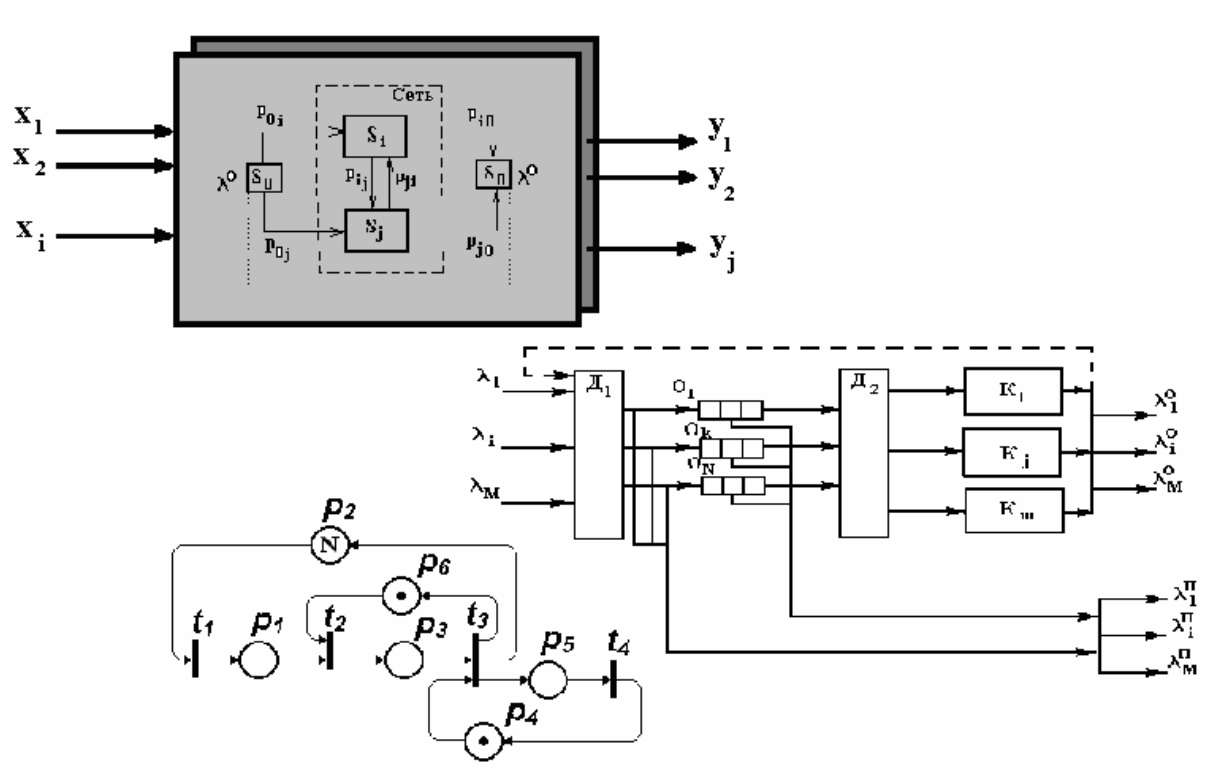
С.П. БОБКОВ, Д.О. БЫТЕВ

МОДЕЛИРОВАНИЕ СИСТЕМ

Учебное пособие



Иваново 2008

Федеральное агентство по образованию

Государственное образовательное учреждение
высшего профессионального образования
Ивановский государственный химико-технологический университет

Международный университет бизнеса и новых технологий (институт)

С.П. БОБКОВ, Д.О. БЫТЕВ

МОДЕЛИРОВАНИЕ СИСТЕМ

Учебное пособие

Рекомендовано учебно-методическим объединением
по образованию в области прикладной информатики
в качестве учебно-методического пособия
для студентов высших учебных заведений.

Иваново 2008

УДК 681.3

Бобков С.П. Моделирование систем: учеб. пособие / С.П. Бобков, Д.О. Бытев; Иван. гос. хим.-технол. ун-т. – Иваново, 2008. – 156 с. - ISBN

Цель учебного пособия – дать студентам общее представление о современных методах моделирования технических и технико-экономических систем и объектов.

В пособии рассматриваются общие вопросы и современная методология моделирования, непрерывные и дискретные детерминированные модели объектов и систем, стохастические модели с дискретным и непрерывным временем. Большое внимание уделено методам имитационного моделирования систем с вероятностными характеристиками. Дается обзор других подходов к моделированию сложных систем, таких как информационно-энтропийный, использование нейронных сетей и сетей Петри.

Учебное пособие предназначено для студентов, обучающихся по специальностям подготовки 080801 «Прикладная информатика» и 230201 «Информационные системы и технологии». Кроме того, пособие может быть полезным для студентов других специальностей и направлений.

Табл.7. Ил.92. Библиогр.:10 назв.

Печатается по решению редакционно-издательского совета Ивановского государственного химико-технологического университета.

Рецензенты:
кафедра прикладной математики Ивановского государственного энергетического университета; доктор физико-математических наук В.А.Соколов, (Ярославский государственный университет).

ISBN 5-9616-0268-6

© ГОУ ВПО Ивановский государственный химико-технологический университет», 2008

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 5
1. ОБЩИЕ ПОНЯТИЯ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ 6
1.1. Классификация видов моделирования 6
1.2. Классификация математических моделей 8
1.3. Параметры моделей и фазовые переменные 9
1.4. Требования к математическим моделям 11
1.5. Понятие математической схемы моделирования 12
1.6. Общая методика создания математических моделей 13
1.7. Основные понятия системного подхода к созданию математических моделей 16
2. ДЕТЕРМИНИРОВАННЫЕ МОДЕЛИ 20
2.1. Математические модели технических объектов 20
2.1.1. Компонентные функциональные уравнения объектов 20
2.1.2. Фазовые переменные и их аналогии 23
2.1.3. Топологические уравнения 24
2.1.4. Примеры создания моделей технических объектов 25
2.1.5. Модели технологических аппаратов 29
2.2. Конечные автоматы 31
2.2.1. Понятие конечного автомата 31
2.2.2. Способы описания и классы конечных автоматов 32
2.2.3. Другие виды конечных автоматов 37
3. СТОХАСТИЧЕСКИЕ МОДЕЛИ 39
3.1. Элементы теории марковских случайных процессов 39
3.1.1. Понятие случайного процесса 39
3.1.2. Дискретные цепи Маркова 40
3.1.3. Стационарное распределение вероятностей 43
3.1.4. Непрерывные марковские цепи 45
3.1.5. Уравнения А.Н. Колмогорова 46
3.1.6. Потоки событий 48
3.2. Основы теории массового обслуживания 51
3.2.1. Обобщенная структурная схема СМО. Параметры и характеристики 52
3.2.2. Разомкнутые СМО с ожиданием и терпеливыми заявками . 58
3.2.3. Предельные варианты разомкнутой СМО 62
3.2.4.Общий случай разомкнутой СМО 64
3.2.5. Замкнутые СМО 68
3.2.6. Сети массового обслуживания с простейшими потоками событий 73
3.3. Вероятностные автоматы 77

4. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ 81
4.1. Определение метода имитационного моделирования 81
4.2. Основные понятия имитационного моделирования 83
4.3. Основные этапы имитационного моделирования 85
4.4. Время в имитационных моделях. Псевдопараллелизм 86
4.5. Обобщённые алгоритмы имитационного моделирования 88
4.6. Моделирование случайных факторов 97
4.6.1. Моделирование базовых случайных величин 98
4.6.2. Моделирование непрерывных случайных величин с произвольным распределением 101
4.6.3. Моделирование дискретных случайных величин 104
4.6.4. Моделирование случайных событий и их потоков 106
4.7 Моделирование случайных процессов 110
4.7.1 Дискретные цепи Маркова 110
4.7.2 Непрерывные цепи Маркова 112
4.8. Обработка и анализ результатов имитационного моделирования . 115
4.8.1. Оценка вероятностных параметров 116
4.8.2. Оценка корреляционных параметров 116
4.8.3. Расчет средних по времени параметров СМО 117
4.9. Планирование экспериментов с имитационными моделями 118
4.10. Общие проблемы имитационного моделирования 121
5. ОБЗОР АЛЬТЕРНАТИВНЫХ ПОДХОДОВ К МОДЕЛИРОВАНИЮ СЛОЖНЫХ СИСТЕМ 123
5.1. Сети Петри 123
5.1.1. Определение сети Петри 123
5.1.2. Функционирование сети Петри 124
5.1.3. Анализ сетей Петри 128
5.2. Нейронные сети 131
5.2.1. Понятие нейронной сети 131
5.2.2. Искусственный нейрон 132
5.2.3. Основные виды активационных функций искусственных нейронов 134
5.2.4. Виды простейших нейронных сетей 138
5.2.5. Рекуррентные и самоорганизующиеся нейронные сети . . . 145
5.2.6. Общие замечания по использованию нейронных сетей 150
5.3. Информационно-энтропийный подход к моделированию систем 151
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ 156
. . . . . .

ВВЕДЕНИЕ

Моделирование является универсальным методом получения и использования знаний об окружающем мире. Моделирование всегда используется человеком в целенаправленной деятельности, особенно в исследовательской. В современных условиях усиливается роль и значение математического моделирования, которое с развитием средств вычислительной техники часто стали называть компьютерным.

Математические (компьютерные) модели, в силу своей логичности и строгого формального характера, позволяют выявить основные факторы, определяющие свойства изучаемых систем и исследовать их реакции на внешние воздействия и изменения параметров. Часто математические модели проще и удобнее использовать, чем натуральные (физические). Они позволяют проводить вычислительные эксперименты, реальная постановка которых затруднена или невозможна.

Изучение основных принципов математического моделирования является неотъемлемой частью подготовки специалистов в технических областях деятельности. Дисциплины, связанные с изучением основных аспектов моделирования объектов и систем в обязательном порядке входят в соответствующие учебные планы, являясь компонентами федеральных образовательных стандартов.

Целью данного учебного пособия является последовательное изложение современных методов моделирования. Пособие предназначено главным образом для студентов, обучающихся по специальностям и направлениям «Информационные системы» и «Прикладная информатика (по отраслям». Однако, учитывая опыт преподавания подобных дисциплин в технических вузах, авторы сочли целесообразным не ограничиваться рассмотрением только информационных систем, но и включить в текст рассмотрение технических и технико-экономических систем и объектов.

Материал пособия выстроен следующим образом. В первой главе рассматриваются общие вопросы и современная методология моделирования, использование системного подхода при создании математических моделей. Вторая глава посвящена рассмотрению непрерывных и дискретных детерминированных моделей объектов и систем. Предлагается использование метода аналогий при синтезе и анализе моделей технических объектов различной физической природы. В третьей главе изучаются стохастические модели с дискретным и непрерывным временем. Большое внимание в пособии уделено методам имитационного моделирования систем с вероятностными характеристиками, что составляет содержание четвертой главы. В пятой главе дается обзор других подходов к моделированию сложных систем, таких как информационно-энтропийный, использование нейронных сетей и сетей Петри.

1. ОБЩИЕ ПОНЯТИЯ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

1.1. КЛАССИФИКАЦИЯ ВИДОВ МОДЕЛИРОВАНИЯ

Задачей любого исследования, выполненного научными методами, является установление связей между воздействием на некоторый объект природы или техники и его реакцией на это воздействие. Этому предшествует выделение объекта из окружающего мира, с которым он связан очень большим числом связей и выявление тех связей или воздействий, которые наиболее существенны с точки зрения предпринимаемого исследования. Именно это является отправной точкой моделирования.

Когда мы исследуем некоторое природное явление, очень важен выбор причин и следствий. Это предполагает некоторый первичный анализ явления и его замену более упрощенным объектом - моделью явления. Важность предварительного анализа особенно проявляется, когда явление воспроизводится в лабораторных условиях и связи, кажущиеся не очень важными, не просто игнорируются, а исключаются.

Модель - это упрощенный образ изучаемого явления, создаваемый для исследования связей между такими его характеристиками, которые нас интересуют в данный момент. Иногда переход к исследованию других характеристик приводит к целесообразности использования совершенно непохожих моделей, хотя исследуемое явление остается одним и тем же.

Рис. 1.1. Классификация видов моделирования

Различают ***физическое*** и ***математическое*** моделирование.

Первое из них требует создания, обычно в специальных лабораториях, опытной установки, имитирующей объект или процесс. При этом обычно физическая модель имеет меньшие размеры, чем натуральный объект, но не исключена и обратная ситуация. Помимо пространственного масштабирования возможно и масштабирование времени, то есть на модели можно за сравнительно короткое время изучить явление, протекающее в природе долгие годы, и наоборот, внимательно рассмотреть мгновенно протекающий процесс.

Таким образом, при физическом моделировании используется сама система, либо подобная ей (летательный аппарат в аэродинамической трубе).

Однако физическому моделированию присущи недостатки, прежде всего, экономического характера. Созданию физической модели предшествуют предварительные работы по ее проектированию, изготовлению узлов и деталей, монтажу и наладке, оснащению вспомогательным оборудованием. Любая лабораторная установка требует площадей для ее размещения и персонала для обслуживания, потребляет энергетические и материальные ресурсы при эксплуатации. Кроме того, диапазон изменений исследуемых характеристик на физических моделях обычно невелик и ограничивается не только разумностью затрат на проведение опытов, но и возможностями конструкционных материалов, из которых изготовлена модель.

Математическое моделирование предполагает эксперименты с математическими моделями явлений. В отличие от физической модели, которая материальна, математическая модель является логическим объектом. Математическая модель – это упрощенный образ изучаемого явления, записанный с помощью математической символики. Процесс моделирования состоит из математических экспериментов, сущность которых основана на выполнении различных операций над математическими моделями. Обычно это решение систем уравнений или логических задач различного вида и сложности.

Таким образом, математическое моделирование – процесс установления соответствия математической модели **M** реальной системе **S** и исследование полученной модели с целью изучения характеристик реальной системы.

Применение математического моделирования позволяет исследовать объекты, реальные эксперименты над которыми затруднены или невозможны (дорого, опасно для здоровья, однократные процессы, невозможные из-за физических или временных ограничений – находятся далеко, еще или уже не существуют и т.п.).

В свою очередь, выделяют следующие виды математического моделирования: ***аналитическое, статистическое, имитационное***.

Аналитическое моделирование заключается в том, что процессы функционирования элементов системы записываются в виде математических соотношений (алгебраических, интегральных, дифференциальных, логических и т.д.). Аналитическая модель может быть исследована аналитическим методом, когда устанавливаются явные зависимости, получаются точные решения. Если математические зависимости, составляющие модель сложно или невозможно решить аналитически, то прибегают к численным методам, когда получаются приближенные решения. В самых сложных случаях аналитическую модель исследуют качественно, т.е. в явном виде находят не само решение, а его некоторые свойства.

Статистическое моделирование – это обработка статистических данных о системе (модели) с целью получения искомых характеристик системы.

Имитационное моделирование – это воспроизведение на ЭВМ (имитация) процесса функционирования исследуемой системы, соблюдая логическую и временную последовательность протекания процессов, что позволяет узнать

данные о состоянии системы или отдельных ее элементов в определенные моменты времени. Для имитации процесса обычно формулируется алгоритм (программа для ЭВМ), что позволяет проводить вычислительные эксперименты.

В соответствии с указанными видами моделирования различают и математические модели – ***аналитические, статистические*** и ***имитационные***. Часто вместо термина «статистические» употребляют понятие ***эмпирические*** модели.

Математическое моделирование получило особенно широкое распространение в связи с возросшими вычислительными возможностями современных компьютеров. Этот вид моделирования свободен от многих недостатков, которыми страдает физическое моделирование. Прежде всего, это гораздо более экономичный и удобный способ познания. Все эксперименты проходят над нематериальным объектом, существующим в виртуальной действительности. Затратами здесь можно считать использование вычислительных ресурсов и умственного труда человека-исследователя. При математическом моделировании диапазон изменения исследуемых параметров лимитируется только здравым смыслом и правилами математики.

Безусловно, создание математической модели и работа с ней требуют определенных затрат, но их объем обычно не идет ни в какое сравнение с затратами на создание и эксплуатацию лабораторных установок. Справедливости ради следует отметить, что в настоящее время все еще не удается полностью отказаться от услуг физического моделирования, особенно в естественных науках, поскольку некоторые параметры исследуемых процессов могут быть определены только экспериментально. Однако считается, что при использовании математического моделирования затраты в среднем сокращаются в 10-100 раз.

В целом в математическом моделировании более развита теоретическая основа. Если при физическом моделировании она проявляется, как правило, при выдвижении исходной гипотезы и осмыслении полученных опытных данных, то при математическом моделировании, кроме того, необходимо формализовать (перевести на язык математики и логики) изучаемые свойства, теоретически обосновать аналогию между моделью и реальным явлением, правильно интерпретировать и обобщить результаты математического эксперимента. Без этого математическое моделирование перестает быть достоверным источником информации о реальных явлениях.

1.2. КЛАССИФИКАЦИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Математическая модель есть совокупность математических объектов - чисел, переменных, матриц, множеств и т.д., а также соотношений между ними. Эта совокупность отражает наиболее важные с точки зрения исследователя свойства описываемого объекта. Математическое моделирование заключается в математических экспериментах, сущность которых основана на выполнении различных операций над математическими моделями.

В результате моделирования прогнозируются характеристики исследуемого объекта (процесса, вещества, технического устройства, системы), проводится его оптимизация, оцениваются возможности вариантов и т.д.

Помимо разделения моделей на приведенные выше классы по принципиальным методам работы с ними (аналитические, статистические и имитационные) существуют иные виды классификации.

В зависимости от целей дальнейшего использования все математические модели можно отнести к одному из двух крупных видов. При исследовании принципов работы исследуемого объекта, характера протекающих процессов, как правило, используются математические модели, отображающие закономерности функционирования объектов. Такие модели называются ***функциональными.*** Они обычно представляют собой системы уравнений различного типа. Функциональные модели используются при проектировании объектов и систем.

В то же время, при конструкторских разработках наиболее важными являются расположение объектов в пространстве, геометрические формы объектов, связи отдельных частей объектов между собой и др. Здесь преобладают математические модели, отражающие структурные характеристики. Такие модели называются ***структурными***. Они чаще всего представляются в виде матриц, таблиц, списков, графов и пр. Структурные модели используются при конструировании объектов.

При разработках информационных систем преобладают математические модели первого вида, поэтому в дальнейшем мы будем рассматривать именно такие модели. Структурные модели обычно используются в системах автоматизированного проектирования.

Все функциональные математические модели можно отнести к одному из двух классов по свойствам моделируемых объектов и виду используемого для анализа математического аппарата.

Объекты, процессы или системы, на функционирование которых существенное влияние оказывают случайные возмущающие факторы и воздействия описываются ***стохастическими (вероятностными)*** моделями. При работе с такими моделями обычно используют математический аппарат теории вероятностей, информационно-статистический или информационно-энтропийный подходы.

Объекты (системы), для которых предполагается, что их поведение можно описать однозначно или где можно игнорировать влияние случайных факторов, анализируются с помощью ***детерминированных*** моделей**.** При работе с ними, как правило, используют методы классической математики и математической физики.

1.3. ПАРАМЕТРЫ МОДЕЛЕЙ И ФАЗОВЫЕ ПЕРЕМЕННЫЕ

Среди свойств объекта, отражаемых в математических моделях, следует различать воздействия на объект и его реакцию на воздействия. Количественное выражение этих величин осуществляется с помощью ***параметров***. Любой

процесс или объект, исходя из внешних признаков, может быть условно изображен следующим образом (рис.1.2).

Рис. 1.2. Условное изображение объекта моделирования

При этом воздействия описываются *входными параметрами* $\mathbf{x_i}$, а реакция объекта моделирования - *выходными параметрами* $\mathbf{y_j}$**.** Последние характеризуют состояние объекта исследования и определяются суммарным воздействием входных параметров.

Среди входных параметров, в свою очередь, можно выделить:

– внешние параметры; их значения могут быть измерены, но возможность воздействовать на них отсутствует;

– управляющие параметры; на них можно оказывать прямое воздействие, в соответствии с теми или иными требованиями, что позволяет управлять процессом;

– возмущающие параметры; они изменяются случайным образом и не доступны для измерения.

Можно привести следующий пример. Для аудиосистемы внешним параметром является уровень (напряжение) входного сигнала, который можно измерить, но обычно нельзя регулировать. Управляющими параметрами здесь являются коэффициенты усиления, которые можно произвольно менять в некоторых пределах. Возмущающими параметрами в данном примере следует считать появление случайных помех в канале передачи. В качестве выходных параметров аудиосистемы выступают, например, выходная мощность сигнала, потребляемая мощность, величина искажений выходного сигнала и пр.

Пусть объект характеризуют **n** входных и **m** выходных параметров.

Тогда векторы этих параметров можно обозначить таким образом:

$$\overline{\mathbf{X}} = (\mathbf{x_1}, \mathbf{x_2}, ... \mathbf{x_n}); \ \overline{\mathbf{Y}} = (\mathbf{y_1}, \mathbf{y_2}, ... \mathbf{y_m}).$$

Поскольку свойства объекта зависят от входных параметров, имеет место зависимость:

$$\overline{\mathbf{Y}} = \mathbf{F}(\overline{\mathbf{X}}). \qquad (1.1)$$

Приведенная система соотношений является примером математической модели объекта.

Наличие математической модели вида (1.1) позволяет легко оценивать выходные параметры по известным значениям вектора **X.** Однако существование данной зависимости не означает, что она известна и может быть представлена именно в таком явном относительно вектора **Y** виде. Как правило, такую математическую модель удается получить только для очень простых объектов. Типичной является ситуация, когда математическое описание процессов в исследуемом объекте задается в форме системы уравнений, в которой фигурирует вектор **фазовых** переменных **V.** В свою очередь, входные и выходные параметры связаны зависимостями с фазовыми переменными:

$$\mathbf{L}[\overline{\mathbf{V}}(\overline{\mathbf{Z}})] = \mathrm{F}(\overline{\mathbf{Z}}), \text{ причем } \overline{\mathbf{X}} = \mathrm{Y}_1(\overline{\mathbf{V}}),\ \overline{\mathbf{Y}} = \mathrm{Y}_2(\overline{\mathbf{V}}), \qquad (1.2),$$

где **L** - некоторый математический оператор; **Z** - вектор независимых переменных, в общем случае включающий время и пространственные координаты; **Ф(Z)** - заданная функция независимых переменных.

Фазовые переменные характеризуют физическое состояние объекта, а их изменения во времени выражают переходные процессы в объекте. Наиболее типичным примером фазовых переменных (для упомянутой выше аудиосистемы) являются величины электрического тока и напряжения, поскольку с их помощью можно описать все входные и выходные параметры данного устройства. При моделировании механических систем фазовыми переменными являются силы и скорости, для гидравлических систем – давления и расходы и т.д.

На практике довольно часто встречаются случаи, когда объект настолько сложен, что его структура либо неизвестна совсем, либо ее корректное математическое описание невозможно. В таких случаях исследователь вынужден игнорировать внутренние процессы, протекающие в объекте, и анализировать лишь влияние входных параметров на выходные. При этом модели получаются путем обработки статистических данных и относятся к классу статистических. Однако в литературе имеется еще одно название для таких математических моделей – модели типа «***черный ящик***».

1.4. ТРЕБОВАНИЯ К МАТЕМАТИЧЕСКИМ МОДЕЛЯМ

К математическим моделям предъявляются требования универсальности, точности, адекватности и экономичности.

Степень ***универсальности*** математической модели характеризует полноту отображения в модели свойств реального объекта. Обычно математическая модель отражает только некоторые свойства объекта. Так, при научных исследованиях большинство математических моделей описывают протекание тех или иных процессов. При этом не требуется, чтобы модель описывала, скажем, форму объекта или его цвет. Универсальность математической модели отражает ее применимость к широкому классу объектов. Степень универсальности не имеет количественной оценки.

Точность математической модели оценивается степенью совпадения значений параметров реального объекта и значений тех же параметров, рассчитанных с помощью математической модели.

Пусть отражаемые в математической модели свойства оцениваются вектором выходных параметров **Y**. Тогда обозначив истинное значение **j**-го выходного параметра через $y_{j\,ист}$, а рассчитанное с помощью математической модели $y_{j\,мм}$, определим относительную погрешность e_j расчета параметра y_j так:

$$e_j = (y_{j\,мм} - y_{j\,ист}) / y_{j\,ист} \ . \qquad (1.3)$$

Таким образом, получается векторная оценка точности математической модели:

$$\overline{E} = (e_1, e_2, \ldots e_m) \ . \qquad (1.4)$$

Если необходимо, можно представить ее в скалярной форме, используя какую-либо норму вектора, например

$$E = \max(e_j). \qquad (1.5)$$

Адекватность математической модели - способность отображать заданные свойства объекта с погрешностью не выше заданной. Поскольку выходные параметры являются функциями вектора входных параметров, погрешность зависит от их значений. Как правило, адекватность модели имеет место лишь в ограниченной области изменения входных и управляющих параметров, в так называемой ***области адекватности*** математической модели.

Экономичность математической модели характеризуется затратами вычислительных ресурсов (затратами машинного времени и памяти компьютера). Впрочем, часто экономичность модели зависит не только от ее свойств, но и от особенностей операционной системы компьютера, языка программирования, модели компьютера.

Можно заметить, что требования точности, универсальности, широкой области адекватности, с одной стороны, и экономичности с другой - противоречивы. Наилучшее компромиссное удовлетворение этих требований зависит от особенностей решаемых задач и полностью определяется разработчиком.

1.5. ПОНЯТИЕ МАТЕМАТИЧЕСКОЙ СХЕМЫ МОДЕЛИРОВАНИЯ

Исходной информацией при разработке математической модели системы или объекта служат данные о назначении и условиях ее работы. Эта информация определяет основную цель моделирования, требования к модели, уровень абстрагирования, выбор математической схемы моделирования.

Понятие ***математическая схема*** позволяет рассматривать математику не как метод расчёта, а как метод мышления, средства формулирования понятий, что является наиболее важным при переходе от словесного описания к формализованному представлению процесса функционирования системы в виде некоторой математической модели.

При пользовании математической схемой исследователя, в первую очередь, должен интересовать вопрос об адекватности отображения реальных процессов в исследуемой системе в виде конкретных модельных схем, а не возможность получения ответа (результата решения) на конкретный вопрос исследования. Например, представление процесса функционирования информаци-

онной системы коллективного пользования в виде сети схем массового обслуживания даёт возможность хорошо описать процессы, происходящие в системе, но при сложных законах входящих потоков и потоков обслуживания не дает возможности получения результатов в явном виде.

Математическую схему можно определить как звено при переходе от содержательного к формализованному описанию процесса функционирования системы с учётом воздействия внешней среды. В этой связи важно рассматривать такое важное свойство системы (объекта), как ее состояние. Совокупность всех возможных значений состояний называется пространством состояний объекта моделирования. Это пространство теоретически может быть бесконечным, а может быть ограниченным. Состояния объекта (системы) могут меняться непрерывно, но могут иметь дискретный характер. Также непрерывными или дискретными могут быть независимые переменные модели (пространственные координаты и время).

Таким образом, учитывая вид и характер параметров, законы функционирования модели можно разделить не только на детерминированные или стохастические, но и на непрерывные или дискретные.

Комбинация этих признаков позволяет выделить следующие классы типовых математических схем:

- непрерывно-детерминированные (D схемы);
- дискретно-детерминированные (F схемы);
- непрерывно-стохастические (Q схемы);
- дискретно-стохастические (P схемы);
- гибридные или комбинированные.

1.6. ОБЩАЯ МЕТОДИКА СОЗДАНИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

В общем случае процесс создания математической модели включает в себя следующие этапы:

1. Постановка задачи исследования. На этом этапе осуществляется выбор свойств объекта (системы), которые подлежат отражению в модели и отбрасывание тех свойств, которые на данном этапе исследования разработчик считает несущественными. Этот выбор основан на анализе возможных областей применения модели и определяет степень ее универсальности. Далее необходим сбор исходной информации о выбранных свойствах объекта. Источниками сведений могут быть опыт и знания разработчиков; научно-техническая литература, прежде всего справочная; описание прототипов - имеющихся моделей для объектов, близких по свойствам к исследуемому; результаты экспериментальных исследований.

2. Синтез структуры математической модели. Этап заключается в получении общего вида математических соотношений без конкретизации числовых значений фигурирующих в них параметров.

3. Определение числовых значений параметров модели. Эта операция ставится, как задача минимизации погрешности математической модели задан-

ной структуры. Она иногда носит название параметрический синтез. При этом часто используются экспериментальные значения выходных параметров объекта исследования при заданных входных и управляющих.

4. Анализ модели. На этом этапе производится оценка точности и адекватности полученной математической модели. Для этой оценки должны применяться те значения выходных параметров, которые не были использованы ранее при определении числовых значений параметров

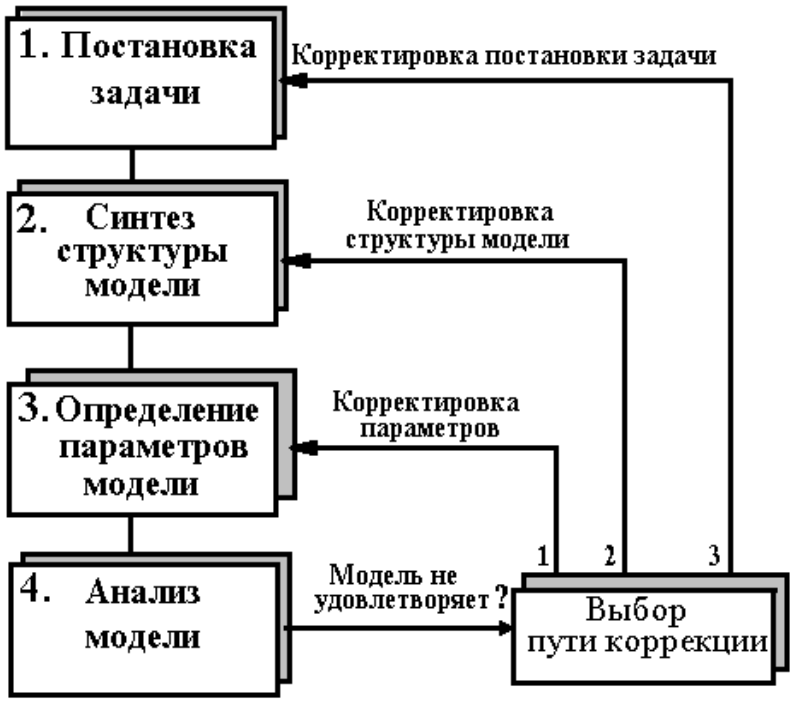


Рис. 1.3. Общая схема получения математической модели.

Следует отметить, что значительную ценность представляют не оценки точности, выполненные в двух-трех точках пространства внешних переменных, а сведения об области адекватности математической модели. И хотя определение области адекватности требует больших вычислительных и экспериментальных затрат, это придает математической модели завершенность и возможность дальнейшего многократного использования.

Общая схема процесса получения аналитических и статистических моделей представлена на рис. 1.3.

Если анализ модели показывает, что она неудовлетворительно описывает объект, то сначала следует попытаться изменить числовые значения некоторых параметров. Когда и данный путь не привел к успеху, нужно скорректировать структуру модели, а затем повторить нижеследующие этапы. Возможна ситуация, при которой мы и теперь не получили желаемого результата. В таком случае ошибка кроется на стадии постановки задачи, то есть необходимо возвратиться в самое начало работы и повторить процесс вновь.

Из рассмотренной схемы видно, что этапы могут выполняться неоднократно в процессе приближения к желаемому результату. Необходимо также добавить, что этап анализа обычно заключается в однократном решении уравнений, составляющих математическую модель. Однако в процессе получения адекватной модели он повторяется много раз.

Этап постановки задачи является основополагающим, наиболее ответственным и важным этапом. Он целиком базируется на знаниях об объекте исследования. Постановка задачи абсолютно не поддается формализации, здесь невозможно дать заранее приготовленных рецептов. Это творческий процесс и его успех полностью зависит от опыта, знаний и интуиции исследователя.

Немаловажную роль опыт и интеллект исследователя играют и на стадии синтеза структуры математической модели. Это тоже творческая, плохо формализуемая операция. Однако существуют определенные методические подходы, позволяющие несколько облегчить получение математических моделей. О них будет сказано ниже.

Другие этапы получения математической модели, а также работа с ней допускают использование методов вычислительной математики и могут быть сравнительно легко формализованы. Поэтому они должны выполняться с использованием современных средств вычислительной техники. Для этого требуется создать соответствующие алгоритмы и разработать программы для компьютера. Следует сказать, что в настоящее время существуют разнообразные пакеты прикладных программ, реализующие многие математические методы решения уравнений и задачи оптимизации. Это программное обеспечение может быть полезно как при параметрическом синтезе, так и при анализе полученной модели. Однако, нельзя забывать, что численные методы могут вносить заметные погрешности и даже существенно искажать полученные данные. Поэтому, применяя их, следует внимательно относиться к интерпретации результатов математического моделирования.

При получении математических моделей можно использовать теоретический и формальный методы.

Аналитические математические модели создаются в результате исследований процессов и их закономерностей. Они базируются на фундаментальных законах, а также общепринятых положениях в данной области науки. При выводе таких моделей применяют строгие математические преобразования.

Также неформально можно получить и статистические модели. Однако при их получении изучаются лишь внешние проявления свойств объекта исследований - выходные параметры и фазовые переменные, а затем производится обработка экспериментальных результатов.

Формальные методы получения моделей используются при известных математических моделях элементов, составляющих объект исследования.

1.7. ОСНОВНЫЕ ПОНЯТИЯ СИСТЕМНОГО ПОДХОДА К СОЗДАНИЮ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Системный подход как новая методология науки и практики сложился ко второй половине XX столетия. Он является качественно новым подходом в изучении, проектировании и создании систем. Формирование системного подхода в качестве самостоятельного исследовательского направления обусловлено общей тенденцией развития науки и общества, которая сложилась к настоящему времени.

Центральным понятием системного анализа является понятие ***система.***

При построении математических моделей принципиальное значение имеют свойства систем. Помимо рассмотренных выше видов классификации систем на детерминированные и стохастические, дискретные и непрерывные следует выделить еще ряд характерных признаков.

Динамические системы характеризуются тем, что их выходные параметры в данный момент времени определяются характером входных воздействий в прошлом и настоящем (зависит от предыстории). В противном случае системы называют ***статическими.***

Если параметры систем изменяются во времени, то она называется ***нестационарной***, противоположным понятием является понятие ***стационарной*** системы.

Различают системы ***линейные*** и ***нелинейные***. Для линейных систем реакция на сумму двух иди более различных воздействий эквивалентна сумме реакций на каждое возмущение в отдельности, для нелинейных – это условие не выполняется.

Можно выделить следующие основные определения и свойства системы:

- система есть совокупность элементов (подсистем). При определенных условиях элементы сами могут рассматриваться как системы, а исследуемая система – как элемент более сложной системы;
- связи между элементами в системе превосходят по силе связи этих элементов с элементами, не входящими в систему. Это свойство позволяет выделить систему из среды;
- для любой системы характерно существование ***интегративных*** качеств (свойство ***эмерджентности***), которые присущи системе в целом, но не свойственны ни одному ее элементу в отдельности: систему нельзя сводить к простой совокупности элементов;
- система всегда имеет цели, для которых она функционирует и существует.

В качестве примера можно привести следующее. Если в качестве системы рассматривать компьютерную сеть, то входящие в нее сервер, рабочие станции, маршрутизаторы и т.п. являются элементами системы (подсистемами). В свою очередь, например рабочий компьютер, тоже состоит из элементов – подсистем более низкого уровня (блок питания, материнская плата, жесткий диск и т.д.). Подобное разбиение можно продолжать.

Методология системного подхода при решении задач анализа систем состоит в следующем. Задача расчленяется на подзадачи анализа элементов системы. Причем каждый из элементов должен рассматриваться не сам по себе, а во взаимодействии с другими элементами. Решение подзадач должно происходить при условии обеспечения общих целей функционирования всей системы.

При создании математических моделей использование системного подхода предполагает выделение нескольких уровней абстракции при описании объекта (системы). Объединение уровней, родственных по характеру используемого математического аппарата, приводит к образованию нескольких уровней в иерархии функциональных моделей. Наиболее наглядно это можно продемонстрировать на примере моделирования технических объектов.

В качестве примера моделей разного уровня, описывающих один и тот же процесс, рассмотрим нагрев твердого тела.

Если некоторое твердое тело рассматривать, как единую систему, нагревающуюся под действием теплового потока, то процесс можно описать с помощью простого балансового уравнения:

$$\mathbf{T} = \mathbf{T_0} + \frac{\mathbf{1}}{\mathbf{C_P}\,\mathrm{r}}\mathbf{Q}, \qquad \text{(a)}$$

где $\mathbf{T}$ – конечная температура тела; $\mathbf{T_0}$ – его начальная температура; $\mathbf{C_P}$, r - теплоемкость и плотность материала тела; $\mathbf{Q}$ - количество теплоты, полученной телом, отнесенное к его объему.

Будем считать модель (a) моделью верхнего, самого общего уровня абстракции.

Допустим, нас интересует не только конечная температура объекта, но и ее изменение во времени, т.е. кинетика процесса нагрева. Тогда следует перейти к более подробному описанию процесса, например, в таком виде:

$$\frac{\mathbf{dT}}{\mathbf{d}\,\mathrm{t}} = \frac{\mathbf{1}}{\mathbf{C_P}\,\mathrm{r}}\mathbf{q(t)}, \qquad \text{(b)}$$

где $\mathbf{q(t)}$ - тепловой поток , отнесенный к единице объема тела; t - время.

Для решения приведенного уравнения следует указать закон изменения во времени теплового потока $\mathbf{q(t)}$ и задать начальную температуру тела:

$$\mathbf{T}\big|_{\,\mathrm{t}=\mathbf{0}} = \mathbf{T_0}\,.$$

Полученное решение позволит определить изменение температуры тела во времени.

Заметим, что модели (a) и (b) рассматривают тело, как единое целое и в них не входят пространственные координаты.

Наконец, если исследователю важно провести анализ изменения температуры не только во времени, но и в различных точках пространства, то следует перейти к еще более подробной детализации процесса. Примем для простоты, что тело имеет форму длинного тонкого стержня. Если пренебречь всеми размерами стержня кроме его длины, модель можно представить в виде следующего уравнения:

$$\frac{d\mathbf{T}}{dt} = \frac{\lambda}{\mathbf{C_P}\rho}\left(\frac{d^2\mathbf{T}}{d\mathbf{x}^2}\right), \qquad (c)$$

которое следует дополнить начальными и граничными условиями, уточняющими протекание процесса, например такими:

$$\mathbf{T}\big|_{\mathbf{x=0}} = \mathbf{f_1}(t)\ ;\ \mathbf{T}\big|_{\mathbf{x=l}} = \mathbf{f_2}(t);\ \mathbf{T}\big|_{t=0} = \varphi(\mathbf{x}),$$

где λ - теплопроводность материала стержня; **l** - его длина; **x** - пространственная координата.

Решение данной модели позволит определить температуру в любой точке стержня в любой момент времени.

Проследим, как изменяется вид уравнений математической модели процесса при переходе от одного уровня абстракции к другому.

Модель процесса, представленная на верхнем уровне простейшим алгебраическим уравнением (a), усложняется на втором уровне абстракции и принимает вид обыкновенного дифференциального уравнения (b). Переход к третьему самому подробному уровню приводит к необходимости использовать дифференциальное уравнение с частными производными (c). Однако за счет усложнения модели мы получаем дополнительную информацию о процессе. Так, если в первом случае мы можем определить лишь конечную температуру моделируемого объекта, то во втором имеем возможность проследить процесс во времени, считая температуру одинаковой во всем объеме тела. Модель третьего уровня уже позволяет исследовать распределение температуры и во времени, и в пространстве.

Безусловно, в данном примере дается чрезвычайно упрощенный подход к описанию процесса. Так не рассматривается отдача тепла нагретым телом в окружающую среду. В модели (c) процесс рассматривается лишь по одной координате, а реальные тела имеют конечные размеры по всем пространственным координатам. Можно добавить и другие условия, не учтенные в примере. Учет этих условий должен привести к появлению новых членов в уравнениях и значительно усложнить их. Однако усложнение математической модели делает ее более адекватной, более приближенной к реальности, хотя и ухудшает ее экономичность.

При моделировании технических объектов часто рассмотренные выше модели и уровни абстракции называются следующим образом. Модель вида (c) называют моделью ***микроуровня***, вида (b) – моделью ***макроуровня***, вида (a) – ***мегауровня***. При этом характерно следующее.

1. На микроуровне абстракции используют математические модели, описывающие физическое состояние и процессы в сплошных средах. Для моделирования применяют аппарат математической физики. Особенностью этих математических моделей является отражение процессов, протекающих в непрерывных пространстве и времени. Типичными математическими моделями этого уровня являются уравнения гидродинамики, теплопереноса, диффузии, упругости. Они представляют собой системы дифференциальных уравнений с ча-

стными производными. В них независимыми переменными являются время и пространственные координаты. Такие математические модели часто называют моделями с ***распределенными параметрами***, поскольку в них параметры и фазовые переменные зависят от координат точек пространства. Если в таких уравнениях время как независимая переменная отсутствует, то они описывают ***стационарный*** процесс и называются стационарными. Исследование таких моделей сводится к решению краевых задач. Следует заметить, что, несмотря на полноту описания процесса, возможности применения таких моделей ограничены. Попытки исследовать с их помощью процессы в многокомпонентных средах не всегда успешны из-за чрезмерных вычислительных затрат.

2. На макроуровне производится укрупнительная дискретизация пространства по функциональному признаку. То есть выделяются характерные зоны, в которых процесс можно считать не зависящим от пространственных координат. Математические модели на этом уровне представляются в виде систем обыкновенных дифференциальных уравнений, где в качестве независимой переменной присутствует только время. Данные модели называют моделями с ***сосредоточенными параметрами***. При рассмотрении стационарного процесса на данном уровне математические модели получают вид систем алгебраических уравнений. Математические модели данного уровня являются универсальными и пригодными к исследованию как динамических, так и статических режимов процесса.

3. На мегауровне с помощью дальнейшего абстрагирования от характера физических процессов удается еще более упростить модель. Обычно в ней фигурируют только фазовые переменные, относящиеся к внешним связям объекта. Типичными моделями этого уровня являются балансовые соотношения в виде систем алгебраических уравнений.

2. ДЕТЕРМИНИРОВАННЫЕ МОДЕЛИ

Все процессы и явления, в которых участвуют объекты материального мира, происходят в пространстве и времени. При этом характеристики процесса или объекта меняются, что дает возможность говорить об изменении его состояния. В каждый момент времени объект находится в одном из возможных состояний и способен переходить во времени и пространстве из одного состояния в другое под действием внешних и внутренних причин. Таким образом, изучаемый объект можно рассматривать как ***динамическую*** (изменяющуюся) ***систему*** в широком смысле этого понятия. В самом общем случае, как пространство состояний, так и пространство независимых переменных могут быть непрерывными или дискретными.

Рассмотрение процессов в непрерывных пространстве и времени традиционно производится с использованием математических моделей в виде дифференциальных уравнений с частными производными. К ним относятся уравнения переноса тепла, переноса массы, уравнения механики сплошной среды, электродинамики, упругости и пр. Все они часто объединяются общим понятием – ***уравнения математической физики***. Подробное рассмотрение этих вопросов выходит за рамки изучаемого предмета, поэтому в данной главе мы ограничимся детерминированными моделями, где в качестве независимой переменной используется только время.

2.1. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ТЕХНИЧЕСКИХ ОБЪЕКТОВ

Физические процессы, как и технические объекты, принято разделять на механические, тепловые, диффузионные, электрические и пр. Модели технических объектов обычно несложно получить, используя фундаментальные законы соответствующих разделов физики. При этом весьма удобно применять ***аналогию*** в математическом описании объектов различной физической природы. Формальная (на уровне модельных представлений) аналогия различных процессов имеет важный физический смысл, иллюстрируя общность фундаментальных свойств материи. Использование аналогии не только помогает при математическом моделировании, но также позволяет применять общие формальные приемы и программные разработки при моделировании явлений различной природы.

2.1.1. Компонентные функциональные уравнения объектов

Любая техническая система состоит из ряда более простых подсистем (элементов). Зависимости, описывающие функционирование элементарных подсистем, которые являются компонентами единой технической системы, часто называют ***компонентными*** уравнениями.

При рассмотрении аналогии компонентных уравнений для различных физических систем целесообразно в качестве базовой системы выбрать электрическую как наиболее наглядную.

Рассмотрим три уравнения, которыми описываются основные процессы в электрической цепи:

$$\mathbf{i} = \frac{\mathbf{u}}{\mathbf{R}} \quad (2.1a); \qquad \mathbf{i} = \mathbf{C_Э}\frac{\mathbf{du}}{\mathbf{d}t} \quad (2.1b); \qquad \mathbf{u} = \mathbf{L_Э}\frac{\mathbf{di}}{\mathbf{d}t}, \qquad (2.1c)$$

где **i** – электрический ток; **u** – электрическое напряжение; **R** – электрическое сопротивление; $\mathbf{C_Э}$ электрическая емкость; $\mathbf{L_Э}$ - электрическая индуктивность.

Первое из данных уравнений описывает процесс рассеивания электрической энергии (преобразование ее в тепло). Второе и третье уравнения описывают процессы накопления энергии в потенциальной и кинетической формах. Отметим, что все эти процессы описываются с использованием тока **i** и напряжения **u**, которые являются здесь фазовыми переменными.

Можно сказать, что в электрической системе существуют три вида простейших элементов – типа **R,** типа **C**, и типа **L**. Сочетанием этих простейших элементов, а также источников фазовых переменных можно получить математическую модель любой электрической цепи.

Выделить указанные типы простейших элементов, на которых энергия рассеивается или накапливается можно в системе любой физической природы. При этом не трудно записать математические модели этих элементов, которые в явном виде не всегда идентичны уравнениям (2.1), но могут быть соответствующим образом преобразованы.

Рассмотрим механическую систему с поступательным движением.

Потери энергии в этой системе происходят из-за трения, накопление – в элементах массы (вследствие инерции), и в упругих элементах.

Известно уравнение вязкого трения, которое можно записать:

$$\mathbf{F = k_{тр}v}, \qquad (2.2a)$$

где **F** – сила; **v** – скорость; $\mathbf{k_{тр}}$ - коэффициент трения.

Показать, что это аналог уравнения (2.1a) можно так:

$$\mathbf{F} = \frac{\mathbf{v}}{\mathbf{R_M}} \text{ где } \mathbf{R_M = 1/k_{тр}}.$$

Аналогом уравнения (2.1b) является второй закон Ньютона

$$\mathbf{F} = \mathbf{m}\frac{\mathbf{dv}}{\mathbf{d}t}, \qquad (2.2b)$$

где масса **m** – мера инерционности, является аналогом электрической емкости.

Наконец, аналогом уравнения (2.1c) может служить уравнение упругости пружины:

$$\mathbf{F = k_y x}, \qquad (2.2c),$$

где $\mathbf{k_y}$ - коэффициент упругости; **x** – перемещение.

На первый взгляд здесь мало общего. Сделаем преобразования:

$$\frac{\mathbf{dF}}{\mathbf{d}t} = \mathbf{k_y}\frac{\mathbf{dx}}{\mathbf{d}t} = \mathbf{k_y v} \text{ или } \quad \mathbf{v} = \mathbf{L_M}\frac{\mathbf{dF}}{\mathbf{d}t},$$

где $\mathbf{L_M = 1/k_y}$ – аналог индуктивности.

В данной системе процессы описываются с использованием в качестве фазовых переменных скорости **v** и силы **F**.

Такие примеры можно продолжать.

Далее приведем без выводов основные компонентные уравнения для других физических систем.

Механическая система с вращательным движением.

Фазовые переменные – момент сил **M** и угловая скорость w**.**

Уравнение вязкого трения вращения:

$$\mathbf{M} = \mathbf{k}_{\text{тр вр}}\omega, \qquad (2.3a)$$

где $\mathbf{k}_{\text{тр вр}}$ - коэффициент трения вращения.

Основное уравнение динамики вращательного движения

$$\mathbf{M} = \mathbf{J}\frac{\mathbf{d}\omega}{\mathbf{d}t}, \qquad (2.3b)$$

где **J** – момент инерции.

Уравнение упругости спиральной пружины:

$$\mathbf{M} = \mathbf{k}_{\text{Ж}}\varphi, \qquad (2.3c)$$

где $\mathbf{k}_{\text{Ж}}$ - коэффициент жесткости пружины; φ – угол закручивания.

Деформирование твердого тела.

Фазовые переменные – механическое напряжение σ и относительная деформация **e.**

Потери на вязкое трение описываются законом Ньютона для вязкой среды:

$$\sigma = \mu\frac{\mathbf{d}\varepsilon}{\mathbf{d}t}, \qquad (2.4a)$$

где μ - коэффициент динамической вязкости.

Накопление энергии упругости подчиняется закону Гука:

$$\sigma = \mathbf{E}\varepsilon, \qquad (2.4b)$$

где **E** – модуль (коэффициент) упругости.

Гидравлическая (пневматическая) система.

Фазовые переменные – скорость потока – **v** и давление **P.** Вместо скорости потока чаще используют величину расхода жидкости (газа):

$\mathbf{G_v = vS}$ объемный расход [м^3/с]**;** $\mathbf{G_m = vS}\rho$ – массовый расход [кг/с],

где **S** – площадь сечения; ρ - плотность жидкости (газа).

Потери на трение для участка трубопровода можно выразить так:

$$\mathbf{G_m} = \frac{\mathbf{1}}{\mathbf{R_Г}}\mathbf{P}, \qquad (2.5a)$$

где $\mathbf{R_Г}$ – гидравлическое сопротивление.

Накопление энергии, вследствие сжимаемости потока в объеме:

$$\mathbf{G_m} = \frac{V\rho}{\mathbf{k}_V}\frac{\mathbf{dP}}{\mathbf{d}t}, \qquad (2.5b)$$

где V – объем; $\mathbf{k}_V$ - коэффициент объемного сжатия.

Накопление энергии вследствие движения потока жидкости по трубопроводу длиной **L**:

$$\mathbf{P} = \frac{\mathbf{L}}{\mathbf{S}}\frac{\mathbf{dG_m}}{\mathbf{d}t}, \qquad (2.5c)$$

Тепловая система.

Фазовые переменные – поток теплоты **q** и температура **T.**

Препятствиями для переноса тепла проводимости служат элементы с плохой проводимостью тепла (кондукционные сопротивления). По аналогии с (2.1a) можно записать:

$$\mathbf{q} = \frac{\mathbf{1}}{\mathbf{R_T}}\mathbf{T}, \qquad (2.6a)$$

где $\mathbf{R_Г} = \mathbf{L/(}\lambda\mathbf{S)}$ – сопротивление теплопроводности, **L S** – длина и сечение участка; λ -коэффициент теплопроводности материала участка.

Тепловая энергия накапливается телом вследствие его теплоемкости:

$$\mathbf{q} = \mathbf{C_T m}\frac{\mathbf{dT}}{\mathbf{d}t}, \qquad (2.6b)$$

где $\mathbf{C_T}$ – теплоемкость; **m** – масса.

Наконец, следует помнить, что тепловой поток в данных задачах эквивалентен мощности источников (стоков) тепла и равен производной по времени от количества переданного тепла:

$$\mathbf{q} = \frac{\mathbf{dQ}}{\mathbf{d}t}. \qquad (2.6c)$$

2.1.2. Фазовые переменные и их аналогии

Анализируя зависимости (2.1) - (2.6) легко заметить следующее.

Функционирование любой системы, по сути, может быть рассмотрено, как преобразование материи (энергии или вещества). В объекте (системе) любой физической природы, в общем случае, можно выделить элементы трех типов:

- рассеивающие энергию;
- накапливающие энергию в кинетической форме;
- накапливающие энергию в потенциальной форме.

Компонентные уравнения для элементов формально идентичны. Все они являются уравнениями связи между соответствующими фазовыми переменными. Во всех рассмотренных системах фазовые переменные характеризуют состояние процесса, а их изменения во времени выражают переходные процессы соответствующей физической природы. Но имеются и важные отличия в свойствах указанных фазовых переменных.

Одна часть из них выражает ***интенсивные*** характеристики рассматриваемой системы, поэтому их можно называть ***потенциалами*** или фазовыми переменными ***потенциального типа***. Типичными представителями этой группы можно назвать напряжение **u** для электрических систем или температура **T** для тепловых. Основным их свойством является то, что под действием разности (градиента) потенциалов возникают потоки энергии или массы. В свою очередь, потоки выражают ***экстенсивные*** характеристики систем и описываются фазовыми переменными ***потокового типа***. Так, разность напряжений вызывает электрический ток, а разность температур – поток теплоты от более нагретого участка к участку с меньшей температурой.

Приведем таблицу фазовых переменных в компонентных уравнениях для систем различной физической природы.

Таблица 2.1

Моделируемая система	Фазовая переменная	
	потенциального типа	потокового типа
Электрическая	Электрическое напряжение	Электрический ток
Механическая	Скорость Угловая скорость Деформация	Сила Вращающий момент Механическое напряжение
Тепловая	Температура	Поток теплоты
Гидравлическая	Давление	Скорость потока (расход)

2.1.3. Топологические уравнения

При синтезе математической модели системы, состоящей из отдельных элементов, только компонентных уравнений недостаточно. Важно также знать, как эти элементы соединены между собой.

Вернемся к электрической системе (электрической цепи постоянного тока). Известно, что при последовательном соединении двух элементов ток в них одинаков, а общее напряжение равно сумме напряжений на каждом элементе. При параллельном соединении наоборот – напряжения на элементах равны, а общий ток равен сумме отдельных токов. Это вытекает из известных законов Кирхгофа.

Равенство нулю суммы токов в узлах схемы:

$$\sum_{k} \mathbf{i_k} = \mathbf{0}, \qquad (2.7a)$$

где $\mathbf{i_k}$ - ток **k**-й ветви

Равенство нулю суммы падений напряжения на элементах при обходе схемы по произвольному контуру:

$$\sum_{j} \mathbf{u_j} = \mathbf{0}, \qquad (2.7b)$$

где $\mathbf{u_j}$ – падение напряжения на **j**-й ветви

Аналогичные законы имеют место и для систем другой физической природы. Они известны (принцип Даламбера и принцип сложения скоростей в механике, уравнение неразрывности в гидродинамике и пр.).

Не приводя конкретных законов, укажем, что для систем различной природы действуют те же закономерности в поведении фазовых переменных при различном соединении элементов:

1. При последовательном соединении двух элементов системы фазовые переменные потокового типа в них равны, а переменные потенциального типа складываются, образуя суммарную величину.
2. При параллельном соединении элементов, составляющих систему, складываются фазовые переменные потокового типа, а потенциальные переменные равны.

Теперь синтез математической модели системы полностью возможен. Для этого система разбивается на элементы. Для каждого элемента устанавливаются соответствующие компонентные уравнения. Они дополняются топологическими уравнениями связи между элементами. Далее к модели могут быть добавлены источники фазовых переменных.

2.1.4. Примеры создания моделей технических объектов

Пример 2.1

В кабельную линию, обладающую сопротивлением **R** и емкостью **C,** подается прямоугольный импульс амплитудой **U**. Определить, через какое время амплитуда импульса на выходе линии достигнет 95% от **U** . Сопротивлением нагрузки пренебречь.

Решение.

Эквивалентная схема кабельной линии изображена на рисунке 2.1.

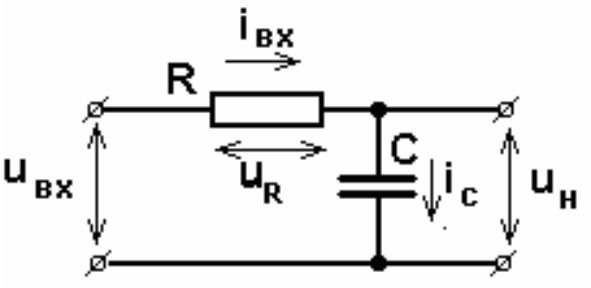


Рис. 2.1. Расчетная схема

Запишем компонентные уравнения:

$$\mathbf{i_R = u_R/R;}$$

$$\mathbf{i_c = C(du_C/dt).}$$

Добавим топологические уравнения:

$$\mathbf{u_{вх} = u_R + u_c\,;}$$

$$\mathbf{u_C = u_H\,;}$$

$$\mathbf{i_{вх} = i_R = i_c\,.}$$

Можно записать:

$$\mathbf{C\frac{du_н}{dt} = \frac{u_R}{R} = \frac{1}{R}(u_{вх} - u_н)}\,.$$

После преобразований получим обыкновенное дифференциальное уравнение первого порядка, составляющее математическую модель процесса передачи напряжения по заданной кабельной линии:

$$\mathbf{C\frac{du_н}{dt} + \frac{1}{R}u_н = \frac{1}{R}u_{вх}}\,.$$

Полученное уравнение можно решить при соответствующих начальных условиях: $\mathbf{u_{вх} = U}$, при $0 < t$ **.**

Решение уравнения:

$$\mathbf{u_н = U(1 - e^{-\frac{1}{RC}t})}\,.$$

При $\mathbf{u_н} = 0{,}95\mathbf{U}$ получим

$0{,}95U = U(1 - e^{-\frac{1}{RC}t})$ или $e^{-\frac{1}{RC}t} = 0{,}05$, далее $t = -RC \times \ln 0{,}05$.

Построим график, иллюстрирующие решения задачи (рис. 2.2).

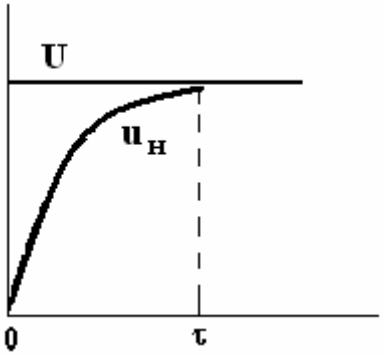


Рис. 2.2. Изменение напряжения импульса на выходе линии

Пример 2.2

При питании технических устройств сжатым воздухом от компрессора (особенно от поршневого) для сглаживания пульсаций давления устанавливают промежуточную емкость - ресивер. На рис. 2.3 показан ресивер (поз. 1), а эквивалент нагрузки обозначен пневматическим сопротивлением потребителя (поз. 2). Система работает в стационарном режиме с постоянным давлением на входе, равном P_o. В момент времени t_o на входе системы возникает импульс давления, который длится до момента времени t^* и имеет форму полусинусоиды. Требуется определить реакцию системы на данный импульс, то есть изменение давления на выходе ресивера P_1.

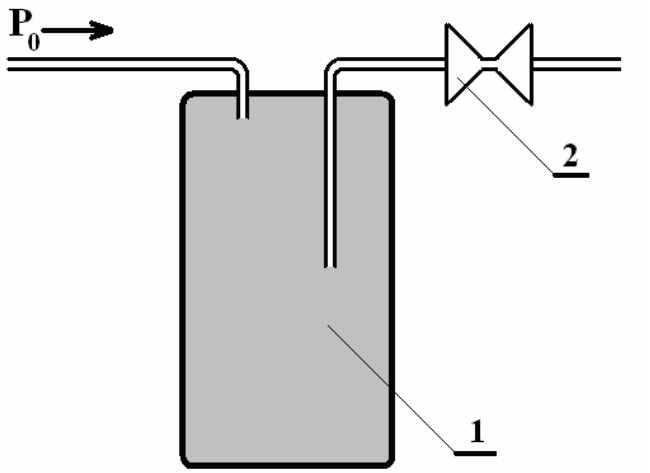


Рис. 2.3. Схема моделируемой системы

Решение.

Опишем движение газа через каждый элемент системы (компонентные уравнения). Используя уравнение (2.5b), запишем поток газа через емкость (ресивер):

$$\mathbf{G_1} = \frac{V\mathrm{r}}{\mathbf{k}_V}\frac{\mathbf{dP_1}}{\mathbf{d}t},$$

где V – объем ресивера; $\mathbf{k}_V$ - коэффициент объемного сжатия воздуха;
r - плотность воздуха; $\mathbf{P_1}$ – давление газа на выходе ресивера.

Поток газа через второй элемент опишется уравнением (2.5a):

$$\mathbf{G_2} = \frac{\mathbf{1}}{\mathbf{R_\Gamma}}\mathbf{P_2},$$

где $\mathbf{R_\Gamma}$ – сопротивление нагрузки; $\mathbf{P_2}$ – давление газа на нагрузке.

Добавим топологические уравнения:

$$\mathbf{G_1 = G_2;} \qquad \mathbf{P_1 + P_2 = P_0.}$$

С учетом записанного получим следующее уравнение:

$$\frac{V\mathrm{r}}{\mathbf{k}_V}\frac{\mathbf{dP_1}}{\mathbf{d}t} = \frac{\mathbf{1}}{\mathbf{R_\Gamma}}\mathbf{(P_0 - P_1)}.$$

Уравнение позволяет установить закон изменения давления на выходе ресивера $\mathbf{P_1}$ при известном законе изменения давления на входе в него $\mathbf{P}(t)$.

Начальные условия запишутся так:

$$\mathbf{P}(t) = \mathbf{P_1}(t) = \mathbf{P_0} \qquad \text{при } t < t_0;$$

$$\mathbf{P}(t) = \mathbf{a}\sin(\mathbf{b}t) \qquad \text{при } t_0 < t < t^*.$$

Построим графики, иллюстрирующие решения задачи, где изображены исходный $\mathbf{P}(t)$ и сглаженный $\mathbf{P_1}$ импульсы давления (рис. 2.4).

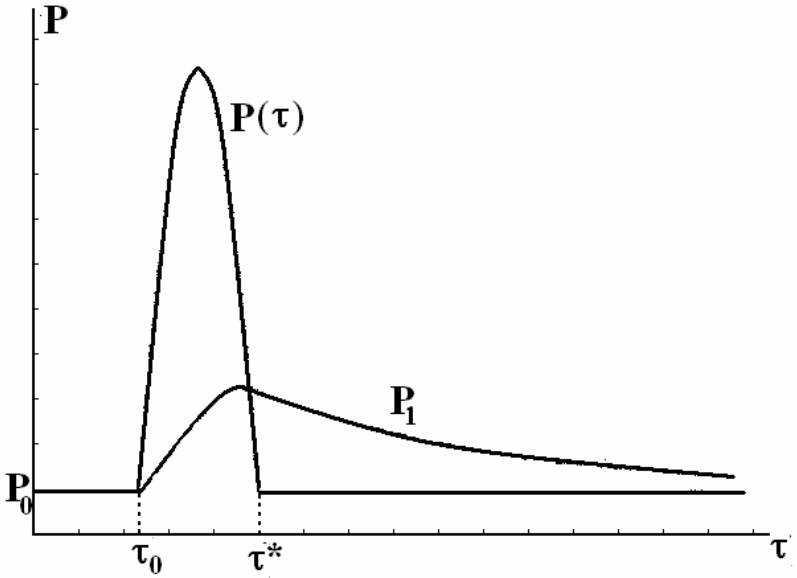


Рис. 2.4.Реакция системы на импульс давления

Пример 2.3

После включения электронного устройства микросхема, имеющая температуру окружающей среды $\mathbf{T_C}$, начинает нагреваться. При этом часть тепла отдается в окружающую среду. Определить изменение температуры микросхемы во времени, если выделение тепла происходит под действием источника постоянной мощности $\mathbf{N_T}$, а отдача тепла с поверхности пропорциональна разности температур с коэффициентом a.

Решение.

В этом примере мы имеем два параллельно протекающих процесса - нагрев тела постоянным источником тепла и охлаждение тела теплоотдачей в окружающую среду.

Запишем компонентные уравнения. Количество тепла, получаемое микросхемой от источника (процесс 1), можно определить по формуле (2.6c):

$$\frac{\mathbf{dQ_1}}{\mathbf{d}t} = \mathbf{N_T}.$$

Количество тепла, отдаваемое в окружающую среду (процесс 2):

$$\frac{\mathbf{dQ_2}}{\mathbf{d}t} = -\,a\mathbf{(T - T_C)},$$

где **T** – температура микросхемы.

Знак «минус» показывает, что данный поток теплоты направлен от микросхемы в окружающую среду.

Топологические уравнения для параллельных процессов имеют вид:

$$\mathbf{Q_{общ} = Q_1 + Q_2;} \qquad \mathbf{T_{общ} = T_1 = T_2}.$$

Отсюда имеем:

$$\frac{\mathbf{dQ}}{\mathbf{d}t} = \mathbf{N_T} - a\mathbf{(T - T_C)}.$$

С учетом (2.6b) можно переписать:

$$\frac{\mathbf{dT}}{\mathbf{d}t} = \frac{\mathbf{1}}{\mathbf{C_T m}}\left[\mathbf{N_T} - a\mathbf{(T - T_C)}\right].$$

Полученное дифференциальное уравнение следует дополнить начальными условиями: $\mathbf{T(}t\mathbf{) = T_C}$ при $t = \mathbf{0}$.

Решение задачи представлено на рис. 2.5.

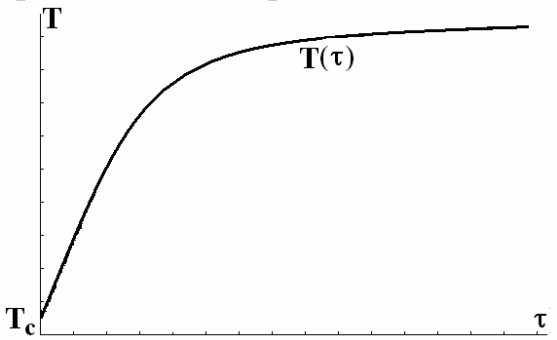


Рис. 2.5. Изменение температуры микросхемы во времени

Результаты показывают, что температура стремится к определенному предельному значению, при котором скорости процессов нагрева и охлаждения становятся равными.

2.1.5. Модели технологических аппаратов

При моделировании технологического оборудования часто применяют модели идеализированных потоков среды в аппарате. Наибольшее распространение получили следующие типовые модели.

Модель идеального смешивания

Согласно этой модели принимается, что субстанция (вещество или температура) распределена в аппарате равномерно. То есть входящая субстанция мгновенно распределяется по всему объему.

Схема модели идеального смешивания приведена на рис. 2.6

Зависимость между концентрациями на входе $\mathbf{C_0}$ и на выходе – **C** следующая:

$$\frac{\mathbf{dC}}{\mathbf{d}t} = \frac{\mathbf{1}}{t^*}(\mathbf{C_0} - \mathbf{C}), \qquad (2.8)$$

где $t^* = \mathbf{G_m/V}$, $\mathbf{G_m}$ – поток через аппарат; **V** – объем аппарата.

Изменение концентрации субстанции в аппарате при ее ступенчатом изменении на входе (реакция на ступенчатое возмущение) приведено на рис.2.7.

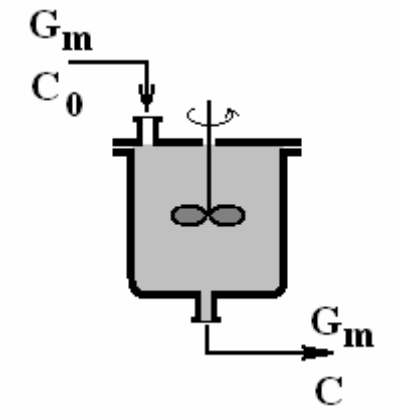


Рис.2.6. Схема модели идеального смешивания

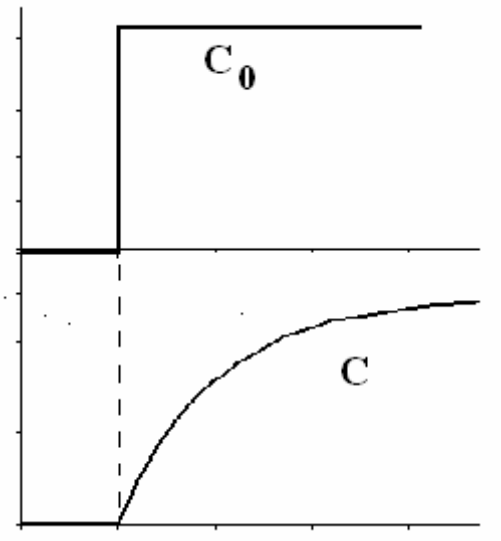


Рис. 2.7. Реакция модели

Модель идеального вытеснения

В соответствии с этой моделью принимается поршневое течение субстанции без перемешивания вдоль оси потока при равномерном распределении в направлении перпендикулярном движению.

Схема модели идеального вытеснения приведена на рис. 2.8.

Математическое описание модели в случае, когда ось координат x направлена вдоль оси аппарата, следующее:

$$\frac{\partial C}{\partial t} = -w\frac{\partial C}{\partial x}, \qquad (2.9)$$

где w = G_m/S ,а **S** – площадь сечения аппарата.

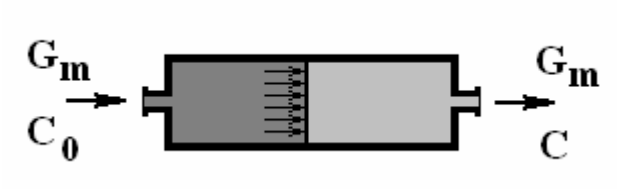


Рис. 2.8.Схема модели идеального вытеснения

Изменение концентрации субстанции в аппарате при ее ступенчатом изменении на входе (реакция на ступенчатое возмущение) приведено на рис.2.9.

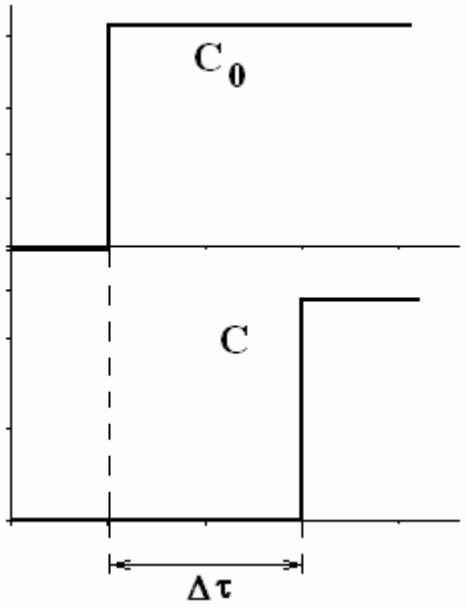


Рис. 2.9. Реакция модели

Ячеечная модель

Основой модели является представление об идеальном перемешивании в пределах ячеек, расположенных последовательно, и отсутствии перемешивания между ячейками. Схема ячеечной модели приведена на рис. 2.10. Параметром модели служит число ячеек **n**.

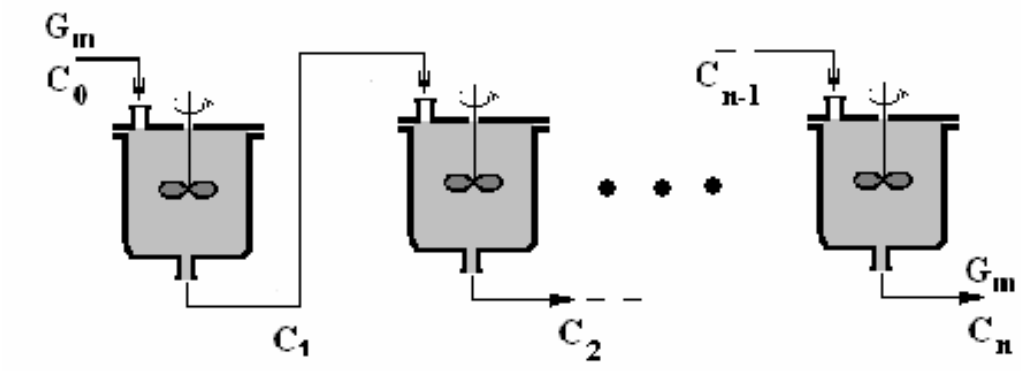


Рис. 2.10. Схема ячеечной модели

Математическое описание модели включает **n** линейных дифференциальных уравнений первого порядка.

$$\frac{1}{n}\frac{dC_i}{dt} = \frac{1}{t^*}(C_{i-1} - C_i), \qquad (2.10)$$

где **i** = 1, 2, . . . **n.**

При **n = 1** ячеечная модель переходит в модель идеального смешивания, а при **n = μ** – в модель идеального вытеснения. Выходные кривые ячеечной модели при ступенчатом возмущении представлены на рис. 2.11.

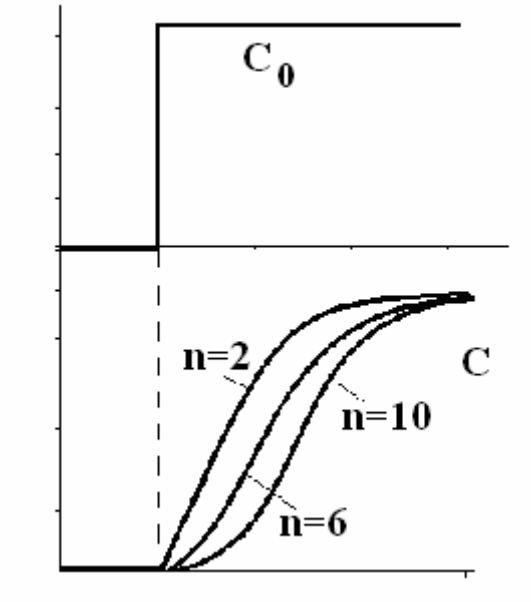


Рис. 2.11.Реакция модели

Заканчивая краткое рассмотрение непрерывно-детерминированных моделей (D-схем), следует указать, что они применяются при исследовании технических и технологических объектов и систем сравнительно давно. Известный и хорошо развитый математический аппарат, а также большое количество современных программных средств анализа и решения обыкновенных дифференциальных уравнений позволяют успешно применять эти модели в практике моделирования.

2.2. КОНЕЧНЫЕ АВТОМАТЫ

В качестве математических моделей элементов сложных систем часто используют модели с дискретным временем. Среди них исключительно важное место занимают автоматы.

Теория автоматов наиболее тесно связана с теорией алгоритмов. Это объясняется тем, что автомат преобразует дискретную информацию по шагам в дискретные моменты времени и формирует результирующую информацию по шагам заданного алгоритма. Эти преобразования возможны с помощью технических и (или) программных средств. Автомат можно представить как некоторое устройство (чёрный ящик), на которое подаются входные сигналы, снимаются выходные сигналы и которое может иметь определенные внутренние состояния.

2.2.1. Понятие конечного автомата

Рассмотрим общую математическую модель конечного автомата.

Введем понятие ***алфавит***, понимая под ним конечное множество объектов любой природы: символы, цифры, значки, рисунки, алгоритмы и т. д. В этом случае сами объекты можно называть ***буквами***. Конечную упорядоченную совокупность букв (не обязательно различных) следует назвать ***словом*** в данном алфавите.

Конечный автомат имеет один вход и один выход. Он представляет собой объект, функционирующий в дискретные моменты времени. В каждый момент времени t_i автомат находится в одном из возможных ***состояний*** $z(t_j)$ (число возможных состояний предполагается конечным). В каждый момент $t_i \in T$, начиная с t_1, на вход автомата поступает ***входной сигнал*** – одна из букв **x** входного алфавита **X**.

Автомат следующим образом реагирует на поступление входных сигналов.

Во-первых, состояние автомата изменяется в соответствии с одношаговой ***функцией переходов***:

$$z(t_i) = \varphi[z(t_{i-1}), x(t_i)]. \qquad (2.11)$$

Во-вторых, в каждый момент автоматного времени на выходе автомата появляется ***выходной сигнал*** $y(t_i)$ – буква выходного алфавита **Y**, определяемый ***функцией выходов:***

$$y(t_j) = \psi[z(t_{j-1}), x(t_j)]. \qquad (2.12)$$

Таким образом, конечный автомат можно определить как кортеж: $A=(X,Y,Z,\varphi,\psi)$, у которого:

1) $X=\{x_1,x_2,...,x_m\}$ - множество входных сигналов (входной алфавит);
2) $Y=\{y_1,y_2,...,y_n\}$ - множество выходных сигналов (выходной алфавит);
3) $Z=\{z_1,z_2,...,z_f\}$ - множество состояний (внутренний алфавит);
4) φ - функция переходов, которая некоторым парам «состояние - входной сигнал» ставит в соответствие новые состояния автомата;
5) ψ - функция выходов, которая некоторым парам «состояние - входной сигнал» ставит в соответствие выходные сигналы автомата.

Смысл работы автомата состоит в том, что он реализует некоторое отображение множества слов входного алфавита **X** во множество слов выходного алфавита **Y**. На уровне абстрактной теории понятие «работа автомата» понимается как преобразование входных слов в выходные. Можно сказать, что в данном случае мы отвлекаемся от внутренней структуры автомата и основное внимание уделяем поведению автомата относительно внешней среды.

2.2.2. Способы описания и классы конечных автоматов

Для задания конечного автомата необходимо описать все элементы множества $A=(X,Y,Z,\varphi,\psi)$, т.е. входной, выходной и внутренний алфавиты, а также функции переходов и выходов. При этом наиболее часто используются табличный, графический и матричный способ. Ниже эти способы будут рассмотрены подробнее.

На практике наибольшее распространение получили два класса автоматов - автоматы Мили (Mealy) и автоматы Мура (Moore).

Автомат Мили функционирует по схеме (2.11 – 2.12). То есть и состояние автомата, и выходной сигнал зависят от входного сигнала и предыдущего состояния.

Рассмотрим табличный способ задания автомата Мили с тремя состояниями, двумя входными и двумя выходными сигналами. При этом используем таблицы переходов и выходов, строки которых соответствуют входным сигналам автомата, а столбцы - его состояниям. На пересечении **i**-й строки и **j**-го столбца таблицы переходов (табл. 2.2) помещается соответствующее значение $j(z_k,x_i)$ функции переходов, а в таблице выходов (табл. 2.3) - $y(z_k, x_i)$ функции выходов.

Таблица 2.2

x	**z**		
	z_0	z_1	z_2
x_1	z_2	z_0	z_0
x_2	z_0	z_2	z_1

Таблица 2.3

x	**z**		
	z_0	z_1	z_2
x_1	y_1	y_1	y_2
x_2	y_1	y_2	y_1

При другом способе задания конечного автомата используется понятие направленного графа. Граф автомата представляет собой набор вершин, соответствующих различным состояниям автомата и соединяющих вершин дуг графа, соответствующих тем или иным переходам автомата. Если входной сигнал x_k вызывает переход из состояния z_i в состояние z_j, то на графе автомата дуга, соединяющая вершину z_i с вершиной z_j, обозначается x_k. Для того чтобы задать функцию выходов, дуги графа необходимо дополнительно отметить соответствующими выходными сигналами.

Граф автомата Мили, заданного таблицами 2.2 и 2.3, показан на рис. 2.12.

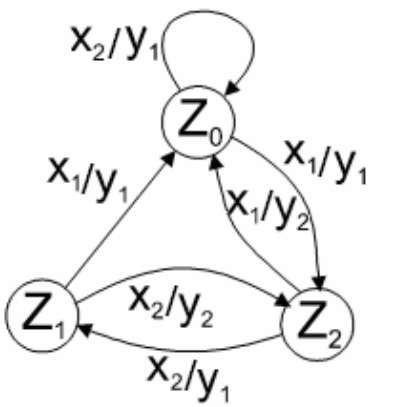


Рис. 2.12. Граф автомата Мили, соответствующий таблицам 2.2 и 2.3

При решении задач моделирования часто более удобной формой является матричное задание конечного автомата. При этом можно рассматривать две матрицы – матрицу переходов и матрицу выходов. Матрица переходов есть квадратная матрица $C=\| c_{ij} \|$, строки которой соответствуют исходным

состояниям, а столбцы - состояниям перехода. Элемент $\mathbf{c_{ij} = x_k}$ соответствует входному сигналу $\mathbf{x_k}$, вызывающему переход из состояния $\mathbf{z_i}$ в состояние $\mathbf{z_j}$. Если переход из состояния $\mathbf{z_i}$ в состояние $\mathbf{z_j}$ происходит под действием нескольких сигналов, элемент матрицы $\mathbf{c_{ij}}$ представляет собой множество входов для этого перехода, соединённых знаком дизъюнкции. Матрица выходов $\mathbf{D=\| d_{ij} \|}$строится аналогично, но ее элемент $\mathbf{d_{ij} = y_S}$ соответствует выходному сигналу $\mathbf{y_S}$, выдаваемому при переходе. Для рассматриваемого автомата Мили матрицы имеют вид:

$$\mathbf{C_1} = \begin{Vmatrix} \mathbf{x_2} & - & \mathbf{x_1} \\ \mathbf{x_1} & - & \mathbf{x_2} \\ \mathbf{x_1} & \mathbf{x_2} & - \end{Vmatrix} \qquad \mathbf{D_1} = \begin{Vmatrix} \mathbf{y_1} & - & \mathbf{y_1} \\ \mathbf{y_1} & - & \mathbf{y_2} \\ \mathbf{y_2} & \mathbf{y_1} & - \end{Vmatrix}$$

Автомат Мура функционирует несколько иначе.

Функция переходов у него имеет тот же вид, как и у автомата Мили, т.е. (2.11). Но функция выходов не зависит от входного сигнала, а является функцией только текущего состояния:

$$\mathbf{y(t_j) = y[\, z(t_j)].} \qquad (2.13)$$

Рассмотрим табличный способ задания автомата Мура с пятью состояниями, двумя входными и тремя выходными сигналами. Ниже приведены таблица переходов (табл. 2.4) и таблица выходов (табл. 2.5).

Таблица 2.4

x	**z**				
	z_0	**z_1**	**z_2**	**z_3**	**z_4**
x_1	z_1	z_4	z_4	z_2	z_2
x_2	z_3	z_1	z_1	z_0	z_0

Таблица 2.5

z_0	**z_1**	**z_2**	**z_3**	**z_4**
y_1	y_1	y_3	y_2	y_3

Графическое отображение автомата Мура практически аналогично отображению автомата Мили, однако значение выходного сигнала обычно размещается около соответствующей вершины.

Граф автомата Мура, заданного таблицами 2.4 и 2.5, показан на рис. 2.13.

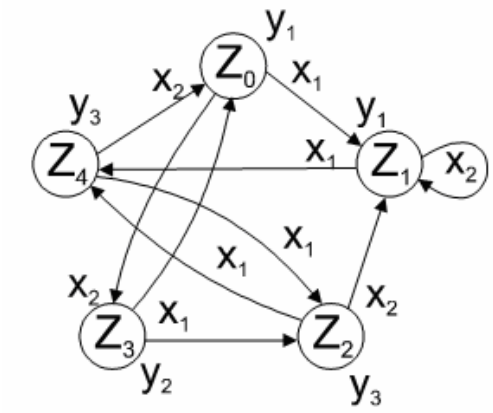


Рис. 2.13. Граф автомата Мура, соответствующий таблицам 2.4 и 2.5

При матричном задании конечного автомата Мура матрица переходов аналогична соответствующей матрице автомата Мили, а выход описывается вектором выходов. Для рассматриваемого автомата Мура эти объекты имеют вид:

$$\mathbf{C}_2 = \begin{Vmatrix} - & \mathbf{x}_1 & - & \mathbf{x}_2 & - \\ - & \mathbf{x}_2 & - & - & \mathbf{x}_1 \\ - & \mathbf{x}_2 & - & - & \mathbf{x}_1 \\ \mathbf{x}_2 & - & \mathbf{x}_1 & - & - \\ \mathbf{x}_2 & - & \mathbf{x}_1 & - & - \end{Vmatrix} \qquad \mathbf{D}_2 = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_1 \\ \mathbf{y}_3 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \end{pmatrix}$$

Для детерминированных автоматов переходы однозначны. Применительно к графическому способу задания автомата это означает, что в графе автомата из любой вершины не могут выходить 2 и более ребра, отмеченные одним и тем же входным сигналом. Аналогично этому в матрице переходов автомата **C** в каждой строке любой входной сигнал не должен встречаться более одного раза.

По характеру отсчёта времени конечные автоматы делятся на ***синхронные*** и ***асинхронные***. Автомат считается синхронным, когда моменты $\mathbf{t_0}$, $\mathbf{t_1}$ $\mathbf{t_2}$ (поступления входных сигналов, изменения состояний и выдачи выходных сигналов), определяются принудительно синхронизирующими сигналами (заранее определены). Реакция автомата на каждое значение входного сигнала заканчивается за один такт синхронизации. Асинхронные автоматы не имеют «жесткой» тактности; они изменяют свои состояния при поступлении входных сигналов, которые могут появляться в произвольные моменты времени из некоторого интервала.

Пример 2.4

Простейшим примером, который адекватно формализуется в виде конечного автомата, является автомат для продажи напитков. Допустим, автомат принимает монеты достоинством 1, 2 и 5 руб. и отпускает стакан напитка стоимостью 5 руб. Его можно представить, как конечный асинхронный автомат Мили с множеством состояний **Z={0, 1, 2, 3, 4}.** Входной алфавит автомата: **X={1, 2, 5**}, выходной алфавит **Y={0, 1},** где **0** соответствует ситуации «напиток не отпускается», а **1 –** ситуации «напиток отпускается».

Функция переходов j **(t)** определяется соотношением:

$$\mathbf{z(t_i) = mod[z(t_{i-1}) + x(t_i)],5,}$$

а функция выходов y – соотношением:

$$\mathbf{y(t_i)} = \begin{cases} \mathbf{0,} & \mathbf{\text{если}} \quad \mathbf{z(t_{i-1}) + x(t_i) \le 4,} \\ \mathbf{1,} & \mathbf{\text{если}} \quad \mathbf{z(t_{i-1}) + x(t_i) > 4} \end{cases}$$

Построим таблицы переходов и выходов рассматриваемого автомата.

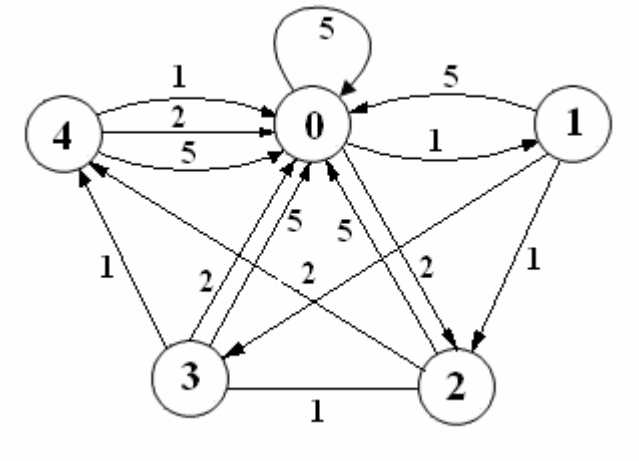
Таблица переходов

x	z				
	0	**1**	**2**	**3**	**4**
1	1	2	3	4	0
2	2	3	4	0	0
5	0	0	0	0	0

Таблица выходов

x	z				
	0	**1**	**2**	**3**	**4**
1	0	0	0	0	1
2	0	0	0	1	1
5	1	1	1	1	1

Граф автомата будет иметь вид:



Матричное задание автомата:

$$\mathbf{C_П} = \begin{Vmatrix} \mathbf{5} & \mathbf{1} & \mathbf{2} & - & - \\ \mathbf{5} & - & \mathbf{1} & \mathbf{2} & - \\ \mathbf{5} & - & - & \mathbf{1} & \mathbf{2} \\ \mathbf{2 Ú 5} & - & - & - & \mathbf{1} \\ \mathbf{1 Ú 2 Ú 5} & - & - & - & - \end{Vmatrix} \qquad \mathbf{D_П} = \begin{Vmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & - & - \\ \mathbf{1} & - & \mathbf{0} & \mathbf{0} & - \\ \mathbf{1} & - & - & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & - & - & - & \mathbf{0} \\ \mathbf{1} & - & - & - & - \end{Vmatrix}$$

2.2.3. Другие виды конечных автоматов

Конечный автомат, как типичная математическая схема для формального описания детерминированных объектов с дискретным временем, находит широкое применение. Необходимо отметить, что на практике, выполняя формальное описание некоторой детерминированной системы с дискретным временем приемами, характерными для конечных автоматов, в некоторых случаях можно прийти к модели, не являющейся, строго говоря, конечным автоматом.

Мы кратко рассмотрим два, достаточно часто встречающихся случая.

Автомат с последействием

Автомат с последействием – это объект $\mathbf{A=(X,Y,Z,}\,\varphi\,\mathbf{,}\psi\mathbf{,k)}$, определяемый следующими характеристиками:

Z – множество состояний; **X** – входной, а **Y** – выходной алфавиты; φ – одношаговая функция переходов; ψ – одношаговая функция выходов; **k** – натуральное число, называемое порядком начального множества.

Состояние автомата изменяется в соответствии с одношаговой ***функцией переходов*** так:

$$\mathbf{z(t_i) = \varphi\,\{[z(t_{i-k}), z(t_{i-k+1}),\ldots, z(t_{i-1})], x(t_i)\}} \qquad (2.14)$$

Функция выходов ψ аналогична рассмотренной ранее и определяется по формуле (2.12).

Здесь набор состояний $\mathbf{[z(t_{i-k}), z(t_{i-k+1}),.. . ., z(t_{i-1})]}$ называется предысторией автомата, а набор моментов времени $\mathbf{t_{i-k}, t_{i-k+1},. . ., t_{i-1}}$ – начальным множеством относительно момента времени $\mathbf{t_{i-1}}$.

Легко видеть, что при $\mathbf{k = 1}$ автомат с последействием превращается в обычный конечный автомат без последействия. При $\mathbf{k > 1}$ мы получаем модель более общую, чем конечный автомат.

Нестационарные автоматы

Другой тип детерминированных систем с дискретным временем, обобщающий понятие конечного автомата, - так называемые нестационарные автоматы. Обычный (стационарный) автомат $\mathbf{A=(X,Y,Z,}\,\varphi\,\mathbf{,}\psi\mathbf{)}$ имеет функции переходов и выходов, которые не зависят явно от времени **t**. Внимание к стационарным автоматам сложилось исторически как к моделям реальной аппаратуры, работающей в стационарных условиях. Более естественна, конечно, модель общего вида, когда функции переходов и выходов могут явно зависеть от времени:

$$\mathbf{z(t_i) = \varphi\,[t_{i-1}, z(t_{i-1}), x(t_i)],} \qquad (2.15)$$

$$\mathbf{y(t_j) = \psi\,[t_{j-1}, z(t_{j-1}), x(t_j)]}\,. \qquad (2.16)$$

В последнее время все чаще на практике встречается именно эта схема. Она относится к случаю непостоянства условий функционирования аппаратуры (изменение факторов внешней среды, строение технических средств, расхо-

дование ресурсов и т. д.). Кроме того, детерминированные системы с дискретным временем, как модели объектов материального мира, проникают в экономику, социологию, биологию и т. п. В связи с этим все чаще приходится иметь дело с объектами, не обладающими свойством стационарности.

Анализ нестационарных автоматов невозможен в рамках методов, развитых для стационарных автоматов. Одним из приемов изучения нестационарных автоматов может служить переход к стационарному автомату, соответствующему данному нестационарному. При этом чтобы функции j и y перестали явно зависеть от времени, его включают в состояние автомата как еще одну координату.

Полученный автомат хотя и стационарный, но он уже не является конечным, поскольку моментов времени бесконечное множество. Однако при моделировании систем на конечном интервале времени мы будем иметь дело с конечным числом моментов времени, и поэтому поведение соответственного стационарного автомата будет аналогично поведению обычного конечного автомата.

С помощью конечных автоматов (F-схем) описываются узлы и элементы ЭВМ, устройства контроля, регулирования и управления, системы временной и пространственной коммутации в технике обмена информацией. Однако широта применения F-схем не означает их универсальность. Этот подход непригоден для описания процессов принятия решений и процессов в динамических системах с наличием стохастических элементов.

3. СТОХАСТИЧЕСКИЕ МОДЕЛИ

Для формального описания сложных систем часто используют стохастические математические схемы, учитывающие действие случайных факторов. С этих позиций, рассмотренные выше детерминированные модели, игнорирующие случайные факторы, можно считать частным случаем более общих, стохастических моделей. Наибольшей популярностью при стохастическом моделировании пользуются следующие математические схемы: вероятностные автоматы (P-схемы) и системы массового обслуживания (Q-схемы). Причем первые позволяют исследовать процессы в дискретном времени, а вторые – в непрерывном.

Рассмотрение случайных влияний неизбежно приводит к необходимости использования вероятностных подходов и, в частности, аппарата теории марковских случайных процессов.

3.1. ЭЛЕМЕНТЫ ТЕОРИИ МАРКОВСКИХ СЛУЧАЙНЫХ ПРОЦЕССОВ

3.1.1. Понятие случайного процесса

При проектировании средств вычислительной техники широкое применение занимают марковские модели, используемые для анализа и синтеза вычислительных структур, которые можно рассматривать как стохастические системы без последствия.

Функционирование широкого класса систем можно представить как процесс перехода из одного состояния в другое под воздействием каких-либо причин. Например, процесс функционирования ЭВМ характеризуется тем, что в каждый момент времени обработкой информации заняты те или иные блоки. Процесс прохождения обрабатываемой информации по блокам ЭВМ можно рассматривать как процесс перехода системы из одного состояния в другое.

Пусть имеется некоторая физическая система **A**, которая в процессе функционирования может принимать различные состояния $\mathbf{A_i}$. Если состояния системы меняются во времени случайным образом, то процесс смены состояний можно рассматривать как случайный процесс, описываемый случайной функцией **X(b)**. Полное множество состояний $\mathbf{A_i}$ исследуемой системы может быть либо конечным $\mathbf{(\bar{i} = \overline{1,n})}$, либо бесконечно большим.

Большинство реальных систем имеют дискретное конечное пространство состояний. Последовательность состояний такой системы $\mathbf{A_i}$ $\mathbf{(i = \overline{1,n})}$ и сам процесс переходов из состояния в состояние называется ***цепью***. Ниже будут рассмотрены только случайные цепи.

В зависимости от времени пребывания системы в каждом состоянии различают процессы с дискретным или непрерывным временем. Системы с непрерывным временем предполагают, что переход системы из одного состояния в другое может осуществляться в любой момент времени, т. е. время пребывания системы в каждом состоянии представляет непрерывную случайную величину.

Для систем с дискретным временем время пребывания системы в каждом состоянии фиксированное, а моменты переходов $t_1, t_2, ..., t_k$ размещаются на временной оси через равные промежутки и называются «шагами». Время нахождения системы в некотором состоянии представляет дискретную случайную величину.

Таким образом, случайный процесс с непрерывными состояниями и непрерывным временем функционирования описывается непрерывной случайной функцией времени. Непрерывные и дискретные цепи описываются дискретными случайными функциями времени.

При исследовании непрерывных и дискретных случайных цепей обычно пользуются графическим представлением функционирования системы. Граф состояний системы представляет собой совокупность вершин, изображающих возможные состояния системы A_i и совокупность ветвей, изображающих возможные переходы системы из одного состояния в другое.

3.1.2. Дискретные цепи Маркова

Случайный процесс, протекающий в системе **A**, называется марковским или случайным процессом без последствия, если для любого момента времени t_0 вероятность любого состояния системы при $t > t_0$ зависит только от ее состояния при $t = t_0$ и не зависит от того, как и когда система пришла в это состояние. Если число состояний A_i, которые система может принимать, конечно, то такие системы описывает марковский случайный процесс с дискретными состояниями, или марковская цепь.

Пример марковского процесса счетчик в такси. Состояние счетчика в момент **t** характеризуется числом километров (десятых долей километров), пройденных автомобилем до данного момента. Пусть в момент t_0 счетчик показывает S_0. Вероятность того, что в момент $t_1 > t_0$ счетчик покажет то или иное число рублей S_1, зависит от S_0, но не зависит от того, в какие моменты времени изменялись показания счетчика до момента t_0.

Если переходы системы из одного состояния в другое возможны в строго определенные, заранее фиксированные моменты времени t_i, то такую систему описывает марковский случайный процесс с дискретным временем. марковский случайный процесс с дискретными состояниями и дискретным временем называют дискретной марковской цепью.

Обычно марковскую цепь изображают в виде графа, вершины которого соответствуют возможным состояниям системы A_i, а дуги - возможным переходам системы из состояния A_i ® A_j. Каждой дуге соответствует переходная вероятность $p_{ij}(k)=p[A_i^{(k)}/A_i^{(k-1)}]$ - это условная вероятность перехода системы на **k**-ом шаге в состояние A_j при условии, что на предыдущем
(k-1)-ом шаге система находилась в состоянии A_i.

Полным описанием марковской цепи служат матрицы переходных вероятностей:

$$\left|\mathbf{P_{ij}}\right| = \begin{vmatrix} \mathbf{p_{11}} & \mathbf{p_{12}} & \cdots & \mathbf{p_{1j}} & \cdots & \mathbf{p_{1n}} \\ \mathbf{p_{21}} & \mathbf{p_{22}} & \cdots & \mathbf{p_{2j}} & \cdots & \mathbf{p_{2n}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{p_{i1}} & \mathbf{p_{i2}} & \cdots & \mathbf{p_{1ij}} & \cdots & \mathbf{p_{in}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{p_{n1}} & \mathbf{p_{1n2}} & \cdots & \mathbf{p_{nj}} & \cdots & \mathbf{p_{nn}} \end{vmatrix} \qquad \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{p_{ij}} = \mathbf{1} \quad \mathbf{(i = \overline{1,n})} \qquad (3.1)$$

Кроме того, на каждом шаге марковская цепь характеризуется вектором вероятностей состояний:

$$\mathbf{|S(t)| = [S_1(t), S_2(t), \ldots S_n(t)].} \qquad (3.2)$$

Вероятностью **i**-го состояния называется вероятность $\mathbf{S_i(t)}$ того, что в момент **t** система будет находиться в состоянии $\mathbf{A_i}$. Очевидно, что для любого момента **t** сумма вероятностей всех состояний равна единице:

$$\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{S_i(t) = 1}. \qquad (3.3)$$

Марковская цепь называется ***однородной***, если переходные вероятности не зависят от номера шага. Если переходные вероятности меняются от шага к шагу, марковская цепь называется неоднородной.

Для ***неоднородной*** марковской цепи требуется **k** матриц, где **k** - число шагов.

Все многообразие марковских цепей подразделяется на ***эргодические*** и ***разложимые***.

Эргодические марковские цепи описываются сильно связанным графом. Это означает, что в такой системе возможен переход из любого состояния $\mathbf{A_i}$ в любое состояние $\mathbf{A_i}$ (i = 1..n) за конечное число шагов.

Разложимые марковские цепи содержат невозвратные состояния, называемые поглощающими. Из поглощающего состояния нельзя перейти ни в какое другое. На графе поглощающему состоянию соответствует вершина, из которой не выходит ни одна дуга.

Пример 3.1

Построить граф состояний и матрицу переходных вероятностей следующего случайного процесса:

Устройство **S** состоит из двух узлов, каждый из которых в случайный момент времени может выйти из строя, после чего мгновенно начинается ремонт узла, продолжающийся заранее неизвестное случайное время.

Решение:

Возможные состояния системы:

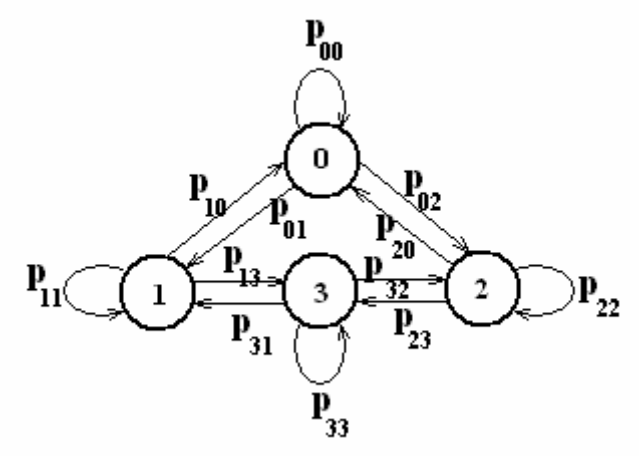
0 – оба узла исправны;

1 – первый узел ремонтируется, второй исправен;

2 – второй узел ремонтируется, первый исправен;

3 – оба узла ремонтируются.

Граф системы и матрица переходных вероятностей имеют вид:



$$|\mathbf{P}| = \begin{vmatrix} p_{00} & p_{01} & p_{02} & 0 \\ p_{10} & p_{11} & 0 & p_{13} \\ p_{20} & 0 & p_{22} & p_{23} \\ 0 & p_{31} & p_{32} & p_{33} \end{vmatrix}$$

Рис 3.1. Граф системы

При рассмотрении марковских цепей наиболее интересно выяснить, как будет изменяться вектор вероятностей состояний **|S(t)|,** когда система сделает один шаг. Для однородных цепей это можно сделать так.

Вернемся к примеру 3.1. Система может иметь 4 возможных состояния. Допустим, на (**k-1**)-м шаге вектор вероятностей состояний системы имел вид:

$$|\mathbf{S(t)}|^{(k-1)} = [S_0^{(k-1)}(t), S_1^{(k-1)}(t), S_2^{(k-1)}(t), S_3^{(k-1)}(t)].$$

Вероятность того, что на **k**-м шаге система будет находиться, например, в состоянии **1** можно вычислить по формуле полной вероятности:

$$S_1^{(k)}(t) = S_0^{(k-1)}(t)\, p_{01} + S_1^{(k-1)}(t)\, p_{11} + S_3^{(k-1)}(t)\, p_{31}\,.$$

Нетрудно заметить, что при вычислениях элементы вектора умножались на элементы соответствующего столбца матрицы переходных вероятностей. То же можно сделать и для других компонентов вектора состояний.

В общем виде можно записать:

$$S_j^{(k)}(t) = \sum_{i=1}^{n} S_i^{(k-1)}(t) \times p_{ij} \quad j = 1, \ldots n. \qquad (3.4)$$

Формула (3.4) позволяет последовательно шаг за шагом определять изменение элементов вектора вероятностей состояния, если известны элементы исходного вектора.

Операции по зависимости (3.4) представляют собой умножение вектора (матрицы-строки) состояний на матрицу переходных вероятностей, поэтому (3.4) можно переписать в матричном виде:

$$|\mathbf{S(t)}|^{(k)} = |\mathbf{S(t)}|^{(k-1)} \times \|\mathbf{P_{ij}}\|. \qquad (3.5)$$

В некоторых случаях при исследовании дискретных цепей Маркова возникает задача определения вероятностей состояния через **m** шагов. Безусловно, можно воспользоваться последовательным применением выражений (3.4) или (3.5), но это не всегда удобно. В ряде случаев для однородных цепей Маркова проще воспользоваться многошаговой матрицей переходных вероятностей $\|\mathbf{P_{ij}(m)}\|$. Она позволяет, используя выражения (3.4) или (3.5), определить элементы вектора вероятностей состояний через необходимое число шагов.

Для получения **m**-шаговой матрицы переходных вероятностей нужно вычислить **m**-ю степень обычной (одношаговой) матрицы:

$$||P_{ij}(m)|| = ||P_{ij}||^m . \qquad (3.6)$$

Выражение (3.6) является следствием известного соотношения, которое называется ***уравнение Колмогорова – Чепмена***:

$$||P_{ij}(m)|| = ||P_{ij}(s)||\ ||P_{ij}(m-s)||. \qquad (3.7)$$

3.1.3. Стационарное распределение вероятностей

Эргодическая цепь Маркова обладает очень важным свойством. При неограниченном увеличении числа шагов (**t** стремится к бесконечности) вероятности состояний системы **[S(t)]** стремятся к предельным значениям и перестают изменяться, т.е. зависеть от времени. Система приходит к стационарному режиму при котором **[S]** = **const**.

В теории случайных процессов доказывается, что если число состояний системы конечно и из каждого из них можно (за конечное число шагов) перейти в любое другое состояние, то стационарное состояние и предельные вероятности существуют.

Очень важным является то обстоятельство, что значения предельных вероятностей состояний не зависят от распределения вероятностей в начальный момент времени. То есть система может прийти к одному и тому же стационарному распределению вероятностей из любого исходного состояния.

Предельная вероятность состояния имеет четкий смысл: она показывает среднее относительное время пребывания системы в этом состоянии. Например, если предельная вероятность состояния равна 0,5, то это означает, что в среднем половину времени система находится в этом состоянии.

Для определения предельных вероятностей состояний следует решить систему линейных уравнений:

$$S_j = \sum_{i=1}^{n} S_i \times p_{ij}\ , \qquad j = 1, \ldots n \qquad (3.8)$$

Можно показать, что система (3.8) является линейно зависимой, поэтому для ее решения необходимо заменить любое из уравнений условием нормировки вероятностей:

$$\sum_{i=1}^{n} S_i = 1 .$$

Составить систему (3.8) довольно просто, используя граф цепи Маркова, если помнить следующее.

В левой части каждого уравнения стоит вероятность рассматриваемого состояния, а в правой части - столько слагаемых, сколько дуг графа входит в рассматриваемое состояние. Каждое слагаемое равно произведению вероятности перехода, соответствующей данной дуге графа, на вероятность того состояния, из которого исходит дуга графа. В частности, для примера 3.1 система

алгебраических уравнений для определения стационарных вероятностей будет иметь вид:

$$\begin{cases} S_0 = p_{00}S_0 + p_{10}S_1 + p_{20}S_2 \\ S_1 = p_{01}S_0 + p_{11}S_1 + p_{31}S_3 \\ S_2 = p_{02}S_0 + p_{22}S_2 + p_{32}S_3 \\ S_3 = p_{13}S_1 + p_{23}S_2 + p_{33}S_3 \end{cases}$$

Пример 3.2

Центральный процессор мультипрограммной компьютерной системы в любой момент времени выполняет либо приоритетную программу, либо фоновую программу, либо находится в состоянии ожидания. Продолжительность нахождения системы в каждом состоянии кратна длительности шага - D**t**. Определить коэффициент использования процессора, если заданы вероятности переходов из одного состояния в другое.

Решение:

Обозначим:

1 - состояние, в котором обслуживается приоритетная программа; **2** - состояние, в котором обслуживается фоновая программа; **3** - состояние простоя.

Пусть матрица переходных вероятностей имеет вид:

$$|p| = \begin{vmatrix} 0,7 & 0,2 & 0,1 \\ 0,8 & 0,1 & 0,1 \\ 0,8 & 0,05 & 0,15 \end{vmatrix}$$

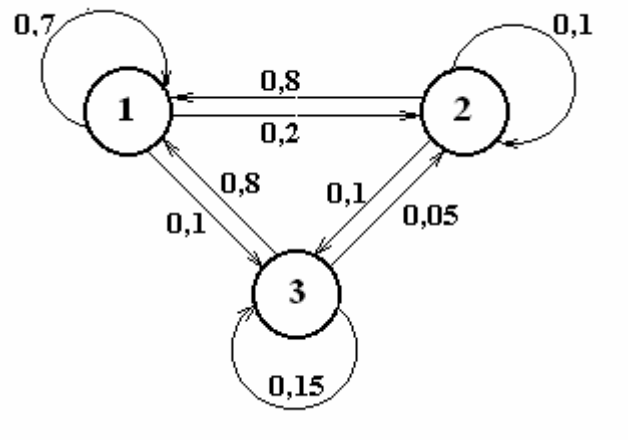


Рис.3.2. Граф системы

Изобразим граф функционирования системы.

Используя систему уравнений (3.8), составим для установившегося режима систему линейных алгебраических уравнений:

$$S_1 = 0,7S_1 + 0,8S_2 + 0,8S_3$$
$$S_2 = 0,2S_1 + 0,1S_2 + 0,05S_3$$
$$S_3 = 0,1S_1 + 0,1S_2 + 0,15S_3$$

Заменим любое из них на условие нормировки вероятностей $S_1+S_2+S_3=1$, а затем решим систему.

В результате решения получаем значение вероятностей состояний в установившемся режиме:

$S_1 = 0,73;$ $S_2 = 0,17;$ $S_3 = 0,1.$

Вероятность простоя процессора $S_3 = 0,1$.

Коэффициент использования процессора Ки = 1 - S_3 = 0,9, при этом на обслуживание приоритетной программы затрачивается 73% времени, а на обслуживание фоновой программы - 17%.

Разложимые марковские цепи, содержащие поглощающие невозвратные состояния, также могут иметь установившийся стационарный режим. В установившемся режиме поглощающему состоянию соответствует вероятность, равная 1.

3.1.4. Непрерывные марковские цепи

Непрерывные марковские цепи описывают функционирование систем, принимающих в процессе работы конечное число состояний $\mathbf{A_i}$ **(i = 1,..n)** и осуществляющих переходы из одного состояния в другое $\mathbf{A_i \rightarrow A_j}$ случайным образом в произвольный момент времени **t**. Иначе говоря, время пребывания системы в любом состоянии представляет непрерывную случайную величину.

Случайный процесс с непрерывным временем называется непрерывной марковской цепью, если поведение системы после произвольного момента времени $\mathbf{t_0}$ зависит только от состояния процесса в момент времени $\mathbf{t_0}$ и не зависит от предыстории процесса, предшествующей моменту времени $\mathbf{t_0}$.

При исследовании непрерывных марковских цепей целесообразно использовать не вероятности переходов между состояниями, а плотности этих вероятностей. Данные величины характеризуют интенсивность воздействий на систему, находящуюся в каком-либо состоянии, и часто называются интенсивностями перехода. В реальных задачах обычно это интенсивность внешних потоков случайных событий, переводящих марковский процесс в некоторое из возможных состояний.

Рассмотрим параметры, определяющие непрерывную марковскую цепь.

Пусть система в момент времени **t** находится в состоянии $\mathbf{A_i}$.

Рассмотрим элементарный промежуток времени D**t**, примыкающий к моменту времени **t**. За интервал D**t** система может перейти из состояния $\mathbf{A_i}$ в состояние $\mathbf{A_j}$ с переходной вероятностью $\mathbf{p_{ij}(t,}$ D**t** $\mathbf{)}$, зависящей в общем случае как от **t**, так и от D**t**.

Рассмотрим предел отношения этой переходной вероятности к ширине интервала D**t** при условии, что D**t** ® **0**:

$$\lim_{\Delta t \to 0} \frac{\mathbf{p_{ij}(t, \Delta t)}}{\Delta \mathbf{t}} = \mathbf{q_{ij}(t)}.$$

Эта характеристика и называется интенсивностью перехода или плотностью вероятности перехода.

Из приведенной формулы следует, что при малом D**t** вероятность перехода $\mathbf{p_{ij}(t,}$ D**t** $\mathbf{)}$ с точностью до бесконечно малых высших порядков равна

$$\mathbf{p_{ij}(t,} \Delta \mathbf{t)} = \mathbf{q_{ij}(t)} \Delta \mathbf{t}. \qquad (3.9)$$

Если плотности вероятностей переходов представляют собой функции времени $\mathbf{q_{ij}(t)}$, марковский процесс называется неоднородным.

Если все плотности вероятностей переходов не зависят от **t** (т.е. от начала отсчета элементарного участка D**t**), то марковский процесс называется однородным $\mathbf{q_{ij}(t) = q_{ij} = const}$.

Кроме интенсивностей переходов, для описания непрерывных марковских цепей должен быть задан вектор вероятностей состояний системы в исходный (нулевой) момент времени

$$\mathbf{[S(0)] = [S_1(0), S_2(0), ..., S_n(0)].}$$

3.1.5. Уравнения А.Н. Колмогорова

Распределение вероятностей состояний $|\mathbf{S(t)}|$ для марковской цепи с непрерывным временем находится из решения системы дифференциальных уравнений для вероятностей состояний, которая носит имя ее автора – российского математика А.Н. Колмогорова.

Общий вид системы уравнений Колмогорова:

$$\frac{\mathbf{dS_i(t)}}{\mathbf{dt}} = -\mathbf{S_i(t)} \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{q_{ji}} + \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{S_j(t) q_{ij}} \qquad \mathbf{j \neq i,} \qquad (3.10)$$

$\mathbf{S_i(0)}$ - вектор начальных условий $\mathbf{(i = \overline{1,n})}$.

Интегрирование этой системы по времени позволяет получить вероятности состояний как ***функции времени*** $\mathbf{S_i(t)}$.

Сформулируем правило составления уравнений Колмогорова с использованием размеченного графа цепи.

В левой части каждого уравнения стоит производная от вероятности рассматриваемого состояния по времени, а в правой части - столько слагаемых, сколько дуг графа связывает рассматриваемое состояние с другими состояниями. Каждое слагаемое равно произведению интенсивности перехода, соответствующей данной дуге графа, на вероятность того состояния, из которого исходит дуга графа. Если стрелка направлена из рассматриваемого состояния, соответствующее произведение имеет знак минус. Если стрелка направлена в состояние, то произведение имеет знак плюс.

Существенно, что в системе уравнений Колмогорова можно ограничиться **n - 1** уравнением. Дополнительно используется условие нормировки:

$$\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{S_i(t) = 1}.$$

Для определения предельных вероятностей $\mathbf{S_i}$ при нахождении системы в стационарном состоянии нужно составить систему **n** линейных однородных алгебраических уравнений с **n** неизвестными. Такую систему легко получить, положив в системе уравнений (3.10) $\mathbf{dS_i/dt = 0.}$

В общем виде система уравнений Колмогорова для стационарного состояния имеет вид:

$$\mathbf{S_i} \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{g_{ji}} = \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{S_j g_{ij}}, \qquad \mathbf{j \neq i.} \qquad (3.11)$$

В системе (3.11) независимых уравнений на единицу меньше общего числа уравнений. Поэтому любое уравнение системы (3.11) следует заменить уравнением нормировки вероятностей.

Как и для дискретных марковских цепей, предельные вероятности характеризуют среднюю долю времени, в течение которого система находится в данном состоянии при наблюдении за системой в течение достаточно продолжительного времени (на бесконечном интервале).

Пример 3.3

Двухпроцессорная вычислительная система предназначена для обработки простейшего потока задач, поступающих с интенсивностью λ. Интенсивность решения задач первым процессором равна μ_1, вторым – μ_2, причем $\mu_1 > \mu_2$.

Задача в первую очередь принимается на обслуживание процессором, имеющим большую производительность. Если оба процессора заняты, пользователь получает отказ. Определить в установившемся режиме вероятность отказа $\mathbf{p_{отк}}$, коэффициенты загрузки процессоров $\mathbf{KSI_1}$, $\mathbf{KSI_2}$.

Решение:

Рассмотрим возможные состояния системы, которые определяются состояниями процессоров:

00 - оба процессора простаивают; **10** - первый процессор занят решением задач, второй простаивает; **01** - второй процессор занят, первый простаивает; **11** - оба процессора заняты решением задач.

Граф функционирования системы имеет вид:

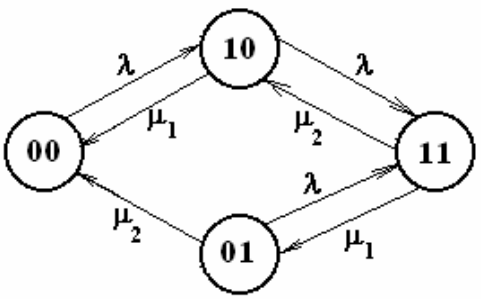


Рис. 3.3. Граф функционирования системы

По графу запишем систему линейных дифференциальных уравнений А.Н.Колмогорова.

$$\mathbf{dS_{00}(t)/dt = -\lambda\, S_{00}(t) + \mu_1\, S_{10}(t) + \mu_2\, S_{01}(t);}$$
$$\mathbf{dS_{10}(t)/dt = \lambda\, S_{00}(t) - (\mu_1 + \lambda) S_{10}(t) + \mu_2\, S_{11}(t);}$$
$$\mathbf{dS_{01}(t)/dt = -(\lambda + \mu_2)\, S_{01}(t) + \mu_1\, S_{11}(t);}$$
$$\mathbf{dS_{11}(t)/dt = \lambda S_{10}(t) + \lambda S_{01}(t) - (\mu_1 + \mu_2) S_{11}(t);}$$
$$\mathbf{S_{00}(t) + S_{10}(t) + S_{01}(t) + S_{11}(t) = 1.}$$

Пусть начальные условия заданы вектором вероятностей:

$$\mathbf{S_{00}(0) = 1;\quad S_{10}(0) = 0;\quad S_{01}(0) = 0;\quad S_{11}(0) = 0.}$$

Решение этой системы при заданных начальных условиях позволяет определить вероятности состояний как функции времени. Последние, в свою

очередь, позволяют определить требуемые характеристики вычислительной системы.

Поскольку марковский процесс, описывающий работу вычислительной системы, является эргодическим, существует стационарный режим, при котором вероятности состояний стремятся к постоянным величинам. При этом система дифференциальных уравнений Колмогорова вырождается в систему линейных уравнений:

$$\begin{cases} -\lambda S_{00} + \mu_1 S_{10} + \mu_2 S_{01} = 0, \\ \lambda S_{00} - (\mu_1 + \lambda) S_{10} + \mu_2 S_{11} = 0, \\ -(\lambda + \mu_2) S_{01} + \mu_1 S_{11} = 0, \\ S_{00} + S_{10} + S_{01} + S_{11} = 1. \end{cases}$$

Последнее уравнение системы заменено условием нормировки вероятностей.

Вероятность отказа совпадает с вероятностью состояния, в котором оба процессора заняты, т. е. $p_{отк} = S_{11}$. Коэффициенты загрузки процессоров KSI_i представляют собой вероятности пребывания соответствующих процессоров в занятом состоянии:

$$KSI_1 = S_{10} + S_{11}; \qquad KSI_2 = S_{01} + S_{11}.$$

3.1.6. Потоки событий

Переход системы в некоторое состояние A_i называется событием. В процессе работы система неоднократно может переходить из состояния в состояние. Последовательность таких однородных событий образует поток событий.

Поток событий удобно отображать в виде отметок на оси времени, соответствующих моментам наступления событий.

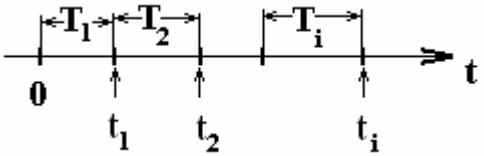


Рис. 3.4. Моменты наступления событий

На рис. 3.4. отмечены моменты наступления t_i событий и интервалы между соседними событиями T_i.

Если интервалы являются неслучайными, то поток называется ***регулярным*** или ***детерминированным*** и полностью характеризуется законом изменения длины интервалов в потоке. В противном случае поток называется ***случайным*** и характеризуется совместным законом распределения системы случайных величин $(T_1, T_2, ..., T_n)$.

На практике наиболее часто приходится иметь дело с потоками, в которых интервалы времени между двумя соседними событиями T_i - непрерывные случайные величины.

Потоки событий имеют такие свойства, как стационарность, ординарность.

Поток называется ***стационарным***, если его характеристики не изменяются во времени. Вероятность попадания того или иного числа **m** событий на участок оси времени **[t , t+**t**]** зависит только от длины участка t и не зависит от **t**. Интенсивность или плотность потока событий, то есть среднее число событий в единицу времени, постоянна: l **= const**.

Поток называется ***ординарным***, если события в нем происходят поодиночке. Поток событий можно назвать ординарным тогда, когда вероятность наступления двух и более событий за любой бесконечно малый промежуток времени D**t** есть величина бесконечно малая более высокого порядка, чем D**t** .

Еще одним свойством случайных потоков событий является наличие или отсутствие последействия.

Случайный поток событий называется ***потоком без последействия***, если для любых непересекающихся участков времени число событий, попадающих на один из них, не зависит от того, сколько событий попало на другие участки. Условие отсутствия последействия означает, что события наступают в системе независимо друг от друга

Если вероятность наступления события зависит от предыдущих событий (их числа, моментов наступления, длительности интервалов между ними) поток называется ***потоком с последействием***. Здесь для любого момента времени последующее течение потока находится в вероятностной зависимости от предыдущего. Предельным случаем такого потока является жесткая статистическая связь между моментами наступления событий, то есть поток становится детерминированным.

На практике встречаются случаи, когда случайные величины $\mathbf{T_i}$ (интервалы между событиями) являются независимыми, но наблюдается зависимость вероятности наступления событий, допустим от числа предыдущих событий. В таком случае случайный поток называется потоком ***с ограниченным последействием.***

<u>Простейший поток</u>

Если число **m** событий потока, попадающих на участок t, распределено по закону Пуассона, то такой поток называется ***пуассоновским***,

$$\mathbf{p_m = a^m/m!\, e^{-a}},$$

где **a** - среднее число событий, попадающих на участок t**.**

Для стационарного потока **a =** l t, где l - интенсивность потока, характеризующая среднее число событий в единицу времени.

Стационарный пуассоновский поток является примером случайного потока без последействия. Для него интервал времени от начала отсчета до наступления первого события представляет собой непрерывную случайную величину $\mathbf{T_i}$, распределенную по экспоненциальному закону

$$\mathbf{f(t) = l e^{-l t}}. \qquad (3.12)$$

Стационарный пуассоновский поток событий, обладающий свойствами ординарности, стационарности и отсутствия последействия, называется ***простейшим потоком***.

Для простейшего потока выражения для математического ожидания и дисперсии имеют вид:

$$\mathbf{M(T_i) = 1/\lambda\ ;} \qquad \mathbf{D(T_i) = 1/\lambda^2\ .} \tag{3.13}$$

Если процесс переходов в системе происходит под воздействием простейшего потока, то такой процесс является марковским, причем плотность вероятности перехода в соответствующей непрерывной марковской цепи совпадает с интенсивностью потока переходов λ.

<u>Потоки с ограниченным последействием</u>

Примерами потоков с ограниченным последействием являются потоки Эрланга. Они образуются путем закономерного просеивания простейшего потока. Под закономерным просеиванием будем понимать такую процедуру, в результате которой, безусловно, исключается некоторая последовательность событий в исходном потоке.

Если в исходном простейшем потоке каждое k-е событие сохранить, а (k - 1) событий исключить, то получим поток Эрланга k-го порядка.

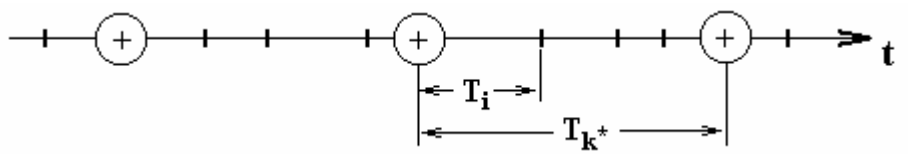


Рис. 3.5. Получение потока 4-го порядка Эрланга 4-го порядка просеиванием простейшего потока

Случайная величина T_{k^*} интервала между соседними событиями потока Эрланга k-го порядка представляет сумму k независимых случайных величин, подчиненных показательному закону распределения:

$$\mathbf{T_{k^*}} = \sum_{i=1}^{k} \mathbf{T_i}\ . \tag{3.14}$$

Плотность распределения имеет вид:

$$\mathbf{f_k(t_k)} = \frac{\lambda(\lambda t_k)^{k-1}}{(k-1)!}\mathbf{e}^{-\lambda t_k}\ . \tag{3.15}$$

Обычно, для сохранения масштаба времени случайную величину T_k^* нормируют коэффициентом k, т. е.

$$\mathbf{T_{kн} = T_{k^*}/k.} \tag{3.16}$$

Для нормированного потока Эрланга k-го порядка математическое ожидание и дисперсия рассчитываются так:

$$\mathbf{M(T_{kн}) = 1/\lambda;} \qquad \mathbf{D(T_{kн}) = 1/(\lambda k)^2\ .} \tag{3.17}$$

Таким образом, при неограниченном увеличении порядка k нормированный поток Эрланга приближается к регулярному потоку с постоянными интервалами, равными $1/\lambda$

Нормированный поток Эрланга в зависимости от порядка k позволяет получить любую степень последействия, от полного отсутствия ($k = 1$) до жесткой статистической связи ($k = \infty$). Благодаря этому реальный поток событий с последействием можно в некоторых случаях аппроксимировать нормированным потоком Эрланга соответствующего порядка, имеющим примерно те же математическое ожидания и дисперсию, что находит широкое применение при моделировании произвольных потоков.

3.2. ОСНОВЫ ТЕОРИИ МАССОВОГО ОБСЛУЖИВАНИЯ

Теория массового обслуживания изучает системы массового обслуживания (СМО) и сети массового обслуживания (СеМО). Рассматриваемые в этом разделе модели и математические схемы относятся к классу непрерывно-стохастических, т.е. это Q – схемы.

Под системой массового обслуживания (СМО) понимают динамическую систему, предназначенную для эффективного обслуживания потока заявок (требований на обслуживание) при ограничениях на ресурсы системы. Модели СМО удобны для описания отдельных подсистем современных вычислительных систем, таких как подсистема процессор - основная память, канал ввода - вывода и т. д.

Вычислительная система в целом представляет собой совокупность взаимосвязанных подсистем, взаимодействие которых носит вероятностный характер. Заявка на решение некоторой задачи, поступающая в вычислительную систему, проходит последовательность этапов счета, обращения к внешним запоминающим устройствам и устройствам ввода - вывода. После выполнения некоторой последовательности таких этапов, число и продолжительность которых зависят от трудоемкости программы, заявка считается обслуженной и покидает вычислительную систему. Таким образом, вычислительную систему в целом можно представлять совокупностью СМО, каждая из которых отображает процесс функционирования отдельного устройства или группы однотипных устройств, входящих в состав системы.

Совокупность взаимосвязанных СМО называется сетью массового обслуживания (стохастической сетью).

Основными задачами, которые решаются в рамках теории массового обслуживания, являются задачи:

- анализа, т. е. определение количественных характеристик СМО и СеМО при заданной структуре и заданных параметрах элементов структуры;
- синтеза оптимальной структуры СМО или СеМО при заданных характеристиках и ограничениях на параметры элементов структуры.

3.2.1. Обобщенная структурная схема СМО. Параметры и характеристики

При исследовании СМО предполагаются известными некоторые их свойства, т. н. параметры СМО. В результате исследования определяются характеристики СМО, являющиеся функцией параметров.

На вход СМО поступают заявки на обслуживание, образующие входящий поток. Первопричину заявок, какова бы ни была ее физическая природа, называют источником заявок. В зависимости от характера источника заявок различают разомкнутые и замкнутые СМО. В разомкнутых СМО число заявок, вырабатываемых источником, считается неограниченным. Поведение источника заявок не связано с состоянием СМО ни в данный, ни в какой-либо из предшествующих моментов времени. Для замкнутых СМО характерно конечное число заявок, циркулирующих в системе источник - СМО. Обслуженные заявки возвращаются в источник и через некоторое, в общем случае, время могут вновь появиться на входе СМО. Поведение источника в замкнутых СМО является некоторой функцией состояния СМО.

Рассмотрим обобщенную структурную схему разомкнутой СМО (рис.3.6), примером которой является многопроцессорная вычислительная система (ВС).

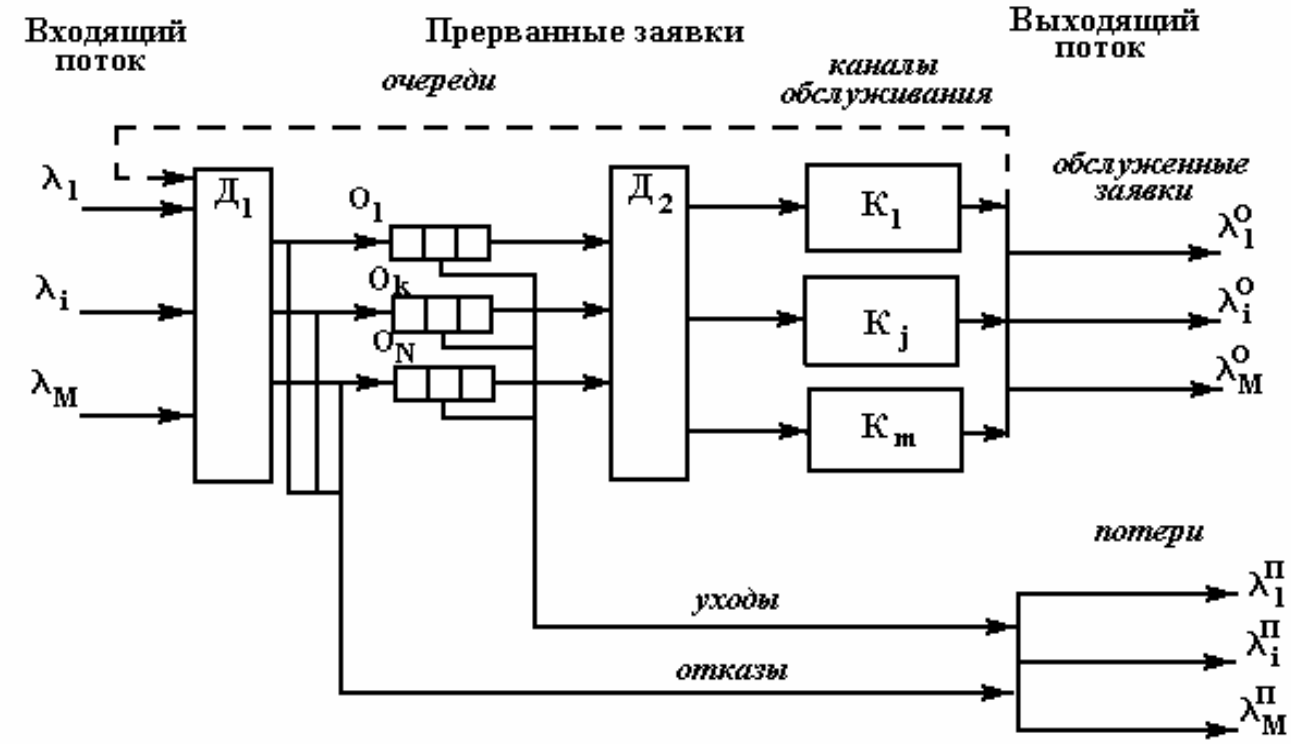


Рис. 3.6. Обобщенная структурная схема СМО

Помимо аппаратных средств (процессоры, память, устройства прерывания, периферийные устройства) в состав ВС входят программные средства, содержащие прикладные и системные управляющие программы. Прикладные управляющие программы реализуют алгоритмы обработки информации, их ис-

полнение процессорами рассматривается как обслуживание заявок, поступающих в ВС. Системные управляющие программы осуществляют управление прохождением заявок через ВС (диспетчирование) и исполняются, как правило, одним из процессоров. Обычно выделяют две основные системные управляющие программы: «Диспетчер 1» (Д1) и «Диспетчер 2» (Д2), реализующие, соответственно, дисциплины ожидания и обслуживания.

Система работает следующим образом.

Появление на входе СМО заявки инициирует выполнение программы Д1, которая распознает приоритет поступившей заявки и ставит ее в соответствующую очередь $\mathbf{O_i}$, реализованную в специально зарезервированной области памяти. Д2 анализирует состояние очередей $\mathbf{O_1, O_2, ..., O_N}$ и состояние каналов $\mathbf{K_1, K_2, ..., K_m}$ (процессоров). При наличии свободного канала Д2 выбирает заявку, имеющую преимущественное право на обслуживание, и инициирует соответствующий канал и необходимую прикладную программу. Инициирование происходит в моменты окончания исполнения прикладных программ и программы Д1.

Параметры входящего потока

Процесс поступления в СМО заявок на обслуживание является в общем случае случайным и может рассматриваться как поток однородных событий, происходящих через случайные промежутки времени. Случайные временные интервалы между поступлениями заявок могут подчиняться различным законам распределения. Наибольшее распространение в теории массового обслуживания получил простейший поток заявок, то есть поток, в котором интервал времени между двумя соседними заявками подчинен экспоненциальному закону распределения с интенсивностью λ:

$$\mathbf{f}(t) = \lambda \mathbf{e}^{(-\lambda t)} \quad . \qquad (3.18)$$

Привлекательность простейшего потока объясняется рядом обстоятельств. Допущение о простейшем потоке заявок позволяет получать аналитические зависимости характеристик СМО от параметров входящего потока, что затруднительно для других видов потока заявок.

Простейший поток в теории массового обслуживания играет такую же роль, как нормальный закон распределения случайных величин в теории вероятностей: при сложении нескольких независимых, ординарных, стационарных случайных потоков образуется суммарный поток, приближающийся по своим свойствам к простейшему.

Если СМО обеспечивает желаемую эффективность функционирования системы при простейшем потоке заявок на входе, то обслуживание системой других случайных потоков заявок с одинаковой интенсивностью будет выполняться не хуже.

Если входящий поток представляет собой совокупность M потоков заявок различных типов с интенсивностями λ_i, ($\mathbf{i = \overline{1, M}}$), то его можно характеризовать суммарной интенсивностью

$$\lambda = \sum_{i=1}^{M} \lambda_i . \qquad (3.19)$$

Степень важности заявок может быть различной. По этому признаку заявки делят на классы, каждому классу присваивается приоритет К ($\mathbf{K} = \overline{\mathbf{1,N}}$), причем наивысшим приоритетом обладают заявки первого класса, с увеличением К приоритет заявки падает.

Различают «терпеливые» заявки, т. е. такие, на время пребывания которых в СМО не накладывается никаких ограничений, и «нетерпеливые», способные уйти из системы, не будучи обслуженными, если время пребывания их в СМО превысит допустимую величину.

Параметры структуры СМО

Каждая система массового обслуживания обладает определенной структурой, характеризующейся совокупностью параметров. Основным компонентом структуры СМО являются каналы обслуживания. В зависимости от числа каналов различают одноканальные и многоканальные СМО. В свою очередь, многоканальные СМО могут содержать одинаковые и различные по производительности каналы обслуживания.

Производительность канала обслуживания обратна длительности обслуживания заявки, равной промежутку времени, необходимому каналу обслуживания для обслуживания заявки. В общем случае это случайная величина с функцией распределения $\mathbf{F(t_{об})}$, плотностью распределения $\mathbf{f(t_{об})}$ и математическим ожиданием $\overline{t_{об}}$.

Типы заявок различаются либо законами распределения, либо только математическими ожиданиями при одинаковых законах распределения. При этом принимается допущение о независимости длительностей обслуживания для различных заявок одного типа, вполне корректное для большинства реальных систем. Наряду с математическим ожиданием длительности обслуживания используется понятие интенсивности потока обслуживания $\mu = 1/\tau_{об}$ - величины, обратной средней длительности обслуживания и характеризующей количество заявок, которое может быть обслужено в единицу времени постоянно загруженным каналом обслуживания. При моделировании СМО наиболее часто используют длительность обслуживания с экспоненциальной плотностью распределения.

$$\mathbf{f(t_{об})} = \mu \mathbf{e}^{(-\mu t_{об})}. \qquad (3.20)$$

Если в момент появления заявки на входе СМО хотя бы один канал свободен от обслуживания, ее обслуживание может быть начато немедленно, без задержки. Однако вполне вероятна ситуация, когда заявка застает СМО полностью загруженной, то есть когда все m каналов обслуживания заняты обслуживанием. В этом случае начало обслуживания задерживается, заявка может занять место в соответствующей очереди. Таким образом, вторым важным компонентом структуры СМО является очередь, параметром которой является число мест в очереди n. В приоритетных системах общая очередь может быть

разделена на несколько очередей по числу различаемых системой приоритетов, для каждой из которых должно быть указано число мест n_i,($\mathbf{i = \overline{1, N}}$). На число мест в очереди часто накладывается ограничение. Это может быть сделано как для каждой очереди в отдельности, так и для всей совокупности очередей в целом. При этом возможны конфликтные ситуации, решением которых является отказ системы принять заявку.

В зависимости от числа мест в очереди различают СМО с отказами и, соответственно, СМО без отказов. В СМО с отказами число мест в очереди конечно и вследствие вероятностного характера как входящего потока, так и процессов обслуживания, существует ненулевая вероятность того, что поступившая на вход СМО заявка застанет все каналы занятыми обслуживанием и все места в очереди занятыми ожидающими обслуживания заявками, то есть она получит отказ. В СМО без отказов заявка либо сразу назначается на обслуживание, если в момент ее поступления свободен хотя бы один канал обслуживания, либо безусловно принимается в очередь на обслуживание.

Параметры законов управления процессами в СМО

Процесс продвижения заявки от входа к выходу СМО происходит в соответствии с некоторым законом управления процессами в СМО, который задается дисциплинами ожидания и обслуживания. Дисциплина ожидания определяет порядок приема заявок в систему и размещения их в очереди, дисциплина обслуживания – порядок выбора заявок из очереди для назначения на обслуживание.

В зависимости от принятых в СМО дисциплин ожидания и обслуживания различают СМО с бесприоритетными и приоритетными дисциплинами.

В СМО с бесприоритетными дисциплинами все заявки считаются равноправными. Возможны следующие бесприоритетные дисциплины обслуживания, то есть правила выборки заявки из очереди при необходимости назначения на обслуживание:

- выбирается первая в очереди заявка - дисциплина «первым пришел - первым вышел» (FIFO - First Input First Output);
- выбирается последняя в очереди заявка – дисциплина «последним пришел - первым вышел» (LIFO - Last Input First Output);
- заявка выбирается из очереди случайным образом.

В приоритетных дисциплинах обслуживания заявкам некоторых типов представляется преимущественное право на обслуживание перед заявками других типов, называемое приоритетом. Различают относительные, абсолютные и смешанные приоритеты.

Относительные приоритеты учитываются только в момент назначения заявки на обслуживание. При освобождении канала обслуживания сравниваются приоритеты заявок, находящихся в очереди в состоянии ожидания, и обслуживание предоставляется заявке с наибольшим приоритетом, после чего выбранная заявка захватывает канал обслуживания.

Абсолютные приоритеты предполагают прерывание обслуживания низкоприоритетной заявки в момент поступления в СМО заявки с более высоким приоритетом, прерванная заявка ставится в начало либо общей очереди, либо очереди заявок соответствующего приоритета.

Обслуживание прерванных заявок может проводиться либо от начала (повторное обслуживание), либо от момента прерывания (дообслуживание), чаще используют второй способ – дообслуживание прерванных заявок.

Смешанные приоритеты предполагают сочетание рассмотренных видов приоритета, причем для отдельных заявок может быть использовано бесприоритетное обслуживание.

Совокупность обслуженных и потерянных заявок образует выходящий поток СМО. В зависимости от структуры выходящего потока различают СМО без потерь («чистые» СМО) и СМО с потерями («смешанные» СМО). Для «чистых» СМО характерно отсутствие ограничений на число мест в очереди (бесконечная очередь) и на время пребывания заявки в системе («терпеливые» заявки). По этой причине выходящий поток будет состоять лишь из обслуженных заявок.

Параметры выходящего потока

Выходящий поток в общем случае распадается на поток обслуженных и поток потерянных заявок, каждый из которых характеризуется законом распределения длительности интервала между соседними заявками.

Если входящий поток содержит заявки M типов с интенсивностями потока заявок типа i, ($\mathbf{i = \overline{1,M}}$) выходящий поток можно характеризовать суммарной интенсивностью потока обслуженных заявок

$$\lambda^{\text{О}} = \sum_{i=1}^{M} \lambda_i^{\text{О}}, \qquad (3.21)$$

где $\lambda_i^{\text{О}}$ - интенсивность потока обслуженных заявок типа i,

и суммарной интенсивностью потока потерянных заявок

$$\lambda^{\text{П}} = \sum_{i=1}^{M} \lambda_i^{\text{П}}, \qquad (3.22)$$

где $\lambda_i^{\text{П}}$ - интенсивность потока потерянных заявок типа i.

Очевидно, что $\lambda_i^{\text{О}} + \lambda_i^{\text{П}} = \lambda_i$.

В свою очередь, поток потерянных заявок может состоять из потока заявок, получивших отказ, и потока «нетерпеливых» заявок, покинувших систему, так как их время пребывания превысило допустимую величину, то есть $\lambda_i^{\text{П}} = \lambda_i^{\text{ОТК}} + \lambda_i^{\text{У}}$.

Показатели эффективности СМО

Под показателями эффективности понимаются количественные показатели, частично характеризующие уровень выполнения СМО возложенных на нее функций. На основании показателей эффективности может быть построен некоторый критерий эффективности, совокупно характеризующий эффектив-

ность СМО при ограничениях на ее параметры. Эффективность СМО может характеризоваться большим числом различных показателей эффективности. Рассмотрим наиболее употребительные из них и их обозначения. Следует помнить, что все эти показатели отражают возможности СМО по обслуживанию заявок, отнюдь не характеризуя качество самого обслуживания.

Вероятность обслуживания $\mathbf{P_{об}}$ характеризует вероятность того, что произвольно выбранная из входящего потока с интенсивностью λ заявка будет обслужена, то есть окажется в потоке обслуженных заявок с интенсивностью $\lambda^{О}$:

$$\mathbf{P_{об}} = \lambda^{О}/\lambda. \qquad (3.23)$$

Иногда вероятность обслуживания называют относительной пропускной способностью.

Вероятность потери $\mathbf{P_{П}}$ характеризует вероятность того, что произвольно выбранная из входящего потока с интенсивностью λ заявка окажется в потоке потерянных заявок с интенсивностью $\lambda^{П}$:

$$\mathbf{P_{П}} = \lambda^{П}/\lambda = (\lambda - \lambda^{О})/\lambda = \mathbf{1} - (\lambda^{О}/\lambda) = \mathbf{1} - \mathbf{P_{об}} \qquad (3.24)$$

и является суммой вероятностей потерь заявок по частным причинам: $\mathbf{P_{отк}}$ - вероятность отказа вследствие переполнения (насыщения) СМО, $\mathbf{P_{У}}$ - вероятность "ухода" нетерпеливых заявок из СМО.

$$\mathbf{P_{П}} = \mathbf{P_{отк}} + \mathbf{P_{У}}. \qquad (3.25)$$

Среднее время ожидания $\mathbf{t_{ож}}$ заявки (среднее время пребывания заявки в очереди) является математическим ожиданием времени ожидания. Время ожидания $\mathbf{t_{ож}}$ заявки является случайной величиной и равно сумме длительностей интервалов времени, в течение которых заявка находится в очереди, начиная с момента появления заявки на выходе СМО и кончая моментом, когда заявка последний раз покидает очередь по причине назначения на обслуживание или ухода из очереди (в случае нетерпеливых заявок).

Среднее время ожидания $\bar{\mathbf{t}}_{\mathbf{ож}}$ в общем случае является суммой двух составляющих:

- $\overline{\mathbf{t}^{\mathbf{Н}}}_{\mathbf{ож}}$ – среднего начального времени ожидания, равного промежутку времени между моментом появления заявки на входе СМО и моментом первого назначения заявки на обслуживание или ухода из очереди;
- $\overline{\mathbf{t}^{\mathbf{П}}}_{\mathbf{ож}}$ - среднего времени ожидания в прерванном состоянии, равного в общем случае сумме промежутков времени между моментами поступления заявки, обслуживание которой было прервано, в очередь и моментами либо повторного назначения заявки на дообслуживание (продолжение обслуживания заявки с того состояния, в котором она находилась на момент очередного прерывания), либо потери заявки за счет ухода:

$$\bar{\mathbf{t}}_{\mathbf{ож}} = \overline{\mathbf{t}^{\mathbf{Н}}}_{\mathbf{ож}} + \overline{\mathbf{t}^{\mathbf{П}}}_{\mathbf{ож}}. \qquad (3.26)$$

Среднее время пребывания заявки в СМО $\mathbf{t_{С}}$ является математическим ожиданием времени пребывания заявки в СМО. Время пребывания $\mathbf{t_{С}}$ заявки в СМО равно промежутку времени от момента поступления заявки на вход СМО

до момента появления ее в выходящем потоке и связано с длительностью процессов ожидания $t_{ож}$ и обслуживания $t_{об}$. Среднее время пребывания заявки в СМО $\bar{t}_c$ равно сумме среднего времени ожидания (пребывания в очереди) $\bar{t}_{ож}$ и среднего времени обслуживания (пребывания в канале обслуживания) $\bar{t}_{об}$:

$$\bar{t}_c = \bar{t}_{ож} + \bar{t}_{об}. \qquad (3.27)$$

Средняя длина очереди $\bar{l}$ представляет собой математическое ожидание числа заявок, находящихся в очереди, то есть длины очереди **l**. Для определения $\bar{l}$ в общем случае необходимо знание совокупности вероятностей $P_{ожі}$, где **i = 0,n**, то есть вероятностей нахождения в очереди **i** заявок. Для систем без потерь средняя длина очереди связана со средним временем ожидания $\bar{t}_{ож}$ простым соотношением

$$\bar{l} = \bar{t}_{ож}\, l\ . \qquad (3.28)$$

Это выражение становится очевидным, если учесть, что за время ожидания $t_{ож}$ в СМО поступает в среднем $l\, t_{ож}$ заявок.

Среднее число занятых каналов обслуживания $\bar{K}$ равно математическому ожиданию числа занятых обслуживанием каналов обслуживания, являющегося случайной величиной, и характеризует степень загрузки обслуживающей системы. Для определения $\bar{K}$ в общем случае необходимо знание совокупности $P_{зані}$, **i = 0,m** - вероятностей того, что в произвольный момент времени занято обслуживанием **i** каналов обслуживания. Важную роль в дальнейшем играет загрузка P_{SI} - вероятность того, что в произвольный момент времени обслуживанием будут заняты все **m** каналов обслуживания:

$$P_{SI} = \bar{K}/m. \qquad (3.29)$$

Среднее число заявок в системе $\bar{Z}$ представляет собой математическое ожидание числа заявок, одновременно находящихся в очереди или в канале обслуживания. Оно представляет собой сумму средней длины очереди и среднего числа занятых каналов обслуживания, так как с каждым каналом обслуживания в произвольный момент времени может быть связана только одна заявка

$$\bar{Z} = \bar{l} + \bar{K}. \qquad (3.30)$$

Для СМО без потерь среднее число заявок в системе связано со средним временем пребывания заявки в системе простым соотношением:

$$\bar{Z} = \bar{l} + \bar{K} = l\,\bar{t}_{ож} + l\,\bar{t}_{об} = l\,(\bar{t}_{ож} + \bar{t}_{об}) = l\,\bar{t}_c \qquad (3.31)$$

3.2.2. Разомкнутые СМО с ожиданием и терпеливыми заявками

Будем рассматривать СМО разомкнутого типа, содержащую **m** однотипных каналов обслуживания, характеризующихся экспоненциальным распределением времени обслуживания со средним значением $\bar{t}_{об}$ или, что эквивалентно, простейшим потоком обслуживаний с интенсивностью $\mu = 1/\bar{t}_{об}$ независимо от типа обслуживаемой заявки.

При полностью загруженных каналах обслуживания заявки могут ждать обслуживания в общей очереди, число мест в которой равно **n**. Дисциплина ожидания - FIFO, заявки становятся в очередь в порядке поступления, при переполнении очереди вновь поступившая заявка получает отказ. Дисциплина обслуживания также FIFO, выбор заявки из очереди при освобождении какого-либо из каналов обслуживания делается из начала очереди. Заявки на входе СМО относятся к одному из **M** типов, причем заявки **i**-го типа образуют простейший поток с интенсивностью λ_i ($\mathbf{i = \overline{1,M}}$).

Поскольку рассматривается бесприоритетная СМО с общей очередью, целесообразно рассматривать объединенный входящий поток, который будет также простейшим с интенсивностью (3.19):

$$\lambda = \sum_{i=1}^{M} \lambda_i .$$

Из-за того, что заявки терпеливые, в такой СМО их потери возможны лишь за счет отказов СМО принять заявку в очередь на обслуживание. Заявка, попавшая в очередь, обязательно дождется назначения на обслуживание.

Возможные состояния СМО будем связывать с числом заявок, находящихся в СМО:

- $\mathbf{S_0}$ - в СМО нет ни одной заявки, каналы обслуживания простаивают, очередь отсутствует;
- $\mathbf{S_1}$ - в СМО одна заявка, ее обслуживанием занят один из каналов обслуживания, другие **(m - 1)** канал обслуживания простаивают, очередь отсутствует;
- . . . . . .
- $\mathbf{S_i}$ - в СМО **i** заявок, их обслуживанием заняты **i** каналов обслуживания, другие **(m - i)** канал обслуживания простаивают, очередь отсутствует;
- . . . . . .
- $\mathbf{S_m}$ - в СМО **m** заявок, все каналы обслуживания загружены, очередь отсутствует;
- $\mathbf{S_{m+1}}$ - в СМО **(m +** 1**)** заявок, все каналы обслуживания загружены, последняя из пришедших в СМО заявок находится в очереди;
- . . . . . .
- $\mathbf{S_{m+l}}$ - в СМО **(m + l)** заявок, все каналы обслуживания загружены, **l** заявок находится в очереди (**l** - длина очереди);
- . . . . . .
- $\mathbf{S_{m+n}}$ - в СМО (**m+n**) заявок, все каналы обслуживания загружены, все **n** мест в очереди заняты ожидающими обслуживания заявками, СМО в этом состоянии не способна принять дополнительно ни одной заявки, все вновь приходящие заявки будут получать отказ (состояние «насыщения» системы).

Переходы между состояниями такой СМО будут происходить под действием входящего и выходящего потоков заявок. Граф функционирования такой системы имеет вид:

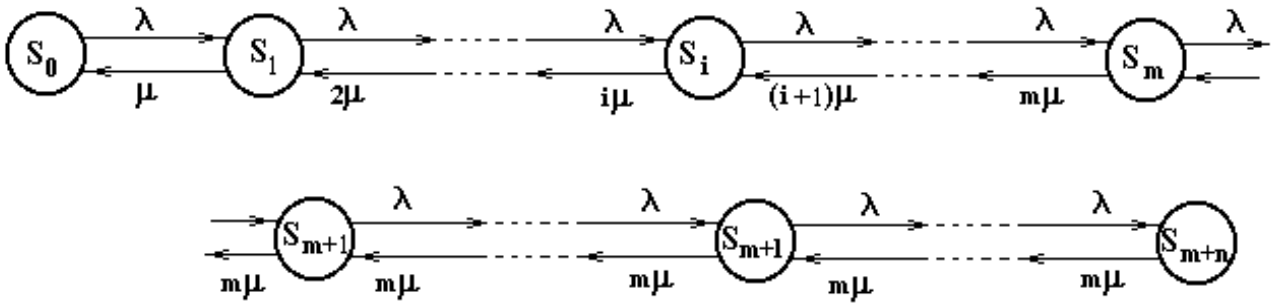


Рис. 3.7. Граф состояний разомкнутой многоканальной СМО с очередью и терпеливыми заявками

На основании представленного графа запишем алгебраическую систему уравнений, позволяющую рассчитать вероятности состояний системы для установившегося режима:

$$\begin{cases} -\lambda P_0 + \mu P_1 = 0, \\ -(\lambda + i\mu)P_i + \lambda P_{i-1} + (i+1)\mu P_{i+1} = 0; \quad 1 \le i \le m-1, \\ -(\lambda + m\mu)P_m + \lambda P_{m-1} + m\mu P_{m+1} = 0, \\ -(\lambda + m\mu)P_{m+i} + \lambda P_{m+i-1} + m\mu P_{m+i+1} = 0; \quad 1 \le i \le n-1, \\ -m\mu P_{m+n} + \lambda P_{m+n-1} = 0. \end{cases} \qquad (3.32)$$

Условие нормировки вероятностей имеет вид:

$$\sum_{i=0}^{m} P_i + \sum_{i=1}^{n} P_{m+i} = 1. \qquad (3.33)$$

Решая систему уравнений, можно определить искомые вероятности.

Рассчитаем показатели эффективности СМО.

Вероятность потери заявки P_n совпадает с вероятностью отказа $P_{ОТК}$ и равна вероятности нахождения СМО в состоянии S_{m+n}:

$$P_П = P_{ОТК} = P_{m+n}.$$

Вероятность обслуживания $P_{ОБ}$ и интенсивность потока обслуженных заявок λ^{o} определяются, соответственно, выражениями:

$$P_{ОБ} = 1 - P_П,$$

$$\lambda^{o} = \lambda P_{ОБ}.$$

Среднее число занятых каналов можно определить согласно выражению:

$$\overline{K} = \sum_{i=0}^{m} i p_i + m \sum_{i=1}^{n} p_{m+i}. \qquad (3.34)$$

Однако более удобно определение среднего числа занятых каналов как отношение интенсивности потока обслуженных заявок λ^{o} к интенсивности обслуживания μ, характеризующей производительность одного канала обслуживания:

$$\overline{K} = \frac{\lambda^{o}}{\mu} = \frac{\lambda P_{ОБ}}{\mu}. \qquad (3.35)$$

Найдем теперь среднюю длину очереди $\bar{l}$:

$$\bar{l} = \sum_{i=1}^{n} i P_{m+i} . \qquad (3.36)$$

Далее определяется среднее число заявок в СМО

$$\bar{Z} = \bar{K} + \bar{l}. \qquad (3.37)$$

Перейдем к определению временных характеристик.

Важным показателем эффективности СМО с ожиданием является среднее время ожидания заявки в очереди $\bar{t}_{ож}$, характеризующее запаздывание заявки за счет наличия в СМО других заявок. Для определения $\bar{t}_{ож}$ сформулируем возможные гипотезы о том, в каком состоянии застанет систему вновь прибывшая заявка и сколько времени ей придется ждать обслуживания с учетом принятых бесприоритетных дисциплин ожидания и обслуживания.

Если заявка застанет СМО в одном из состояний $S_0, S_1, ..., S_{m-1}$, ей вообще не придется ждать ($t_{ОЖ} = 0$), так как для этих состояний характерно наличие в системе хотя бы одного свободного канала, следовательно, задержка отсутствует и обслуживание заявки будет начато немедленно.

Застав СМО в состоянии S_m, когда все m каналов заняты, заявка должна будет встать в очередь (занять первое место в очереди) и ждать окончания обслуживания в одном из каналов обслуживания. Суммарный поток обслуживания при полностью загруженных каналах складывается из m простейших потоков обслуживаний с одинаковой для всех каналов средней длительностью обслуживания $\bar{t}_{об}$, следовательно, суммарный поток будет характеризоваться средней длительностью обслуживания $\bar{t}_{об} / m$ и интенсивностью обслуживания $m\mu$. Вследствие отсутствия последействия в простейшем суммарном потоке обслуживаний время ожидания заявки в рассматриваемой ситуации равно в среднем $1/(m\mu)$, причем вероятность этой ситуации равна P_m.

Застав СМО с вероятностью P_{m+1} в состоянии S_{m+1} (все каналы обслуживания заняты, в очереди одна заявка), вновь прибывшая заявка займет второе место в очереди и должна будет ждать в среднем $2/(m\mu)$ единиц времени и т.д.

Последнее состояние, находясь в котором СМО еще способна принять заявку в очередь, S_{m+n-1} имеет вероятность P_{m+n-1}, время ожидания заявки, заставшей СМО в состоянии S_{m+n-1}, равно в среднем $n/(m\mu)$. Время ожидания заявки, заставшей СМО в состоянии S_{m+n}, равно нулю, так как заявка получает отказ и является потерянной для СМО.

Поскольку рассмотренные гипотезы случайны, среднее время ожидания заявки в очереди определим как математическое ожидание времен ожидания, связанных с различными гипотезами:

$$\bar{t}_{ож} = \sum_{i=0}^{n-1} \left(\frac{i+1}{m\mu} \right) P_{m+i} . \qquad (3.38)$$

Между средней длиной очереди $\bar{l}$ и средним временем ожидания заявки в очереди существует связь:

$$\bar{l} = l\bar{t}_{ож}. \qquad (3.39)$$

Определим среднее время пребывания заявки в канале обслуживания.

Время пребывания в канале обслуживания $t_{об}$ - случайная величина, для определения которой необходимо рассмотреть различные возможные ситуации.

Если заявка застает СМО в состоянии S_{m+n}, вероятность которого равна $P_{m+n} = P_{отк}$, время пребывания ее в канале обслуживания равно нулю, поскольку заявка получает отказ и тотчас же попадает в выходящий поток СМО.

Застав СМО в любом другом состоянии, что происходит с вероятностью $1 - P_{m+n} = 1 - P_{отк} = P_{об}$, заявка попадает в систему и поскольку другие виды потерь в рассматриваемой СМО отсутствуют непременно проходит обслуживание, то есть время пребывания в канале обслуживания в данной ситуации равно случайной величине, распределенной экспоненциально со средней длительностью обслуживания $\bar{t}_{об}$.

Теперь нетрудно найти среднее время обслуживания:

$$\bar{t}_{об} = \frac{1}{m}p_{об}. \qquad (3.40)$$

Сравним среднее время обслуживания со средним числом занятых каналов $\overline{K}$. Убедимся в справедливости соотношения, связывающего эти две характеристики:

$$\overline{K} = l\bar{t}_{об}. \qquad (3.41)$$

Рассмотрим еще один показатель эффективности СМО – среднее время пребывания заявки в системе. Среднее время пребывания произвольной заявки в системе складывается из среднего времени пребывания в очереди (ожидания) $\bar{t}_{ож}$ и из среднего времени пребывания в канале обслуживания $\bar{t}_{об}$. Вследствие независимости процессов обслуживания и ожидания справедливо соотношение:

$$\bar{t}_{с} = \bar{t}_{ож} + \bar{t}_{об}.$$

Отсюда можно найти среднее число заявок в системе:

$$\overline{Z} = l\bar{t}_{с}. \qquad (3.42)$$

3.2.3. Предельные варианты разомкнутой СМО

Рассмотренная выше разомкнутая СМО имела ограниченное число мест в очереди. С методологической точки зрения полезно исследовать два крайних случая такой СМО, а именно – полное отсутствие мест в очереди и очередь бесконечной длины.

Для этой СМО с отсутствием очереди ($n = 0$) заявка, поступившая на вход СМО, либо сразу попадает на обслуживание, если свободен хотя бы один из каналов обслуживания, либо получает отказ и попадает в ту часть выходя-

щего потока, которая соответствует потерям. В каждый момент времени с системой может быть связано не более **m** заявок, где **m** - число каналов обслуживания.

Граф состояний m-канальной СМО с отказами приведен на рис.3.8.

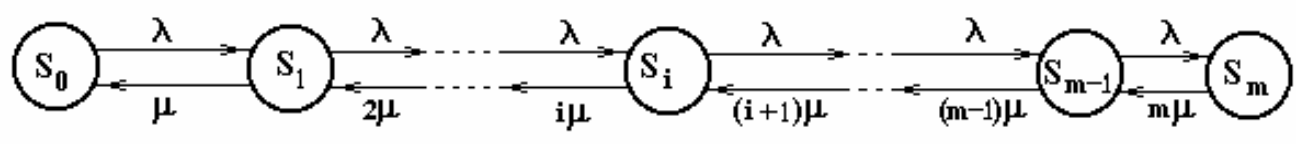


Рис. 3.8. Граф состояний разомкнутой многоканальной СМО без очереди

Исходя из этого графа, для установившегося режима запишем алгебраическую систему уравнений в виде:

$$\begin{cases} -\lambda P_0 + \mu P_1 = 0, \\ -(\lambda + i\mu)P_i + \lambda P_{i-1} + (i+1)\mu P_{i+1} = 0; \quad 1 \le i \le m-1, \\ -m\mu P_m + \lambda P_{m-1} = 0. \end{cases} \qquad (3.43)$$

Условие нормировки вероятностей имеет вид:

$$\sum_{i=0}^{m} P_i = 1.$$

Решая данную систему уравнений, можно найти вероятности пребывания системы в **i**-х состояниях.

Отказ получает заявка, заставшая СМО в состоянии S_m, следовательно

$$P_{отк} = P_m.$$

Вероятность обслуживания $P_{об}$ и интенсивность потока обслуженных заявок λ^{o} равны, соответственно

$$P_{об} = 1 - P_{отк} = 1 - P_m,$$
$$\lambda^{o} = \lambda P_{об} = \lambda(1 - P_m).$$

Среднее число занятых каналов можно определить либо как отношение интенсивности потока обслуженных заявок λ^{o} к производительности одного канала обслуживания, характеризуемой интенсивностью обслуживания μ.:

$$\overline{K} = \frac{\lambda^{o}}{\mu} = \frac{\lambda P_{об}}{\mu},$$

либо по зависимости:

$$\overline{K} = \sum_{i=0}^{m} i P_i .$$

Среднее число заявок $\overline{Z}$, связанных с системой, совпадает со средним числом каналов обслуживания:

$$\overline{Z} = \overline{K}.$$

Среднее время пребывания заявки в системе совпадает со средним временем ее обслуживания.

$$\bar{t}_{об} = \frac{1}{m} P_{об}.$$

Рассмотрим другой предельный вариант СМО, когда число мест в очереди бесконечно (**n** ® ¥) и на время пребывания заявки в системе не наложено ограничений. Такие СМО называются чистыми СМО с ожиданием, или СМО без потерь, поскольку любая заявка, поступающая на вход системы, либо немедленно назначается на обслуживание, либо ставится в очередь на обслуживание, причем из-за отсутствия потерь все заявки из входящего потока рано или поздно будут обслужены.

Граф состояний такой СМО имеет вид:

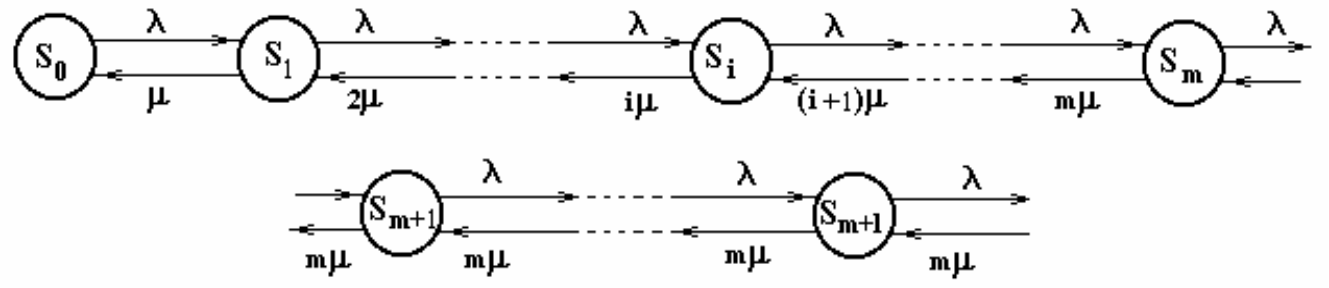


Рис. 3.9. Граф состояний разомкнутой многоканальной СМО с бесконечной очередью

Число состояний такой системы бесконечно велико. Следовательно, система алгебраических уравнений для нее будет иметь бесконечную размерность.

Такие системы без потерь не имеют аналогов среди реальных СМО, однако удобны тем, что позволяют получить предельные соотношения для случая, когда число мест в очереди заметно превышает среднюю длину очереди.

Анализ рассматриваемой СМО, проведенный специальными методами, показывает, что для нее возможен установившийся режим (вероятности состояний отличны от нуля) при выполнении следующего условия:

$$\lambda/(m\mu) < 1.$$

То есть интенсивность входящего потока λ должна быть меньше максимальной интенсивности потока обслуживания $m\mu$. Невыполнение указанного неравенства приведет к тому, что каналы обслуживания не будут справляться с потоком заявок, и длина очереди будет неограниченно возрастать.

Потери в такой СМО отсутствуют, поэтому вероятность обслуживания равна единице $P_{об} = 1$, интенсивность потока обслуженных заявок совпадает с интенсивностью входящего потока $\lambda = \lambda^{o}$.

3.2.4.Общий случай разомкнутой СМО

Рассматривая общий случай разомкнутой СМО, заявки следует считать нетерпеливыми, то есть имеющими право пробыть в СМО не более $t_{доп}$ единиц времени. Если время пребывания заявки в системе t_C превышает $t_{доп}$, заявка покидает систему и считается для нее потерянной. Нетерпеливость заявок

хорошо отражает свойство старения информации в вычислительных системах реального времени. Будем считать $t_{доп}$ случайной величиной, распределенной экспоненциально с математическим ожиданием $M(t_{доп}) = \bar{t}_{доп}$.

Удобной для дальнейшего рассмотрения абстракцией является представление о простейшем потоке уходов из СМО с интенсивностью $n = 1/\bar{t}_{доп}$. Уходы заявки возможны либо из очереди, если $t_{ож} > \bar{t}_{доп}$, либо из канала обслуживания, если $t_{ож} \le \bar{t}_{доп} \le t_C$. Методически удобно рассматривать два потока уходов с интенсивностями, соответственно, $n_{ож} + n_{об} = n = 1/\bar{t}_{доп}$.

Граф состояний, соответствующий описанной СМО, приведен на рисунке 3.10.

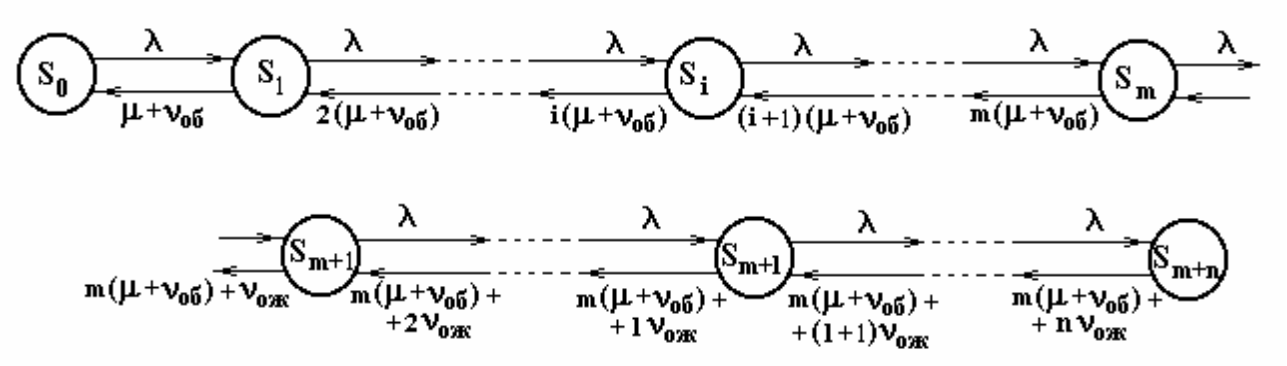


Рис. 3.10. Граф состояний разомкнутой многоканальной СМО с очередью и нетерпеливыми заявками

Переходы между состояниями такой СМО будут происходить под действием входящего потока заявок, потоков уходов нетерпеливых заявок из очереди или канала обслуживания и потоков обслуживаний.

Система линейных алгебраических уравнений для предельных вероятностей состояний системы имеет вид:

$$
\begin{cases}
-\lambda P_0 + (m + n_{ОБ}) P_1 = 0, \\
-[\lambda + i(m + n_{ОБ}) P_i + \lambda P_{i-1} + (i+1)(m + n_{ОБ}) P_{i+1} = 0; \quad 1 \le i \le m - 1, \\
-[\lambda + m(m + n_{ОБ})] P_m + \lambda P_{m-1} + [m(m + n_{ОБ}) + n_{ОЖ}] P_{m+1} = 0, \\
-\{\lambda + [m(m + n_{ОБ}) + i n_{ОЖ}]\} P_{m+i} + \lambda P_{m+i-1} + [m(m + n_{ОБ}) (i+1) n_{ОЖ}] P_{m+i+1} = 0; \quad 1 \le i \le n - \\
-[m(m + n_{ОБ}) + n n_{ОЖ}] P_{m+n} + \lambda P_{m+n-1} = 0.
\end{cases}
$$

(3.44)

Условие нормировки вероятностей:

$$\sum_{i=0}^{m} P_i + \sum_{i=1}^{n} P_{m+i} = 1.$$

Решая данную систему уравнений, можно найти искомые вероятности.

Основные показатели эффективности СМО могут быть найдены по уже известным формулам:

Среднее число занятых каналов:

$$\overline{K} = \sum_{i=0}^{m} i p_i + m \sum_{i=1}^{n} p_{m+i} .$$

Средняя длина очереди:

$$\bar{l} = \sum_{i=1}^{n} i P_{m+i} .$$

Среднее число заявок в СМО

$$\overline{Z} = \overline{K} + \bar{l}.$$

В рассматриваемой СМО потери заявок возможны либо в форме отказа вследствие переполнения системы, либо в форме ухода нетерпеливых заявок из системы.

Вероятность отказа $P_{ОТК}$ может быть определена как вероятность нахождения системы в состоянии P_{m+n}, то есть

$$P_{ОТК} = P_{m+n} .$$

Уход нетерпеливой заявки из СМО возможен либо во время ожидания, либо во время обслуживания. Поскольку такие случайные события как уход заявки из очереди и уход заявки из канала обслуживания несовместны, то вероятность ухода может быть представлена суммой:

$$P_У = P_У^{ОЖ} + P_У^{ОБ},$$

где $P_У^{ОЖ}$ - вероятность ухода заявки во время ожидания; $P_У^{ОБ}$ - вероятность ухода заявки во время обслуживания.

Вероятность ухода заявки во время обслуживания можно определить как отношение суммарной интенсивности ухода во время обслуживания, равной произведению среднего числа занятых каналов $\overline{K}$ на интенсивность уходов из одного канала обслуживания $n_{ОБ}$, к интенсивности входящего потока l, то есть

$$p_У^{ОБ} = \frac{\overline{K} n_{ОБ}}{l} .$$

Аналогично можно выразить вероятность ухода во время ожидания через среднюю длину очереди $\bar{l}$ интенсивность ухода одной заявки из очереди $n_{ОЖ}$ и интенсивность входящего потока l

$$p_У^{ОЖ} = \frac{\bar{l} n_{ОЖ}}{l} .$$

Вследствие несовместимости таких случайных событий, как отказ системы принять заявку к обслуживанию и уход нетерпеливой заявки из системы, по теореме сложения вероятностей можно записать выражение для вероятности потери заявки:

$$P_П = P_{ОТК} + P_У^{ОЖ} + P_У^{ОБ}.$$

Вероятность обслуживания $P_{ОБ}$, то есть вероятность появления в потоке обслуженных заявок произвольной заявки из входящего потока, может быть определена как дополнение вероятности потерь до единицы:

$$P_{ОБ} = 1 - P_П .$$

Отсюда можно получить такую характеристику выходящего потока СМО, как интенсивность потока обслуженных заявок λ^{O}:

$$\lambda^{O} = \lambda P_{ОБ} = \lambda(1 - P_{П})$$

Пример. 3.4

Проектируется трехпроцессорная ВС оперативной обработки информации. Интенсивность потока задач $\lambda = 1{,}2\ c^{-1}$, средняя трудоемкость задач $\bar{t} = 4 \cdot 10^5$ операций. Используются процессоры со средним быстродействием $\bar{B} = 2 \cdot 10^5$ оп/с. Объем буферной памяти не позволяет иметь очередь. Потоки событий в системе можно считать простейшими.

Необходимо сравнить два варианта организации вычислительного процесса:

- каждый из процессоров по отдельности реализует последовательный алгоритм обработки;
- все три процессора реализуют параллельный алгоритм обработки, причем среднее время обработки за счет организации параллелизма уменьшается в 2,667 раза.

Для сравнения вариантов использовать комплексный показатель эффективности: **E = 25 усл. ед** λ^{O} **- 10 усл. ед** $\bar{t}_{ОБ}$ **- 10 усл. ед** $\lambda^{П}$.

Решение:

1. Для анализа первого варианта используем модель трехканальной СМО без очереди:

$$m = 3 \qquad \mu_1 = \bar{B} / \bar{t} = 0{,}5\ c^{-1}.$$

Система уравнений для установившегося режима будет иметь вид:

$$\begin{cases} \lambda P_0 - 1{,}2P_0 + 0{,}5P_1 = 0, \\ 1{,}2P_0 - (1{,}2 + 0{,}5)P_1 + 2 \cdot 0{,}5P_2 = 0, \\ 1{,}2P_1 - (1{,}2 + 2 \cdot 0{,}5)P_2 + 3 \cdot P_3 = 0, \\ P_0 + P_1 + P_2 + P_3 = 1. \end{cases}$$

Решение данной системы:

$$P_0 = 0{,}12;\ P_1 = 0{,}28;\ P_2 = 0{,}33;\ P_3 = 0{,}27.$$

Вероятность потери заявок:

$$P_{ОТК} = P_3 = 0{,}27.$$

Интенсивность потока потерянных заявок $\lambda^{П}$

$$\lambda^{П} = \lambda P_{ОТК} = 0{,}322\ c^{-1}.$$

Вероятность обслуживания $P_{ОБ}$ и интенсивность потока обслуженных заявок λ^{O} равны, соответственно

$$P_{ОБ} = 1 - P_{ОТК} = 0{,}73,$$
$$\lambda^{O} = \lambda P_{ОБ} = 0{,}878\ c^{-1}.$$

Среднее время обслуживания заявки.

$$\bar{t}_{ОБ} = \frac{1}{\mu_1} P_{ОБ} = 1{,}46\ c.$$

Показатель эффективности:

$$\mathbf{E = 25 * 0{,}878 - 10 * 1{,}46 - 10 * 0{,}322 = 4{,}1\ усл.ед}.$$

2. Анализ варианта с параллельной обработкой проведем на модели одноканальной СМО, для которой интенсивность потока обслуживания равна:

$$\mu_2 = 2{,}667\mu_1 = 1{,}333\ \mathbf{c^{-1}}\ .$$

Система уравнений для установившегося режима будет иметь вид:

$$\begin{cases} \lambda - 1{,}2P_0 + 1{,}333P_1 = 0, \\ P_0 + P_1 = 1. \end{cases}$$

Решение данной системы:

$$P_0 = 0{,}53;\ P_1 = 0{,}47.$$

Вероятность потери заявок:

$$P_{ОТК} = P_1 = 0{,}47.$$

Интенсивность потока потерянных заявок $\lambda^{П}$

$$\lambda^{П} = \lambda P_{ОТК} = 0{,}57\ \mathbf{c^{-1}}$$

Вероятность обслуживания $P_{ОБ}$ и интенсивность потока обслуженных заявок $\lambda^{О}$ равны, соответственно

$$P_{ОБ} = P_0 = 0{,}53,$$

$$\lambda^{О} = \lambda P_{ОБ} = 0{,}63\ \mathbf{c^{-1}}\ .$$

Среднее время обслуживания заявки.

$$\bar{t}_{ОБ} = \frac{1}{\mu_2} P_{ОБ} = 0{,}39\ \mathbf{с}.$$

Показатель эффективности:

$$\mathbf{E = 25 * 0{,}63 - 10 * 0{,}39 - 10 * 0{,}57 = 6{,}2\ усл.ед}.$$

Вывод

При заданных коэффициентах показателя эффективности вариант параллельной обработки более предпочтителен.

3.2.5. Замкнутые СМО

Важный и интересный класс СМО составляют СМО замкнутого типа. Как говорилось ранее, для замкнутых СМО характерно конечное число источников заявок, причем параметры суммарного входящего потока СМО зависят от состояния самой СМО.

Примером замкнутой СМО может служить вычислительная система оперативной обработки с диалоговым режимом работы. Упрощенно представим ее функционирование на некотором интервале времени следующим образом.

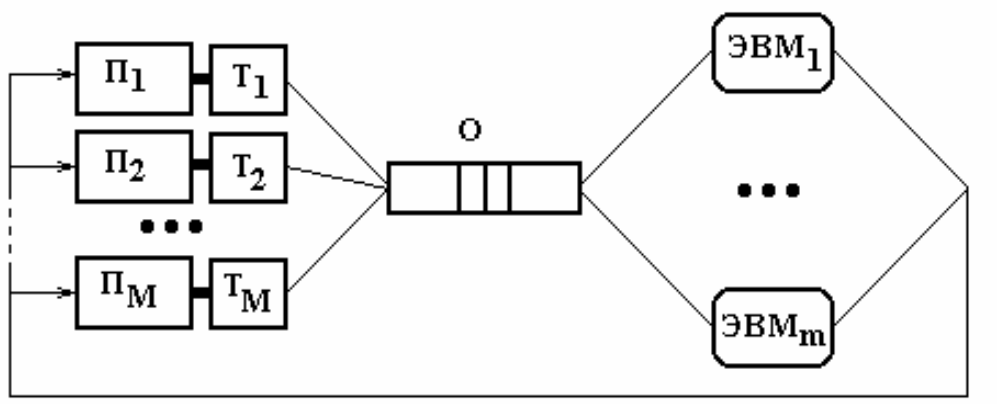


Рис. 3.11. Схема замкнутой СМО

Система оперативной обработки содержит **M** терминалов, за каждым из которых работает пользователь (**П**), формирующий запросы на обслуживание заявки. Обслуживание запросов выполняется совокупностью из **m** однотипных ЭВМ (**m** £ **M**), рассматриваемых без детализации их внутренней структуры, как каналы обслуживания с длительностью обслуживания, распределенной по экспоненциальному закону с математическим ожиданием $\overline{t}_{об}$. Операционная система реализует бесприоритетные дисциплины ожидания и обслуживания в порядке поступления запросов. Причем все ресурсы некоторой ЭВМ (каналы обслуживания) полностью монополизируются назначенной на обслуживание заявкой до конца ее обслуживания. Заявка, заставшая все каналы обслуживания занятыми, занимает место в очереди, число мест в которой **n = M - m**, заявки считаются терпеливыми. Формирование нового запроса пользователь начинает лишь после получения ответа на предыдущий запрос. Причем длительность промежутка времени, необходимого пользователю для формирования очередного запроса, будем считать распределенной экспоненциально с математическим ожиданием $\overline{\mathbf{T}}$, что позволяет рассматривать пользователя как источник простейшего потока с интенсивностью $\lambda = 1/\overline{\mathbf{T}}$.

Построим граф состояний такой СМО.

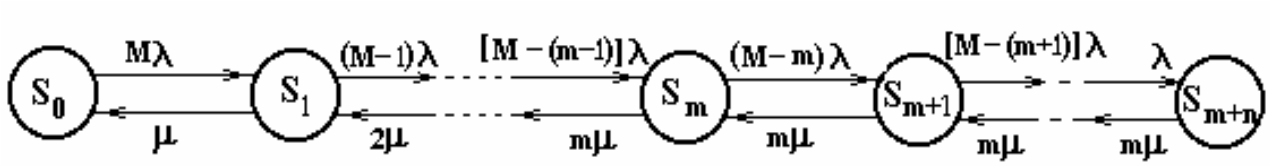


Рис. 3.12. Граф состояний многоканальной замкнутой СМО

Возможные состояния системы будем связывать с числом пользователей, ожидающих ответа на сделанные запросы, то есть с числом заявок, находящихся на обслуживании и в очереди:

- **S_0** - в системе нет ни одной заявки, каналы обслуживания простаивают. Все пользователи независимо друг от друга заняты подготовкой запросов, следовательно, интенсивность суммарного потока заявок, переводящего СМО в состояние **S_1**, равна **M**λ;

- $\mathbf{S_1}$ - в системе одна заявка, обслуживанием которой занят один канал обслуживания. Пославший запрос пользователь ждет ответа и не формирует новых запросов, следовательно, интенсивность потока переходов в соседнее справа состояние равна $\mathbf{(M - 1)}\lambda$. Интенсивность потока переходов в соседнее слева состояние связана с интенсивностью суммарного потока обслуживаний, равной произведению числа занятых каналов на интенсивность потока обслуживаний одного канала, то есть μ;
- $\mathbf{S_m}$ - в системе **m** заявок, все ЭВМ заняты обслуживанием запросов пользователей. Очереди на обслуживание еще нет, интенсивность суммарного потока заявок равна $\mathbf{(M - m)}\lambda$, суммарного потока обслуживания - $\mathbf{m}\mu$;
- $\mathbf{S_{m+l}}$ - в системе $\mathbf{m + l}$ заявок, все ЭВМ заняты и одна заявка стоит в очереди на обслуживание. Интенсивность суммарного потока заявок равна $\mathbf{[M-(m + l)]}\lambda$, где **l** - длина очереди, суммарный поток обслуживаний имеет интенсивность по-прежнему $\mathbf{m}\mu$;
- $\mathbf{S_{m+n}}$ - в системе $\mathbf{m + n}$ заявок, то есть все пользователи сформировали и ввели в систему запросы на обслуживание, **m** ЭВМ обслуживают **m** заявок, $\mathbf{n=M - m}$ заявок находятся в очереди на обслуживание. Интенсивность суммарного потока заявок равна нулю, так как все пользователи ждут ответа на свои запросы, интенсивность суммарного потока обслуживания равна $\mathbf{m}\mu$;

Для исследования переходного режима в рассматриваемой СМО необходимо составить и решить систему дифференциальных уравнений Колмогорова.

На основании представленного графа запишем алгебраическую систему уравнений, позволяющую рассчитать вероятности состояний системы для установившегося режима:

$$\begin{cases} -\mathbf{M}\lambda P_0 + \mu P_1 = 0 \\ [\mathbf{M - (i - 1)}]\lambda P_{i-1} - [\mathbf{(M - i)}\lambda + \mathbf{i}\mu]P_i + \mathbf{(i + 1)}\mu P_{i+1} = 0; \quad 1 \le \mathbf{i} \le \mathbf{m - 1}; \\ [\mathbf{M - (m - 1)}]\lambda P_{m-1} - [\mathbf{(M - m)}\lambda + \mathbf{m}\mu]P_m + \mathbf{m}\mu P_{m+1} = 0; \\ \{\mathbf{M - [m + (i - 1)]}\}\lambda P_{m+i-1} - \{\mathbf{m}\mu + \mathbf{[M - (m + i)]}\lambda\}P_{m+i} + \mathbf{m}\mu P_{m+i+1} = 0; \quad 1 \le \mathbf{i} \le \mathbf{n - 1} \\ -\mathbf{m}\mu P_{m+n} + \lambda P_{m+n-1} = 0 \end{cases} \tag{3.45}$$

Условие нормировки вероятностей имеет вид

$$\sum_{i=0}^{m} \mathbf{P_i} + \sum_{i=1}^{n} \mathbf{P_{m+i}} = \mathbf{1}.$$

Решая систему уравнений, можно определить искомые вероятности.

Вероятность обслуживания для замкнутой СМО равна единице, так как любая заявка будет в конце концов обслужена: $\mathbf{P_{ОБ} = 1}$.

Среднее число занятых каналов обслуживания $\overline{\mathbf{K}}$ может быть найдено как математическое ожидание числа занятых каналов:

$$\overline{\mathbf{K}} = \sum_{i=0}^{m} \mathbf{i}p_i + \mathbf{m}\sum_{i=1}^{n} \mathbf{p_{m+i}}.$$

Интенсивность потока обслуживания заявок λ^{o} для замкнутой СМО имеет смысл средней суммарной производительности каналов обслуживания, равной произведению среднего числа занятых каналов обслуживания $\overline{K}$ на производительность одного канала обслуживания μ:

$$\lambda^{o} = \overline{K}\mu.$$

Зная среднее число занятых каналов и, соответственно, интенсивность потока обслуживания заявок замкнутой СМО, можно найти среднее число заявок в системе из следующих рассуждений.

Поскольку пользователь, сформировав и введя в систему запрос на обслуживание, до получения ответа на свой запрос не формирует новых запросов, суммарный поток заявок на обслуживание будет определяться $(M - \overline{Z})\lambda$. Поскольку система находится в установившемся режиме, интенсивность суммарного потока заявок на входе системы должна уравновешиваться интенсивностью потока обслуженных заявок, то есть в среднем должно выполняться соотношение $(M - \overline{Z})\lambda = \overline{K}\mu$.

Отсюда несложно найти среднее число заявок в системе:

$$\overline{Z} = \frac{M\lambda - \overline{K}\mu}{\lambda}.$$

Зная $\overline{Z}$ и $\overline{K}$, находим среднюю длину очереди:

$$\bar{l} = \overline{Z} - \overline{K}.$$

Важной характеристикой замкнутой СМО является среднее время пребывания заявки в системе $\bar{t}_{C}$, характеризующее и среднее время, в течение которого пользователь ждет ответа на свой запрос, то есть среднее время реакции системы.

Поскольку для замкнутой системы из-за отсутствия потерь среднее время обслуживания $\bar{t}_{об}$ совпадает со средней длительностью обслуживания $\bar{t}_{об}$, следует найти среднее время пребывания заявки в очереди, то есть $\bar{t}_{ож}$.

Рассмотрим возможные гипотезы относительно времени ожидания $\bar{t}_{ож}$ в очереди заявки, поступающей на вход системы в случайный момент времени.

Заявка, застающая СМО в одном из состояний $S_0, ..., S_{m-1}$, не будет ждать $(t_{ож} = 0)$, так как застает хотя бы один свободный канал обслуживания. Нулевое время ожидания мы должны приписать и состоянию S_{m+n}, так как в этом состоянии внутри замкнутой системы не может появиться новых заявок.

С состоянием S_m следует связать ненулевое время ожидания, в среднем равное $1/(m\mu)$, так как застав СМО в состоянии S_m, заявка должна будет стать в очередь и дождаться ближайшего события в суммарном потоке обслуживаний, то есть ближайшего момента окончания обслуживаний в каком-либо из m независимо функционирующих занятых каналов обслуживания.

Заявка, заставшая СМО в состоянии S_{m+1}, занимает в очереди второе место и должна будет дождаться второго из ближайших событий в суммарном потоке обслуживаний, время ожидания для нее удваивается.

Заявка, застающая СМО в состоянии S_{m+i}, **(i = 1,n-1)**, должна ждать в среднем **(i+1)/(mμ)**, единиц времени. Среднее время ожидания найдем как математическое ожидание времени ожидания, связанного с произвольным состоянием СМО:

$$\bar{t}_{ож} = \sum_{i=0}^{n-1} \frac{(i+1)}{m\mu} p_{m+i} . \qquad (3.46)$$

Среднее время пребывания заявки в системе $\bar{t}_C$, или время реакции системы, равно

$$\bar{t}_C = \bar{t}_{ож} + \bar{t}_{об} = \sum_{L=0}^{n-1} \frac{(l+1)}{m\mu} p_{m+l} + \frac{1}{\mu}. \qquad (3.47)$$

Пример.3.5

В лаборатории 5 однотипных ЭВМ, которые обслуживаются двумя инженерами. ЭВМ требует обслуживания в среднем каждые 2 часа. Инженер тратит на обслуживание в среднем 12 минут. ЭВМ имеет пропускную способность 3 зад/ч. Сравнить по пропускной способности лаборатории два варианта организации обслуживания ЭВМ:

1) каждый инженер обслуживает ЭВМ самостоятельно;

2) инженеры обслуживают ЭВМ совместно, однако их взаимопомощь увеличивает производительность в 1,9 раз.

Решение.

В этой задаче источником заявок на обслуживание являются ЭВМ, роль каналов обслуживания играют инженеры.

Первый вариант организации обслуживания сводится к замкнутой СМО с параметрами:

$$\mathbf{M = 5; \quad m = 2; \quad n = (M - m) = 3.}$$

Интенсивность входящего потока $\lambda = 0{,}5$ час$^{-1}$, интенсивность потока обслуживания $\mu_1 = 5$ ч$^{-1}$.

Система уравнений для установившегося режима будет иметь вид:

$$\begin{cases} -2{,}5P_0 + 5P_1 = 0, \\ 2{,}5P_0 - 7P_1 + 10P_2 = 0, \\ 2P_1 - 11{,}5P_2 + 10P_3 = 0, \\ 1{,}5P_2 - 11P_3 + 10P_4 = 0, \\ P_3 - 10{,}5P_4 + 10P_5 = 0, \\ P_0 + P_1 + P_2 + P_3 + P_4 + P_5 = 1. \end{cases}$$

Решение данной системы:

$$\mathbf{P_0 = 0{,}618;\ P_1 = 0{,}309;\ P_2 = 0{,}062;\ P_3 = 0{,}01;\ P_4 = 0{,}001;\ P_5 = 0.}$$

Среднее число занятых инженеров:

$$\overline{K} = P_1 + 2P_2 + 2(P_3 + P_4 + P_5) = 0{,}45.$$

Среднее число обслуживаемых ЭВМ:

$$\overline{Z} = (5 \times 0{,}5 - 0{,}45 \times 5)/0{,}5 = 0{,}5$$

Средняя длина очереди:

$$\bar{l} = 0{,}5 - 0{,}45 = 0{,}05.$$

Среднее число ЭВМ, занятых производительной работой, равно

$$M - \overline{Z} = 4{,}5.$$

Пропускная способность системы равна $3 \times 4{,}5 = 13{,}5$ **зад/ч**.

Второй вариант с взаимной помощью сводится к замкнутой СМО с параметрами:

$M = 5;$ $m = 1;$ $n = 4;$ $\lambda = 0{,}5$ **ч**$^{-1}$, $\mu_2 = 1{,}9\mu_1 = 9{,}5$ **ч**$^{-1}$,

Система уравнений для установившегося режима будет иметь вид:

$$\begin{cases} -2{,}5P_0 + 9{,}5P_1 = 0, \\ 2{,}5P_0 - 11{,}5P_1 + 9{,}5P_2 = 0, \\ 2P_1 - 11P_2 + 9{,}5P_3 = 0, \\ 1{,}5P_2 - 10{,}5P_3 + 9{,}5P_4 = 0, \\ P_3 - 10P_4 + 9{,}5P_5 = 0, \\ P_0 + P_1 + P_2 + P_3 + P_4 + P_5 = 1. \end{cases}$$

Решение данной системы:

$$P_0 = 0{,}751;\ P_1 = 0{,}199;\ P_2 = 0{,}042;\ P_3 = 0{,}007;\ P_4 = 0{,}001;\ P_5 = 0.$$

Среднее число занятых инженеров:

$$\overline{K} = P_1 + P_2 + P_3 + P_4 + P_5 = 0{,}249.$$

Среднее число обслуживаемых ЭВМ:

$$\overline{Z} = (5 \times 0{,}5 - 0{,}249 \times 9{,}5)/0{,}5 = 0{,}269.$$

Средняя длина очереди:

$$\bar{l} = 0{,}269 - 0{,}249 = 0{,}02.$$

Среднее число ЭВМ, занятых производительной работой, равно

$$M - \overline{Z} = 4{,}73.$$

Пропускная способность системы равна $3 \times 4{,}73 = 14{,}2$ **зад/ч**.

Вывод

Организация работы по второму варианту повышает пропускную способность лаборатории.

3.2.6. Сети массового обслуживания с простейшими потоками событий

Рассмотренные ранее системы массового обслуживания (СМО) позволяют с той или иной степенью точности описывать процессы, протекающие в от-

дельных устройствах или подсистемах современных вычислительных систем. Для исследования процессов в вычислительной системе в целом необходимо рассматривать взаимодействие некоторой совокупности систем массового обслуживания (стохастическую сеть).

Ограничимся рассмотрением линейных стохастических сетей, которые могут быть построены из конечного числа **n** СМО без потерь $\mathbf{S_i}$, $\mathbf{i = 1,\ldots n}$ и источника заявок $\mathbf{S_0}$ следующим образом. Заявки из источника заявок $\mathbf{S_0}$ поступают в сеть, причем с постоянной вероятностью $\mathbf{p_{0i}}$, $\mathbf{i = 1,\ldots n}$, заявка, появляющаяся на выходе источника, поступит в систему $\mathbf{S_i}$, $\mathbf{i = 1,\ldots n}$. Заявки, обслуженные системой $\mathbf{S_i}$, $\mathbf{i = 1,\ldots n}$., с постоянной вероятностью $\mathbf{p_{ij}}$, $\mathbf{i = 1,\ldots n}$, $\mathbf{j = 1,\ldots n}$ поступают в систему $\mathbf{S_j}$, $\mathbf{j = 1,\ldots n}$ или покидают сеть $\mathbf{(j = 0)}$, причем заявки, покидающие сеть, возвращаются в источник заявок. Очевидно, должно выполняться равенство

$$\sum_{j=0}^{n} \mathbf{p_{ij}} = \mathbf{1}, \qquad \mathbf{i = 0,\ldots n},$$

причем $\mathbf{p_{00} = 0}$.

Структура сети может быть представлена графом, вершины которого соответствуют системам массового обслуживания $\mathbf{S_i}$, $\mathbf{i = 1,\ldots n}$ и источнику заявок $\mathbf{S_0}$, а дуги, взвешенные вероятностями $\mathbf{p_{ij}}$, $\mathbf{i = 1,\ldots n}$, $\mathbf{j = 1,\ldots n}$ - передачами между элементами сети. Не следует путать этот граф, называемый графом передачи сети, с графом переходов между состояниями системы, встречавшимся ранее при рассмотрении СМО с простейшими потоками событий.

Различают разомкнутые и замкнутые стохастические сети.

Разомкнутая сеть характеризуется постоянной и независящей от состояния сети интенсивностью потока заявок λ^o на выходе источника заявок $\mathbf{S_0}$, причем интенсивность λ^o заранее известна и является параметром сети.

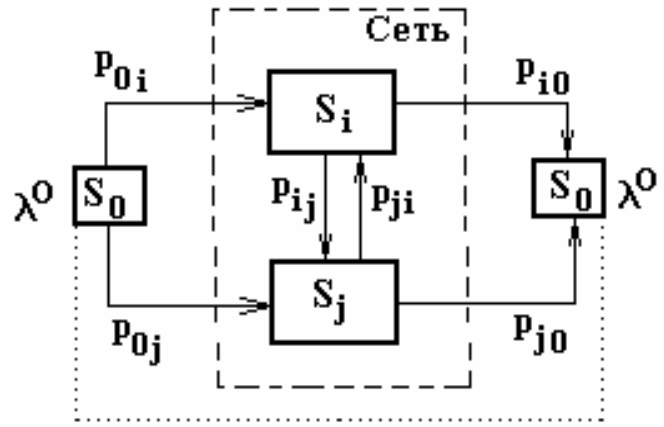


Рис. 3.13. Граф передачи для разомкнутой СеМО

Разомкнутые сети применяются для описания вычислительных систем, в которых на обработке может находиться переменное число задач, например, вычислительных систем оперативной обработки с разделением времени.

Замкнутая сеть характеризуется тем, что в нее не могут попадать заявки извне, число заявок M, циркулирующих в ней, всегда постоянно и определяется начальными условиями.

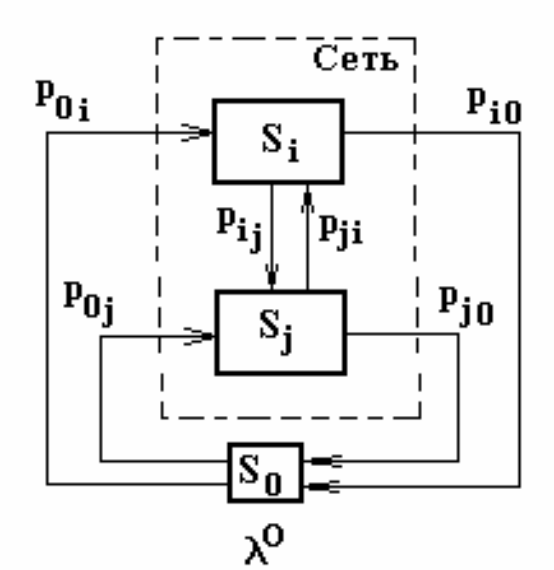


Рис. 3.14. Граф передачи для замкнутой СеМО

Заявка, появляющаяся в выходящем потоке замкнутой сети, тут же вызывает появление новой заявки во входящем потоке, интенсивность l^0 фиктивного источника заявок характеризует производительность замкнутой сети, не зависящую от каких-либо внешних причин, а определяемую конфигурацией сети и ее параметрами. Замкнутые стохастические сети используются для описания вычислительных систем, с которыми в каждый момент времени связано фиксированное число заявок, например, систем пакетной обработки.

Полный перечень сведений (параметров) о стохастической сети содержит:

1) число **n** СМО, образующих сеть;

2) число каналов обслуживания $\mathbf{m_i}$, входящих в состав СМО $\mathbf{S_i}$, **i = 1,…n**;

3) интенсивности потоков обслуживания m_i в СМО $\mathbf{S_i}$, **i = 1,…n**;

4) матрицу вероятностей передач $\mathbf{p = [p_{ij}]}$ **i,j = 0,…n**;

5) интенсивность l^0 потока заявок на выходе источника заявок $\mathbf{S_0}$ для разомкнутой сети или число **M** заявок, циркулирующих в замкнутой сети.

Матрица вероятностей передач **[p]**, содержащая как в случае разомкнутой так и замкнутой сети n + 1 строку n + 1 столбец является важным параметром сети, однако не может быть непосредственно использована для получения характеристики сети и составляющих ее СМО, поскольку для этого необходимо знание интенсивностей потоков заявок на входах СМО $\mathbf{S_i}$, **i = 1,…n**.

$$[p_{ij}] = \begin{array}{c|cccccc} & S_0 & S_1 & \dots & S_j & \dots & S_n \\ S_0 & 0 & p_{01} & \dots & p_{0j} & \dots & p_{0n} \\ S_1 & p_{10} & p_{11} & \dots & p_{1j} & \dots & p_{1n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ S_i & p_{i0} & p_{i1} & \dots & p_{ij} & \dots & p_{in} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ S_n & p_{n0} & p_{n1} & \dots & p_{nj} & \dots & p_{nn} \end{array} \qquad (3.48)$$

Возникает вопрос о трансформации входящего потока заявок с интенсивностью λ^o ко входам составляющих сеть СМО. Будем рассматривать только установившийся режим.

Обозначим через a_i, $i = 1,\dots n$ коэффициент передачи (трансформации) входящего потока заявок ко входу системы S_i, $i = 1,\dots n$, количественно равный среднему числу появлений произвольной заявки из входящего потока сети во входящем потоке СМО S_i, иначе говоря равный среднему числу обслуживаний, получаемых произвольной заявкой в СМО S_i за все время пребывания ее в сети. Тогда интенсивность входящего потока СМО S_i, , может быть выражена через интенсивность входящего потока сети λ^o:

$$\lambda_i = a_i \lambda^o, \qquad i = 1,\dots n. \qquad (3.49)$$

Поскольку рассматриваемая сеть без потерь, интенсивности выходящих потоков СМО S_i совпадают с интенсивностями их входящих потоков. Интенсивность входящего потока СМО S_j равна сумме долей потока, поступающих от источника заявок S_0 и от других СМО сети:

$$\lambda_j = p_{0j} \times \lambda^o + \sum_{i=1}^{n} (p_{ij} \times \lambda_i) \quad j = 1,\dots n \qquad (3.50)$$

Выражая интенсивность λ_i входящих и выходящих потоков СМО S_i через интенсивность источника заявок и сокращая λ^o в обеих частях равенства, получим систему линейных неоднородных алгебраических уравнений относительно искомых коэффициентов передач a_i, имеющую единственное решение независимо от того, является ли рассматриваемая сеть разомкнутой или замкнутой.

$$\begin{array}{l} (p_{11} - 1) \times a_1 + p_{21} a_2 + \dots + p_{n1} a_n = -p_{01} \\ p_{12} a_1 + (p_{22} - 1) \times a_2 + \dots + p_{n2} a_n = -p_{02} \\ \text{- - - - - - - - - - - - - - - - - - - -} \\ p_{1n} a_1 + p_{2n} a_2 + \dots + (p_{nn} - 1) \times a_n = -p_{0n} \end{array} \qquad (3.51)$$

Поскольку интенсивность источника заявок λ^o известна, анализ сети сводится к анализу совокупности независимых систем массового обслуживания S_i, интенсивности входящих потоков которых λ_i выражены через интенсивность входящего потока сети λ^o и коэффициенты передачи a_i, .

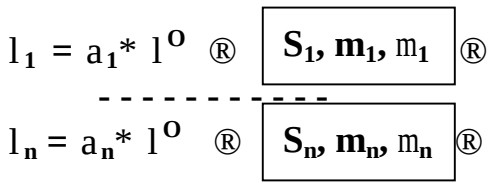


Рис. 3.15. Процедура анализа СеМО

Каждая из составляющих сеть СМО представляет собой СМО без потерь с числом каналов обслуживания $\mathbf{m_i}$, характеризующихся интенсивностью потока обслуживаний μ_i . Однако разложение сети на независимые СМО возможно лишь при условии существования установившегося режима в каждой из СМО сети, имеющего вид:

$$\frac{\lambda_i}{\mathbf{m}_i \mu_i} < 1. \qquad (3.52)$$

Поскольку $\lambda_i = a_i, \lambda^o$ то данное выражение приводится к виду:

$$\lambda^o < \frac{\mathbf{m}_i \mu_i}{a_i}. \qquad (3.53)$$

Данное неравенство позволяет сформулировать ограничение сверху на интенсивность входящего потока сети λ^o из условия существования установившегося режима в стохастической сети

$$\lambda^o < \mathbf{min}\left[\frac{\mathbf{m}_i \mu_i}{a_i}\right], \qquad \mathbf{i = 1,\ldots n} \qquad (3.54)$$

Если установившийся режим в сети существует, то для каждой из составляющих сеть СМО $\mathbf{S_i}$ нетрудно найти показатели эффективности $\mathbf{t_{ожi}}$, $\mathbf{t_{ci}}$, $\mathbf{l_i}$, $\mathbf{K_i}$, пользуясь выражениями, приведенными ранее.

На основании показателей эффективности отдельных СМО находятся показатели эффективности сети в целом:

$$\bar{t}_{ож} = \sum_{i=1}^{n} (a_i \times \bar{t}_{ожi}); \qquad \bar{t}_C = \sum_{i=1}^{n} (a_i \times \bar{t}_{Ci}) \quad \bar{l} = \sum_{i=1}^{n} (\bar{l}_i); \qquad \overline{K} = \sum_{i=1}^{n} (\overline{K}_i);$$

$$\mathbf{p_{SI}} = \frac{\overline{K}}{\sum_{i=1}^{n} \mathbf{m}_i} = \frac{\sum_{i=1}^{n} \overline{K}_i}{\sum_{i=1}^{n} \mathbf{m}_i}. \qquad (3.55)$$

3.3. ВЕРОЯТНОСТНЫЕ АВТОМАТЫ

Вероятностный автомат является типичным представителем стохастической динамической системы с дискретным временем.

Как было отмечено выше, для детерминированного конечного автомата функция переходов вида (2.11)

$$\mathbf{z(t_j) = j\,[\,z(t_{j-1}), x(t_j)]}$$

позволяет определить состояние автомата $\mathbf{z(t_i)}$ в момент $\mathbf{t_i}$ по предыдущему состоянию $\mathbf{z(t_{j-1})}$ и входному сигналу $\mathbf{x(t_j)}$.

Представим себе ситуацию, при которой состояние $\mathbf{z(t_{j-1})}$ и входной сигнал $\mathbf{x(t_j)}$ определяют не конкретное состояние $\mathbf{z(t_j)}$, а распределение вероятностей $\mathbf{P_{ij}}$ перехода из состояния $\mathbf{z(t_{j-1})}$ в одно из состояний $\mathbf{z_j}$ Î $\mathbf{Z}$ в момент $\mathbf{t_i}$ под действием входного сигнала $\mathbf{x(t_j)}$. То есть мы приходим к стохастической динамической системе, которую называют ***автоматом со случайными переходами***.

Другими словами, в отличие от детерминированных автоматов, у которых выполнено условие однозначности переходов, у автоматов со случайными переходами при заданном состоянии и заданном входном сигнале возможен переход с заданной вероятностью в различные состояния.

Для того чтобы задать автомат со случайными переходами, необходимо указать совокупность матриц вида $\| \mathbf{P_{ij}(x)} \|$, элементы которых представляют собой условные вероятности переходов. Кроме того, естественно, нужно представить функцию выходов вида (2.12) или (2.13):

$$\mathbf{y(t_j)} = y\mathbf{[\, z(t_{j-1}), x(t_j)]}, \qquad \mathbf{y(t_j)} = y\mathbf{[\, z(t_j)]}.$$

Автомат со случайными переходами функционирует следующим образом.

1. В момент $\mathbf{t_i}$ в автомат поступает входной сигнал $\mathbf{x(t_j)} = \mathbf{x^*}$.
2. По значению $\mathbf{x^*}$ из совокупности матриц $\| \mathbf{P_{ij}(x)} \|$ выбирается конкретная матрица $\| \mathbf{P_{ij}(x^*)} \|$.
3. По значению $\mathbf{z(t_{j-1})} = \mathbf{z_{i^*}}$ состояния автомата из этой матрицы выбирается строка

$$\mathbf{p_{i^*1}, p_{i^*2}, \ldots p_{i^*n}}, \qquad (3.56)$$

где

$$\sum_{j=1}^{n} \mathbf{p_{i^*j}} = \mathbf{1}.$$

4. В соответствии с вероятностями (3.56) выбирается по жребию конкретное состояние $\mathbf{z_j = z(t_j)}$ и вычисляется при помощи функции выходов соответствующий выходной сигнал.

Математическое описание автомата со случайными переходами, имея в виду только процесс перехода в новое состояние, можно свести к функции вида (2.11):

$$\mathbf{z(t_j)} = j\mathbf{[\, z(t_{j-1}), x(t_j)]},$$

если под j понимать случайную функцию, определяющую состояния $\mathbf{z\,(t_i)}$ с вероятностями (3.56).

Для фактического выбора состояния по жребию может быть использован датчик случайных чисел, генератор «белого шума», а также любой физический случайный процесс.

Автомат со случайными переходами, как стохастическая динамическая система с дискретным временем, представляет собой удобную схему для фор-

мального описания широкого класса объектов материального мира. Однако существуют и другие разновидности вероятностных автоматов.

Рассмотрим другой тип стохастической динамической системы с дискретным временем. Пусть опять речь идет о детерминированном конечном автомате с заданными функциями переходов и выходов. Если начальное состояние $\mathbf{z(t_0)}$ автомата считать не фиксированным, а случайным, то приходим к стохастической схеме, которую называют ***автоматом со случайными начальными состояниями***. Начальное состояние такого автомата соответствует распределению вероятностей

$$p = (p_1, p_2, \ldots, p_n), \qquad (3.57)$$

где $\mathbf{0 \le p_i \le 1;}$ $\mathbf{i = 1,2,\ldots n;}$ $\sum_{i=1}^{n} p_i = 1$.

Наконец, существует третий тип стохастической динамической системы с дискретным временем. Если функция выходов определяет не конкретный выходной сигнал $\mathbf{y(t_i) \in Y}$, а лишь распределение вероятностей на множестве **Y**, в соответствии с которыми выбирается выходной сигнал в момент $\mathbf{t_i}$, то мы имеем ***автомат со случайными выходами***. Чтобы задать такой автомат, необходимо наряду с функцией переходов вида (4.1) указать совокупность матриц вида $\| \mathbf{q_{ij}(x)} \|$, элементы которых представляют собой условные вероятности того, что при заданном состоянии и заданном входном сигнале возможен тот или иной выходной сигнал.

Перейдем теперь к общему определению вероятностного автомата.

Вероятностным автоматом **A** называется объект, определяемый множествами **X, Y, Z** и семейством матриц **{M (y/x)}.** Если $\mathbf{X=\{x_1,x_2,...,x_l\}}$ – входной алфавит, $\mathbf{Y=\{y_1,y_2,...,y_m\}}$ – выходной алфавит, $\mathbf{Z=\{z_1,z_2,...,z_n\}}$ – множество состояний, то **{M (y/x)}** – семейство $\mathbf{l \times m}$ матриц размерностью $\mathbf{n \times n}$.

Элемент $\mathbf{\mu_{ij}(y_p/x_r)}$ матрицы $\mathbf{M(y_p/x_r)}$ есть вероятность того, что, находясь в состоянии $\mathbf{z_i}$ и получив входной сигнал $\mathbf{x_r}$, автомат перейдет в состояние $\mathbf{z_j}$, а выходной сигнал будет $\mathbf{y_p}$.

Если $\mathbf{\mu_{ij}(y_p/x_r)}$ принимают только значения единица или нуль, имеем частный случай вероятностного автомата – обычный детерминированный конечный автомат.

Когда $\mathbf{y_p}$ и $\mathbf{z_j}$ – условно-независимые случайные элементы (величины, векторы и т. д.) при фиксированных $\mathbf{z_i}$ и $\mathbf{x_r}$, говорят, что **A** – вероятностный автомат Мили. Если же выходной сигнал $\mathbf{y_p}$ зависит только от финального состояния $\mathbf{z_j}$ и не зависит от переходов, приводящих автомат в это состояние, то **A** – вероятностный автомат Мура.

Пример 3.6

Некоторая технологическая линия выпускает изделия. В каждый такт $\mathbf{t_j}$, (j = 1,2, . . .) работы линии становится известной информация о готовности очередного изделия. Длительность изготовления очередного изделия во много раз превышает длительность такта $\mathbf{t_j}$. Готовое изделие с вероятностью **0,9** ока-

зывается годным и с вероятностью **0,1** – бракованным. Необходимо описать рассматриваемый процесс производства на некотором конечном интервале времени как автомат со случайными переходами.

Решение.

Состояния автомата – числа $\mathbf{z(t_j)}$ – количество годных изделий, выпущенных технологической линией к данному моменту времени $\mathbf{t_j}$.

Входной сигнал **x** имеет смысл числа готовых изделий, выдаваемых линией в момент $\mathbf{t_j}$. Если изделие не готово к очередному моменту $\mathbf{t_j}$, то $\mathbf{x(t_j) = 0}$, если оно готово, то $\mathbf{x(t_j) = 1}$.

Функцию переходов можно записать следующим образом:

$$\mathbf{z(t_i) = z(t_{i-1}) + l x(t_i),}$$

где

$$l = \begin{cases} \mathbf{1\_c\_вероятностью\_0{,}9,\_если\_x(t_j) = 1} \\ \mathbf{0\_c\_вероятностью\_0{,}1,\_если\_x(t_j) = 1} \\ \mathbf{не\_определено,\_если\_x(t_j) = 0} \end{cases}$$

В качестве выходного сигнала автомата возьмем число изделий, выданных линией к данному моменту. Тогда функция выходов приобретает вид:

$$\mathbf{y(t_j) = z(t_j).}$$

Приведенные выше выражения можно использовать для вычисления элементов матриц переходов. Всего в совокупность матриц войдут 2 матрицы:

Для $\mathbf{x(t_j) = 0}$

$$\mathbf{P_{ij}} = \begin{vmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{...} & \mathbf{...} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{...} & \mathbf{...} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{...} & \mathbf{...} \\ - & - & - & - & - & - \end{vmatrix}$$

Для $\mathbf{x(t_j) = 1}$

$$\mathbf{P_{ij}} = \begin{vmatrix} \mathbf{0{,}1} & \mathbf{0{,}9} & \mathbf{0} & \mathbf{0} & \mathbf{...} & \mathbf{...} \\ \mathbf{0} & \mathbf{0{,}1} & \mathbf{0{,}9} & \mathbf{0} & \mathbf{...} & \mathbf{...} \\ \mathbf{0} & \mathbf{0} & \mathbf{0{,}1} & \mathbf{0{,}9} & \mathbf{0} & \mathbf{...} \\ - & - & - & - & - & - \end{vmatrix}$$

Матрицы выходных сигналов будут иметь аналогичный вид.

4. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ

Имитационное моделирование заключается в логико-аналитической имитации функционирования системы. При этом для детерминированных систем определяются изменения их состояния во времени под влиянием внешних воздействий. Для стохастических систем, помимо информации об изменениях состояния системы, получают выборки значений выходных параметров, по которым определяются их основные вероятностные характеристики.

Имитационное моделирование – это метод исследования, который основан на том, что анализируемая динамическая система заменяется имитатором и с ним производятся эксперименты для получения информации о поведении изучаемой системы. Роль имитатора обычно выполняет программа ЭВМ.

4.1. ОПРЕДЕЛЕНИЕ МЕТОДА ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

В отличие от аналитических моделей, которые предполагают наличие математического описания процессов, протекающих в оригинале и которые обычно строятся при жестких ограничениях на параметры, имитационные модели являются более универсальными и могут быть построены при отсутствии математического описания оригинала. Идея имитационного моделирования очень проста и заключается в том, что строится некий алгоритм поведения подсистем и отдельных элементов систем во времени. Этот алгоритм может быть реализован в виде программы для ЭВМ. Многократно «прогоняя» имитационную модель (ИМ) в условиях случайных потоков событий на входе и в самой системе, можно накопить статистическую информацию об изменении переменных состояния ИМ. Статистическая обработка этой информации позволяет получить статистические оценки показателей эффективности.

Однако, в отличие от аналитической модели, ИМ обладает принципиальной методической погрешностью, существенно зависящей от объема выборки и, соответственно, от времени наблюдения за ИМ.

Проиллюстрируем сказанное.

В качестве оригинала (моделируемой системы) возьмем однопроцессорную систему оперативной обработки, на которую поступает случайный поток заявок, а возможность буферирования запросов (создания очереди) отсутствует. Продолжительность обслуживания заявки также является случайной величиной. Требуется определить параметры работы системы (рис. 4.1).

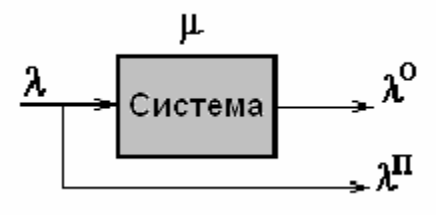


Рис. 4.1

Если предположить, что потоки событий являются простейшими, то можно использовать известную модель одноканальной СМО без очереди (рис. 4.2). При этом справедливы следующие зависимости для расчета вероятностей состояний:

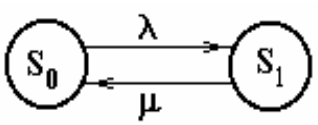
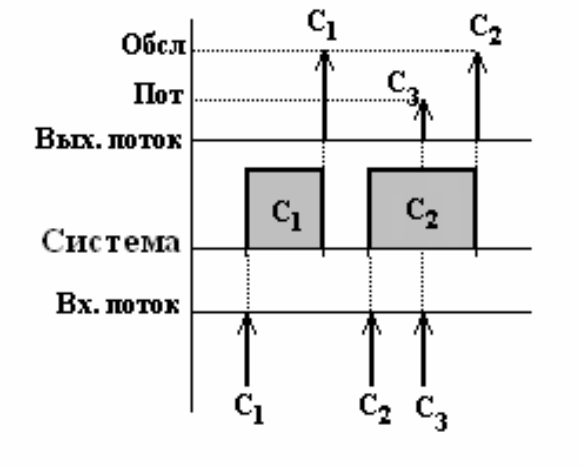


Рис. 4.2

$$p_0 = \frac{m}{l+m}, \qquad p_1 = 1 - p_0 = \frac{l}{l+m}.$$

Однако в этом случае делается много упрощающих допущений, относительно характеристик входного потока требований и характеристик обслуживания.

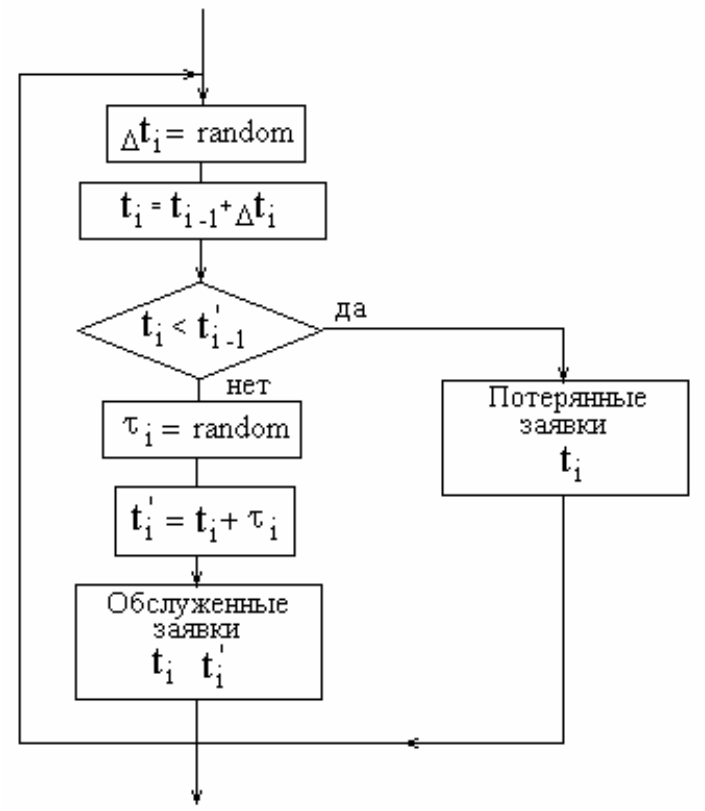
Рассмотрим процедуру имитационного моделирования данной системы.



Специальным образом генерируется входящий поток заявок, при этом момент поступления каждой заявки является случайным событием. Если на момент поступления заявки система свободна, то она поступает на обслуживание и при этом генерируется время обслуживания. Если система занята, то заявка покидает систему. Таким образом, создаются выходящие потоки обслуженных и потерянных заявок (рис. 4.3).

Рис. 4.3

Фрагмент блок-схемы программы имитации системы может быть, например, таким (рис. 4.4).



На схеме для требования с номером **i** обозначено:

t_i - момент поступления требования;

t_i' - момент окончания обслуживания;

t_i - продолжительность обслуживания;

Рис. 4.4

В общем случае, для определения статистических параметров какой-либо случайной величины **y**, которая является функцией **y=f(**a,b,....,w) случайных величин a,b,....,w с известными функциями распределения, можно применить следующий алгоритм:

1) по каждой из величин a,b,....,w производится случайное испытание, в результате каждого определяется одно конкретное значение случайных величин $a_i, b_i, ..., w_i$;

2) используя найденные величины, определяется одно частное значение y_i по вышеприведенной зависимости;

3) предыдущие операции повторяются N раз, в результате чего определяется N значений случайной величины **y**;

4) на основании N значений величины находятся статистические параметры искомой величины **y** или её эмпирическая функция распределения**.**

4.2. ОСНОВНЫЕ ПОНЯТИЯ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

Имитационное моделирование технических и информационных систем предполагает использование следующих обстоятельств.

Модель системы всегда представляет собой совокупность моделей элементов и их функциональные взаимосвязи. Модель элемента системы – это обычно наборы правил (алгоритмов) поведения элемента по отношению к входным воздействиям и изменений состояний элемента.

Если элемент отображает функциональное устройство на том или ином уровне детализации, то в простейшем случае он может находиться в работоспособном состоянии или в состоянии отказа. В работоспособном состоянии элемент может быть занят (например, выполнением операции) или быть свободным. К правилам поведения устройств вычислительной техники относятся правила выборки заявок из очереди; реакция устройства на поступление заявки, когда устройство занято или к нему имеется очередь заявок; реакция устройства на возникновение отказа в процессе обслуживания заявки и некоторые другие.

Таким образом, целями имитационного моделирования являются:

- воспроизведение с необходимой достоверностью поведения отдельных элементов системы в процессе реализации ею функции системы;
- - накопление статистических данных о поведении элементов;
- - последующая статистическая обработка этих данных для получения статистических оценок, количественных характеристик, законов распределения оцениваемых показателей эффективности.

При имитационном моделировании информационных и вычислительных систем, характеризующихся стохастическим поведением, возможны следующие аспекты их анализа:

- при анализе производительности случайным является поток запросов на обслуживание, случайна также трудоемкость запроса, определяемая числом

операций, которые необходимы для выполнения программы, обрабатывающей запрос пользователя;
- при анализе надежности случайны процессы отказов элементов системы, случайны также интервалы времени, необходимого для ремонта отказавших элементов.

Имитационное моделирование информационных и вычислительных систем может использовать различные уровни абстракции:
- системное моделирование - моделирование на уровне системы (анализ производительности отдельных элементов и системы в целом);
- моделирование на уровне архитектуры ЭВМ и регистровых передач (анализ эффективности системы команд, анализ корректности микропрограмм операций и пр.);
- моделирование на уровне логических схем;
- схемо-техническое моделирование.

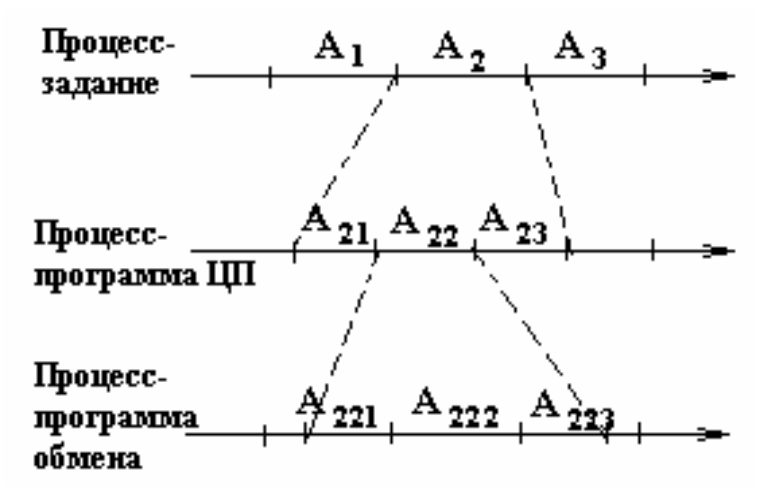
При имитационном моделировании информационных объектов в основном применяется системный уровень абстракции, при котором исследуется такой важный показатель эффективности, как производительность. Существует ряд понятий имитационного моделирования системного уровня.

Активность - элементарная работа, рассматриваемая в рамках данной имитационной модели как неделимая.

Процесс - логически связанная последовательность активностей.

Событие - факт начала или завершения некоторой активности.

Понятие активности и процесса взаимосвязаны. Переходя с одного уровня детализации на другой, можно рассматривать активность как процесс более низкого уровня и, напротив, рассматривать процесс как активность более высокого уровня. Пример использования различных уровней детализации приведен на рис. 4.5.



A_1 – инициация задания;
A_2 – исполнение задания;
A_3 – завершение задания.

A_{21} – выполнение вычислений;
A_{22} – обмен данными с диском;
A_{23} – выполнение вычислений.

A_{221} – позиционирование;
A_{222} - ожидание зоны обмена;
A_{223} - обмен данными.

Рис. 4.5

Имитационная модель системного уровня предполагает описание функционирования системы в терминах активности, событий или процессов на некотором языке программирования. В принципе модель может быть построена на любом универсальном языке программирования, однако компактность и соответственно время разработки модели возрастают при использовании специальных языков и пакетов имитационного моделирования.

4.3. ОСНОВНЫЕ ЭТАПЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

Процедура имитационного моделирования системы включает в себя следующие основные этапы.

1. Постановка задачи.

Так же как и при любом другом методе моделирования, на этом этапе осуществляется выбор свойств системы, которые подлежат отражению в имитационной модели и отбрасывание тех свойств, которые на данном этапе исследования можно считать несущественными. На этом этапе необходим сбор исходной информации о выбранных свойствах системы. Источниками сведений обычно являются: научно-техническая и справочная литература, результаты экспериментальных исследований и пр.

2. Формирование и построение концептуальной модели.

При исследовании характеристик информационных и вычислительных систем концептуальной моделью является обычно СМО. Для формализации может быть применен язык схем с очередями. Объектами модели в данном случае могут являться:

- источники заявок;
- каналы обслуживания;
- очереди с ограниченным или с бесконечным числом мест.

Далее на основе типовых схем имитационного моделирования и логических предпосылок создаются алгоритмы, формализующие реакцию системы на происходящие события.

3. Выбор языка ИМ и построение модели.

На данном этапе имитирующие алгоритмы записываются на каком-либо специализированном или универсальном языке.

4. Тестирование ИМ.

Этап предполагает проверку правильности функционирования имитационной модели на инструментальной ЭВМ обычно с использованием контрольных примеров.

5. Планирование эксперимента.

Цель этапа: построить план эксперимента, обеспечивающий получение максимальной информации об объекте с минимальными затратами ресурсов инструментальной ЭВМ.

6. Модельный эксперимент.

Этап предполагает накопление результатов моделирования и их статистическую обработку.

7. Интерпретация результатов модельного эксперимента.

Здесь производится анализ полученных результатов, делаются выводы об их достоверности и адекватности, устанавливаются области применения полученных данных.

Для описания имитационного процесса весьма актуален вопрос выбора языка программирования. Теоретически возможно описать модель на любом из широко распространенных универсальных языков Фортране, Паскале и пр. Однако опыт развития теории и практики имитационного моделирования показывает, что наиболее эффективным средством являются специальные имитационные языки, которых к настоящему времени создано уже немало и многие из них эффективно используются, особенно за рубежом, где ни один крупный проект не реализуется без проверки на имитационной модели. Наиболее известны языки: GPSS, GASP, SIMSCRIPT и DYNAMO, реализующие различные подходы к моделированию. Одним из наиболее распространенных специализированных языков имитационного моделирования является язык GPSS (General Purpose System Simulator)**.**

Язык GPSS предназначен для моделирования систем с материальными и информационными потоками. Можно описывать систему и управлять маршрутами прохождения через объекты системы специальных динамических элементов - транзактов (транзакт, транзакция - сообщение). Транзакт может восприниматься как динамическая единица материального или информационного потока, способная перемещаться от объекта к объекту и имитировать последовательность обслуживания, которую получает транзакт за время пребывания его в системе.

4.4. ВРЕМЯ В ИМИТАЦИОННЫХ МОДЕЛЯХ. ПСЕВДОПАРАЛЛЕЛИЗМ

Поскольку функционирование систем происходит во времени, важной проблемой имитационного моделирования становится моделирование времени.

При имитационном моделировании различают три вида времени: физическое, модельное и инструментальное.

- Физическое время $\mathbf{t_ф}$ – это реальное непрерывное время в объекте моделирования, где происходят и развиваются события и процессы.
- Модельное время $\mathbf{t_м}$ – это время в модели, которое обычно дискретно. Оно характеризует моменты наступления событий и продолжительность процессов в модели. Часто его рассматривают, как безразмерное.
- Инструментальное время $\mathbf{t_и}$ – это реальное время, в течение которого на инструментальной ЭВМ реализуется имитационная модель.

Переход от реального физического времени к квантованному модельному часто приводит к тому, что события, происходящие в разные моменты физического времени, могут быть отнесены к одному моменту модельного времени (рис. 4.6).

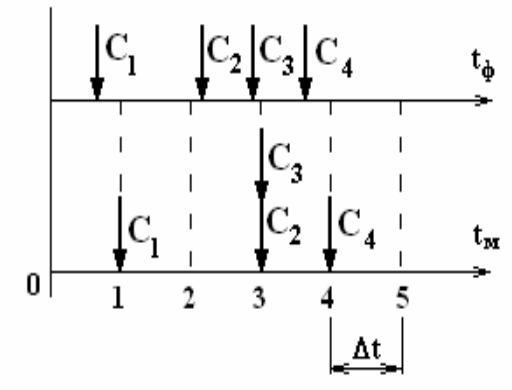


Рис. 4.6

Рассмотрим взаимосвязь трех видов времени. Пусть в исследуемой системе в отдельные моменты физического времени наступают события трех типов **a, b, c**. Моменты наступления событий в модельном времени будут зависеть от выбранного шага квантования. Каждому из событий в имитационной модели будут соответствовать определенные процедуры, требующие выполнения на инструментальной ЭВМ соответствующих программ и требующие инструментального времени. Последовательность обработки этих событий в инструментальном времени определяется приоритетами событий. Пусть в нашем примере приоритеты событий следующие: **b > c > a**. Взаимосвязь трех видов времени представлена на рис. 4.7.

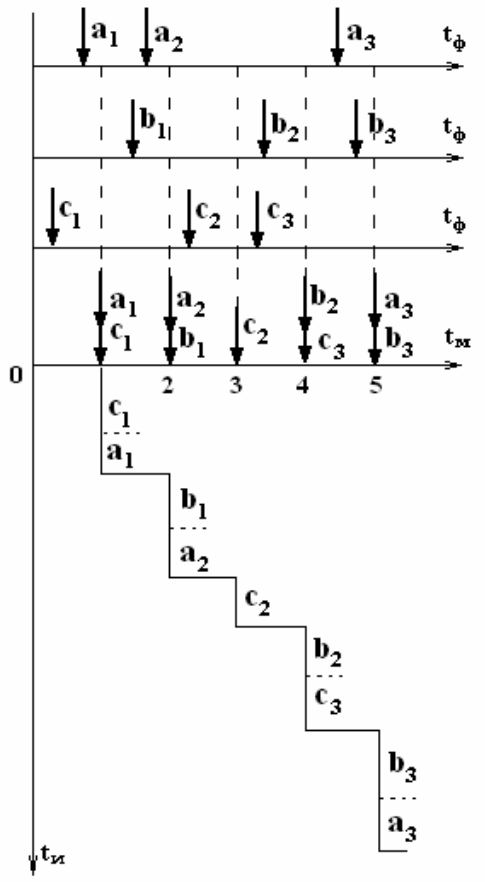


Рис. 4.7

В рассмотренном примере в физическом времени параллельно происходили события трех типов. Проблема необходимости имитировать реальные параллельные процессы на инструментальной ЭВМ, которая обычно может реализовать лишь последовательные процессы (рассматриваются однопроцессорные компьютеры), называется проблемой псевдопараллелизма. Для реализации псевдопараллелизма надо определенным образом организовать последовательное выполнение процедур, относящихся к различным активностям, процессам или событиям, протекающим параллельно во времени.

4.5. ОБОБЩЁННЫЕ АЛГОРИТМЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

Алгоритм моделирования по принципу особых состояний

Предположим, исследуется система, состоящая из двух устройств ввода УВв1 иУВв2, центрального устройства ЦУ и устройства вывода УВыв (рис. 4.8).

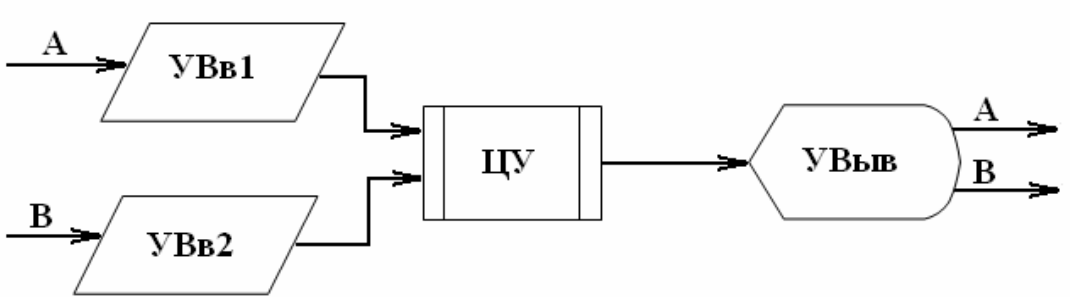


Рис. 4.8. Исследуемая система

Через устройство ввода УВв1 поступает поток заданий А, через УВв2 – поток заданий В. Эти задания обрабатываются центральным устройством, и результаты выдаются на УВыв. Потоки заданий независимы. Периоды следования заявок t_A и t_B, длительность обслуживания заявок в **k**-м устройстве T_A^k, T_B^k случайны, но известны их функции распределения. Требуется определить времена загрузки каждого устройства и времена реакции по каждому из потоков.

Процесс имитационного моделирования работы системы может происходить, например, следующим образом.

1. Определяются моменты поступления в систему 1-й заявки из каждого потока t_{A1} и t_{B1}. Для этого сначала по результатам случайного испытания в соответствии с функцией распределения периода следования заявок соответствующего потока определяются величины t_{A1} и t_{B1}, которые складываются с начальным моментом времени:

$$t_{A1} = 0 + t_{A1}, \qquad t_{B1} = 0 + t_{B1}.$$

Допустим, эти моменты совпали с первым и третьим квантами модельного времени (рис. 4.9):

$$t_{A1} = t_1^M, \qquad t_{B1} = t_3^M.$$

Таким образом, в системе могут наступить два события – поступление в систему заявок из первого или второго потоков.

2. Находится момент наступления наиболее раннего события, т.е. минимальное время. В примере это время $\mathbf{t_1^M}$. Определяется тип события. Поскольку это поступление заявки, то она должна обслуживаться на устройстве УВв1. Методом случайного испытания определяется время ее обслуживания устройством ввода УВв1 $\mathbf{T_{A1}^{УВв1}}$ и отмечается момент окончания обслуживания:

$$\mathbf{t_{A1}^{УВв1} = t_1^M + T_{A1}^{УВв1}}.$$

Допустим, этот момент совпал с пятым квантом модельного времени

$$\mathbf{t_{A1}^{УВв1} = t_5^M}.$$

Устройство УВв1 переходит в состояние «занято». Одновременно определяется момент поступления следующей заявки потока А:

$$\mathbf{t_{A2} = t_1^M + t_{A2}}.$$

Допустим, этот момент совпал с двенадцатым квантом модельного времени:

$$\mathbf{t_{A2}^{УВв1} = t_{12}^M}.$$

Таким образом, в системе могут наступить три события – поступление первой заявки из потока В, окончание обслуживания первой заявки потока А на устройстве УВв1 и поступление второй заявки потока А. (рис. 4.10).

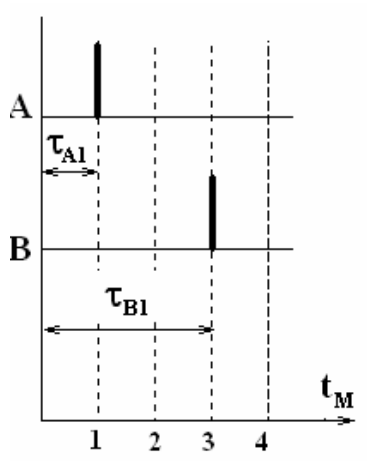


Рис. 4.9

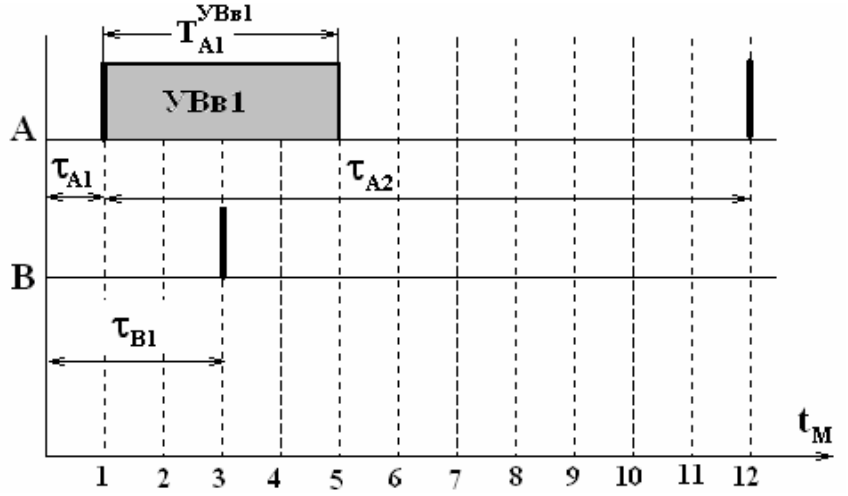


Рис. 4.10

3. Находится минимальное время, т.е. момент наступления наиболее раннего события из рассматриваемых. В примере это время поступления первой заявки потока В – $\mathbf{t_3^M}$. Согласно типу наступившего события она должна быть обслужена на устройстве УВв2. Для этой заявки методом случайного испытания определяется время ее обслуживания устройством ввода УВв2 $\mathbf{T_{B1}^{УВв2}}$ и отмечается момент окончания обслуживания:

$$\mathbf{t_{B1}^{УВв2} = t_3^M + T_{B1}^{УВв2}}.$$

Допустим, этот момент совпал с четвертым квантом модельного времени:

$$\mathbf{t_{B1}^{УВв2} = t_4^M}.$$

Устройство УВв2 переходит в состояние «занято». Одновременно определяется момент поступления следующей заявки потока В:

$$t_{B2} = t_3^M + t_{B2}.$$

Допустим, этот момент совпал с восьмым квантом модельного времени:

$$t_{B2} = t_8^M.$$

Таким образом, в системе могут наступить четыре события – поступление второй заявки из потока В, поступление второй заявки из потока А, окончание обслуживания первой заявки потока А на устройстве УВв1 и. окончание обслуживания первой заявки потока В на устройстве УВв2 (рис. 4.11).

4. Снова находится минимальное время. В примере это время окончание обслуживания первой заявки потока В на устройстве УВв2 – t_4^M. Согласно типу наступившего события заявки поступает на обслуживание центральным устройством. Определяется время ее обслуживания $T_{B1}^{ЦУ}$ и отмечается момент окончания обслуживания:

$$t_{B1}^{ЦУ} = t_4^M + T_{B1}^{ЦУ}.$$

Допустим, этот момент совпал с шестым квантом модельного времени:

$$t_{B1}^{ЦУ} = t_6^M.$$

Центральное устройство переходит в состояние «занято».

Таким образом, в системе могут наступить четыре события – поступление второй заявки из потока В, поступление второй заявки из потока А, окончание обслуживания первой заявки потока А на устройстве УВв1 и окончание обслуживания первой заявки потока В на центральном устройстве (рис. 4.12).

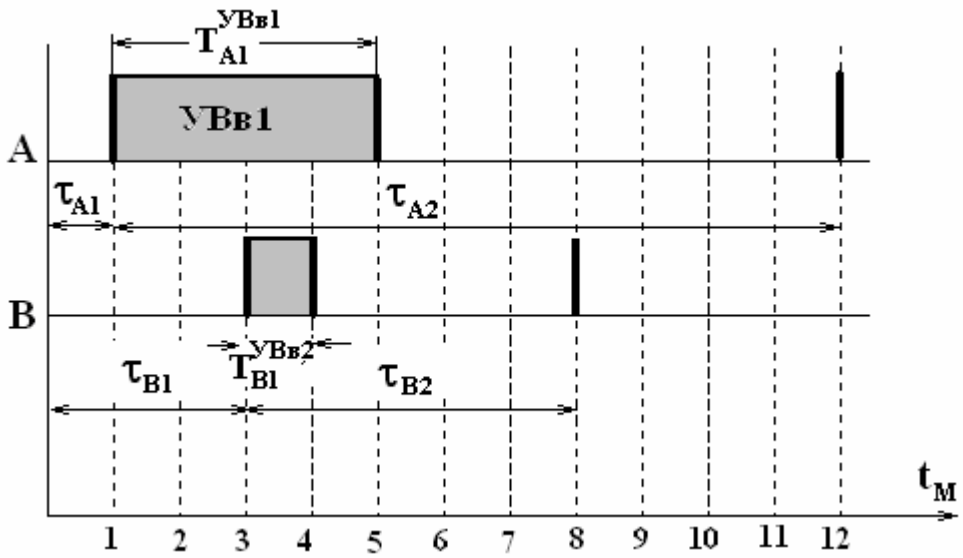


Рис. 4.11

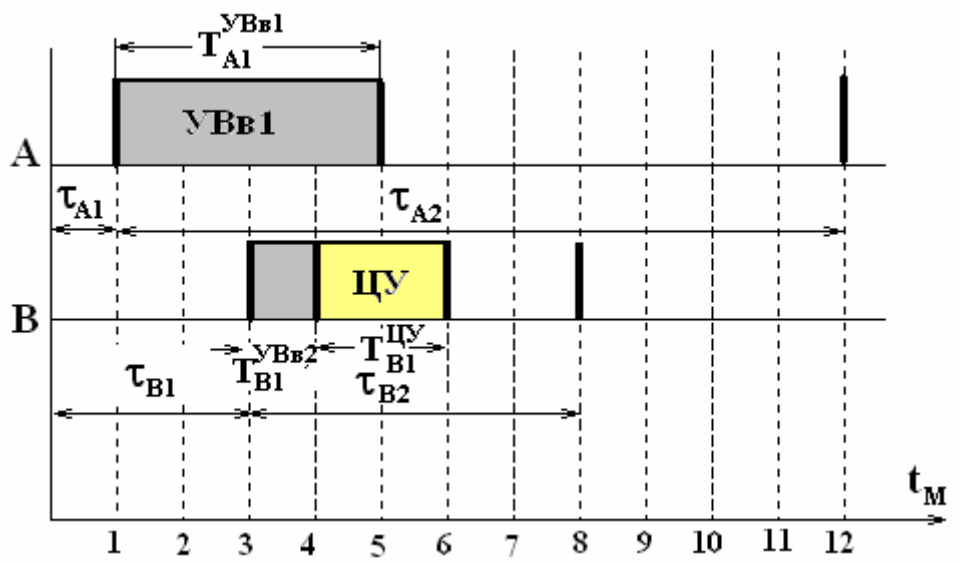


Рис. 4.12

5. Следующее минимальное время t_5^M - момент завершения обслуживания заявки потока А устройством УВв1. С этого момента заявка может начать обрабатываться центральным устройством, но оно занято обслуживанием потока В. Тогда заявка потока А переходит в состояние ожидания и становится в очередь.

6. Следующее минимальное время t_6^M - момент завершения обслуживания заявки потока В центральным устройством. С этого момента центральное устройство начинает обрабатывать заявку потока А, а заявка потока В переходит на обслуживание устройством вывода УВыв. Далее определяются соответствующие времена обслуживания: $T_{A1}^{ЦУ}$ и $T_{B1}^{УВыв}$, а также отмечаются моменты времени

$$t_{A1}^{ЦУ} = t_6^M + T_{A1}^{ЦУ}, \qquad t_{B1}^{УВыв} = t_6^M + T_{B1}^{УВыв}.$$

Допустим, эти моменты совпали с девятым и седьмым квантами модельного времени (рис. 4.13):

$$t_{A1}^{ЦУ} = t_9^M, \qquad t_{B1}^{УВыв} = t_7^M.$$

В момент t_7^M полностью завершается обработка первой заявки потока В. По разности времени t_7^M и t_3^M вычисляется время реакции по этой заявке

$$u_{B1} = t_7^M - t_3^M.$$

7. Следующий минимальный момент времени t_8^M - это появление 2-й заявки потока В. Определяет время поступления следующей заявки этого потока и, допустим, в нашем примере этот момент совпал с пятнадцатым квантом модельного времени:

$$t_{B3} = t_8^M + t_{B3}, \qquad t_{B3} = t_{15}^M.$$

Затем вычисляется время обслуживания 2-й заявки потока В на устройстве УВв2 - $T_{B2}^{УВв2}$ и отмечается момент окончания ее обслуживания на этом устройстве. Допустим, этот момент совпал с десятым квантом модельного времени:

$$t_{B2}^{УВв2} = t_8^M + T_{B2}^{УВв2}, \qquad t_{B2}^{УВв2} = t_{10}^M.$$

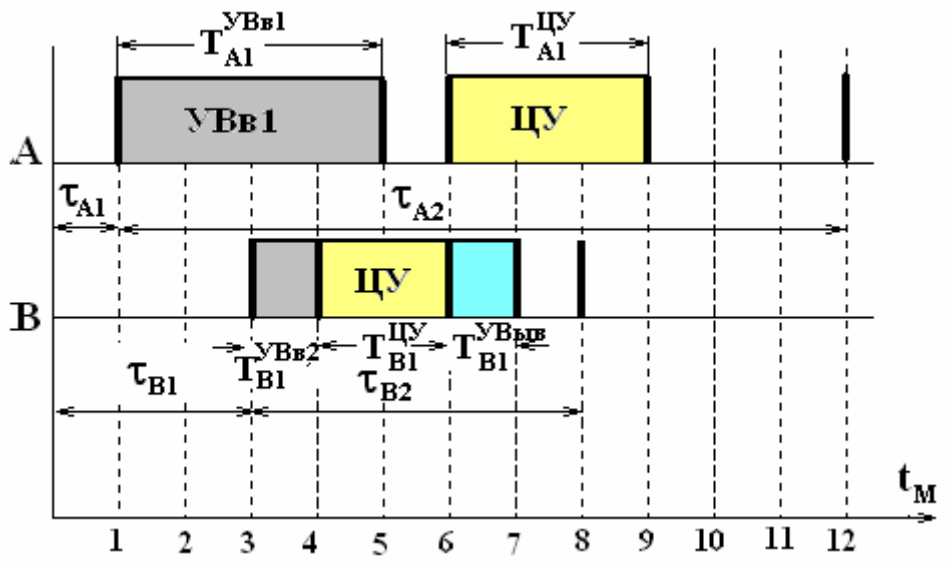


Рис. 4.13

8. Следующее минимальное время $\mathbf{t_9^M}$ - момент завершения обслуживания первой заявки потока А центральным устройством. В этот момент заявка начинает обслуживаться устройством УВыв. Определяется продолжительность обслуживания $\mathbf{T_{A1}^{УВыв}}$ по результатам случайных испытаний и отмечается момент окончания обслуживания, который, допустим, соответствует одиннадцатому кванту модельного времени:

$$\mathbf{t_{A1}^{УВыв} = t_9^M + T_{A1}^{УВыв}}, \qquad \mathbf{t_{A1}^{УВыв} = t_{11}^M}.$$

В момент времени $\mathbf{t_{11}^M}$ завершается полное обслуживание 1-й заявки потока А (рис. 4.14). Разность между этим моментом и моментом времени $\mathbf{t_1^M}$ – первое значение времени реакции по потоку А:

$$\mathbf{u_{A1} = t_{11}^M - t_1^M}.$$

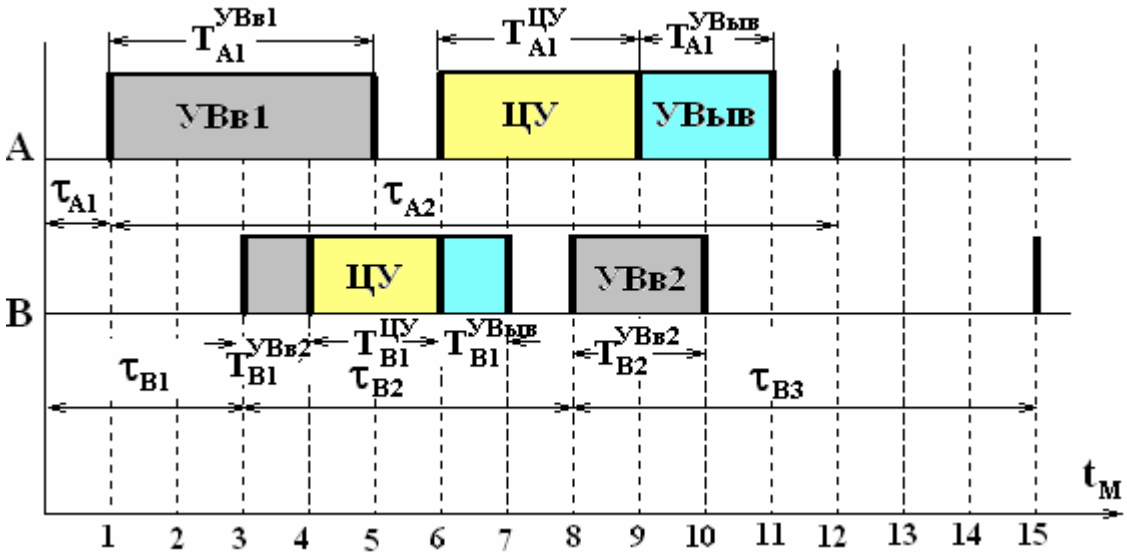


Рис. 4.14

Указанные процедуры выполняются до истечения времени моделирования. В результате получается некоторое количество (выборка) случайных значений времени реакции $\mathbf{u_A}$ и $\mathbf{u_B}$ по первому и второму потокам. По этим значениям могут быть определены эмпирические функции распределения и вычислены количественные вероятностные характеристики времени реакции. В процессе моделирования можно суммировать продолжительности занятости каждого устройства обслуживанием всех потоков. Если результаты суммирования разделить на время моделирования, то получатся коэффициенты загрузки устройств. Можно определить время ожидания заявок в очереди, обслуженных системой, среднюю и максимальную длину очереди заявок к каждому устройству, требуемую ёмкость памяти и др.

Таким образом, процесс имитации развивается с использованием управляющих последовательностей, определяемых по функциям распределения вероятностей исходных данных путём проведения случайных испытаний. В качестве управляющих последовательностей в примере использовались последовательности значений периодов следования заявок по каждому i-му потоку $\{t_i\}$ и длительности обслуживания заявок i-го потока k-м устройством $\{T_i^k\}$. Моменты наступления будущих событий определялись по простым рекуррентным соотношениям.

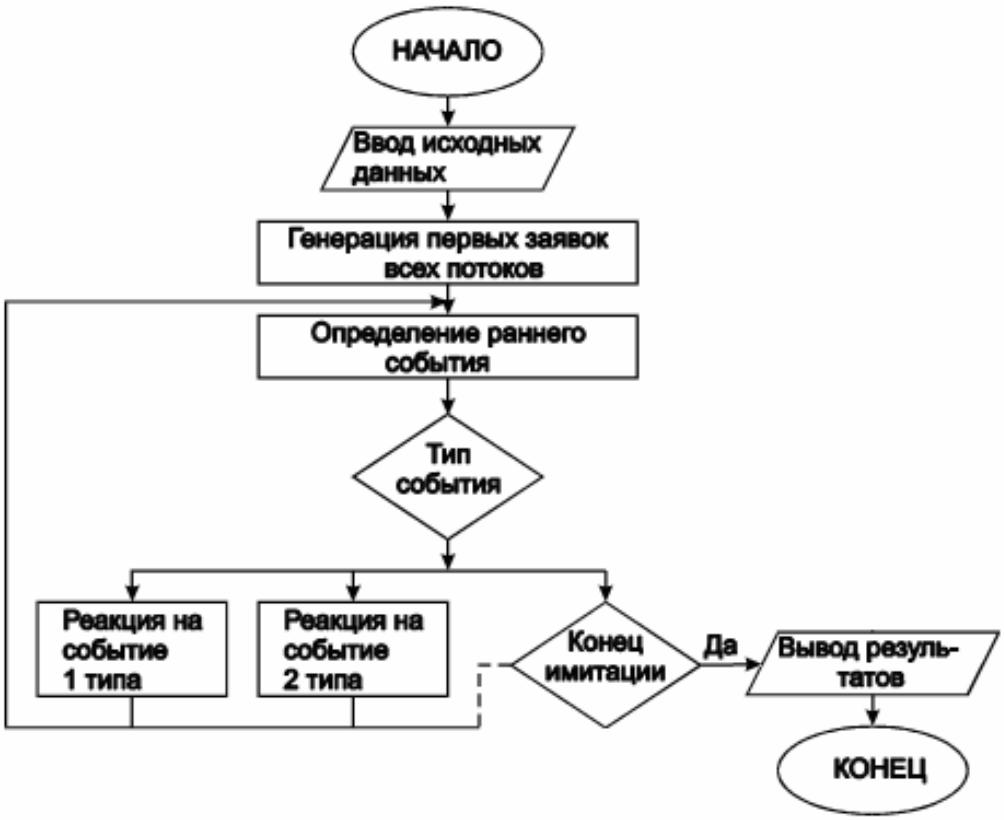


Рис. 4.15 Схема алгоритма моделирования по принципу особых состояний

Эта особенность даёт возможность построить простой циклический алгоритм моделирования, который сводится к следующим действиям:

1) определяется событие с минимальным временем –наиболее раннее событие;

2) модельному времени присваивается значение времени наступления наиболее раннего события;

3) определяется тип события;

4) в зависимости от типа события предпринимаются действия, направленные на загрузку устройств и продвижение заявок в соответствии с алгоритмом их обработки, и вычисляются моменты наступления будущих событий; эти действия называют реакцией модели на события;

5) перечисленные действия повторяются до истечения времени моделирования.

Обобщённая схема алгоритма моделирования по принципу особых состояний приведена на рисунке 4.15.

Алгоритм моделирования по принципу постоянного приращения модельного времени

В качестве объекта моделирования рассмотрим двухканальную СМО с очередью из двух мест. Заявки, поступающие в систему, равноправные. Дисциплина обслуживания FIFO. Каждый канал будем рассматривать целиком, т.е. без устройств ввода, обработки и вывода.

Данный алгоритм имеет много общего с приведенным выше. В частности, значения периодов следования заявок $\{t_i\}$ и длительности обслуживания заявок каждым каналом $\{T_i^k\}$ будут генерироваться по функциям распределения вероятностей исходных данных путём проведения случайных испытаний. Следует отметить, что в данном случае возможно генерирование всего потока заявок с самого начала процесса имитации.

Процесс имитационного моделирования работы системы может происходить, например, следующим образом.

1. Вводятся исходные данные и генерируются случайные моменты поступления заявок. Модельное время устанавливается на ноль.

2. Модельное время увеличивается на величину приращения **dt** (в примере это один квант модельного времени) и становится равным $\mathbf{t_1^M}$. Анализируется наличие событий в системе. В примере в данный момент поступила первая заявка. Согласно типу события выполняются действия: генерируется продолжительность обслуживания заявки и определяется момент его окончания. В примере это шестой квант модельного времени:

$$\mathbf{t_1^M + T_1^1 = t_6^M}.$$

Первый канал СМО переходит в состояние «занято».

3. Модельное время увеличивается на один квант и становится равным $\mathbf{t_2^M}$. Анализ показывает отсутствие событий в системе.

4. Модельное время увеличивается и становится равным $\mathbf{t_3^M}$. В систему поступает вторая заявка. Согласно типу события генерируется продолжитель-

ность обслуживания заявки и определяется момент его окончания. В примере это двенадцатый квант модельного времени:

$$\mathbf{t_3^M + T_2^2 = t_{12}^M}.$$

Второй канал СМО переходит в состояние «занято».

5. Модельное время устанавливается равным $\mathbf{t_4^M}$. В систему поступает третья заявка. Согласно типу события и состоянию системы заявка занимает первое место в очереди.

6. Модельное время устанавливается равным $\mathbf{t_5^M}$. Анализ показывает отсутствие событий в системе.

7. Модельное время устанавливается равным $\mathbf{t_6^M}$. В системе происходят два события – заканчивается обслуживание первой заявки и поступает четвертая заявка. Реакция системы:

- первая заявка отмечается в выходном потоке обслуженных заявок, и канал освобождается;
- третья заявка освобождает место в очереди и поступает на обслуживание освободившимся каналом;
- генерируется продолжительность обслуживания заявки и определяется момент его окончания. В примере это тринадцатый квант модельного времени;
- четвертая заявка занимает освободившееся первое место в очереди.

8. Модельное время устанавливается равным $\mathbf{t_7^M}$. В систему поступает пятая заявка. Согласно типу события и состоянию системы заявка занимает второе место в очереди.

9. Модельное время устанавливается равным $\mathbf{t_8^M}$. Анализ показывает отсутствие событий в системе.

10. Модельное время устанавливается равным $\mathbf{t_9^M}$. Анализ показывает отсутствие событий в системе.

11. Модельное время устанавливается равным $\mathbf{t_{10}^M}$. В систему поступает шестая заявка. Согласно типу события и состоянию системы заявка получает отказ и отмечается в выходном потоке потерянных заявок.

12. Модельное время устанавливается равным $\mathbf{t_{11}^M}$. В систему поступает седьмая заявка. Согласно типу события и состоянию системы заявка получает отказ и отмечается в выходном потоке потерянных заявок.

13. Модельное время устанавливается равным $\mathbf{t_{12}^M}$. В системе происходят два события – заканчивается обслуживание второй заявки и поступает восьмая заявка. Реакция системы:

- вторая заявка отмечается в выходном потоке обслуженных заявок, и канал освобождается;
- четвертая заявка освобождает первое место в очереди и поступает на обслуживание освободившимся вторым каналом;
- генерируется продолжительность обслуживания заявки и определяется момент его окончания. В примере это двадцатый квант модельного времени;
- пятая заявка занимает освободившееся первое место в очереди;

- поступившая восьмая заявка занимает второе место в очереди.

14. Модельное время устанавливается равным $\mathbf{t_{13}^{м}}$. В системе заканчивается обслуживание первым каналом третьей заявки. Реакция системы:
- третья заявка отмечается в выходном потоке обслуженных заявок, и канал освобождается;
- пятая заявка освобождает первое место в очереди и поступает на обслуживание освободившимся каналом;
- генерируется продолжительность обслуживания заявки и определяется момент его окончания. В примере это восемнадцатый квант модельного времени;
- восьмая заявка занимает освободившееся первое место в очереди;
- второе место в очереди освобождается.

Процесс имитации продолжается до достижения установленного конечного значения модельного времени. Диаграмма процесса представлена на рис. 4. 16.Нетрудно заметить, что имеется возможность построить циклический алгоритм моделирования.

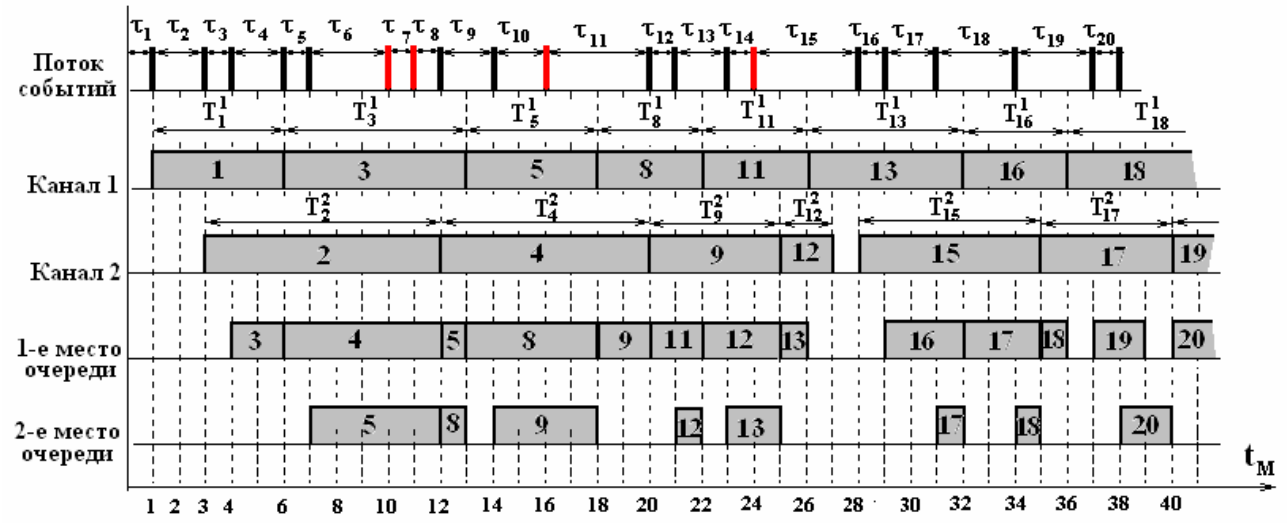


Рис. 4.16

Статистическая обработка результатов моделирования позволяет определить необходимые параметры эффективности СМО – вероятность и продолжительность обслуживания, вероятность отказа, время ожидания, среднее число занятых мест в очереди и пр. Диаграмма потоков заявок приведена на рис. 4.17.

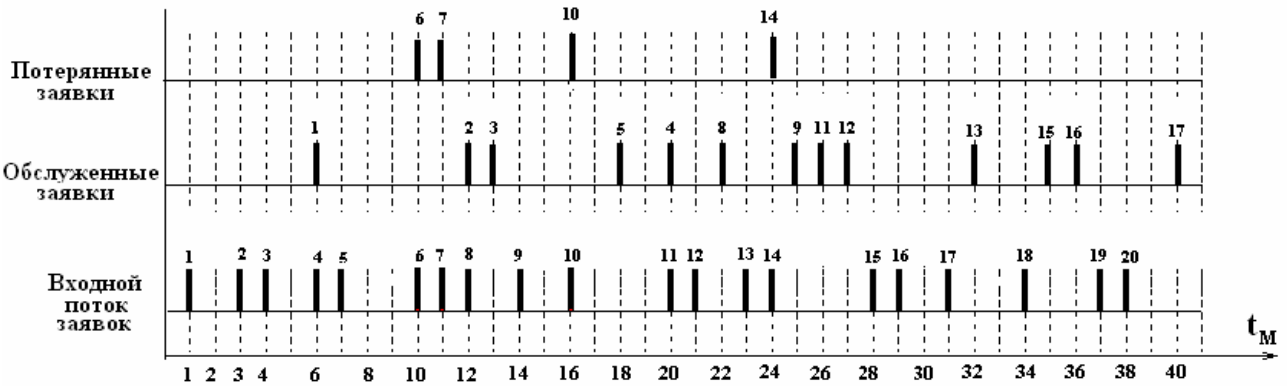


Рис. 4.17

Укрупнённая схема моделирующего алгоритма представлена на рис. 4.18.

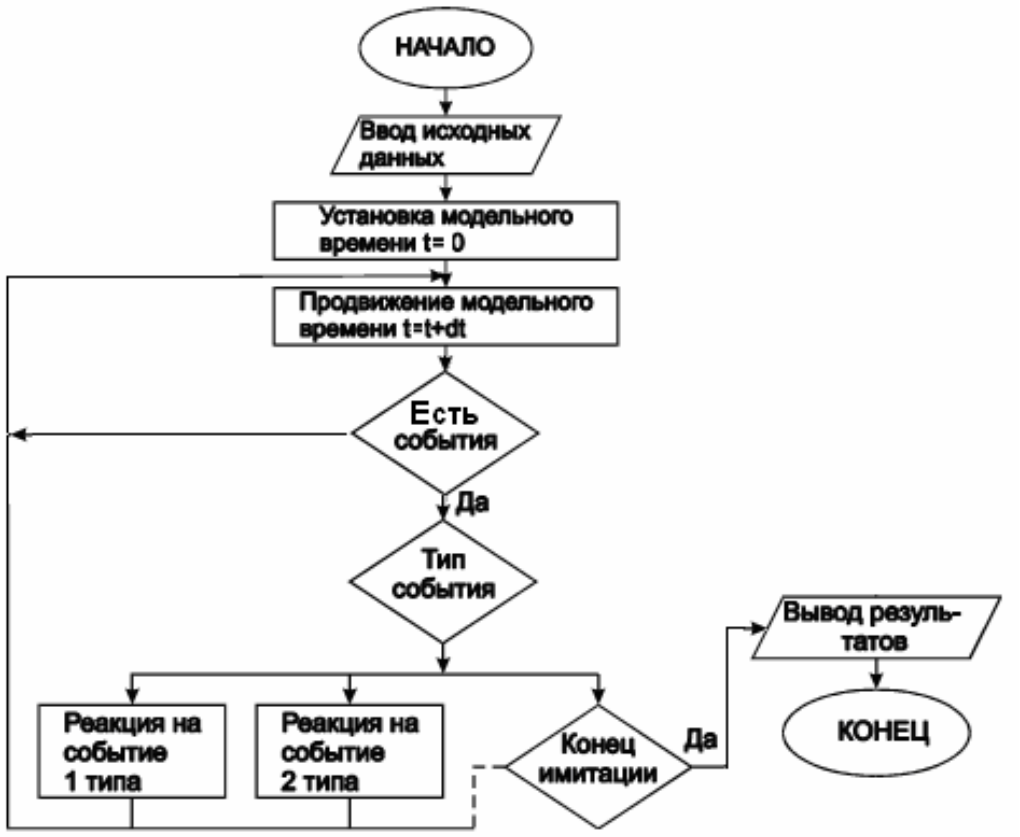


Рис. 4.18. Схема алгоритма моделирования по принципу Δt

Следует также отметить, что имитация даёт возможность учесть характеристики надежности системы. В частности, если известны времена наработки на отказ и восстановления всех входящих в систему устройств, то определяются моменты возникновения отказов устройств в период моделирования и моменты восстановления. Если устройство отказало, то возможны решения:

- снятие заявки без возврата;
- помещение заявки в очередь и дообслуживание после восстановления;
- поступление на повторное обслуживание из очереди.

4.6. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ФАКТОРОВ

Случайные факторы при имитационном моделировании могут иметь характер случайных величин, случайных процессов, случайных событий, потоков случайных событий. На практике обычно можно по результатам имитации случайных величин имитировать другие виды факторов.

На практике возникают задачи моделирования случайных величин с произвольными законами распределения. При этом данная задача решается следующим образом. Берется некоторое базовое распределение, которое подвергается функциональным преобразованиям, в результате чего обеспечивается заданное распределение случайной величины (рис. 4.19).

Рис. 4.19

4.6.1. Моделирование базовых случайных величин

В качестве базовых используются случайные величины с равномерным законом распределения.

Известно, что случайная величина называется равномерно распределенной в заданном интервале, если она с одинаковой вероятностью может принимать любые значения в данном интервале и не может принимать значений вне этого интервала.

Функция распределения вероятностей **F(z)** и плотность вероятности **f(z)** равномерно распределенной случайной величины в интервале [0 – 1] запишутся так

$$\mathbf{F(z)} = \begin{cases} 0, & \mathbf{z} < 0 \\ \mathbf{z}, & 0 \le \mathbf{z} \le 1 \\ 1, & \mathbf{z} > 1 \end{cases} \qquad \mathbf{f(z)} = \begin{cases} 1, & 0 \le \mathbf{z} \le 1 \\ 0, & \mathbf{z} < 0; \mathbf{z} > 1 \end{cases} \qquad (4.1)$$

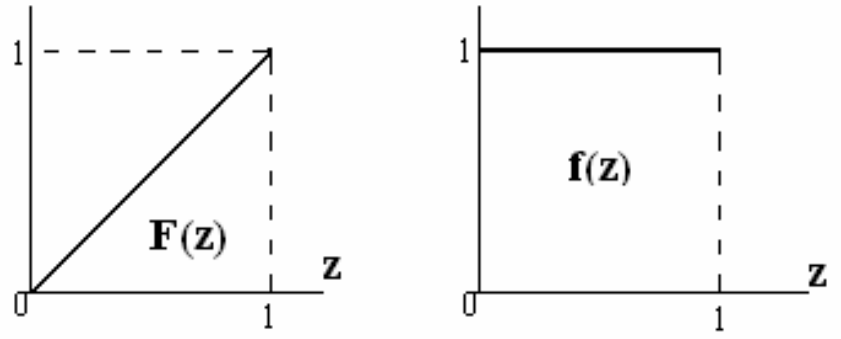


Рис. 4.20. Графики функций **F(z)** и **f(z)**
равномерно распределенной случайной величины

Базовый датчик (генератор случайных величин) выдает независимые равномерно распределенные в каком-либо диапазоне случайные величины.

Для генерации базовых случайных величин используются различные способы:

- аппаратный (приставки к ЭВМ);
- табличный;
- алгоритмический.

1. Аппаратный способ.

Датчик реализуется в виде некоторого аппаратного расширителя, использующего случайные свойства какого-либо из физических явлений.

Датчики часто используют свойства полупроводниковых устройств, издающих тепловые шумы. Эти шумы усиливаются и, далее, с помощью амплитудного ограничителя (дискриминатора) создается случайная последователь-

ность импульсов, которая преобразуется в двоичный код, соответствующий числу накопленных за интервал T импульсов из входной случайной последовательности.

Достоинства – абсолютно случайные свойства подобных датчиков;

Недостатки – нестабильность (температурная и временная) случайной последовательности, невозможность ее повторного воспроизведения.

2. Табличный способ.

Заключается в хранении в памяти некоторой реализации случайного процесса достаточной длины, полученной с помощью аппаратного датчика либо по таблице случайных чисел.

Достоинства – возможность повторного воспроизведения случайной последовательности;

Недостатки – значительные затраты памяти для хранения таблицы.

3. Алгоритмический способ.

Способ предполагает генерацию последовательности детерминированных чисел, обладающих при достаточно большой длине последовательности свойствами, приближающимися к случайным (псевдослучайные числа).

Достоинства – возможность получения повторяемых псевдослучайных последовательностей;

Недостатки – трудность моделирования систем случайных величин.

В алгоритмах генерации чаще всего используются рекуррентные соотношения:

$$z_i = f(z_{i-r}, \ldots, z_{i-1}).$$

К началу генерации должны быть заданы первые **r** значений. В большинстве случаев **r = 1:**

$$z_i = f(z_{i-1}).$$

Рассмотрим некоторые методы генерации псевдослучайных последовательностей.

Метод срединных квадратов

Метод заключается в возведении в квадрат предыдущего отсчета и в выделении в полученном произведении **n** промежуточных цифр:

$$z_i = f(z_{i-1}) = CUT_n[(z_{i-1})^2].$$

Пример 4.1

Десятичной частью псевдослучайного числа считается пять цифр, начиная с третьей в квадрате предыдущего числа. $z_1 = 0{,}15926$.

$0{,}15926^2 = 0{,}02\underline{53637}476 \rightarrow z_2 = 0{,}53637$

$0{,}53637^2 = 0{,}28\underline{76927}769 \rightarrow z_3 = 0{,}76927$

$0{,}76927^2 = 0{,}59\underline{17763}329 \rightarrow z_3 = 0,\ 17763$ и т.д.

Строим последовательность: 0,15926, 0,53637, 0,76927, 0, 17763, . . .

Мультипликативный метод

Наиболее популярный в настоящее время, он предполагает выделение остатка по $\mathbf{MOD}_g$ суммы $a+bz_{i-1}$, что принимается в качестве следующего значения.

Пример 4.2

Пусть $\alpha = 2$; $\beta = 3$; $\gamma = 10$; $z_1 = 1$.

$z_2 = MOD_{10}(2 + 3*1) = 5;$

$z_3 = MOD_{10}(2 + 3*5) = 7;$

$z_4 = MOD_{10}(2 + 3*7) = 3.$

Строим последовательность 1, 5, 7, 3, ...

При внимательном рассмотрении полученной последовательности псевдослучайных чисел можно заметить периодичность повторения чисел в этой последовательности. Данный недостаток присущ этому методу при любых a**;** b**;** g. Но в общем случае можно выделить период **T** и отрезок апериодичности в начале последовательности. Свойства генератора случайных чисел тем выше, чем больше длина периода **T**. При моделировании объем выборки не должен превышать длины периода, иначе будет наблюдаться статистическая зависимость результатов испытаний за счет повторения элементов псевдослучайной последовательности. Генераторы псевдослучайных чисел современных компьютеров позволяют получить $T \approx 2^{40}$ (например, при использовании чисел двойной длины и $g = 5^{17}$).

Полученная последовательность целых чисел может быть за счет деления на величину g преобразована в последовательность действительных чисел, изменяющихся в диапазоне [0, 1].

Для получения случайной величины, равномерно распределенной в интервале [a, b], можно использовать следующий прием:

$$\mathbf{z = z^*(b - a) + a,} \qquad (4.2)$$

где **z*** – случайная величина, равномерно распределенная в интервале [0, 1].

При этом **(b – a)** является коэффициентом масштабирования, **a** – величиной сдвига.

В качестве примера рассмотрим рис. 4.21, где на непрерывной числовой оси расположены случайные числа с равномерным распределением в диапазоне [0,1]. Для их получения использовался генератор случайных чисел табличного процессора Excel.

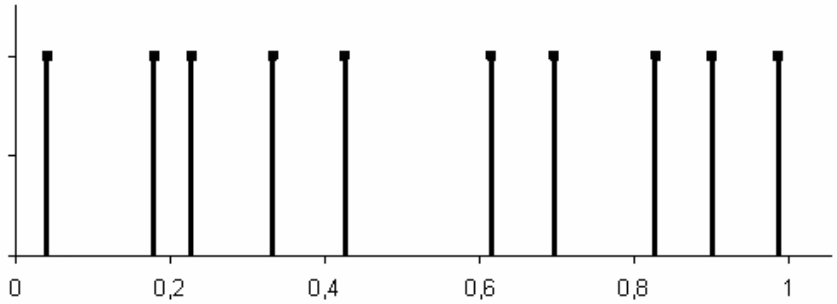


Рис. 4.21. Пример случайных чисел с равномерным распределением

4.6.2. Моделирование непрерывных случайных величин с произвольным распределением

Имитационное моделирование явлений и объектов, формальное описание которых возможно с помощью представления их в виде случайных величин с заданным законом распределения, основывается на использовании преобразований случайных величин с равномерным законом распределения. Такие преобразования могут быть осуществлены на основе: метода обратной функции; предельных теорем теории вероятности, приближенных методов и т.п.

Метод обратной функции основан на следующем.

Равномерно распределённая на интервале [0,1] величина x может быть преобразована в величину h с требуемой плотностью распределения $\mathbf{f_h(z)}$ путем решения уравнения $\mathbf{F_h(z)} = x$ относительно h. Отсюда следует, что искомое значение может быть определено из уравнения $h = \mathbf{F_h^{-1}}(x)$, т.е. через обратную функцию.

Экспоненциальное распределение

Функция распределения вероятностей **F(z)** и плотность вероятности **f(z)** случайной величины, имеющей экспоненциальное (показательное) распределение запишутся так:

$$\mathbf{F(z) = 1 - e^{-lz}},$$
$$\mathbf{f(z)} = l\ \mathbf{e^{-lz}}, \qquad (4.3)$$

где l - параметр распределения.

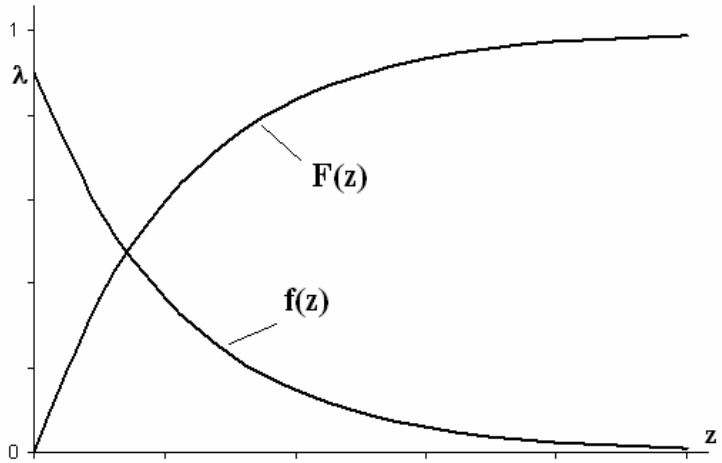


Рис. 4.22 Графики функций **F(z)** и **f(z)** экспоненциально распределенной случайной величины

Воспользуемся методом обратной функции.

$$\mathbf{F(z) = 1 - e^{-lz}} = x \qquad \text{или} \quad \mathbf{1} - x = \mathbf{e^{-lz}}.$$

Тогда

$$\mathbf{z} = -\frac{\mathbf{1}}{l}\mathbf{ln(1} - x\mathbf{)}. \qquad (4.4)$$

Таким образом, получая значение x с помощью датчика равномерно распределенных случайных чисел на интервале [0,1], можно получить значения **z,**

т.е. экспоненциально распределенной случайной величины в соответствии с выражением (4.4).

Пример ряда случайных чисел с экспоненциальным распределением приведен на рис. 4.23. Распределение получено путем преобразования равномерных случайных чисел с использованием выражения (4.4) при $\lambda = 1$.

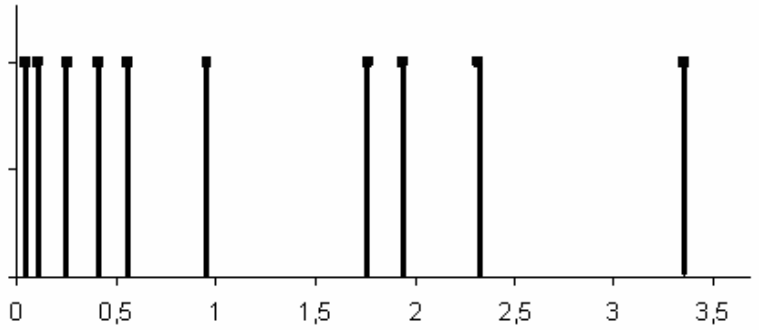


Рис. 4.23. Пример случайных чисел с экспоненциальным распределением

Напомним, что экспоненциальный закон распределения особенно часто используется для исследования систем массового обслуживания и определения показателей надежности систем.

Например. При рассмотрении СМО часто считается, что продолжительность обслуживания заявки каналом является экспоненциально распределенной случайной величиной. Тогда, допустим, что средняя продолжительность обслуживания $\mathbf{T_{ср}} = 5$ с. Отсюда интенсивность обслуживания $\mathbf{1/T_{ср}} = 0{,}2$ с$^{-1}$. Это означает, что вероятность окончания обслуживания очередного требования в интервале от **t** до **t**+D**t** равна **0,2**D**t** и не зависит от **t.** Вычисленные по формуле (4.4) конкретные продолжительности обслуживания при $\lambda = 0{,}2$ будут иметь, например, такие значения.

Случайные числа x	0,01962	0,33165	0,51227	0,67281	0,233	0,47109
Случайные T_i (с)	0,1	2,0	3,6	5,6	1,3	3,2

Нормальное распределение

Для случайной величины, имеющей нормальное распределение, функция распределения вероятностей **F(z)** и плотность вероятности **f(z)** имеют вид:

$$\mathbf{F(z)} = \frac{\mathbf{1}}{\sigma\sqrt{2\pi}} \int_{-\infty}^{z} \mathbf{exp}\left[-\frac{\mathbf{(z-m)^2}}{\mathbf{2\sigma^2}}\right],$$

$$\mathbf{f(z)} = \frac{\mathbf{1}}{\sigma\sqrt{2\pi}}\mathbf{exp}\left[-\frac{\mathbf{(z-m)^2}}{\mathbf{2\sigma^2}}\right]. \qquad (4.5)$$

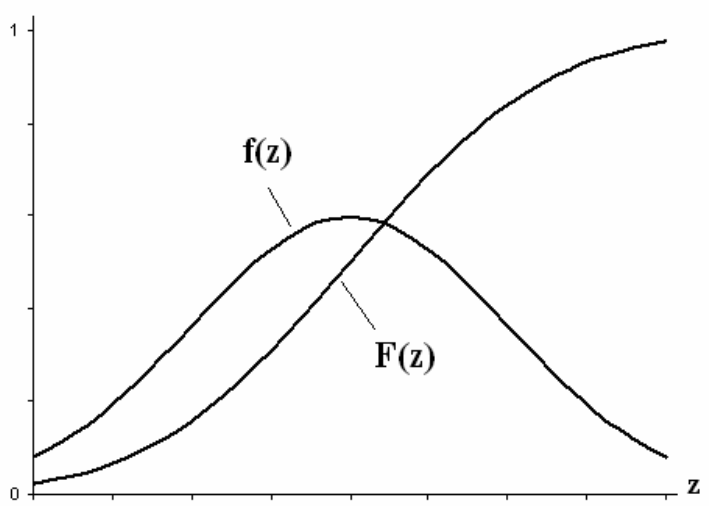


Рис. 4.24 Графики функций **F(z)** и **f(z)**
нормально распределенной случайной величины

Для нормального закона распределения не удается получить аналитические преобразования по методу обратной функции. Здесь используется центральная предельная теорема, согласно которой закон распределения суммы независимых случайных величин стремится к нормальному с увеличением числа слагаемых.

$$\mathbf{z} = \frac{\mathbf{1}}{\sqrt{\mathbf{k}}\mathbf{s}}\left(\sum_{\mathbf{i=1}}^{\mathbf{k}} x_i \; - \mathbf{km}\right), \qquad (4.6)$$

где x_i – отсчеты базовой случайной величины, **m** – математическое ожидание; s^2 - дисперсия.

Найдем необходимые параметры базовой случайной величины x**,** равномерно распределенной на интервале [0,1].

Плотность вероятности этой случайной величины: **f(**x**) = 1**.

Математическое ожидание

$$\mathbf{m} = \bar{x} = \int_{-\infty}^{+\infty} x\mathbf{f}(x)\mathbf{d}x = \int_{\mathbf{0}}^{\mathbf{1}} x\mathbf{d}x = \mathbf{0,5}.$$

Дисперсия

$$s^2 = \int_{-\infty}^{+\infty} (x - \bar{x})^2\mathbf{f}(x)\mathbf{d}x = \int_{\mathbf{0}}^{\mathbf{1}} (x - \mathbf{0,5})^2\mathbf{d}x = \frac{\mathbf{1}}{\mathbf{12}}.$$

Для суммы **k** базовых случайных величин:

$$\mathbf{m(сум) = 0.5k,}$$
$$s^2\mathbf{(сум) = k/12.}$$

Тогда (4.4) принимает вид:

$$\mathbf{z} = \sqrt{\frac{\mathbf{12}}{\mathbf{k}}}\left(\sum_{\mathbf{i=1}}^{\mathbf{k}} x_i \; - \frac{\mathbf{k}}{\mathbf{2}}\right). \qquad (4.7)$$

Наиболее удобной для расчетов формула (4.7) становится при **k** = 12.

Таким образом, для получения нормального распределения центрированной (**m=0**) и нормированной (σ**=1**) случайной надо выполнить преобразования:

$$\mathbf{z} = \sum_{i=1}^{12} x_i \; \mathbf{-\ 6}. \qquad (4.8)$$

Для получения нормально распределенной величины с произвольными **m** и σ пользуются дополнительным преобразованием:

$$\mathbf{z} = \sigma \mathbf{z^*} + \mathbf{m}, \qquad (4.9)$$

где **z*** – центрированная и нормированная величина.

Следует отметить, что согласно литературным данным зависимость (4.7) дает достаточно точные результаты уже для **k**=3, 4.

На рис. 4.25 приведен пример случайных чисел с нормальным законом распределения. Числа получены по зависимости (4.9) и (4.7), в которых **k** = 4; σ = 1; **m** = 1.

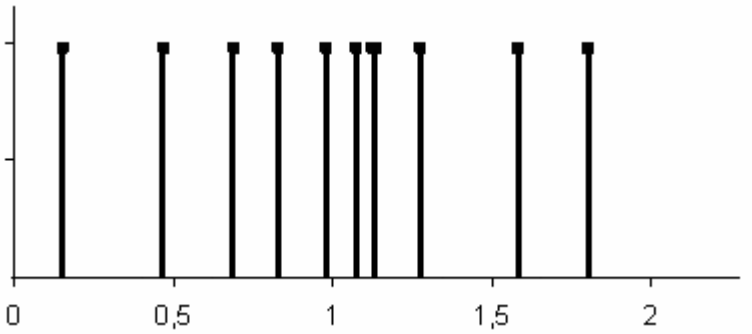


Рис. 4.25. Пример случайных чисел с нормальным распределением

4.6.3. Моделирование дискретных случайных величин

Равномерное распределение

Процедура получения целых дискретных чисел, распределенных равномерно в каком-либо диапазоне, является наиболее простой. Для этого базовую случайную величину x с равномерным законом распределения на интервале [0,1] преобразуют в целую в необходимом диапазоне, используя выражение (4.2), и далее берут целую часть полученного числа. В практике имитационного моделирования, когда требуется получить равномерно распределенные целые положительные числа в интервале [**0**,**n**], часто используют выражение:

$$\mathbf{Z = Int[}x\mathbf{(n+1)]}. \qquad (4.10)$$

Распределение Пуассона

Закон Пуассона описывает число событий, происходящих за одинаковые промежутки времени, при условии независимости этих событий. В основе алгоритма получения случайных чисел, распределенных по этому закону, лежит предельная теорема. В соответствии с ней, если **n** – количество событий велико, а **p** – вероятность успеха мала, то вероятность того, что при **n** испытаниях событие произойдет **k** раз, равна:

$$\mathbf{p_k = p\{z = k\}} = \frac{\mathbf{a^k}}{\mathbf{k!}} \mathbf{e^{-a}}, \qquad (4.11)$$

где **a = np** – параметр закона Пуассона.

Процедура получения чисел, распределенных по закону Пуассона, заключается в следующем. Проводится **n** испытаний. В процессе каждого испытания значение случайного числа с равномерным на интервале [0,1] законом распределения x сравнивается с **p**. Если выполняется условие $x \le p$, то к счетчику событий добавляется 1. После **n** испытаний, содержимое счетчика можно считать случайным числом, распределенным по Пуассону.

Следует помнить, что поскольку закон Пуассона справедлив для появления редких событий, необходимо выбирать величину **p** не слишком большой (рекомендуется не более 0,1), число испытаний **n**, наоборот, принимать значительной.

Блок-схема рассмотренной процедуры приведена на рис. 4.26.

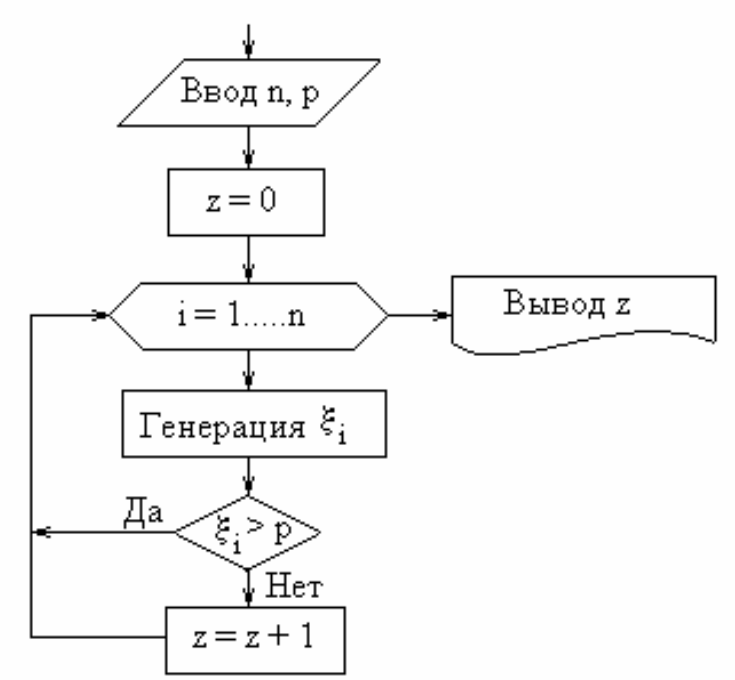


Рис. 4.26. Получение пуассоновского распределения

Ниже приведен ряд из десяти дискретных чисел, полученных с использованием рассмотренного алгоритма при **n** = 20 и **p** = 0,1**.**

2, 4, 1, 0, 0, 1, 0, 2, 6, 1

Геометрическое распределение

Рассматривается последовательность испытаний, в каждом из которых с вероятностью **p** может произойти некоторое событие. Геометрическое распределение описывает число испытаний до наступления этого события. Вероятность того, что событие произойдет после **k** испытаний, равна:

$$p\{z = k\} = p(1-p)^k . \qquad (4.12)$$

Процедура получения чисел, имеющих геометрическое распределение, такова. Проводится ряд испытаний. В процессе каждого испытания значение случайного числа с равномерным на интервале [0,1] законом распределения x сравнивается с **p**. Если выполняется условие $x < p$, то к счетчику испытаний

добавляется 1. После невыполнения условия содержимое счетчика можно считать случайным числом, имеющим геометрическое распределение.

Блок-схема процедуры приведена на рис. 4.27.

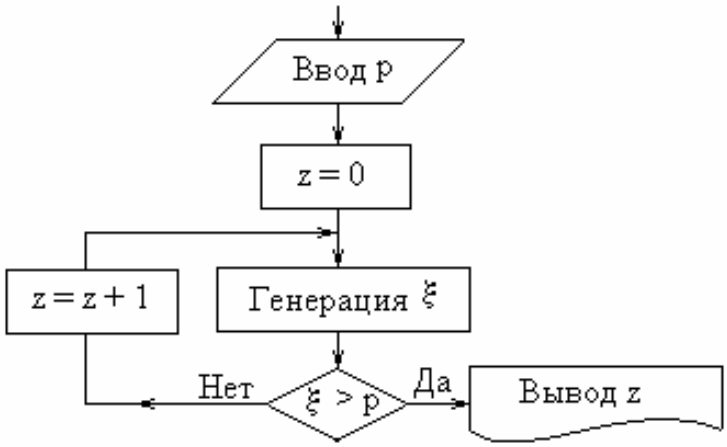


Рис. 4.27. Получение геометрического распределения

Получить последовательность случайных дискретных чисел с геометрическим распределением можно также по следующей зависимости:

$$\mathbf{z} = \mathbf{Int}\left[\frac{\mathbf{ln}\,x}{\mathbf{ln(1-p)}}\right]. \qquad (4.13)$$

Ниже приведен ряд из десяти дискретных чисел, полученных с использованием выражения (4.13) при $\mathbf{p} = 0{,}5$.

2, 0, 0, 1, 0, 1, 6, 0, 1, 1

4.6.4. Моделирование случайных событий и их потоков

Одиночное случайное событие

Пусть некоторое событие **A** происходит с вероятностью $\mathbf{p_A}$. Можно показать, что ему равновероятно другое событие **B**, состоящее в том, что случайное число xc равномерным на интервале [0,1] законом распределения меньше либо равно $\mathbf{p_A}$. Отсюда следует алгоритм имитации факта появления события **A**.

1) с помощью датчика случайных чисел получить x;
2) проверить выполнение неравенства $x \le \mathbf{p_A}$;
3) если оно выполняется, то событие **A** произошло.

Сложные события

Пусть сложное событие состоит, например, из двух независимых элементарных событий **A** и **B** с вероятностями $\mathbf{p_A}$ и $\mathbf{p_B}$ соответственно. Имитация в этом случае заключается в проверке неравенств:

$$x_1 \le \mathbf{p_A} \quad \text{и} \quad x_2 \le \mathbf{p_B},$$

где x_1 и x_2 – случайные числа с равномерным законом распределения на интервале [0,1].

В зависимости от исхода проверки неравенств (аналогично рассмотренному выше алгоритму) делается вывод о том, какой из вариантов имеет место:

1) произошли оба события;
2) произошло только событие **A**;
3) произошло только событие **B**;
4) оба события не произошли.

Пусть сложное событие состоит из двух элементарных зависимых событий, причем появление события **B** зависит от появления события **A**.

Здесь в качестве исходных данных задаются $\mathbf{p_A}$, $\mathbf{p_B}$ и условная вероятность $\mathbf{p_{B/A}}$. По формуле полной вероятности:

$$\mathbf{p_B} = \mathbf{p_A} \times \mathbf{p_{B/A}} + \mathbf{p_{\bar{A}}} \times \mathbf{p_{B/\bar{A}}}.$$

Отсюда легко выразить $\mathbf{p_{B/\bar{A}}}$

Алгоритм имитации такого события использует проверку следующих неравенств:

$$\begin{cases} x_1 \le \mathbf{p_A} \\ x_2 \le \mathbf{p_{B/A}} \end{cases} \qquad \begin{cases} x_1 \le \mathbf{p_A} \\ x_2 > \mathbf{p_{B/A}} \end{cases} \qquad \begin{cases} x_1 > \mathbf{p_A} \\ x_2 \le \mathbf{p_{B/\bar{A}}} \end{cases} \qquad \begin{cases} x_1 > \mathbf{p_A} \\ x_2 > \mathbf{p_{B/\bar{A}}} \end{cases}$$

В зависимости от того, какая из этих четырех систем неравенств выполняется, делается вывод о том, какой из этих четырех возможных исходов имеет место:

1) произошли оба события;
2) произошло только событие **A**;
3) произошло только событие **B**;
4) оба события не произошли.

События, составляющие полную группу

Пусть множество событий $\mathbf{A_i}$ ($\mathbf{i = 1,\ldots n}$) составляют полную группу. Тогда их вероятности $\mathbf{p(A_i)}$ таковы, что

$$\sum_{i=1}^{n} \mathbf{p(A_i)} = 1.$$

Имитация факта появления одного из событий $\mathbf{A_i}$ сводится к проверке следующих неравенств:

$$\sum_{i=0}^{k-1} \mathbf{p(A_i)} \le x < \sum_{i=0}^{k} \mathbf{p(A_i)}, \text{ где } (\mathbf{k = 1,\ldots n}),\ \mathbf{p(A_0) = 0.}$$

Выполнение **j**-го неравенства эквивалентно появлению события $\mathbf{A_j}$.

Описанный алгоритм называют иногда алгоритмом «розыгрыша по жребию». Его можно интерпретировать как установление номера **j**-го отрезка длинной $\mathbf{p(A_j)}$, на который пало случайное число x, при условии разбиения отрезка единичной длины на интервалы с длинами $\mathbf{p(A_1), p(A_2),...p(A_n)}$ (рис 4.28)

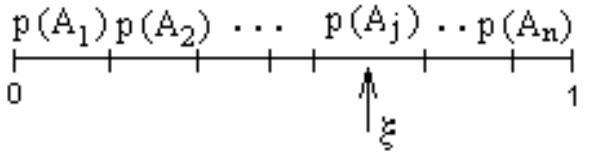


Рис. 4.28. Моделирование событий, составляющих полную группу

Последовательность шагов алгоритма имитации:

1) с помощью датчика случайных чисел получить x;

2) определить интервал, содержащий значение x**,**

3) номер интервала принять в качестве номера наступающего в опыте события.

Простейший поток событий

Моделирование потока событий сводится к моделированию моментов времени, в которые они происходят. Напомним, что простейшим потоком событий называется стационарный пуассоновский поток. Он обладает свойствами ординарности, стационарности и отсутствия последействия.

1) Ординарность означает, что в один момент времени может произойти не более одного события.

2) Стационарность устанавливает независимость характеристик потока от времени, т.е. вероятность наступления определенного количества событий на некотором интервале времени зависит только от его длины и не зависит от его положения на временной оси:

3) Отсутствие последействия предполагает независимость событий друг от друга – каждое событие наступает независимо от того, наступали ли другие.

Согласно закону Пуассона (4.11) можно определить вероятность возникновения **k** событий потока на интервале времени t:

$$\mathbf{p_k} = \frac{\mathbf{a^k}}{\mathbf{k!}}\mathbf{e^{-a}}.$$

Для стационарного потока **a** = λt, где λ - интенсивность потока, характеризующая среднее число событий в единицу времени.

Рассмотрим интервал времени от начала отсчета до наступления первого события t_1 (т.е. интервал на котором событий нет). Из (4.11) для **k** = 0 имеем:

$$\mathbf{p_0 = \exp(-\lambda t_1).}$$

То есть, рассматриваемый интервал времени представляет собой непрерывную случайную величину, распределенную по экспоненциальному закону.

Рассуждая аналогично, можно показать, что такими же непрерывными случайными величинами являются и другие интервалы между событиями.

Следовательно, моменты наступления событий в простейшем потоке могут моделироваться с использованием выражения:

$$\mathbf{t_{i+1}} = \mathbf{t_i} + \left(-\frac{\ln x}{\lambda}\right). \qquad (4.14)$$

Потоки Эрланга

Распространенной моделью потока событий, позволяющей менять его свойства в широком диапазоне, служат потоки Эрланга. Они являются примерами потоков с ограниченным последействием. Данные потоки образуются пу-

тем закономерного просеивания простейшего потока. Например, при получении потока Эрланга **k**-го порядка, просеивание сводится к выбору из исходного простейшего (базового) потока каждого **k**-го события. Это эквивалентно образованию длины интервала потока Эрланга в виде суммы **k** смежных интервалов (рис.4.29).

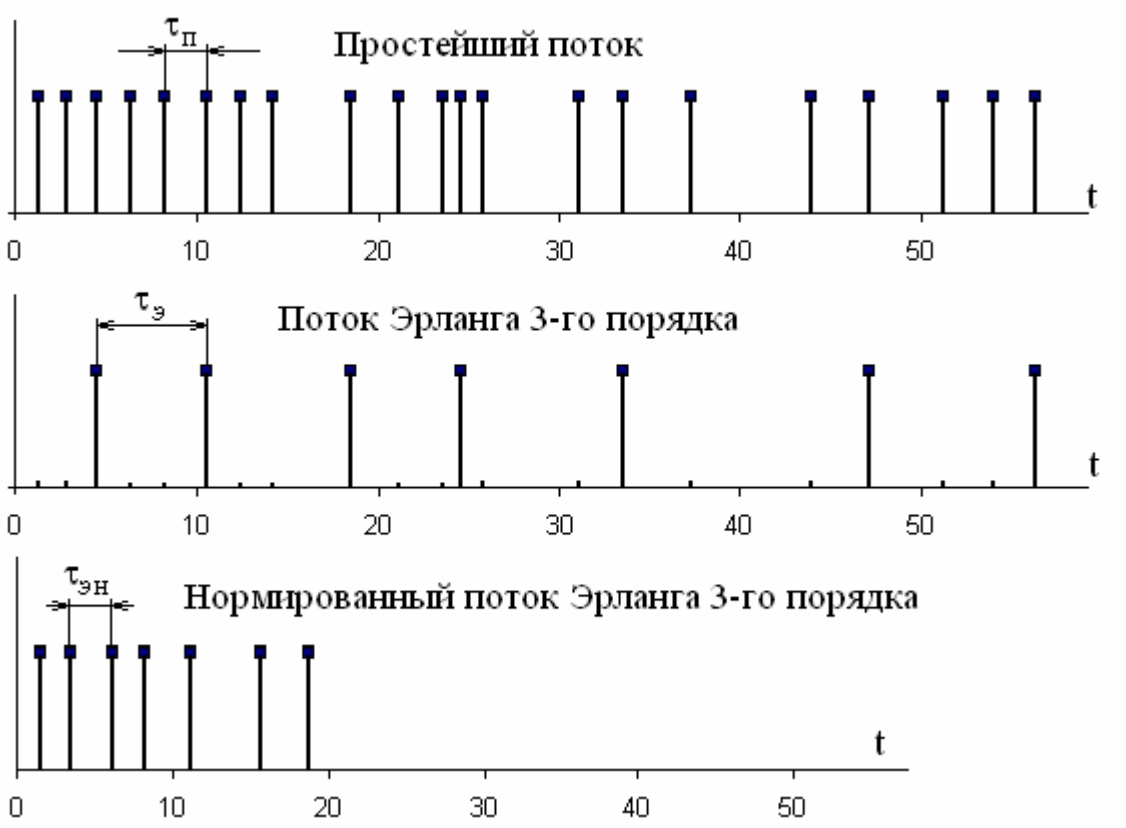


Рис. 4.29. Получение потока Эрланга просеиванием базового потока

При моделировании произвольных потоков событий интервалы между событиями в потоке Эрланга обычно нормируют коэффициентом **k** в целях коррекции масштаба времени:

$$t_{ЭН} = t_Э/k.$$

Для нормированного потока Эрланга **k**-го порядка математическое ожидание и дисперсия рассчитываются так:

$$M(t_{ЭН}) = 1/\lambda, \qquad D(t_{ЭН}) = 1/(\lambda k)^2. \qquad (4.15)$$

Нормированный поток Эрланга в зависимости от порядка **k** позволяет получить любую степень последействия: от полного отсутствия (**k = 1**) до регулярного потока с постоянными интервалами, равными $1/\lambda$ (**k = ¥**). Благодаря этому реальный поток событий с последействием можно в некоторых случаях моделировать нормированным потоком Эрланга соответствующего порядка, имеющим примерно те же математическое ожидание и дисперсию.

4.7. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПРОЦЕССОВ

4.7.1. Дискретные цепи Маркова

Имеется система, которая в каждый момент времени может находиться в одном из нескольких состояний $\mathbf{A_1, A_2, \ldots A_n}$. В некоторые дискретные моменты времени $\mathbf{t_1, t_2}$ эта система может менять свое состояние. Основное свойство цепи Маркова состоит в том, что состояние, в котором система окажется в следующий момент времени зависит только от ее текущего состояния и не зависит от всех предыдущих.

Переход из состояния в состояние определяется матрицей вероятностей переходов $|\mathbf{P_{ij}}|$, которая считается заданной:

$$\left|\mathbf{P_{ij}}\right| = \begin{vmatrix} p_{11} & p_{12} & \cdots & p_{1j} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2j} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{i1} & p_{i2} & \cdots & p_{1ij} & \cdots & p_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{n1} & p_{1n2} & \cdots & p_{nj} & \cdots & p_{nn} \end{vmatrix} \qquad \sum_{j=1}^{n} p_{ij} = 1 \qquad (i = \overline{1,n}),$$

где $\mathbf{p_{ij}}$ – вероятность перехода из состояния **i** в состояние **j**.

Каждому моменту времени соответствует вектор состояний $\mathbf{[S(t_k)]}$, элементы которого есть вероятности нахождения системы в каждом из возможных состояний:

$$\mathbf{[S(t_k)] = [S_1(t_k), S_2(t_k), \ S_n(t_k)]}, \qquad \sum_{i=1}^{n} S_i(t_k) = 1 \qquad (i = \overline{1,n}).$$

Система дифференциальных уравнений Колмогорова позволяет определить искомые элементы вектора состояний как функции времени. Если для системы существует стационарный режим, можно определить и предельное распределение вероятностей состояний.

Однако в ряде случаев знать общие, вероятностные характеристики поведения системы недостаточно. Для моделирования поведения системы можно использовать имитационные алгоритмы.

Распространен следующий прием имитации поведения дискретной цепи Маркова.

Предположим, что начальное состояние системы задано. Строка матрицы вероятностей переходов для этого состояния устанавливает вероятности переходов из текущего состояния в любое другое. Можно заметить, что эти переходы представляют собой события, составляющие полную группу. То есть можно воспользоваться алгоритмом «розыгрыша по жребию» - установить номер события, на который пало случайное число x, равномерно распределенное в диапазоне [0,1]. Таким образом, новое состояние мы получаем, как дискретную случайную величину. Далее процесс повторяется.

Если начальное состояние системы не задано, то часто поступают следующим образом. Рассчитывают предельные распределения вероятностей для стационарного режима. Эти компоненты вектора также рассматривают, как полную группу событий и исходное состояние тоже разыгрывают выпадением случайного числа x.

Пример 4.3

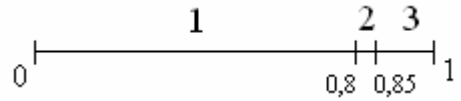
Ранее решалась задача (пример 3.2), в которой центральный процессор мультипрограммной компьютерной системы в любой момент времени выполняет либо приоритетную программу, либо фоновую программу, либо находится в состоянии ожидания. Продолжительность нахождения системы в каждом состоянии кратна длительности шага - D**t**. Исходное состояние – простой процессора. Матрица вероятностей переходов имела вид:

$$|\mathbf{p}| = \begin{vmatrix} \mathbf{0,7} & \mathbf{0,2} & \mathbf{0,1} \\ \mathbf{0,8} & \mathbf{0,1} & \mathbf{0,1} \\ \mathbf{0,8} & \mathbf{0,05} & \mathbf{0,15} \end{vmatrix}$$

Произведем процесс имитации функционирования этой системы.

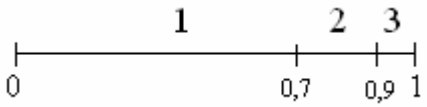
Пусть **1** - состояние обслуживания основной программы; **2** - состояние обслуживания фоновой программы; **3** - состояние простоя.

Рассмотрим третью строку матрицы переходов, соответствующую исходному состоянию. Ряд вероятностей имеет вид:



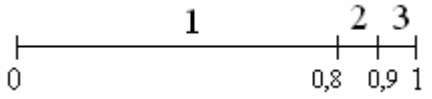
Генерируем равномерное случайное число x. Допустим оно равно 0,54870. Это означает, что система перешла в состояние **1**.

Ряд вероятностей для этого состояния (первая строка матрицы):



Сгенерированное равномерное случайное число x. равно 0,79154. То есть система перешла в состояние **2**.

Ряд вероятностей для этого состояния (вторая строка матрицы):



Новое случайное число равно 0,86392, система остается в состоянии **2**.

Следующее случайное число равно 0,07876 – система переходит в состояние **1**. И так далее.

Результат имитации функционирования системы приведен на рис. 4.30.

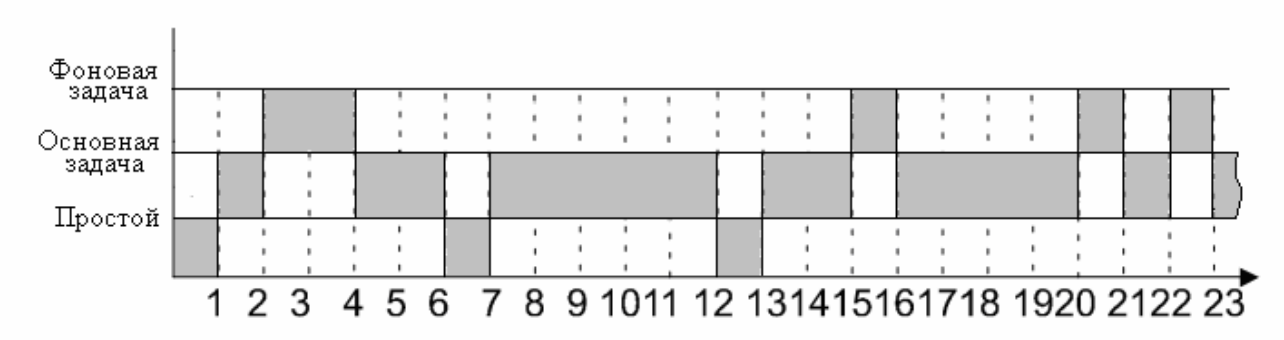


Рис. 4.30 Имитация работы компьютерной системы

4.7.2. Непрерывные цепи Маркова

Непрерывными цепями Маркова моделируются системы, которые имеют дискретное пространство состояний $\mathbf{A_1, A_2, \ldots A_n}$., но переход между состояниями осуществляется в произвольные моменты непрерывного времени. В этом состоит главное отличие непрерывных цепей от ранее рассмотренных дискретных цепей, которые могут менять состояние в определенные (заданные) моменты времени. Но основное свойство цепей Маркова – зависимость текущего состояния только от одного предыдущего, безусловно, сохраняется.

Основными параметрами непрерывных цепей Маркова являются интенсивности переходов между состояниями $\mathbf{q_{ij}}$, которые связаны с вероятностями переходов зависимостью:

$$\mathbf{p_{ij} = q_{ij}(t)\Delta t.}$$

Приведенная зависимость справедлива с точностью до бесконечно малых высших порядков при малом $\Delta\mathbf{t}$.

Аналогично дискретным цепям, здесь также каждому моменту времени соответствует вектор состояний $\mathbf{[S(t_k)]}$, элементы которого есть вероятности нахождения системы в каждом из возможных состояний:

$$\mathbf{[S(t_k)] = [S_1(t_k), S_2(t_k), \ S_n(t_k)]}, \qquad \sum_{i=1}^{n} \mathbf{S_i(t_k) = 1} \qquad \mathbf{(i = \overline{1,n})}.$$

Анализ поведения непрерывных цепей возможен с использованием дифференциальных уравнений Колмогорова, в которых вероятности переходов $\mathbf{p_{ij}}$ могут быть вычислены через значения интенсивностей переходов $\mathbf{q_{ij}}$.

Для имитационного моделирования непрерывной цепи Маркова применим подход, использующий следующие рассуждения.

Пусть в момент времени **0** система находится в состоянии $\mathbf{S_i}$. Необходимо выяснить, в какое состояние она перейдет и когда осуществится этот переход.

Известно, что для непрерывной марковской цепи плотность вероятности перехода совпадает с интенсивностью потока переходов $\mathbf{q_{ij}}$. Кроме того, интервал времени от начала отсчета до наступления первого события представляет собой непрерывную случайную величину, распределенную по экспоненциальному закону. Поэтому для вероятности наступления события (перехода из состояния $\mathbf{S_i}$ в другое состояние) можно записать:

$$p_i(t) = \exp\left(-\sum_{j=1}^{n} q_{ij} \times t\right) \quad i \neq j,$$

где t - интервал времени до наступления события.

Отсюда можно найти t:

$$t = -\frac{\ln p_i(t)}{\sum_{j=1}^{n} q_{ij}} \qquad i \neq j. \qquad (4.16)$$

Вероятности перехода в каждое из **n** возможных состояний можно оценить так:

$$p_{ij} = \frac{q_{ij}}{\sum_{j=1}^{n} q_{ij}} \qquad i \neq j. \qquad (4.17)$$

Таким образом, процесс имитационного моделирования может состоять из следующих действий.

Устанавливается начальный момент времени t_0. Определяется интервал времени до перехода системы в новое состояние. Для этого генерируется случайное число x, равномерно распределенное в диапазоне [0,1] и подставляется в зависимость (4.16):

$$t = -\frac{\ln x}{\sum_{j=1}^{n} q_{ij}}.$$

С использованием выражения (4.17) вычисляются вероятности переходов из текущего состояния в другое. Поскольку эти переходы представляют собой события, составляющие полную группу, можно воспользоваться алгоритмом «розыгрыша по жребию». Теперь мы получили новое состояние системы, в которое она перешла в момент времени t_0+t. Далее процесс повторяется. Результатом моделирования будет массив $[t_0,S(t_0); t_1,S(t_1);\ldots t_k,S(t_k)]$.

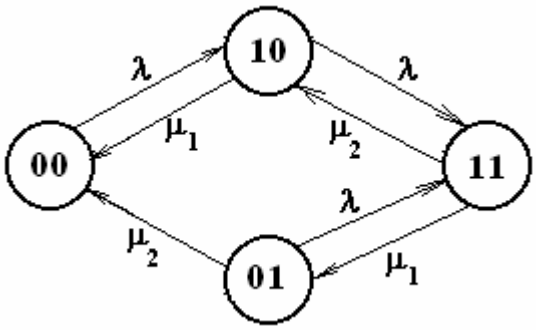
Пример 4.4

Рассмотрим двухпроцессорную вычислительную систему из примера 3.3. В ней обрабатывался поток задач, поступающих с интенсивностью l. Производительности процессоров заданы и, соответственно, равны m_1 и m_2, причем $m_1 > m_2$. Задача в первую очередь принималась на обслуживание первым процессором, имеющим большую производительность.

Проведем процесс имитации функционирования этой системы, если $l = 2$; $m_1 = 5$; $m_2 = 3$.

Возможные состояния системы обозначим следующим образом: **00** - оба процессора простаивают; **10** - первый процессор занят, второй простаивает; **01** - второй процессор занят, первый простаивает; **11** - оба процессора заняты.

Граф функционирования системы имеет вид:



1. Допустим, в исходном состоянии система свободна.

t = 0 Состояние $\mathbf{S_{00}}$.

2. Определим время t, за которое система перешла из состояния **«00»** в состояние «**10**». Генерируем случайное число. В примере x = 0,5744.

$$t = -\ln(x)/l = 0{,}27719,$$

t = 0 + 0,27719 = **0,27719** Состояние $\mathbf{S_{10}}$.

3. Определим время, за которое система перешла из состояния **«10»** в другое состояние. Генерируем случайное число. В примере x = 0,08203.

$$t = -\ln(x)/(l+m_1) = 0{,}35724.$$

Определим, в какое состояние перешла система.

Вероятность перехода в состояние **«11»** $\mathbf{p_{10\text{-}11}} = l/(l+m_1) = \mathbf{0{,}268.}$

Вероятность перехода в состояние **«00»** $\mathbf{p_{10\text{-}00}} = m_1/(l+m_1) = \mathbf{0{,}732.}$

Генерируем случайное число. В примере x = 0,25918. Система перешла в состояние **«11».**

t = 0,27719 + 0,35724 = **0,63443** Состояние $\mathbf{S_{11}}$.

4. Определим время, за которое система перешла из состояния **«11»** в другое состояние. Генерируем случайное число. В примере x = 0,60888.

$$t = -\ln(x)/(m_1+m_2) = 0{,}06243.$$

Определим, в какое состояние перешла система.

Вероятность перехода в состояние **«10»** $\mathbf{p_{11\text{-}10}} = m_1/(m_1+m_2) = \mathbf{0{,}375.}$

Вероятность перехода в состояние **«01»** $\mathbf{p_{11\text{-}01}} = m_2/(m_1+m_2) = \mathbf{0{,}625.}$

Генерируем случайное число. В примере x = 0,27879. Система перешла в состояние **«10».**

t = 0,63443 + 0,06243 = **0,69686** Состояние $\mathbf{S_{10}}$.

5. Продолжаем аналогично.

Результат имитации функционирования системы приведен на рис. 4.31.

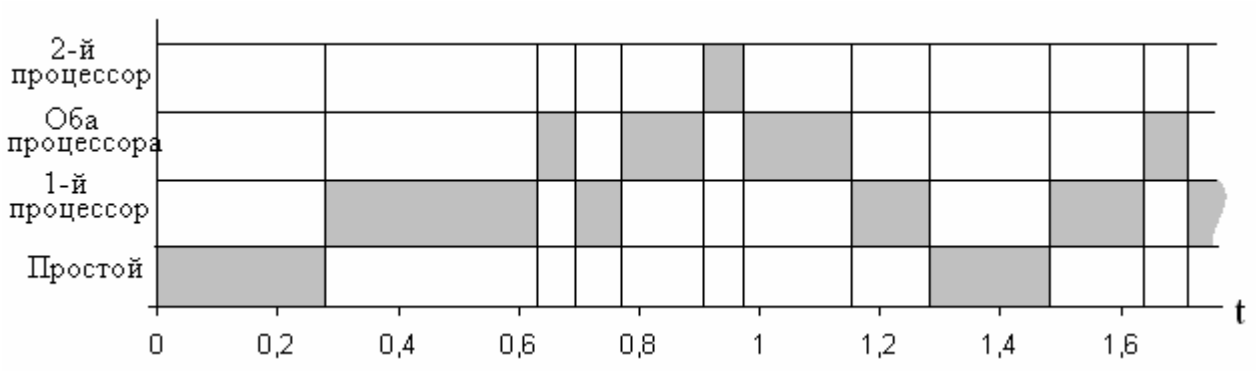


Рис. 4.31 Имитация работы двухпроцессорной компьютерной системы

4.8. ОБРАБОТКА И АНАЛИЗ РЕЗУЛЬТАТОВ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

При имитационном моделировании можно измерять значения любых характеристик, интересующих исследователя. Полученная информация является исходным статистическим материалом для нахождения приближенных значений показателей эффективности имитируемой системы. Обычно по результатам вычислений определяются характеристики всей системы, каждого потока и устройства.

Для систем массового обслуживания производится подсчёт поступивших заявок, полностью обслуженных и покинувших систему заявок без обслуживания по тем или иным причинам. Соотношения этих величин характеризуют производительность имитируемой системы при определённой рабочей нагрузке. По каждому потоку заявок могут вычисляться времена реакций и ожидания, количества обслуженных и потерянных заявок. По каждому устройству определяется время загрузки при обслуживании одной заявки и число обслуженным устройством заявок, время простоя устройства в результате отказов и количество отказов, возникших в процессе моделирования, длины очередей и занимаемые ёмкости памяти.

При статистическом моделировании большая часть характеристик – это случайные величины. По каждой такой характеристике **y** определяется **N** значений, по которым строится гистограмма относительных частот, вычисляется математическое ожидание, дисперсия и моменты более высокого порядка, определяются средние по времени и максимальные значения, строится гистограмма относительных частот.

Определение условий удовлетворения стохастических ограничений при имитационном моделировании производится путём простого подсчёта количества измерений, вышедших и не вышедших за допустимые пределы.

В условиях большого количества информации обработка результатов моделирования может решаться только с применением методов, оптимальных по времени и обеспечивающих экономию памяти ЭВМ. Перечислим ряд таких приемов.

4.8.1. Оценка вероятностных параметров

Оценкой вероятности какого-либо события или состояния является частота появления.

$$\hat{p}(A) = \frac{m}{N}. \qquad (4.18)$$

Для ее получения обычно на программном уровне организуют 2 счетчика: один для подсчета общего количества экспериментов **N**, второй - для подсчета общего количества положительных исходов **m**.

Оценку математического ожидания получают по известным формулам, как среднее арифметическое значение случайной величины:

$$\hat{m} = \frac{1}{N}\sum_{i=1}^{N} z_i . \qquad (4.19)$$

Сумму лучше всего вычислять путем постепенного накапливания во избежание непроизводительных затрат памяти.

Оценку дисперсии тоже можно вычислять по известной формуле:

$$S^2 = \frac{1}{N-1}\sum_{i=1}^{N}(z_i - \hat{m})^2 .$$

Однако это связано с непроизводительным использованием памяти ЭВМ. Поэтому лучше воспользоваться формулой, позволяющей использовать постепенное накапливание:

$$\mathbf{S^2 = \frac{1}{N-1}\left[\sum_{i=1}^{N} z_i^2 - \frac{1}{N}\left(\sum_{i=1}^{N} z_i\right)^2\right]}. \qquad (4.20)$$

Иногда в качестве характеристик исследуемой системы выступает закон плотности распределения. Его приближенно можно охарактеризовать ***гистограммой***. Для ее построения интервал изменения случайной величины разбивают на отрезки $\mathbf{t_i}$, каждому из них сопоставляют счетчик, где накапливают $\mathbf{m_i}$ - количество попаданий значений величины в отрезок $\mathbf{t_i}$. На каждом отрезке строится прямоугольник с высотой $\mathbf{m_i/(Nt_i)}$.

Полученную гистограмму можно сгладить.

4.8.2. Оценка корреляционных параметров

Для оценки корреляционного момента двух случайных величин, из соображений эффективного использования памяти компьютера, рекомендуется использовать формулу:

$$\hat{k} = \frac{1}{N}\left[\sum_{i=1}^{N} x_i y_i - \frac{1}{N}\left(\sum_{i=1}^{N} x_i\right)\left(\sum_{i=1}^{N} y_i\right)\right]. \qquad (4.21)$$

Для вычисления оценки характеристик случайных процессов производят статистическую обработку по **N** реализациям. Для этого интервал задания случайных процессов разбивают на части с шагом $\Delta\mathbf{t} = \mathbf{const}$. Математические

ожидания и дисперсии для каждого $t_k=kDt$ можно вычислить по формулам (4.19), (4.20) приведенным выше. Оценку корреляционной функции - по формуле:

$$k(t_k,t_j)=\frac{1}{N-1}\left[\sum_{i=1}^{N}F_i(t_k)F_i(t_j)-\frac{1}{N}\left(\sum_{i=1}^{N}F_i(t_k)\right)\left(\sum_{i=1}^{N}F_i(t_j)\right)\right]. \quad (4.22)$$

Здесь $t_k=kDt$, $t_j=jDt$.

Важной задачей обработки статистической информации, полученной в результате имитационного моделирования, является задача определения необходимого количества реализаций **N**, обеспечивающих заданную точность получения оценок. Для определения **N** при оценке вероятности пользуются формулой:

$$N=\frac{U_a^2 p^*(1-p^*)}{\varepsilon^2}. \quad (4.23)$$

При оценке математического ожидания

$$N=\frac{\sigma^2 U_a^2}{\varepsilon^2}. \quad (4.24)$$

В формулах (4.23) и (4.24) U_a - квантиль для нормального, центрированного закона распределения, соответствующий значению **a = 1-p**, где **p** заданная достоверность; **p***- оцениваемая вероятность; s^2 - дисперсия; e - допустимая погрешность.

В (4.23) и (4.24) значение **p*** неизвестно, а s^2 - может быть неизвестным. Поэтому производят предварительно 50-100 реализаций, получают по ним оценки **p*** и s^2, а далее подставляют их в (4.23) и (4.24) для вычисления уточненного значения **N**.

4.8.3. Расчет средних по времени параметров СМО

Показатели эффективности СМО рассчитываются по простым зависимостям с усреднением по времени.

Коэффициент загрузки **k**-го устройства:

$$r_k=V_k N_{ok}/T_m. \quad (4.25)$$

где V_k- среднее время обслуживания одной заявки **k**-м устройством; N_{ok} - количество обслуженных им заявок за время моделирования T_m.

Средняя длина очереди к каждому устройству:

$$L_{CP}=\frac{\sum_{i=1}^{N}L_i t_i}{T_m}, \quad (4.26)$$

где **i** - номер очередного изменения состояния очереди (занесение заявки в очередь или исключение из очереди); **N** - количество изменений состояния очереди; t - интервал времени между двумя последними изменениями очереди.

Средняя занятая емкость устройства-накопителя:

$$Q_{CP} = \frac{\sum_{i=1}^{N} Q_i t_i}{T_m}, \qquad (4.27)$$

где Q_i - ёмкость накопителя, занятая в интервале между двумя последними обращениями к нему для ввода-вывода заявки.

Аналогично могут быть рассчитаны и другие необходимые средние параметры эффективности.

4.9. ПЛАНИРОВАНИЕ ЭКСПЕРИМЕНТОВ С ИМИТАЦИОННЫМИ МОДЕЛЯМИ

Машинный эксперимент предполагает наблюдение за поведением имитационной модели. При этом каждому эксперименту (прогону модели) соответствует определенная комбинация значений параметров модели и длительность интервала времени, обеспечивающая набор необходимой статистики, при условии обеспечения заданной точности модели.

Основная цель планирования машинных экспериментов заключается в получении необходимой информации об исследуемой системе при ограниченных ресурсах (затраты машинного времени, памяти и т.п.). К числу частных задач, решаемых при планировании машинных экспериментов, относятся задачи уменьшения погрешности результатов моделирования, проверки адекватности модели и т.п.

Эффективность машинных экспериментов существенно зависит от выбора плана эксперимента, т.к. именно план определяет объём и порядок проведения вычислений на ЭВМ, приёмы накопления и статистической обработки результатов моделирования системы. Таким образом, при машинном моделировании необходимо не только рационально планировать и проектировать саму модель системы, но и процесс её использования, т.е. проведения с ней эксперимента.

При планировании машинных экспериментов возникает целый ряд проблем, взаимно связанных как с особенностью функционирования моделируемого объекта, так и с особенностью машинной реализации модели и обработки результатов эксперимента. В первую очередь к таким относятся проблемы построения плана машинного эксперимента, стохастической сходимости результатов, ограниченности машинных ресурсов, уменьшения дисперсии оценок, полученных на машинной модели и т.д.

Рассмотрим основные понятия теории планирования эксперимента.

При имитационном моделировании объект-оригинал рассматривается как «черный ящик», на который действует вектор входных переменных **[Xi] (i=1,2,…,n).** В теории планировании эксперимента входные переменные обычно называются ***факторы***. Откликом на воздействие входных факторов является вектор ***реакций*** **[Yj] (j=1,2,…,m)** (рис. 4.32).

Рис. 4.32

Так как исследуемые объекты имеют стохастическую природу, то реакции являются случайными компонентами вектора **[Yj]** и отражают влияние как учтенных, так и неучтенных факторов. Исследователь ставит задачу определения неизвестных функций $\mathbf{y_j}$ по результатам экспериментов.

Каждый фактор $\mathbf{x_i}$, может принимать в эксперименте одно или несколько значений, называемых ***уровнями***. Фиксированный набор уровней факторов определяет одно из возможных состояний рассматриваемой системы. Одновременно этот набор представляет собой условия проведения одного из возможных экспериментов.

Каждому фиксированному набору уровней факторов соответствует определённая точка в многомерном пространстве, называемая ***факторным пространством***. Реакцию (отклик) системы можно представить в виде семейства зависимостей: $\mathbf{y_j=Y_j(x_1, x_2,\ldots,x_n)}$; **(j = 1…m)**. Функцию Y_j, связанную с факторами, называют функцией отклика, а её геометрический образ – поверхностью отклика. Исследователю заранее не известен вид зависимостей Y_j, поэтому используются приближенные соотношения.

Эксперименты не могут быть реализованы во всех точках факторного пространства, а лишь в принадлежащих допустимой области, как это, например, показано для случая двух факторов $\mathbf{x_1}$ и $\mathbf{x_2}$ на рис. 4.33.

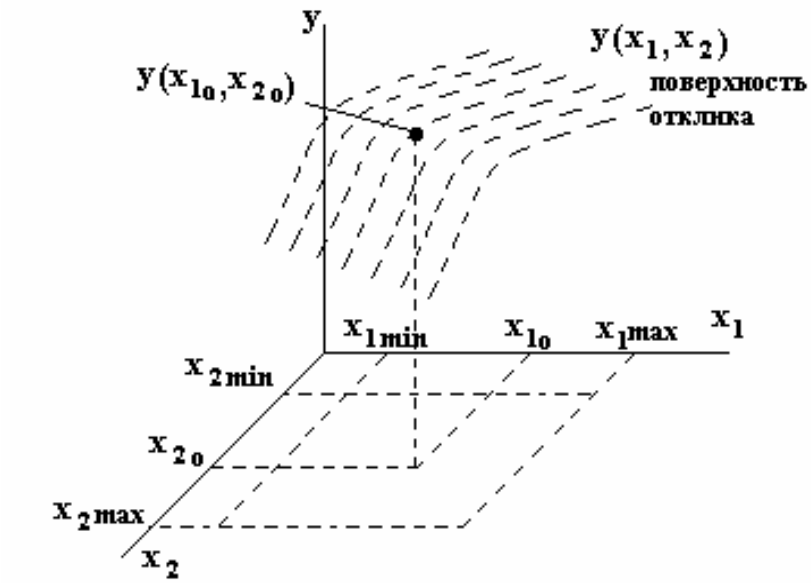


Рис. 4.33

На рисунке 4.33 показаны допустимые интервалы изменения факторов $[\mathbf{x_{1min}}- \mathbf{x_{1max}}]$, $[\mathbf{x_{2min}} - \mathbf{x_{2max}}]$ и точка факторного пространства $(\mathbf{x_{1o}},\mathbf{x_{2o}})$, в окрестности которой исследуется поведение функции отклика **y**.

Зависимости Y_j находятся по данным эксперимента. Единичный эксперимент выполняется в одной точке факторного пространства с целью накопления значений функции отклика (накопление выборки определенного объема) для последующей статистической обработки. Результат усреднения выборки будет являться случайной величиной с некоторым законом распределения и будет отклоняться от истинного значения функции отклика в данной точке факторного пространства.

Для построения некоторого приближения поверхности отклика надо решить вопрос о числе экспериментов и о наборах значений факторов в каждом из этих экспериментов.

Факторы при проведении эксперимента могут быть управляемыми и неуправляемыми, количественными или качественными, фиксированными и случайными. Фактор относится к изучаемым, если он включён в модель для изучения свойств системы. Количественными факторами являются интенсивности входящих потоков заявок, интенсивности потоков обслуживания, ёмкости накопителей, количество обслуживающих каналов и другие. Качественным факторам не соответствует числовая шкала (дисциплины постановки на очередь, обслуживание каналов и другие).

Фактор является управляемым, если его уровни целенаправленно выбираются экспериментатором.

При планировании эксперимента обычно изменяются несколько факторов. Основные требования, предъявляемые к факторам - независимость и совместимость. Совместимость означает, что все комбинации факторов осуществимы.

Различают тактическое и стратегическое планирование машинных экспериментов.

Тактическое планирование

Определяет условия проведения единичного эксперимента и чаще всего сводится к определению объема выборки, обеспечивающего заданную статистическую погрешность оценок числовых характеристик распределения функции отклика. При этом используют два вида оценок - оценки вероятностей событий и оценки математического ожидания.

Для оценок обоих видов предполагается знание истинных значений оцениваемых количественных характеристик. Поэтому объем выборки **N** определяется итеративно. Задаются некоторые начальные значения $\mathbf{N^{(1)}}$, определяются оценки статистических характеристик и, приравнивая их истинным значениям, определяют новые значения $\mathbf{N^{(2)}}$.Выполнив эксперимент для $\mathbf{N^{(2)}}$, уточняют значения оценок (находят новое значение $\mathbf{N^{(3)}}$) до тех пор, пока значения в цепи $\mathbf{N^{(1)} \text{—>} N^{(2)} \text{—>} N^{(3)} \text{—>}}$ … будут отличаться незначительно.

Стратегическое планирование эксперимента

При работе с имитационной моделью существенно понимание цели, поставленной экспериментатором. Если задача сводится к анализу, т. е. к определению вектора характеристик для некоторого набора параметров, то фактиче-

ски можно ограничиться тактическим планированием, рассматривая эту задачу как эксперимент в одной точке. Если решается задача синтеза, т. е. определения параметров, обеспечивающих требуемое качество функционирования исследуемой системы, необходимо знание зависимости характеристик от параметров. В этом случае возможны две постановки задачи.

Прямая задача

Проводится некоторая серия экспериментов и по их результатам необходимо определить функциональную зависимость, связывающую характеристики исследуемой системы с параметрами. При этом, обычно, задаются классом исследуемых зависимостей.

Рассмотренная задача названа прямой в том смысле, что не рассматривается вопрос о том, как был организован эксперимент, т. е. как выбиралось значение **N** и значения $\mathbf{x_i}$, для которых определялись значения $\mathbf{y_i}$.

Наиболее распространенным подходом здесь является метод «наименьших квадратов».

Обратная задача

Необходимо так спланировать эксперимент, чтобы при минимальном числе опытов построить экспериментальную функцию отклика (реакции), связывающую отклик (реакцию) системы **[Y]** с множеством входных независимых переменных факторов **[X]**, которыми можно варьировать при постановке эксперимента. В общем случае вид функции, связывающей входные переменные и реакцию системы заранее не известен.

Кроме того, наряду с управляемыми факторами на систему могут воздействовать неуправляемые и неконтролируемые факторы. Поэтому по результатам эксперимента может быть построена только приближенная зависимость. Эта зависимость - уравнение регрессии, которое связывает математическое ожидание отклика системы со значением переменных факторов.

Для получения независимых оценок коэффициентов в уравнении регрессии надо соответствующим образом спланировать эксперимент. Здесь используются методы полного и дробного факторного эксперимента.

4.10. ОБЩИЕ ПРОБЛЕМЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

Имитационные методы моделирования для анализа систем, преобладающими в которых являются стохастические воздействия, получили широкое применение в связи с развитием вычислительной техники.

Известный американский ученый Роберт Шеннон дал следующее определение: «Имитационное моделирование есть процесс конструирования модели реальной системы и постановки экспериментов на этой модели с целью либо понять поведение системы, либо оценить различные стратегии, обеспечивающие ее функционирование». В отличие от аналитических моделей, где для получения необходимой информации необходимо их «решать», в имитационном моделировании необходимо осуществлять «прогон» имитационных моде-

лей, т. е. подачу некоторой последовательности сигналов или данных на вход модели и фиксацию выходной информации. Происходит своего рода «выборка» конкретных состояний объекта моделирования из множества состояний. Насколько представительной окажется эта выборка, настолько результаты моделирования будут соответствовать действительности. Этот вывод показывает важность статистических методов оценки результатов имитации.

Применение имитационного моделирования целесообразно при наличии следующих условий, определенных Р. Шенноном:

1) не существует законченной математической постановки данной задачи, либо еще не разработаны аналитические методы решения сформулированной математической модели;

2) аналитические методы имеются, но математические процедуры столь сложны и трудоемки, что имитационное моделирование дает более простой способ решения задачи;

3) кроме оценки определенных параметров, желательно осуществить на имитационной модели наблюдение за ходом процесса в течение определенного периода.

Необходимо обозначить ряд проблем, возникающих в процессе моделирования систем. Исследователь должен акцентировать на них внимание и попытаться их разрешить, дабы избежать получения недостоверных сведений об изучаемой системе.

Первая проблема, которая касается и аналитических методов моделирования, состоит в нахождении «золотой середины» между упрощением и сложностью системы. Если модель слишком упрощена и в ней не учтены некоторые существенные факторы, то высока вероятность получить ошибочные данные. С другой стороны, если модель излишне сложная и в нее включены факторы, имеющие незначительное влияние на изучаемую систему, то резко повышаются затраты на создание такой модели и возрастает риск ошибки в логической структуре модели.

Вторая проблема заключается в искусственном воспроизводстве случайных воздействий окружающей среды. Этот вопрос очень важен, так как большинство технических и экономических систем являются стохастическими, и при их моделировании необходимо качественное воспроизведение случайности, в противном случае результаты, полученные на модели, могут не соответствовать действительности.

Третьей проблемой является оценка качества модели и полученных с ее помощью результатов (этап проблема актуальна и для аналитических методов). Адекватность моделей может быть оценена методом экспертных оценок, сравнением с другими моделями (уже подтвердившими свою достоверность), по полученным результатам. В свою очередь, для проверки полученных результатов часть из них сравнивается с уже имеющимися данными.

5. ОБЗОР АЛЬТЕРНАТИВНЫХ ПОДХОДОВ К МОДЕЛИРОВАНИЮ СЛОЖНЫХ СИСТЕМ

5.1. СЕТИ ПЕТРИ

Сети Петри – аппарат для моделирования динамических дискретных систем (преимущественно асинхронных параллельных процессов). Впервые описаны Карлом Петри в 1962 году Моделирование с использованием сетей Петри осуществляется на событийном уровне. Определяются, какие действия происходят в системе, какие состояния предшествовали этим действиям и какие состояния примет система после выполнения действия. Анализ результатов может сказать о том, в каких состояниях пребывала или не пребывала система, какие состояния в принципе не достижимы. Однако такой анализ не дает числовых характеристик, определяющих состояние системы.

В настоящее время определены и изучены разнообразные классы сетей Петри. Далее будут рассмотрены самые общие понятия и возможности их использования.

5.1.1. Определение сети Петри

Сеть Петри есть двудольный ориентированный граф. Напомним, что двудольный граф - это такой граф, множество вершин которого разбивается на два подмножества и не существует дуги, соединяющей две вершины из одного подмножества. Итак, сеть Петри - это набор

$$\mathbf{N = (T,P,A),\ T \cap P = \varnothing,}$$

где $\mathbf{T = \{t_1, t_2, ..., t_n\}}$ – подмножество вершин, называющихся переходами;
$\mathbf{P = \{p_1, p_2, ..., p_m\}}$ – подмножество вершин, называющихся позициями;
$\mathbf{A \subseteq (T \times P) \cup (P \times T)}$ – множество ориентированных дуг.

По определению дуги соединяют либо позицию с переходом, либо переход с позицией.

На рис. 5.1 приведен пример сети Петри в графическом представлении. Переходы обозначены черточками, а позиции - окружностями. Каждый переход **t** имеет набор входных **in{t}** и набор выходных **out{t}** дуг.

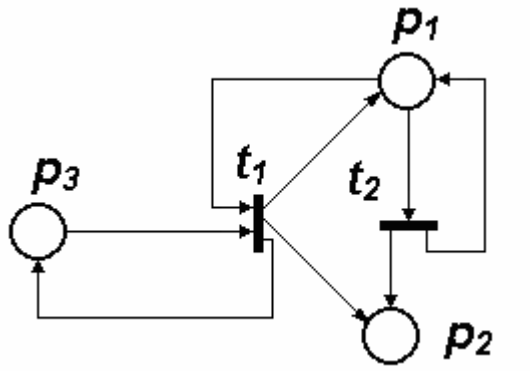


Рис. 5.1. Графическое изображение сети Петри

Сети Петри могут представляться также в форме продукционных правил вида: **in{t} ® out{t}.** Так, для сети, изображенной на рис. 5.1, правила будут следующими:

$$t_1: \{p_1, p_3\} \to \{p_1, p_2, p_3\},$$
$$t_2: \{p_1\} \to \{p_1, p_2\}.$$

Сети Петри наиболее интересны тем, что они позволяют представлять и изучать в динамике поведение системы параллельных процессов в любом другом дискретном устройстве или в программе.

5.1.2. Функционирование сети Петри

Сеть Петри можно понимать (интерпретировать) по-разному. Можно представить себе, что позиции представляют условия (буфер пуст, файл закрыт и т.п.), а переходы - события (посылка или получение сообщения в буфер, запись в файл).

Состояние сети Петри в каждый текущий момент определяется системой условий. Для того чтобы стало возможным и удобным задавать условие типа «в буфере находится 3 записи», в модель сети Петри добавляются маркеры, которые изображаются точками внутри позиции. В применении к программированию переходы можно представлять как процедуры, а позиции - как переменные или буфер.

Распределение маркеров по позициям называют ***маркировкой сети.*** Маркер свидетельствует о том, что переменная (буфер) имеет значение, а если позиция имеет, к примеру, 3 маркера, то это может интерпретироваться как наличие трех разных значений в буфере. Если позиции содержат маркеры, то сеть называется маркированной. Начальное распределение маркеров задает начальную маркировку $\mathbf{M_0}$ сети. Маркировка сети определяет ее текущее состояние. Функция **M** представляется вектором, в котором **i**-й компонент задает маркировку места $\mathbf{p_i}$. Например, начальная маркировка сети, которая в начальном состоянии содержит один маркер в позиции $\mathbf{p_3}$, представляется вектором $\mathbf{M_0 = (0,0,1)}$ (рис. 5.2).

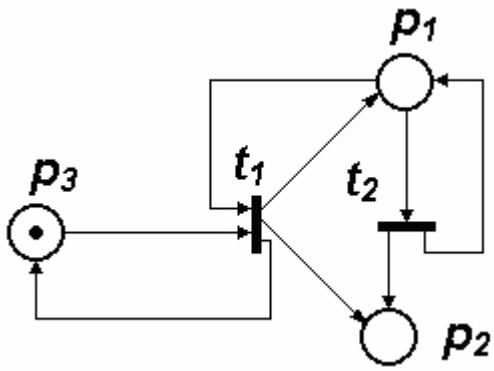


Рис. 5.2. Сеть в начальном состоянии

Маркеры могут перемещаться в сети. Каждое изменение маркировки называют ***событием***, причем каждое событие связано с определенным перехо-

дом. Считается, что события происходят мгновенно и разновременно при выполнении некоторых условий.

Каждому условию в сети Петри соответствует определенная позиция. Совершению события соответствует срабатывание (возбуждение или запуск) перехода, при котором маркеры из входных позиций этого перехода перемещаются в выходные позиции. Последовательность событий образует моделируемый процесс.

Правила срабатывания переходов следующие:

– переход может сработать, если есть хотя бы один маркер во всех его входных позициях;

– при срабатывании перехода из всех входных позиций забирается по одному маркеру и во все выходные позиции добавляется по одному маркеру.

Рассмотрим последовательность состояний сети Петри в ходе срабатывания переходов. Начальная разметка $\mathbf{M_0 = (0,0,1)}$ показана на рис. 5.2. В этом состоянии может сработать только переход $\mathbf{t_1}$. Маркировка сети $\mathbf{M_1 = (1,1,1)}$ после срабатывания $\mathbf{t_1}$ показана на рис. 5.3.

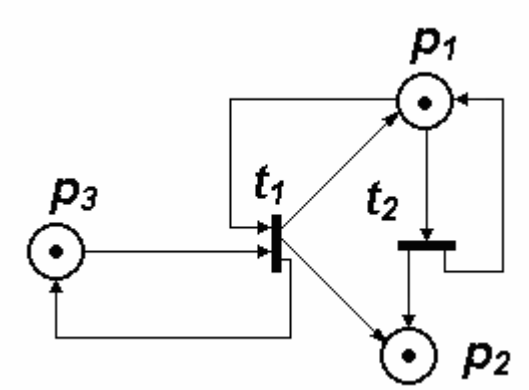


Рис. 5.3. Маркировка сети после срабатывания перехода $\mathbf{t_1}$

Теперь возможно одновременное срабатывание переходов $\mathbf{t_1}$ и $\mathbf{t_2}$. Маркировка $\mathbf{M_2 = (2,3,1)}$ после их срабатывания показана на рис. 5.4.

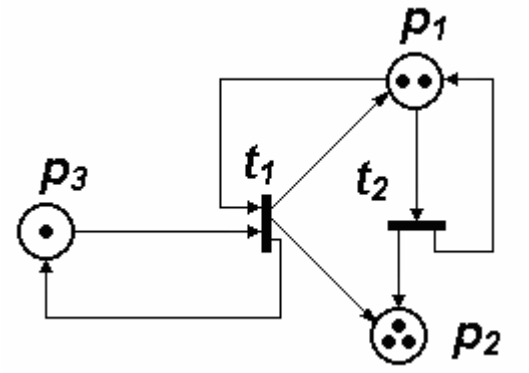


Рис. 5.4. Маркировка сети после срабатывания переходов $\mathbf{t_1}$ и $\mathbf{t_2}$

Если представить себе переход как процедуру, то она корректно выполняется при наличии значений всех своих аргументов и вырабатывает значения всех выходных переменных.

В другой интерпретации переход может представлять некоторое устройство, которое может сработать, если выполнились все входные условия.

Можно вводить ряд дополнительных правил и условий в алгоритмы моделирования, получая ту или иную разновидность сетей Петри. Так, прежде всего, полезно ввести модельное время, чтобы моделировать не только последовательность событий, но и их привязку ко времени. Это осуществляется приданием переходам веса – продолжительности (задержки) срабатывания, которую можно определять, используя задаваемый при этом алгоритм. Полученную модель называют ***временной сетью Петри.***

Если задержки являются случайными величинами, то сеть называют ***стохастической сетью Петри.*** В стохастических сетях возможно введение вероятностей срабатывания возбужденных переходов. Так, на рис. 5.5 представлен фрагмент сети Петри, иллюстрирующий альтернативную ситуацию – маркер в позиции **p** может запустить либо переход $\mathbf{t_1}$, либо переход $\mathbf{t_2}$. В стохастической сети предусматривается вероятностный выбор срабатывающего перехода в таких ситуациях. Если несколько переходов готовы сработать, то срабатывает один из них (любой) или некоторые из них, или все.

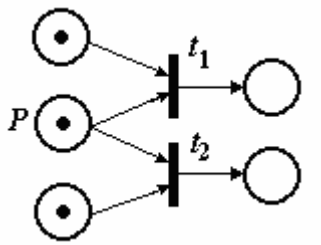


Рис. 5.5. Альтернативная ситуация

Если задержки определяются как функции некоторых аргументов, которыми могут быть количества маркеров в каких-либо позициях, состояния некоторых переходов и т.п., то имеем ***функциональную сеть Петри***.

Во многих задачах динамические объекты могут быть нескольких типов, и для каждого типа нужно вводить свои алгоритмы поведения в сети. В этом случае каждый маркер должен иметь хотя бы один параметр, обозначающий тип маркера. Такой параметр обычно называют ***цвет.*** Его можно использовать как аргумент в функциональных сетях. Сеть при этом называют ***цветной сетью Петри.***

Среди других разновидностей сетей Петри следует упомянуть ***ингибиторные сети Петри***, характеризующиеся тем, что в них возможны запрещающие (ингибиторные) дуги. Наличие маркера во входной позиции, связанной с переходом ингибиторной дугой, означает запрещение срабатывания перехода.

Введенные понятия поясним на следующих примерах.

Пример 5.1

Требуется описать с помощью сети Петри работу группы пользователей на единственной рабочей станции при заданных характеристиках потока за-

просов на пользование станции и характеристиках поступающих задач. Сеть Петри представлена на рис. 5.6.

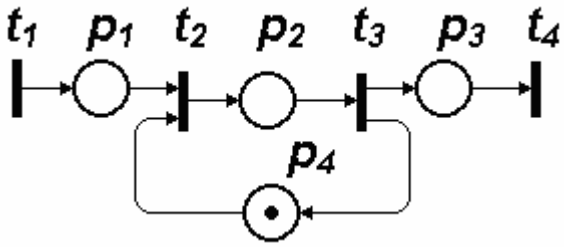


Рис. 5.6. Сеть Петри для примера 5.1 в исходном состоянии

Здесь переходы связаны со следующими событиями: $\mathbf{t_1}$ – поступление запроса на использование рабочей станции, $\mathbf{t_2}$ –занятие станции, $\mathbf{t_3}$ – освобождение станции, $\mathbf{t_4}$ –выход обслуженной заявки.

Позиция $\mathbf{p_4}$ используется для отображения состояния рабочей станции: если в $\mathbf{p_4}$ имеется метка, то станция свободна, и пришедшая заявка (рис. 5.7) вызывает срабатывание перехода $\mathbf{t_2}$ (рис. 5.8). Пока эта заявка не будет обслужена, метки в $\mathbf{p_4}$ не будет, следовательно, пришедшие в позицию $\mathbf{p_1}$ запросы вынуждены ожидать срабатывания перехода $\mathbf{t_3}$.

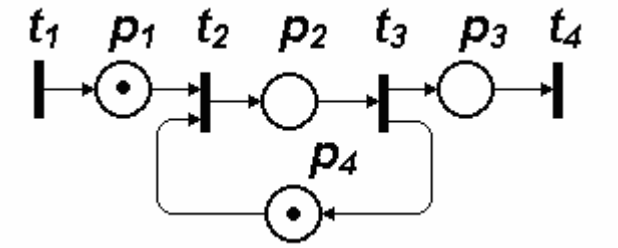


Рис. 5.7. Приход заявки в примере 5.1

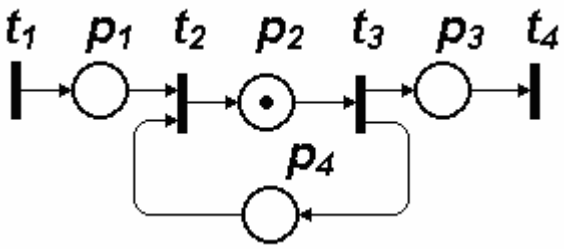


Рис. 5.8. Поступление заявки на обслуживание для примера 5.1

Пример 5.2

Требуется описать с помощью сети Петри процессы возникновения и устранения неисправностей в некоторой технической системе, состоящей из множества однотипных блоков; в запасе имеется один исправный блок. Известны статистические данные об интенсивностях возникновения отказов и длительностях таких операций, как поиск неисправностей, замена и ремонт отказавшего блока. Поиск и замену отказавшего блока производит одна бригада, а ремонт замененного блока - другая. Сеть Петри показана на рис.5.9.

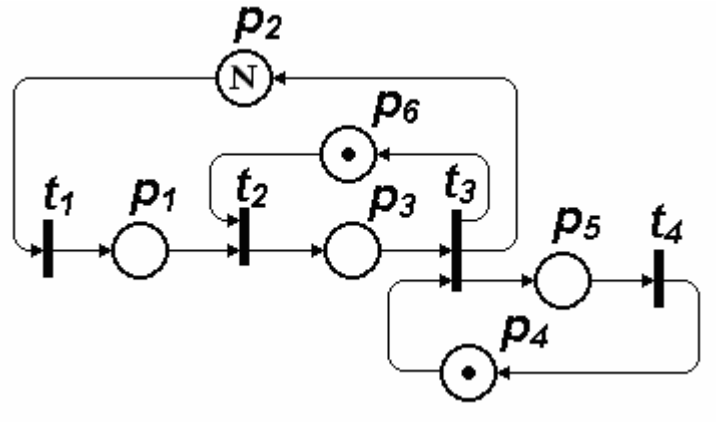


Рис. 5.9. Сеть Петри для примера 5.2

Отметим, что при числе меток в позиции, равном **N**, можно в ней не ставить **N** точек, а записать в позиции значение **N**. В нашем примере значение **N** в позиции $\mathbf{p_2}$ соответствует числу имеющихся в системе блоков.

Переходы отображают следующие события: $\mathbf{t_1}$ – отказ блока, $\mathbf{t_2}$ – поиск неисправного блока, $\mathbf{t_3}$ – его замена, $\mathbf{t_4}$ – окончание ремонта.

Очевидно, что при непустой позиции $\mathbf{p_2}$ переход срабатывает, но с задержкой, равной вычисленному случайному значению моделируемого отрезка времени между отказами. После выхода маркера из $\mathbf{t_1}$ он попадает через $\mathbf{p_1}$ в $\mathbf{t_2}$, если имеется метка в позиции $\mathbf{p_6}$. Это означает, что обслуживающая систему бригада специалистов свободна и может приступить к поиску возникшей неисправности. В переходе $\mathbf{t_2}$ метка задерживается на время, равное случайному значению длительности поиска неисправности. Далее маркер оказывается в позиции $\mathbf{p_3}$. Теперь, если имеется запасной блок (маркер в $\mathbf{p_4}$), то запускается переход $\mathbf{t_3}$, из которого маркеры выйдут в позиции $\mathbf{p_2}$, $\mathbf{p_5}$ и $\mathbf{p_6}$ через отрезок времени, требуемый для замены блока. После этого в $\mathbf{t_4}$ имитируется восстановление неисправного блока.

Рассматриваемая модель описывает функционирование системы в условиях, когда отказы могут возникать как в рабочем, так и в неисправном состояниях системы. Поэтому не исключены ситуации, при которых более чем один маркер окажется в позиции $\mathbf{p_1}$.

5.1.3. Анализ сетей Петри

Анализ сложных систем на базе сетей Петри можно выполнять посредством имитационного моделирования. В частности СМО можно представлять в виде сетей Петри. При этом задают входные потоки заявок и определяют соответствующую реакцию системы. Выходные параметры СМО рассчитывают путем обработки накопленного при моделировании статистического материала.

Возможен и другой подход к использованию сетей Петри для анализа объектов, исследуемых на системном уровне. Он не связан с имитацией процессов и основан на исследовании таких свойств сетей Петри, как достижимость, ограниченность, безопасность, сохраняемость, живость.

Достижимость – возможность перехода сети из одного заданного состояния в другое.

Ограниченность имеет место, если число меток в любой позиции сети не может превысить значения некоторого значения **K**. При проектировании информационных систем определение **K** позволяет обоснованно выбирать емкости накопителей. Возможность неограниченного роста числа меток свидетельствует об опасности неограниченного роста длин очередей.

Безопасность – это частный случай ограниченности, а именно **K** = 1.

Сохраняемость характеризуется постоянством загрузки ресурсов, т.е.

$$\sum \mathbf{A_i N_i} = \mathbf{const},$$

где $\mathbf{N_i}$ - число маркеров в **i**-й позиции, $\mathbf{A_i}$ - весовой коэффициент.

Живость определяется возможностью срабатывания любого перехода при функционировании моделируемого объекта. Отсутствие живости означает либо избыточность аппаратуры в проектируемой системе, либо свидетельствует о возможности возникновения зацикливаний, тупиков, блокировок.

В основе исследования перечисленных свойств сетей Петри лежит анализ достижимости.

Один из методов анализа достижимости – построение ***графа достижимости***. Начальная вершина графа отображает начальное состояние с маркировкой $\mathbf{M_0}$. Дуга из $\mathbf{M_i}$ в $\mathbf{M_j}$ означает событие $\mathbf{M_i} \to \mathbf{M_j}$ и соответствует срабатыванию перехода. В сложных сетях граф может содержать чрезмерно большое число вершин и дуг. Однако при построении графа можно не отображать все вершины, так как многие из них являются дублями (действительно, от состояния с маркировкой $\mathbf{M_k}$ всегда порождается один и тот же подграф вне зависимости от того, из какого состояния система пришла в $\mathbf{M_k}$). Тупики обнаруживаются по отсутствию разрешенных переходов из какой-либо вершины, т.е. по наличию терминальных вершин. Неограниченный рост числа маркеров в какой-либо позиции свидетельствует о нарушениях ограниченности.

Приведем примеры анализа достижимости.

Пример 5.3

На рис. 5.10 и 5.11 представлены сеть Петри и граф достижимых разметок.

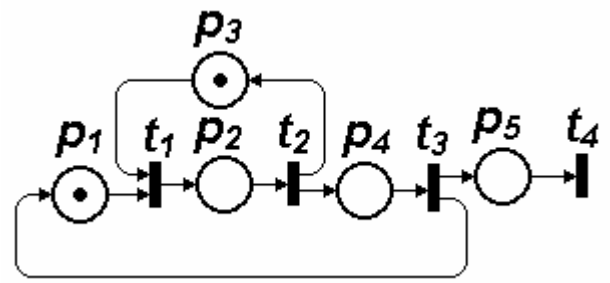


Рис. 5.10. Сеть Петри для примера 5.3

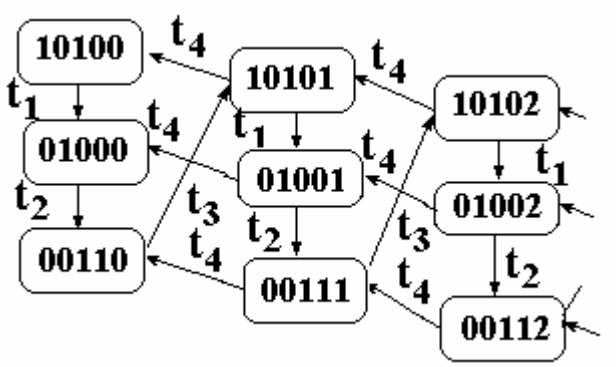


Рис. 5.11. Граф достижимых разметок сети для примера 5.3

На рисунке вершины графа изображены в виде маркировок, дуги помечены срабатывающими переходами.

Сеть является неограниченной и живой, так как метки могут накапливаться в позиции $\mathbf{p_5}$, срабатывают все переходы, тупики отсутствуют.

Пример 5.4

Сеть Петри, моделирующая работу двух процессоров с одной общей памятью, приведена на рис. 5.12.

Здесь переходы связаны со следующими событиями: $\mathbf{t_1}$ – готовность памяти к обмену данными; $\mathbf{t_2}$ – «захват» памяти первым процессором; $\mathbf{t_4}$ – обмен данными между первым процессором и памятью; $\mathbf{t_6}$– окончание обмена данными; $\mathbf{t_3}$, $\mathbf{t_5}$, $\mathbf{t_7}$ - то же для второго процессора.

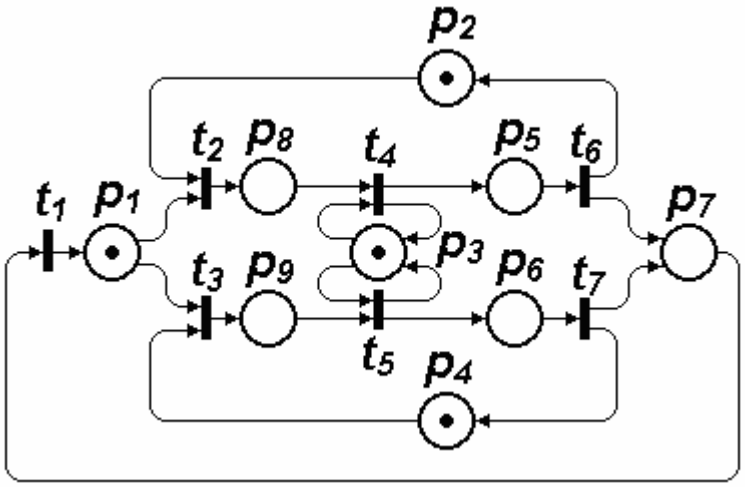


Рис. 5.12. Сеть Петри для примера 5.4

Граф достижимых разметок представлен на рис. 5.13.

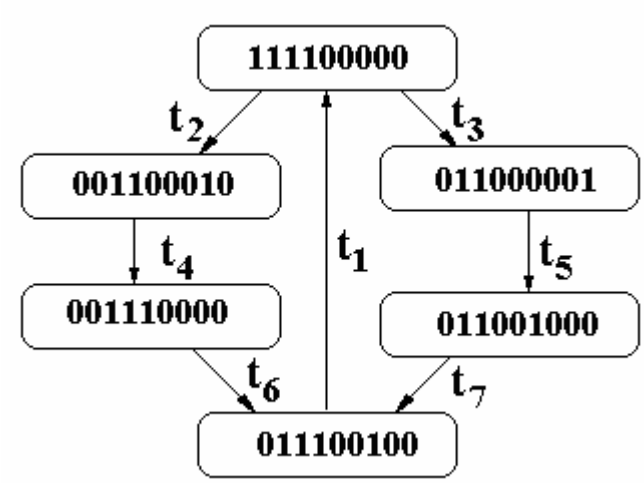


Рис. 5.13. Граф достижимых разметок сети для примера 5.4

Можно сделать вывод, что сеть является безопасной, живой, все разметки достижимы.

5.2. НЕЙРОННЫЕ СЕТИ

В последние десятилетия в мире бурно развивается новая перспективная область прикладной математики – искусственные нейронные сети. Актуальность исследований в этом направлении подтверждается широким спектром применений нейронных сетей. Это автоматизация процессов распознавания образов, прогнозирование, создание экспертных систем и многое другое. С помощью нейронных сетей можно, например, создавать самообучающиеся системы, способные управлять техническими устройствами, выполнять распознавание видео- и аудио- сигналов, предсказывать рыночные показатели.

Одним из перспективных подходов является использование искусственных нейронных сетей в компьютерном моделировании и построении вычислительных и информационных систем новых поколений.

5.2.1. Понятие нейронной сети

Искусственная нейронная сеть – это математическая модель системы соединённых и взаимодействующих между собой простых элементов (искусственных нейронов). Так же называют и устройства для параллельных вычислений.

Искусственная нейронная сеть осуществляет преобразование вектора входных сигналов (воздействий) **[X]** в вектор выходных сигналов **[Y]**. Интерпретация векторов **[X]** и **[Y]** зависит от постановки решаемой задачи и предметной области. Искусственная нейронная сеть в настоящее время рассматривается как грубая (первого приближения) модель мозга человека и других живых существ. Искусственные нейронные сети нашли применение в следующих областях.

Кластеризация и классификация. Нейронная сеть способна на предъявленных ей «эталонных» векторах **[X]** выделить характеризующие их призна-

ки, накопить их и использовать в дальнейшем для оценки степени близости вновь предъявляемых входных векторов к эталонным (классификация). Некоторые типы нейронных сетей могут самостоятельно выделить во множестве предъявляемых входных векторов **[X]** обособленные группы, определяемые «усредненными» векторами $\bar{X}$ (кластеризация).

Аппроксимация. Нейронные сети дают возможность с любой требуемой точностью аппроксимировать нелинейную непрерывную функцию **Y = f(X)**, т.е. построить формальную математическую модель объекта.

Прогнозирование. Нейронные сети с обратными связями (рекуррентные нейронные сети) способны предсказывать будущее состояние моделируемого объекта/процесса по его состоянию на **k** предыдущих шагах модельного времени и текущему воздействию **[X]**.

Традиционно используемым для описания нейронных сетей математическим языком является аппарат векторной и матричной алгебры.

Существует много вариантов построения ИНС. Для их классификации используются следующие основные критерии:

- тип нейронов, составляющих сеть;
- количество слоев нейронов в сети;
- направление передачи сигналов в сети;

Сеть, состоящая целиком из нейронов одного типа, называется ***однородной***, если же в ней комбинируются слои нейронов разного типа, то сеть ***гибридная***.

Сеть, все нейроны которой расположены в одной «плоскости» (т.е. отсутствует хотя бы одна непосредственная связь выхода одного нейрона со входом другого), называется ***однослойной***, иначе сеть ***многослойная***.

Сеть называется ***однонаправленной***, если в ней отсутствуют обратные связи (т.е. нет передачи сигнала с последующих слоев на предыдущие). Сеть с обратными связями называется ***рекуррентной***.

Основные достоинства нейронных сетей состоят в следующем:

- способность к адаптации (обучению и самообучению);
- параллельность обработки информации;
- устойчивость к отдельным сбоям в работе.

5.2.2. Искусственный нейрон

Искусственный нейрон имитирует в первом приближении свойства природной нервной клетки мозга (биологического нейрона). Обобщенная схема искусственного нейрона представлена на рис. 5.14.

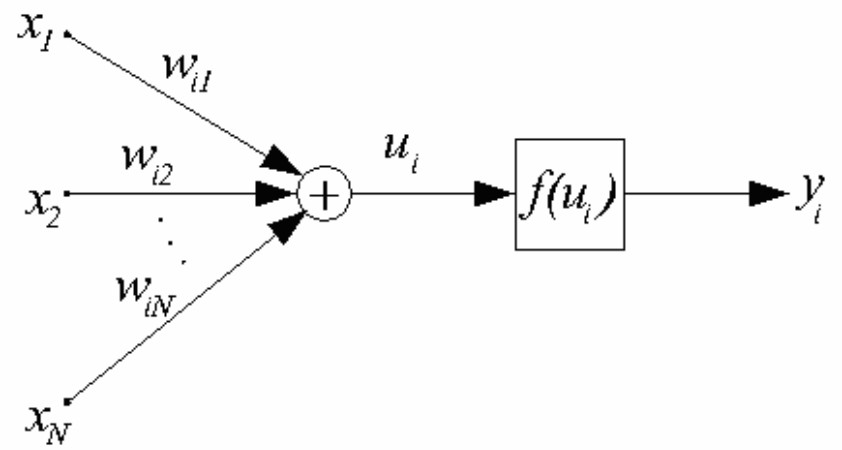


Рис. 5.14. Обобщенная схема искусственного нейрона

На вход искусственного нейрона поступает некоторое множество сигналов, образующих вектор входных сигналов **[X] = {x_1, x_2,…x_N}**. В общем случае **x_j** – действительные числа, возможно, размерные, но чаще нормированные. Во многих моделях **x_j** дискретны.

Для каждого искусственного нейрона существует ***вектор весов*** входных сигналов **[W_i] = {w_{i1}, w_{i2},…w_{iN}},** элементы **w_{ij}** которого в общем случае действительные числа.

Веса входных сигналов используются для ***вычисления взвешенной суммы*** **u_i** входных сигналов **i**-го нейрона по формуле:

$$u_i = \sum_{j=1}^{N} w_{ij} x_j . \qquad (5.1)$$

Взвешенная сумма входных сигналов **u_i** служит аргументом ***функции активации*** нейрона **f(u_i)**, определяющей значение выходного сигнала **y_i.** Виды активационных функций будут рассмотрены ниже.

В простейших однослойных (без обратных связей и каскадов) сетях входные сигналы **x_j** нейрона - входные сигналы сети в целом, а выходные сигналы **y_i** нейрона - выходные сигналы сети в целом. В многослойных сетях роль входных сигналов некоторых нейронов играют выходные сигналы предыдущих слоев нейронной сети.

Как уже упоминалось ранее, одним из типичных назначений искусственных нейронов и сетей на их основе является классификация и распознавание векторов входных сигналов **[X]**. В такой задаче вычисленное по входному вектору **[X]** значение **y_i** определяет принадлежность входного вектора тому или иному **i**-му классу. Например, в качестве значений **x_j** входного вектора могут выступать биометрические данные пациента (температура тела, кровяное давление, параметры крови и т.п.), тогда выходной сигнал нейрона **y_i** может определять степень уверенности в наличии у пациента болезни **H_i**.

Понятно, что степень успеха в классификации отдельным нейроном и сетью в целом зависит в общем случае от «правильности» подбора (назначения) весовых коэффициентов **w_{ij}** и функции активации **f(u_i).** Однако на практике,

функции активации, как правило, назначаются однократно и варьированию не подлежат. Таким образом, объектом подбора служат только весовые коэффициенты w_{ij}.

Для отыскания наилучших значений элементов вектора $[W_i]$ с точки зрения решения задачи классификации входных векторов **[X],** необходимо ***обучение сети***, предваряющее собственно этап классификации. Различают два основных режима обучения: «с учителем» и «без учителя».

Обучение с учителем предполагает, что для каждого входного вектора **[X]** существует выходной целевой вектор **[D]**, представляющий собой требуемое значение выходного вектора **[Y]**. Вместе они называются обучающей парой $<X^k,D^k>$. Обычно сеть обучается на некотором числе таких обучающих пар. Обучение ведется следующим образом. По заданному входному вектору $[X^k]$ сеть вычисляет выходной вектор, который сравнивается с соответствующим целевым вектором $[D^k]$. Разность (ошибка) с помощью обратной связи подается в сеть, и веса изменяются в соответствии с алгоритмом, стремящимся минимизировать ошибку. Векторы обучающего множества предъявляются последовательно, вычисляются ошибки и веса подстраиваются для каждого вектора до тех пор, пока ошибка по всему обучающему массиву не достигнет приемлемо низкого уровня.

Обучение без учителя не нуждается в целевом векторе для выходов и, следовательно, не требует сравнения с предопределенными идеальными ответами. Обучающее множество состоит лишь из входных векторов. Обучающий алгоритм подстраивает веса сети так, чтобы получались согласованные выходные векторы, т.е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы. Процесс обучения, следовательно, выделяет статистические свойства обучающего множества и группирует сходные векторы в классы.

5.2.3. Основные виды активационных функций искусственных нейронов

Пороговая функция активации

При использовании пороговой функции активации искусственный нейрон считается пороговым бинарным элементом.

Выходной сигнал нейрона может принимать только два значения {0,1} по следующему правилу:

$$y_i = \begin{cases} 1, & \text{если } u_i \geq C \\ 0, & \text{если } u_i < C \end{cases}, \qquad (5.2)$$

где **C** – некоторая константа.

График пороговой функции (5.2) представлен на рис.5.15.

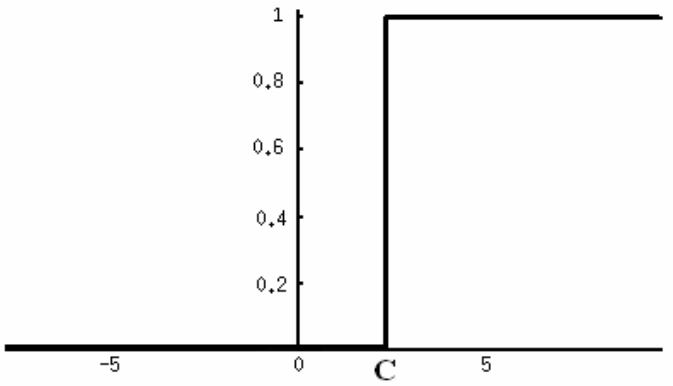


Рис. 5.15. График пороговой функции

Обучение нейрона (отыскание весовых коэффициентов $\mathbf{w_{ij}}$) требует учителя и сводится к задаче минимизации целевой функции, которая согласно методу «наименьших квадратов» равна:

$$\mathbf{E(W_i)} = \frac{\mathbf{1}}{\mathbf{2}} \sum_{\mathbf{k=1}}^{\mathbf{p}} \mathbf{(y_i^k - d_i^k)^2}, \qquad (5.3)$$

где $\mathbf{y_i}$ – реальный выходной сигнал; $\mathbf{d_i}$ - ожидаемое значение выходного сигнала.

На практике для обучения нейрона чаще всего используется следующий простой алгоритм.

1. Выбираются (как правило, случайно) начальные значения весов $\mathbf{w_{ij}}$.
2. Для каждой обучающей пары $\mathbf{<X^k,d^k>}$ выполняется ряд циклов уточнения значений входных весов по формуле:

$$\mathbf{w_{ij}(t+1) = w_{ij}(t) + dw_{ij}(t),} \qquad (5.4)$$

где: $\mathbf{dw_{ij}(t) = 0,}$ если $\mathbf{y_i(t) = d_i^k,}$

$\mathbf{dw_{ij}(t) = x_j^k,}$ если $\mathbf{y_i(t) = 0,}$ а $\mathbf{d_i^k = 1,}$

$\mathbf{dw_{ij}(t) = -x_j^k,}$ если $\mathbf{y_i(t) = 1,}$ а $\mathbf{d_i^k = 0.}$

3. Процесс обработки текущей обучающей пары завершается либо на цикле, в котором все $\mathbf{dw_{ij}(t) = 0}$, либо после достижения предельного количества циклов.

Функционирование обученного порогового нейрона в режиме классификации легко проиллюстрировать графически на следующем примере.

Пример 5.5

Нейрон, структурная схема которого дана на рис. 5.16. имеет два рабочих входа $\mathbf{x_1}$ и $\mathbf{x_2}$, .

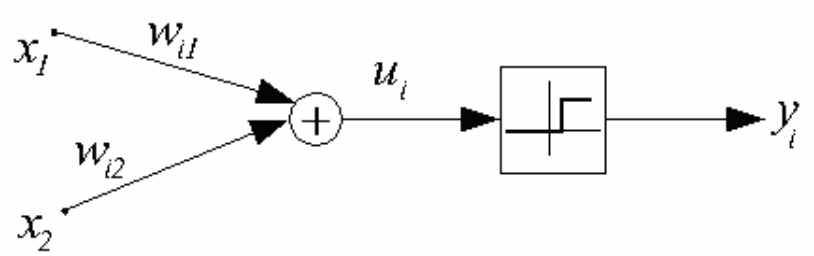


Рис. 5.16. Структурная схема нейрона из примера 5.5

Для такого нейрона сумма входных сигналов будет иметь вид:

$$u_i = w_{i1}x_1 + w_{i2}x_2.$$

Это выражение определяет плоскость в трехмерном пространстве $<x_1, x_2, u_i>$. Эта плоскость пересекается с плоскостью $<x_1, x_2>$ по линии, определяемой следующим уравнением:

$$u_i = w_{i1}x_1 + w_{i2}x_2 - C = 0.$$

Данная линия разбивает пространство входных сигналов на две области: в одной из них (заштрихованной) значения $u_i > 0$, и, следовательно, функция активации принимает значение 1; в другой - $u_i < 0$, и функция активации равна нулю (рис. 5.17).

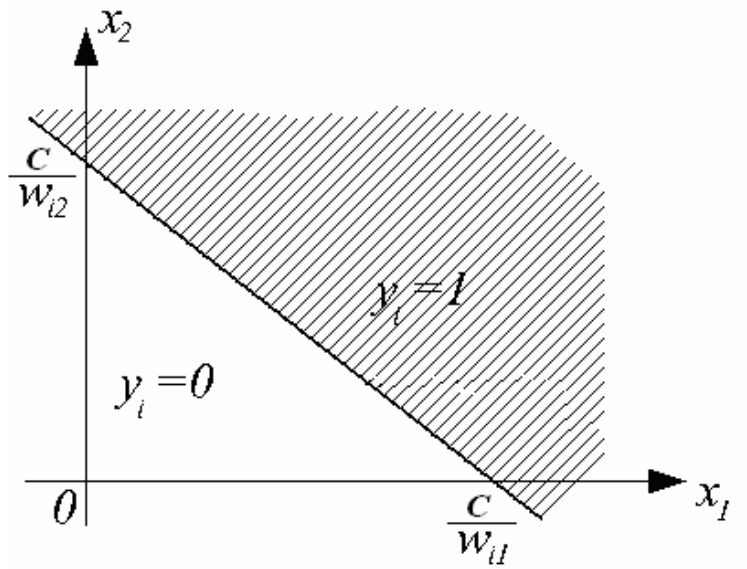


Рис. 5.17. Разделение пространства входных данных двухвходовым пороговым нейроном

Таким образом, наглядно видно, что пороговый нейрон является простейшим линейным классификатором.

S-образная функция активации

Основной недостаток пороговой функции – ее разрывность, которая ограничивает применение многих известных методов оптимизации при обучении сети. S-образная функция не имеет этого недостатка.

Здесь в качестве функции активации $f(u_i)$ выступают функции, графики которых похожи на букву «**S**». На практике используются следующие виды функций.

S-образная функция, изменяющаяся в диапазоне [0,1], обычно представляется формулой:

$$y_i = \frac{1}{1 + \exp(-bu_i)}. \qquad (5.5)$$

Аналогичная функция для диапазона [-1,1] может быть задана через гиперболический тангенс:

$$y_i = \tanh(-bu_i). \qquad (5.6)$$

Графики функции (5.5) представлены на рис.5.18.

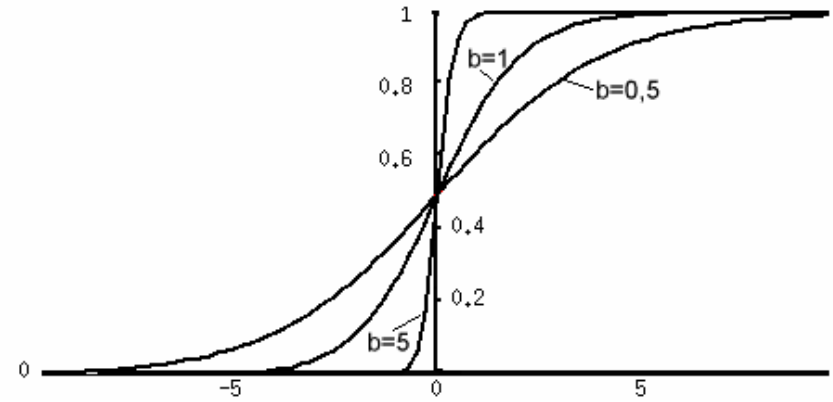


Рис. 5.18. Графики S-образной функции (5.5)

Из рисунка видно, что при уменьшении **b** функция становится более пологой, в пределе при **b** = 0 вырождаясь в горизонтальную линию на уровне 0,5. При увеличении **b** функция приближается по внешнему виду к функции единичного скачка с порогом в точке **x** = 0.

Следует отметить, что S-образные функции дифференцируемы на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того, они обладают свойством усиливать слабые сигналы лучше, чем сильные, так как они соответствуют областям аргументов, где функции имеют пологий наклон. Это предотвращает насыщение нейронов большими сигналами.

Для обучения S-образного нейрона используется стратегия «с учителем», однако, в отличие от порогового нейрона, для поиска минимума целевой функции:

$$\mathbf{E(W_i)} = \frac{\mathbf{1}}{\mathbf{2}} \sum_{\mathbf{k=1}}^{\mathbf{p}} (\mathbf{y_i^k} - \mathbf{d_i^k})^2 .$$

Здесь используются методы поисковой оптимизации первого порядка, в которых целенаправленное изменение весовых коэффициентов $\mathbf{w_{ij}}$ осуществляется в направлении отрицательного градиента $\mathbf{E(W_i)}$.

Линейные функции активации

Нередко в качестве активационных функций искусственных нейронов используют линейные функции. При этом либо функция имеет простейший вид:

$$\mathbf{y_i} = \mathbf{u_i}, \qquad (5.7)$$

либо функция линейна на определенном участке (линейная пороговая функция):

$$\mathbf{y_i} = \begin{cases} \mathbf{1}, & \text{если } \mathbf{u_i} > \mathbf{C} \\ \mathbf{u_i / C}, & \text{если } \mathbf{0} < \mathbf{u_i} < \mathbf{C} \\ \mathbf{0}, & \text{если } \mathbf{u_i} < \mathbf{0} \end{cases}, \qquad (5.8)$$

где **C** – некоторая константа.

График функции (5.8) представлен на рис.5.19.

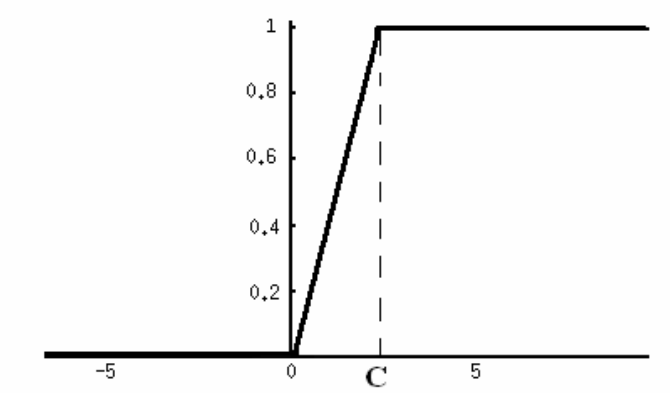


Рис. 5.19. График линейной пороговой функции (5.8)

5.2.4. Виды простейших нейронных сетей

Принцип параллельной обработки сигналов в нейронных сетях достигается путем объединения большого числа нейронов в так называемые слои, где нейроны соединены определенным образом. При этом обработка взаимодействия всех нейронов ведется послойно.

Теоретически число слоев и число нейронов в каждом слое может быть произвольным, однако фактически оно ограничено ресурсами компьютера или специализированной микросхемы, на которых обычно реализуется сеть. Чем сложнее сеть, тем более сложные задачи она способна решать.

Однослойный перцептрон

В качестве примера простейшей нейронной сети рассмотрим сеть, состоящую из нейронов, имеющих активационную функцию в виде единичного скачка, т.е. функцию вида (5.2). Такую сеть обычно называют ***перцептроном***. Схема однослойного перцептрона из трех нейронов приведена на рис. 5.20.

Перцептрон работает следующим образом. На **n** входов поступают сигналы, проходящие на 3 нейрона, которые выдают три выходных сигнала:

$$\mathbf{y}_j = \mathbf{f}\left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{w}_{ij}\right) \quad \mathbf{j} = 1,2,3. \qquad (5.9)$$

Очевидно, что все весовые коэффициенты одного слоя нейронов можно свести в матрицу **W**, в которой каждый элемент $\mathbf{w_{ij}}$ задает величину **i**-й связи **j**-го нейрона. Таким образом, процесс, происходящий в нейронной сети, может быть записан в матричной форме:

$$\mathbf{Y=F(XW),} \qquad (5.10)$$

где **X** и **Y** – соответственно входной и выходной векторы, **F(V)** – активационная функция, применяемая поэлементно к компонентам вектора **V**.

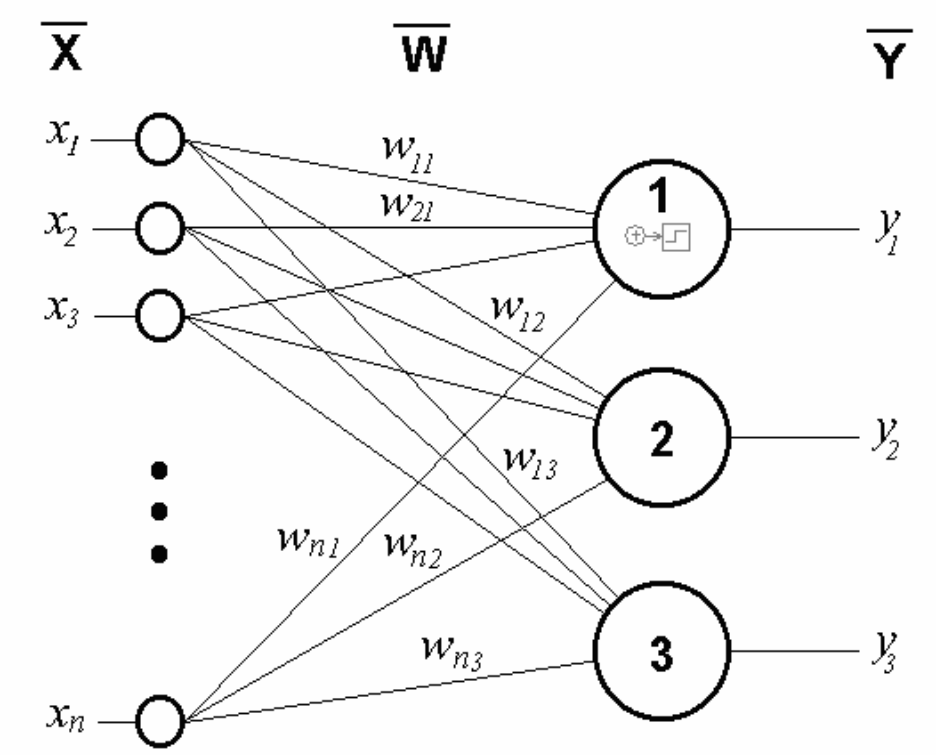


Рис. 5.20. Схема однослойного перцептрона

Рассмотрим один из алгоритмов обучения сети, изображенной на рис. 5.20 по стратегии «с учителем».

1. Установить начальные значения элементов весовой матрицы **W** (обычно случайные значения).

2. Подать на входы сети один из входных векторов и вычислить ее выход.

3. Если выход правильный, перейти на шаг 4.

Иначе вычислить разницу между идеальным и полученным значениями выхода:

$$d = \mathbf{d}_j - \mathbf{y}_j.$$

Модифицировать веса в соответствии с формулой:

$$\mathbf{w}_{ij}(\mathbf{t}+\mathbf{1}) = \mathbf{w}_{ij}(\mathbf{t}) + nd\mathbf{x}_i,$$

где **t** и **t+1** – номера соответственно текущей и следующей итераций; n – коэффициент скорости обучения, **0<n<1**; **i** – номер входа; **j** – номер нейрона в слое.

Очевидно, что если $\mathbf{y_j} < \mathbf{d_j}$, весовые коэффициенты будут увеличены и тем самым уменьшат ошибку. В противном случае они будут уменьшены, и $\mathbf{y_j}$ тоже уменьшится, приближаясь к $\mathbf{d_j}$.

4. Повторять цикл с шага 2, пока сеть не перестанет ошибаться.

На втором шаге на разных итерациях поочередно в случайном порядке предъявляются все возможные входные вектора. К сожалению, нельзя заранее определить число итераций, которые потребуется выполнить, а в некоторых случаях и гарантировать полный успех.

Многослойные перцептроны

Каждый нейрон перцептрона является формальным пороговым элементом, принимающим значения (0,1) Таким образом, при заданных значениях весов и порогов нейрон имеет определенное значение выходной активности для каждого возможного вектора входов. В примере 5 было показано, что множество входных векторов, при которых нейрон активен ($\mathbf{y_i} = 1$), отделено от множества векторов, на которых нейрон пассивен ($\mathbf{y_i} = 0$), *гиперплоскостью*.

Следовательно, нейрон способен ***отделить*** (иметь различный выход) только такие два множества входных векторов, для которых имеется гиперплоскость, отсекающая одно множество от другого. Такие множества называют ***линейно разделимыми***. Однако далеко не все функции обладают данным свойством. Например, на рис. 5.21 представлена одна из ситуаций, когда множества белых и черных точек разделить одной прямой нельзя, вследствие линейной неразделимости.

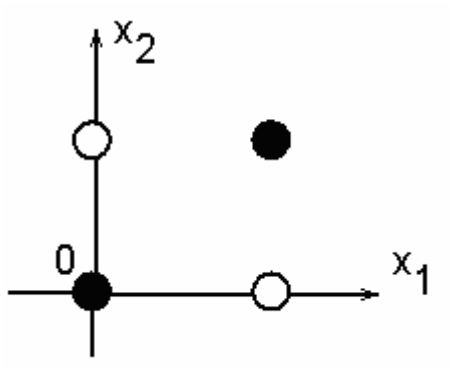


Рис. 5.21

При возрастании числа аргументов ситуация еще более катастрофична: относительное число функций, которые обладают свойством линейной разделимости резко уменьшается. А значит и резко сужается класс функций, который может быть реализован перцептроном.

Это подтолкнуло исследователей к созданию более сложных, и в частности, иерархических (многослойных) архитектур нейронных сетей. Идея относительно проста - на ***низших*** уровнях иерархии классы преобразуются таким образом, чтобы сформировать линейно разделимые множества, которые в свою очередь будут успешно распознаваться нейронами на следующих (***высших***) уровнях иерархии.

На рисунке 5.22 представлен двухслойный перцептрон, полученный из перцептрона с рисунка 5.20 путем добавления второго слоя, состоящего из двух нейронов. Для каждого из слоев существует своя матрица весовых коэффициентов: $\mathbf{W^{(1)}}$ – на первом слое и $\mathbf{W^{(2)}}$ – на втором.

Аналогично строятся и многослойные сети, когда каждый нейрон на последующем слое принимает и обрабатывает сигналы от каждого нейрона более низкого уровня. Таким образом, в данной сети имеется выделенное направление распространения сигналов - от входного слоя через один или несколько скрытых слоев к выходному слою нейронов.

Многослойный перцептрон переводит входной образ, определяющий степени возбуждения нейронов самого первого слоя (нижнего уровня), в выходной образ, определяемый нейронами последнего слоя (верхнего уровня). Число последних, обычно, сравнительно невелико. Состояние возбуждения нейрона на верхнем уровне говорит о принадлежности входного образа к той или иной категории.

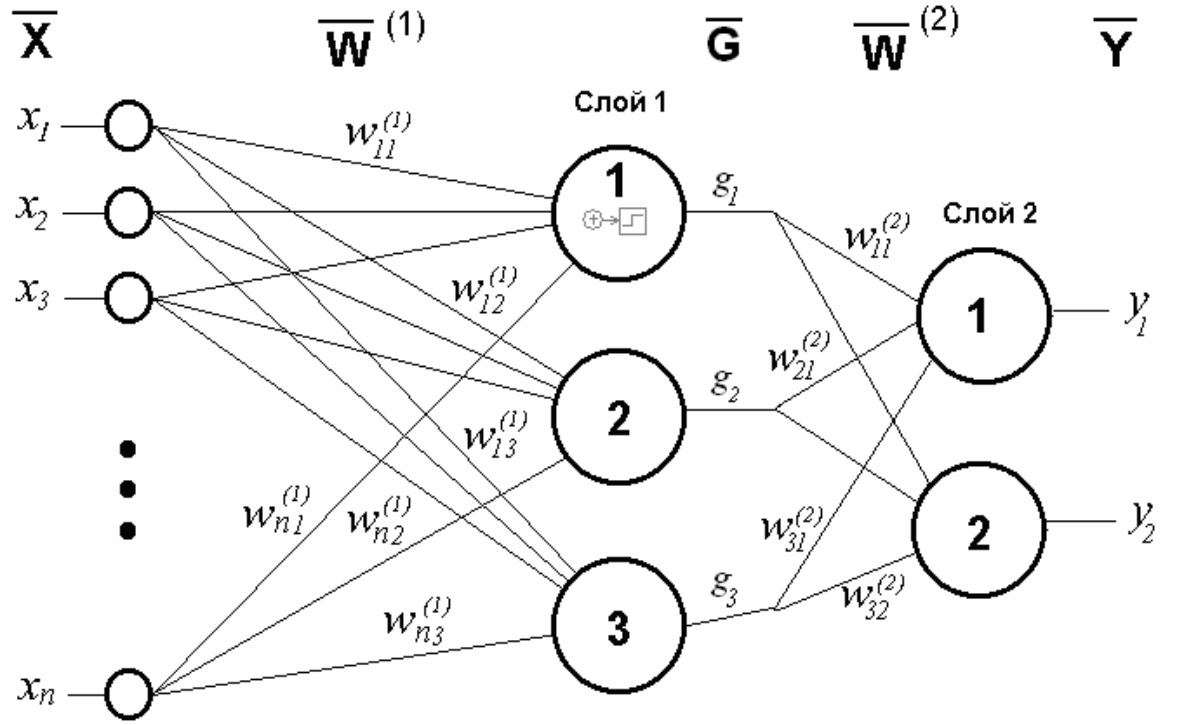


Рис. 5.22. Схема двухслойного перцептрона

Выходные сигналы нейронных слоев легко рассчитываются по следующим формулам:

$$\mathbf{g_k} = \mathbf{f}\left(\sum_{j=1}^{N} \mathbf{w_{kj}^{(1)} x_j}\right), \qquad \mathbf{k = 1, 2, \ldots, L;} \tag{5.11}$$

$$\mathbf{y_i} = \mathbf{f}\left(\sum_{i=1}^{L} \mathbf{w_{ik}^{(2)} g_k}\right) = \mathbf{f}\left(\sum_{i=1}^{L} \mathbf{w_{ik}^{(2)}} \times \mathbf{f}\left(\sum_{j=1}^{N} \mathbf{w_{kj}^{(1)} x_j}\right)\right), \qquad \mathbf{i = 1, 2, \ldots, M,} \tag{5.12}$$

где **N** – число входных сигналов; **L** – число нейронов первого слоя; **M** – число нейронов второго слоя.

Рассмотренный двухслойный перцептрон относится к классу бинарных нейронных сетей, поскольку выход каждого нейрона, формируемый пороговой функцией, может принимать только два значения: логический ноль и логическая единица. Но в ряде практических случаев не менее важно, чтобы выходные значения нейронов были способны принимать непрерывные значения. Это становится возможным при замене пороговой активационной функции на S-образную (5.5) или (5.6). Схема однонаправленной двухслойной нейронной сети с S-образной активационной функцией будет аналогична, рассмотренной выше (рис. 5.22). Однако такая сеть будет уже относиться к классу аналоговых сетей.

Обучение многослойного перцептрона

Для обучения многослойного перцептрона нельзя использовать рассмотренный выше алгоритм, применимый к однослойной структуре. Дело в том, что при обучении «с учителем» необходимо знать не только все текущие выходы нейронов $\mathbf{y_i}$ и $\mathbf{g_i}$, но и требуемые правильные их значения. В случае многослойной сети эти правильные значения имеются только для нейронов выходно-

го слоя, а требуемые значения выходов для нейронов скрытых слоев неизвестны.

Существует несколько вариантов решения этой проблемы.

Первый из них – разработка наборов правильных выходных сигналов, соответствующих входным, для каждого слоя перцептрона. Однако такой путь не всегда осуществим, поскольку является очень трудоемким.

Второй вариант – подстройка весовых коэффициентов случайным образом. Здесь обычно выбираются наиболее слабые связи и изменяются на малую величину в ту или иную сторону. При этом сохраняются только те изменения, которые повлекли уменьшение ошибки на выходе всей сети. Несмотря на простоту, данный метод, основанный на случайном выборе, может потребовать значительного количества вычислений.

Третий вариант использует следующую идею. Ошибки в сигналах нейронов выходного слоя, которые являются известными, возникают вследствие ошибок нейронов скрытых слоев, которые неизвестны. Но чем больше значение связи между нейроном скрытого слоя и выходным нейроном, тем сильнее ошибка первого влияет на ошибку второго. Следовательно, оценку ошибки элементов скрытых слоев можно получить как взвешенную сумму ошибок последующих слоев. При обучении информация распространяется от низших слоев к высшим, а оценки ошибок, делаемые сетью - в обратном направлении. Этот алгоритм обучения получил название процедуры обратного распространения. Рассмотрим его суть применительно к нейронам с S-образной функцией активации.

Для упрощения обозначений ограничимся ситуацией, когда сеть имеет только один скрытый слой (рис.5.22). Матрицу весовых коэффициентов от входов к скрытому слою обозначим $\mathbf{W^{(1)}}$, а матрицу весов, соединяющую скрытый и выходной слои - $\mathbf{W^{(2)}}$. Для индексов примем следующие обозначения: входы будем нумеровать только индексом **i**, элементы скрытого слоя - индексом **j**, а выходы, соответственно, индексом **k**.

Согласно методу наименьших квадратов, минимизируемой целевой функцией ошибки является величина (5.3):

$$\mathbf{E(W_i)} = \frac{\mathbf{1}}{\mathbf{2}} \sum_{\mathbf{p=1}}^{\mathbf{n}} \mathbf{(y_i^p - d_i^p)^2},$$

где $\mathbf{y_i^k}$ – реальное выходное состояние нейрона **i**-го выходного слоя при подаче вход **p**-го образа; $\mathbf{d_i^k}$ – идеальное (желаемое) выходное состояние этого нейрона.

Минимизация ведется методом градиентного спуска. При этом итерационное уточнение элементов матрицы $\mathbf{W^{(2)}}$ можно проводить согласно формуле:

$$\mathbf{w_{jk}^{(2)}(t+1)} = \mathbf{w_{jk}^{(2)}(t)} - \eta \frac{\partial \mathbf{E}}{\partial \mathbf{w_{jk}^{(2)}}}, \qquad (5.13)$$

где η – коэффициент скорости обучения, $\mathbf{0<\eta<1}$.

Наибольшую трудность при работе с зависимостью (5.13) вызывает применение в ней частной производной. Для упрощения ситуации следует использовать полезное свойство S-образных функций, для которых $f'(x)=f(x)[1-f(x)]$. Тогда, используя формулы неявного дифференцирования сложной функции можно записать:

$$\frac{\partial E}{\partial w_{jk}^{(2)}} = (y_k - d_k) \cdot y_k (1 - y_k) \cdot g_j . \qquad (5.14)$$

Для матрицы скрытого слоя формула итерационного уточнения элементов будет аналогична (5.13):

$$w_{ij}^{(1)}(t+1) = w_{ij}^{(1)}(t) - \eta \frac{\partial E}{\partial w_{ij}^{(1)}} . \qquad (5.15)$$

Однако формула для определения значения частной производной несколько усложняется с целью учета взвешенной суммы ошибок последнего слоя:

$$\frac{\partial E}{\partial w_{ij}^{(1)}} = \sum_k [(y_k - d_k) \cdot y_k (1 - y_k) \cdot w_{jk}^{(2)}] \cdot [g_j (1 - g_j) \cdot x_i], \qquad (5.16)$$

где x_i – значение элемента входного вектора обучающей пары.

Таким образом, полный алгоритм обучения двухслойной нейронной сети с помощью процедуры обратного распространения строится так.

1. Установить начальные значения элементов весовых матриц $W^{(1)}$ и $W^{(2)}$ (обычно случайные значения).

2. Подать на входы сети один из входных векторов и вычислить ее выход.

3. Рассчитать по формулам (5.13) и (5.14) скорректированные элементы матрицы выходного слоя $W^{(2)}$.

4. Рассчитать по формулам (5.15) и (5.16)) скорректированные элементы матрицы скрытого слоя $W^{(1)}$.

5. Сравнить выходной вектор сети с эталонным. Если ошибка сети существенна, перейти на шаг 2. В противном случае – конец.

Сети на шаге 2 попеременно в случайном порядке предъявляются все тренировочные образы, чтобы сеть, образно говоря, не забывала одни по мере запоминания других.

Следует отметить, что поскольку в рассматриваемом подходе использовались производные, то пороговая функция и прочие активационные функции с неоднородностями не подходят для рассматриваемого метода.

Рассмотрим еще один, ранее намеренно пропущенный момент, касающийся функций активации. Из графика пороговой функции (рис. 5.15) видно, что пороговое значение **C** в общем случае может принимать произвольное значение. Более того, оно должно принимать некое произвольное, неизвестное заранее значение, которое подбирается на стадии обучения вместе с весовыми коэффициентами. То же самое относится и к центральной точке S-образной за-

висимости, которая может сдвигаться вправо или влево по оси x, а также и ко всем другим активационным функциям. Это, однако, не отражено в формуле (5.1), которая должна была бы выглядеть так:

$$\mathbf{u_i} = \sum_{j=1}^{N} \mathbf{w_{ij}x_j} - \mathbf{C}.$$

Дело в том, что такое смещение обычно вводится путем добавления к нейронам еще одного входа, на который подается дополнительный сигнал, значение которого всегда равняется 1. Если присвоить этому входу номер 0, то можно использовать все ранее приведенные зависимости, начав в них нумерацию входов не с единицы, а с нуля.

Пример 5.6

Рассмотрим использование простейшей нейронной сети для сжатия данных с потерями. Дано прямоугольное изображение, каждый пиксель которого характеризуется своей яркостью. Изображение разбивается на **M** прямоугольных кадров размером **a ´ b** пикселей каждый. Кадр сжимается в вектор данных размерностью **L**, причем **L < a×b**. Сжатую информацию после хранения или передачи по медленным каналам связи можно восстановить в кадр исходного размера **a ´ b**.

Решение данной задачи возможно с использованием простейшего двухслойного перцептрона с линейными функциями активации (рис. 5.23).

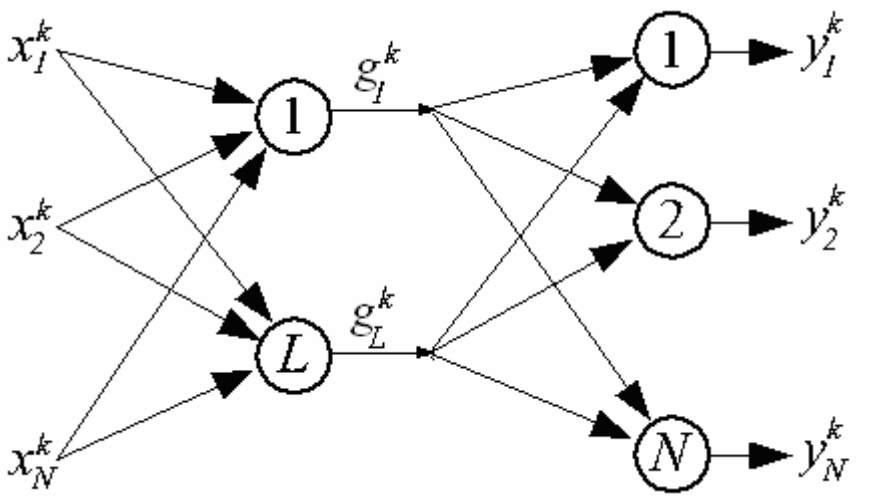


Рис. 5.23. Схема двухслойного перцептрона из примера 5.6

Сжатие (компрессия) данных осуществляется первым слоем нейронов. На вход первого слоя подаются компоненты исходного вектора яркостей каждого пикселя в текущем кадре. То есть, $\mathbf{x_j^k}$ - яркость **j**-го пикселя в **k**-м кадре (**j = 1,2,. . N, N = a ´ b, k = 1,2,…M**).

Результатом сжатия **k**-го кадра является вектор яркости меньшей размерности, то есть $\mathbf{g_j^k}$ - элемент вектора сжатых данных (**j = 1,2,…L; L < a×b).**

Восстановление данных (декомпрессию) производит второй слой нейронов, на вход которого можно подать вектор сжатых данных, а выходом будут

являться восстановленные яркости каждого пикселя $\mathbf{y_j^k}$ **(j = 1,2,. . N, k = 1,2,…M**).

Вследствие линейности функций активации и однонаправленности распространения сигналов имеем:

$$\mathbf{Y^{(k)} = W^{(2)} \times W^{(1)} \times X^{(k)}}.$$

Обучение сети состоит в оптимальном подборе весов, составляющих матрицы $\mathbf{W^{(1)}}_{\times}$ и $\mathbf{W^{(2)}}$, и подразумевает минимизацию целевой функции в виде:

$$\mathbf{E(W)} = \frac{1}{2} \sum_{k=1}^{M} \sum_{i=1}^{N} (y_i^k - x_i^k)^2 .$$

Процесс обучения облегчается тем, что выходной $\mathbf{Y^{(k)}}$ и входной $\mathbf{X^{(k)}}$ векторы сети должны совпадать.

5.2.5. Рекуррентные и самоорганизующиеся нейронные сети

Рекуррентные сети представляют собой развитие однонаправленных перцептронных сетей. В них сигнал с выходных нейронов или нейронов скрытого слоя частично передается обратно на входы нейронов входного слоя. Как любая система, имеющая обратную связь, рекуррентная сеть стремится к устойчивому состоянию. Тем не менее, алгоритмы обучения таких сетей более сложны, чем алгоритмы обучения однонаправленных нейронных сетей.

Однослойная сеть Хопфилда

Простейшей рекуррентной сетью можно считать однослойную сеть Хопфилда. Она состоит из единственного слоя нейронов, число которых является одновременно числом входов и выходов сети. Каждый нейрон имеет один вход, через который осуществляется ввод сигнала. Но выходной сигнал конкретного нейрона поступает на все остальные нейроны сети. Структурная схема сети Хопфилда приведена на рис. 5.24.

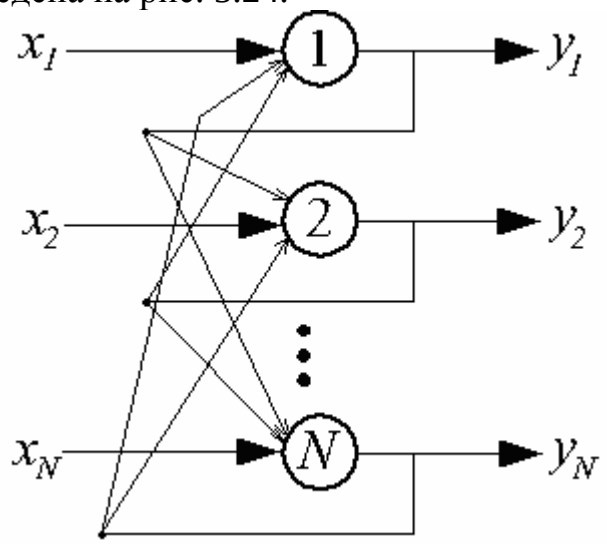


Рис. 5.24. Схема однослойной сети Хопфилда

Активационная функция нейронов сети имеет вид скачка (рис. 5.25), то есть напоминает пороговую и подчиняется закону:

$$y_i = \begin{cases} 1, & \text{если } u_i \geq 0 \\ -1, & \text{если } u_i < 0 \end{cases}$$

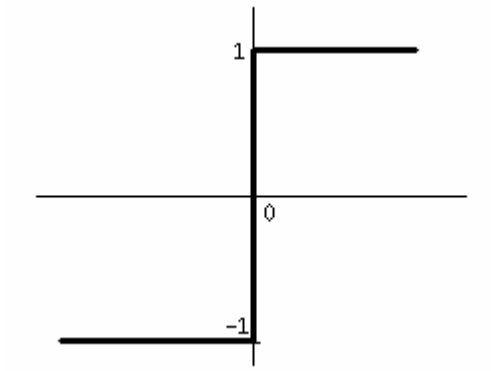


Рис. 5.25. Активационная функция нейронов сети Хопфилда

Поэтому элементы выходного вектора (выходные сигналы) сети могут принимать только два значения {-1,1}. Элементы вектора входных сигналов **[X] = {x_1, x_2,…x_N}** также равны либо +1, либо -1.

Сеть Хопфилда обычно используется для организации ассоциативной памяти, при этом задача, решаемая сетью, формулируется следующим образом.

Известен некоторый набор двоичных сигналов (изображений, звуковых оцифровок, прочих данных, описывающих некие объекты или характеристики процессов), которые считаются образцовыми. При подаче на вход сети произвольного неидеального сигнала, сеть должна:

- Либо выделить(«вспомнить» по частичной информации) соответствующий идеальный сигнал;
- Либо дать «заключение» о том, что входные данные не соответствуют ни одному из образцов.

Обозначим вектор, описывающий **k**-й образец, через **[X^k]**, а его компоненты, соответственно, – x_i^k, (**k = 0...m-1**, где **m** – число образцов). Если сеть его распознает (или «вспомнит») на основе предъявленного входного вектора, ее выходы будут содержать идентичный вектор **[Y]**. В противном случае, выходной вектор не совпадет ни с одним образцовым.

Если, например, информация представляет собой некие изображения, то, отобразив в графическом виде данные с выхода сети, можно будет увидеть картинку, полностью совпадающую с одной из образцовых или же несуществующий образ.

На стадии инициализации сети весовые коэффициенты устанавливаются следующим образом:

$$w_{ij} = \begin{cases} \sum_{k=1}^{m} x_i^k x_j^k, & i \neq j \\ 0, & i = j \end{cases}. \qquad (5.17)$$

Здесь **i** и **j** – индексы, соответственно, нейрона, от которого идет сигнал, и нейрона к которому идет сигнал; x_i^k, x_j^k – **i**-й и **j**-й элементы вектора **k**-ого образца.

Диагональные элементы матрицы весовых коэффициентов равны нулю **($w_{ii} = 0$)**, что исключает эффект воздействия нейрона на самого себя.

Алгоритм функционирования сети следующий:

1. На входы сети подается неизвестный сигнал. Фактически его ввод осуществляется непосредственной установкой значений выходов:

$$y_i(0) = x_i, \; i = 1...N.$$

Ноль в скобке справа от y_i означает нулевую итерацию в цикле работы сети.

2. Рассчитываются новые значения выходов (**t** – номер итерации):

$$y_j(t+1) = f\left[\sum_{i=1}^{N} w_{ij} y_i(t)\right], \; j = 1...N,$$

где **f** – активационная функция в виде скачка, приведенная на рис. 5.25.

3. Если за последнюю итерацию выходные значения изменились, то переход к пункту 2, иначе – конец. При этом выходной вектор представляет собой образец, наилучшим образом сочетающийся с входными данными.

Рассматривая процесс обучения сети Хопфилда, можно сказать, что его нельзя назвать обучением «с учителем» или «без учителя». Здесь весовые коэффициенты рассчитываются только однажды перед началом функционирования сети на основе информации об обрабатываемых данных, и все обучение сети сводится именно к этому расчету. С одной стороны, предъявление исходной информации можно расценивать, как помощь учителя, но с другой – сеть фактически просто запоминает образцы до того, как на ее вход поступают реальные данные, и не может изменять свое поведение. Поэтому говорить об обучении в строгом смысле нельзя.

Сеть Хопфилда имеет ограниченные возможности. Так, установлено, что число запоминаемых образов **m** не должно превышать величины, примерно равной 0,15**N**. Кроме того, если два образа сильно похожи, они, возможно, будут вызывать у сети перекрестные ассоциации, то есть предъявление на входы сети одного вектора приведет к появлению на ее выходах другого, и наоборот.

Самоорганизующиеся сети

Сетями с самоорганизацией называются сети, не требующие для своего обучения «учителя» и самостоятельно адаптирующие свои веса под обучающие данные. Такие сети строятся из нейронов, которые используются группами и конкурируют между собой.

Схема простейшей сети с самоорганизацией на основе конкуренции представлена на рис. 5.26.

Все конкурирующие нейроны в группе получают одни и те же входные сигналы. Каждый нейрон рассчитывает свою взвешенную сумму (сигнал своего сумматора) обычным образом , т.е. по формуле (5.1):

$$u_i = \sum_{j=0}^{N} w_{ij} x_j ,$$

По результатам сравнения всех u_i **(i = 1,2,…M)** выбирается нейрон-победитель, обладающий наибольшим значением u_i .Выходной сигнал y_i нейрона-победителя получает значение «**1**», выходные сигналы всех остальных нейронов – «**0**».

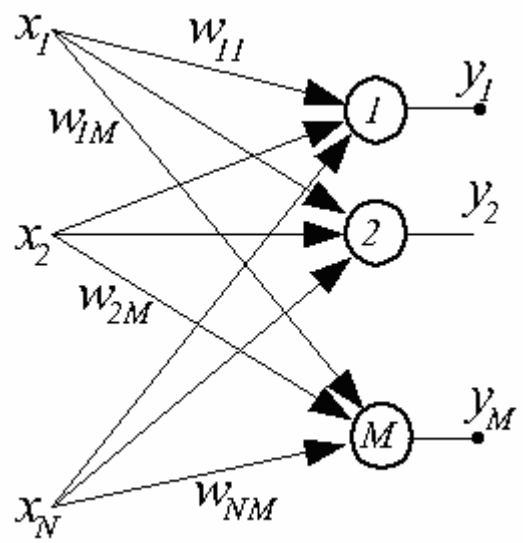


Рис. 5.26. Схема однослойной сети с самоорганизацией на основе конкуренции

Для обучения такой сети не требуется учитель. Начальные значения весовых коэффициентов всех нейронов выбираются случайным образом с последующей нормализацией относительно 1. При предъявлении каждого обучающего вектора $[X^k]$ определяется нейрон-победитель, что дает ему право уточнить свои весовые коэффициенты по упрощенному правилу:

$$w_{ij}(t+1) = w_{ij}(t) - \eta[x_j^k - w_{ij}(t)].$$

Все проигравшие нейроны оставляют свои весовые коэффициенты неизменными.

Дальнейшим развитием самоорганизующихся сетей являются ***сети Кохонена,*** где в процессе функционирования нейроны образуют кластерные группы. В простейшем случае нейроны создают одномерный кластер (цепочку), при этом каждый нейрон имеет, в общем случае, двух ближайших соседей (слева и справа). В более сложном случае нейроны сети Кохонена образуют двумерный кластер – сетку с четырьмя соседями у каждого нейрона (слева, справа, сверху, снизу). В еще более сложном случае сетка гексагональная – у каждого нейрона шесть соседей на плоскости. Кластерные группы соревнуются друг с другом за право наилучшим образом сочетаться с входным вектором сигналов, и побеждает тот нейрон кластера, чей вектор весов ближе всего к входному вектору. Таким образом, каждый входной вектор относится к некоторому кластерному элементу.

В общем случае процесс обучения сети выглядит следующим образом. На вход сети подается обучающий вектор $\mathbf{[X^k]}$, для каждого нейрона определяется расстояние (в смысле выбранной метрики) между векторами $\mathbf{[X^k]}$ и $\mathbf{[W_i]}$. Определяется нейрон-победитель, для которого это расстояние оказывается наименьшим. Вокруг нейрона-победителя образуется окрестность $\mathbf{S_w^k}$ из нейронов-соседей. Веса нейрона-победителя и веса его соседей уточняются. Веса нейронов вне окрестности $\mathbf{S_w^k}$ не изменяются.

В результате обучения соседние нейроны становятся типичными представителями кластеров, соседствующих в многомерном пространстве. В этом достоинство сетей Кохонена – наглядность в представлении многомерных данных путем одномерной или двумерной визуализации.

Когнитрон

Этим термином называют самоорганизующуюся сеть сложной архитектуры, предназначенную для инвариантного распознавания образов. Когнитрон имеет иерархическую многослойную организацию, в которой нейроны между слоями связаны только локально. Своей структурой он определенным образом моделирует строение зрительной коры мозга. Функции распознавания образов в когнитроне реализованы так. Входной слой чувствителен к простым образам (линии, и их ориентация), в то время как реакция других слоев является более сложной, абстрактной и независимой от позиции образа.

Когнитрон состоит из иерархически связанных слоев нейронов разных типов. Среди них следует выделить возбуждающие и тормозящие, первые из которых стремятся вызвать возбуждение последующего нейрона, а вторые – тормозят это возбуждение (рис. 5.27).

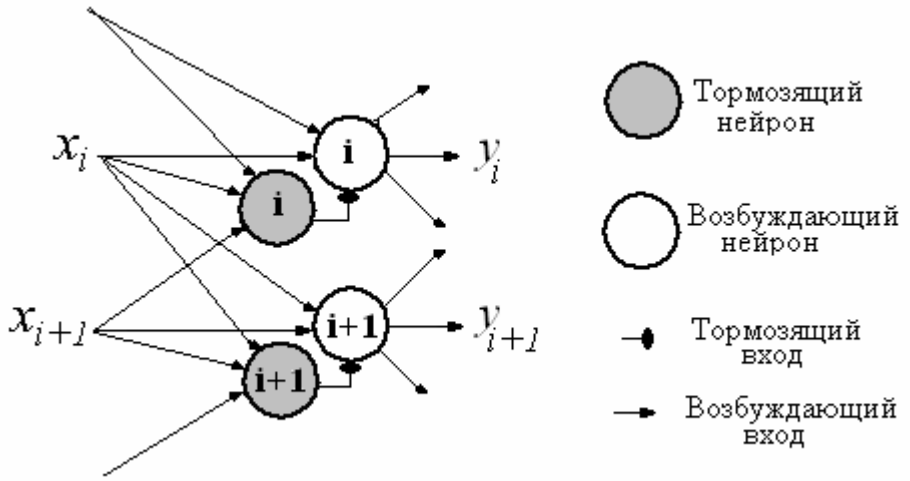


Рис. 5.27. Тормозящие и возбуждающие нейроны когнитрона

Реакция нейрона определяется взвешенной суммой его возбуждающих и тормозящих входов, однако в действительности механизм может быть более сложным, чем простое суммирование.

Каждый нейрон связан не со всеми нейронами предыдущего слоя, а лишь с теми, которые принадлежат их локальной области связей (кластеру). Кла-

стерные области близких друг к другу нейронов перекрываются, поэтому активность каждого нейрона будет сказываться на все более расширяющейся области нейронов следующих слоев иерархии. Так же как и в сети Кохонена, нейроны способны конкурировать между собой (рис. 5.28).

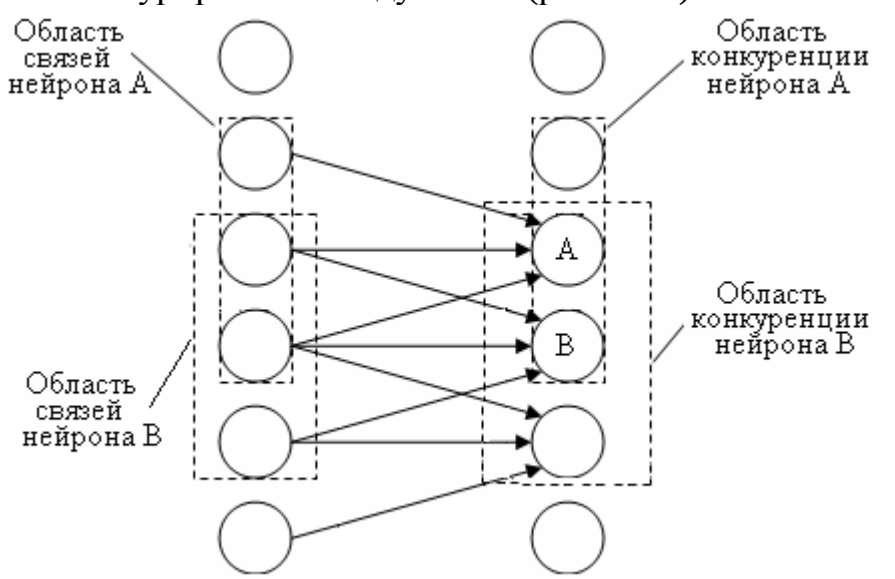


Рис. 5.28. Фрагмент когнитрона с областями связей и конкуренции

Так как когнитрон реализован в виде многослойной сети, он может обучаться конкурентным обучением («без учителя»). Получая обучающий набор входных образов, сеть самоорганизуется посредством изменения весовых коэффициентов (силы связей). При этом отсутствуют предварительно определенные выходные образы, представляющие требуемую реакцию сети, однако сеть настраивается самостоятельно в процессе обучения.

Дальнейшим развитием идеи когнитрона является ***неокогнитрон***, который более точно отражает строение зрительной системы и позволяет распознавать образы независимо от их преобразований, вращений, искажений и изменений масштаба. Неокогнитрон может как самообучаться, так и обучаться с учителем. Неокогнитрон получает на входе двумерные образы, аналогичные изображениям на сетчатой оболочке глаза, и обрабатывает их в последующих слоях аналогично тому, как это делается в зрительной коре мозга человека. Неокогнитрон может использоваться не только для обработки визуальных данных, он достаточно универсален и может найти широкое применение как обобщенная система распознавания образов.

5.2.6. Общие замечания по использованию нейронных сетей

Подробное рассмотрение искусственных нейронных сетей выходит за рамки данной работы. Вопрос о необходимых и достаточных свойствах сети для решения того или иного рода задач представляет собой целое направление нейрокомпьютерной науки. Так как проблема синтеза нейронной сети в значительной степени зависит от решаемой задачи, дать общие подробные рекомен-

дации затруднительно. В большинстве случаев оптимальный вариант получается на основе интуитивного подбора.

В приложении к моделированию сложных систем можно сделать следующие замечания.

Выбор конкретного типа и структуры нейронной сети осуществляется в соответствии с особенностями и сложностью задачи. Для решения некоторых отдельных типов задач уже существуют оптимальные на сегодняшний день конфигурации. Если же задача не может быть сведена ни к одному из известных типов, разработчику приходится решать сложную проблему синтеза новой архитектуры сети. При этом следует руководствоваться такими принципами:

§ возможности сети возрастают с увеличением числа ячеек сети, плотности связей между ними и числом слоев;

§ введение обратных связей, наряду с увеличением возможностей сети, поднимает вопрос о динамической устойчивости сети;

§ сложность алгоритмов функционирования сети (в том числе, например, введение нескольких типов нейронов – возбуждающих, тормозящих и др.) также способствует усилению мощности сети.

5.3. ИНФОРМАЦИОННО-ЭНТРОПИЙНЫЙ ПОДХОД К МОДЕЛИРОВАНИЮ СИСТЕМ

Энтропийные методы моделирования объединяют в себе приемы и основные принципы теоретического описания в форме статистического вывода. В основе такого описания лежит информационный принцип максимального правдоподобия.

Информационно-энтропийный подход можно рассматривать, как совокупность методических рекомендаций для рассмотрения и решения задач в условиях неопределенности, т.е. при некотором недостатке исходной информации. Это означает, что моделирующая объект система уравнений не замкнута, т.е. имеет бесконечное множество решений. Для перехода к замкнутой системе используется вариационный принцип. В качестве критерия правдоподобия принимается положение, известное в литературе как ***формализм Е. Джейнса***. Его можно изложить следующим образом. ***«Из всех возможных состояний системы наиболее вероятным является то, при котором информационная энтропия максимальна»***. Поэтому рассматриваемый подход часто называют методом максимума информационной энтропии.

Рассмотрим интерпретацию понятия информационной энтропии.

Пусть имеется случайная величина **x**, которая может принимать значения $x_1, x_2, \ldots, x_n$ с вероятностями $p_1, p_2, \ldots, p_n$, т. е. для случайной величины **x** существует дискретная функция распределения вероятностей $P(x_i)$ со значениями $p_1, p_2, \ldots, p_n$.

Возникает вопрос, каким образом можно количественно охарактеризовать связь между априорной информацией о случайной величине (например, ее средним значением, дисперсией и т. д.) и видом функции $P(x_i)$. Наши интуи-

тивные представления сводятся к тому, что более размытое распределение вероятностей связано с большей неопределенностью (с меньшей априорной информацией), чем распределение с явно выраженным пиком. Такая мера неопределенности была введена К. Шенноном в виде:

$$S(p_1, p_2, \ldots p_n) = -k \sum_i p_i \ln p_i \,. \qquad (5.18)$$

По определению это есть энтропия вероятностного распределения $P(x_i)$.

Если о случайной величине никакой дополнительной информации нет, то максимизация энтропии S (5.18) при условии $\sum_i p_i = 1$ дает оптимальное распределение $P(x_i) = 1/n$, что совпадает с качественными представлениями о неопределенности.

Допустим, что известно среднее значение некоторой функции $f(x)$. Тогда поиск оптимального распределения сводится к решению следующей задачи:

$$S \to max, \quad \text{при}$$

$$\sum_i p_i f(x_i) = E, \qquad (5.19)$$

$$\sum_i p_i = 1. \qquad (5.20)$$

Ее решение методом неопределенных множителей Лагранжа имеет вид

$$p_i = exp[-\lambda - \mu f(x_i)], \qquad (5.21)$$

где λ и μ - множители Лагранжа, связанные с уравнениями (5.19) и (5.20) соответственно. Этот результат может быть легко обобщен на случай, когда известны средние значения нескольких функций.

Общее правило для формирования моделей при использовании информационно-энтропийного подхода, можно сформулировать следующим образом

1) выделить переменные величины, определяющие интересующую систему, и записать ограничения на эти величины;

2) определить информационную энтропию системы либо с помощью соответствующего распределения вероятностей, либо непосредственно;

3) вычислить значения переменных, при которых энтропия максимальна при принятых ограничениях.

Многие модели, полученные таким способом, можно получить и с помощью более традиционных средств. Но, как правило, процедура максимизации энтропии, по меньшей мере, дает возможность последовательно анализировать сложные ситуации.

Рассмотрим два примера, один из которых связан с исследованием технологического процесса, а другой – с экономическим объектом.

Пример 5.7

Имеется сыпучий материал, состоящий из частиц одинакового размера r_0, общей массой M, подвергающийся измельчению путем механического воздействия. В процессе измельчения к частицам материала подводится энергия, в результате чего частицы дробятся. Естественно, чем большая энергия подведена к материалу, тем меньше будет размер полученных частиц. Возникает задача:

определить, сколько частиц и какого размера будет содержать полученная смесь, если подвести к ней определенное количество энергии.

Решать данную задачу с использованием традиционных детерминированных подходов затруднительно, поскольку процесс измельчения вероятностный и традиционные подходы позволяют лишь определить усредненные показатели процесса (например, средний размер полученных частиц). Рассмотрим применимость информационно-энтропийного подхода.

Допустим, после окончания процесса измельчения в смеси имеются частицы следующих фракций размером $\mathbf{r_0}$, $\mathbf{r_1}$, $\mathbf{r_{32}}$ **. . .** $\mathbf{r_n}$. Обозначим массу каждой полученной фракции $\mathbf{m_0}$, $\mathbf{m_1}$, $\mathbf{m_2}$, **. . .** $\mathbf{m_n}$. Введем соотношение:

$$\frac{\mathbf{m_i}}{\mathbf{M}} = \mathbf{c_i}, \qquad (5.22)$$

где $\mathbf{c_i}$ – доля частиц размером $\mathbf{r_i}$ в полученной смеси.

Нетрудно заметить, что $\mathbf{c_i}$ является случайной величиной.

Из (5.22) вытекает следующее уравнение, характеризующее материальный баланс процесса:

$$\sum_{0}^{n} \mathbf{c_i} = \mathbf{1}. \qquad (5.23)$$

Примем, что для измельчения частиц от размера $\mathbf{r_0}$ до размера $\mathbf{r_i}$ нужно подвести $\mathbf{e_i}$ энергии. Запишем уравнение энергетического баланса:

$$\sum_{0}^{n} \mathbf{e_i m_i} = \mathbf{E}, \qquad (5.24)$$

где **E** – общая энергия, подведенная к материалу.

С учетом (5.22) выражение (5.24) можно переписать:

$$\sum_{0}^{n} \mathbf{e_i c_i} = \frac{\mathbf{E}}{\mathbf{M}}. \qquad (5.25)$$

Выражение для информационной энтропии, как меры неопределенности состояния системы запишем в виде (5.18):

$$\mathbf{S(c_1, c_2, ... c_n)} = -\sum_{i} \mathbf{c_i \ln c_i}. \qquad (5.26)$$

Выше было показано, что наиболее вероятным состоянием рассматриваемой системы будет такое, при котором $\mathbf{c_i}$ будут иметь значения, приводящие к максимуму выражения (5.26). То есть задача сводится к нахождению максимума функции (5.26) с ограничениями (5.23) и (5.25).

Результаты моделирования при разной величине подводимой энергии приведены на рис.5.29.

При моделировании принято, что число получающихся фракций – 5. Подводимая энергия для рис. 5.29а равна $\mathbf{E_1}$, рис. 5.29б - $\mathbf{E_2}$, рис. 5.29в - $\mathbf{E_3}$, рис. 5.29г - $\mathbf{E_4}$, причем $\mathbf{E_1 < E_2 < E_3 < E_4}$.

Из рисунка можно сделать следующие выводы:

1. Видно, что при любом количестве подведенной энергии в материале присутствуют не измельченные частицы с исходным размером $\mathbf{r_0}$.

2. По мере увеличения количества подводимой энергии, доля исходных частиц падает, а доля мелких частиц – возрастает.

Данные выводы полностью совпадают с теоретическими представлениями о природе процесса измельчения.

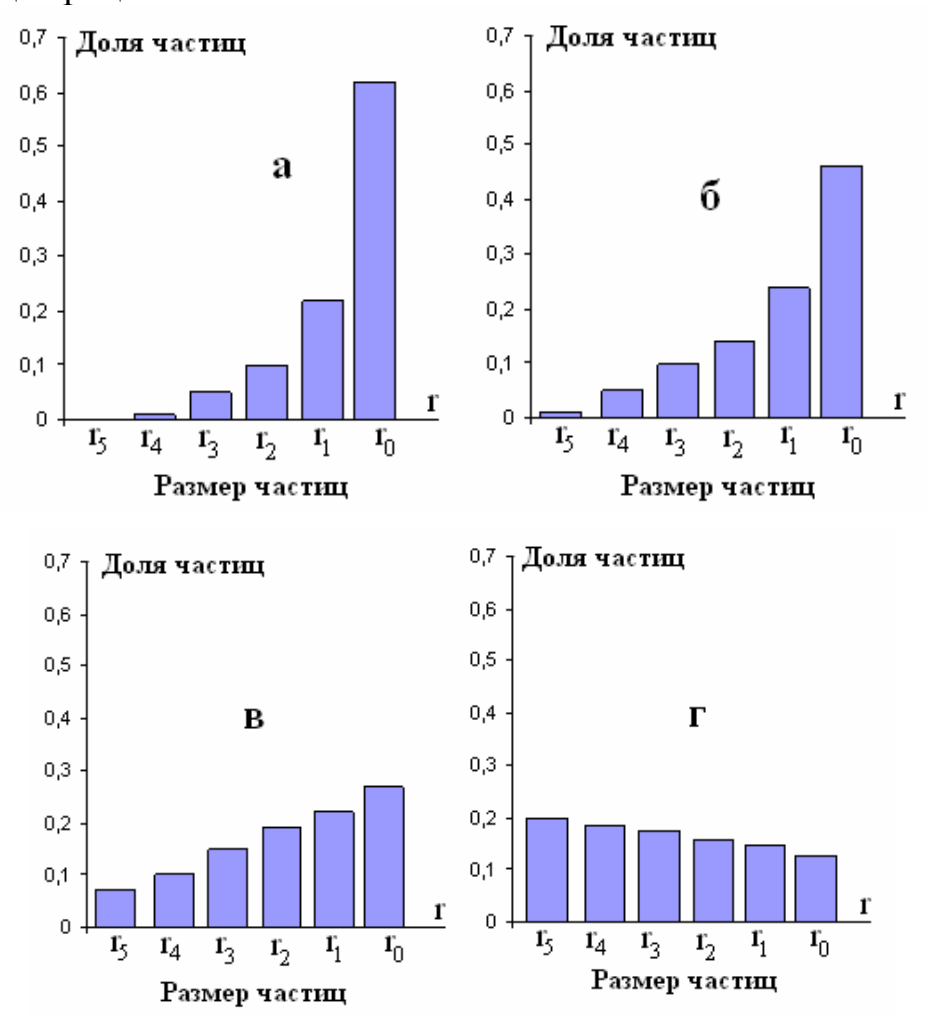


Рис.5.29. Доли частиц отдельных фракций в продукте при разной величине подведенной энергии

Пример 5.8

Для обеспечения производства всегда предполагается создание определенного количества запасов сырья, полуфабрикатов и пр. Необходимость создания запасов определяется дискретностью поставок; случайными колебаниями в длительности интервалов между поставками и т.д. Все это создает тенденцию к увеличению запасов. С другой стороны, существует ряд соображений, согласно которым запасы следует уменьшать. Например, расходы на хранение запаса; ухудшение потребительских свойств при хранении и т.д. Следовательно, возникает проблема оптимального управления количеством запасов на складах предприятия.

Использование информационно-энтропийного подхода для решения данной задачи будет заключаться в следующем.

Обозначим через $\mathbf{z_1, z_2, \ldots, z_n}$ величину избыточных запасов материалов вида 1, 2, . . .,n. Пусть $\mathbf{a_1, a_2, \ldots, a_n}$ – стоимость хранения единицы соответствующего материала, а $\mathbf{d_1, d_2, \ldots, d_n}$ – стоимость доставки для пополнения запаса.

Доля i-го материала в общем количестве избыточных запасов может быть вычислена следующим образом:

$$\mathbf{c_i} = \mathbf{z_i} \Big/ \sum_{j=1}^{n} \mathbf{z_j} . \qquad (5.27)$$

Безусловно, должно выполняться условие:

$$\sum_{i=1}^{n} \mathbf{c_i} = \mathbf{1} . \qquad (5.28)$$

Функция информационной энтропии для данного случая примет вид:

$$\mathbf{S} = -\sum_{i} \mathbf{c_i} \ln \mathbf{c_i} . \qquad (5.29)$$

Для придания определенности задаче нахождения максимума функции (5.29) следует ввести два ограничения. В качестве первого из них можно использовать уравнение (5.28). Второе ограничение можно получить из следующих соображений. Если ввести предельно допустимую величину издержек на доставку и хранение запасов материала **F**, то можно записать:

$$\sum_{i=1}^{n} \mathbf{z_i}(\mathbf{a_i} + \mathbf{d_i}) = \mathbf{F} . \qquad (5.30)$$

Таким образом, задача сводится к нахождению максимума функции (5.29) в условиях ограничений, заданных равенствами (5.28) и (5.30).

Рассмотренную задачу можно решить любым из известных методов поиска экстремума с ограничениями.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Советов Б.Я., Яковлев С.А. Моделирование систем. М.: Высш. шк., 2007.
2. Советов Б.Я., Яковлев С.А. Моделирование систем. Практикум. М.: Высш. шк.,. 2005.
3. Норенков И.П. Основы автоматизированного проектирования. 3 изд., перераб. и доп. МГТУ им. Баумана; 2006.
4. Бенькович Е.С., Колесов Ю.Б., Сениченков Ю. Б. Практическое моделирование динамических систем. СПб.: БХВ-Петербург, 2002.
5. Бусленко Н. П., Моделирование сложных систем. М.: Наука, 1978.
6. Уемов А.И. Системный подход и общая теория систем. М.: Мысль, 1978.
7. Шэннон Р. Имитационное моделирование систем: искусство и наука. М.: Мир, 1978.
8. Котов В.Е. Сети Петри. М.: Наука, 1984.
9. Калан Р. Основные концепции нейронных сетей. М.: Издательский дом «Вильямс», 2001.
10. Вильсон А. Дж. Энтропийные методы моделирования сложных систем. М.: Наука, 1978.