

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

Кафедра автоматической электросвязи

**С.В. Ваняшин**

## **«Методы моделирования и оптимизации»**

**УЧЕБНОЕ ПОСОБИЕ**

по направлению подготовки магистра

11.04.02 «Инфокоммуникационные технологии и системы связи»



Самара  
2017

УДК 621.391

ББК

Р75

Рекомендовано к изданию методическим советом ПГУТИ, протокол № \_\_\_\_, от \_\_\_\_\_ 2017 г.

**Рецензенты:**

Зав. кафедрой систем связи ФГБОУ ВО ПГУТИ, д.т.н., профессор Васин Н.Н.

Директор департамента проектирования ООО «Старт2ком» филиал Самарский Мердеев Э.М.

**Ваняшин, С.В.**

**Р75** **Методы моделирования и оптимизации:** учеб. пособие. Самара: ФГБОУ ВО ПГУТИ, 2017. 83 с.

Учебное пособие содержит материал по дисциплине «Методы моделирования и оптимизации», читаемой для студентов очной полной формы обучения по направлению подготовки магистра 11.04.02 «Инфокоммуникационные технологии и системы связи», в котором рассматриваются целый ряд технологий построения мультисервисных телекоммуникационных сетей, приводятся основные понятия и определения теории моделирования. Содержание курса обеспечивает слушателей необходимым объемом знаний для освоения основ построения и анализа современных мультисервисных телекоммуникационных сетей.

© Ваняшин С.В., 2017

## Содержание

Введение .....	4
Лекция 1. Введение в дисциплину методы моделирования и оптимизации .....	5
Лекция 2. Математические схемы моделирования систем ..	29
Лекция 3. Имитационное моделирование систем .....	46
Лекция 4. Моделирование случайных воздействий.....	59
Лекция 5. Моделирование систем с использованием типовых математических схем и автоматизированных программ.....	67
Список обозначений и сокращений .....	80
Список литературы .....	81
Глоссарий .....	82

## Введение

Телекоммуникационная отрасль является одной из самых быстроразвивающихся отраслей экономики. Ее бурный и динамический рост обуславливает потребность в высокопрофессиональных специалистах, подготовку которых должны обеспечить высшие учебные заведения страны. Такие специалисты должны владеть знаниями современных технологий построения телекоммуникационных сетей и навыками их проектирования.

Настоящий курс лекций является частью учебно-методического комплекса и предназначен для студентов, обучающихся по программе «Методы моделирования и оптимизации» по направлению 210700 «Инфокоммуникационные технологии и системы связи».

Конспект лекций состоит из 5 лекций, предусмотренных учебной программой ПГУТИ. В лекции 1 приводятся основные понятия и определения теории моделирования. Лекция 2 посвящена математическим схемам моделирования. В 3-й лекции обсуждаются вопросы имитационного моделирования систем. Лекция 4 предлагает слушателю обзор анализ моделирования случайных воздействий, а в Лекции 5 приводятся практические вопросы моделирования систем с использованием типовых математических схем и автоматизированных программ имитационного моделирования.

В списке источников даны ссылки на нормативные документы, статьи и монографии, использованные при написании курса лекций.

# **Лекция 1. ВВЕДЕНИЕ В ДИСЦИПЛИНУ МЕТОДЫ МОДЕЛИРОВАНИЯ И ОПТИМИЗАЦИИ**

## **Цель лекции**

В вводной лекции будут рассмотрены основные понятия и определения теории моделирования, включающие в себя модель, систему. Также будет рассмотрена классификация моделей и основные этапы моделирования.

## **1.1. Понятие моделирования, модели и системы**

Моделирование – это замещение одного объекта (оригинала) другим (моделью) и фиксация или изучение свойств оригинала путем исследования свойств модели. Объект (система) определяется совокупностью параметров и характеристик. Множество параметров системы отражает ее внутреннее содержание – структуру и принципы функционирования. Характеристики системы – это ее внешние свойства, которые важны при взаимодействии с другими системами. Характеристики системы находятся в функциональной зависимости от ее параметров.

Теория моделирования представляет собой взаимосвязанную совокупность положений, определений, методов и средств создания и изучения моделей. Эти положения, определения, методы и средства, как и сами модели, являются предметом теории моделирования.

Основная задача теории моделирования заключается в том, чтобы вооружить исследователей технологией создания таких моделей, которые достаточно точно и полно фиксируют интересующие свойства оригиналов, проще или быстрее поддаются исследованию и допускают перенесение его результатов на оригиналы.

Трудно переоценить роль моделирования в научных изысканиях, инженерном творчестве и, вообще, в жизни человека. Разработка и познание любой системы сводится по существу, к созданию ее модели. Особую ценность имеют конструктивные модели, то есть такие, которые допускают не только

фиксацию свойств, но и исследование зависимостей характеристик от параметров системы. Такие модели позволяют оптимизировать функционирование систем.

При изложении теоретических знаний по данной дисциплине предлагаются наиболее известные в литературе методы и средства моделирования. Выбор методов осуществлялся также с точки зрения возможности их применения для исследования вычислительных систем (ВС). В качестве примеров при изложении материала также, как правило, выступают ВС различного назначения и их компоненты.

Модель объекта – это физическая или абстрактная система, адекватно представляющая объект исследования. В теории моделирования используются преимущественно абстрактные модели – описания объекта исследования на некотором языке. Абстрактность модели проявляется в том, что компонентами модели являются не физические элементы, а понятия, в качестве которых наиболее широко используются математические. Абстрактная модель, представленная на языке математических отношений, называется математической моделью. Математическая модель имеет форму функциональной зависимости  $Y = F(X)$ , где  $Y = \{y_1, \dots, y_m\}$  и  $X = \{x_1, \dots, x_n\}$  - соответственно характеристики и параметры моделируемой системы, а  $F$  - функция, воспроизводимая моделью. Построение модели сводится к выявлению функции  $F$  и представлению ее в форме, пригодной для вычисления значений  $Y = F(X)$ . Модель позволяет оценивать характеристики  $Y$  для заданных параметров  $X$  и выбирать значения параметров, обеспечивающие требуемые характеристики, с использованием процедур оптимизации.

Модель может воспроизводить полную совокупность свойств системы либо их подмножество. Состав свойств устанавливается в зависимости от цели исследований и при построении модели определяет:

- состав воспроизводимых характеристик  $Y = \{y_1, \dots, y_m\}$ ;
- состав параметров  $X = \{x_1, \dots, x_n\}$ , изменение которых должно влиять на характеристики  $Y$ ;

- область изменения параметров  $x_j \in x_j^*$ ,  $j = 1, \dots, n$
- область определения модели;
- точность – предельная допустимая погрешность оценки характеристик  $Y$  на основе модели.

Состав характеристик  $Y$  определяется в зависимости от исследуемых свойств системы – производительности, надежности, стоимости и других свойств и должен гарантировать полноту отображения этих свойств. Состав параметров  $X$  должен охватывать все существенные аспекты организации системы, изучение влияния которых на качество функционирования составляет цель исследования, производимого с помощью модели.

Область определения модели характеризует диапазон исследуемых вариантов организации системы. Чем шире состав характеристик и параметров, а также область определения модели, тем универсальнее модель в отношении задач, которые можно решать с ее использованием.

Допустимые погрешности оценки характеристик и точность задания параметров определяют требования к точности модели. Так, если изменения характеристик в пределах 10% не существенны для выбора того или иного варианта системы, то точность определения характеристик должна составлять  $\pm 5\%$ . В большинстве случаев параметры, и в первую очередь параметры рабочей нагрузки, могут быть заданы лишь приближенно, с относительной погрешностью 10-25%. В таких случаях нет смысла предъявлять высокие требования к точности воспроизведения моделью характеристик системы и погрешности их оценки на уровне 5-15% вполне приемлемы.

Модель, удовлетворяющая вышеперечисленным требованиям по составу характеристик и параметров и точности воспроизведения характеристик по всей области определения, называется адекватной системе.

Существенное влияние на адекватность оказывает область определения модели. Практически любая модель обеспечивает высокую точность воспроизведения характеристик в пределах малой окрестности точки  $X = \{x_1, \dots, x_n\}$ . Чем шире область определения модели, тем меньше шансов, что некоторая модель окажется адекватной системе.

Другое свойство модели – сложность. Сложность модели принято характеризовать двумя показателями: размерностью и сложностью вычислений, связанных с определением характеристик. Размерность модели – число величин, представляющих в модели параметры и характеристики. Сложность вычислений, выполняемых при расчете характеристик  $Y = F(X)$ , оценивается числом операций, приходящихся на одну реализацию оператора  $F$ . Обычно сложность вычислений связывается с затратами ресурсов ЭВМ и характеризуется числом процессорных операций и емкостью памяти для хранения информации, относящейся к модели. Сложность вычислений – монотонно возрастающая функция размерности модели. Поэтому более сложной модели присущи одновременно большая размерность и сложность вычислений.

Сложность модели определяется также сложностью моделируемой системы и назначением модели (состав характеристик и параметров, воспроизводимых моделью), размером области определения и точностью модели. Чем сложнее система, то есть чем больше число входящих в нее элементов и процессов, из которых слагается функционирование системы, тем сложнее модель. Увеличение числа воспроизводимых характеристик и параметров, области определения и точности оценки характеристик приводят к увеличению сложности модели.

## **1.2. Краткий обзор методов моделирования**

Модели объектов делятся на два больших класса: материальные (физические) и абстрактные (математические). Среди физических моделей наибольшее распространение получили аналоговые модели. С развитием математики широкое применение получили математические модели. По существу вся математика создана для составления и исследования моделей объектов или процессов.

Создание математической модели преследует две основные цели:



- дать формализованное описание структуры и процесса функционирования системы для однозначности их понимания;
- попытаться представить процесс функционирования в виде, допускающем аналитическое исследование системы.

Единой методики построения математических моделей не существует. Это обусловлено большим разнообразием классов систем:

- статические и динамические;
- структурным или программным управлением;
- постоянной и переменной структурой;
- постоянным (жестким) или сменным (гибким) программным управлением.

По характеру входных воздействий и внутренних состояний системы подразделяются:

- непрерывные и дискретные;
- линейные и нелинейные;
- стационарные и нестационарные;
- детерминированные и стохастические.

Построение модели, отражающей статику системы (состав компонентов и структуру связей) не вызывает больших затруднений. Для динамической системы статику необходимо дополнить описанием работы системы.

### **1.3. Основные этапы моделирования**

Для моделирования необходимо создать модель и провести ее исследование. Некоторые математические модели могут быть исследованы без применения средств вычислительной техники. В настоящее время это практически исключено.

Моделирование на ЭВМ предполагает выполнение следующих этапов:

1. формулирование цели моделирования;
2. разработка концептуальной модели;
3. подготовка исходных данных;
4. разработка математической модели;

5. выбор метода моделирования;
6. выбор средств моделирования;
7. разработка программной модели;
8. проверка адекватности и корректировка модели;
9. планирование экспериментов;
10. моделирование на ЭВМ, анализ результатов моделирования.

#### 1.4. Цель моделирования

Для одной и той же системы  $S_0$  можно построить множество моделей  $\{S_m\}$ .

Эффективность системы должна быть определена до появления новой системы. Если предполагается несколько вариантов системы, то можно выбрать наилучший вариант. Эффективность системы сопоставляется с затратами и достигнутыми характеристиками:

$$E = E(Y_0)$$

$E$  - показатель эффективности;

$Y_0$  - множество характеристик.

Однокритериальная оценка ограничивается оценкой эффективности системы по одному частному показателю качества  $y_{opt}$ ,  $y_{opt} \in Y$ . По остальным характеристикам накладываются ограничения на их допустимые изменения:

$$E = y_{opt}$$

$$y_{i_{\min}} \leq y_i \leq y_{i_{\max}}, \quad i = 1, 2, \dots, n$$

где  $y_{i_{\min}}$ ,  $y_{i_{\max}}$  - нижний и верхний пределы  $i$ -го частного показателя качества;

$n$  - число учитываемых характеристик.

При многокритериальной оценке часто используется нормированный аддитивный критерий:

$$E = \sum_{i=1}^n b_i \gamma(y_i)$$

Функции  $\gamma(y_i)$  подобраны так, чтобы исключить размерность  $i$ -ой характеристики и обеспечить условие  $\gamma(y_i) \in [0,1]$ .

Весовые коэффициенты  $b_i$  удовлетворяют условию

$$\sum_{i=1}^n b_i = 1; \quad b_i > 0.$$

В частном случае, когда  $0 \leq y_i \leq y_{i_{\max}}$ ,  $\gamma(y_i) = y_i / y_{i_{\max}}$ , нормированный аддитивный критерий принимает линейную форму.

## 1.5. Создание концептуальной модели

### Ориентация.

При разработке модели выделяются следующие этапы описания: концептуальный, математический, программный.

Концептуальная (содержательная) модель в словесной форме определяет состав и структуру системы, свойства компонентов и причинно-следственные связи между ними. Здесь приводятся сведения о природе и параметрах элементарных явлений исследуемой системы, о виде и степени взаимодействия между ними, о месте и значении каждого явления в общем процессе функционирования системы.

Процесс создания концептуальной модели никогда не будет формализован, поэтому иногда говорят, что моделирование является не только наукой, но и искусством.

### Стратификация

Стратификация – это выбор уровня детализации модели. Уровни детализации иногда называют стратами. Выбор уровня детализации часто определяется параметрами, допускающими варьирование в процессе моделирования. Такие параметры обеспечивают определение интересующих характеристик. Остальные параметры должны быть, по возможности, исключены из модели.

### Детализация

Здесь вводится понятие элементарной операции, для которой известны и могут быть получены зависимости выходных параметров от входных. Если при этом получается модель большой размерности, то ее можно преобразовать в иерархическую модульную структуру. Каждый модуль такой структуры состоит из совокупности модулей более низких рангов, в том числе и элементарных операций.

### Структуризация. Управление.

Связи между компонентами системы могут быть вещественными и информационными. Вещественные отражают возможные пути перемещения продукта преобразования от одного элемента к другому. Информационные связи обеспечивают передачу между компонентами управляющих воздействий и информации о состоянии. В простых системах информационные связи могут отсутствовать, что соответствует принципу структурного управления.

В более сложных системах управление включает указания, какому компоненту какой исходный объект когда и откуда взять, какую операцию по преобразованию выполнить и куда передать. Такие системы функционируют в соответствии с программным или алгоритмическим принципом управления. В концептуальной модели должны быть конкретизированы все решающие правила или алгоритмы управления рабочей нагрузкой, компонентами и процессами.

### Локализация

На этапе локализации осуществляется представление внешней среды в виде генераторов внешних воздействий, включаемых в состав модели в качестве компонентов. Сюда входят генераторы рабочей нагрузки, генераторы управляющих и возмущающих воздействий. Приемники выходных воздействий обычно не включают в модель.

### Выделение процессов

Функционирование системы заключается в выполнении технологических процессов преобразования вещества, энергии

или информации. В сложных системах, как правило, одновременно протекает несколько процессов. Каждый процесс состоит из определенной последовательности отдельных элементарных операций. Часть операций может выполняться параллельно разными активными компонентами системы. Задается технологический процесс одним из видов представления алгоритмов. В системах с программным управлением, обеспечивающих параллельное выполнение нескольких процессов, имеются алгоритмы управления совокупностью параллельно функционирующих процессов.

## 1.6. Подготовка исходных данных

### Сбор фактических данных

Концептуальная модель определяет совокупность параметров  $S_0$  и внешних воздействий  $X$ . Для количественных параметров необходимо определить их конкретные значения, которые будут использованы в виде исходных данных при моделировании. Сбор данных должен учитывать следующее:

- значения параметров могут быть детерминированными и стохастическими;
- не все параметры являются стационарными;
- речь идет о несуществующей системе (проектируемой, модернизируемой).

Ряд параметров являются случайными величинами по своей природе. Для упрощения модели часть из них представляются детерминированными средними значениями. Это допустимо, если величина имеет небольшой разброс, или, когда цель моделирования достигается при использовании средних значений.

Иногда детерминированные параметры представляются случайной величиной. Например, при многократном выполнении программы количество обрабатываемых данных может изменяться. Всю совокупность вариантов данных можно представить случайной величиной с заданным законом распределения вероятностей.

### Подбор закона распределения

Для случайных параметров выявляется возможность представления их теоретическими законами распределения. Процедура подбора вида закона распределения заключается в следующем.

По совокупности значений параметра строится гистограмма относительных частот – эмпирическая плотность распределения. Гистограмма аппроксимируется плавной кривой, которая сравнивается с кривыми плотности распределения различных теоретических законов распределения. По наилучшему совпадению выбирается один из законов. Далее по эмпирическим значениям вычисляются параметры этого распределения. Проверка совпадения эмпирического и теоретического распределения осуществляется по одному из критериев согласия: Пирсона (хи-квадрат), Колмогорова, Смирнова, Фишера, Стьюдента.

Особую сложность представляет сбор данных по случайным параметрам, зависящим от времени, что характерно для внешних воздействий. Пренебрежение фактами нестационарности параметров существенно влияет на адекватность модели.

### Аппроксимация функций

Для каждого компонента системы существует функциональная связь между параметрами входных воздействий и выходными характеристиками. Иногда вид функциональной зависимости может быть легко выявлен. Однако для некоторых компонентов удается получить лишь экспериментальные данные по входным параметрам и выходным характеристикам.

В этом случае вводится некоторая гипотеза о характере функциональной зависимости, то есть аппроксимация ее некоторым математическим уравнением. Поиск математических зависимостей между двумя или более переменными по собранным данным может выполняться с помощью методов регрессионного, корреляционного или дисперсионного анализа.

Вид математического уравнения задает исследователь. Затем методом регрессионного анализа вычисляются коэффициенты уравнения. Приближение кривой к экспериментальным данным оценивается по критерию наименьших квадратов.

Для выяснения того, насколько точно выбранная зависимость согласуется с опытными данными, используется корреляционный анализ.

При проектировании новой системы отсутствует возможность сбора фактических данных. Для таких параметров выдвигаются гипотезы об их возможных значениях.

Этап сбора и обработки данных заканчивается классификацией на внешние и внутренние, постоянные и переменные, непрерывные и дискретные, линейные и нелинейные, стационарные и нестационарные, детерминированные и стохастические. Для переменных количественных параметров определяются границы их изменений, а для дискретных – возможные значения.

### **1.7. Разработка математической модели**

Функционирование системы заключается в выполнении технологического процесса. При этом в системе одновременно протекает несколько технологических процессов. Технологический процесс представляет собой определенную последовательность отдельных операций. Часть операций может выполняться параллельно разными активными ресурсами системы. Задается технологический процесс одним из способов представления алгоритмов.

В системах с программным принципом управления, обеспечивающих параллельное выполнение нескольких технологических процессов, имеются алгоритмы управления совокупностью процессов. Основное назначение таких алгоритмов заключается в разрешении конфликтных ситуаций, возникающих, когда два или более процесса претендуют на один и тот же ресурс. Совокупность алгоритмов управления  $A_0$  совместно с параметрами входных воздействий  $X_0$  и компонентов  $S_0$  описывают динамику функционирования системы. Обычно алгоритмы управления представляются в виде  $A_m$  удобном для моделирования. Данный подход к описанию динамики работы системы наиболее удобен для имитационного моделирования и

является естественным способом определения характеристик системы:

$$Y = F(X, S, A, T), \quad (1.1)$$

где  $F$  множество операторов определения выходных характеристик,

$T$  – календарное время моделирования.

Для систем со структурным принципом управления получил распространение другой подход к описанию динамики работы системы. Для каждого компонента выбирается определенный параметр  $S$  или несколько параметров, значения которых изменяются в ходе функционирования компонента и отражает его состояние в текущий момент времени  $z(t)$ . Множество таких параметров по всем  $n = 1, N$  элементам системы  $\{z_n\}$  отражает состояние системы  $Z(t)$ . Функционирование системы представляется в виде последовательной смены состояний:  $Z(t_0), Z(t_1), \dots, Z(T)$ . Множество  $\{Z\}$  возможных состояний системы называют пространством состояний. Текущее состояние системы в момент времени  $t (t_0 < t \leq T)$  отражается в виде координаты точки в  $n$  – мерном пространстве состояний, а вся реализация процесса функционирования системы за время  $T$  – в виде некоторой траектории.

При заданном начальном состоянии системы  $Z^0 = Z(t_0)$  можно определить ее состояние в любой момент  $t$  из интервала  $[t_0, T]$ , если известна зависимость

$$Z(t) = H(X, Z^0, Z, t). \quad (1.2)$$

В этом случае выходные характеристики системы определяются по выражению

$$Y = G(Z, T). \quad (1.3)$$

В выражениях (1.1), (1.3) операторы  $F$  и  $G$  называют операторами выходов, а в (1.2) оператор  $H$  – оператором переходов. С целью перевода обобщенной модели в форме (1.1) или в форме (1.1), (1.3) в конструктивную необходимо конкретизировать свойства множеств переменных и операторов. Для опре-



деленных классов систем разработаны формализованные схемы и математические методы, которые позволяют описать функционирование системы, а в некоторых случаях – выполнить аналитические исследования.

Одной из наиболее общих формализованных схем является описание в виде агрегативных систем. Этот метод в наибольшей мере приспособлен для описания систем, у которых характерно представление входных и выходных воздействий в виде ”сообщений”, составленных из совокупностей “сигналов”. В основе метода лежит понятие агрегата как компонента системы. Математическая модель агрегата выражается в виде зависимостей (1.2), (1.3) с конкретизацией входных воздействий, состояний и операторов переходов и выходов. Единообразное математическое описание исследуемых объектов в виде агрегативных систем позволяет использовать универсальные средства имитационного моделирования.

Для описания стохастических систем с дискретными множествами состояний, входных и выходных воздействий, функционирующих в непрерывном времени, широко используются стохастические сети. Стохастическая сеть представляет собой совокупность систем массового обслуживания, в которой циркулируют заявки, переходящие из одной системы в другую.

Если в модели системы не учитывается воздействие случайных факторов, а операторы переходов и выходов непрерывны, то зависимости (1.2), (1.3) могут быть представлены в виде дифференциальных уравнений

$$\frac{dz}{dt} = h(z(t), x(t), t),$$
$$y = g(z(t), t),$$

где  $h, g$  – вектор–функции состояний ее выходов соответственно;

$x, z, y$  – векторы входных воздействий, состояний и выходных воздействий соответственно.

Системы состояния которых определены в дискретные моменты времени  $t_0, t_1, t_2, \dots$ , получили название автоматов. Если за единицу времени выбран такт  $\Delta t = t_i - t_{i-1}$ , то дискретные мо-

менты времени принимаются в виде  $0, 1, 2, \dots$ . В каждый дискретный момент времени, за исключением  $t_0$ , в автомат поступает входной сигнал  $x(t)$ , под действием которого автомат переходит в новое состояние в соответствии с функцией переходов

$$z(t) = \varphi_a(z(t-1), x(t))$$

и выдает выходной сигнал, определяемый функцией выходов

$$y(t) = \psi_a(z(t-1), x(t)).$$

Если автомат характеризуется конечными множествами состояний  $z$ , входных сигналов  $x$ , и выходных сигналов  $y$ , он называется конечным автоматом. Функции переходов и выходов конечного автомата задаются таблицами, матрицами или графиками.

Стохастические системы, функционирующие в дискретном времени, можно представлять вероятностными автоматами. Функция переходов вероятностного автомата определяет не одно конкретное состояние, а распределение вероятностей на множестве состояний, а функция выходов – распределение вероятностей на множестве выходных сигналов. Функционирование вероятностных автоматов изучается при помощи аппарата цепей Маркова. Для анализа систем, представляемых в виде автоматов, могут использоваться аналитические или имитационные методы.

В классе автоматов для исследования систем, функционирование которых описывается совокупностью параллельных взаимодействующих процессов, широко используются сети Петри и их различные модификации, сети Керка,  $E$  – сети.

## 1.8. Выбор метода моделирования

Математическая модель может быть исследована различными методами – аналитическими или имитационными. В некоторых случаях наличие имитационной модели делает возможным применение математических методов оптимизации. Для использования аналитических методов необходимо матема-

тическую модель преобразовать к виду явных аналитических зависимостей между характеристиками и параметрами системы и внешних воздействий. Однако это удастся лишь для сравнительно простых систем и при наличии хорошо разработанной теории исследуемых объектов.

При имитационном моделировании динамические процессы системы – оригинала подменяются процессами, имитируемыми в абстрактной модели, но с соблюдением таких же соотношением длительностей и временных последовательностей отдельных операций. Поэтому метод имитационного моделирования мог бы называться алгоритмическим или операционным.

Методы имитационного моделирования различаются в зависимости от класса исследуемых систем, способа продвижения модельного времени и вида количественных переменных параметров системы и внешних воздействий. Методы моделирования разрабатываются для дискретных, непрерывных и смешанных дискретно – непрерывных систем.

В зависимости от способа продвижения модельного времени методы моделирования подразделяются на методы с приращением временного интервала и методы с продвижением времени до особых состояний. В первом случае модельное время продвигается на некоторую величину  $\Delta t$ . Далее определяют изменения состояний элементов и выходных воздействий системы, которые произошли за это время. После этого модельное время снова продвигается на величину  $\Delta t$ , и процедура повторяется. Так продолжается до конца периода моделирования  $T_m$ . Этот метод моделирования называют “принципом  $\Delta t$ ”.

Во втором случае в текущий момент модельного времени  $t$  сначала анализируются те будущие особые состояния – поступление дискретного входного воздействия, завершение обслуживания и т.п., для которых определены моменты их наступления  $t_i > t$ . Выбирается наиболее раннее особое состояние, и модельное время продвигается до момента наступления этого состояния. Затем анализируется реакция системы на выбранное особое состояние. В частности, в ходе анализа определяется момент наступления нового особого случая. Затем анализируются будущие особые состояния, и модельное время про-

двигается до ближайшего. Так продолжается до завершения периода моделирования  $T_m$ . Данный метод называют “принципом особых состояний”. Метод может использоваться лишь, когда имеется возможность определения моментов наступления будущих очередных особых состояний.

Параметры системы и внешних воздействий могут быть детерминированными или случайными. По этому признаку различают детерминированное и статистическое моделирование. При статистическом моделировании для получения достоверных вероятностных характеристик процессов функционирования системы требуется их многократное воспроизведение с различными конкретными значениями случайных факторов и статистической обработкой результатов измерений. В основу статистического моделирования положен метод статистических испытаний, или метод Монте–Карло.

При разработке математической модели различают также принятую систему формализации – алгоритмический (программный) или структурный (агрегатный) подход. В первом случае процессы управляют компонентами (ресурсами) системы, а во втором – компоненты управляют процессами, определяют порядок функционирования системы.

## **1.9. Выбор средств моделирования**

### *Технические средства моделирования*

Современные ЭВМ позволяют моделировать сложные распределенные динамические системы. Фактор распределенности играет важную роль и предполагает построение многопроцессорных вычислительных систем на основе локальных вычислительных сетей. Поэтому для моделирования таких систем перспективным представляется использование распределенных многопроцессорных вычислительных систем.

### *Алгоритмические языки общего назначения*

Такие языки применимы для аналитических методов моделирования. При использовании алгоритмических языков для программирования имитационных алгоритмов возникают труд-

ности. Первая из них заключается в том, что алгоритмы поведения сложных систем являются параллельными, то есть предполагают выполнение более чем одного преобразования в каждый момент времени. Программная имитация параллельных процессов при использовании языков общего назначения сводится к организации псевдопараллельного развития параллельных процессов, что достаточно сложно для программирования.

Вторая трудность состоит в том, что объемы данных в имитационных алгоритмах трудно оценить априорно. Поэтому требуется использовать динамическое распределение памяти, что не всегда обеспечивает язык программирования.

### Языки моделирования

При создании программ имитационного моделирования возникают задачи, общие для широкого класса моделей:

- организация псевдопараллельного выполнения алгоритмов;
- динамическое распределение памяти;
- операции с модельным временем, отражающим астрономическое время функционирования оригинала;
- имитация случайных процессов;
- ведение массива событий;
- сбор и обработка результатов моделирования.

Решение перечисленных выше задач осуществляется полностью или частично внутренними средствами языка моделирования.

По структуре и правилам программирования языки моделирования подобны процедурно-ориентированным алгоритмическим языкам высокого уровня. Операторы языков моделирования выполняют более сложные процедуры, поэтому они имеют более высокий уровень и упрощают составление программ. Языки моделирования рассматриваются как формализованный базис для создания математических моделей.

### Автоматизированные системы моделирования

Дальнейшее упрощение и ускорение процесса программирования привело к необходимости его автоматизации. К настоящему времени создан ряд систем автоматизации имита-

ционного моделирования, среди которых можно отметить OPNET Modeler, NS-3, Simulink и др.

### 1.10. Проверка адекватности и корректировка модели

#### Проверка адекватности

Адекватность модели нарушается по многим причинам: из-за идеализации внешних условий и режимов функционирования; исключения тех или иных параметров; пренебрежения некоторыми случайными факторами. Отсутствие точных сведений о внешних воздействиях, определенных нюансах структуры системы, принятые аппроксимации, интерполяции, предположения и гипотезы также ведут к уменьшению соответствия между моделью и системой. Это приводит к тому, что результаты моделирования будут существенно отличаться от реальных.

Простейшей мерой адекватности может служить отклонение некоторой характеристики  $y_0$  оригинала и  $y_m$  модели,

$$\Delta y = |y_0 - y_m| \quad \text{или} \quad \Delta y = |y_0 - y_m| / y_0$$

Тогда можно считать, что модель адекватна с системой, если вероятность того, что отклонение  $\Delta y$  не превышает предельной величины  $\Delta$ , больше допустимой вероятности  $P_\Delta$ :

$$P(\Delta y < \Delta) \geq P_\Delta$$

Практическое использование данного критерия адекватности зачастую невозможно по следующим причинам:

- для проектируемых или модернизируемых систем отсутствует информация о значении характеристики  $y_0$ ;
- система оценивается не по одной, а по множеству характеристик, у которых может быть разная величина отклонения;
- характеристики могут быть случайными величинами и функциями, а часто и нестационарными;
- отсутствует возможность априорного точного задания предельных отклонений  $\Delta$  и допустимых вероятностей  $P_\Delta$ .

Несмотря на это проверять адекватность необходимо иначе по неверным результатам моделирования будут приняты неправильные решения. На практике оценка адекватности обычно проводится путем экспертного анализа разумности результатов моделирования. Можно выделить следующие виды проверок:

- проверка моделей компонентов;
- проверка модели внешних воздействий (аппроксимации и гипотезы необходимо оценить математическими методами);
- проверка концептуальной модели функционирования системы;
- проверка формализованной и математической модели;
- проверка способов измерения и вычисления выходных характеристик;
- проверка программной модели.

#### Корректировка модели

Если по результатам проверки адекватности выявляется недопустимое рассогласование модели и системы, возникает необходимость в корректировке или калибровке модели. При этом выделяются следующие типы изменений: глобальные, локальные и параметрические.

Стратегия корректировки модели должна быть направлена на первоочередное введение глобальных изменений, затем – локальных и параметрических изменений. Для выработки тактики параметрических изменений большое значение имеет анализ чувствительности модели к вариациям ее параметров.

Завершается этап проверки адекватности и корректировки модели определением и фиксацией области пригодности модели. Под областью пригодности понимается множество условий, при соблюдении которых точность результатов моделирования находится в допустимых пределах.

## 1.11. Планирование экспериментов с моделью

### Стратегическое планирование

Цели моделирования достигаются путем исследования разработанной модели. Исследования заключаются в проведении экспериментов по определению выходных характеристик системы при разных значениях управляемых переменных параметров модели. Эксперименты следует проводить по определенному плану.

Особо важным является планирование экспериментов при численном и статистическом моделировании. Это обусловлено большим числом возможных сочетаний значений управляемых параметров, а каждый эксперимент проводится при определенном сочетании значений параметров. Например, при пяти управляемых параметрах, имеющих по три значения, количество сочетаний параметров равно 243, при десяти параметрах по пять значений, число сочетаний приближается к 10 млн. Поэтому возникает необходимость в выборе определенных сочетаний параметров и последовательности проведения экспериментов. Это называется стратегическим планированием.

Разработка плана начинается на ранних этапах создания модели, когда выявляются характеристики качества и параметры, с помощью которых предполагается управлять качеством функционирования системы. Эти параметры называют факторами. Затем выделяются возможные значения количественных параметров и варианты качественных (функциональных) параметров. Их называют уровнями  $q$ . При этом число сочетаний

$$N_s = q_1 \times q_2 \times \dots \times q_k,$$

где  $k$  - число факторов.

Если число факторов велико, то для проведения исследований системы используется один из методов составления плана по неполному факторному анализу. Эти методы хорошо разработаны в теории планирования экспериментов.

### Тактическое планирование

Совокупность методов уменьшения длительности машинного эксперимента при обеспечении статистической досто-



верности результатов имитационного моделирования получила название тактического планирования. На длительность одного эксперимента (периода моделирования  $T_m$ ) влияет степень стационарности системы, взаимозависимости характеристик и значения начальных условий моделирования.

Данные, собранные в эксперименте, можно рассматривать как временные ряды, состоящие из замеров определенных характеристик. Ряд замеров характеристики  $y$  может рассматриваться как выборка из стохастической последовательности. Если эта последовательность стационарна, ее среднее  $\bar{y}$  не зависит от времени. Оценкой  $\bar{y}_N$  является среднее по временному ряду  $y_1, \dots, y_N$ . Для эргодической последовательности точность этой оценки возрастает с ростом  $N$ .

Если заданы максимальная допустимая ошибка оценки (доверительный интервал) и минимальная вероятность того, что истинное среднее  $\bar{y}$  лежит внутри этого интервала, то существует минимальный размер исследуемой выборки. Этот размер соответствует минимальной длительности эксперимента. Для оценки нескольких характеристик период моделирования определяется по максимальному значению.

## **1.12. Моделирование на ЭВМ и анализ результатов моделирования**

### Обработка измерений имитационного моделирования

При статистическом моделировании в ходе имитационного эксперимента измеряются множества значений по каждой выходной характеристике. Эти выборки необходимо обрабатывать для удобства последующего анализа и использования. Поскольку выходные характеристики зачастую являются случайными величинами или функциями, обработка заключается в вычислении оценок математических ожиданий, дисперсий и корреляционных моментов.

Для того, чтобы исключить необходимость хранения в машине всех измерений, обработку проводят по рекуррентным формулам, когда оценки вычисляют в процессе эксперимента

методом нарастающего итога по мере появления новых измерений.

Для стохастических характеристик можно построить гистограмму относительных частот – эмпирическую плотность распределения. С этой целью область предполагаемых значений характеристики  $y$  разбиваются на интервалы. В ходе эксперимента по мере измерений определяют число попаданий характеристики в каждый интервал и подсчитывают общее число измерений. После завершения эксперимента для каждого интервала вычисляют отношение числа попаданий характеристики к общему числу измерений и длине интервала.

Для случайных нестационарных характеристик период моделирования  $T_m$  разбивается на отрезки  $\Delta t$  и запоминаются значения характеристики в конце каждого  $\Delta t$ . Проводится серия экспериментов с разными последовательностями случайных параметров модели. Измерения по каждому  $\Delta t$  обрабатываются как при оценке случайных величин.

#### Определение зависимостей характеристик от параметров системы

Для анализа зависимостей характеристик от параметров системы и внешних воздействий можно воспользоваться корреляционным, дисперсионным или регрессионным методами.

С помощью корреляционного анализа можно установить наличие связи между двумя или более случайными величинами. Оценкой связи служит коэффициент корреляции при наличии линейной связи между величинами и нормальном законе их совместного распределения.

Дисперсионный анализ можно использовать для установления относительного влияния различных факторов на значения выходных характеристик. При этом общая дисперсия характеристики разлагается на компоненты, соответствующие рассматриваемым факторам. По значениям отдельных компонентов делают вывод о степени влияния того или другого фактора на анализируемую характеристику.

Когда все факторы в эксперименте являются количественными, можно найти аналитическую зависимость между

характеристиками и факторами. Для этого используются методы регрессионного анализа.

К анализу результатов моделирования можно отнести задачу анализа чувствительности модели к вариациям ее параметров. Под анализом чувствительности понимают проверку устойчивости характеристик процесса функционирования системы к возможным отклонениям значений параметров.

### Использование результатов моделирования

Результаты моделирования используются для принятия решения о работоспособности системы, для выбора лучшего проектного варианта или для оптимизации системы. Решение о работоспособности принимается по тому, выходят или не выходят характеристики системы за установленные границы при любых допустимых изменениях параметров. При выборе лучшего варианта из всех работоспособных вариантов выбирается тот, у которого максимальное значение критерия эффективности. Наиболее общей и сложной является оптимизация системы: требуется найти такие сочетания значений переменных параметров системы или рабочей нагрузки из множества допустимых, которое максимизирует значение критерия эффективности:

$$E_{opt} = \max(\min) E(y)$$

при соблюдении ограничений на все  $n$  характеристик

$$y_{i_{\min}} \leq y_i \leq y_{i_{\max}}, \quad i = 1, 2, \dots, n$$

Если выходная характеристика  $y_i$  является случайной величиной с некоторой плотностью распределения  $f(y_i)$ , целесообразно ввести в задачу оптимизации стохастические ограничения следующего вида:

$$P(y_{i_{\min}} \leq y_i \leq y_{i_{\max}}) = \int_{y_{i_{\min}}}^{y_{i_{\max}}} f(y_i) dy_i \geq \beta_i,$$

где  $\beta_i$  - минимально допустимая вероятность того, что конкретные значения  $y_i$  не выйдут за ограничивающие пределы.

### **Выводы по лекции 1:**

Моделированием называется замещение одного объекта, называемого системой, другим объектом, называемым моделью. Для описания системы необходимо определить ее структурную и функциональную организацию. Область определения модели характеризует диапазон исследуемых вариантов организации системы. Чем шире состав характеристик и параметров, а также область определения модели, тем универсальнее модель в отношении задач, которые можно решать с ее использованием. Модели объектов делятся на два больших класса: материальные (физические) и абстрактные (математические).

### **Вопросы для самопроверки по лекции 1:**

- Назовите цели моделирования.
- Дайте определение понятий система и модель.
- От каких факторов зависит адекватность модели?
- Назовите основные признаки, необходимые для классификации моделей.
- Чем определяется сложность модели?
- Назовите и охарактеризуйте основные этапы моделирования.
- Перечислите примеры систем автоматизации имитационного моделирования.

## **Лекция 2. МАТЕМАТИЧЕСКИЕ СХЕМЫ МОДЕЛИРОВАНИЯ СИСТЕМ**

### **Цель лекции**

В данной лекции будут рассмотрены основные подходы к построению математических моделей систем, а также математические схемы моделирования систем.

### **2.1. Основные подходы к построению математических моделей систем**

Исходной информацией при построении математических моделей (ММ) процессов функционирования систем служат данные о назначении и условиях работы исследуемой (проектируемой) системы  $S$ . Эта информация определяет основную цель моделирования, требования к ММ, 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 её функционирования в виде некоторой ММ.

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

Например, представление процесса функционирования информационно-вычислительной сети 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 даёт 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. Т.е. имеет место цепочка: описательная модель — математическая схема — имитационная модель.

Каждая конкретная система  $S$  характеризуется набором свойств, под которыми понимаются величины, отображающие поведение моделируемого объекта (реальной системы) и учитываются условия её функционирования во взаимодействии с внешней средой (системой)  $E$ .

При построении ММ системы  $S$  необходимо решить вопрос о её полноте. Полнота моделирования регулируется, в основном, выбором границ "Система  $S$  — среда  $E$ ". Также должна быть решена задача упрощения ММ, которая помогает выделить основные свойства системы, отбросив второстепенные в плане цели моделирования.

ММ объекта моделирования, т.е. системы  $S$  можно представить в виде множества величин, описывающих процесс функционирования реальной системы и образующих в общем случае следующие подмножества:

- совокупность  $X$  – входных воздействий на  $S$   $x_i \in X$ ,  $i=1 \dots n_x$ ;
- совокупность воздействий внешней среды  $v_l \in V$ ,  $l=1 \dots n_v$ ;
- совокупность внутренних (собственных) параметров системы  $h_k \in H$ ,  $k=1 \dots n_h$ ;
- совокупность выходных характеристик системы  $y_j \in Y$ ,  $j=1 \dots n_y$ .

В перечисленных множествах можно выделить управляемые и неуправляемые величины. В общем случае  $X$ ,  $V$ ,  $H$ ,  $Y$  не пересекаемые множества, содержат как детерминированные так и стохастические составляющие. Входные воздействия  $E$  и внутренние параметры  $S$  являются **независимыми (экзогенными) переменными**,  $\bar{X}(t); \bar{V}(t); \bar{H}(t)$ . Выходные характеристики – **зависимые переменные (эндогенные)**  $\bar{Y}(t)$ . Процесс функционирования  $S$  описывается оператором  $F_S$ :

$$\bar{Y}(t) = F_S(\bar{X}, \bar{V}, \bar{H}, t) \quad (2.1)$$

$\bar{Y}(t)$  - выходная траектория.  $F_S$  – закон функционирования  $S$ .  $F_S$  может быть функция, функционал, логические условия, алгоритм, таблица или словесное описание правил.

Алгоритм функционирования  $A_S$  – метод получения выходных характеристик  $\bar{Y}(t)$  с учётом входных воздействий  $\bar{X}(t); \bar{V}(t); \bar{H}(t)$ . Очевидно один и тот же  $F_S$  может быть реализован различными способами, т.е. с помощью множества различных  $A_S$ .

Соотношение (2.1) является математическим описанием поведения объекта  $S$  моделирования во времени  $t$ , т.е. отражает его динамические свойства. (2.1) – это динамическая модель системы  $S$ . Для статических условий ММ есть отображения  $X, V, H$  в  $Y$ , т.е.  $\bar{Y} = f(\bar{X}, \bar{V}, \bar{H})$  (2.2)

Соотношения (2.1), (2.2) могут быть заданы формулами, таблицами и т.д.

Также соотношения в ряде случаев могут быть получены через свойства системы в конкретные моменты времени, называемые состояниями.

Состояния системы  $S$  характеризуются векторами:

$$\bar{Z}^I (\bar{Z}_1^I, \dots, \bar{Z}_k^I) \text{ и } \bar{Z}^{II} (\bar{Z}_1^{II}, \dots, \bar{Z}_k^{II}),$$

где  $\bar{Z}_1^I = Z_1(t^I) \dots Z_k^I = Z_k(t^I)$  в момент  $t^I \in (t_0, T)$ .

$$\bar{Z}_1^{II} = Z_1(t^{II}) \dots Z_k^{II} = Z_k(t^{II}) \text{ в момент } t^{II} \in (t_0, T) \text{ и т.д. } k=1 \dots n_Z.$$

$Z_1(t), Z_2(t) \dots Z_k(t)$  – это координаты точки в  $k$ -мерном фазовом пространстве. Каждой реализации процесса будет соответствовать некоторая фазовая траектория.

Совокупность всех возможных значений состояний  $\{ \bar{Z} \}$  называется пространством состояний объекта моделирования  $Z$ , причём  $z_k \in Z$ .

Состояние системы  $S$  в интервале времени  $t_0 < t \leq T^I$  полностью определяется начальными условиями  $\bar{Z}^0 (Z_1^0, \dots, Z_k^0)$ , где  $Z_1^0 = Z_1(t_0) \dots$  входными  $\bar{X}(t)$ , внутренними параметрами  $\bar{H}(t)$  и

воздействиями внешней среды  $\vec{V}(t)$ , которые имели место за промежутки времени  $t^* - t_0$  с помощью 2-х векторных уравнений:

$$\vec{Z}(t) = \Phi(\vec{z}^0, \vec{X}, \vec{V}, \vec{h}, t); \quad (2.3)$$

$$\vec{Y}(t) = F(\vec{Z}, t). \quad (2.4)$$

$$\text{иначе: } Y(t) = F(\Phi(\vec{z}^0, \vec{X}, \vec{V}, \vec{h}, t)). \quad (2.5)$$

Время в модели  $S$  может рассматриваться на интервале моделирования  $(t_0, T)$  как непрерывное, так и дискретное, т.е. квантованное на отрезке длин.  $\Delta t$ .

Таким образом под ММ объекта понимаем конечное множество переменных  $\{\vec{X}, \vec{Z}, \vec{h}\}$  вместе с математическими связями между ними и характеристиками  $\vec{Y}$ .

Моделирование называется детерминированным, если операторы  $F, \Phi$  детерминированные, т.е. для конкретного входа выход детерминированный. Детерминированное моделирование - частный случай стохастического моделирования. В практике моделирование объектов в области системного анализа на первичных этапах исследования рациональнее использовать типовые математические схемы: дифференциальные уравнения, конечные и вероятностные автоматы, СМО и т.д.

Не обладающие такой степенью общности, как модели (2.3), (2.4), типовые математические схемы имеют преимущество простоты и наглядности, но при существенном сужении возможности применения.

В качестве **детерминированных** моделей, когда при исследовании случайный факт не учитывается, для представления систем, функционирующих в непрерывном времени, используются дифференциальные, интегральные и др. уравнения, а для представления систем, функционирующих в дискретном времени — конечные автоматы и конечно разностные схемы.

В начале стохастических моделей (при учёте случайного фактора) для представления систем с дискретным временем используются вероятностные автоматы, а для представления систем с непрерывным временем — системы массового обслуживания (СМО). Большое практическое значение при исследовании сложных индивидуальных управленческих систем, к кото-



рым относятся автоматизированные системы управления, имеют так называемые **агрегативные** модели.

Агрегативные модели (системы) позволяют описать широкий круг объектов исследования с отображением системного характера этих объектов. Именно при агрегативном описании сложный объект расчленяется на конечное число частей (подсистем), сохраняя при этом связи, обеспечивая взаимодействие частей.

## 2.2. Непрерывно детерминированные модели (Д - схемы).

Рассмотрим особенности непрерывно детерминированного подхода на примере, используя в качестве ММ **дифференциальные уравнения**.

Дифференциальными уравнениями называются такие уравнения, в которых неизвестными будут функции одной переменной или нескольких переменных, причём в уравнение входят не только их функции но их производные различных порядков.

Если неизвестные – функции многих переменных, то уравнения называются — **уравнения в частных производных**. Если неизвестные функции одной независимой переменной, то имеют место **обыкновенные дифференциальные уравнения**.

Математическое соотношение для детерминированных систем в общем виде:

$$\vec{y}' = \vec{f}(\vec{y}, t), \vec{y}(t_0) = \vec{y}_0 \quad (2.6)$$

Например, процесс малых колебаний маятника описан обыкновенными дифференциальным уравнением

$$m_1 l_1^2 \frac{\partial^2 \Theta(t)}{\partial t^2} + m_2 g l \Theta(t) = 0,$$

где  $m_1, l_1$  – масса, длина подвески маятника,  $\Theta$  – угол отклонения маятника от положения равновесия. Из этого уравнения можно найти оценки интересующих характеристик, например период колебаний  $T = 2\pi\sqrt{l/g}$ .

Дифференциальные уравнения, Д – схемы являются математическим аппаратом теории систем автоматического регулирования, управления.

При проектировании и эксплуатации систем автоматического управления необходимо выбрать такие параметры системы, которые бы обеспечивали требуемую точность управления.

Следует отметить, что часто используемые в системах автоматического управления системы дифференциальных уравнений определяются путём линеаризацией управления объекта (системы), более сложного вида, имеющего нелинейности:

$$F(y^n, y^{n-1}, \dots, y, x^m, x^{m-1}, \dots, x_n) = 0;$$

$$\frac{dF}{dy_0^n} \Delta y^n + \frac{dF}{dy_0^{n-1}} \Delta y^{n-1} \dots \frac{dF}{dy_0} \Delta y + \Delta y = \frac{dF}{dx_0^m} \Delta y^m + \dots \frac{dF}{dx_0} \Delta x + \Delta x.$$

### 2.3. Дискретно-детерминированные модели (F-схемы)

Дискретно-детерминированные модели (ДДМ) являются предметом рассмотрения теории автоматов (ТА). ТА – раздел теоретической кибернетики, изучающей устройства, перерабатывающие дискретную информацию и меняющего свои внутренние состояния лишь в допустимые моменты времени.

Конечный автомат имеет множество внутренних состояний и входных сигналов, являющихся конечными множествами. Автомат задаётся F- схемой:

$$F = \langle z, x, y, \varphi, \psi, z_0 \rangle, \tag{2.7}$$

где  $z, x, y$  – соответственно конечные множества входных, выходных сигналов (алфавитов) и конечное множество внутренних состояний (алфавита).  $z_0 \in Z$  – начальное состояние;  $\varphi(z, x)$  – функция переходов;  $\psi(z, x)$  – функция выхода. Автомат функционирует в дискретном автоматном времени, моментами которого являются такты, т.е. примыкающие друг к другу равные интервалы времени, каждому из которых соответствуют постоянные значения входного, выходного сигнала и внутреннего состояния. Абстрактный автомат имеет один входной и один выходной каналы.

В момент  $t$ , будучи в состоянии  $z(t)$ , автомат способен воспринять сигнал  $x(t)$  и выдать сигнал  $y(t)=\psi[z(t),x(t)]$ , переходя в состояние  $z(t+1)=\phi[z(t),z(t)]$ ,  $z(t)\in Z$ ;  $y(t)\in Y$ ;  $x(t)\in X$ . Абстрактный КА в начальном состоянии  $z_0$  принимая сигналы  $x(0)$ ,  $x(1)$ ,  $x(2)$  ... выдаёт сигналы  $y(0)$ ,  $y(1)$ ,  $y(2)$ ... (выходное слово).

Существуют  $F$ - автомат 1-ого рода (Миля), функционирующий по схеме:

$$z(t+1)=\phi[z(t),z(t)], t=0,1,2,\dots \quad (2.8)$$

$$y(t)=\psi[z(t),x(t)], t=0,1,2,\dots \quad (2.9)$$

$F$ - автомат 2-ого рода:

$$z(t+1)=\phi[z(t),z(t)], t=0,1,2,\dots \quad (2.10)$$

$$y(t)=\psi[z(t),x(t-1)], t=1,2,3,\dots \quad (2.11)$$

Автомат 2-ого рода, для которого

$$y(t)=\psi[z(t)], t=0,1,2,\dots \quad (2.12)$$

т.е. функция выходов не зависит от входной переменной  $x(t)$ , называется автоматом Мура.

Таким образом, уравнения 2.8-2.12 полностью задающие  $F$ - автомат, являются частным случаем уравнения

$$\bar{z}(t) = \Phi(\bar{z}_0, \bar{x}, \bar{v}, \bar{h}, t) \quad (2.13)$$

где  $\bar{z}$  - вектор состояния,  $\bar{x}$  - вектор независимых входных переменных,  $\bar{v}$  - вектор воздействий внешней среды,  $\bar{h}$  - вектор собственных внутренних параметров системы,  $\bar{z}_0$  - вектор начального состояния,  $t$  - время; и уравнение

$$\bar{y}(t) = F(\bar{z}, t), \quad (2.14)$$

когда система  $S$  - денормированная и на её вход поступает дискретный сигнал  $x$ .

По числу состояний конечные автоматы бывают с памятью и без памяти. Автоматы с памятью имеют более одного состояния, а автоматы без памяти (комбинационные или логические схемы) обладают лишь одним состоянием. При этом согласно (2.9), работа комбинационной схемы заключается в том, что она ставит в соответствие каждому входному сигналу  $x(t)$  определённый выходной сигнал  $y(t)$ , т.е. реализует логическую функцию вида:

$$y(t)=\psi[x(t)], t=0,1,2,\dots$$

Эта функция называется булевой, если алфавиты  $X$  и  $Y$ , которым принадлежат значения сигналов  $x$  и  $y$  состоят из 2-х букв.

По характеру отсчёта времени (дискретному)  $F$ - автоматы делятся на синхронные и асинхронные. В синхронных автоматах моменты времени, в которые автомат "считывает" входные сигналы, определяются принудительно синхронизирующими сигналами. Реакция автомата на каждое значение входного сигнала заканчивается за один такт синхронизации. Асинхронный  $F$ - автомат считывает входной сигнал непрерывно и поэтому, реагируя на достаточно длинный входной сигнал постоянной величины  $x$ , он может, как это следует из 1-5, несколько раз изменить своё состояние, выдавая соответствующее число выходных сигналов, пока не перейдёт в устойчивое.

Для задания  $F$ - автомата необходимо описать все элементы множества  $F = \langle z, x, y, \varphi, \psi, z_0 \rangle$ , т.е. входной, внутренний и выходной алфавиты, а также функции переходов и выходов. Для задания работы  $F$ - автоматов наиболее часто используются табличный, графический и матричный способ.

В табличном способе задания используется таблицы переходов и выходов, строки которых соответствуют входным сигналам автомата, а столбцы - его состояниям. При этом обычно 1-ый столбец слева соответствует начальному состоянию  $z_0$ . На пересечении  $i$ -ой строки и  $j$ -ого столбца таблицы переходов помещается соответствующее значение  $\varphi(z_k, x_i)$  функции переходов, а в таблице выходов -  $\psi(z_k, x_i)$  функции выходов. Для  $F$ - автомата Мура обе таблицы можно совместить, получив т.н. отмеченную таблицу переходов, в которой над каждым состоянием  $z_k$  автомата, обозначающим столбец таблицы, стоит соответствующий этому состоянию, согласно (2.12), выходной сигнал  $\psi(z_i)$ .

Описание работы  $F$ - автомата Мили таблицами переходов  $\varphi$  и выходов  $\psi$  иллюстрируется таблицей 2.1, а описание  $F$ - автомата Мура - таблицей переходов 2.2. Примеры табличного способа задания  $F$ - автомата Мили  $F1$  с тремя состояниями, двумя входными и двумя выходными сигналами приведены в таблице 2.3, а для  $F$ - автомата Мура  $F2$  - в таблице 2.4.

**Таблица 2.1. Описание работы F- автомата Мили**

$X_j$	$Z_k$			
	$Z_0$	$Z_1$	...	$Z_k$
Переходы				
$X_1$	$\varphi(Z_0, X_1)$	$\varphi(Z_1, X_1)$	...	$\varphi(Z_k, X_1)$
$X_2$	$\varphi(Z_0, X_2)$	$\varphi(Z_1, X_2)$	...	$\varphi(Z_k, X_2)$
.....				
$X_i$	...	...	...	...
Выходы				
$X_1$	$\psi(Z_0, X_1)$	$\psi(Z_1, X_1)$	...	$\psi(Z_k, X_1)$
.....				
$X_i$	$\psi(Z_0, X_i)$	$\psi(Z_1, X_i)$	...	$\psi(Z_k, X_i)$

**Таблица 2.2. Описание работы F- автомата Мура**

$X_i$	$\psi(Z_k)$			
	$\psi(Z_0)$	$\psi(Z_1)$	...	$\psi(Z_k)$
	$Z_0$	$Z_1$	...	$Z_k$
$X_1$	$\varphi(Z_0, X_1)$	$\varphi(Z_1, X_1)$	...	$\varphi(Z_k, X_1)$
$X_2$	$\varphi(Z_0, X_2)$	$\varphi(Z_1, X_2)$	...	$\varphi(Z_k, X_2)$
.....				
$X_i$	$\varphi(Z_0, X_i)$	$\varphi(Z_1, X_i)$	...	$\varphi(Z_k, X_i)$

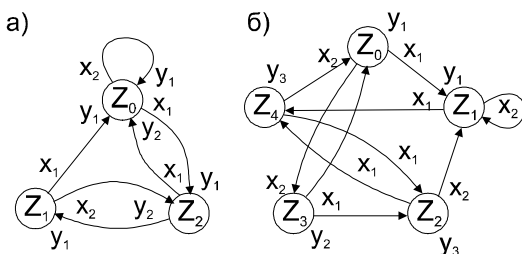
**Таблица 2.3. Табличный способ задания F- автомата Мили F1**

$X_j$	$Z_0$		
	$Z_0$	$Z_1$	$Z_2$
Переходы			
$X_1$	$Z_2$	$Z_0$	$Z_0$
$X_2$	$Z_0$	$Z_2$	$Z_1$
Выходы			
$X_1$	$y_1$	$y_1$	$y_2$
$X_2$	$y_1$	$y_2$	$y_1$

**Таблица 2.4. Табличный способ задания F- автомата Мура F2**

$X_i$	$y$				
	$y_1$	$y_1$	$y_3$	$y_2$	$y_3$
	$Z_0$	$Z_1$	$Z_2$	$Z_3$	$Z_4$
$X_1$	$Z_1$	$Z_4$	$Z_4$	$Z_2$	$Z_2$
$X_2$	$Z_3$	$Z_1$	$Z_1$	$Z_0$	$Z_0$

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 переходов, дуги графа необходимо отметить соответствующими выходными сигналами. Для автоматов Мили эта разметка производится так: если входной сигнал  $x_k$  действует на состояние  $z_i$ , то согласно сказанному получается дуга, исходящая из  $z_i$  и помеченная  $x_k$ ; эту дугу дополнительно отмечают выходным сигналом  $y = \psi(z_i, x_k)$ . Для автомата Мура аналогичная разметка графа такова: если входной сигнал  $x_k$ , действуя на некоторое состояние автомата, вызывает переход в состояние  $z_j$ , то дугу, направленную в  $z_j$  и помеченную  $x_k$ , дополнительно отмечают выходным сигналом  $y = \psi(z_j, x_k)$ . На рис. 2.1 приведены заданные ранее таблицами  $F$ - автоматы Мили  $F1$  и Мура  $F2$  соответственно.



**Рис. 2.1** Графы автоматов Мили (а) и Мура (б)

При решении задач моделирования часто более удобной формой является матричное задание конечного автомата. При этом матрица соединений автомата есть квадратная матрица  $C = \|c_{ij}\|$ , строки которой соответствуют исходным состояниям, а столбцы - состояниям перехода. Элемент  $c_{ij} = x_k/y_s$  в случае автомата Мили соответствует входному сигналу  $x_k$ , вызывающему переход из состояния  $z_i$  в состояние  $z_j$  и выходному сигналу  $y_s$ , выдаваемому при этом переходе. Для автомата Мили  $F1$ , рассмотренного выше, матрица соединений имеет вид:

$$C_1 = \begin{pmatrix} x_2/y_1 & - & x_1/y_1 \\ x_1/y_1 & - & x_2/y_2 \\ x_1/y_2 & x_2/y_1 & - \end{pmatrix}.$$

Если переход из состояния  $z_i$  в состояние  $z_j$  происходит под действием нескольких сигналов, элемент матрицы  $c_{ij}$  представляет собой множество пар "вход/выход" для этого перехода, соединённых знаком дизъюнкции.

Для  $F$ - автомата Мура элемент  $c_{ij}$  равен множеству входных сигналов на переходе ( $z_i z_j$ ), а выход описывается вектором выходов:

$$\bar{y} = \begin{bmatrix} \Psi(z_0) \\ \Psi(z_1) \\ \dots \\ \Psi(z_k) \end{bmatrix} \text{ } i\text{-ая компонента которого выходной сигнал,}$$

отмечающий состояние  $z_i$ .

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

## 2.4. Непрерывно-стохастические модели (Q - схемы)

К непрерывно-стохастическим моделям относятся системы массового обслуживания (англ. queueing system), которые называют  $Q$ - схемами.

Предмет теории массового обслуживания – системы массового обслуживания (СМО) и сети массового обслуживания. Под СМО понимают динамическую систему, предназначенную для эффективного обслуживания случайного потока заявок при ограниченных ресурсах системы. Обобщённая структура СМО приведена на рис. 2.2.

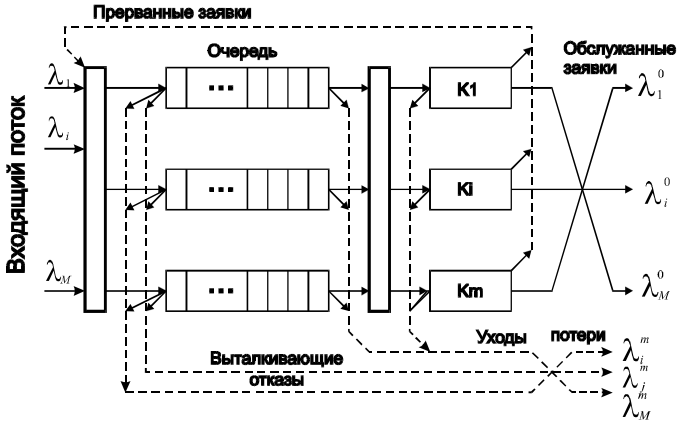


Рис. 2.2 Схема СМО

Поступающие на вход СМО однородные заявки в зависимости от порождающей причины делятся на типы, интенсивность потока заявок типа  $i$  ( $i=1..M$ ) обозначено  $\lambda_i$ . Совокупность заявок всех типов – входящий поток СМО.

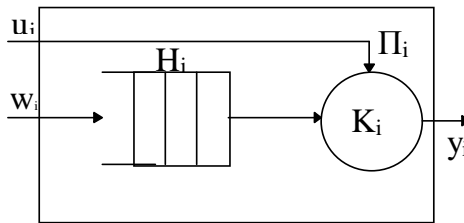
Обслуживание заявок выполняется  $m$  каналами. Различают универсальные и специализированные каналы обслуживания. Для универсального канала типа  $j$  считается известными функции распределения  $F_{ji}(\tau)$  длительности обслуживания заявок произвольного типа. Для специализированных каналов функции распределения длительности обслуживания каналов заявок некоторых типов являются неопределёнными, назначение этих заявок на данный канал.

В качестве процесса обслуживания могут быть представлены различные по своей физической природе процессы функционирования экономических, производственных, технических и других систем, например, потоки поставок продукции некоторому предприятию, потоки деталей и комплектующих изделий на сборочном конвейере цеха, заявки на обработку информации ЭВМ от удалённых терминалов и т.д. При этом характерным для работы таких объектов является случайное поведение заявок (требований) на обслуживание и завершение обслуживания в случайные моменты времени.



$Q$ - схемы можно исследовать аналитически и имитационными моделями. Последнее обеспечивает большую универсальность. Рассмотрим понятие массового обслуживания.

В любом элементарном акте обслуживания можно выделить две основные составляющие: ожидание обслуживания заявкой и собственно обслуживание заявки. Это можно отобразить в виде некоторого  $i$ -ого прибора обслуживания  $\Pi_i$ , состоящего из накопителя заявок, в котором может находиться одновременно  $l_i=0 \dots L_i^H$  заявок, где  $L_i^H$  - ёмкость  $i$ -ого накопителя, и канала обслуживания заявок,  $k_i$ .



**Рис. 2.3** Схема прибора СМО

На каждый элемент прибора обслуживания  $\Pi_i$  поступают потоки событий: в накопитель  $H_i$  поток заявок  $w_i$ , на канал  $k_i$  - поток обслуживания  $u_i$ .

**Потоком событий** (ПС) называется последовательность событий, происходящих одно за другим в какие-то случайные моменты времени. Различают потоки однородных и неоднородных событий. Однородный ПС (ОПС) характеризуется только моментами поступления этих событий (вызывающими моментами) и задаётся последовательностью  $\{t_n\} = \{0 \leq t_1 \leq t_2 \dots \leq t_n \leq \dots\}$ , где  $t_n$  - момент поступления  $n$ -ого события - неотрицательное вещественное число. ОПС может быть также задан в виде последовательности промежутков времени между  $n$ -ым и  $n-1$ -ым событиями  $\{z_n\}$ .

Неоднородным ПС называется последовательность  $\{t_n, f_n\}$ , где  $t_n$ - вызывающие моменты;  $f_n$ - набор признаков события. Например, может быть задана принадлежность к тому или ино-

му источнику заявок, наличие приоритета, возможность обслуживания тем или иным типом канала и т.п.

Рассмотрим ОПС, для которого  $\tau_i \in \{\tau_n\}$  - случайные величины, независимые между собой. Тогда ПС называется потоком с ограниченным последствием.

ПС называется ординарным, если вероятность того, что на малый интервал времени  $\Delta t$ , примыкающий к моменту времени  $t$  попадает больше одного события  $P_{\geq 1}(t, \Delta t)$  пренебрежительно мала.

Если для любого интервала  $\Delta t$  событие  $P_0(t, \Delta t) + P_1(t, \Delta t) + P_{\geq 1}(t, \Delta t) = 1$ ,  $P_1(t, \Delta t)$  - вероятность попадания на интервал  $\Delta t$  ровно одного события. Как сумма вероятностей событий, образующих полную группу и несовместных, то для ординарного потока событий  $P_0(t, \Delta t) + P_1(t, \Delta t) \approx 1$ ,  $P_{\geq 1}(t, \Delta t) = \Theta(\Delta t)$ , где  $\Theta(\Delta t)$  - величина, порядок малости который выше, чем  $\Delta t$ , т.е.  $\lim(\Theta(\Delta t)) = 0$  при  $\Delta t \rightarrow 0$ .

Стационарным ПС называется поток, для которого вероятность появления того или иного числа событий на интервале времени  $\tau$  зависит от длины этого участка и не зависит от того, где на оси времени  $0 - t$  взят этот участок. Для ОПС справедливо  $0 * P_0(t, \Delta t) + 1 * P_1(t, \Delta t) = P_1(t, \Delta t)$  - среднее число событий на интервале  $\Delta t$ . Среднее число событий, наступающих на участке  $\Delta t$  в единицу времени составляет  $P_1(t, \Delta t) / \Delta t$ . Рассмотрим предел этого выражения при  $\Delta t \rightarrow 0$

$$\lim P_1(t, \Delta t) / \Delta t = \lambda(t) * (1/\text{един.вр.}).$$

Если этот предел существует, то он называется интенсивностью (плотностью) ОПС. Для стандартного ПС  $\lambda(t) = \lambda = \text{const}$ .

Применительно к элементарному каналу обслуживания  $k_i$  можно считать что поток заявок  $w_i \in W$ , т.е. интервалы времени между моментами появления заявок на входе  $k_i$  образуют подмножество неуправляемых переменных, а поток обслуживания  $u_i \in U$ , т.е. интервалы времени между началом и окончанием обслуживания заявки образуют подмножество управляемых переменных.

Заявки, обслуженные каналом  $k_i$  и заявки, покинувшие прибор  $\Pi_i$  по различным причинам не обслуженными, образуют выходной поток  $y_i \in Y$ .

Процесс функционирования прибора обслуживания  $\Pi_i$  можно представить как процесс изменения состояний его элементов во времени  $Z_i(t)$ . Переход в новое состояние для  $\Pi_i$  означает изменение кол-ва заявок, которые в нём находятся (в канале  $k_i$  и накопителе  $H_i$ ). Т.о. вектор состояний для  $\Pi_i$  имеет вид:  $\vec{Z}_i = Z_i^H, Z_i^K$ , где  $Z_i^H$  - состояния накопителя, ( $Z_i^H = 0$  - накопитель пуст,  $Z_i^H = 1$  - в накопителе одна заявка...,  $Z_i^H = L_i^H$  - накопитель занят полностью;  $Z_i^K$  - состояние канала  $k_i$  ( $Z_i^K = 0$  - канал свободен,  $Z_i^K = 1$  канал занят).

$Q$ -схемы реальных объектов образуются композицией многих элементарных приборов обслуживания  $\Pi_i$ . Если  $k_i$  различных приборов обслуживания соединены параллельно, то имеет место многоканальное обслуживание (многоканальная  $Q$ -схема), а если приборы  $\Pi_i$  и их параллельные композиции соединены последовательно, то имеет место многофазное обслуживание (многофазная  $Q$ -схема).

Таким образом, для задания  $Q$ -схемы необходимо оператор сопряжения  $R$ , отражающий взаимосвязь элементов структуры.

Связи в  $Q$ -схеме изображают в виде стрелок (линий потока, отражающих направление движения заявок). Различают разомкнутые и замкнутые  $Q$ -схемы. В разомкнутой выходной поток не может снова поступить на какой-либо элемент, т.е. обратная связь отсутствует.

Собственными (внутренними) параметрами  $Q$ -схемы будут являться количество фаз  $L_\phi$ , количество каналов в каждой фазе,  $L_{kj}$ ,  $j=1 \dots L_\phi$ , количество накопителей каждой фазы  $L_{kj}$ ,  $k=1 \dots L_\phi$ , ёмкость  $i$ -ого накопителя  $L_i^H$ . Следует отметить, что в теории массового обслуживания в зависимости от ёмкости накопителя применяют следующую терминологию:

- системы с потерями ( $L_i^H=0$ , накопитель отсутствует);
- системы с ожиданием ( $L_i^H \rightarrow \infty$ );

- системы с ограниченной ёмкостью накопителя  $H_i$  (смешанные).

Обозначим всю совокупность собственных параметров  $Q$ -схемы как подмножество  $H$ .

Для задания  $Q$ -схемы также необходимо описать алгоритмы её функционирования, которые определяют правила поведения её заявок в различных неоднозначных ситуациях.

В зависимости от места возникновения таких ситуаций различают алгоритмы (дисциплины) ожидания заявок в накопителе  $H_i$  и обслуживания заявок каналом  $k_i$ . Неоднородность потока заявок учитывается с помощью введения класса приоритетов.

В зависимости от динамики приоритетов  $Q$ -схемы различают статические и динамические. Статические приоритеты назначаются заранее и не зависят от состояний  $Q$ -схемы, т.е. они являются фиксированными в пределах решения конкретной задачи моделирования. Динамические приоритеты возникают при моделировании. Исходя из правил выбора заявок из накопитель  $H_i$  на обслуживание каналом  $k_i$  можно выделить относительные и абсолютные приоритеты. **Относительный приоритет** означает, что заявка с более высоким приоритетом, поступившая в накопитель  $H$ , ожидает окончания обслуживания представляющей заявки каналом  $k_i$  и только после этого занимает канал. **Абсолютный приоритет** означает, что заявка с более высоким приоритетом, поступившая в накопитель, прерывает обслуживание каналом  $k_i$  заявки с более низким приоритетом и сама занимает канал (при этом вытесненная из  $k_i$  заявка может либо покинуть систему, либо может быть снова записана на какое-то место в  $H_i$ ).

Необходимо также знать набор правил, по которым заявки покидают  $H_i$  и  $k_i$ : для  $H_i$  – либо правила переполнения, либо правила ухода, связанные с истечением времени ожидания заявки в  $H_i$ ; для  $k_i$  – правила выбора маршрутов или направлений ухода. Кроме того, для заявок необходимо задать правила, по которым они остаются в канале  $k_i$ , т.е. правила блокировок канала. При этом различают блокировки  $k_i$  по выходу и по входу. Такие блокировки отражают наличие управляющих связей в  $Q$ -схеме, регулирующих поток заявок в зависимости от состоя-

ний  $Q$ -схемы. Набор возможных алгоритмов поведения заявок в  $Q$ -схеме можно представить в виде некоторого оператора алгоритмов поведения заявок  $A$ .

Таким образом,  $Q$ -схема, описывающая процесс функционирования СМО любой сложности однозначно задаётся в виде набора множеств:  $Q = \langle W, U, H, Z, R, A \rangle$ .

### **Выводы по лекции 2:**

Исходной информацией при построении математических моделей процессов функционирования систем служат данные о назначении и условиях работы исследуемой системы. В практике моделирование объектов в области системного анализа на первичных этапах исследования рациональнее использовать типовые математические схемы: дифференциальные уравнения, конечные и вероятностные автоматы, СМО и т.д. Существуют различные схемы моделирования систем, среди которых Д-схемы, F-схемы и Q-схемы.

### **Вопросы для самопроверки по лекции 2:**

- Какая исходная информация необходима для построения математических моделей процессов функционирования систем?
- Дайте определение понятия математическая схема.
- Дайте определения агрегативной модели и охарактеризуйте её значение для исследования индивидуальных управленческих систем.
- Дайте понятие и охарактеризуйте непрерывно-детерминированную модель.
- Дайте понятие и охарактеризуйте дискретно-детерминированную модель.
- Дайте понятие и охарактеризуйте непрерывно-стохастическую модель.
- Приведите примеры работы F-автомата Мили и Мура.

## **Лекция 3. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ СИСТЕМ**

### **Цель лекции**

В лекции будут рассмотрены процедура имитационного моделирования, описан процесс имитации функционирования системы, общие алгоритмы моделирования систем и некоторые методы определения характеристик моделируемых систем.

### **3.1. Процедура имитационного моделирования**

Метод имитационного моделирования (ИМ) заключается в создании логико-аналитической (математической модели системы и внешних воздействий), имитации функционирования системы, т.е. в определении временных изменений состояния системы под влиянием внешних воздействий и в получении выборок значений выходных параметров, по которым определяются их основные вероятностные характеристики. Данное определение справедливо для стохастических систем.

При исследовании детерминированных систем отпадает необходимость изучения выборок значений выходных параметров.

Модель системы со структурным принципом управления представляет собой совокупность моделей элементов и их функциональные взаимосвязи. Модель элемента (агрегата, обслуживающего прибора) – это, в первую очередь, набор правил (алгоритмов) поведения устройства по отношению к выходным воздействиям (заявкам) и правил изменений состояний элемента. Элемент отображает функциональное устройство на том или ином уровне детализации. В простейшем случае устройство может находиться в работоспособном состоянии или в состоянии отказа. В работоспособном состоянии устройство может быть занято, например, выполнение операции по обслуживанию заявки или быть свободным. К правилам поведения устройства относятся правила выборки заявок из очереди; реакция устройства на поступление заявки, когда устройство занято или к нему име-

ется очередь заявок; реакция устройства на возникновение отказа в процессе обслуживания заявки и некоторые другие.

Имитационное моделирование – это метод исследования, который основан на том, что анализируемая динамическая система заменяется имитатором и с ним производятся эксперименты для получения об изучаемой системе. Роль имитатора зачастую выполняет программа ЭВМ.

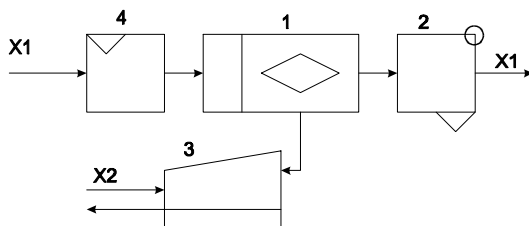
Основная идея метода ИМ состоит в следующем. Пусть необходимо определить функцию распределения случайной величины  $y$ . Допустим, что искомая величина  $y$  может быть представлена в виде зависимости:  $y=f(\alpha,\beta,\dots,\omega)$  где  $\alpha,\beta,\dots,\omega$  случайные величины с известными функциями распределения.

Для решения задач такого вида применяется следующий алгоритм:

- 1) по каждой из величин  $\alpha,\beta,\dots,\omega$  производится случайное испытание, в результате каждого определяется некоторое конкретное значение случайной величины  $\alpha_i,\beta_i,\dots,\omega_i$ ;
- 2) используя найденные величины, определяется одно частное значение  $y_i$  по выше приведённой зависимости;
- 3) предыдущие операции повторяются  $N$  раз, в результате чего определяется  $N$  значений случайной величины  $y$ ;
- 4) на основании  $N$  значений величины находится её эмпирическая функция распределения.

### **3.2. Имитация функционирования системы**

Предположим, исследуется вычислительная система (ВС), в которую входит процессор 1 с основной памятью, устройство ввода перфокарт 4, алфавитно-цифровое печатающее устройство (АЦПУ) 2 и дисплей 3 (рис. 3.1.).



**Рис. 3.1 Упрощённая схема моделируемой системы**

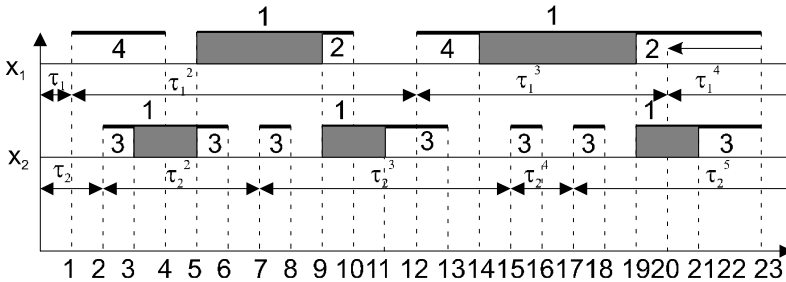
Через устройство 4 поступает поток заданий  $X1$ . Процессор обрабатывает задания и результаты выдаёт на АЦПУ 2. Одновременно с этим ВС используется, например, как информационно-справочная система. Оператор-пользователь, работающий за дисплеем, посылает в систему запросы  $X2$ , которые обрабатываются процессором и ответы выводятся на экран дисплея. Процессор работает в 2-х программном режиме: в одном разделе обрабатываются задания  $X1$ , в другом, с более высоким относительным приоритетом запросы  $X2$ . Представим данную ВС в упрощённом варианте в виде стохастической сети из 4-х СМО. Потоки заданий и запросы будем называть потоками заявок. Считаем потоки  $X1$  и  $X2$  независимыми. Известны функции распределения периодов следования заявок  $\tau_1$  и  $\tau_2$  и длительность обслуживания  $T_{1k}$ ,  $T_{2k}$  заявок в  $k$ -ом устройстве. Требуется определить времена загрузки каждого устройства и времена реакции по каждому из потоков.

Вначале определяется момент поступления в систему 1-ой заявки потока  $X1$  по результатам случайного испытания в соответствии с функцией распределения периода следования заявок.

На рис. 3.2 это момент времени  $t_1=0+\tau_1^1$  (здесь и далее верхний индекс обозначает порядковый номер заявки данного потока). То же самое делается для потока  $X2$ . На рис. 3.2 момент поступления 1-ой заявки потока  $X2$   $t_2=0+\tau_2^1$ . Затем находится минимальное время, т.е. наиболее раннее событие. В примере это время  $t_1$ . Для 1-ой заявки потока  $X1$  определяется время обслуживания устройством ввода перфкарт  $T_{14}$  методом случай-



ного испытания и отмечается момент окончания обслуживания  $t_4=t_1+T_{14}^1$ .



**Рис. 3.2** Временная диаграмма функционирования ВС

На рис. 3.2 показан переход устройства 4 в состояние "занято". Одновременно определяется момент поступления следующей заявки потока X1:  $t_{12}=t_1+\tau_1^2$ . Следующее минимальное время это момент поступления заявки потока X2 –  $t_2$ . Для этой заявки находится время обслуживания на дисплее  $T_{23}^1$  и отслеживается время окончания обслуживания  $t_3=t_2+T_{23}^1$ . Определяется момент поступления второй заявки потока X2:  $t_7=t_2+\tau_2^2$ . Снова выбирается минимальное время – это  $t_3$ . В этот момент заявка потока X2 начинает обрабатываться процессором. По результату случайного испытания определяется время её обслуживания  $T_{21}^1$  и отмечается момент  $t_5=t_3+T_{21}^1$  окончания обслуживания. Следующее минимальное время  $t_4$  – момент завершения обслуживания заявки потока X1 устройством 4. С этого момента заявка может начать обрабатываться процессором, но он занят обслуживанием потока X2. Тогда заявка потока X1 переходит в состояние ожидания, становится в очередь. В следующий момент времени  $t_5$  освобождается процессор. С этого момента процессор начинает обрабатывать заявку потока X1, а заявка потока X2 переходит на обслуживание дисплеем, т.е. ответ на запрос пользователя передаётся из основной памяти в буферный накопитель дисплея. Далее определяются соответствующие времена обслуживания:  $T_{11}^1$  и  $T_{23}^1$  и отмечают моменты времени  $t_9=t_5+T_{11}^1$  и  $t_6=t_5+T_{23}^1$ . В момент  $t_6$  полностью завершается обработка первой заявки потока X2. По разности времени  $t_6$  и  $t_2$

вычисляется время реакции по этой заявке  $u^1_2 = t_6 - t_2$ . Следующий минимальный момент  $t_7$  – это наступление 2-ой заявки потока  $X_2$ . Определяет время поступления очередной заявки этого потока  $t_{15} = t_7 + \tau_2^3$ . Затем вычисляется время обслуживания 2-ой заявки на дисплее  $T^2_{23}$  и отмечается момент  $t_8 = t_7 + T^2_{23}$ , после чего заявка становится в очередь, т.к. процессор занят. Эта заявка поступит на обслуживание в процессор только после его освобождения в момент  $t_9$ . В этот момент заявка потока  $X_1$  начинает обслуживаться в АЦПУ. Определяются времена обслуживания  $T^2_{21}$  и  $T^1_{12}$  по результатам случайных испытаний и отмечаются моменты окончания обслуживания  $t_{11} = t_9 + T^2_{23}$  и  $t_{10} = t_9 + T^1_{12}$ . В момент времени  $t_{10}$  завершается полное обслуживание 1-ой заявки потока  $X_1$ . Разность между этим моментом и моментом времени  $t_1$  даёт 1-ое значение времени реакции по потоку  $X_1$   $u^1_1 = t_{10} - t_1$ .

Указанные процедуры выполняются до истечения времени моделирования. В результате получается некоторое количество (выборка) случайных значений времени реакции ( $u_1$ ) и ( $u_2$ ) по 1-ому и 2-ому 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. Например, на рис. 3.2 занятость процессора 1 выделена заштрихованными ступеньками. 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, требуемая ёмкость памяти и др.

Имитация даёт возможность учесть надёжностные характеристики ВС. 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;

### **3.3. Алгоритм моделирования по принципу особых состояний**

Алгоритм моделирования по принципу особых состояний использовался в приведённом выше примере. В качестве событий выделены:

- поступление заявки в систему;
- освобождение элемента после обслуживания заявки;
- завершения моделирования.

Другими типами событий являются:

- возникновение отказа устройств;
- завершение восстановления устройств.

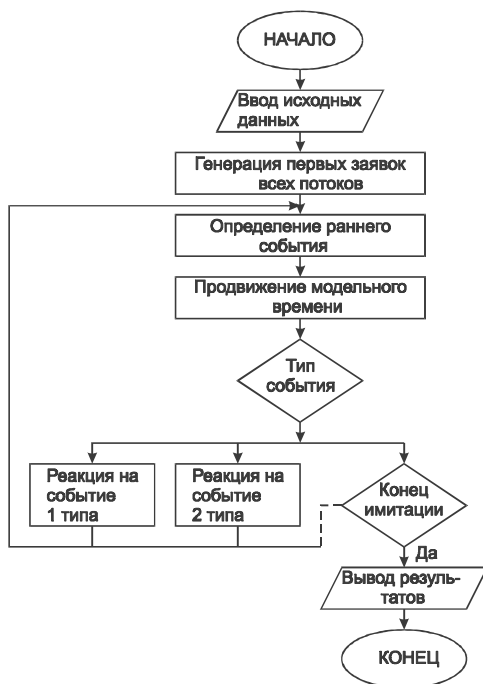
Процесс имитации развивался с использованием управляющих последовательностей, определяемых по функциям распределения вероятностей исходных данных путём проведения случайных испытаний. В качестве управляющих последовательностей использовались в примере последовательности значений периодов следования заявок по каждому  $i$ -ому потоку  $\{\tau_i\}$  и длительности обслуживания заявок  $i$ -ого потока устройством  $\{T_{ik}\}$ . Моменты наступления будущих событий определялись по простым рекуррентным соотношениям. 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 буду-

щих событий; эти действия называют реакцией модели на события;

- 5) перечисленные действия повторяются до истечения времени моделирования.

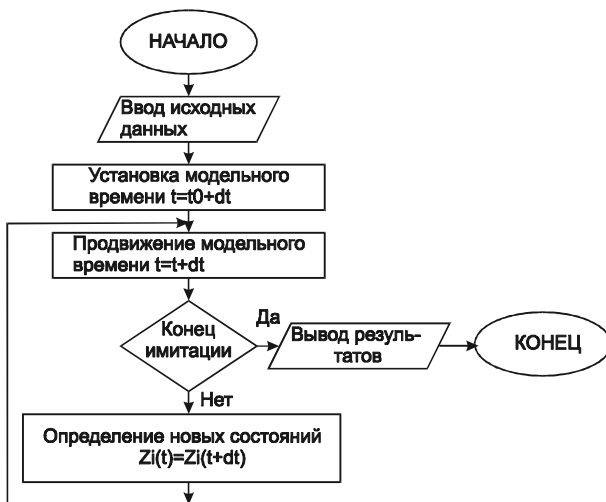
В процессе моделирования производится измерение и статистическая обработка значений выходных характеристик. Обобщённая схема алгоритма моделирования по принципу особых состояний приведена на рис. 3.3.



**Рис. 3.3 Обобщённый алгоритм моделирования систем по принципу особых состояний**

### **3.4. Алгоритм моделирования по принципу $\Delta t$**

Укрупнённая схема моделирующего алгоритма, который реализует принцип постоянного приращения модельного времени (принципа  $\Delta t$ ), представлен на рис. 3.4.



**Рис. 3.4** Обобщённый алгоритм моделирования систем по принципу приращений « $\Delta t$ »

В начале инициализируется программа, в частности вводятся значения  $Z_i(t_0)$ ,  $i=1,2,\dots,k$ . Которые характеризуют состояние системы в  $k$ -мерном фазовом пространстве состояний в начальный момент времени  $t_0$ . Модельное время устанавливается  $t = t_0 = 0$ . Основные операции по имитации системы осуществляется в цикле. Функционирование системы отслеживается по последовательной схеме состояний  $Z_i(t)$ . Для этого модельному даётся некоторое приращение  $dt$ . Затем по вектору текущих состояний определяются новые состояния  $Z_i(t + dt)$ , которые становятся текущими. Для определения новых состояний по текущим в формализованном описании системы должны существовать необходимые математические зависимости. По ходу имитации измеряются, вычисляются, фиксируются необходимые выходные характеристики. При моделировании стохастических систем вместо новых состояний вычисляются распределения вероятностей для возможных состояний. Конкретные значения вектора текущих состояний определяются по результатам случайных испытаний. В результате проведения имитационного эксперимента получается одна из возможных реализаций

случайного многомерного процесса в заданном интервале времени  $(t_0, T_k)$ .

Моделирующий алгоритм, основанный на применении  $dt$  применим для более широкого круга систем, чем алгоритм, построенный по принципу особых состояний. Однако при его реализации возникают проблемы определения величины  $dt$ . Для моделирования ВС на системном уровне в основном используются принцип особых состояний.

### **3.5. Методы определения характеристик моделируемых систем**

#### **Измеряемые характеристики моделируемых систем**

При имитационном моделировании можно измерять значения любых характеристик, интересующих исследователя. Обычно по результатам вычислений определяются характеристики всей системы, каждого потока и устройства.

Для всей системы производится подсчёт поступивших в систему заявок, полностью обслуженных и покинувших систему заявок без обслуживания по тем или иным причинам. Соотношения этих величин характеризует производительность 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  $m$  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. Коэффициенты загрузки устройств вычисляются по формуле:

$$\rho_k = V_k * N_{ok} / T_m$$

$V_k$  – среднее время обслуживания одной заявки  $k$ -ым устройством;  $N_{ok}$  – количество обслуженных заявок устройством за время моделирования  $T_m$ .

Определение условий удовлетворения стохастических ограничений при имитационном моделировании производится путём простого подсчёта количества измерений, вышедших и не вышедших за допустимые пределы.

### **Расчёт математического ожидания и дисперсии выходной характеристики**

В случае стационарного эргодического процесса функционирования системы вычисление  $M(y)$  и  $D(y)$  выходной характеристики  $y$  производится усреднением не по времени, а по множеству  $N_{\text{знач.}}$  измеренных по одной реализации достаточной длительности. В целях экономия ОЗУ ЭВМ  $M(y)$  и  $D(y)$  вычисляются по рекуррентным формулам:

$$m_n = m_{n-1} * (n-1) / n + y / n;$$

где  $m_{n-1}$  - математическое ожидание, вычисленное на предыдущем шаге.

$$d_n = d_{n-1} * (n-2) / (n-1) + 1/n * (y_n - m_{n-1})^2 .$$

Здесь  $d_{n-1}$  – дисперсия, вычисленная на предыдущем шаге. При большом количестве измерений эти оценки являются состоятельными и несмещёнными.

### **Расчёт среднего по времени значения выходной характеристики**

Например, средняя длина очереди к каждому устройству вычисляется по формуле:

$$l_N = \sum_{i=1}^N \frac{l_i \tau_i}{T_m} ,$$

где  $i$  – номер очередного изменения состояния очереди (занесение заявки в очередь или исключение из очереди);  $N$  – количество изменений состояния очереди;  $\tau_i$  – интервал времени между двумя последними изменениями очереди.

Ёмкость накопителя:

$$q_n = \sum_{i=1}^N \frac{q_i \tau_i}{T_m},$$

где  $q_i$  – ёмкость накопителя, занятая в интервале между двумя последними обращениями к накопителю для ввода-вывода заявки.

### Построение гистограммы для стационарной системы

$\Gamma$  – эмпирическая плотность распределения вероятностей. Задаются границы изменения интересующей характеристики.  $y_i \rightarrow [y_n; y_6]$ , числом интервалов  $N_g$ . Определяется ширина интервала  $\Delta = (y_n - y_6) / N_g$ .

Затем в процессе моделирования по мере появления значений  $y_i$  определяется число попаданий этой случайной величины в каждый из интервалов  $R_i$  гистограммы. По этим данным вычисляется относительная частота по каждому интервалу:  $G_i = R_i / (N * \Delta)$ , где  $N$  – общее число измерений  $y$ . Площадь гистограммы равна единице, равна сумме площадей:

$$S_i = \sum_1^{N_g} G_i \Delta = \sum \frac{R_i}{N \Delta} \Delta = \sum \frac{R_i}{N} = 1, \text{ т.к. } N = \sum_1^{N_g} R_i.$$

При необходимости выдвигается гипотеза о том, что эмпирическое распределение согласуется с некоторым теоретическим распределением. Эта гипотеза проверяется по тому или иному критерию. Например, при использовании критерия  $\chi^2$  в качестве меры расхождения используется выражение

$$\chi^2 = \frac{\sum_1^{N_g} (R_i - NP_i)^2}{NP_i},$$

где  $P_i$  определяется из выбранного теоретического распределения вероятность попадания случайной величины в  $i$ -ый интервал.

$$P_i = \int_{y_i}^{y_{i+1}} \varphi(x) dx = F(y_{i+1}) - F(y_i).$$

Из теоремы Пирсона следует, что для любой функции распределения  $F(y)$  случайной величины  $y$  при  $N \rightarrow \infty$  распределения величины  $\chi^2$  имеет вид:



$$M_k(z) = p(\chi^2 < z) = \frac{1}{2^{k/2} \Gamma(k/2)} \int_0^z t^{k/2-1} e^{-t/2} dt,$$

где  $z$  – значение случайной величины  $\chi^2$ ,  $k = N_g - (r + 1)$  – число степеней свободы распределения  $\chi^2$ ,  $r$  – количество параметров теоретического распределения,  $\Gamma(k/2)$  – гамма функция.

Функция распределения  $\chi^2$  табулирована. По вычисленному значению  $\chi^2$  и числу степеней свободы с помощью таблиц определяется вероятность  $P(\chi^2 < Z)$ . Если она превышает заданный уровень значимости  $C$ , то выдвинутая гипотеза принимается.

### **Выводы по лекции 3:**

Метод имитационного моделирования заключается в создании логико-аналитической (математической модели системы и внешних воздействий), имитации функционирования системы, т.е. в определении временных изменений состояния системы под влиянием внешних воздействий и в получении выборок значений выходных параметров, по которым определяются их основные вероятностные характеристики. В процессе моделирования производится измерение и статистическая обработка значений выходных характеристик. При имитационном моделировании можно измерять значения любых характеристик, интересующих исследователя. Обычно по результатам вычислений определяются характеристики всей системы, каждого потока и устройства.

### **Вопросы для самопроверки по лекции 3:**

- В чем заключается метод имитационного моделирования?
- Охарактеризуйте алгоритм моделирования по принципу особых состояний.
- Поясните моделирующий алгоритм, который реализует принцип постоянного приращения модельного времени.

- Перечислите измеряемые характеристики моделируемых систем.
- Охарактеризуйте методы определения характеристик моделируемых систем.

## Лекция 4. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ВОЗДЕЙСТВИЙ

### Цель лекции

В лекции будут рассмотрены особенности моделирования случайных событий, особенности преобразования и вычисления случайных величин, а также моделирование нормально распределённой случайной величины.

### 4.1. Особенности моделирования случайных событий

Важной задачей в практике имитационного моделирования систем на ЭВМ является расчёт случайных величин. В языках программирования существуют датчики равномерно распределённых псевдослучайных величин в интервале  $\{0,1\}$ . Остановимся на вопросах преобразования последовательности псевдослучайных величин  $\{X_i\}$  в последовательности  $\{Y_i\}$  с заданным законом распределения и моделировании различных случайных событий.

Пусть имеются случайные числа  $x_i$ , т.е. возможные значения случайной величины  $\xi$ , равномерно распределённой в интервале  $\{0,1\}$ . Необходимо реализовать случайное событие  $A$ , наступающее с заданной вероятностью  $P$ . Определим  $A$  как событие, состоящее в том, что выбранное значение  $x_i$  удовлетворяет неравенству:

$$x_i \leq P. \quad (4.1)$$

Тогда вероятность события  $A$  будет :  $P(A) = \int_0^P dx = p$ .

Противоположное событию  $A$  состоит в том, что  $x_i > p$ . Тогда  $P(\bar{A}) = 1 - P$ . Процедура моделирования состоит в этом случае в выборе значений  $x_i$  и сравнение их с  $p$ . При этом, если условие (4.1) удовлетворяется, то исходом испытания будет событие  $A$ .

Таким же образом можно рассмотреть группу событий. Пусть  $A_1, A_2, \dots, A_n$  – полная группа событий, наступающая с вероятностями  $P_1, P_2, \dots, P_n$  соответственно. Определим  $A_m$  как собы-

тие, состоящее, в том, что выбранное значение  $x_i$  случайной величины  $\xi$  удовлетворяет неравенству:

$$l_{m-1} < x_i < l_m, \text{ где } l_k = \sum_{i=1}^k p_i. \quad (4.2)$$

Тогда  $P(A_m) = \int_{l_{m-1}}^{l_m} dx = P_m$ . Процедура моделирования ис-

пытаний в этом случае состоит в последовательности сравнений случайных чисел  $x_i$  со значениями  $l_k$ . Исходом испытания оказывается событие  $A_m$ , если выполняется условие (4.2). Эту процедуру называют определением исхода по жребию в соответствии с вероятностями  $P_1, P_2, \dots, P_n$ .

При моделировании систем часто необходимо осуществить такие испытания, при которых искомый результат является сложным событием, зависящим от 2-х и более простых.

Пусть например, независимые события  $A$  и  $B$  имеют вероятности наступления  $P_A$  и  $P_B$ . Возможными исходами совместных испытаний в этом случае будут события  $AA, \overline{AB}, A\overline{B}, \overline{AB}$  с вероятностями  $P_AP_B, (1-P_A)P_B, P_A(1-P_B), (1-P_A)(1-P_B)$ . Для моделирования совместных испытаний можно использовать последовательную проверку условия (4.1). Он требует двух чисел  $x_i$ .

Рассмотрим случай, когда события  $A$  и  $B$  являются зависимыми и наступают с вероятностями  $P_A$  и  $P_B$ . Обозначим через  $P(B/A)$  условную вероятность события  $B$  при условии, что событие  $A$  произошло. Считаем, что  $P(B/A)$  задана. Из последовательности случайных чисел  $\{X_i\}$  извлекается определённое число  $x_m$  и проверяется справедливость неравенства  $x_m < P_A$ . Если это неравенство справедливо, то наступило событие  $A$ . Для испытания, связанного с событием  $B$  используется вероятность  $P(B/A)$ . Из совокупности чисел  $\{X_i\}$  берётся очередное число  $x_{m+1}$  и проверяется условие  $x_{m+1} \leq P(B/A)$ . В зависимости от того выполняется или нет это неравенство, исходом испытания является  $AB$  или  $\overline{AB}$ . Если неравенство  $x_m < P_A$  не выполняется, то наступило событие  $\overline{A}$ . Поэтому для испытания, связанного с событием  $B$  необходимо определить вероятность:

$$P(B/\bar{A}) = [P(B) - P(A)P(B/A)] / (1 - P(A)).$$

Выберем из совокупности  $\{X_i\}$  число  $x_{m+1}$  и проверим справедливость неравенства  $x_m \leq P(B/\bar{A})$ . В зависимости от того, выполняется оно или нет, получаем исходы испытания  $\bar{AB}$ . Алгоритм вычислений можно представить в виде схемы, которая изображена на рисунке 4.1.

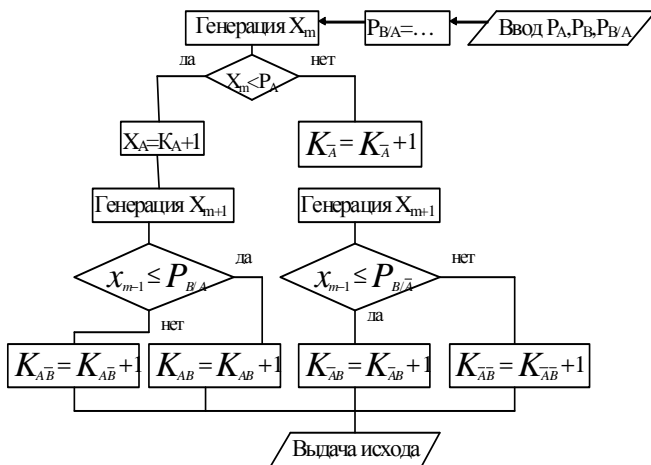


Рис. 4.1 Схема моделирования группы случайных событий

## 4.2. Преобразование случайных величин

Дискретная случайная величина  $\eta$  принимает значения  $y_1 \leq y_2 \leq y_3 \dots y_l$  с вероятностями  $P_1, P_2, \dots, P_l$  составляющими дифференциальное распределение вероятностей:

$$\begin{array}{ccc}
 y & y_1 & y_2 \dots y_j \dots \\
 P(\eta=y) & P_1, & P_2, \dots, P_j, \dots
 \end{array} \quad (4.3)$$

При этом интегральная функция распределения

$$F_\eta(y) = P(\eta \leq y) = \sum_{j=1}^m P_j, \quad y_m \leq y_{m+1}; \quad m=1, 2, \dots$$

$$F_\eta(y) = 0, \quad y < y_1. \quad (4.4)$$

Для получения дискретных случайных величин можно использовать метод обратной функции. Если  $\xi$  – равномерно распределённая на интервале  $(0, 1)$ , случайная величина  $\eta$  получается с помощью преобразования

$$\eta = F_{\eta}^{-1}(\xi), \text{ где } F_{\eta}^{-1} - \text{ функция, обратная } F_{\eta}. \quad (4.5)$$

Алгоритм вычисления по (4.4) и (4.5) сводится к выполнению следующих действий:

$$\begin{aligned} &\text{если } x_1 < P_1 \text{ то } \eta = y_1 \text{ иначе,} \\ &\text{если } x_2 < P_1 + P_2 \text{ то } \eta = y_2 \text{ иначе, } \dots \\ &\dots \text{ если } x_j < \sum_{j=1}^m P_j \text{ то } \eta = y_m \text{ иначе } \dots \end{aligned} \quad (4.6)$$

При счёте по (4.6) среднее число циклов сравнения равняется  $\sum_{j=1}^{\infty} jP_j$ .

### 4.3. Вычисление непрерывных случайных величин

Непрерывная случайная величина  $\eta$  задана интегральной функцией распределения:

$$F_{\eta}(y) = P(\eta \leq y) = \int_{-\infty}^y f_{\eta}(y) dy,$$

где  $f_{\eta}(y)$  – плотность вероятностей.

Для получения непрерывных случайных величин с заданным законом распределения, как и для дискретных величин можно использовать метод обратной функции. Взаимно однозначная монотонная функция  $\eta = F_{\eta}^{-1}(\xi)$ , полученная решением относительно  $\eta$  уравнения  $F_{\eta}(y) = \xi$  преобразует равномерно распределённую на интервале  $(0,1)$  величину  $\xi$  в  $\eta$  с требуемой плотностью  $f_{\eta}(y)$ .

Действительно, если случайная величина  $\eta$  имеет плотность распределения  $f_{\eta}(y)$ , то распределение случайной величины  $\xi = \int_0^{\eta} f_{\eta}(y) dy$  является равномерным.

Таким образом, чтобы получить число, принадлежащее последовательность случайных чисел  $\{y\}$ , имеющих функцию плотности  $f_{\eta}(y)$ , необходимо разделить относительно  $y_i$  уравнение

$$\int_{-\infty}^{y_i} f_{\eta}(y) dy = x_j \quad (4.7)$$

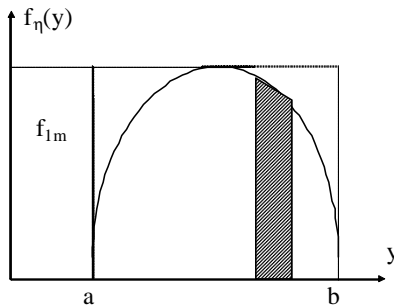
Рассмотрим универсальный метод моделирования непрерывных случайных величин (**метод исключений**).

При моделировании случайной величины  $y$  с плотностью распределения вероятностей  $f_{\eta}(y)$  в интервале  $a \leq y \leq b$  независимые значения  $x_m$  и  $x_{m+1}$  преобразуются в значения

$$y_{1m} = a + (b-a)x_m \quad (4.8)$$

$$z_{1m+1} = f_{1\eta}(y)x_{m+1} \quad (4.9)$$

где  $f_{1\eta}(y) = \max |f_{\eta}(y)|$ . При этом  $y_{1m}$  и  $z_{1m+1}$  – значения случайных величин, равномерно распределенных на интервале  $(a, b)$  и  $(0, f_{1\eta})$ . Эти значения можно рассматривать как абсциссы и ординаты случайных точек, равномерно распределяющихся внутри прямоугольника со сторонами  $b-a$  и  $f_{1\eta}$ , охватывающего кривую распределения  $f_{\eta}(y)$  (см. рис. 4.2).



**Рис. 4.2** Иллюстрация метода исключений

$$\text{Если } z_{1m+1} \leq f_{\eta}(y_{1m}), \quad (4.10)$$

тогда пара  $y_{1m}, z_{1m+1}$  определяет случайную точку под кривой  $f_{\eta}$ . Вероятность попадания случайной точки, удовлетворяющей условию (4.10) под кривую  $f_{\eta}$  равна единице, а вероятность попадания в заштрихованную элементарную площадку равна

$f_{\eta}(y_{1m})\Delta y_{1m}$ . Это обозначает, что абсциссы  $y_{1m}$  случайных точек, попадающих под кривую  $f_{\eta}$  - значения случайной величины  $y$  с заданной плотностью вероятности  $f_{\eta}(y)$ . Моделируемый алгоритм состоит из функций: 1) получения  $x_{m1}$  и  $x_{m+1}$  от датчика; 2) расчёта  $y_{1v}$  и  $z_{1m+1}$  согласно (4.8) и (4.9); 3) вычисления  $f_{\eta}(y_{1m})$ ; 4) сравнения  $z_{1m+1}$  с  $f_{\eta}(y_{1m})$ . Если условия (4.10) выполняются, то  $y=y_{1m}$ ; если нет, то значения  $y_{1m}$  и  $z_{1m+1}$  исключаются и процесс повторяется, начиная с пункта 1. При моделировании системы 2-х случайных величин  $(y_1, y_2)$  с плотностью вероятности  $f(y_1, y_2)$ ,  $a_1 \leq y_1 \leq b_1$ ;  $a_2 \leq y_2 \leq b_2$ , аналогично моделированию одной случайной величины, три значения:  $x_m, x_{m+1}, x_{m+2}$ , выданные датчиком  $E$ , преобразуются в значения:

$$y_{1m} = a_1 + (b_1 - a_1)x_m \quad (4.11)$$

$$y_{2m} = a_2 + (b_2 - a_2)x_{m+1} \quad (4.12)$$

$$z_{3m} = f_{\eta m} x_{m+2}, \text{ где } f_{\eta m} = \max[f(x_1, x_2)] \quad (4.13)$$

Если

$$z_{3m} \leq f(y_{1m}, y_{2m}) \quad (4.14)$$

то  $y_1 = y_{1m}$ ;  $y_2 = y_{2m}$ .

В этом случае случайные точки с координатами  $y_{1m}, y_{2m}, z_{3m}$  – равномерно распределены в пределах параллелепипеда со сторонами, равными  $(b_1 - a_1)(b_2 - a_2)$ ,  $f_{\eta m}$  и условие (4.14) означает попадание точки под поверхность  $f_{\eta}$ . Аналогично моделируется система  $n$  случайных величин  $(y_1, y_2, \dots, y_n)$ .

#### 4.4. Моделирование нормально распределённой случайной величины «у»

Моделирование нормально распределённой случайной величины  $y$  может быть осуществлено на основании центральной предельной теоремы, согласно которой закон распределения суммы независимых случайных величин стремится к нормальному с увеличением числа слагаемых. Для решения некоторых задач практически сумму  $y = \sum_{i=1}^n y_i$  значений, выданных с генератором случайных чисел с характеристиками  $f(x_i) = 1, 0 \leq x_i \leq 1$ ,  $m_{x_i} = 0.5$ ,  $\sigma_{x_i} = 0.5/\sqrt{3}$ .



Можно считать значениями распределённой случайной величины  $y = \sum_{i=1}^n x_i$  при  $n \geq 8$ . Так как все слагаемые  $x_i$  имеют одинаковые математические ожидания  $m_x$  и дисперсии  $D_x$ , то  $m_y = nm_x$ ,  $D_y = nD_x$ . В табл. 4.1 приведены формулы для расчёта случайных величин для различных видов распределений на базе случайной величины с равномерным распределением.

**Таблица 4.1. Формулы для расчёта случайных величин для различных видов распределений на базе случайной величины с равномерным распределением**

Нужное распределение	Плотность распределения	Способ получения случайной величины
Экспоненциальное	$f(y) = \lambda e^{-\lambda(y-\mu)}, \mu \leq y < \infty$	$y' = -\frac{1}{\lambda} \ln(1 - \chi_i) + \mu$
Гамма распределение (целочисленные значения $\eta$ )	$f(y) = \frac{\lambda^\eta}{\Gamma(\eta)} e^{-\lambda y} y^{\eta-1}, 0 < y < \infty$	$y' = -\frac{1}{\lambda} \sum_{i=1}^{\eta} \ln(1 - \chi_i)$
Распределение $\chi^2$	$f(y) = \frac{1}{2^{\gamma/2} \Gamma(\gamma/2)} y^{(\gamma/2)-1} e^{-y/2}, 0 \leq y < \infty$	$y' = \sum_{i=1}^{\gamma} R_N; R_N$ – нормированная случайная величина с нормальным законом распределения
Логарифмические нормальные распределения	$f(y) = \frac{1}{\sqrt{2\pi}\sigma y} \exp\left(-\frac{1}{2} \left[\frac{\ln y - \mu}{\sigma}\right]^2\right), 0 \leq y < \infty$	$y' = e^{\sigma R_N + \mu}$
Вейбулла	$f(y) = \frac{\eta}{\sigma} y^{\eta-1} \exp\left[-\left(\frac{y}{\sigma}\right)^\eta\right], 0 \leq y < \infty$	$y' = -\sigma [\ln(1 - \chi_i)]^{1/\eta}$

#### Выводы по лекции 4:

При моделировании систем часто необходимо осуществить такие испытания, при которых искомый результат явля-

ется сложным событием, зависящим от 2-х и более простых. Для получения непрерывных случайных величин с заданным законом распределения, как и для дискретных величин можно использовать метод обратной функции. Метод исключений является универсальным методом моделирования непрерывных случайных величин. Моделирование нормально распределённой случайной величины  $u$  может быть осуществлено на основании центральной предельной теоремы, согласно которой закон распределения суммы независимых случайных величин стремится к нормальному с увеличением числа слагаемых.

#### **Вопросы для самопроверки по лекции 4:**

- Поясните принцип вычисления непрерывных случайных величин.
- Поясните принципы преобразования случайных величин.
- Охарактеризуйте метод исключений, применяемый для моделирования непрерывных случайных величин.
- Каким образом осуществляется моделирование нормально распределённой случайной величины?
- Приведите формулы для расчёта случайных величин для различных видов распределений на базе случайной величины с равномерным распределением.

## **Лекция 5. МОДЕЛИРОВАНИЕ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ ТИПОВЫХ МАТЕМАТИЧЕСКИХ СХЕМ И АВТОМАТИЗИРОВАННЫХ ПРОГРАММ**

### **Цель лекции**

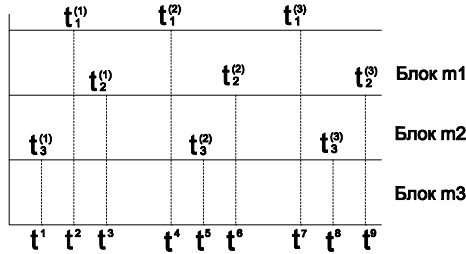
В лекции будут рассмотрены блочные иерархические модели процессов функционирования систем, особенности реализации процессов с использованием Q-схем, применение автоматизированных систем имитационного моделирования, а также наиболее популярные системы моделирования имитационного моделирования. Более подробно будет рассмотрен пакет OPNET Modeler.

### **5.1. Блочные иерархические модели процессов функционирования систем**

Рассмотрим машинную модель  $M_m$ , системы  $S$  как совокупность блоков  $\{m_i\}$ ,  $i=1,2\dots n$ . Каждый блок модели можно охарактеризовать конечным набором возможных состояний  $\{Z_0\}$ , в которых он может находиться. Пусть в течение рассматриваемого интервала времени  $(0, T)$  блок  $i$  изменяет состояние в моменты времени  $t_j^i \leq T$ , где  $j$  – номер момента времени. Момент времени можно разделить на три группы:

- случайные, связанные с внутренними свойствами блока;
- случайные, связанные с изменением состоянием других блоков, имитирующая воздействие среды  $E$ ;
- детерминированные моменты, связанные с заданным расписанием функционирования блока.

Моментами смены состояний модели  $M_m$  в целом  $t^{(k)} \leq T$  будем считать все моменты изменения блоков  $\{m_i\}$ , рис. 5.1. см. ниже.



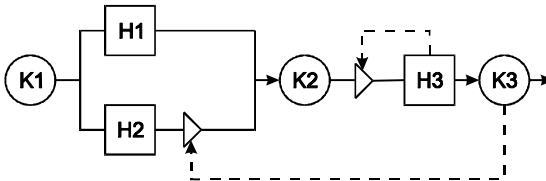
**Рис. 5.1** Смена состояний модели для случаев 3-х блоков

При этом моменты  $t_i^{(j)}$  и  $t^k$  являются моментами системного времени, т.е. времени, в котором функционирует система  $S$ . При машинной реализации модели  $M_m$  её блки представляются соответствующими программными модулями.

## 5.2. Особенности реализации процессов с использованием Q-схем

При моделировании  $Q$ -схем следует адекватно учитывать как связи, отражающие движения заявок (сплошные линии) так и управляющие связи (пунктирные линии).

Рассмотрим фрагмент  $Q$ -схемы (Рис. 5.2.):



**Рис. 5.2** Фрагмент  $Q$ -схемы

Примерами управляющих связей являются различные блокировки обслуживающих каналов (по входу и по выходу): "клапаны" изображены в виде треугольников, а управляющие связи пунктирными линиями. Блокировка канала по входу означает, что этот канал отключается от входящего потока заявок, а блокировка канала по выходу указывает, что заявка обслужена.

ная блокированным каналом, остаётся в этом канале до момента снятия блокировки. В этом случае, если перед накопителем нет "клапана", то при его переполнении будут иметь место потери заявок.

Моделирующий алгоритм должен отвечать следующим требованиям:

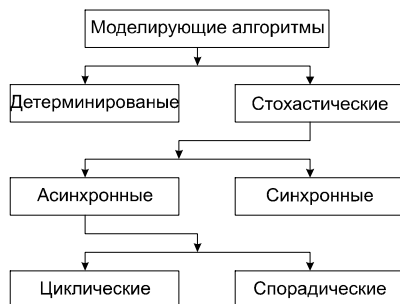
- обладать универсальностью относительно структуры, алгоритмов функционирования и параметров системы  $S$ ;
- обеспечивать одновременную и независимую работу системы  $S$ ;
- укладываться в приемлемые затраты ресурсов ЭВМ (памяти, времени расчёта для реализации машинного эксперимента);
- проводить разбиение на достаточно автономные логические части (блоки);
- гарантировать выполнение рекуррентного правила расчётов.

При этом необходимо иметь виду, что появление одной заявки входящего потока в некоторый момент времени  $t_i$  может вызвать изменение состояния не более чем одного из элементов  $Q$ -схемы, а окончание обслуживания заявки в момент  $t_i$  в некотором канале  $K$  может привести в этот момент времени к последовательному изменению состояний нескольких элементов  $(H, K)$ , т.е. будет иметь место процесс распространения смены состояний в направлении противоположном движению заявки в системе  $S$ . Поэтому просмотр элементов  $Q$ -схемы должен быть противоположным движению заявок.

Все виды моделирующих алгоритмов  $Q$ -схемы можно классифицировать так, как показано на рис. 5.3).

Алгоритмы моделирующие  $Q$ -схему по принципу " $\Delta t$ " являются детерминированными (по шагу), а по принципу особых состояний – стохастические. Последние могут быть реализованы синхронным и асинхронным способами.

При синхронном способе один из элементов  $Q$ -схемы ( $I$ ,  $H$  или  $K$ ) выбирается в качестве ведущего и по нему "синхронизируется" весь процесс моделирования.



**Рис. 5.3** Виды моделирующих алгоритмов  $Q$ -схемы

При асинхронном способе – ведущий (синхронизирующий) элемент не используется, а очередному шагу моделирования (просмотру элементов  $Q$ -схемы) может соответствовать любое особое состояние всего множества элементов  $I$ ,  $H$  и  $K$ . При этом просмотр элементов  $Q$ -схемы организован так, что при каждом особом состоянии либо циклически просматриваются все элементы, спорадически – только те элементы, которые в этом случае могут изменить своё состояние (просмотр с прогнозированием).

### **5.3. Применение автоматизированных систем имитационного моделирования**

Существуют специальные, ориентированные на моделирование вычислительных сетей программные системы, в которых процесс создания модели упрощен. Такие программные системы сами генерируют модель сети на основе исходных данных о ее топологии и используемых протоколах, об интенсивностях потоков запросов между компьютерами сети, протяженности линий связи, о типах используемого оборудования и приложений. Программные системы моделирования могут быть узко специализированными и достаточно универсальными, позволяющие имитировать сети самых различных типов. Качество результатов моделирования в значительной степени зависит от точности исходных данных о сети, переданных в систему имитационного моделирования.

Программные системы моделирования сетей – инструмент, который может пригодиться любому администратору корпоративной сети, особенно при проектировании новой сети или внесении кардинальных изменений в уже существующую. Продукты данной категории позволяют проверить последствия внедрения тех или иных решений еще до оплаты приобретаемого оборудования. Конечно, большинство из этих программных пакетов стоят достаточно дорого, но и возможная экономия может быть тоже весьма ощутимой.

Программы имитационного моделирования сети используют в своей работе информацию о пространственном расположении сети, числе узлов, конфигурации связей, скоростях передачи данных, используемых протоколах и типе оборудования, а также о выполняемых в сети приложениях.

Обычно имитационная модель строится не с нуля. Существуют готовые имитационные модели основных элементов сетей: наиболее распространенных типов маршрутизаторов, каналов связи, методов доступа, протоколов и т.п. Эти модели отдельных элементов сети создаются на основании различных данных: результатов тестовых испытаний реальных устройств, анализа принципов их работы, аналитических соотношений. В результате создается библиотека типовых элементов сети, которые можно настраивать с помощью заранее предусмотренных в моделях параметров.

Системы имитационного моделирования обычно включают также набор средств для подготовки исходных данных об исследуемой сети – предварительной обработки данных о топологии сети и измеренном трафике. Эти средства могут быть полезны, если моделируемая сеть представляет собой вариант существующей сети и имеется возможность провести в ней измерения трафика и других параметров, нужных для моделирования. Кроме того, система снабжается средствами для статистической обработки полученных результатов моделирования.

Основными элементами, из которых состоят системы моделирования сетей, являются библиотеки моделей, среда моделирования, подсистема генерации трафика, подсистема генера-

ции топологии сети, подсистема анализа результатов моделирования.

Библиотеки моделей представляют собой хранилища структурных элементов модели и могут оцениваться по следующим критериям:

- возможность моделировать стандартные сетевые устройства;
- возможность создания моделей устройств, удовлетворяющих требованиям пользователя;
- уровень параметризации элементов библиотеки;
- число классов моделируемых объектов;
- масштабируемость времени моделирования;
- точность и соответствие моделей реальным объектам.

Среда моделирования используется для сбора данных о функционировании модели. Ее организация редко раскрывается разработчиком. Поэтому организацию функционирования среды прогона нельзя использовать как критерий сравнения.

Графический интерфейс пользователя представляет собой модуль для взаимодействия с подсистемами генерации трафика и топологии сети. Он должен обеспечивать максимальное удобство для пользователя и реализовывать функции управления средой моделирования, в числе которых: функция drag-and-drop; анимация процесса моделирования работы сети; возможность приостанавливать или прерывать процесс моделирования, повторный запуск и пошаговый анализ моделирования; обеспечение наглядности элементов рабочего пространства; задание параметров генерации трафика в удобной форме; возможность иерархического представления элементов сети.

Подсистема анализа результатов моделирования обрабатывает данные, собранные при прогоне модели, вычисляет характеристики производительности и представляет результаты в удобной для пользователя форме. В значительной степени возможность этой подсистемы зависит от тех данных, которые собирает среда прогона. Определяющими для этой части системы являются следующие моменты: количество и тип характеристик,



собираемых в результате работы модели, а также различные виды представления результирующей информации.

Одним из плюсов создания модели сети с помощью программного обеспечения является то, что уровень гибкости, обеспечиваемый ядром моделирования, тот же, что и для моделирования, написанных с нуля, но объектное построение среды позволяет пользователю намного быстрее делать разработку, усовершенствования и производить модели для многократного использования.

#### **5.4. Наиболее популярные системы моделирования имитационного моделирования**

Существует множество программных средств для имитационного моделирования, каждое из которых дает возможность изучения работы сети в большей или меньшей степени.

*BONeS* (фирма Systems and Networks) – графическая система моделирования общего назначения для анализа архитектуры систем, сетей и протоколов. Описывает модели на транспортном уровне и на уровне приложений. Дает возможность анализа воздействия приложений типа клиент - сервер и новых технологий на работу сети.

*Netmaker* (фирма OPNET Technologies) – проектирование топологии, средства планирования и анализа сетей широкого класса. Состоит из различных модулей для расчета, анализа, проектирования, визуализации, планирования и анализа результатов.

*Optimal Perfomance* (фирма Compuware; Optimal Networks) – имеет возможности быстрого оценочного и точного моделирования, помогает оптимизировать распределенное программное обеспечение.

*Prophesy* (компания Abstraction Software) – простая система для моделирования локальных и глобальных сетей. Позволяет оценить время реакции компьютера на запрос, количество "хитов" на WWW-сервере, количество рабочих станций для обслуживания активного оборудования, запас производительности сети при поломке определенного оборудования.

*Семейство CANE* (компания ImageNet) – проектирование и реинжиниринг вычислительной системы, оценка различных вариантов, сценарии "что, если". Моделирование на различных уровнях модели OSI. Развитая библиотека устройств, которая включает физические, электрические, температурные и другие характеристики объектов. Возможно создание своих библиотек.

*Семейство COMNET* (фирма Compuware; CACI Products Company) – объектно-ориентированная система моделирования локальных и глобальных сетей. Позволяет моделировать уровни: приложений, транспортный, сетевой, канальный. Использует все известные на сегодня технологии и протоколы, а также системы клиент-сервер. Легко настраивается на модель оборудования и технологий. Возможность импорта и экспорта данных о топологии и сетевом трафике. Моделирование иерархических сетей, многопротокольных локальных и глобальных сетей; учет алгоритмов маршрутизации.

*Семейство OPNET* (фирма OPNET Technologies) – средство для проектирования и моделирования локальных и глобальных сетей, компьютерных систем, приложений и распределенных систем. Возможность импорта и экспорта данных о топологии и сетевом трафике. Анализ воздействия приложений типа клиент-сервер и новых технологий на работу сети. Моделирование иерархических сетей, многопротокольных локальных и глобальных сетей; учет алгоритмов маршрутизации. Объектно-ориентированный подход. Исчерпывающая библиотека протоколов и объектов. Включает следующие продукты: Netbiz (проектирование и оптимизация вычислительной системы), Modeler (моделирование и анализ производительности сетей, компьютерных систем, приложений и распределенных систем), ITGuru (оценка производительности коммуникационных сетей и распределенных систем).

*Stressmagic* (фирма NetMagic Systems) – поддержка стандартных тестов измерения производительности; имитация пиковой нагрузки на файл-сервер и сервер печати. Возможно моделирование взаимодействия различных пользователей с файл-сервером. Включает 87 тестов производительности.

Более подробные сведения об этих системах и их характеристиках приведены в табл. 5.1. К числу наиболее мощных и интересных относится OPNET фирмы OPNET Technologies (ранее называлась MIL3).

**Таблица 5.1. Характеристики систем имитационного моделирования**

Компания	Продукт	Стоимость, долл.	Тип сети	Операционная система
Systems and Networks	Bones	20000 - 40000	LAN, WAN, клиент-серверные архитектуры	Sun Solaris, Sun OS, HP/UX
ImageNet	CANE	7900 - 25000	LAN, WAN, клиент-серверные архитектуры	Windows NT
Optimal Networks	Optimal Performance	5000 - 30000	LAN, WAN	Windows 98/NTs
Abstraction Software	Prophesy	599	LAN, WAN	Windows 98/NT, OS/2
Network Analysis Center ( <a href="http://www.nacmind.com/">http://www.nacmind.com/</a> , <a href="http://www.salestar.com/">http://www.salestar.com/</a> )	WinMIND	9500 - 41000	WAN	Windows 98/NT
CACI Products (Compuware)	Семейство COMNET	19000 - 60000	LAN, WAN, клиент-серверные архитектуры	Windows 98/NT, OS/2, AT&T Unix, IBM AIX, DEC Ultrix, Sun Solaris, Sun OS, HP/UX
OPNET Technologies (MIL3)	Семейство OPNET	16000 - 40000	LAN, WAN, клиент-серверные архитектуры	DEC AXP, Sun Solaris, Sun OS, HP/UX, Silicon Graphics IRIX, IBM AIX, Windows
NetMagic Systems	StressMagics	3000 на 1 файл-сервер	LAN	Windows 98/NT

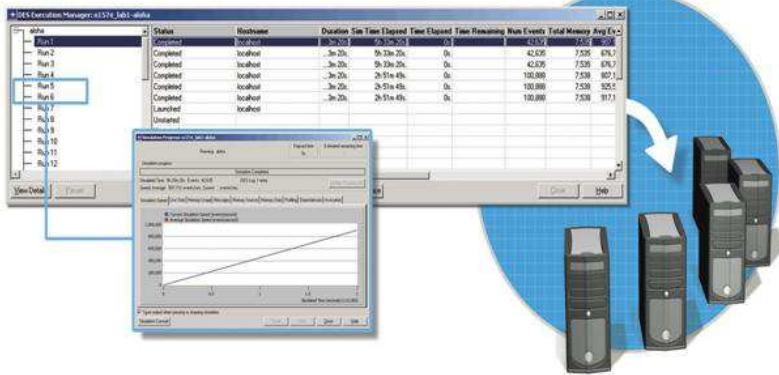
## 5.5. Система моделирования OPNET Modeler

Наиболее распространенной и эффективной средой имитационного моделирования сетей является OPNET Modeler. OPNET Modeler – это гибкий и наглядный инструмент, один из лучших сетевых симуляторов не только для обучения, но и для разработок и исследований. OPNET используют более 500 различных компаний по всему миру – провайдеры Интернет-услуг, правительственные организации, производители телекоммуникационного оборудования, ВУЗы.

OPNET Modeler предлагает пользователям графическую среду для создания, выполнения и анализа событийного моделирования сетей связи. Это удобное программное обеспечение может быть использовано для большого ряда задач, например, типичные создание и проверка протокола связи, анализ взаимодействий протокола, оптимизация и планирование сети. Также возможно осуществить с помощью пакета проверку правильности аналитических моделей, и описание протоколов.

Основные характеристики следующие:

- самый быстрый механизм дискретного моделирования среди существующих;
- сотни моделей протоколов и сетевых устройств ведущих производителей;
- объектно-ориентированное моделирование;
- иерархическая структура оболочки моделирования;
- дискретное, гибридное и аналитическое (опция) моделирование;
- 32-bit и 64-bit ядро параллельного моделирования;
- поддержка распределенного моделирования (рис. 5.4);
- дополнительно, специальные возможности по взаимодействию с «живыми» системами;
- открытый интерфейс для интеграции внешних объектов, библиотек и других систем моделирования;
- встроенные графические средства отладки и анализа.



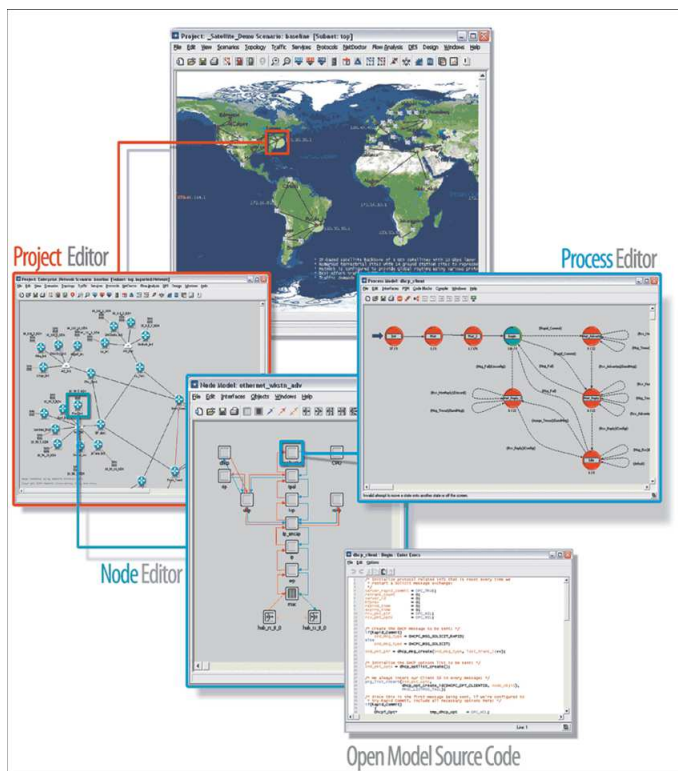
**Рис. 5.4. Поддержка распределенного моделирования в среде OPNET**

Рассматриваемый продукт содержит несколько редакторов для разработки библиотек моделируемых систем (рис. 5.5). Организация объектов - иерархическая, сетевые объекты (модели) связаны набором узлов и объектов связи, в то время как объекты узла связаны набором объектов, типа модулей очередности, модулей процессора, передатчиков и приемников. Логику поведения процессора и модулей очередности определяет модель процесса, которую пользователь может создавать и изменять в пределах редактора процесса. В редакторе процесса пользователь может определить модель процесса через комбинацию алгоритма работы конечного автомата (finite-state machine – FSM) и операторов языка программирования C/C++.

Вызов события модели процесса в течение моделирования управляется возбуждением прерывания, а каждое прерывание соответствует событию, которое должно быть обработано моделью процесса.

Основа связи между процессами – структура данных, называемая пакетом. Могут быть заданы форматы пакета, то есть они определяют, какие поля могут содержать такие стандартные типы данных, как целые числа, числа с плавающей запятой и указатели на пакеты (эта последняя способность позволяет инкапсулировать моделирование пакета). Структура дан-

ных, вызывающая информацию по контролю за интерфейсом (interface control information - ICI), может быть разделена между двумя событиями моделей процесса - это ещё один механизм для межпроцессорной связи, это очень удобно для команд моделирования и соответствует архитектуре многоуровневого протокола. Процесс также может динамически порождать дочерние процессы, которые упростят функциональное описание таких систем, как серверы.



**Рис. 5.5. Иерархическая организация редакторов в среде OPNET**

Несколько основных моделей процесса входят в базовую комплектацию пакета, моделируя популярные протоколы рабо-

ты с сетями и алгоритмы, вроде протокола шлюза границы (border gateway protocol - BGP), протокола контроля передачи, интернет протокол (TCP/IP), ретрансляции кадров (frame relay), Ethernet, асинхронного режима передачи (asynchronous transfer mode -ATM), и WFQ (weighted fair queuing). Базовые модели полезны для быстрого развития сложных имитационных моделей для общих архитектур сети. Существует возможность сопровождения комментариями и графикой (с поддержкой гипертекста) моделей сети, узла или процесса.

### **Выводы по лекции 5:**

При моделировании  $Q$ -схем следует адекватно учитывать как связи, отражающие движения заявок так и управляющие связи. Существуют специальные, ориентированные на моделирование вычислительных сетей программные системы, в которых процесс создания модели упрощен – автоматизированные системы имитационного моделирования. Наиболее распространенной и эффективной средой имитационного моделирования сетей является OPNET Modeler.

### **Вопросы для самопроверки по лекции 5:**

- Приведите примеры управляющих связей.
- Каким требованиям должен удовлетворять моделирующий алгоритм?
- Приведите классификацию моделирующих алгоритмов.
- Для чего нужны системы имитационного моделирования?
- По каким критериям оцениваются библиотеки моделей?
- Какие задачи решает подсистема анализа результатов моделирования системы?
- Перечислите наиболее популярные системы моделирования и дайте их краткую характеристику.
- В чем преимущества системы OPNET Modeler.

## Список обозначений и сокращений

<b>ATM</b>	(asynchronous transfer mode) – асинхронный режим передачи
<b>BGP</b>	(Border Gateway Protocol) – пограничный протокол маршрутизации
<b>FSM</b>	(Finite-state machine) – конечный автомат
<b>ICI</b>	(Interface control information) – информация по контролю за интерфейсом
<b>LAN</b>	(Local Area Network) – локальная вычислительная сеть
<b>TCP/IP</b>	стек протоколов
<b>WAN</b>	(Wide Area Network) – территориально распределенная сеть
<b>WFQ</b>	(Weighted Fair Queuing) – взвешенное справедливое обслуживание
<b>АЦПУ</b>	Алфавитно-цифровое печатающее устройство
<b>ВС</b>	Вычислительная система
<b>ИМ</b>	Имитационная модель
<b>ММ</b>	Математическая модель
<b>ОЗУ</b>	Оперативное запоминающее устройство
<b>ОПС</b>	Однородный поток событий
<b>ПС</b>	Поток событий
<b>СМО</b>	Система массового обслуживания
<b>ТА</b>	Теория автоматов
<b>ЭВМ</b>	Электронно-вычислительная машина



## Список литературы

1. Зарубин, В.С. Математическое моделирование в технике: Учеб. для вузов / Под ред. В.С. Зарубина, А.П. Крищенко. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. – 496 с.
2. Советов, Б.Я., Яковлев С.А. Моделирование систем: Учеб.для вузов/ Б. Я. Советов, С. А. Яковлев. - 3-е издание, переработанное и дополненное - М.: Высш.школа, 2001 – 341 с.: ил.
3. Советов, Б.Я. Моделирование систем: Практикум : Учебное пособие для вузов / Б. Я. Советов, С. А. Яковлев. - 2-е издание, переработанное и дополненное. – М.: Высшая школа, 2003. - 295с.
4. Бочаров, П.П. Теория массового обслуживания / П.П. Бочаров, А.В Печинкин. – М., Изд-во. РУДН, 1995. – 529 с.
5. Башарин, Г.П. Лекции по математической теории телеграфика / Г.П. Башарин – М, Изд-во РУДН. 2007. – 270 с.
6. Рыжиков, Ю.И. Имитационное моделирование / Ю.И. Рыжиков. – СПб., Изд-во. КОРОНА принт, М., Изд-во. Альтекс-А., 2004. – 380 с.
7. Апанасович, В.В. Цифровое моделирование стохастических систем / В.В. Апанасович, О.М. Тихоненко. – Минск, Изд-во «Университетское», 1986. – 127 с.
8. Бражник, А.Н. Имитационное моделирование: возможности GPSS World / А.Н. Бражник. – СПб., Изд-во Реноме, 2006 г. – т 440 стр.
9. Кудрявцев, Е. М. GPSS World. Основы имитационного моделирования различных систем / Е.М. Кудрявцев. – Изд-во ДМК пресс, 2004. – 320 стр.

## Глоссарий

**Имитационное моделирование** – Это метод исследования, который основан на том, что анализируемая динамическая система заменяется имитатором и с ним производятся эксперименты для получения об изучаемой системе.

**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 её функционирования в виде некоторой математической модели.

**Метод имитационного моделирования** – Заключается в создании логико-аналитической (математической модели системы и внешних воздействий), имитации функционирования системы, т.е. в определении временных изменений состояния системы под влиянием внешних воздействий и в получении выборок значений выходных параметров, по которым определяются их основные вероятностные характеристики.

**Моделирование** – Замещение одного объекта (оригинала) другим (моделью) и фиксация или изучение свойств оригинала путем исследования свойств модели.

**Модель объекта** – Это физическая или абстрактная система, адекватно представляющая объект исследования. В теории моделирования используются преимущественно абстрактные модели – описания объекта исследования на некотором языке. Абстрактность модели проявляется в том, что компонентами модели являются не физические элементы, а понятия, в качестве которых наиболее широко используются математические. Абстрактная модель, представленная на языке математических отношений, называется математической моделью.

**Потоко событий** – Последовательность событий, происходящих одно за другим в какие-то случайные моменты времени.

**Протокол** (сетевой протокол) – Набор правил, позволяющий осуществлять соединение и обмен данными между двумя включёнными в сеть компьютерами; набор правил, описывающих

формат и назначение пакетов, которыми обмениваются одно-ранговые сущности.

**Сеть связи** – Совокупность линий связи и промежуточного оборудования/промежуточных узлов, терминалов/оконечных узлов, предназначенных для передачи информации от отправителя до получателя с заданными параметрами качества обслуживания.

**Стратификация** – Это выбор уровня детализации модели.

**Теория моделирования** – Представляет собой взаимосвязанную совокупность положений, определений, методов и средств создания и изучения моделей. Эти положения, определения, методы и средства, как и сами модели, являются предметом теории моделирования.