Ожидаемые технико-экономические результаты создания АСУ*» содержит:

- перечень основных источников экономической эффективности, получаемых в результате создания АСУ (в том числе экономию производственных ресурсов, улучшение качества продукции, повышение производительности труда и т.д.) и оценку ожидаемых изменений основных технико-экономических и социальных показателей (например, показателей по номенклатуре и объемам производства, себестоимости продукции, рентабельности, уровня социального развития);
- оценку ожидаемых затрат на создание и эксплуатацию АСУ с распределением их по очередям ввода и календарным срокам;
- ожидаемые обобщающие показатели экономической эффективности АСУ.

Раздел «*Выводы и предложения*» рекомендуется разбивать на следующие подразделы:

- выводы о производственно-хозяйственной необходимости и технико-экономической целесообразности создания АСУ;
- предложения по совершенствованию организации и технологии процесса деятельности;
- рекомендации по созданию АСУ.

Первый подраздел должен содержать сопоставление ожидаемых результатов создания АСУ с заданными целями и критериями по целевым показателям и нормативным требованиям, а также принципиальное решение вопроса о создании АСУ – положительное или отрицательное.

Во второй подраздел включают предложения по совершенствованию производственно-хозяйственной деятельности, организационной и функциональной структур системы, методов деятельности и видов обеспечения АСУ.

Третий подраздел содержит рекомендации:

- по виду создаваемой АСУ, ее совместимости с другими АСУ и неавтоматизируемой частью соответствующей системы;
- организационной и функциональной структуре создаваемой АСУ;
- составу и характеристикам подсистем и видов обеспечения АСУ;
- организации использования имеющихся и приобретению дополнительных технических средств;
- рациональной организации разработки и внедрения АСУ;
- определению основных и дополнительных, внешних и внутренних источников и видов объемов финансирования и материального обеспечения разработок;
- обеспечению производственных условий создания АСУ;
- любые другие рекомендации.

В случае принятия положительного решения о создании АСУ оформляется заявка на разработку (тактико-техническое задание). Она составляется в произвольной форме и содержит предложения организации-пользователя к организации-разработчику на проведение работ по созданию АСУ, а также требования к системе, сведения об имеющихся ограничениях или условиях и ресурсах на создание АСУ.

**Разработка концепции АСУ.** Исходными данными для разработки концепции создаваемой АСУ является утвержденный отчет предыдущей стадии и тактико-техническое задание.

На этапах изучения объекта автоматизации и проведения научно-исследовательских работ организация-разработчик детально изучает объект автоматизации и необходимые научно-исследовательские работы, связанные с поиском путей и возможностью реализации требований пользователя. Здесь особенно важно организовать процессы обучения разработчиков, донести до них особенности автоматизируемых бизнес-процессов и их взаимодействия, соподчиненности и совместной работы подразделений, участвующих в бизнес-процессах, их ролей, ответственности и полномочий.

Как правило, работы данных этапов включают:

- более детальное описание и анализ бизнес-процессов объекта автоматизации;
- оценку влияния территориального размещения частей объекта автоматизации на возможные изменения информационных потоков в системе;
- прогнозирование возможных изменений объекта и его характеристик при реализации утвержденной стратегии развития предприятия;
- знакомство с техническими решениями и опытом создания подобных систем на предприятиях партнеров и конкурентов;
- изучение достижений в интересующих областях информационных технологий;
- изучение достижений и тенденций в теории управления;
- изучение и оценку рынка предлагаемых готовых решений;
- разработку прототипов системы или ее частей для оценки ее реализуемости и получения качественных и количественных характеристик вариантов исследования;
- другие работы, которые вытекают из целей и задач этапов.

Если научно-исследовательские работы выполняются по отдельным договорам, то для них разрабатываются отдельные технические задания, а результаты оформляются соответствующими отчетами.

Далее разрабатываются прототип и варианты концепции АСУ. В результате разработки прототипа появляется программное приложение, которое обладает требуемой функциональностью и имеет реальный пользовательский интерфейс, что, безусловно, повышает достоверность оценочных характеристик создаваемой АСУ и качество работы в целом. Здесь следует отметить, что с появлением инструментариев быстрой разработки приложений (Rapid Application Development, RAD), таких как C++ Builder, Visual C++, Delphi и другие, время и затраты на разработку прототипов проекта существенно сократились.

При разработке вариантов концепции АСУ желательно рассмотреть как можно больше альтернативных технических решений, удовлетворяющих функциональным требованиям к системе и налагаемым ограничениям по времени создания и имеющимся ресурсам. Это могут быть, например, следующие варианты:

- принятие существующей информационной системы в качестве базовой с модернизацией части технических и программных средств;

- покупка готовых модулей аналогичной автоматизированной системы, их доработка и интеграция с существующими аппаратно-программными комплексами;

- полная замена существующей АСУ покупной или вновь разработанной;

- варианты централизованной и децентрализованной системы;

- варианты, базирующиеся на различных информационных технологиях;

- варианты, базирующиеся на различных аппаратных или программных платформах;

- любые другие варианты, вытекающие из существующего положения вещей, требований и возможностей.

Несмотря на кажущееся обилие рассматриваемых вариантов, обычно их количество, с учетом имеющихся ограничений, не превышает двух-трех.

Завершающим этапом данной стадии проектирования АСУ является оформление отчета и утверждение концепции.

В основной части отчета о выполненной работе приводят:

- описание результатов изучения объекта автоматизации;

- описание и оценку преимуществ и недостатков разработанных альтернативных вариантов концепции создания АСУ;

- сопоставительный анализ требований пользователя к АСУ и вариантов концепции на предмет удовлетворения этих требований;

- обоснование выбора оптимального варианта концепции и описание предлагаемой АСУ;

- ожидаемые результаты и эффективность реализации выбранного варианта концепции АСУ;

- ориентировочный план реализации выбранного варианта концепции АСУ;

- необходимые затраты ресурсов на разработку, ввод в действие и обеспечение функционирования;

- требования, гарантирующие качество АСУ;
- условия приемки системы.

**Техническое задание.** Техническое задание (ТЗ) на АСУ является основным документом, определяющим требования и порядок создания, развития или модернизации автоматизированной системы, в соответствии с которым проводится ее разработка, ввод в действие и приемка.

ТЗ разрабатывают на систему в целом, предназначенную для работы самостоятельно или в составе другой системы. Основными исходными данными для разработки являются тактико-техническое задание и утвержденный вариант концепции АСУ. Однако задаваемые в ТЗ требования не должны ограничивать разработчика системы в поиске и реализации наиболее эффективных технических, технико-экономических и других решений.

ТЗ является текстовым документом. Необходимые чертежи, схемы и большие по объему таблицы рекомендуется оформлять в виде приложений. В зависимости от вида, назначения и специфических особенностей объекта автоматизации и условий функционирования системы допускается оформлять разделы ТЗ в виде приложений, вводить дополнительные, исключать или объединять подразделы. Дополнительно могут быть разработаны ТЗ на части АСУ – так называемые, частные ТЗ, в которые не включают разделы, дублирующие содержание разделов ТЗ на АСУ в целом.

ТЗ содержит следующие разделы, которые, в свою очередь, могут быть разбиты на подразделы:

- общие сведения;
- назначение и цели создания (развития) системы;
- характеристика объекта автоматизации;
- требования к системе;
- состав и содержание работ по созданию системы;
- порядок контроля и приемки системы;
- требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие;
- требования к документированию;
- источники разработки.

В разделе «*Общие сведения*» указывают полное наименование системы и ее условное обозначение, названия предприятий разработчика и заказчика, плановые сроки начала и окончания работ, сведения об источниках и порядке финансирования, порядок оформления и предъявления заказчику результатов работ по созданию системы и ее частей.

Раздел *«Назначение и цели создания системы»* содержит сведения о виде автоматизируемой деятельности (управление, проектирование и т. п.) и перечне объектов автоматизации, на которой предполагается использовать АСУ. Здесь же приводят наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта автоматизации, которые должны быть достигнуты в результате создания АСУ, и указывают критерии оценки достижения целей.

В разделе *«Характеристика объекта автоматизации»* приводят краткие сведения об объекте или ссылки на документы, содержащие такую информацию, а также сведения об условиях эксплуатации объекта и характеристиках окружающей среды.

Раздел *«Требования к системе»* состоит из следующих подразделов:

- требования к системе в целом;
- требования к функциям (задачам), выполняемым системой;
- требования к видам обеспечения.

В подразделе *«Требования к системе в целом»* указывают:

- требования к структуре и функционированию системы (перечень подсистем, их назначение и основные характеристики, а также требования к числу уровней иерархии и степени централизации системы; требования к способам и средствам связи для информационного обмена между компонентами системы; требования к характеристикам взаимосвязей системы со смежными системами и к ее совместимости; требования к режимам функционирования системы, ее диагностированию, перспективам развития и модернизации);

- требования к численности и квалификации персонала системы, режиму его работы, порядку подготовки и контролю знаний и навыков;

- показатели назначения (значения параметров, характеризующих степень соответствия системы ее назначению); для АСУ указывают степень приспособляемости системы к изменению процессов и методов управления, к отклонениям параметров объекта управления, а также допустимые пределы модернизации и развития системы, вероятностно-временные характеристики, при которых сохраняется целевое назначение системы);

- требования к надежности (состав и количественные значения показателей надежности для системы в целом и ее частей; перечень аварийных ситуаций, по которым должны быть регламентированы требования к надежности и значения соответствующих показателей);

требования к надежности технических средств и программного обеспечения; требования к методам оценки и контроля показателей надежности на разных стадиях создания системы);

- требования к безопасности (требования по обеспечению безопасности при монтаже, наладке, эксплуатации, обслуживании и ремонте технических средств (защита от воздействия электрического тока, электромагнитных полей, акустических шумов и т. п.), по допустимым уровням освещенности, вибрационных и шумовых нагрузок);

- требования к эргономике и технической эстетике (показатели АСУ, задающие необходимое качество взаимодействия человека с машиной и комфортность условий работы персонала);

- требования к транспортабельности для подвижных АСУ (конструктивные требования, обеспечивающие транспортабельность технических средств системы, а также требования к транспортным средствам);

- требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы (условия и регламент эксплуатации, которые должны обеспечивать использование технических средств системы с заданными показателями, в том числе виды и периодичность обслуживания; предварительные требования к допустимым площадям для размещения персонала и технических средств, к параметрам сетей энергоснабжения и т. п.; требования по количеству, квалификации обслуживающего персонала и режимам его работы; требования к составу, размещению и условиям хранения комплекта запасных изделий и приборов; требования к регламенту обслуживания);

- требования к защите информации от несанкционированного доступа;

- требования по сохранности информации при авариях (приводят перечень событий: аварий, отказов технических средств (в том числе потери питания) и т. п., при которых должна быть обеспечена сохранность информации в системе);

- требования к защите от влияния внешних воздействий (радиоэлектронной защите средств АСУ, защите по стойкости, устойчивости и прочности к внешним воздействиям среды применения);

- требования по стандартизации и унификации (показатели, устанавливающие требуемую степень использования стандартных унифицированных методов реализации функций и задач системы, предоставляемых программных средств, типовых математических методов, моделей и проектных решений, унифицированных форм управленческих

документов, общесоюзных классификаторов технико-экономической информации и классификаторов категорий по области их применения; требования к использованию типовых автоматизированных рабочих мест, компонентов и комплексов);

– дополнительные требования по усмотрению разработчика или заказчика системы.

В подразделе «Требования к функциям (задачам), выполняемым системой» приводят:

– перечень функций, задач или их комплексов (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации (по каждой подсистеме);

– при создании системы в две или более очередей – перечень функциональных подсистем, отдельных функций или задач, вводимых в действие в первой и последующих очередях;

– временной регламент реализации каждой функции, задачи или комплекса задач;

– требования к качеству реализации каждой функции, форме представления выходной информации, характеристикам необходимой точности и времени выполнения; требования одновременности выполнения группы функций, достоверности выдаваемых результатов;

– перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

В подразделе «Требования к видам обеспечения» в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

Для математического обеспечения системы приводят требования к составу, области применения и способам использования математических методов и моделей, типовых алгоритмов и алгоритмов, подлежащих разработке.

Для информационного обеспечения указывают требования:

– к составу, структуре и способам организации данных в системе;

– информационному обмену между компонентами системы;

– информационной совместимости со смежными системами;

– использованию международных, республиканских, отраслевых классификаторов, унифицированных документов и классификаторов, действующих на данном предприятии;

- применению систем управления базами данных;
- структуре процесса сбора, обработки, передачи и представлению данных в системе;
- защите данных от разрушений при авариях и сбоях электропитания;
- контролю, хранению, обновлению и восстановлению данных;
- процедуре придания юридической силы документам, создаваемым техническими средствами АСУ.

Для лингвистического обеспечения системы приводят требования к применению языков программирования, взаимодействия пользователей и технических средств, а также к кодированию и декодированию данных, языкам ввода–вывода данных и манипулирования данными, средствам описания предметной области (объекта автоматизации), способам организации диалога.

Для программного обеспечения системы приводят перечень покупных программных средств, а также требования к независимости программных средств от используемых технических средств и операционной среды, качеству программных средств и способам его обеспечения и контроля.

Для технического обеспечения системы указываются требования:

- к видам технических средств, в том числе комплексов технических средств, программно-технических комплексов и других комплектующих изделий, допустимых к использованию;
- функциональным, конструктивным и эксплуатационным характеристикам.

Для организационного обеспечения приводят требования:

- к структуре и функциям подразделений, участвующих в функционировании системы или обеспечивающих эксплуатацию;
- организации функционирования системы и порядку взаимодействия персонала АСУ и персонала объекта автоматизации;
- защите от ошибочных действий персонала системы.

Раздел «*Состав и содержание работ по созданию системы*» должен содержать перечень стадий и этапов работ по созданию системы, сроки их выполнения, перечень организаций-исполнителей и ссылки на документы, подтверждающие их согласие на участие в создании системы, или запись, определяющую ответственного (заказчик или разработчик) за проведение этих работ. В этом разделе также приводят:

- перечень документов, предъявляемых по окончании соответствующих стадий и этапов работ;

- вид и порядок проведения экспертизы технической документации (стадию, этап, объем проверяемой документации, организацию-эксперта);

- программу работ, направленных на обеспечение требуемого уровня надежности разрабатываемой системы (при необходимости);

- перечень работ по метрологическому обеспечению на всех стадиях создания системы с указанием их сроков и организаций-исполнителей (при необходимости).

В разделе *«Порядок контроля и приемки системы»* указывают:

- виды, состав, объем и методы испытаний системы и ее составных частей;

- общие требования к приемке работ по стадиям (перечень участвующих предприятий и организаций, место и сроки проведения), порядок согласования и утверждения приемочной документации;

- статус приемочной комиссии (государственная, межведомственная, ведомственная).

В разделе *«Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие»* необходимо привести перечень основных мероприятий и их исполнителей, которые следует выполнить при подготовке объекта автоматизации к вводу АСУ в действие. В этот перечень включают:

- приведение поступающей в систему информации (в соответствии с требованиями к информационному и лингвистическому обеспечению) к виду, пригодному для обработки с помощью ЭВМ;

- изменения, которые необходимо осуществить в объекте автоматизации, например, изменения применяемых методов управления;

- создание условий функционирования объекта автоматизации, при которых гарантируется соответствие создаваемой системы требованиям ТЗ;

- создание необходимых для функционирования системы подразделений и служб;

- сроки и порядок комплектования штатов и обучения персонала.

В разделе *«Требования к документированию»* приводят:

- согласованный разработчиком и заказчиком системы перечень подлежащих разработке комплектов и видов документов;

- требования по документированию комплектующих элементов;

- при отсутствии государственных стандартов, определяющих требования к документированию элементов системы, дополнительно включают требования к составу и содержанию таких документов.

В разделе «*Источники разработки*» должны быть перечислены документы и информационные материалы (технико-экономическое обоснование, отчеты о законченных научно-исследовательских работах, информационные материалы на отечественные и зарубежные системы-аналоги и др.), на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

В состав ТЗ на АСУ при наличии утвержденных методик также включают приложения, содержащие расчет ожидаемой эффективности и оценку научно-технического уровня системы.

Проект ТЗ на АСУ разрабатывает организация-разработчик системы с участием заказчика на основании технических требований (заявки, тактико-технического задания и т. п.). При конкурсной организации работ варианты проекта ТЗ рассматриваются заказчиком, который либо выбирает предпочтительный вариант, либо на основании сопоставительного анализа подготавливает с участием будущего разработчика АСУ окончательный вариант ТЗ.

Срок согласования ТЗ в каждой организации не должен превышать 15 дней со дня его получения. Замечания по проекту представляются с техническим обоснованием. Решения по замечаниям должны быть приняты разработчиком проекта и заказчиком до утверждения ТЗ. Если при согласовании проекта возникли разногласия между разработчиком и заказчиком (или другими заинтересованными организациями), то составляется протокол разногласий (форма произвольная) и конкретное решение принимается в установленном порядке. Необходимость согласования проекта ТЗ на АСУ с органами государственного надзора и другими заинтересованными организациями определяют совместно заказчик системы и разработчик проекта ТЗ.

Согласование проекта ТЗ на АСУ разрешается оформлять отдельным документом (письмом). В этом случае под грифом «Согласовано» делают ссылку на этот документ. Утверждение ТЗ на АСУ осуществляют руководители предприятий разработчика и заказчика системы. Согласование и утверждение дополнений к ТЗ проводят в установленном порядке.

**Эскизный проект.** Выполнение стадии эскизного проектирования не является строго обязательным. Если основные проектные решения определены ранее или достаточно очевидны для конкретной АСУ и объекта автоматизации, то эта стадия может быть исключена из общей последовательности работ.

Содержание эскизного проекта на систему приводится в ТЗ. Как правило, на этапе разработки предварительных проектных решений по системе и ее частям определяются:

- функции АСУ;
- функции подсистем, их цели и ожидаемый эффект от внедрения;
- состав комплексов задач и отдельных задач;
- концепция информационной базы и ее укрупненная структура;
- функции системы управления базой данных;
- состав вычислительной системы и других технических средств;
- функции и параметры основных программных средств.

По результатам проделанной работы оформляется, согласовывается и утверждается документация в объеме, необходимом для описания полной совокупности принятых проектных решений и достаточном для дальнейшего выполнения работ по созданию АСУ.

**Технический проект.** На начальном этапе создания технического проекта проводится разработка общих решений по автоматизированной системе, ее частям, функционально-алгоритмической структуре, функциям персонала и организационной структуре, структуре и составу технических средств, алгоритмам решения задач и применяемым языкам, организации и ведению информационной базы, системе классификации и кодирования информации, программному обеспечению. По результатам работы оформляется, согласовывается и утверждается соответствующая документация.

На следующем этапе проводят разработку документации на поставку серийно выпускаемых изделий для комплектования АСУ, а также определяют технические требования и составляют ТЗ на разработку изделий, не изготовляемых серийно.

На стадии технического проекта также осуществляют разработку, оформление, согласование и утверждение заданий на проектирование в смежных частях проекта для проведения, если это необходимо, строительных, электротехнических, санитарно-технических и других подготовительных работ на объекте автоматизации.

**Рабочая документация.** На этой стадии создания АСУ осуществляют разработку рабочей документации, содержащей все необходимые и достаточные сведения для обеспечения выполнения работ по вводу АСУ в действие и ее эксплуатации, а также для поддержания уровня эксплуатационных характеристик (качества) системы. Разработанная документация должна быть соответствующим образом оформлена, согласована и утверждена.

Также на этой стадии осуществляется разработка или адаптация программ создаваемой АСУ, что является одним из наиболее важных этапов проекта. Если на ранних стадиях проектирования применялись диаграммы языка UML, то здесь они могут быть использованы как исходные данные для дальнейшей разработки.

Традиционно этап разработки программных средств является наиболее закрытым, менее понятным и трудно отслеживаемым заказчиком. С 1 марта 2012 г. вступил в действие ГОСТ Р ИСО/МЭК 12207–2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств», который определяет основные процессы создания программных средств для заказчика и исполнителя и является некоторым путеводителем в этой области. Планирование жизненного цикла программных средств в соответствии с рекомендациями этого стандарта гарантирует, что заказчику и исполнителю удастся избежать ошибок, приводящих к появлению продуктов, не соответствующих требованиям технического задания.

**Ввод в действие.** Стадия ввода АСУ в действие подразумевает передачу созданной системы в рабочую эксплуатацию.

На этапе «*Подготовка объекта автоматизации*» проводят следующие организационные подготовительные работы:

- реализацию проектных решений по организационной структуре АСУ;
- обеспечение подразделений объекта управления инструктивно-методическими материалами;
- внедрение классификаторов информации.

Этап «*Подготовка персонала*» включает в себя проведение мероприятий по обучению персонала и проверке его способности обеспечить функционирование АСУ.

На этапе «*Комплектация АСУ поставляемыми изделиями*» обеспечивают получение комплектующих изделий серийного и единичного производства, материалов и монтажных изделий, а также проводят входной контроль их качества.

Этап «*Строительно-монтажные работы*» включает в себя:

- выполнение работ по строительству специализированных зданий (помещений) для размещения технических средств и персонала АСУ;
- сооружение кабельных каналов;
- выполнение работ по монтажу технических средств и линий связи;

- испытание смонтированных технических средств;
- сдачу технических средств для проведения пусконаладочных работ.

На этапе *«Пусконаладочные работы»* проводят автономную наладку технических и программных средств, загрузку информации в базу данных и проверку системы ее ведения, после чего производят комплексную наладку всех средств системы.

После пусконаладочных работ проводят испытания АСУ, представляющие собой процесс проверки выполнения заданных функций АСУ, определения и проверки соответствия требованиям ТЗ ее количественных и (или) качественных характеристик, выявления и устранения недостатков в действиях системы, в разработанной документации. Для АСУ устанавливают следующие основные виды испытаний: предварительные, опытная эксплуатация и приемочные. При необходимости допускается дополнительно проведение других видов испытаний АСУ и ее частей.

В зависимости от взаимосвязей частей АСУ и объекта автоматизации испытания могут быть автономные или комплексные. Автономные испытания охватывают части АСУ. Их проводят по мере готовности частей системы к сдаче в опытную эксплуатацию. Комплексные испытания организуют для групп взаимосвязанных частей системы или для АСУ в целом.

Для планирования проведения всех видов испытаний разрабатывают документ *«Программа и методика испытаний»*, который устанавливает необходимый и достаточный объем испытаний, обеспечивающий заданную достоверность получаемых результатов. Данный документ может разрабатываться либо на АСУ в целом, либо на отдельные ее части АСУ. В качестве приложения в него могут включаться тесты или контрольные примеры. Разработчик документа устанавливается в договоре или ТЗ.

Испытания АСУ следует проводить на объекте заказчика. По согласованию между заказчиком и разработчиком предварительные испытания и приемку программных средств АСУ допускается осуществлять на технических средствах разработчика при создании условий получения достоверных результатов испытаний. Допускается последовательное проведение испытаний и сдача частей АСУ в опытную и постоянную эксплуатацию при установленной в ТЗ очередности ввода системы в действие.

На этапе «Проведение предварительных испытаний» определяется работоспособность АСУ и решается вопрос о возможности ее приемки в опытную эксплуатацию. Предварительные испытания следует выполнять после проведения разработчиком отладки и тестирования поставляемых программных и технических средств системы и представления соответствующих документов об их готовности к испытаниям, а также после ознакомления персонала АСУ с эксплуатационной документацией.

Как и другие виды испытаний, предварительные испытания могут быть автономными и комплексными.

Автономные испытания следует проводить в соответствии с программой и методикой автономных испытаний, разрабатываемых для каждой части АСУ, в которой указывают:

- перечень функций, подлежащих испытаниям;
- описание взаимосвязей объекта испытаний с другими частями АСУ;
- условия, порядок и методы проведения испытаний и обработки результатов;
- критерии приемки частей по результатам испытаний.

К программе автономных испытаний следует прилагать график их проведения.

Подготовленные и согласованные тесты (контрольные примеры) на этапе автономных испытаний должны обеспечить:

- полную проверку функций и процедур по перечню, согласованному с заказчиком;
- необходимую точность вычислений, установленную в ТЗ;
- проверку основных временных характеристик функционирования программных средств (в тех случаях, когда это существенно);
- проверку надежности и устойчивости функционирования программных и технических средств.

В качестве исходной информации для теста рекомендуется использовать фрагмент реальной информации организации-заказчика в объеме, достаточном для обеспечения необходимой достоверности испытаний.

В случае если проведенные автономные испытания будут признаны недостаточными, либо будет выявлено нарушение требований регламентирующих документов по составу или содержанию документации, указанная часть АСУ может быть возвращена на доработку и назначен новый срок испытаний.

Результаты автономных испытаний частей АСУ следует фиксировать в протоколах испытаний. Он должен содержать заключение о возможности (невозможности) допуска части АСУ к комплексным испытаниям.

*Комплексные испытания* проводят путем выполнения комплексных тестов. Комплексный тест должен соответствовать следующим требованиям:

- быть логически увязанным;
- обеспечивать проверку выполнения функций частей АСУ во всех режимах функционирования, установленных в ТЗ, в том числе всех связей между ними;
- обеспечивать проверку реакции системы на некорректную информацию и аварийные ситуации.

В программе комплексных испытаний АСУ или ее частей указывают:

- перечень объектов испытания;
- состав предъявляемой документации;
- описание проверяемых взаимосвязей между объектами испытаний;
- очередность испытаний частей АСУ;
- порядок и методы испытаний, в том числе состав программных средств и оборудования, необходимых для проведения испытаний, включая специальные стенды и полигоны.

Для проведения комплексных испытаний должны быть предоставлены:

- программа комплексных испытаний;
- заключение по автономным испытаниям соответствующих частей АСУ и устранение ошибок и замечаний, выявленных при автономных испытаниях;
- комплексные тесты;
- программные и технические средства и соответствующая им эксплуатационная документация.

При комплексных испытаниях допускается использовать в качестве исходной информацию, полученную на автономных испытаниях частей АСУ.

Если в ходе проведения комплексных испытаний были обнаружены какие-либо недостатки, проводится их устранение, после чего проводят повторные комплексные испытания в необходимом объеме.

Результаты испытаний отражают в протоколе, который содержит заключение о возможности (невозможности) приемки АСУ

в опытную эксплуатацию, а также перечень необходимых доработок и рекомендуемые сроки их выполнения. Работу завершают оформлением акта приемки в опытную эксплуатацию.

Целью этапа «Проведение опытной эксплуатации» является определение фактических значений количественных и качественных характеристик системы и готовности персонала к работе в условиях ее функционирования, а также фактической эффективности документации и, при необходимости, ее корректировка.

Опытную эксплуатацию проводят в соответствии с программой, в которой указывают:

- условия и порядок функционирования АСУ в целом и отдельных ее частей;
- продолжительность опытной эксплуатации, достаточную для проверки правильности функционирования АСУ при выполнении каждой функции и готовности персонала к работе в условиях функционирования системы;
- порядок устранения недостатков, выявленных в процессе опытной эксплуатации.

Во время опытной эксплуатации АСУ ведут рабочий журнал, в который заносят сведения о продолжительности функционирования АСУ, отказах, сбоях, аварийных ситуациях, изменениях параметров объекта автоматизации, проводимых корректировках документации и программных средств, наладке технических средств. Сведения фиксируют в журнале с указанием даты и ответственного лица. Также в него могут быть занесены замечания персонала по удобству эксплуатации АСУ.

По результатам опытной эксплуатации принимают решение о возможности (или невозможности) предъявления частей АСУ и системы в целом на приемочные испытания. Работа завершается оформлением акта о завершении опытной эксплуатации и допуске системы к приемочным испытаниям.

На этапе «Проведение приемочных испытаний» определяют соответствие АСУ техническому заданию, оценивают качество опытной эксплуатации и решают вопрос о возможности приемки системы в постоянную эксплуатацию. Этому этапу должна предшествовать опытная эксплуатация АСУ на объекте.

Приемочные испытания проводят в соответствии с программой, в которой указывают:

- перечень объектов, выделенных в системе для испытаний, и требований, которым должны соответствовать объекты (со ссылкой на пункты ТЗ);

- критерии приемки системы и ее частей;
- условия и сроки проведения испытаний;
- средства для проведения испытаний;
- фамилии лиц, ответственных за проведение испытаний;
- методику испытаний и обработки их результатов;
- перечень оформляемой документации.

Для проведения приемочных испытаний должна быть предъявлена следующая документация:

- техническое задание на создание АСУ;
- акт приемки в опытную эксплуатацию;
- рабочие журналы опытной эксплуатации;
- акт завершения опытной эксплуатации и допуска АСУ к приемочным испытаниям;
- программа и методика испытаний.

В ходе приемочных испытаний АСУ проверяют:

- качество выполнения комплексом программных и технических средств автоматических функций во всех режимах функционирования системы согласно ТЗ на создание АСУ;
- знание персоналом эксплуатационной документации и наличие у него навыков, необходимых для выполнения установленных функций во всех режимах функционирования системы;
- полноту содержащихся в эксплуатационной документации указаний персоналу по выполнению им функций во всех режимах работы АСУ согласно ТЗ;
- количественные и качественные характеристики выполнения автоматических и автоматизированных функций АСУ в соответствии с ТЗ;
- другие свойства АСУ, которым она должна соответствовать по ТЗ.

Приемочные испытания следует проводить на функционирующем объекте. В первую очередь, они должны включать проверку:

- полноты и качества реализации функций при штатных, предельных, критических значениях параметров объекта автоматизации и в других условиях функционирования АСУ, указанных в ТЗ;
- выполнения каждого требования, относящегося к интерфейсу системы;
- работы персонала в диалоговом режиме;
- средств и методов восстановления работоспособности АСУ после отказов;
- комплектности и качества эксплуатационной документации.

Проверку полноты и качества выполнения функций АСУ рекомендуется проводить в два этапа. На первом этапе испытывают отдельные функции (задачи, комплексы задач). При этом проверяют выполнение требований ТЗ к функциям (задачам, комплексам задач). На втором этапе оценивают взаимодействие задач в системе и выполнение требований ТЗ к системе в целом. По согласованию с заказчиком проверка задач в зависимости от их специфики может проводиться автономно или в составе комплекса. Объединение задач при проверке в комплексах целесообразно проводить с учетом общности используемой информации и внутренних связей.

Проверку работы персонала в диалоговом режиме проводят с учетом полноты и качества выполнения функций системы в целом. Проверке подлежит:

- полнота сообщений, директив, запросов, доступных оператору и их достаточность для эксплуатации системы;
- сложность процедур диалога, возможность работы персонала без специальной подготовки;
- реакция системы и ее частей на ошибки оператора, средства сервиса.

Проверка средств восстановления работоспособности АСУ после отказов компьютеров должна включать:

- проверку наличия в эксплуатационной документации рекомендаций по восстановлению работоспособности и полноту их описания;
- практическую выполнимость рекомендованных процедур;
- работоспособность средств автоматического восстановления функций (при их наличии).

Проверку комплектности и качества эксплуатационной документации следует проводить путем анализа документации на соответствие требованиям нормативно-технических документов ТЗ.

Результаты испытаний объектов, предусмотренных программой, фиксируют в протоколах, содержащих следующие разделы:

- назначение испытаний и номер раздела требований ТЗ на АСУ, по которой проводят испытание;
- состав технических и программных средств, используемых при испытаниях;
- указание методик, в соответствии с которыми проводились испытания, обработка и оценка результатов;
- условия проведения испытаний и характеристики исходных данных;

- средства хранения и условия доступа к конечной, тестирующей программе;
- обобщенные результаты испытаний;
- выводы о результатах испытаний и соответствии созданной системы или ее частей определенному разделу требований ТЗ на АСУ.

Протоколы испытаний объектов по всей программе обобщают в едином протоколе, на основании которого делают заключение о соответствии системы требованиям ТЗ на АСУ и возможности оформления акта ее приемки в постоянную эксплуатацию. Работу завершают оформлением акта о приемке АСУ в постоянную эксплуатацию.

**Сопровождение АСУ.** Сопровождение АСУ ведется в два этапа.

На этапе *«Выполнение работ в соответствии с гарантийными обязательствами»* осуществляются работы по устранению недостатков, выявленных при эксплуатации АСУ в течение установленных гарантийных сроков, внесению необходимых изменений в документацию.

На этапе *«Послегарантийное обслуживание»* выполняют работы:

- по анализу функционирования системы;
- выявлению отклонений фактических эксплуатационных характеристик АСУ от проектных значений;
- установлению причин этих отклонений;
- устранению выявленных недостатков и обеспечению стабильности эксплуатационных характеристик АСУ;
- внесению необходимых изменений в документацию на АСУ.

Основные этапы проектирования и внедрения автоматизированных систем управления формируют стадии жизненного цикла АСУ, сопровождаемые программным, техническим, организационным, лингвистическим, математическим и прочими видами обеспечения. Также стоит отметить, что могут быть реализованы дополнительные стадии жизненного цикла АСУ, такие как этапы маркетинговых исследований, эксплуатации и утилизации.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Назовите стадии и этапы создания АСУ.
2. Как формируются требования к АСУ?
3. Какова структура технического задания на АСУ?
4. Что включают в себя эскизный и технический проекты?
5. Охарактеризуйте этапы автономного и комплексного испытания АСУ.

## Глава 6

# МЕТОДЫ АНАЛИЗА И РАСЧЕТА ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

### 6.1. СИСТЕМНЫЙ АНАЛИЗ ЭФФЕКТИВНОСТИ АВТОМАТИЗАЦИИ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

Современные условия функционирования производства характеризуются отказом от взгляда на производство как на объект, существующий сам по себе, что связано с изменениями, происходящими как во всем мире (интенсивной интеграцией экономических систем, появлением относительно большого рынка предложений при ограниченном рынке сбыта, возникновением социальных проблем, обусловленных автоматизацией производства, и глобальных экономических проблем), так и в экономике нашей страны.

Переход науки и промышленности на хозяйственный расчет, появление тенденции к самостоятельности развития предприятий и зарождению рынка сбыта уже не могут удовлетворяться определением эффективности систем автоматизации в терминах производительности, себестоимости и рентабельности только самого производства. На современного производителя продуктов и услуг (товаров) воздействует окружающая его среда, включающая целый комплекс (сумму) факторов, который состоит из чрезвычайно большого числа составляющих, а потому полностью совпасть по направлению с суммой факторов, определяющих интересы производителя, может только случайно. Столь же случайна и полная противоположность этих факторов. Однако в силу того, что действия производителя и реакция окружающей среды довольно сложно взаимосвязаны, попытки удовлетворить интересы производителя любыми средствами, в том числе и автоматизацией технологических процессов без учета действующих на него внешних факторов – не лучшая стратегия поведения.

При оценке целесообразности автоматизации технологических процессов приходится ориентироваться на некоторый согласованный оптимум, предполагающий наименьшее расхождение суммарных интересов производителя и окружающей его среды. Факторы окружающей среды могут иметь достаточно большие постоянные времени, поэтому говоря о последствиях автоматизации, следует рассматривать и временные срезы, позволяющие оценить ее отдаленные последствия. К сожалению, в настоящее время нет методики комплексного

анализа эффективности автоматизации производства, поэтому проблему системного анализа эффективности автоматизации технологических процессов приходится рассматривать в основном на понятийном уровне. К тому же не все факторы удастся измерить в одних единицах (например, в денежных), что дает дополнительные основания к выполнению анализа на уровне понятий и экспертных оценок [46; 47].

Факторы окружающей производителя среды, рассматриваемые в связи с автоматизацией технологических процессов, достаточно сложны и разнообразны. Они могут как способствовать внедрению систем автоматизации технологических процессов, так и противодействовать им (табл. 6.1).

Таблица 6.1

**Факторы среды, окружающей производителя**

Фактор	Направленные действия*	
	Благоприятное	Противодействующее
Рынок производителей	Возможность заимствовать технические, организационные и прочие решения	Конкуренция
Рынок потребителей	Возможность удовлетворять и расширять спрос на товар	Желание повысить собственную прибыль (в том числе за счет снижения цены на товар)
Рынок труда	Возможность оперативно набирать специалистов требуемых профилей	Противодействие сокращению численности занятых работников
Рынок капитала	Возможность получать дополнительный доход за счет новых инвестиций	Сокращение роста капиталовложений (сокращение капитала в обороте)
Социальная среда	Свобода конкуренции	Ограничения законодательства на монополию, прибыль и т. п.
Эргономическая среда	Климатические, природные ресурсы и т. п.	Ограничения на загрязнение окружающей среды, продукты и т. п. (связанные с ним дополнительные затраты)
Объективные законы природы и организации	Возможность максимально-полезного использования законов природы в условиях автоматизации, возможность работы в критических режимах с максимальным выходом продукта	Дополнительные затраты на исследования, выход технологических процессов на критические режимы при отказе систем управления

\* Относительно достижения максимальной суммы интересов производителя.

Благоприятные и противодействующие факторы взаимно дополняют друг друга. Так, отсутствие конкуренции может рассматриваться как благоприятный фактор, но тогда исчезнет и возможность заимствования передовых проектных решений, а это превращает указанный фактор в противодействующий. Достаточно широкий рынок труда можно считать благоприятным фактором, но если на нем наблюдается дефицит специалистов по проектированию и эксплуатации автоматизированных технологических процессов, то это уже противодействующий фактор. В случае, если названные факторы можно вычислить или дать им экспертную оценку, то они на правах факторов доходов и затрат входят в расчет экономической эффективности от внедрения АСУ ТП, рассматриваемый в методике «затраты–прибыль». В противном случае они расцениваются как некоторые ограничения на области допустимых проектных решений по АСУ ТП.

Рассмотрим основные факторы и источники прибыли в системе «первичный производитель – промежуточный потребитель – конечный пользователь». Определим, что первичный производитель – это элемент системы, производящей на базе собственных ресурсов некоторые материалы, комплектующие изделия или полуфабрикаты, которые не может потреблять конечный пользователь. Указание на собственные ресурсы позволяет разорвать достаточно длинные цепочки связей последовательно чередующихся элементов «первичный производитель – промежуточный потребитель» в реальных производственно-экономических системах, а то, что продукцию производителя не может непосредственно потреблять пользователь, свидетельствует, что первичный производитель не может непосредственно получить прибыль от конечного пользователя (она может быть только перераспределена между первичным производителем и промежуточным потребителем). Теперь ясно, что промежуточный потребитель может быть (если между ним и производителем нет соглашений о перераспределении прибыли) обладателем прибыли от продажи продукции (товаров) конечному пользователю. В свою очередь, конечный потребитель, если он получает прибыль от использования товара и не имеет соглашения с промежуточным потребителем, не получает прибыль от внедрения АСУ ТП, и, следовательно, расчеты экономической эффективности от внедрения АСУ ТП необходимо откорректировать по этому фактору.

Реально прибыль, получаемая в сфере конечного потребления, между элементами системы «производитель – промежуточный потре-

битель – конечный пользователь» может быть перераспределена посредством соглашения о ценах в рассматриваемой системе или выплат каких-то вознаграждений и т. п. Первичный производитель и промежуточный потребитель могут полностью получить только ту часть прибыли, которая образуется за счет роста производительности труда, экономии ресурсов и других факторов, действие которых экономически замыкается в нем.

С другой стороны, конечный пользователь один терпит убытки от полученного товара, если нет каких-либо соглашений о распределении их компенсации в рассматриваемой системе. Поэтому, как правило, действуют соглашения о компенсации убытков, и в общем случае производитель обязан возместить часть потерь. В связи с этим в расчетах экономической эффективности АСУ ТП учет потерь от рекламаций и прочих аналогичных претензий может не соответствовать реальным потерям конечного пользователя, что завышает расчетную экономическую эффективность от внедрения АСУ ТП [45; 48].

Внедрение АСУ ТП направлено на повышение эффективности производства за счет повышения производительности труда, увеличения объема и улучшения качества выпускаемой продукции, использования основных фондов, материалов и сырья, снижения объема незавершенного производства и страховых запасов, уменьшения числа работающих на предприятии (табл. 6.2).

Таблица 6.2

**Основные источники экономической эффективности АСУ ТП**

Класс АСУ ТП	Источник эффективности
Автоматизированные системы логико-программного управления без оптимизации	Повышение производительности труда при выполнении технологической операции; резкое сокращение ошибок оперативного персонала и брака по причине ошибок; стабилизация технологического процесса; сокращение числа работающих при мультипрограммном управлении
Автоматизированные системы оптимального управления технологическим процессом (АСУ ОТП), автоматизированные системы комплексного управления	Повышение качества и надежности выпускаемых изделий (продукта); увеличение объема выпуска продукции; оптимизация номенклатурного распределения выпускаемых изделий; сокращение потерь рабочего времени на участках и технологических линиях; повышение оперативности управления производственным процессом со стороны оперативного персонала; повышение качества управления технологическим процессом

Внедрение систем управления отличается от обычных работ по внедрению новой техники тем, что позволяет перевести производственный процесс на качественно новую ступень развития, характеризуемую более высокой, по сравнению с предшествующей, ступенью организацией (упорядоченностью). Качественное улучшение организации производства обусловлено значительным ростом объема обрабатываемой в системе управления информации, резким увеличением скорости ее обработки и применением для выработки управляющих решений более сложных методов и алгоритмов, чем те, что использовались до внедрения АСУ ТП.

В соответствии с теоретическими основами оценки экономической эффективности систем управления [48–50] экономический эффект, получаемый от внедрения одной и той же системы, зависит от уровня организованности производства (стабильности и настроенности технологического процесса) до и после внедрения АСУ ТП, т. е. может быть различным для разных производств.

Значение основных источников экономической эффективности применительно к различным технологическим процессам и классам систем управления можно проиллюстрировать примерами из отечественной и зарубежной практики применения АСУ ТП. Так, благодаря внедрению системы управления нажимным устройством блюминга 1150 за счет повышения производительности обжимного стана производство получило годовой экономический эффект свыше 120 тыс. руб. Внедрение адаптивной системы управления прокатным станом на трубoproкатном заводе обеспечило снижение затрат в производстве более чем на 225 тыс. руб. за счет сокращения разброса размеров труб. Внедрение системы управления участком формовки точечных диодов на полупроводниковом заводе дало возможность сократить на несколько десятков человек число работников, занятых на данной технологической операции, и тем самым получить ежегодный экономический эффект более 100 тыс. руб. Сокращение производственного персонала при внедрении на приборостроительном заводе электронного контролера электрических счетчиков АСУ ТП «Аргус» дало экономический эффект более 200 тыс. руб.

Большую экономическую эффективность от внедрения АСУ ТП можно показать также на примере ряда зарубежных фирм целлюлозно-бумажной промышленности [13]. Внедрение системы управления бумагоделательными машинами позволило уменьшить среднеквадратичное отклонение массы  $1 \text{ м}^2$  бумаги и ее влажности в 3–5 раз.

При внедрении АСУ ТП благодаря 5%-процентному увеличению скорости машин производительностью 200 т высокосортной бумаги в день получен дополнительный валовой доход около 500 тыс. долл. в год, а машин производительностью 500 т упаковочного картона в день – 450 тыс. долл. в год. Время переналадки технологического процесса с выпуска одного типа бумаги на выпуск другого при внедрении АСУ ТП сокращено на 20 %, скорость производства бумаги увеличена на 15 %, выпуск продукции – на 19 %, а эффективность бумагоделательной машины – на 2 %. В результате внедрения АСУ ТП на процессе отбеливания изменение белизны бумаги уменьшилось на 50 %, вязкости – на 47 %, остаточной щелочи – на 43 %, остаточного хлора – на 73 %; затраты на химикаты снизились на 14,8 %, потребление пара – на 11,6 %. Затраты на разработку и внедрение системы управления процессом отбеливания окупились за 8 месяцев эксплуатации системы.

На другом предприятии затраты на внедрение управляющей ЭВМ окупились за год благодаря увеличению выпуска продукции, снижению брака, уменьшению стоимости отбеливания и стабилизации качества. При внедрении системы управления технологическими процессами при производстве крафт-картона годовой доход увеличился на 500 тыс. долл. за счет улучшения управления массой 1 м<sup>2</sup> картона и уменьшения его влажности.

В ряде случаев внедрение АСУ ТП позволило существенно сократить капитальные вложения, связанные с необходимостью расширить производство или часто его перестраивать с выпуска одного типа продукции на выпуск другого. Так, в производстве целлюлозы требуется относительно часто менять сорт [13; 51]. Чтобы удовлетворить потребности заказчика, в этом случае нужно было бы ввести в строй дополнительный целлюлозный завод с отбеливающей установкой, строительство которого обошлось бы примерно в 50 млн долл. Применение же управляющей ЭВМ для оперативной перенастройки технологического процесса позволило обойтись имеющимися производственными мощностями.

Следует отметить, что помимо явных источников экономической эффективности внедрение АСУ ТП в производство дает возможность использовать и скрытые его ресурсы за счет получения достоверной оперативной информации, повышения качества управления производством и надежности системы управления, упрощения контроля за состоянием производства и ряд других.

При разработке АСУ ТП невозможно предварительно оценить выгоду, получаемую от реализации скрытых ресурсов, но по результатам эксплуатации систем можно сделать вывод, что на их долю может приходиться до 50 % всего дохода, связанного с внедрением АСУ ТП.

Основные затраты на создание системы управления состоят, как правило, из затрат на предпроектные и проектные работы  $S_{\text{п}}$  и затрат на приобретение специального оборудования, устанавливаемого в системе управления,  $S_{\text{об}}$ . При этом в стоимость проектных работ включаются все расходы, связанные с разработкой проекта (в том числе разработкой математического обеспечения и внедрением системы управления), в стоимость оборудования – стоимость средств управляющей вычислительной техники, устройств подготовки, передачи и отображения информации, а также стоимость тех узлов технологического оборудования, модернизация или разработка которых вызвана условиями работы оборудования в системе «технологический процесс – АСУ ТП» (табл. 6.3).

Таблица 6.3

**Среднее распределение затрат при создании АСУ ТП**

Статья затрат	Затраты, %
Технические средства:	
– ЭВМ;	20
– внешняя память;	15
– УВВИ:	
аналоговые;	7
цифровые;	3
стандартные	5
Разработка и внедрение:	
– программирование;	25
– руководство проектом, разработка и ввод в эксплуатацию	25

Кроме затрат на создание системы управления, предприятие несет затраты и на ее эксплуатацию. Полагая, что затраты на создание и эксплуатацию АСУ ТП равномерно окупаются по годам ее функционирования, можно определить ориентировочные годовые затраты на систему управления:

$$Z = (S_{\text{п}} + S_{\text{об}}) / T + S_{\text{экс}}, \quad (6.1)$$

где  $T$  – время эксплуатации;  $S_{\text{экс}}$  – годовые эксплуатационные затраты.

Для систем управления технологическими процессами время эксплуатации принимается обычно равным 5–7 годам. При этом если

в процессе работы в систему были внесены изменения, приведшие к изменению  $S_{\text{п}}$ ,  $S_{\text{экс}}$  или  $S_{\text{об}}$ , то годовые затраты на эксплуатацию системы управления необходимо пересчитать.

Эксплуатационные затраты на систему управления можно определить по формуле

$$S_{\text{экс}} = S'_{\text{з.п}} + S_{\text{а.ф}} + S_{\text{э}} + S_{\text{м}}, \quad (6.2)$$

где  $S'_{\text{з.п}}$  – годовой фонд заработной платы персонала, обслуживающего систему управления;  $S_{\text{а.ф}}$  – амортизационные отчисления и плата за фонды;  $S_{\text{э}}$  – затраты на электроэнергию, воду и т. п.;  $S_{\text{м}}$  – годовые затраты на материалы и комплектующие изделия.

Амортизационные отчисления и плата за фонды определяются по выражению

$$S_{\text{а.ф}} = \sum_{i=1}^n S_{\text{оби}} (a_{\text{аи}} + a_{\text{ф}}),$$

где  $S_{\text{оби}}$  – стоимость  $i$ -го типа оборудования;  $a_{\text{аи}}$  – амортизационные отчисления по  $i$ -му типу оборудования;  $a_{\text{ф}}$  – отчисления за фонды.

Годовой фонд заработной платы персонала, обслуживающего систему управления рассчитывается следующим образом:

$$S'_{\text{з.п}} = t_{\text{р}} \bar{S} (1 + k_{\text{ц.н}}) m', \quad (6.3)$$

где  $t_{\text{р}}$  – время работы обслуживающего персонала за год;  $S'_{\text{з.п}}$  – средняя годовая ставка обслуживающего персонала (с учетом различных начислений);  $k_{\text{ц.н}}$  – коэффициент цеховых накладных расходов;  $m'$  – численность обслуживающего систему персонала, в том числе персонала, обслуживающего специализированные устройства технологического оборудования.

Системный анализ эффективности автоматизации технологических процессов предполагает комплексную оценку по ряду показателей, в том числе с учетом таких акторов как рынки, социальная и эргономическая среда, конкуренция. При этом основными критериями оценки эффективности АСУ ТП остаются, те, что влияют на выходные параметры технологического процесса и связанную с этим себестоимость конечной продукции – максимально эффективное использование материалов и энергоносителей, показывающее уровень брака продукции, и количество трудозатрат, отражающее степень автоматизации.

## 6.2. МЕТОДИКА ДЕТЕРМИНИРОВАННОГО РАСЧЕТА ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ

Увеличение объема производства при внедрении системы управления связано с увеличением темпа прироста объема выпускаемой продукции по отношению к объему при отсутствии системы управления. Объем выпускаемой продукции при внедрении системы управления можно определить по выражению

$$V' = V_{\text{в}} + B_{\text{п}} + B'_{\text{п}}, \quad (6.4)$$

где  $V_{\text{в}}$  – объем выпущенной продукции за прошедший год до внедрения системы управления;  $B_{\text{п}}$  – планируемый прирост объема выпускаемой продукции при отсутствии системы управления;  $B'_{\text{п}}$  – дополнительное (сверхплановое) увеличение объема выпускаемой продукции при внедрении системы управления. Величину  $B'_{\text{п}}$  определяют, как правило, в процессе опытной эксплуатации системы управления; если же необходимо провести предварительные (до опытного внедрения) экономические расчеты, то используют метод аналогий.

Внедрение системы управления позволяет улучшить качество продукции и увеличить долю (процент) изделий высших сорта или классификационных групп, имеющих большую потребительскую стоимость и цену, а следовательно, дающих производству большую прибыль. Прибыль, которую получает предприятие за счет выпуска изделий высших сорта или классификационных групп с учетом увеличения объема производства при внедрении системы управления, можно рассчитать по формуле

$$P' = \sum_{i=1}^n B'_i P'_i - \sum_{i=1}^n (B_{\text{ви}} + B_{\text{пи}}) P_i, \quad (6.5)$$

где  $B'_i$  – объем выпускаемой продукции  $i$ -й группы после внедрения системы управления;  $B_{\text{ви}}$  – объем выпущенной продукции  $i$ -й группы за прошедший год до внедрения системы управления;  $B_{\text{пи}}$  – планируемый прирост объема выпускаемой продукции  $i$ -й группы при отсутствии системы управления.

Прибыль, полученную на единицу продукции без системы управления  $P_i$  и с ней  $P'_i$ , вычисляют следующим образом:

$$P_i = C_{io} - S_i,$$

$$P'_i = C'_{io} - S'_i,$$

где  $C_{io}$  и  $C'_{io}$  – цена единицы продукции при отсутствии системы управления и с ней соответственно;  $S_i$  и  $S'_i$  – себестоимость единицы продукции при отсутствии системы управления и с ней соответственно.

Если внедрение системы управления ведет к сокращению числа основных и вспомогательных рабочих, а также численности административно-управленческого персонала, то заработная плата с цеховыми накладными расходами в производстве уменьшится на величину

$$S'_{с.з.п} = t_{p.c} S_{с.з.п} (1 + k_{ц.н}) m_c, \quad (6.6)$$

где  $t_{p.c}$  – время работы персонала, подлежащего сокращению, за прошедший год;  $S_{с.з.п}$  – средняя часовая ставка сокращенного персонала (с учетом начислений);  $m_c$  – численность сокращенного персонала.

Тогда снижение удельных трудовых затрат от внедрения системы управления можно определить по формуле

$$\Delta S'_{з.п} = (1 + k_{з.н}) \left[ S_{з.п} / (B_{vi} + B_{pi}) - (S'_{з.п} - S_{с.з.п}) / B'_п \right], \quad (6.7)$$

в которой  $k_{з.н}$  – коэффициент общезаводских накладных расходов;  $S_{з.п}$  – фонд заработной платы с общезаводскими накладными расходами до внедрения системы управления.

Снижение удельного расхода затрат на материалы и комплектующие изделия рассчитывают по выражению

$$\Delta C'_{к.м} = C_{к.м} / (B_v + B_p) - C'_{к.м} / B', \quad (6.8)$$

где  $C_{к.м}$  и  $C'_{к.м}$  – стоимость комплектующих изделий и материалов на планируемый год при отсутствии системы управления и с ней соответственно.

С учетом затрат на создание и эксплуатацию системы управления получаем снижение себестоимости  $i$ -го типа изделий:

$$\Delta S'_i = S_i - \Delta S'_{з.п} - \Delta C'_{к.м} + Z / B. \quad (6.9)$$

При определении экономической эффективности АСУ ТП необходимо учитывать потери в производстве, которые могут возникнуть в связи с выходом из строя системы управления. Полагая,

что при отказе системы управления предприятие несет убытки  $D_{ар}$  только от невыполнения плана реализации, получим:

$$D_{пр} = \sum_{i=1}^n B_{прi} P'_i (t'_{пр} / t_{пр}),$$

где  $B_{прi}$  – объем продукции, недовыпущенной из-за простоя системы управления;  $t'_{пр}$  – время простоя системы управления в течение года во время работы технологического оборудования;  $t_{пр}$  – плановое время работы технологического оборудования в году.

Прибыль от внедрения системы управления с учетом убытков от ее простоев находим по формуле

$$P'_{пр} = \sum_{i=1}^n B'_i P'_i (1 - t'_{пр} / t_{пр}) - \sum_{i=1}^n (B_{Ви} + B_{Пи}) P_i. \quad (6.10)$$

Из формул (6.2) и (6.10) получаем срок окупаемости затрат на создание и функционирование системы управления (без учета уровня общей организованности производства или технологического процесса):

$$T_{ок} = (S_{п} + S_{об}) / (P'_{пр} - S_{экс}). \quad (6.11)$$

Результаты, полученные при детерминированном расчете экономической эффективности, справедливы лишь при условии абсолютной упорядоченности производства, т. е. его абсолютной стабильности. Ясно, что абсолютно стабильных технологических процессов в реальном производстве нет, поскольку они подвержены воздействию многих случайных факторов, изменение которых вызвано изменением свойств как объектов производства, так и организационных процессов управления им. Нестабильность производства всегда приводит к уменьшению предварительного расчетного (в предположении абсолютной стабильности производства) экономического эффекта, получаемого от внедрения системы управления.

В качестве обобщенного критерия эффективности работы системы управления, который учитывает нестабильность производства, принимают неупорядоченность системы  $r_n$  [49]. При этом за основу оценки  $r_n$  берут отклонение контролируемой переменной  $y(t)$  от ее оптимального значения  $y_{опт}(t)$ . Действительно, в случае идеальной

системы управления результатом ее действия в каждый момент времени было бы такое значение  $y_{\text{опт}}(t)$ , которое для системы является оптимальным, т. е. при любом  $t$  выполнялось бы условие  $y(t) = y_{\text{опт}}(t)$ . Из-за несовершенства системы управления  $y(t)$  отклоняется от  $y_{\text{опт}}(t)$ ; эти отклонения и характеризуют степень неупорядоченности системы управления в целом. Если для технологических процессов оптимальное значение  $y_{\text{опт}}(t)$  нельзя вычислить исходя из теоретических предположений, в расчетах используют значение  $y_{\text{опт}}(t)$ , определяемое на базе экспериментальных данных.

Неупорядоченность  $r_n$  связана с объемом используемой и перерабатываемой в системе управления информации  $I$  следующим соотношением:

$$dr_n = -\alpha r_n dI, \quad (6.12)$$

где  $\alpha$  – коэффициент пропорциональности.

Знак «минус» в формуле (6.12) объясняется тем, что увеличение объема обрабатываемой информации ведет к уменьшению неупорядоченности процесса. При больших  $r_n$  (т. е. при несовершенной системе управления) даже небольшой объем дополнительной (по отношению к ранее используемой) информации о состоянии объекта позволяет заметно улучшить качество управления им, тогда как при малых  $r_n$  (т. е. высоком качестве системы управления) для дальнейшего улучшения работы системы необходимо получить большой объем дополнительной информации.

Интегрировав уравнения (6.11), (6.12), получим выражение

$$r_n = r_{n\text{max}} e^{-\alpha I}, \quad (6.13)$$

в котором  $r_{n\text{max}}$  – максимальная неупорядоченность системы при  $I = 0$  ( $r_{n\text{max}} = 1, 0$ ).

Если известно промежуточное значение  $r_n^{(0)}$   $[0 \leq r_n^{(0)} \leq r_{\text{max}}]$  для некоторого соответствующего ему управления, характеризуемого объемом обрабатываемой информации  $I = I_0$ , то коэффициент  $\alpha$  определяется из соотношения

$$\alpha = \frac{1}{I_0} \ln \frac{r_{n\text{max}}}{r_n^{(0)}}.$$

Тогда

$$r_n = r_0 \exp \left[ - (I / I_0) \ln (r_{n \max} / r_n^0) \right]. \quad (6.14)$$

Так как  $\ln (r_{n \max} / r_n^0) > 0$ , то из формулы (6.14) следует, что  $r_n$  убывает по экспоненциальному закону с ростом объема обрабатываемой в системе управления информации [49].

Оптимальному действию системы управления объектом с неупорядоченностью соответствует выполнение не только условия

$$y(t) - y_{\text{опт}}(t) = 0, \quad (6.15)$$

но и условия

$$dy(t) - dy_{\text{опт}}(t) = 0 \quad (6.16)$$

при любом моменте времени  $t$ .

Невыполнение системой управления условий (6.15) и (6.16) ведет в производстве к потерям двух видов. Потери первого вида связаны с отклонением  $y(t)$  от  $y_{\text{опт}}(t)$ , они определяются по формуле

$$A = \int_0^I \varphi(t, y_{\text{опт}}, y) / y_{\text{опт}}(t) - y(t) dt,$$

где  $\varphi(t, y_{\text{опт}}, y)$  – весовая функция отклонения  $y(t)$  от  $y_{\text{опт}}(t)$  для разных моментов времени.

Потери второго вида вызваны тем, что если в некоторый момент времени  $t$  произошло отклонение  $y(t)$  от  $y_{\text{опт}}(t)$ , то для минимизации потерь первого вида система управления за возможно малый промежуток времени  $dt$  должна компенсировать это отклонение, но тогда  $dy(t)$  не будет уже равно  $dy_{\text{опт}}(t)$ . Потери второго вида определяются функционалом

$$B = \int_0^T \psi(t_1, y_{\text{опт}}, y) (dy_{\text{опт}} - dy) dt, \quad (6.17)$$

где  $\psi(t_1, y_{\text{опт}}, y)$  – весовая функция.

В формуле (6.17) выражена цена отклонения  $dy(t)$  от  $dy_{\text{опт}}(t)$ , которой характеризуется несовершенство системы (качество управле-

ния). Учет потерь второго вида приводит к следующей оценке влияния неупорядоченности:

$$r'_n = \frac{\int_0^T \Psi(t_1, y_{\text{опт}}, y)(dy_{\text{опт}} - dy)}{y_{\text{опт}}(T) + y(T)}, \quad (6.18)$$

т. е.  $r'_n$  – нормированное значение  $B$ .

Поскольку из математической модели системы (6.18) следует, что неупорядоченность системы есть случайная величина, и за нее можно принять математическое ожидание величины  $r'_n - r = M[r'_n]$ , выражение (6.18) принимает следующий вид:

– для непрерывных технологических процессов:

$$r = M \frac{\int_0^T \Phi[t, y_{\text{опт}}(t), y(t)] / y_{\text{опт}}(t) - y(t) / dt}{y_{\text{опт}}(T) + y(T)},$$

– для дискретных:

$$r = M \frac{\sum \Psi[t_k, y_{\text{опт}}(t_k), y(t_k)] |\delta_{\text{опт}}(t_k) - \delta(t_k)|}{y_{\text{опт}}(T) + y(T)}. \quad (6.19)$$

Здесь

$$\delta_{\text{опт}}(t_k) = y_{\text{опт}}(t_k) - y_{\text{опт}}(t_k - t_0);$$

$$\delta(t_k) = y(t_k) - y(t_k - t_0).$$

В общем случае переменная  $y(t)$  является векторной функцией  $y = \{y_1(t), y_2(t), \dots, y_n(t)\}$ , и неупорядоченность производства определяется по формуле

$$r = \frac{\int_0^T \Psi |dy_{\text{опт}} - dy|}{\|y_{\text{опт}}(T) + y(T)\|},$$

где  $\Psi(t, y_{\text{опт}}, y, dy_{\text{опт}}, dy)$  – вектор-функция  $\Psi$ .

В качестве нормы  $\|y_{\text{опт}}(T) + y(T)\|$  может быть выбрана сумма компонент вектора  $y$ , которая представляет собой выпуск продукции по всем видам изделий за время  $T$ :

$$\|y_{\text{опт}}(T) + y(T)\| = \sum_{i=1}^n [y_{\text{опт}i}(t) + y_i(t)].$$

Определив коэффициент упорядоченности как  $R = 1/r$ , найдем экономическую эффективность проектируемой системы управления с учетом неупорядоченности производства:

$$\xi = \xi_{\text{max}} \left(1 - R_0 e^{-S \ln R_0 / S_0}\right), \quad (6.20)$$

где  $\xi_{\text{max}}$  – максимальное значение экономической эффективности от внедрения АСУ;  $R_0$  – начальная упорядоченность производства на момент введения АСУ в действие (определяется по экспериментальным данным);  $S$  и  $S_0$  – стоимость сбора и обработки информации для проектируемой и существующей системы управления).

Как следует из уравнения (6.20), учет неупорядоченности производства всегда ведет к снижению оценок экономической эффективности АСУ ТП, определенной по детерминированной методике расчета.

Оценка эффективности – немаловажна с позиций анализа технико-экономических показателей как отдельных технологических процессов, так и автоматизированных систем управления ими. На сегодня основной методикой решения поставленной задачи является детерминированный расчет экономической эффективности, позволяющий оценить перспективы внедрения АСУ как в объеме выпущенной продукции, так и в стоимостном выражении, в том числе в прибыли предприятия.

### КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Каковы основные условия функционирования производства?
2. Дайте определение основных источников экономической эффективности в АСУ.
3. Как определяются базовые затраты на создание и эксплуатацию систем управления?
4. Охарактеризуйте методику детерминированного расчета экономической эффективности АСУ.
5. Приведите базовые выражения для расчета экономической эффективности с учетом неупорядоченности производства и технологического процесса.

## ЗАКЛЮЧЕНИЕ

К настоящему времени сложились два основных пути разработки АСУ производственными и технологическими процессами: индивидуальное проектирование и проектирование с использованием типовых элементов. Первый подход представляет собой поэтапную разработку корпоративной системы собственными силами (включая использование готовых или заказных программных продуктов сторонних фирм и организаций, позволяющих автоматизировать отдельные рабочие места или производственные процессы), второй подразумевает внедрение готовой информационной системы корпоративного уровня. В условиях массового применения предприятиями корпоративных систем управления предпочтителен второй способ разработки АСУ.

В системах с высокой степенью автоматизации организационно-технологических процессов, включая программно-информационные технологии, происходит слияние процессов производства и их проектирования. Это объединение целесообразно поддержать интегрированием их технических средств, баз данных, персонала. Организационные проблемы и условия такого интегрирования весьма многообразны. Так, для улучшения условий технического обслуживания и эффективного использования средств автоматизации проектирования и управляющей вычислительной системы производства аппаратный состав вычислительной техники целесообразно реализовать на совместимых средствах, интегрированных в единый комплекс.

Внедрение корпоративных информационных систем управления производством (КСУП) как основы для комплексной автоматизации деятельности предприятий направлено на поддержку принятия управленческих решений менеджерами высшего звена корпорации. Предполагается, что предварительно должны быть решены задачи автоматизации рабочих мест, связанных с выполнением текущих производственных функций и оперативным управлением производственными процессами на уровне нижнего и среднего звена менеджеров.

Есть преимущества и у индивидуального проектирования АСУ. Главное из них состоит в том, что в создаваемой собственными силами системе в наибольшей степени можно учесть потребности и специфику работы конкретного предприятия. При этом следует иметь в виду, что не всегда это качество является достоинством. Нет гарантий, что существующий бизнес-процесс не является плохо организованным, а автоматизация этих процессов улучшит показатели их эффективности. Поэтому разработке информационной системы должен

предшествовать анализ, а если необходимо, то и реинжиниринг производственной деятельности. Тем не менее «эволюционный» характер постепенных улучшений с возможностью поэтапного финансирования разработок во многих случаях выглядит более привлекательным по сравнению с риском кардинальных преобразований и значительных затрат, связанных с внедрением готовых систем. К сожалению, этот путь решения проблемы автоматизации оказывается слишком растянутым во времени и зачастую превращается в «постоянный процесс разработки», когда разработчики не успевают за изменениями, происходящими в организации.

Рассмотренные в данном пособии методологии структурного анализа и проектирования информационных систем определяют руководящие указания для оценки и выбора проекта разрабатываемого модельного, алгоритмического и программного обеспечения КСУП, шаги работ, которые должны быть выполнены, их последовательность, правила распределения и назначения операций и методов. Они жестко регламентируют фазы анализа требований и проектирования спецификаций и отражают подход к разработке программно-алгоритмического обеспечения КСУП с позиций рецептов «кулинарной книги». Спецификации требований включают особенности программного обеспечения и его прогнозируемые характеристики, проекты пользовательских интерфейсов (меню, экраны и формы), критерии работоспособности программного обеспечения, программное и аппаратное окружение. Полученный документ спецификаций требований в дальнейшем преобразуется в проект архитектуры, детализирующий предполагаемую реализацию корпоративной информационной системы.

Корпорации, располагающие необходимыми финансовыми средствами, отдают предпочтение готовым программным системам. Однако успех от внедрения такой системы в значительной степени зависит от готовности (и возможности) корпорации работать по правилам, диктуемым ей. Готовая информационная система, как правило, имеет модульную архитектуру, а потому процесс ее внедрения может быть выполнен по этапам – начиная с модулей, автоматизирующих наиболее критичные участки работы. При этом обеспечивается целостность системы, позволяющая воспользоваться на соответствующих рабочих местах новыми функциями подключаемых модулей.

Определение способа построения распределенной программно-алгоритмической структуры КСУП неоднозначно и требует тщательного анализа со стороны руководства предприятия – только в этом случае возможно выбрать наиболее подходящий путь именно для своей организации.

## БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1. Левин А. А., Вальденберг Ю. С., Барышников И. В. Классификация автоматизированных систем управления технологическими процессами и производствами // Приборы и системы управления. 1970. № 4. С. 10–11.
2. Вальков В. М. Автоматизация управления производством изделий электроники. М. : Радио и связь, 1982.
3. Вальков В. М. Микроэлектронные управляющие вычислительные комплексы: системное проектирование и конструирование. Л. : Машиностроение, 1990.
4. Митропольский А. К. Техника статистических вычислений. 2-е изд. М. : Наука, 1971.
5. Алиев Р. А. Промышленные инвариантные системы автоматического управления. М. : Энергия, 1971.
6. Алиев Р. А. Принцип инвариантности и его применение для проектирования промышленных систем управления. М. : Энергоатомиздат, 1985.
7. Данилюк В. В., Кузнецов Б. И. Анализ инвариантных структур систем стабилизации прокатки // Теория инвариантности, теория чувствительности и их применение : тез. докл. VI Всесоюз. совещания. М. : Наука, 1982. С. 198–199.
8. Кухтенко А. И. Основные этапы формирования инвариантности. Ч. 1. Основополагающие работы // Автоматика. 1984. № 2. С. 3–13.
9. Кухтенко А. И. Основные этапы формирования инвариантности. Ч. 2. Расширение тематики исследований // Автоматика. 1985. № 2. С. 3–14.
10. Кухтенко, А. И. Основные этапы формирования инвариантности. Ч. 3. Нелинейные инвариантные системы // Автоматика. 1985. № 6. С. 3–14.
11. Павлов В. В. Инвариантность и автономность нелинейных систем управления. Киев : Наук. думка, 1971.
12. Колпакова Н. П. К теории систем с несколькими регулируемыми координатами // Теория инвариантности в системах автоматического управления : тр. II Всесоюз. совещания. М.: Наука, 1964. С. 144–154.
13. Вальков В. М., Вершин В. Е. Автоматизированные системы управления технологическими процессами. 3-е изд., перераб. и доп. Л. : Политехника, 1991.

14. Интегрированная АСУ кислородно-конвертерным цехом Днепропетровского металлургического комбината / В. А. Карлик, В. Ф. Лебедин, В. И. Шапиро и др. // Приборы и системы управления. 1988. № 2. С. 1–3.
15. Коуден Д. Д. Статистические методы контроля качества : пер. с англ. М. : Физматгиз, 1961.
16. Венецкий И. Г., Венецкая В. И. Основные математико-статистические понятия и формулы в экономическом анализе : справ. 2-е изд. М. : Статистика, 1979.
17. Венецкий И. В., Кильдишов С. Г. Основы математической статистики. М. : Госстатиздат, 1963.
18. Кучер В. Я., Лазарев Н. М. Определение оптимального уровня допустимой доли дефектной продукции // Приборы и системы управления. 1985. № 6. С. 38–39.
19. Кулик В. Т. Алгоритмизация объектов управления : справ. Киев : Наук. думка, 1968.
20. Налимов В. В. Теория эксперимента. М. : Наука, 1971.
21. Горбачев В. С. Микропроцессорные системы управления промышленными роботами // Микропроцессорные средства и системы. 1987. № 2. С. 30–37.
22. Ильин В. П., Лягунов М. М. Аппаратура дистанционного контроля и управления станками с УЧПУ // Микропроцессорные средства и системы. 1987. № 2. С. 41–42.
23. Ивахненко А. Г., Мюллер И. А. Самоорганизация прогнозирующих моделей. Киев : Техніка, 1985.
24. Ивахненко А. Г., Юрачковский Ю. П. Моделирование сложных систем по экспериментальным данным. М. : Радио и связь, 1987.
25. Степашко В. С. Алгоритмы МГУА как основа автоматизации процесса моделирования по экспериментальным данным // Автоматика. 1988. № 4. С. 44–55.
26. Степашко В. С., Юрачковский Ю. П. Современное состояние метода группового учета аргументов // Автоматика. 1986. № 4. С. 36–44.
27. Ивахненко А. Г. О выборе множества выходных переменных и применении МГУА для пассивного и активного планирования эксперимента // Автоматика. 1988. № 1. С. 92–94.
28. Атрощенко В. В., Брусиловский П. А., Фридман А. Н. Коллектив моделей для идентификации сложных технологических объектов управления // Автоматика. 1987. № 4. С. 14–20.

29. Юрачковский Ю. П. Методы самоорганизации модели в случае активного эксперимента // Автоматика. 1988. № 1. С. 94.
30. Аралбаева Г. Г., Сарычев А. П. Прогнозирование анодного эффекта в алюминиевых электролизерах по спектральным характеристикам на основе МГУА // Автоматика. 1987. № 6. С. 38–44.
31. Болтянский В. Г. Математические методы оптимального управления. М. : Наука, 1971.
32. Бойчук Л. М., Ивахненко Н. А., Путятин В. Г. Определение оптимальных весов для цифрового прогнозирующего фильтра // Автоматика. 1984. № 16. С. 459–462.
33. Методы и средства фильтрации периодических помех в АСУ ТП, связанные с формированием материальных потоков / Р. Т. Франко, М. С. Фельзер, И. Д. Стецюк и др. // Управляющие системы машины. 1984. № 5. С. 114–117.
34. Липаев В. В. Проектирование программных средств. М. : Высш. шк., 1990.
35. Филипс Д. Методы анализа сетей : пер. с англ. М. : Мир, 1984.
36. Головкин Б. А. Расчет характеристик и планирование параллельных вычислительных процессов. М. : Радио и связь, 1983.
37. Барский А. Б. Параллельные процессы в вычислительных системах. Планирование и организация. М. : Радио и связь, 1990.
38. Воеводин В. В. Математические модели и методы в параллельных процессах. М. : Наука, 1986.
39. Gonzalez J. M. Deterministic Processor Scheduling // Computing Surveys. 1977. Vol. 9, № 3. P. 173–204.
40. Ковалев И. В., Царев Р. Ю., Шиповалов Ю. Г. Параллельные процессы в информационно-управляющих системах. Формирование и оптимизация / под ред. А. В. Медведева ; НИИ СУВПТ. Красноярск, 2001.
41. Лебедев В. А., Трохов Н. Н., Царев Р. Ю. Параллельные процессы обработки информации в управляющих системах / НИИ СУВПТ. Красноярск, 2001.
42. Антамошкин А. Н. Регулярная оптимизация псевдодобулевых функций / Краснояр. гос. ун-т. Красноярск, 1989.
43. Алгоритмизация детерминированных моделей технологических циклов автоматизированных систем управления / И. В. Ковалев, П. В. Зеленков, В. В. Лосев и др. // Вестник СибГАУ. 2016. Вып. 17, № 3. С. 569–574.

44. Алексеев Н. А., Гаврилов Е. С., Слободин М. Ю. Универсальная информационная система мониторинга жизненного цикла продукции // Недра Кузбасса. Инновации : материалы 5-й Всерос. науч.-практ. конф. Кемерово : ИНТ, 2006. С. 51–52.

45. Мартин Д. Планирование развития автоматизированных систем. М. : Финансы и статистика, 1984.

46. Гаврилов Е. С., Зырянов Д. С. Модели и методы анализа бизнес-функций в корпоративных информационно-управляющих системах // Вестник университетского комплекса : сб. науч. тр. / под общ. ред. Н. В. Василенко. Вып. 3 (17) / ВСФ РГУИТП, НИИ СУВПТ. Красноярск, 2004. С. 201–217.

47. Ковалев И. В., Гаврилов Е. С. Формирование алгоритмов распределенной обработки и управления с учетом периодических задач АСУ // Актуальные проблемы экономики, права и информационных технологий : сб. науч. ст. Ч. 1 / Красноярский филиал МЭСИ. Красноярск, 2005. С. 102–107.

48. Царев Р. Ю., Семенько Т. И., Гаврилов Е. С. Информатика и вычислительная техника. Модели формирования и алгоритмы распределенной обработки информации и управления : учеб. пособие / Краснояр. гос. техн. ун-т. Красноярск, 2005.

49. Трапезников В. А. Автоматическое управление и экономика // Автоматика и телемеханика. 1966. № 1. С. 5–22.

50. Гаврилов Е. С., Алексеев Н. А., Чикизов А. А. Универсальная информационная система для отслеживания жизненного цикла продукции // Вестник университетского комплекса : сб. науч. тр. / под общ. ред. Н. В. Василенко. Вып. 6 (20) / ВСФ РГУИТП, НИИ СУВПТ. Красноярск, 2005. С. 88–90.

51. Оззу М. Т., Валдуриц П. Распределенные и параллельные системы баз данных // СУБД. 1996. № 4. С. 4–26.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Алексеев, Г. В. Компьютерные технологии при проектировании и эксплуатации технологического оборудования : учеб. пособие для вузов / Г. В. Алексеев, И. И. Бриденко, В. А. Головацкий и др. – 3-е изд., испр. и доп. – СПб. : ГИОРД, 2012. – 252 с.
2. Афонин, А. М. Теоретические основы разработки и моделирования систем автоматизации : учеб. пособие для сред. проф. образования / А. М. Афонин, Ю. Н. Царегородцев, А. М. Петрова и др. – М. : Форум, 2011. – 190 с.
3. Венделева, М. А. Информационные технологии управления : учеб. пособие / М. А. Венделева, Ю. В. Вертакова. – М. : Юрайт, 2012. – 462 с.
4. Ермоленко, А. Д. Автоматизация процессов нефтепереработки : учеб. пособие / А. Д. Ермоленко, О. Н. Кашин, Н. В. Лисицын и др. ; под общ. ред. В. Г. Харазова. – СПб. : Профессия, 2012. – 303 с.
5. Петровский, В. С. Автоматизация технологических процессов и производств в деревообрабатывающей отрасли : учебник для вузов / В. С. Петровский, А. Д. Данилов ; Воронеж. гос. лесотехн. акад. – Воронеж, 2010. – 431 с.
6. Петровский, В. С. Управление в автоматизированном производстве (лесопромышленный комплекс) : учебник для вузов / В. С. Петровский ; Воронеж. гос. лесотехн. акад. – Воронеж, 2013. – 448 с.
7. Схиртладзе, А. Г. Автоматизация технологических процессов и производств : учебник для вузов / А. Г. Схиртладзе, А. В. Федотов, В. Г. Хомченко. – М. : Высш. шк. ; Абрис, 2012. – 565 с.
8. Тарченков, В. Ф. Проектирование автоматизированных систем : конспект лекций / В. Ф. Тарченков ; Сиб. гос. технол. ун-т. – Красноярск, 2010. – 87 с.

## КЛЮЧЕВЫЕ ПОНЯТИЯ

Автоматизированная система управления  
Вариационное исчисление  
Градиентные методы  
Детерминированный процесс  
Жизненный цикл  
Компьютерный тренажер  
Корреляционный анализ  
Математическая модель  
Метод наименьших квадратов  
Метод ранжирования  
Микропроцессорная вычислительная техника  
Обработка информации  
Объемно-временные характеристики информационных потоков  
Планирование эксперимента  
Программирование  
Регрессионный анализ  
Системы комплексного (организационно-технологического) управления  
Системы оптимального управления  
Технологический процесс  
Этапы проектирования  
Экономическая эффективность  
GERT-моделирование

Учебно-теоретическое издание

**Ковалев Игорь Владимирович**  
**Лосев Василий Владимирович**  
**Брезицкая Валерия Витальевна**  
**Ковалев Дмитрий Игоревич**  
**Перанцева Александра Владимировна**  
**Туева Евгения Вячеславовна**

# **АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ**

*Учебное пособие*

Редактор *И. Д. Бочарова*  
Оригинал-макет и верстка *И. Д. Бочаровой*

Подписано в печать 09.12.2016. Формат 60×84/16. Бумага офисная.  
Печать плоская. Усл. печ. л. 14. Уч.-изд. л. 16,0. Тираж 50 экз.  
Заказ . С 197/16.

Санитарно-эпидемиологическое заключение  
№ 24.49.04.953.П.000032.01.03 от 29.01.2003 г.

Редакционно-издательский отдел Сиб. гос. аэрокосмич. ун-та.  
Отпечатано в отделе копировально-множительной техники  
Сиб. гос. аэрокосмич. ун-та.  
660037, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31.