Министерство образования Республики Беларусь

Учреждение образования

Белорусский государственный университет информатики и радиоэлектроники

Факультет непрерывного и дистанционного обучения

Кафедра информатики

Электронный учебно-методический комплекс

по дисциплине

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Для студентов специальности

1-31 03 04 Информатика

Минск 2011

# Общие сведения

## Сведения об ЭУМК

Электронный учебно-методический комплекс по дисциплине «Математическое моделирование» предназначен для студентов специальности «Информатика», а также может быть использован преподавателями, аспирантами и практическими работниками предприятий.

Рабочая учебная программа составлена на основе учебной программы «Математическое моделирование», утвержденной ректором БГУИР 02. 03. 2010 регистрационный № УД-31-156/уч и учебного плана специальности 1-31 03 04 Информатика

**Составитель:**

**Н.А. Волорова,** доцент кафедры информатики Учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук.

Рассмотрен и рекомендован к изданию на заседании кафедры информатики, протокол № 2 от 12.09.2011.

Одобрен и рекомендован к изданию методической комиссией факультета компьютерных систем и сетей, протокол № \_\_ от \_\_.\_\_.2011.

## Методические рекомендации по изучению дисциплины

В соответствии с учебным планом студенты дистанционной формы обучения экономических специальностей изучают курс «Математическое моделирование».

Учебным планом по предусмотрено изучение теоретических вопросов, практические задачи по наиболее актуальным темам, выполнение одной контрольной работы и 2-х ИПР с ИКТ. Изучение курса заканчивается зачетом. К сдаче зачета студенты допускаются только при условии выполненных и защищенных контрольных работ ИПР с ИКТ.

Рекомендуется изучать курс «Математическое моделирование» в соответствии с рабочей программой. Сначала необходимо ознакомиться с содержанием курса, затем изучить рекомендуемую литературу, обращая внимание на вопросы, выделенные в рабочей программе, после чего изучить теоретическое изложение курса по приведенным разделам, темам и вопросам, ответить на контрольные вопросы, выполнить задачи для решения (выполнения контрольных работ) и задания ИПР с ИКТ в соответствии с вариантом.

Так как теоретический материал излагается в строгой логической последовательности, рекомендуется изучать данную дисциплину, придерживаясь данной логики.

## Рабочая учебная программа

**Учреждение образования**

**«Белорусский государственный университет**

**информатики и радиоэлектроники»**

УТВЕРЖДАЮ

Декан факультета непрерывного и дистанционного обучения

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. М. Бондарик

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2010 г.

Регистрационный № УД-11-23-\_\_\_/р.

**МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ**

Рабочая учебная программа

для направления специальности 1-31 03 04

**Информатика**

Факультет **непрерывного и дистанционного обучения**

Кафедра **информатики**

Курс **четверный**

Контрольные работы **1 работа**

ИПР с ИКТ **2 работы**

Всего часов **50 часов**

зачет **4 курс**

Форма получения

высшего образования **дистанционная**

Минск 2011

Рабочая учебная программа составлена на основе учебной программы «Математическое моделирование», утвержденной ректором БГУИР 02. 03. 2010 регистрационный № УД-31-156/уч и учебного плана специальности 1-31 03 04 Информатика

**Составитель:**

**Н.А.Волорова,** доцент кафедры информатики Учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук.

Рассмотрена и рекомендована к утверждению на заседании кафедры информатики, протокол № 18 от « 03 » мая 2010 г.

Заведующий кафедрой Минченко Л.И.

Одобрена и рекомендована к утверждению Советом факультета компьютерных систем и сетей Учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»,

протокол № 15 от « 24 » мая 20010 г.

Председатель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.М.Лукашевич

СОГЛАСОВАНО

Начальник отдела

методического обеспечения

учебного процесса \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ц. С. Шикова

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ**

**ПО ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ**

**С ДРУГИМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

|  |  |  |  |
| --- | --- | --- | --- |
| Название дисциплины,  с которой требуется согласование | Кафедра, обеспечивающая изучение этой дисциплины | Предложения об изменениях в содержании учебной программы по изучаемой дисциплине | Решение, принятое кафедрой, разработавшей учебную программу (с ука- занием даты и но- мера протокола)1 |
| 1 | 2 | 3 | 4 |
| Программирование | информатики | нет | Согласовано |
| Архитектура компьютера | информатики | нет | Согласовано |
| Теория вероятностей и математическая статистика | информатики | нет | Согласовано |
| Дискретная математика и математическая логика | информатики | нет | Согласовано |

СОГЛАСОВАНО:

Зав. кафедрой Информатики Л.И.Минченко

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**Цель преподавания:** изучение принципов системного анализа и методов аналитического и имитационного моделирования для решения задач анализа и синтеза сложных систем. Основное внимание уделяется моделированию сложных вычислительных систем, их аппаратного и программного обеспечения.

**Задачи изучения дисциплины**: изучение системного подхода в исследовании сложных систем, обучение методологии моделирования как эффективного инструмента системного анализа.

В результате изучения дисциплины студенты должны:

**знать:**

* типовые математические схемы моделирования;
* технологию имитационного моделирования на ЭВМ;
* перспективы развития методов математического моделирования в задачах анализа и синтеза сложных систем, проектирования ВС и их программного обеспечения,

**уметь:**

* анализировать основные особенности вычислительных сис­тем, выбирать адекватные типам ВС модели, уметь разрабатывать модели сложных систем.

**иметь представление:**

* системах имитационного математического моделирования сложных систем, современных методах и средствах имитационного и статистического моделирования.

Перечень дисциплин, усвоение которых необходимо для изучения данной дисциплины

|  |  |  |
| --- | --- | --- |
| № пп | Название дисциплины | Раздел, тема |
| 1 | Программирование | Все разделы дисциплины |
| 2 | Архитектура компьютера | Все разделы дисциплины |
| 3 | Теория вероятностей и математическая статистика | Все разделы дисциплины |
| 4 | Дискретная математика и математическая логика | Все разделы дисциплины |

**СОДЕРЖАНИЕ ДИСЦИПЛИНЫ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Номер | | Название тем | Контрольная работа | ИПР с ИКТ | Литература  (номера) | Рекомендуемый объём для изучения  (в часах) | Форма контроля знаний |
| Недели | Темы |

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Четвертый курс** | | | | | | | |
| 1-2 | 1 | Понятие сложной системы. Задачи исследования сложных систем. Классификация методов моделирования. Роль моделирования в процессе принятия решений. |  |  | 1,4 | 6 |  |
| 3-4 | 2 | Формализация процессов функционирования сложных систем. Математические схемы. Непрерывно-детерминированные модели (D-схемы). Дискретно-детерминированные модели (f-схемы). Дискретно-стохастические модели (Р-схемы). Непрерывно-стохастические модели (q-схемы). Обобщенные модели (А-схемы) |  |  | 1,4 | 6 |  |
| 5-6 | 3 | Понятие СМО. Потоки событий. Марковские процессы в СМО. СМО с отказами СМО с ожиданием. Построение и реализация алгоритмов моделирования систем массового обслуживания. |  |  | 4,5 | 7 |  |
| 7-8 | 4 | Понятие статистического эксперимента. Область применения и классификация имитационных моделей. Описание поведения системы. |  | 1 | 2,3,6 | 6 | Зачёт по ИПР с ИКТ |
| 9-10 | 5 | Виды параллельных процессов в сложных системах. Методы описания параллельных процессов в системах и языках моделирования. Применение сетевых моделей для описания параллельных процессов |  |  | 1,4, | 6 |  |
| 11-12 | 6 | Датчики базовой случайной величины (БСВ). Построение датчиков БСВ. Характеристики датчиков базовых случайных величин. Тестирование БСВ. Имитация случайных событий. Имитация случайных величин. Алгоритмы получения значений систем случайных величин (случайных векторов). |  | 2 | 2,3,6 | 6 | Зачёт по ИПР с ИКТ |
| 13-14 | 7 | Планирование модельных экспериментов. Стратегическое планирование эксперимента. Тактическое планирование эксперимента |  |  | 2,3,6 | 6 |  |
| 15-16 | 8 | Оценка качества имитационной модели. Оценка адекватности модели. Оценка устойчивости модели. Оценка чувствительности ИМ. Калибровка модели. Оценка влияния и взаимосвязи факторов | 1 |  | 2,3,6 | 7 | Защита контрольной работы |
| 17 |  |  |  |  |  |  | зачет |
|  | | | | | | |  |

**1. Наименование тем, их содержание**

**Тема 1. Основные понятия теории моделирования.**

Понятие сложной системы. Задачи исследования сложных систем. Классификация методов моделирования. Роль моделирования в процессе принятия решений.

**Р.Л.: [**1,4**].**

**Тема 2. Математические модели.**

Формализация процессов функционирования сложных систем. Математические схемы. Непрерывно-детерминированные модели (D-схемы). Дискретно-детерминированные модели (f-схемы). Дискретно-стохастические модели (Р-схемы). Непрерывно-стохастические модели (q-схемы). Обобщенные модели (А-схемы)

**Р.Л.: [**1,4**].**

**Тема 3. Моделирование систем массового обслуживания.**

Понятие СМО. Потоки событий. Марковские процессы в СМО. СМО с отказами СМО с ожиданием. Построение и реализация алгоритмов моделирования систем массового обслуживания.

**Р.Л.: [**4,5**].**

**Тема 4. Принципы имитационного моделирования.**

Понятие статистического эксперимента. Область применения и классификация имитационных моделей. Описание поведения системы..

**Р.Л.: [**2,3,6**].**

**Тема 5. Разработка имитационных моделей.**

Управление модельным временем. Моделирование параллельных процессов. Методы описания параллельных процессов в системах и языках моделирования. Сетевые модели для описания параллельных процессов.

**Р.Л.: [**1,4**].**

**Тема 6. Моделирование случайных факторов**

Датчики базовой случайной величины (БСВ). Построение датчиков БСВ. Характеристики датчиков базовых случайных величин. Тестирование БСВ. Имитация случайных событий. Имитация случайных величин. Алгоритмы получения значений систем случайных величин (случайных векторов).

**Р.Л.: [**2,3,6**].**

**Тема 7. Планирование модельных экспериментов.**

Стратегическое планирование эксперимента. Тактическое планирование эксперимента

**Р.Л.: [**2,3,6**].**

**Тема 8. Обработка и анализ результатов моделирования.**

Оценка качества имитационной модели. Оценка адекватности модели. Оценка устойчивости модели. Оценка чувствительности ИМ. Калибровка модели. Оценка влияния и взаимосвязи факторов

**Р.Л.: [**2,3,6**].**

**2. Контрольные работы, их характеристика**

|  |  |  |  |
| --- | --- | --- | --- |
| №  п./п. | Название темы | Характеристика | Объём в часах |
|  | Построение имитационных моделей | В ходе выполнения работы изучаются принципы имитационного моделирования на примере многоканальной многофазной системы массового обслуживания.  В ходе выполнения работы на языке высокого уровня реализуется имитационная модель, исследуются полученные характеристики (8 часов) | 36 |
| **Итого** | | | **8** |

**3. ИПР с ИКТ, их характеристика**

|  |  |  |  |
| --- | --- | --- | --- |
| №  п./п. | Название темы | Характеристика | Объём в часах |
| 1. | Построение и исследование характеристик датчиков базовых случайных величин | В ходе выполнения работы на языке высокого уровня реализуются 2 алгоритма формирования БСВ и проводиться их сравнительный анализ | 4 |
|  | Имитация непрерывных случайных величин (метод обратных функций) | В ходе выполнения работы на языке высокого уровня реализуется алгоритм обратных функций и проводиться анализ характеристик воспроизводимой СВ | 4 |
| **Итого** | | | **8** |

**4. ЛИТЕРАТУРА**

*Основная*

* + 1. Советов, Б.Я.Моделирование систем: Учебник для ВУЗов. / Б. Я. Советов, С. А. Яковлев -М.:Высшая школа,2005.-344с.:ил.
    2. Харин Ю.С. и др. Основы имитационного и статистического моделирования. – Мн.: Дизайн ПРО, 1997, 287 с.
    3. Кельтон В.Д., Лоу А.М. Имитационное моделирование. Классика CS. 3-е изд. - СПб.: Питер; Киев: Издательская группа BHV, 2004. - 847 с.: ил.
    4. Бусленко Н.П. Моделирование сложных систем. Главная редакция физико-математической литературы.-М: «Наука», 1968.
    5. Гнеденко, Б. П. Введение в теорию массового обслуживания. [/ Б. П. Гнеденко, И. Н. Коваленко. М.: Наука. 1987.
    6. Максимей И.В. Имитационное моделирование на ЭВМ. – М.: Радио и связь, 1988.

*Дополнительная*

* + 1. Вентцель Е.С., Овчаров Л.А. Теория случайных процессов и ее инженерные приложения. - М: Наука, 1991.
    2. Четвериков В.Н., Баканович Э.А. Стохастические вычислительные устройства систем моделирования. – М.: Машиностроение, 1989.
    3. Советов, Б. Я. Моделирование систем: Лабораторный практикум. / Б. Я. Советов, С. А. Яковлев. М.: Высшая школа, 1999 г.

**4. ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ, НАГЛЯДНЫХ И ДРУГИХ ПОСОБИЙ, МЕТОДИЧЕСКИХ УКАЗАНИЙ И МАТЕРИАЛОВ И ТЕХНИЧЕСКИХ СРЕДСТВ ОБУЧЕНИЯ**

1. Среда разработки программ на языке С++

2. Программные пакеты по математике MATLAB, MATHCAD.

3. ПЭВМ

# Теоретический раздел

## ЛЕКЦИИ

### ВВЕДЕНИЕ

При создании и проектировании сложных систем возникают многочисленные задачи, требующие знаний количественных и качественных закономерностей, свойственных рассматриваемым системам. Особенное значение приобрели так называемые *общесистемные* вопросы, относящиеся к общей структуре системы, организации взаимодействия между ее элементами, совокупному взаимодействию элементов с внешней средой, централизованному управлению функционированием элементов и т.д. Эти вопросы составляют существо так называемого системного подхода к изучению свойств реальных объектов и содержание направления, получившего название *системотехника*.

Наиболее полное и всестороннее исследование сложной системы на всех этапах разработки, начиная с этапа постановки задачи, подготовки технического задания и заканчивая внедрением системы в эксплуатацию, невозможно без методов моделирования. Именно моделирование является средством, позволяющим без капитальных затрат решить проблемы построения больших систем.

***Модель — это объект-заместитель объекта-оригинала, обеспечивающий изучение некоторых свойств оригинала.***

Замещение одного объекта другим с целью получения информации о важнейших свойствах объекта-оригинала с помощью объекта-модели называется *моделированием.*

***Моделирование может быть определено как представление объекта моделью для получения информации об этом объекте путем проведения экспериментов с его моделью.*** Теория замещения одних объектов (оригиналов) другими объектами (моделями) и исследования свойств объектов на их моделях называется***теорией моделирования****.* Если результаты моделирования подтверждаются и могут служить основой для прогнозирования процессов, протекающих в исследуемых объектах, то говорят, что модель адекватна объекту. При этом ***адекватность*** модели зависит от цели моделирования и принятых критериев.

### Тема 1. ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МОДЕЛИРОВАНИЯ.

#### 1.1. Понятие сложной системы.

Сложная система, составной объект, части которого можно рассматривать как системы, закономерно объединённые в единое целое в соответствии с определенными принципами или связанные между собой заданными отношениями.

Понятием Сложная система пользуются в системотехнике, системном анализе, операций исследовании и при системном подходе в различных областях науки, техники и народный хозяйства. Сложную систему можно расчленить (не обязательно единственным образом) на конечное число частей, называемое подсистемами; каждую такую подсистему (высшего уровня) можно в свою очередь расчленить на конечное число более мелких подсистем и т. д., вплоть до получения подсистем первого уровня, т. н. *элементов*.

Подсистема, т. о., с одной стороны, сама является Сложной системой из нескольких элементов (подсистем низшего уровня), а с другой стороны - элементом системы старшего уровня.

В каждый момент времени элемент Сложная система находится в одном из возможных состояний; из одного состояния в другое он переходит под действием внешних и внутренних факторов.

#### 1.2.Задачи исследования сложных систем.

Одной из важных проблем в области разработки и создания современных сложных технических систем является исследование динамики их функционирования на различных этапах проектирования, испытания и эксплуатации. При исследовании сложных систем возникают задачи исследования как отдельных видов оборудования и аппаратуры, входящих в систему, так и системы в целом.

При проектировании сложных систем ставится задача разработки систем, удовлетворяющих заданным техническим характеристикам. Поставленная задача может быть решена одним из следующих методов:

* методом синтеза оптимальной структуры системы с заданными характеристиками;
* методом анализа различных вариантов структуры системы для обеспечения требуемых технических характеристик.

Оптимальный синтез систем в большинстве случаев практически невозможен в силу сложности поставленной задачи и несовершенства современных методов синтеза сложных систем. Методы анализа сложных систем, включающие в себя элементы синтеза, в настоящее время достаточно развиты и получили широкое распространение.

Любая синтезированная или определенная каким-либо другим образом структура сложной системы для оценки ее показателей должна быть подвергнута испытаниям. Проведение испытаний системы является задачей анализа ее характеристик. Таким образом, конечным этапом проектирования сложной системы, осуществленного как методом синтеза структуры, так и методом анализа вариантов структур, является анализ показателей эффективности проектируемой системы.

Среди известных методов анализа показателей эффективности систем и исследования динамики их функционирования следует отметить:

* аналитический метод;
* метод натуральных испытаний;
* метод полунатурального моделирования;
* моделирование процесса функционирования системы на ЭВМ.

Строгое аналитическое исследование процесса функционирования сложных систем практически невозможно. Определение аналитической модели сложной системы затрудняется множеством условий, определяемых особенностями работы системы, взаимодействием ее составляющих частей, влиянием внешней среды и т.п.

Натуральные испытания сложных систем связаны с большими затратами времени и средств. Проведение испытаний предполагает наличие готового образца системы или ее физической модели, что исключает или затрудняет использование этого метода на этапе проектирования системы.

Широкое применение для исследования характеристик сложных систем находит метод полунатурального моделирования. При этом используется часть реальных устройств системы. Включенная в такую полунатуральную модель ЭВМ имитирует работы остальных устройств системы, отображенных математическими моделями. Однако в большинстве случаев этот метод также связан со значительными затратами и трудностями, в частности, аппаратной стыковкой натуральных частей с ЭВМ.

Исследование функционирования сложных систем с помощью моделирования их работы на ЭВМ помогает сократить время и средства на разработку.

Затраты рабочего времени и материальных средств на реализацию модели оказываются незначительными по сравнению с затратами, связанными с натурным экспериментом. Результаты моделирования по своей ценности для практического решения задач часто близки к результатам натурного эксперимента.

Основной метод исследования сложных систем -- математическое моделирование, в том числе имитация процессов функционирования Сложная система на ЭВМ (машинный эксперимент).

Концепция применения методов математического моделирования для решения задачи исследования и проектирования сложных систем базируется на следующих основных принципах:

1. Для любой технической системы можно создать математическую модель, которая будет описывать необходимые свойства системы, или ряд моделей.
2. Техническую систему можно исследовать с помощью натурного эксперимента или с помощью математического моделирования.
3. Не всякий натурный эксперимент можно произвести, но всякий эксперимент можно промоделировать.
4. Инженерные решения можно принимать на основе адекватных математических моделей.
5. Для получения адекватных математических моделей необходим эксперимент.
6. Чтобы научиться разрабатывать адекватные математические модели можно применять сравнение численных результатов с теоретическими результатами на основе аналитических решений.
7. Математическая модель состоит из: уравнений, параметров, граничных условий.
8. Ошибка в любом компоненте математической модели даст ошибку в результате математического моделирования.
9. Конечным подтверждением принятого технического решения является натурный эксперимент.

#### 1.3.Основные принципы моделирования

***Принцип информационной достаточности*.** При полном отсутствии информации об исследуемой системе построение ее модели невозможно. При наличии полной информации о системе ее моделирование лишено смысла. Существует некоторый критический уровень априорных сведений о системе (уровень информационной достаточности), при достижении которого может быть построена ее адекватная модель.

***Принцип осуществимости.*** Создаваемая модель должна обеспечить достижение поставленной цели исследования с вероятностью, существенно отличающейся от нуля, и за конечное время. Обычно задают некоторое пороговое значение *P*0 вероятности достижения цели моделирования *P*(*t*), а также приемлемую границу *t*0 времени достижения этой цеди. Модель считают осуществимой, если одновременно выполнены два неравенства:

*P*(*t*) ≥*P0* ; *t* ≤ *t*0

***Принцип множественности моделей.*** Данный принцип, несмотря на его порядковый номер, является ключевым. Речь идет о том, что создаваемая модель должна отражать в первую очередь те свойства реальной системы (или явления), которые влияют на выбранные показатель эффективности. Соответственно при использовании любой конкретной модели познаются лишь некоторые стороны реальности. Для более полного ее исследования необходим ряд моделей, позволяющих с разных сторон и с разной степенью детальности отражать рассматриваемый процесс.

***Принцип агрегирования.*** В большинстве случаев сложную систему можно представить состоящей из агрегатов (подсистем), для адекватного математического описания которых оказываются пригодными некоторые стандартные математические схемы. Принцип агрегирования позволяет, кроме того, достаточно гибко перестраивать модель в зависимости от задач исследования.

***Принцип параметризации.*** В ряде случаев моделируемая система имеет в своем составе некоторые относительно изолированные подсистемы,характеризующиеся определенным параметром, в том числе векторным. Такие подсистемы можно заменять в модели соответствующими числовыми величинами, а не описывать процесс их функционирования. При необходимости зависимость значений этих величин от ситуации может задаваться в виде таблицы, графика или аналитического выражения (формулы). Принцип параметризации позволяет сократить объем и продолжительность моделирования. Однако надо иметь в виду, что параметризация снижает адекватность модели.

Степень реализации перечисленных принципов в каждой конкретной модели может быть различной, причем это зависит не только от желания разработчика, но и от соблюдения им технологии моделирования. А любая технология предполагает наличие определенной последовательности действий.

***Компьютерное моделирование - это математическое моделирование с использованием******средств вычислительной техники.*** Соответственно, технология компьютерного моделирования предполагает выполнение следующих действий:

1. определение цели моделирования;
2. разработка концептуальной модели;
3. формализация модели;
4. программная реализация модели;
5. планирование модельных экспериментов;
6. реализация плана эксперимента;
7. анализ и интерпретация результатов моделирования.

Содержание первых двух этапов практически не зависит от математического метода, положенного в основу моделирования (и даже наоборот – их результат определяет выбор метода). А вот реализация остальных шести существенно различается для каждого из двух основных подходов к построению модели. Именуются эти подходы в разных книгах по – разному, мы используем для их обозначения термины «аналитическое» и «имитационное» моделирование.

***Аналитическое моделирование*** предполагает использование математической модели реального объекта в форме алгебраических, дифференциальных, интегральных и других уравнений, связывающих выходные переменные с входными, дополненных системой ограничений. При этом предполагается наличие однозначной вычислительной процедуры получения точного решения уравнений.

При ***имитационном*** ***моделировании*** используемая математическая модель воспроизводит логику («алгоритм») функционирования исследуемой системы во времени при различных сочетаниях значений параметров системы и внешней среды.

Примером простейшей аналитической модели может служить уже упоминавшееся уравнение прямолинейного движения. При исследовании такого процесса с помощью имитационной модели должно быть реализовано наблюдение за изменением пройденного пути с течением времени.

Очевидно, в одних случаях более предпочтительным является аналитическое моделирование, в других – имитационное (или сочетание того и другого). Чтобы выбор был удачным, необходимо ответить на два вопроса:

* с какой целью проводится моделирование;
* к какому классу может быть отнесено моделируемое явление.

Ответы на оба эти вопроса могут быть получены в ходе выполнения двух первых этапов моделирования.

#### 1.4.Концептуальная модель

***Концептуальная*** (содержательная) ***модель*** - это абстрактная модель, определяющая структуру моделируемой системы, свойства ее элементов и причинно – следственные связи, присущие системе и существенные для достижения цели моделирования.

Построение концептуальной модели включает следующие этапы:

1. определение типа системы;
2. описание рабочей нагрузки;
3. декомпозиция системы.

На первом этапе осуществляется сбор фактических данных (на основе работы с литературой и технической документацией, проведения натурных экспериментов, сбора экспертной информации и т.д.), а также выдвижение гипотез относительно значений параметров и переменных, в тех случаях, когда отсутствует возможность получения фактических данных. Если полученные результаты соответствуют принципам информационной достаточности и осуществимости, то они могут служить основой для отнесения моделируемой системы к одному из известных типов (классов).

Наиболее важные в этом отношении ***классификационные признаки*** приведены ниже.

***1. множество*** ***состояний*** моделируемой системы. По этому признаку системы делят на *статические* и *динамические*. Система называется статической, если множество ее состояний содержит один элемент. Если состояний больше одного, и они могут изменяться во времени, система называется динамической.

Различают *два основных типа динамических* систем:

* с дискретными состояниями (множество состояний конечно или счетно);
* с непрерывным множеством состояний.

Возможны смешанные случаи.

2. ***движением*** ***системы --*** процесс смены состояний.

Смена состояний может происходить либо в фиксированные моменты времени, множество которых дискретно и заранее определено (например, поступление новых партий товара на склад), либо непрерывно (изменение курсов валюты в ходе торгов). При этом различают *детерминированные системы и стохастические.* В *детерминированных* системах новое состояние зависит только от времени и текущего состояния системы. Другими словами, если имеются условия, определяющие переход системы в новое состояние, то для детерминированной системы можно однозначно указать, в какое именно состояние она перейдет.

Для *стохастической* системы можно указать лишь множество возможных состояний перехода и, в некоторых случаях, вероятности перехода в каждое из этих состояний.

Рассмотренная схема классификации систем важна не сама по себе. На этапе разработки концептуальной модели она, во – первых, позволяет уточнить цели и задачи моделирования и, во – вторых, облегчает переход к этапу формализации модели, знание классификационных признаков дает возможность оценить степень ее соответствия первоначальному замыслу разработчика.

3. ***рабочая*** ***нагрузка*** – это совокупность внешних воздействий, оказывающих влияние на эффективность применения данной системы в рамках решаемой задачи,

Описание рабочей нагрузки является не только важной, но и достаточно сложной задачей. Особенно в тех случаях, когда приходится учитывать влияние случайных факторов, или когда идет о рабочей проектируемой принципиальной новой системы. В связи с этим многие вводят понятие модели рабочей нагрузки, подчеркивая сопоставимость уровня сложности описания собственно системы и ее рабочей нагрузки.

Модель рабочей нагрузки (РН) должна обладать следующими основными свойствами:

* совместимостью с моделью системы;
* представительностью;
* управляемостью;
* системной независимостью.

Свойство ***совместимости*** предполагает, что, во – первых степень детализации описания РН соответствует детализации описания системы; во – первых, модель РН должна быть сформулирована в тех же категориях предметной области, что и модель системы. Например, если в модели системы исследуется использование ресурсов, РН должна быть выражена в запросах на ресурсы;

***Представительность*** модели РН определяется ее способностью адекватно представить РН в соответствии с целями исследования. Другими словами, модель РН должна отвечать целям исследования системы. Например, если оценивается пропускная способность, должна выбирать РН, «насыщающая» систему.

Под ***управляемостью*** понимается возможность изменения параметров модели РН в некотором диапазоне, определяемом целями исследования.

***Системная*** ***независимость*** – это возможность переноса модели РН с одной системы на другую с сохранением ее представительности. Данное свойство наиболее важно при решении задачи сравнения различных систем или различных модификаций одной системы. Если модель РН зависит от конфигурации исследуемой системы или других ее параметров, то использование такой модели для решения задачи выбора невозможно.

И наконец, обратимся к этапу, завершающему построение концептуальной модели системы – ее декомпозиции.

***4. Декомпозиция*** ***системы*** производится исходя из выбранного уровня детализации модели, который, в свою очередь, определяется тремя факторами:

* целями моделирования;
* объемом априорной информации о системе;
* требованиями к точности и достоверности результатов моделирования.

Уровни детализации иногда называют ***стратами***, а процесс выделения уровней – ***стратификацией***.

Детализация системы должна производиться до такого уровня, чтобы для каждого элемента были известны или могли быть получены зависимости его выходных характеристик от входных воздействий, существенные с точки зрения выбранного показателя эффективности.

Повышение уровня детализации описания системы позволяет получить более точную ее модель, но усложняет процесс моделирования и ведет к росту затрат времени на проведение.

При имитационном моделировании для оценки выбранного уровня детализации можно использовать специальные критерии.

Первый из них – отношение реального времени функционирования системы к времени моделирования (т. е. к затратам машинного времени, необходимого на проведение модельного эксперимента). Например, если при одних и мех же подходах к программной реализации модели моделирование одного часа работы системы требует в одном случае 3 минуты машинного времени, а в другом – 10 минут, то во втором случае степень детализации описания выше (соотношение 3:10).

Второй критерий – разрешающая способность модели, в том числе:

разрешающая *способность* *по* *времени* – может быть определена как кратчайший интервал модельного времени между соседними событиями;

разрешающая способность *по* *информации* – наименьшая идентифицируемая порция информации, представимая в модели (для вычислительных систем, например, такими порциями могут быть слово, страница, программа, задание).

Третий критерий – число различных моделируемых состояний системы (или типов).

Для тех компонентов, относительно которых известно или предполагается, что они сильнее влияют на точность результатов, степень детальности может быть выше других.

Необходимо отметить, что с увеличением детальности возрастает устойчивость модели, но возрастают и затраты машинного времени на проведение модельного эксперимент.

### Тема 2. МАТЕМАТИЧЕСКИЕ МОДЕЛИ

Под математическим моделированием понимают способ исследования различных процессов путем изучения явлений, имеющих *различное физическое содержание,* но описываемых *одинаковыми математическими соотношениями.* При изучении любого процесса методом математического моде­лирования необходимо в первую очередь построить его матема­тическое описание, или, как мы далее будем говорить, *матема­тическую модель.* В простейших случаях, математическая модель позволяет для данного процесса-оригинала подобрать на основании известных аналогий удобные физические процессы-модели, а также установить соот­ношения подобия, связывающие их параметры, без которых трудно использовать результаты моделирования для изучения процесса-оригинала. В более сложных случаях, когда для моде­лирования создаются специальные моделирующие установки (стенды) или используются вычислительные машины, математическая модель необходима для определения структуры и параметров стенда или построения моделирующего алгоритма.

#### 2.1. Понятие математической модели

Математическая модель, описывает *формализованный* про­цесс функционирования системы и в состоянии охватить только основные, характерные его закономерности.

Процесс функционирования любой системы будем рассма­тривать как последовательную смену ее состояний в некотором интервале времени (*t0*,*t1).* Состояния системы в каждый момент времени *t* из упомянутого интервала характеризуются набором величин *z1, z2, …, zn*. Процесс функционирования системы рассматриваем как последовательную смену состояний, и *z1(t), z2(t), …, zn(t)* являются функциями времени *t.* В дальнейшем будем называть их *характеристиками состояний* системы.

**Под *математической моделью* реальной системы будем пони­мать *совокупность соотношений* (например, формул, уравнений, неравенств, логических условий, операторов и т. д.), *определяю­щих характеристики состояний системы (а через них и выход­ные сигналы) в зависимости от параметров системы, входных сигналов, начальных условий и времени.***

Однако при исследовании реальных систем не всегда удается построить математические модели в виде явных функций или уравнений.

Перейдем к некоторым общим замечаниям, связанным с по­нятием математической модели.

1. *Однозначность* определения характе­ристик состояний системы и выходных сигналов через пара­метры системы, входные сигналы и начальные условия. Это тре­бование выполняется для так называемых детерминированных моделей, представляющих собой совокупность неслучайных со­отношений. Если при этом начальные условия и входные сиг­налы не случайны, то модель оказывается *вполне детерминированной.* На практике нередко приходится рассматривать *случайные* процессы функционирования различных систем. Характеристики состояний системы для таких процессов оказываются случай­ными функциями времени. Будем говорить, что при помощи математической модели *однозначно* определяются *рас­пределения вероятностей* для характеристик состояний системы, если заданы распределения вероятностей для начальных условий, параметров системы и возмущений, действующих на ее элементы, а также для входных сигналов.
2. *Выбор совокупности пара­метров*, характеризующих исследуемую систему. Реальные процессы, если их рассматривать во всех деталях, весьма сложны. Учет большого количества второ­степенных деталей оказывается практически нецелесообразным. В большинстве случаев при решении прикладных задач доста­точно учитывать лишь основные стороны исследуемого процесса. Поэтому обычно при построении математической модели про­цесса ограничиваются сравнительно небольшим количеством па­раметров. В таких условиях, естественно, об однозначности оп­ределения набора параметров, характеризующих систему, не может быть и речи.
3. *Определение совокупности начальных условий*. На этапе формализации процесса, когда контуры математической модели еще недостаточно выяснены, определить перечень начальных условии не представляется воз­можным. Когда же математическая модель построена, перечень начальных условий может быть определен однозначно. Естественно, что перечень начальных условий зависит от того, какие выбраны характеристики состояний системы.

Математическая модель может появиться только как след­ствие четкого формального описания рассматриваемого процесса с требуемой степенью приближения к действительности, только в результате *формализации* процесса.

#### 2.2. Формализация процессов функционирования сложных систем

Математическая модель является результатом *формализа­ции* процесса, т. е. построения четкого формального (математи­ческого) описания процесса с необходимой степенью приближе­ния к действительности.

Модель 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 ***входных воздействий*** на систему*х∈Х, i=1,..nx;*
* совокупность ***воздействий внешней сред****ы vi∈V, i=1, ..,пv***;**
* совокупность ***внутренних (собственных) параметров*** системы *hi∈H, i=1, ..,пh***;**
* совокупность ***выходных характеристик*** системы *yi∈Y, i=1, ..,пy***;**

Причем в перечисленных подмножествах можно выделить управляемые и неуправляемые переменные.

Совокупность зависимостей выходных характеристик системы от времени *уj(t)* для всех видов *j=1,…, пу* называется ***выходной траекторией*** *.* Зависимость называется *законом функ­ционирования системы S* и обозначается *Fs.*

Весьма важным для описания и исследования системы S являет­ся понятие ***алгоритма функционирования As****,* под которым понимает­ся метод получения выходных характеристик с учетом входных воз­действий *x(t),* воздействий внешней среды *v(t)* и собственных па­раметров системы *h (t).* Очевидно, что один и тот же закон функ­ционирования *Fs* системы S может быть реализован различными способами, т. е. с помощью множества различных алгоритмов функционирования *As*.

Очевидно, что детерминированная модель является частным случаем стохастической модели.

Приведенные математические соотношения представляют собой математические схемы общего вида и позволяют описать широкий класс систем. Однако в практике моделирования на первоначальных этапах исследования системы рациональнее использовать ***типовые математические схемы****:* дифференциальные уравнения, конечные и вероятностные автоматы, системы массового обслуживания и т. д.

При построении математических моделей про­цессов функционирования систем можно выделить следующие основные подходы: непрерывно–детерминированный (например, дифференциальные уравнения); дискретно–детерминированный (ко­нечные автоматы); дискретно-стохастический (вероятностные авто­маты); непрерывно-стохастический (системы массового обслужи­вания); обобщенный или универсальный (агрегативные системы).

Математические схемы, рассматриваемые в последующих па­раграфах данной главы, должны помочь оперировать различными подходами в практической работе при моделировании конкретных систем.

#### 2.3.Математические схемы

##### 2.3.1.Непрерывно-детерминированные модели (D-схемы)

Обычно в таких математических моделях в качестве независи­мой переменной, от которой зависят неизвестные искомые функции, служит время *t.* Тогда математическое соотношение для детерми­нированных систем в общем виде будет

,

где:

и  -- n-мерные векторы,

-- вектор-функция, которая определена на некотором (n+1)—мерном множестве (ŷ,t) и является непрерывной.

Так как математические схемы такого вида отражают динами­ку изучаемой системы, т. е. ее поведение во времени, то они назы­ваются ***D-схемами*** (англ. dynamic)**.**

Использование *D-схем* позволяет формализовать процесс функционирования непрерывно–детерминированных систем S и оценить их основные характеристики, применяя анали­тический или имитационный подход, реализованный в виде соответствующего языка для моделирования непрерывных систем или использующий аналоговые и гибридные средства вычислительной техники.

##### 2.3.2. Дискретно-детерминированные модели (f-схемы)

Особенности дискретно–детерминированного подхода на этапе формализации процесса функционирования систем может быть рассмотрен на примере использования в качестве математического аппарата теории автоматов. На основе этой теории система представляется в виде автомата, перерабатывающего дискретную информацию и меняющего свои внутренние состояния лишь в допустимые моменты времени.

Абстрактно конечный автомат (англ. finite automata) можно представить как математическую схему ***(F-схему****),* характеризующуюся шестью элементами:

* конечным множеством Х входных сигналов (входным алфавитом);
* конечным множеством Y выходных сигналов (выходным алфавитом);
* конечным множеством Z внутренних состояний (внутренним алфавитом или алфавитом состояний);
* начальным состоянием z0, z0∈Z;
* функцией переходов *ϕ(z,x);*
* функцией выходов *ψ(z, x).*

Автомат, задаваемый *F-схемой,* принято обозначать*:*

F=*<****Z, X, Y****,ϕ,ψ,z0>.*

Абстрактный конечный автомат имеет один входной и один вы­ходной каналы. В каждый момент *t=0,* 1, 2,... дискретного вре­мени F-автомат находится в определенном состоянии *z(t)* из мно­жества ***Z*** состояний автомата, причем в начальный момент времени *i=0* он всегда находится в начальном состоянии *z(0)=zo*. В мо­мент *t,* будучи в состоянии *z(t),* автомат способен воспринять на входном канале сигнал *x(t)∈X* и выдать на выходном канале сигнал *у(t)*=*ψ[z(t), x(t)],* переходя в состояние *z(t+1)=* =*ϕ[z(t),x(t)], z(t)∈****Z****, y(t)∈****Y****.* Другими словами, если на вход конечного автомата, установленного в на­чальное состояние *z0*, подавать в некоторой последовательности буквы входного алфавита *х(0), х(1), х(2),...,* т. е. входное слово, то на выходе автомата будут последовательно появляться буквы выходного алфавита *у(0), у(1), у (2),...,* образуя вы­ходное слово.

Сказанное выше можно описать следующими уравнениями: для F-автомата первого рода, называемого также автоматом Мили,

*z(t+1)=* =*ϕ[z(t),x(t)], у(t)*=*ψ[z(t), x(t)], t*=0,1,2,...; (2.1)

для F-автомата второго рода

*z(t+1)=ϕ[z(t),x(t)], у(t)*=*ψ[z(t), x(t-1)], t*=0,1,2,...; (2.2)

Автомат второго рода, для которого *у(t)*=*ψ[z(t)], t*=0,1,2,..., т. е. функция выходов не зависит от входной переменной *х {t),* называется автоматом Мура.

По числу состояний различают конечные автоматы с памятью и без памяти. Автоматы с памятью имеют более одного со­стояния, а автоматы без памяти (комбинационные или логические схемы) обладают лишь одним состоянием.

По характеру отсчета дискретного времени конечные автоматы делятся на *синхронные и асинхронные*. В синхронных F-автоматах моменты времени, в которые автомат «считывает» входные сигналы, определяются принудительно синхронизирующи­ми сигналами. Реакция автомата на каждое значе­ние входного сигнала заканчивается за один такт, длительность которого определяется интервалом между соседними синхронизи­рующими сигналами. Асинхронный F-автомат считывает входной сигнал непрерывно и поэтому, реагируя на достаточно длинный входной сигнал постоянной величины *х,* он может, несколько раз изменять состояние, выдавая соот­ветствующее число выходных сигналов, пока не перейдет в устой­чивое, которое уже не может быть изменено данным входным.

##### 2.3.3. Дискретно-стохастические модели (Р-схемы)

Рассмотрим особенности построения математических схем при дискретно-стохастическом подходе к формализации процесса функ­ционирования исследуемой системы S. Поскольку сущность дискре­тизации времени при этом подходе остается аналогичной рассмот­ренным конечным автоматам, то влияние фактора стохастичности проследим также на разновидности таких автоматов, а именно на вероятностных (стохастических) автоматах. В общем виде *вероятностный автомат* (англ. probabilistic automat) можно определить как дискретный потактный преобразо­ватель информации с памятью, функционирование которого в каж­дом такте зависит только от состояния памяти в нем и может быть описано статистически.

Вероятностный автомат может быть описан либо таблицей переходов, либо матрицей переходов Р-автомата и начальным распределением вероятностей. Математический аппарат, используемый при исследовании Р-автоматов, является аппарат марковских цепей.

##### 2.3.4. Непрерывно-стохастические модели (q-схемы)

Особенности непрерывно-стохастического подхода рассмотрим на примере использования в качестве типовых математических схем *систем массового обслуживания,* (англ. queueing system), которые будем называть *Q-схемами.* Системы массового обслуживания представляют собой класс математических схем, разработанных в теории массового обслуживания и различных приложениях для формализации процессов функционирования систем, которые по своей сути являются процессами обслуживания.

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 удаленных терминалов и т. д. При этом характерным для работы таких объек­тов является случайное появление заявок (требований) на обслу­живание и завершение обслуживания в случайные моменты време­ни, т. е. стохастический характер процесса их функционирования. Остановимся на основных понятиях массового обслуживания, не­обходимых для использования *Q-схем* как при аналитическом, так и при имитационном подходе.

Работа любой системы массового обслуживания состоит в выполнении поступающего на ее вход потока заявок. Заявки поступают в некоторые, в общем случае случайные, моменты времени. Обслуживание заявки продолжается какое-то время, также случайное, после чего канал освобождается для обслуживания следующей заявки. Предмет теории массового обслуживания – установление зависимостей между характером потока заявок, производительностью отдельного канала обслуживания, числом каналов и эффективностью обслуживания.

Случайный процесс, протекающий в СМО состоит в том, что система в случайные моменты времени переходит из одного состояния в другое. СМО представляет собой физическую систему дискретного типа, а переход системы из одного состояния в другое происходит скачком.

В любой момент времени система пребывает в одном из возможных состояний и очевидно, что для любого t справедливо:



Случайные процессы со счетным множеством состояний бывают двух типов: с дискретным или непрерывным временем.

С дискретным временем: переходы из состояния в состояние могут происходить только в строго определенные, разделенные конечными интервалами моменты времени t1, t2 … .

С непрерывным временем: переход системы из состояния в состояние возможен в любой момент времени.

Случайные процессы, протекающие в СМО как правило являются процессами с непрерывным временем. Граф перехода системы из состояние в состояние может быть проиллюстрирован рис.



##### 2.3.5. Обобщенные модели (А-схемы)

Наиболее известным общим подходом к формальному описанию процессов функционирования систем является подход, предложенный Н.П.Бусленко. Этот подход позволяет описывать поведение непрерывных и дискретных, детерминированных и стохастических, т.е. по сравнению с рассмотренными является обобщенным (универсальным) и базируется на понятии агрегатируемой системы, представляющей собой формальную схему общего вида, которую принято называть А-схемой.

Анализ существующих средств моделирования показывает, что комплексное решение проблем, возникающих в процессе создания и реализации модели, возможно только в том случае, когда моделирующие системы имеют в своей основе единую формальную математическую схему. Такая схема должна выполнять следующие функции:

* являться адекватным математическим описанием объекта моделирования;
* служить основой для построения алгоритмов и программ, реализующих модель;
* позволять в упрощенном варианте проводить аналитические исследования.

В качестве элемента А-схемы выступает агрегат. Связь между агрегатами (внутри системы S и внешней средой Е) осуществляется опеатором R. Агрегат может разбиваться на агрегаты следующего уровня.

Любой агрегат характеризуется следующими множествами:

* моментами времени Т;
* входными сигналами Х;
* выходными сигналами У;
* состояниями на каждый момент времени Z(t).

Переход агрегата из состояния в состояние происходит за малый интервал времени δt z(t2)≠z(t1). Изменение состояния определяется скачком δz. Агрегат из состояния в состояние переходит в зависимости от собственных (внутренних) параметров h(t) и входных сигналов x(t).

В начальный момент времени агрегат находится в состоянии z(t0)=z0, которое задается законом L(z(t0)).

Процесс функционирования агрегата в случае воздействия сигнала xn описывается случайным оператором V. Пусть в момент времени tn поступил сигнал xn. Состояние агрегата определиться так:

Z(tn+0)=V(tn,z(tn),xn).

Если в течение времени (tn,tn+1) не пришло ин одного входного сигнала, то агрегат может перейти в другое состояние за счет изменение внутреннего состояния в соответствии со случайным оператором U:

z(t)=U(t,tn,z(tn+0)).

Совокупность случайных операторов V и U рассматривается как оператор перехода автомата в новые состояния. При этом процесс функционирования агрегата состоит из скачков состояний δz в моменты поступления новых сигналов х и изменений состояний агрегата между этими моментами. Моменты скачков δz называются особыми состояниями А-схемы. Для описания скачков в особые моменты используется оператор W, представляющий собой частный случай оператора U:

*z(tδ)=W(tδ,z(tδ)).*

В множестве состояний агрегата выделяется подмножество Z(Y), которое является подмножеством выдачи выходного сигнала:

*Y=G(tδ,z(tδ)).*

Таким образом, под агрегатом будем понимать объект, определяемый упорядоченной совокупностью рассмотренных множеств T, X, Y, Z, Z(Y), H и случайных операторов V, U, W, G.

Последовательность входных сигналов, расположенных в порядке поступления их на вход А-схемы называют входным сообщением, а последовательных выходных – выходным сообщением.

Существует класс больших систем, которые ввиду их сложности не могут быть формализованы в виде математических схем одиночных агрегатов, поэтому их формализуют некоторой конструкцией из отдельных агрегатов. Для описания системы в целом, необходимо иметь описание как отдельных агрегатов, так и связей между ними.

Для построения формального понятия А-схемы необходимо выбрать способы математического описания взаимодействия между агрегатами. Для этого вводится ряд предположений о закономерностях функционирования А-схем, которые согласуются с опытом исследования реальных сложных систем:

* взаимодействие между А-схемой и внешней средой Е, а также между отдельными агрегатами внутри системы осуществляется при передаче сигналов, причем взаимные влияния, имеющие место вне механизма передачи сигналов не учитываютяся;
* для описания сигнала достаточно некоторого конечного набора характеристик;
* элементарные сигналы мгновенно передаются в А-схеме независимо друг от друга по элементарным каналам;
* ко входному контакту любого элемента А-схемы подключается не более чем один элементарный канал, к выходному контакту – любое конечное число элементарных каналов.

Взаимодействие А-схемы с внешней средой рассматривается как обмен сигналами между внешней средой и элементами А-схемы. В связи с этим внешнюю среду можно представить в виде фиктивного элемента А-схемы.

Таким образом, использование обобщенной типовой математической схемы моделирования А\_схемы в принципе не отличается от использования рассмотренных ранее D, F, P, Q-схем. Для частного случая результаты могут быть получены аналитическим методом. В более сложных случаях прибегают к имитационному методу.

Представление объекта моделирования в виде А-схемы может являться тем фундаментом, на котором базируется построение имитационной системы и ее внешнего и внутреннего математического обеспечения. Стандартная форма представления исследуемого объекта в виде А-схемы приводит к унификации не только алгоритмов имитации, но и к возможности применять стандартные методы обработки и анализа результатов моделирования.

### Тема 3. МОДЕЛИРОВАНИЕ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ

#### 3.1. Понятие СМО

*Системой массового обслуживания* (СМО) называется любая система предназначенная для обслуживания каких-либо заявок (требований), посту­пающих на нее в случайные моменты времени.

В качестве процесса обслуживания могут быть представлены различные по своей физической природе процессы функционирования экономических, производственных, технических и других систем. Примеры систем массового обслуживания следующие: потоки поставок продукции некоторому предприятию, потоки деталей и комплектующих изделий на сборочном конвейере цеха, заявки на обработку информации ЭВМ от удаленных терминалов и т. д. При этом характерным для работы таких объек­тов является случайное появление заявок (требований) на обслу­живание и завершение обслуживания в случайные моменты време­ни, т. е. стохастический характер процесса их функционирования. Остановимся на основных понятиях массового обслуживания, не­обходимых как при аналитическом, так и при имитационном подходе.

Работа любой системы массового обслуживания состоит в выполнении поступающего на ее вход потока ***заявок***. Заявки поступают в некоторые, в общем случае случайные, моменты времени. Обслуживание заявки продолжается какое-то время, также случайное, после чего канал освобождается для обслуживания следующей заявки. Предмет теории массового обслуживания – установление зависимостей между характером потока заявок, производительностью отдельного канала обслуживания, числом каналов и эффективностью обслуживания.

Различают СМО *с отказами и* *СМО с очередью.* В СМО с отказами заявка, пришедшая в момент, когда все каналы заняты, получает отказ, покидает СМО и в дальнейшем в процессе ее работы не участвует. В СМО с очередью заявка, при­шедшая в момент занятости всех каналов, не покидает СМО, а становится в оче­редь и ждет, пока не освободится какой-нибудь канал. Число мест в очереди m может быть как ограниченным, так и неограниченным. При m = О СМО с оче­редью превращается в СМО с отказами. Очередь может иметь ограничения не только по количеству стоящих в ней заявок (длине очереди), но и по времени ожидания (такие СМО называются «системами с нетерпеливыми клиентами»).

СМО с очередью различаются не только по ограничениям очереди, но и по *дисциплине обслуживания:* обслуживаются ли заявки в порядке поступления, или в случайном порядке, или же некоторые заявки обслуживаются вне оче­реди (так называемые «СМО с приоритетом»). Приоритет может иметь несколько градаций или рангов.

Аналитическое исследование СМО является наиболее простым, если все потоки событий, переводящие ее из состояния в состояние, — простейшие (ста­ционарные пуассоновские). Это значит, что интервалы времени между события­ми в потоках имеют показательное распределение с параметром, равным интен­сивности соответствующего потока. Для СМО это допущение означает, что как поток заявок, так и поток обслуживании — простейшие. Под *потоком обслужи­вании* понимается поток заявок, обслуживаемых одна за другой одним непрерыв­но занятым каналом. Этот поток оказывается простейшим, только если время обслуживания заявки *Тобс* представляет собой случайную величину, имеющую показательное распределение. Параметр этого распределения  есть величина, обратная среднему времени обслуживания. Вместо «поток обслуживании — простейший» часто говорят «время обслужива­ния — показательное». Условимся в дальнейшем для краткости всякую СМО, в которой все потоки простейшие, называть *простейшей* СМО. В этой главе мы будет рассматривать главным образом простейшие СМО.

Если всё потоки событий простейшие, то процесс, протекающий в СМО, представляет собой марковский случайный процесс с дискретными состояниями и непрерывным, временем. При выполнении некоторых условий для этого процес­са существует финальный стационарный режим, при котором как вероятности со­стояний, так и другие характеристики процесса не зависят от времени.

Задачи теории массового обслуживания — нахождение вероятностей раз­личных состояний СМО, а также установление зависимости между заданными параметрами (числом каналов n, интенсивностью потока заявок , распределе­нием времени обслуживания и т, д.) и *характеристиками эффективности* ра­боты СМО. В качестве таких характеристик могут рассматриваться, например, следующие:

* среднее число заявок *А,* обслуживаемое СМО в единицу времени, или *абсолютная пропускная способность* СМО;
* вероятность обслуживания- поступившей заявки Q или *относительная пропускная способность* СМО;*;*
* вероятность отказа Ротк т.е вероятность того, что поступившая заявка не будет обслужена, получит отказ; Ротк = 1 - q;

Рассмотри процессы, протекающие в системе массового обслуживания.

#### 3.2.Мнемоническое обозначение СМО.

В теории массового обслуживания приняты очень удобные сокращенные обозначения для различных СМО, позволяющие легко охарактеризовать систему. В основе этих обозначений лежит трехбуквенная комбинация вида А/В/N, где:

А — описывает распределение (или задает характер закона распределения) интервалов поступления заявок;

В — описывает распределение длительностей обслуживания заявок;

N — задает количество обслуживающих приборов в СМО.

Иногда, когда СМО является системой с ограниченной емкостью накопителя (или с ограниченной очередью), приведенное обозначение расширяется до четырех букв А/В/N/К, где последняя буква (на самом деле число, как и N) К задает емкость накопителя (количество мест ожидания).

Приведенные трех или четырех буквенные обозначения называют обозначениями Кендалла. В этих обозначениях А и В могут принимать значения из следующего набора символов {M, D, Ek, Hk, G, U}. При этом:

а) А или В=M, если распределение интервалов поступления или длительностей обслуживания заявок является экспоненциальным (М — от слова Markovian — Марковский);

б) А или В=D, если интервалы поступления или длительности обслуживания являются детерминированными (D — Determinate);

в) А или В=Ek, если соответствующие распределения являются Эрланговскими порядка k (E — Erlang);

г) А или В=Hk, в случае гиперэкспоненциальных распределений порядка k (H — Hyperexponential);

д) А или В= G, в случае распределений общего (произвольного) вида (G — General — общий, общего вида);

е) А или В= U — при равномерных распределениях соответствующих случайных величин (U — Uniform distribution — равномерное распределение).

Так, например, обозначение вида:

М/М/1 означает СМО с простейшим потоком на входе и экспоненциально распределенной длительностью обслуживания заявок в приборе (один)

D/Е2/3/5 — СМО с регулярным потоком на входе, длительностью обслуживания, распределенной по закону Эрланга 2-го порядка, тремя обслуживающими приборами и пятью местами ожидания;

М/G/2 — СМО с простейшим потоком на входе, длительностью обслуживания, распределенная по закону произвольного вида, и двумя обслуживающими приборами.

В случае СМО с неоднородной нагрузкой используются обозначения вида, где символ вектора над буквами А и В указывает на неоднородность нагрузки, а индекс Н задает количество классов заявок. Например, — это обозначение СМО с одним обслуживающим прибором, четырьмя классами заявок, которые образуют на входе системы простейшие потоки и имеют общие законы распределения длительностей обслуживания.

#### 3.3. СМО с отказами

Системы массового обслуживания делятся на системы с отказами и системы с ожиданием.

В системах с отказами заявка, поступившая в момент, когда все каналы обслуживания заняты, немедленно получает отказ, покидает систему и в дальнейшем в процессе обслуживания не участвует.

Пусть имеется n-канальная СМО с отказами. Рассмотрим конечное множество состояний этой системы:

*z0* – свободны все каналы;

*z1* – занят один канал;

. . . . . . . . . . . . . . . . . . .

*zk* – заняты k каналов;

. . . . . . . . . . . . . . . . ..

*zn* – заняты все n каналов.



*Определим вероятности состояния системы pk(t) для любого момента времени в предположении, что поток заявок простейший, с интенсивностью λ, время обслуживания показательное, с параметром μ.*

*Поскольку оба потока заявок в системе (заявок и обслуживания) являются простейшими, то процесс, протекающий в системе будет марковским.*

*Очевидно, что для любого момента времени*

*.*

*Составим дифференциальные уравнения для всех вероятностей состояний системы. Для этого, зафиксируем момент времени t и найдем вероятность pk(t+Δt) того, что в момент (t+Δt) система будет находиться в состоянии zk.*

*Для состояния z0 это может произойти двумя способами:*

*событие A – в момент времени t система находилась в состоянии z0 и осталась в этом состоянии. Вероятность этого события равна вероятности того, что за время Δt на вход системы не пришла ни одной заявки:*

**

*Следовательно, P(A)=p0(t)(1-λ˙Δt).*

*событие B – вероятность того, что система была в состоянии z1 и перешла в состояние z0. Вероятность этого события равна:*

**

*Следовательно, P(B)=p1(t)μ˙Δt.*

*Таким образом:*

*p0(t+Δt)=p0(t)(1-λ˙Δt)+ p1(t)μ˙Δt.*

**

*Аналогично составляются дифференциальные уравнения для других состояний системы. Для состояния zk вероятность pk(t+Δt) определиться как сумма вероятностей трех событий:*

*событие A – в момент времени t система находилась в состоянии zk и осталась в этом состоянии. Вероятность этого события равна вероятности того, что за время Δt на вход системы не пришла ни одной заявки и ни одна из k заявок из системы не ушла (не обслужилась):*

**

*Следовательно, P(A)=pk(t)[1-(λ+kμμ)˙Δt].*

*событие B – вероятность того, что система была в состоянии zk-1 и перешла в состояние zk. (пришла одна заявка). Вероятность этого события равна:*

*P(B)=pk-1(t)λ˙Δt.*

*событие C – вероятность того, что система была в состоянии zk+1 и перешла в состояние zk. (обслужена одна заявка). Вероятность этого события равна:*

*P(С)=pk+1(t)(k+1)μ˙Δt.*

*Таким образом:*

*pk(t+Δt)= pk(t)[1-(λ+kμμ)˙Δt]+ pk-1(t)λ˙Δt+ pk+1(t)(k+1)μ˙Δt.*

**

*Составим уравнение для последней вероятности pn:*

*pn(t+Δt)≈ pn(t)(1-nμ˙Δt)+ pn-1(t)λ˙Δt.*

**

*Таким образом, получена система дифференциальных уравнений для вероятностей состояний системы:*

**

*. . . . . . .*

**

*. . . . . . (0<k<n),*

**

*Эти уравнения называются уравнениями Эрланга.*

Вероятности *pk(t)* характеризуют среднюю загрузку системы и ее изменение с течением времени.

Вероятность *pn(t)=Pотк* есть вероятность того, что заявка, пришедшая в систему в момент времени t получит отказ.

Величина q(t)=1-pn(t) называется пропускной способностью системы.

Введем обозначение α=λ/μ и назовем величину α *приведенной плотностью потока заявок*. Эта величина есть также среднее число заявок, приходящееся на среднее время обслуживания одной заявки: α=λmобсл.

В новых обозначениях вероятности pk принимает вид:



Приведенные выше формулы выражают вероятности pk через p0. Для того, чтобы выразить эти вероятности через характеристики системы α и n, воспользуемся условием нормировки:



откуда



Окончательное выражение для вероятностей состояния системы принимают вид:

 (0≤k≤n).

Вероятность отказа (все каналы заняты):



Для одноканальной системы (n=1):

.

Относительная пропускная способность :



Формулы Эрланга и их следствия были получены в предположении о показательном распределении времени обслуживания заявок. Однако исследования показали, что эти формулы справедливы при любом законе распределения времени обслуживания, лишь бы входной поток был простейшим.

#### 3.4. СМО с ожиданием

Система массового обслуживания называется системой с ожиданием, если заявка, заставшая все каналы занятыми, становится в очередь и ждет, пока не освободится какой-нибудь канал.

Если время ожидания заявки в очереди ничем не ограничено, то система называется *чистой системой с ожиданием*. Если оно ограничено некоторыми условиями, то система называется системой смешанного типа. Ограничения, наложенные на ожидание могут быть различного типа, например:

* ограничение на время пребывания заявки в очереди;
* ограничение на длину очереди;
* ограничение на время пребывания заявки в системе.

В системах с ожиданием существенную роль играет так называемая *дисциплина очереди*. Каждый тип системы с ожиданием имеет свои особенности и математическую теорию. Мы остановимся на простейшем случае смешанной системы, являющимся обобщением задачи Эрланга для системы с отказами.

Рассмотрим СМО с n каналами, на вход которой поступает простейший поток с параметром λ. Время обслуживания заявок также имеет показательное распределение с параметром μ. Заявка, заставшая все каналы занятыми, становится в очередь и ожидает обслуживания. Время ожидания заявки в очереди ограничено некоторым сроком Tож. Если до истечения этого срока заявка не будет обслужена, то она покидает систему. Срок ожидания обслуживания будем полагать случайной величиной с показательным распределением и параметром ν. Очевидно, что при ν→∞, система смешанного типа превращается в чистую систему с отказами, а при ν→0, система смешанного типа превращается в чистую систему с ожиданиями.

Отметим, что в предположении о показательном распределении срока ожидания пропускная способность системы не зависит от того, обслуживаются ли заявки в порядке очереди ли в случайно порядке: для каждой заявки закон распределения оставшегося времени ожидания не зависит от того, сколько времени заявка стояла в очереди.

* для любого k≤n 
* для любого s≥1: 

В приведенных выше формулах в качестве сомножителя присутствует вероятность p0. Определим эту вероятность из дополнительного условия:



Введем обозначения:

λ/μ=λmtобсл= α; ν/μ=νmtобсл=β.

Параметра α и β выражают соответственно среднее число заявок и среднее число необслуженных заявок приходящееся на среднее время обслуживания одной заявки.

В новых обозначениях приведенные выше выражения принимают вид:

 (0<k≤n)

; (s≥1).

Зная вероятности состояния системы можно определить и другие интересующие нас характеристики, в частности вероятность того, что заявка покинет систему не обслуженной. Определим эту вероятность из следующих соображений: при установившемся режиме вероятность Pн есть отношение среднего числа заявок, уходящих из очереди в единицу времени. Определим среднее число заявок, находящихся в очереди:



Чтобы найти вероятность Pн, нужно среднее число заявок в очереди умножить на среднюю плотность уходов (определим среднее число заявок, покидающих систему) и умножим на интенсивность входного потока заявок:



Относительная пропускная способность системы: q=1-Pн.

Очевидно, что пропускная система с ожиданиями выше, чем пропускная способность системы с отказами и пропускная способность увеличивается с увеличением среднего времени ожидания mtож=1/ν.

Рассмотрим, во что превратиться система с ожиданиями при изменении параметра β. Очевидно, что при β→∞ система с ожиданиями превращается в чистую систему с отказами, а при β→0 – в чистую систему с ожиданиями. В такой системе вероятность того, что заявка уйдет из системы не обслуженной, равна нулю. Однако, в такой системе не всегда имеется предельный стационарный режим при t→∞. Такой режим существует только при α<n, т.е., когда среднее число заявок, приходящееся на время обслуживания одной заявки не выходит за пределы возможностей n-канальной системы. В противном случае, число заявок в очереди будет неограниченно возрастать.

Полагая, что α<n, найдем предельные вероятности состояния системы (β→0):



Отсюда найдем:

 (0≤k≤n).

 (s≥0).

Среднее число заявок в очереди:



#### 3.5 Простейшая многофазовая СМО с очередью.

Анализ многофазовых СМО в общем случае затруднен, тем что входящий поток каждой последующей фазы является выходным потоком предыдущей и в общем случае имеет последействие.Однако *если на вход С МО с неограниченной очередью поступает простейший поток заявок, а время обслуживания показательное, то выходной поток, этой СМО* — *простейший,* с той же интенсивностью *,* что и входящий. Из этого следует, что многофазовую СМО с неограниченной очередью перед каждой фазой, простейшим входящим потоком заявок и показательным временем обслуживания на каждой фазе можно анализировать как простую последовательность простейших СМО.

Если очередь к фазе ограничена, то выходной поток этой фазы перестает быть простейшим и вышеуказанный прием может применяться только в качестве приближенного.

### Тема 4. ПРИНЦИПЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

#### 4.1. Понятие статистического эксперимента

Имитационное моделирование представляет собой наблюдение поведения модели системы под влиянием входных воздействий. При этом часть из них (а может быть и все) носят случайный характер. В результате такого наблюдения исследователь получает набор экспериментальных данных, на основе которых могут быть оценены характеристики системы.

Очевидно, что аналитические модели для проведения имитационного эксперимента не годятся, и здесь нужна специальная «имитационная» модель, которая должна отвечать следующим основным требованиям:

* Отражать логику функционирования исследуемой системы во времени;
* Обеспечить возможность проведения статистического эксперимента.

Одним из основных понятий имитационного моделирования является понятие статистического эксперимента.

В его основе лежит метод статистических испытаний (метод Монте-Карло). Суть метода заключается в том, что результат испытания ставится в зависимость от значения некоторой случайной величины (СВ), распределенной по заданному закону. Результат каждого конкретного испытания носит случайный характер.

Проведя серию испытаний получают множество частных значений наблюдаемой характеристики (то есть выборку). Поученные статистические данные обрабатываются и представляются в виде численных оценок интересующих исследователя параметров.

Отметим, что метод статистических испытаний применим для исследования как стохастических, так и детерминированных систем.

Важной особенностью метода является то, что его применение практически невозможно без использования компьютерной техники.

Имитационное моделирование не ограничивается разработкой модели и написанием соответствующей программы, а требует подготовки и проведения статистического эксперимента. В вязи с этим результаты имитационного моделирования следует рассматривать как экспериментальные данные, требующие специальной обработки и анализа. Для любого модельного эксперимента необходимо ответить на следующие вопросы:

1. Какова должна быть продолжительность эксперимента для достижения стационарных условий?
2. Как получить статистически независимые наблюдения?
3. Сколько наблюдений необходимо для обеспечения требуемой точности?

#### 4.2. Область применения и классификация имитационных моделей

***Имитационная модель*** (ИМ) — это формальное (то есть выполненное на не­котором формальном языке) описание логики функционирования исследуемой си­стемы и взаимодействия отдельных ее элементов во времени, учитывающее наибо­лее существенные причинно-следственные связи, присущие системе, и обеспечивающее проведение статистических экспериментов.

Необходимо отметить два важных обстоятельства:

1) взаимосвязь между отдельными элементами системы, описанными в мо­дели, а также между некоторыми величинами (параметрами) может быть пред­ставлена в виде аналитических зависимостей (например, при моделировании полета управляемой ракеты отработка поступающих на борт команд может быть описана на уровне логики, а возникающие перегрузки рассчитываются анали­тически);

2) модель можно считать реализуемой и имеющей практическую ценность только в том случае, если в ней отражены лишь те свойства реальной системы, которые влияют на значение выбранного показателя эффективности.

Как было отмечено выше, для ИМ практически отсутствуют ограничения на область их применения (по типу моделируемой системы), и речь может идти толь­ко о целесообразности использования ИМ в данной предметной области и об объе­ме трудозатрат на ее разработку.

Поскольку основой имитационного моделирования является метод статисти­ческих испытаний, наибольший эффект от его применения достигается при иссле­довании сложных систем, на функционирование которых существенное влияние оказывают случайные факторы.

Применение имитационного моделирования целесообразно также в следующих случаях:

1) если не существует законченной постановки задачи на исследование и идет процесс познания объекта моделирования;

2) если характер протекающих в системе процессов не позволяет описать эти процессы в аналитической форме;

3) если необходимо наблюдать за поведением системы (или отдельных ее ком­понентов) в течение определенного периода, в том числе с изменением скорости протекания процессов;

4) при изучении новых ситуаций в системе либо при оценке функционирования ее в новых условиях;

5) если исследуемая система является элементом более сложной системы, дру­гие элементы которой имеют реальное воплощение;

6) когда необходимо исследовать поведение системы при введении в нее новых компонентов;

7) при подготовке специалистов и освоении новой техники (в качестве тре­нажеров).

Но имитационные модели имеют целый ряд недостатков. Первый, и весьма существенный, заключается в том, что разра­ботка ИМ, как правило, требует больших затрат времени и сил. Кроме того, любая имитационная модель сложной системы значительно менее «объектив­на», чем аналитическая модель, поскольку она прежде всего отражает субъек­тивные представления разработчика о моделируемой системе. Причем бывает достаточно сложно как опровергнуть, так и обосновать адекватность создан­ной ИМ, особенно если речь идет о проектируемой системе. И, наконец, еще одно обстоятельство. Результаты имитационного моделирования, как и при любом численном методе, всегда носят частный характер. Для получения обо­снованных выводов необходимо проведение серии модельных экспериментов, а обработка результатов требует применения специальных статистических процедур.

Каким же образом можно преодолеть указанные недостатки?

Во-первых, современное состояние вычислительной техники и ее программно­го обеспечения позволило создать пакеты моделирования, использование которых существенно сокращает трудозатраты на создание моделей, статистичес­кий анализ и визуализацию полученных результатов.

Во-вторых, «объективность» создаваемой модели может быть обеспечена в том случае, когда для каждого вариан­та постановки задачи исследования выбирается соответствующая схема построения модели.

В этом отношении знание существующих схем построения имитационных мо­делей является весьма полезным.

Наиболее важный признак — ***способ представления в модели динамики (дви­жения) системы.*** Она может быть описана посредством событий, работ (активно­стей), процессов и транзактов.

Другой важный признак — ***способ изменения модельного времени.*** По этому признаку различают моделирование с постоянным шагом и моделирование по осо­бым состояниям.

Все эти понятия являются основополагающими в теории имитационного моде­лирования.

В зависимо­сти от этапа и назначения проводимых исследований применяется один из трех наиболее распространенных видов имитационных экспериментов:

1) исследование относительного влияния различных факторов на значения вы­ходных характеристик системы;

2) нахождение аналитической зависимости между интересующими исследова­теля выходными характеристиками и факторами;

3) отыскание оптимальных значений параметров системы (так называемый «эк­стремальный эксперимент»).

Вид эксперимента влияет не только на выбор схемы ее формализации, но также на построение плана эксперимента и выбор метода обработки его результатов.

С точки зрения организации взаимодействия исследователя с моделью в ходе эксперимента ИМ делятся на автоматические и диалоговые.

***Автоматическими*** называются ИМ, взаимодействие пользователя с которы­ми сводится только к вводу исходной информации и управлению началом и окон­чанием работы моделей.

***Диалоговыми*** называются ИМ, позволяющие исследователю активно управ­лять ходом моделирования.

#### 4.3. Описание поведения системы

Описание динамики системы, или, проще говоря, ее поведения, составляет ос­нову любой имитационной модели. В качестве исходных посылок для решения этой задачи используются результаты, полученные на этапе разработки концептуаль­ной модели системы. К ним относятся:

• определение принадлежности моделируемой системы одному из известных классов;

• описание рабочей нагрузки системы;

• выбор уровня детализации представления системы в модели и ее декомпозиция.

Все последующие действия исследователя по созданию модели могут быть от­несены к этапу ее формализации, который в общем случае предполагает:

• выбор метода отображения динамики системы (на основе событий, процессов или транзактов);

• формальное (математическое) описание случайных факторов, подлежащих учету в модели;

• выбор механизма изменения и масштаба модельного времени.

***Работа (активность)*** *—* это единичное действие системы по обработке (пре­образованию) входных данных. В зависимости от природы моделируемой системы под входными данными могут пониматься информационные данные или какие-либо материальные ресурсы. Каждая из работ характеризуется временем выполнения и потребляемыми ре­сурсами.

Под ***процессом*** понимают логически связанный набор работ. Некоторые процессы могут рассматриваться, в свою очередь, как работы в процессе более высокого уровня. Процесс характеризуется совокупностью статических и динамических характеристик.

**К *статическим*** характеристикам процесса относятся:

• длительность;

• результат;

• потребляемые ресурсы;

• условия запуска (активизации);

• условия останова (прерывания).

В общем случае статические характеристики процесса не изменяются в ходе его реализации, однако, при необходимости любая из них может быть представлена в модели как случайная величина, распределенная по заданному закону.

***Динамической характеристикой*** процесса является его состояние (активен или находится в состоянии ожидания).

Моделирование в терминах процессов производится в тех случаях, когда систе­ма оценивается по каким-либо временным показателям, либо с точки зрения по­требляемых ресурсов.

*Например, при оценке производительности вычислительной сети обработка заданий может быть представлена в модели как совокупность соответствующих процессов, использующих ресурсы сети (оперативную память, пространство на жестких дисках, процессорное время, принтеры и т. д.).*

В том случае, если модель строится с целью изучения причинно-следственных связей, присущих системе, динамику системы целесообразно описывать в терми­нах событий.

***Событие*** представляет собой мгновенное изменение некоторого элемента сис­темы или состояния системы в целом.

Событие характеризуется:

• условиями (или законом) возникновения;

• типом, который определяет порядок обработки (дисциплину обслуживания) данного события;

• нулевой длительностью.

Обычно события подразделяют на две категории:

***события следования,*** которые управляют инициализацией процессов (или от­дельных работ внутри процесса);

***события изменения состояний*** (элементов системы или системы в целом).

*Как было отмечено, механизм событий используется в качестве основы постро­ения моделей, предназначенных для исследования причинно-следственных связей в системах при отсутствии временных ограничений. К таким задачам можно отне­сти, например, некоторые задачи по оценке надежности.*

Еще один способ имитационного моделирования систем основан на использо­вании понятия транзакта.

***Транзакт*** *—* это некоторое сообщение (заявка, на обслуживание), которое по­ступает извне на вход системы и подлежит обработке. В некоторых случаях, на­пример, при моделировании автоматизированных систем управления, более удоб­но проследить функционирование системы именно относительно алгоритма обработки транзакта. В рамках одной ИМ могут рассматриваться транзакты не­скольких типов. Каждый транзакт характеризуется соответствующим алгоритмом обработки и необходимыми для его реализации ресурсами системы. Учитывая это, прохождение транзакта по системе можно в некоторых случаях рассматривать как последовательную активизацию процессов, реализующих его обработку («обслу­живание заявки»).

В связи с упоминанием термина «обслуживание заявки» уместно вспомнить о существовании теории массового обслуживания. При разработке и исследовании имитационных мо­делей на основе транзактов целесообразно использовать методику и показатели, применяемые при анализе систем массового обслуживания.

### Тема 5. РАЗРАБОТКА ИМИТАЦИОННОЙ МОДЕЛИ

#### 5.1. Управление модельным временем

Приступая к изучению механизмов управления модельным временем, уместно поговорить о роли времени в имитационном моделировании. Ранее было отмечено, что имитационное моделирование представляет собой наблюдение за поведением системы в течение некоторого промежутка времени. Конечно, далеко не во всех статистических испытаниях фактор времени играет ведущую роль, а в некоторых и вообще может не рассматриваться. Но значительно больше таких за­дач, в которых оценка эффективности моделируемой системы напрямую связана с временными характеристиками ее функционирования. К ним относятся задачи по оценке производительности, некоторые задачи по оценке на­дежности, качества распределения ресурсов, а также все задачи, связанные с иссле­дованием эффективности процессов обслуживания. Характерной особенностью большинства практических задач является то, что скорость протекания рассматри­ваемых в них процессов значительно ниже скорости реализации модельного экспе­римента. Например, если моделируется работа авторемонтной мастерской в тече­ние недели, вряд ли кому-то придет в голову воспроизводить этот процесс в модели в таком же масштабе времени. А в ряде задач требуется именно реализация реального масштаба времени.

При разработке практически любой имитационной модели и пла­нировании проведения модельных экспериментов необходимо соотносить между собой три представления времени:

• реальное время, в котором происходит функционирование имитируемой системы;

• модельное (или, как его еще называют, системное) время, в масштабе которо­го организуется работа модели;

• машинное время, отражающее затраты времени ЭВМ на проведение ими­тации.

С помощью механизма модельного времени решаются следующие задачи:

1) отображается переход моделируемой системы из одного состояния в другое;

2) производится синхронизация работы компонент модели;

3) изменяется масштаб времени «жизни» (функционирования) исследуемой системы;

4) производится управление ходом модельного эксперимента.

5) моделируется квазипараллельная реализация событий в модели;

Приставка «квази» в данном случае отражает последовательный характер об­работки событий (процессов) в ИМ, которые в реальной системе возникают (про­текают) одновременно.

Необходимость решения последней задачи связана с тем, что в распоряжении исследователя находится, как правило, однопроцессорная вычислительная систе­ма, а модель может содержать значительно большее число одновременно работаю­щих подсистем. Поэтому действительно параллельная (одновременная) реализа­ция всех компонент модели невозможна. Даже если используется так называемая распределенная модель, реализуемая на нескольких узлах вычислительной сети, совсем необязательно число узлов будет совпадать с числом одновременно рабо­тающих компонент модели. Следует отметить, что реали­зация квазипараллельной работы компонент модели является достаточно сложной технической задачей. Некоторые возможные методы ее решения рассматриваются в следующем разделе.

Ранее были названы два метода реализации механизма модельного времени — с постоянным шагом и по особым состояниям.

Выбор метода реализации механизма модельного времени зависит от назначе­ния модели, ее сложности, характера исследуемых процессов, требуемой точности результатов и т. д.

При использовании метода ***постоянного шага*** отсчет системного времени ве­дется через фиксированные, выбранные исследователем интервалы времени. Со­бытия в модели считаются наступившими в момент окончания этого интервала. Погрешность в измерении временных характеристик системы в этом случае зави­сит от величины шага моделирования Δt.

Метод постоянного шага предпочтительнее, если:

• события появляются регулярно, их распределение во времени достаточно рав­номерно;

• число событий велико и моменты их появления близки;

• невозможно заранее определить моменты появления событий.

Данный метод управления модельным временем достаточно просто реализовать в том случае, когда условия появления событий всех типов в модели можно пред­ставить как функцию времени.

Пусть, например, событие состоит в том, что летящий самолет пересекает неко­торый воздушный рубеж, расстояние до которого равно *R.* Если самолет движется по прямой с постоянной скоростью *V,* то можно вычислять путь, пройденный само­летом, с интервалом времени Δ*t: S*=S+V·Δt. Соответственно событие считается наступившим, если выполняется условие S > *R,* а момент времени наступления со­бытия принимается равным *п • Δt,* где *п* — номер шага моделирования, на котором условие стало истинным.

Выбор величины шага моделирования является нелегким и очень важным делом. Универсальной методики решения этой проблемы не су­ществует, но во многих случаях можно использовать один из следующих под­ходов:

• принимать величину шага равной средней интенсивности возникновения со­бытий различных типов;

• выбирать величину Δt равной среднему интервалу между наиболее частыми (или наиболее важными) событиями.



• принимать величину шага равной средней интенсивности возникновения со­бытий различных типов;

• выбирать величину Δt равной среднему интервалу между наиболее частыми (или наиболее важными) событиями.

При моделировании ***по особым состояниям*** системное время каждый раз из­меняется на величину, строго соответствующую интервалу времени до момента наступления очередного события. В этом случае события обрабатываются в поряд­ке их наступления, а одновременно наступившими считаются только те, которые являются одновременными в действительности.

Метод моделирования по особым состояниям сложнее в реализации, так как для него требуется разработка специальной процедуры планирования событий (так называемого календаря событий).

Моделирование по особым состояниям целесообразно использовать, если:

• события распределяются во времени неравномерно или интервалы между ними велики;

• предъявляются повышенные требования к точности определения взаимного положения событий во времени;

• необходимо реализовать квазипараллельную обработку одновременных событий.

Дополнительное достоинство метода заключается в том, что он позволяет эко­номить машинное время, особенно при моделировании систем периодического дей­ствия, в которых события длительное время могут не наступать.

Таким образом:

* Выбор механизма изменения модельного времени определяет и технологию реализации имитационной модели.
* На выбор метода моделирования влияет целый ряд факторов, однако определяющим является тип моделирующей системы: для дискретных систем, события в которых распределены во времени неравномерно, более удобным является изменение модельного времени по особым состояниям.

Если в модели должны быть представлены компоненты реальной системы, работа которых измеряется в разных единицах времени, то они должны быть предварительно приведены к единому масштабу.

#### 5.2. Моделирование параллельных процессов

Практически любая более или менее сложная система имеет в своем составе компоненты, работающие одновременно, или, как принято говорить на языке тех­ники, параллельно. Параллельно работающие подсистемы могут вза­имодействовать самым различным образом, либо вообще работать независимо друг от друга. Способ взаимодействия подсистем определяет вид параллельных процес­сов, протекающих в системе. В свою очередь, вид моделируемых процессов влияет на выбор метода их имитации.

##### 5.2.1.Виды параллельных процессов в сложных системах

***Асинхронный параллельный процесс*** *—* такой процесс, состояние которого не зависит от состояния другого параллельного процесса (ПП).

Пример асинхронных ПП из области вычислительной техники: выполнение вычислений процессором и вы­вод информации на печать.

***Синхронный ПП*** *—* такой процесс, состояние которого зависит от состояния взаимодействующих с ним ПП.

Пример синхронного ПП: работа торговой организации и доставка товара со склада (нет товара — нет торговли).

Один и тот же процесс может быть синхронным по отношению к одному из ак­тивных ПП и асинхронным по отношению к другому. Так, при работе вычисли­тельной сети по технологии «клиент-сервер» каждый из узлов сети синхронизиру­ет свою работу с работой сервера, но не зависит от работы других узлов.

***Подчиненный ПП*** *—* создается и управляется другим процессом (более высоко­го уровня). Весьма характерным примером таких процессов является ведение боевых дей­ствий подчиненными подразделениями.

***Независимый ПП—*** не является подчиненным ни для одного из процессов. Скажем, после запуска неуправляемой зенитной ракеты ее полет можно рас­сматривать как независимый процесс, одновременно с которым самолет ведет бое­вые действия другими средствами.

Способ организации параллельных процессов в системе зависит от физической сущности этой системы.

Остановимся несколько подробнее на особенностях реализации параллельных процессов в вычислительных системах (ВС). Это обусловлено следующей причи­ной.

Разработка и использование любой ИМ предполагает ее программную реализа­цию и исследование с применением ВС. Поэтому для реализации моделей, имити­рующих параллельные процессы, в некоторых случаях применимы механизмы, используемые при выполнении параллельных вычислений.

Вместе с тем, реализация параллельных процессов в ВС имеет свои особенности:

• на уровне задач вычислительные процессы могут быть истинно параллельны­ми только в многопроцессорных ВС или вычислительных сетях;

• многие ПП используют одни и те же ресурсы, поэтому даже асинхронные ПП в пределах одной ВС вынуждены согласовывать свои действия при обращении к общим ресурсам;

• в ВС дополнительно используется еще два вида ПП: родительский и дочер­ний ПП; особенность их состоит в том, что процесс-родитель не может быть завер­шен, пока не завершатся все его дочерние процессы.

В силу перечисленных особенностей для организации взаимодействия парал­лельных процессов в ВС используются три основных подхода:

• на основе «взаимного исключения»;

• на основе синхронизации посредством сигналов;

• на основе обмена информацией (сообщениями).

***«Взаимное исключение»*** предполагает запрет доступа к общим ресурсам (об­щим данным) для всех ПП, кроме одного, на время его работы с этими ресурсами (данными).

***Синхронизация*** подразумевает обмен сигналами между двумя или более про­цессами по установленному протоколу. Такой «сигнал» рассматривается как некоторое событие, вызывающее у получившего его процесса соответствующие действия.

Часто возникает необходимость передавать от одного ПП другому более под­робную информацию, чем просто «сигнал-событие». В этом случае процессы со­гласуют свою работу на основе обмена сообщениями.

Перечисленные механизмы реализуются в ВС на двух уровнях — системном и прикладном.

Механизм взаимодействия между ПП на системном уровне определяется еще на этапе разработки ВС и реализуется в основном средствами операционной систе­мы (частично — с использованием аппаратных средств).

На прикладном уровне взаимодействие между ПП реализуется программистом средствами языка, на котором разрабатывается программное обеспечение.

Наибольшими возможностями в этом отношении обладают так называемые язы­ки реального времени (ЯРВ) и языки моделирования.

Языки реального времени — это языки, предназначенные для создания про­граммного обеспечения, работающего в реальном масштабе времени, например для разработки различных автоматизированных систем управления (предприя­тием, воздушным движением и т. д.). К ним, в частности, относятся: язык *Ада,* язык *Модула* и практически единственный отечественный язык реального време­ни — *Эль-76* (использовавшийся в многопроцессорных вычислительных комп­лексах семейства «Эльбрус»).

##### 5.2.2. Методы описания параллельных процессов в системах и языках моделирования

Языки моделирования по сравнению с языками реального времени требуют от разработчика значительно менее высокого уровня подготовки в области програм­мирования, что обусловлено двумя обстоятельствами:

• во-первых, средства моделирования изначально ориентированы на квазипа­раллельную обработку параллельных процессов;

• во-вторых, механизмы реализации ПП относятся, как правило, к внутренней орга­низации системы (языка) моделирования и их работа скрыта от программиста.

В практике имитационного моделирования одинаково широко используются как процессно-ориентированные языки (системы) моделирования, например *SIMULA,* так и языки, ориентированные на обработку транзактов (например, язык *GPSS).* В тех и других используются аналогичные методы реализации квазипарал­лелизма, основанные на ведении списков событий. В процессно-ориентированных системах используются списки событий следования, а в транзактных системах — списки событий изменения состояний.

Современные языки и системы моделирования, ориентированные на использо­вание в среде многозадачных операционных систем типа Windows, частично используют их механизмы управления процессами, что делает их применение еще более эффективным. В пакете MATLAB также имеется собственный язык модели­рования, и к нему в полной мере можно отнести сказанное выше. Тем не менее во многих случаях оказывается полезным знание общего механизма реализации ПП в языках моделирования.

Рассмотрим его применительно к моделированию на основе тракзактов.

В этом случае под событием понимается любое перемещение транзакта по сис­теме, а также изменение его состояния (обслуживается, заблокирован и т. д.).

Событие, связанное с данным транзактом, может храниться в одном из следую­щих списков.

***Список текущих событий.*** В этом списке находятся события, время наступле­ния которых меньше или равно текущему модельному времени. События с «мень­шим» временем связаны с перемещением тех транзактов, которые должны были начать двигаться, но были заблокированы.

***Список будущих событий.*** Этот список содержит события, время наступле­ния которых больше текущего модельного времени, то есть события, которые дол­жны произойти в будущем (условия наступления которых уже определены — на­пример, известно, что транзакт будет обслуживаться некоторым устройством 10 единиц времени).

***Список прерываний.*** Данный список содержит события, связанные с возобнов­лением обработки прерванных транзактов. События из этого списка выбираются в том случае, если сняты условия прерывания.

В списке текущих событий транзакты расположены в порядке убывания при­оритета соответствующих событий; при равных приоритетах — в порядке поступ­ления в список.

Каждое событие (транзакт) в списке текущих событий может находиться либо в активном состоянии, либо в состоянии задержки. Если событие активно, то соот­ветствующий транзакт может быть продвинут по системе; если продвижение не­возможно (например, из-за занятости устройства), то событие (и транзакт) пере­водится в состояние задержки.

Как только завершается обработка (продвижение) очередного активного транзакта, просматривается список задержанных транзактов, и ряд из них пе­реводится в активное состояние. Процедура повторяется до тех пор, пока в спис­ке текущих событий не будут обработаны все активные события. После этого просматривается список будущих событий. Модельному времени присваивает­ся значение, равное времени наступления ближайшего из этих событий. Данное событие заносится в список текущих событий. Затем просматриваются осталь­ные события списка. Те из них, время которых равно текущему модельному вре­мени, также переписываются в список текущих событий. Просмотр заканчива­ется, когда в списке остаются события, времена которых больше текущего модельного времени.

В качестве иллюстрации к изложенному рассмотрим небольшой пример.

*►Пусть в систему поступают транзакты трех типов, каждый из которых обслу­живается отдельным устройством. Известны законы поступления транзактов в систему и длительность их обслуживания. Таким образом, в системе существуют три параллельных независимых процесса (PI, P2, РЗ).*

*Временная диаграмма работы системы при обслуживании одного транзакта каж­дого типа показана на рис.2.7.*

*На рисунке события, относящиеся к процессу Р1, обозначены как С1i, относя­щиеся к Р2 и к РЗ — соответственно как С2i. и СЗi. Моменты времени tвх и tвых соот­ветствуют началу и окончанию обслуживания транзакта.*



*Для каждого процесса строится своя цепь событий, однако списки событий являются общими для всей модели. Формирование списков начинается с за­полнения списка будущих событий. Как было отмечено выше, в этот список помещаются события, время наступления которых превышает текущее значе­ние модельного времени. Очевидно, что на момент заполнения списка время наступления прогнозируемых событий должно быть известно. На первом шаге tм=0, и в список будущих событий заносятся события С11, С21, С31. Затем событие с наименьшим временем наступления — С11 — переносится в список те­кущих событий; если одновременных с ним событий нет, то оно обрабатывается и исключается из списка текущих событий. После этого вновь корректирует­ся список будущих событий и т.д., пока не истечет заданный интервал моде­лирования.*

*Динамика изменения списков текущих и будущих событий для рассмотренного примера отражена в приведенной ниже таблице.*

|  |  |  |
| --- | --- | --- |
| ***t*** | ***Список текущих событий*** | ***Список будущих событий*** |
| ***0*** | *0* | *C11, C21,C31* |
| ***t11*** | *C11* | *C21, C31, C12* |
| ***t21*** | *C21* | *C31, C12, C22* |
| ***t31*** | *C31* | *C12, C22, C32* |
| ***t12*** | *C12* | *C22, C32, C13* |
| ***t22*** | *C22* | *C32, C13, C23* |
| ***t32*** | *C32* | *C13, C23, C33* |
| ***t13*** | *C13* | *C23, C33* |
| ***t23*** | *C23, C33* |  |

*◄*

Многие авторы книг по имитационному моделированию считают, что знание механизма ведения списков событий просто необходимо разра­ботчику модели; умение проследить в динамике цепь происходящих в модели собы­тий, во-первых, повышает уверенность создателя модели в том, что она работает пра­вильно и, во вторых, существенно облегчает процесс отладки и модификации модели.

##### 5.2.3. Применение сетевых моделей для описания параллельных процессов

Этапу программной реализации модели (т. е. ее опи­санию на одном из языков программирования) должен предшествовать так на­зываемый этап алгоритмизации. Другими словами, прежде чем превратить имитационную модель в работающую программу, ее создатель должен воспользоваться каким-то менее формальным и более наглядным сред­ством описания логики работы будущей программы. Это требова­ние не является обязательным, т.к. при наличии достаточного опыта программа не очень сложной модели может быть написана сразу. Однако при моделировании более сложных систем даже опытные разработчики бывают вынуждены немного «притормозить» на этапе алгоритмизации. Для описания логики работы модели могут быть использова­ны различные средства: либо русский язык (устный или письменный), либо тра­диционные схемы алгоритмов, либо какие-то другие «подручные» средства. Первые два варианта являются, как правило, наи­более знакомыми и наиболее часто используемыми. Однако такие схемы совершенно не приспо­соблены для описания параллельных процессов.

Одним из наиболее элегантных и весьма распространенных средств описания параллель­ных процессов — описание ***сетями Петри.*** Рассмотрим те основные сведения, кото­рые необходимы с точки зрения реализации технологии имитационного моделирования параллельных процессов.

Одно из основных достоинств аппарата сетей Петри заключается в том, что они могут быть представлены как в графической форме (что обеспечивает наглядность), так и в аналитической (что позволяет автоматизировать процесс их анализа).

При графической интерпретации сеть Петри представляет собой граф осо­бого вида, состоящий из вершин двух типов — *позиций* и *переходов,* соединен­ных ориентированными дугами, причем каждая дуга может связывать лишь раз­нотипные вершины (позицию с переходом или переход с позицией). Вершины-позиции обозначаются кружками, вершины-переходы — черточками. С содержательной точки зрения, переходы соответствуют событиям, присущим исследуемой системе, а позиции — условиям их возникновения. Таким образом, совокупность переходов, позиций и дуг позволяет описать причинно-следствен­ные связи, присущие системе, но в статике. Чтобы сеть Петри «ожила», вводят еще один вид объектов сети — так называемые *фишки,* или метки позиций. Пе­реход считается активным (событие может произойти), если в каждой его вход­ной позиции есть хотя бы одна фишка. Расположение фишек в позициях сети называется ***разметкой сети*** (пример перемещения фишек по сети приведен на рис.5.6).



В аналитической форме сеть Петри может быть представлена следующим образом:

*P=(B,D,I,0,M),*

где ***В =***{bi} — конечное непустое множество позиций;

***D***= ***{di}*** — конечное непустое множество переходов;

***I : B*х*D ->*** *0,1* — входная функция (прямая функция инцидентности), которая для каждого перехода задает множество его входных позиций;

***О : D*x*B* ->** 0,1 — выходная функция (обратная функция инцидентности), кото­рая для каждого перехода задает множество его входных позиций;

***М*** *—* функция разметки сети, ***М : В ->***0, 1, 2,... — ставит каждой позиции сети в соответствие неотрицательное целое число.

С учетом введенных обозначений необходимое условие срабатывания перехода *dj* может быть записано следующим образом:

***∀bi ∈ I(di) {M(bi)≥1}***

(*для всех входных позиций разметка должна быть >1*).

Срабатывание перехода *dj* изменяет разметку сети *М(В)* на разметку *М’(В)* по следующему правилу:

***M’(B)=M(B)-I(dj)+O(dj),***

то есть переход *dj* изымает по одной метке из каждой своей входной позиции и добав­ляет по одной метке в каждую из выходных позиций. Смену разметки обозначают так:

***dj***

***Mo⎪⎯M’***

Входная и выходная функции сети Петри (***I*** и ***О***) позволяют описать любую сеть с помощью двух матриц размера *т х п* (матриц входных и выходных позиций), имеющих следующую структуру:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | d1 | d2 | ... | dj | ... | dn |
| b1 | 0 | 1 | ... | 0 | ... | 0 |
| b2 | 1 | 1 | ... | 0 | ... | 1 |
| ... | ... | ... | ... | ... | ... | ... |
| bj | 0 | 1 | ... | 0 | ... | 1 |
| ... | ... | ... | ... | ... | ... | ... |
| bm | 1 | 0 | ... | 1 | ... | 0 |

Основные направления анализа сети Петри следующие:

1. *Проблема достижимости:* в сети Петри с начальной разметкой М0 требуется определить, достижима ли принципиально некоторая разметка *М'* из M0.

С точки зрения исследования моделируемой системы, эта проблема интер­претируется как проблема достижимости (реализуемости) некоторого состоя­ния системы.

2. *Свойство живости.* Под живостью перехода понимают возможность его сра­батывания в данной сети при начальной разметке М0. Анализ модели на свойство живости позволяет выявить невозможные состо­яния в моделируемой системе (например, неисполняемые ветви в программе).

3. *Безопасность сети.* Безопасной является такая сеть Петри, в которой ни при каких условиях не может появиться более одной метки в каждой из позиций. Для исследуемой системы это означает возможность функционирования ее в стационарном режиме. На основе анализа данного свойства могут быть определе­ны требования к буферным накопителям в системе.

Итак, достоинства сетей Петри заключаются в том, что они:

1) позволяют моделировать ПП всех возможных типов с учетом вероятных кон­фликтов между ними;

2) обладают наглядностью и обеспечивают возможность автоматизирован­ного анализа;

3) позволяют переходить от одного уровня детализации описания системы к другому (за счет раскрытия/закрытия переходов).

Вместе с тем, сети Петри имеют ряд недостатков, ограничивающих их возмож­ности. Основной из них — время срабатывания перехода считается равным 0, что не позволяет исследовать с помощью сетей Петри временные характеристики мо­делируемых систем.

В результате развития аппарата сетей Петри был разработан ряд расширений. Одно из наиболее мощных — так называемые .Е-сети (evaluation — «вычисления», «оценка») — «оценочные сети».

В отличие от сетей Петри, в E-сетях:

1) имеются несколько типов вершин-позиций: простые позиции, позиции-оче­реди, разрешающие позиции;

2) фишки (метки) могут снабжаться набором признаков (атрибутов);

3) с каждым переходом может быть связана ненулевая задержка и функция пре­образования атрибутов фишек;

4) введены дополнительные виды вершин-переходов.

5) в любую позицию может входить не более одной дуги и выходить также не более одной.

В связи с этим любой переход может быть описан тройкой параметров:

***dj=(S,t(dj),ρ(dj)),***

где S — тип перехода,

*t(dj), —* функция задержки,

*ρ(dj)* — функция преобразования атрибутов.

Особенности E-сетей существенно расширяют их возможности для моделирования дискретных систем вообще и параллельных процессов в част­ности. Технология моделирования систем в виде Е-сетей может быть реализована с помощью инструмента SIMULINK, вхо­дящего в состав пакета М ATLAB.

### Тема 6. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ФАКТОРОВ

Имитационная модель позволяет исследовать поведение различных систем с учетом влияния случайных факторов. Эти факторы в зависимости от их природы могут быть отражены в модели как случайные события, случайные величины (дис­кретные или непрерывные), или как случайные функции (процессы).

В основе всех методов и приемов моделирования случайных факторов лежит использование базовой случайной величины -- случайных чисел, имеющих равномерное распределение на интер­вале [0; 1].

#### 6.1.Построение датчиков БСВ

##### 6.1.1. Датчики БСВ

***Базовой*** случайной величиной (БСВ) в статистическом моделировании называют непрерывную случайную величину *Z*, равномерно распределенную на интервале (0,1). Ее плотность распределения вероятностей имеет вид:



Математическое ожидание M[z] и дисперсия D[z] БСВ составляют



соответственно.

БСВ моделируется на ЭВМ с помощью *датчиков* БСВ. Датчик БСВ  это устройство или программа, выдающая по запросу одно или несколько независимых значений *z1* , ..., *zn* БСВ.

Датчики БСВ могут быть трех типов: табличные, физические и программные.

***Табличный*** датчик БСВ  это просто таблица случайных чисел. Основной недостаток такого датчика  ограниченное количество случайных чисел в таблицах. А в статистическом эксперименте часто требуется не ограниченное заранее их количество.

***Физический*** датчик БСВ  это специальное радиоэлектронное устройство в ЭВМ, содержащее источник электронного шума. Шум преобразуется в случайные числа с распределением. Недостатки физического датчика БСВ: невозможность повторения каких-либо ранее полученных реализаций *z1*, ... , *zn* без их предварительной записи в память ЭВМ, схемная нестабильность и сложность тиражирования датчика.

***Программный*** датчик БСВ обычно вычисляет значения *z1*, *z2*,..., по какой-либо рекуррентной формуле типа

*zi = f ( zn),*

при заданном стартовом значении *z0*.

Заданное значение *z0* полностью определяет всю последовательность реализаций *z1*, *z2*,..., поэтому *z* часто называют *псевдослучайной* величиной. Но ее статистические свойства идентичны свойствам "чисто случайной" последовательности, что и обеспечивает успех статистического моделирования.

Программный датчик БСВ имеет следующие преимущества: простота создания датчика, простота применения, простота тиражирования, надежность, быстродействие, высокая точность достижения необходимых статистических свойств, сравнимая с точностью представления вещественных чисел, компактность, повторяемость, когда это нужно, любых последовательностей случайных значений без их предварительного запоминания.

В дальнейшем мы будем рассматривать только программные датчики БСВ.

Имея датчик БСВ *Z*, можно промоделировать любые случайные факторы: непрерывные или дискретные случайные величины (как простые, так и многомерные), случайные события, случайные процессы и поля и т.д. Для этого достаточно соответствующим образом преобразовать последовательность *z1*, *z2*, ... . Поэтому БСВ *Z* и называют базовой.

Теоретически в качестве базовой можно было бы взять почти любую случайную величину. Использование СВ *z* с распределением обусловлено технологическими соображениями: простотой и экономичностью датчика, простотой преобразования *Z* в другие случайные факторы, относительной простотой тестирования датчика.

##### 6.1.2. Метод середины квадрата

Метод середины квадрата предложен для получения псевдослучайных чисел Д. фон Нейманом в 1946 г. Один из вариантов этого метода заключается в следующем.

1. Возьмем произвольное *n*-разрядное число.
2. Возведем полученное число в квадрат и, если необходимо, добавим к результату слева нули до 2*n*-разрядного числа.
3. Возьмем четыре цифры из середины 2*n*-разрядного в качестве нового случайного *n*-разрядного числа.
4. Если нужны еще случайные числа, то перейдем к пункту 2.

*Например*, если взять в качестве начального числа 1994, то из него получается следующая последовательность псевдослучайных чисел: 9760 2576 6357 4114 9249 5440 5936 2360 5696 4444 7491 1150 3225 4006 0480 2304 3084 5110 1121 2566 ...

Сам по себе метод середины квадрата не получил широкого распространения, так как выдает "больше чем нужно малых значений". Но открытый в нем принцип используется во многих, если не во всех, более поздних датчиках БСВ. Этот принцип состоит в вырезании нескольких цифр из результата какой-либо операции над числами.

##### 6.1.3. Мультипликативный конгруэнтный метод

Так называемый мультипликативный конгруэнтный датчик БСВ задается двумя параметрами: модулем *m* и множителем *k*. Обычно это достаточно большие целые числа.

При заданных *m*, *k* числа *z1*, *z2*, ..., вычиcляются по рекуррентной формуле:

*Ai* = (*kAi -1*) mod *m*, *i* = 1, 2,...,                      (6.1)

*zi = Ai / m*,

где *m * модуль,

*k * множитель,

*A0*  начальное значение,

mod  операция вычисления остатка от деления *kAi -1* на *m*.

Таким образом, *A1* определяется как остаток от деления  *kA0* на *m*; *A2* - как остаток от деления *kA1* на *m* и т.д. Поскольку все числа *Ai*  это остатки от деления на *m*, то 0   *Ai* < *m*. Разделив последнее неравенство на *m*, видим, что 0  *Ai / m*< 1, т. е. 0  *zi* <1.

Из неравенства 0  *Ai* < *m* вытекает также, что датчик (6.1) дает периодическую последовательность *Ai*. Действительно, число всех возможных остатков от 0 до *m* - 1 равно *m* и, рано или поздно, на каком-то шаге *i* обязательно появится значение *Ai*, уже встречавшееся ранее. С этого момента последовательность *Ai* “зациклится".

Длина периода *T* будет не больше *m -* 1. Например, если встретится остаток *Ai*= 0, то далее, согласно (4.1), будет *Ai+ 1* = 0, *Ai+ 2* = 0, ... , т.е. длина периода *T* = 1. Ненулевых же остатков в интервале 0 *Ai* < *m* всего *m -* 1, и, если все они войдут в период, будет *T* = *m -* 1. Это имеет место, например, при *m* = 13, *k* = 7; в этом случае ряд *Ai* выглядит так:

1, 7, 10, 5, 9, 11, 12, 6, 3, 8, 4, 2,  1, 7,... .  
\\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

*T* = *m -* 1 = 12

Поскольку в качестве случайной можно использовать лишь подпоследовательность *Ai* внутри одного периода, то параметры датчика выбирают так, чтобы длина периода *T* была максимальной. С учетом ограничения *T m -* 1 модуль *m* берут максимально возможным. Чтобы упростить вычисление остатков по (6.1), для двоичных ЭВМ часто берут *m* = 2*n*. Рекомендуется также брать достаточно большой множитель *k*, причем взаимно простой с *m*.

В можно найти подробные рекомендации по выбору параметров *m*, *k* и начального значения *A0* . Заметим, однако, что в настоящее время не известны правила, которые гарантировали бы высокое качество датчика без его специального статистического тестирования.

Датчик (4.1) называют мультипликативно-конгруэнтным потому, что он использует две основные операции  умножение (англ. multiplication) и вычисление остатка (в теории чисел  получение конгруэнтного числа). Можно было бы поэтому перевести его название и как "множительно-остатковый датчик".

Обратим внимание также и на то, что операция вычисления остатка воплощает здесь упоминавшийся в п. 4.1.2 неймановский принцип вытаскивания цифр. Это становится очевидным, если записывать числа в системе счисления с основанием *m*. Тогда операция *X* mod *m* означает выбор последней цифры из числа *X*. Для *m* = 2*n* операция *X* mod *m* означает также выделение последних *n* цифр из двоичной записи числа *X*.

В качестве примера рассмотрим таблицу параметров датчиков, предлагаемых в некоторых публикациях и программных продуктах.

|  |  |  |
| --- | --- | --- |
| Место использования датчика  (программный продукт или публикация) | модуль  *m* | множитель  *k* |
| Язык моделирования СИМУЛА | 2*35* | 5*16* |
| Пакеты LLRANDOM, IMSL | 2*31*  1 (простое число) | 16807 |
| Язык моделирования SIMSCRIPT | 2*31*  1 | 63036001 |

#### 6.2. Характеристики датчиков базовых случайных величин

Практика показывает, что результаты имитационного моделирования существенно зависят от качества используемых последовательностей псевдослучайных чисел. Поэтому используемые в имитационной модели генераторы случайных чисел должны пройти тесты на пригодность. Основные анализируемые характеристики генерируемых датчиком последова­тельностей:

• равномерность;

• стохастичность (случайность);

• независимость.

Рассмотрим методы проведения такого анализа, наиболее часто применяемые на практике.

##### 6.2.1. Тестирование равномерности

Обозначим равномерное распределение вероятностей на интервале (0,1) через *R*[0,1]. Тогда утверждение, что БСВ *Z* имеет распределение *R*[0,1], можно кратко записать в виде *z* ~ *R*[0,1].

С помощью статистических тестов проверяют два свойства датчика, делающих его точной моделью идеальной БСВ,  это *равномерность* распределения чисел *Zi*, выдаваемых датчиком на интервале (0,1), и их статистическая *независимость*. При этом числа *zi* рассматривают как реализации некоторой **СВ.**, т.е. как статистическую выборку.

Достаточно простым методом проверки равномерности распределения является частотный тест. Он основан на законе больших чисел и выполняется по следующему алгоритму.

1. Разобьем интервал (0,1) на *K* равных отрезков (например, *K* = 10).

2. Сгенерируем *n* чисел *z1*,..., *zn* с помощью тестируемого датчика БСВ (например, *n* = 100).

3. Подсчитаем, сколько чисел попало в каждый из *k* отрезков, т.е. найдем числа попаданий *n1*,...,*nk*.

1. Рассчитаем относительные частоты попаданий в отрезки:



5. Построим гистограмму частот на *K* отрезках интервала (0,1).

6. Повторим действия (2)  (5) для большего значения *n* (например, для *n* =10 000).

7. Оценим по полученным гистограммам сходимость каждой частоты к вероятности *p* = 1/*K* того, что БСВ попадет в *i*-й отрезок. Согласно закону больших чисел должно быть

 (6.2.)

Это значит, что высоты столбиков во второй гистограмме должны в целом быть ближе к уровню 1/*K*, чем в первой.

Тестирование датчика на равномерность можно совместить с оцениванием **математического ожидания *m\**** и дисперсии S2. Оценки ***m\**** и S2 рассчитываются соответственно по формулам:

 (6.3)

 (6.4)

С ростом *n* оценки  и  должны сходиться по вероятности к точным значениям *M*(*z*) = 1/2, *D*(*z*) = 1/12 = 0.08333... .

##### 6.2.2. Тестирование стохастичности

Рассмотрим один из основных методов проверки – метод комбинаций. Суть его сводится к следующему. Выбирают достаточно большую последовательность случайных чисел xi и для нее определяют вероятность появления в каждом из xi ровно j единиц. При этом могут анализироваться как все разряды числа, так и только l старших. Теоретически закон появления j единиц в l разрядах двоичного числа может быть описан как биномиальный закон распреде­ления (исходя из независимости отдельных разрядов).

Тогда при длине выборки N ожидаемое число появлений случайных чисел xi с j единицами в проверяемых l:

**

Для полученной последовательности определяется эта же характеристика. Про­верка соответствия реального значения теоретическому выполняется с помощью одного из статистических критериев согласия.

##### 6.2.3. Тестирование независимости

Простейшую проверку статистической независимости реализаций *z1*, *z2*, ..., можно осуществить, оценивая корреляцию между числами *zi* и *zi+s*, отстоящими друг от друга на шаг *s* >1.

Для вывода формулы, по которой можно рассчитать коэффициент корреляции чисел *zi* и *zi+ s* , рассмотрим две произвольные **с.в**. *x*, *y*. Коэффициент корреляции определяется для них формулой:

                 (6.5)

С ростом *n* оценка *R*' должна приближаться к нулю, в противном случае датчик БСВ не отвечает требованию независимости.

Конечно, если *R*' сходится к нулю, то это еще не гарантирует наличие независимости, но все же один из тестов оказывается успешно выдержанным. При желании всегда можно продолжить испытания датчика другими методами.

Еще одна важная характеристика датчика СЧ — **длина отрезка периодичности *L****.* Если в основу работы датчика положен мультипликативный метод, то оценить L несложно: она определяется величиной константы *М.*

#### 6.3. Случайные события и их имитация

##### 6.3.1.Имитация случайного события

Пусть некоторое событие А происходит с вероятностью . Требуется воспроизвести факт наступления события А. Поставим в соответствие событию А событие В, состоящее в том, что *х* меньше либо равно, где х здесь и в дальнейшем – случайное число (СЧ) с равномерным на интервале (0,1) законом распределения. Вычислим вероятность события В:



Таким образом, события А и В являются равновероятными. Отсюда следует процедура имитации факта появления события А. Она сводится к проверке неравенства  меньше, либо равно Р, а алгоритм заключается в следующем:

1. С помощью датчика случайных чисел (СЧ) получают СЧ *Х*;

2. Проверяют выполнение неравенства Х меньше, либо равно ;

3. Если оно выполняется, то событие А – произошло, если нет – то произошло 

##### 6.3.2. Имитация сложного события

Имитация сложного события, состоящего, например, из двух независимых элементарных событий А и В, заключается в проверке неравенств:

,

где  и – вероятности событий А и В, а *х1* и *х2* – СЧ с равномерным законом распределения.

В зависимости от исхода проверки неравенств (аналогично алгоритму 4.2.1.) делается вывод какой из вариантов:

имеет место.

##### 6.3.3. Имитация сложного события, состоящего из зависимых событий.

В случае, когда сложное событие состоит из элементарных зависимых событий А и В имитация сложного события производится с помощью проверки следующих неравенств:

В зависимости от того, какая из этих четырех систем неравенств выполняется, делается вывод о том, какой из этих четырех возможных исходов  имеет место.

В качестве исходных данных задаются, и условная вероятность , вероятность  может быть вычислена. По формуле полной вероятности:

,

где

, отсюда легко выразить 

##### 6.3.4. Имитация событий, составляющих полную группу

Пусть событие Аi (i=1,n) составляют полную группу, тогда их вероятности Рi, таковы что:



Имитация факта появления одного из событий Аi (i=1,n) сводится к проверке следующих неравенств:



Выполнение К-го неравенства эквивалентно выполнению события АК. Описанный алгоритм называют иногда алгоритмом “розыгрыша по жребию”. Его можно интерпретировать как установление номера К-го отрезка длинной РK, на который пало СЧ х, при условии разбиения отрезка единичной длины на отрезки с длинами P1,P2,...Pn (рис 4.3.)

ris29

Рис. 4.3

#### 6.4. Имитация непрерывных случайных величин

##### 6.4.1. Метод обратной функции

Пусть непрерывная случайная величина Y задана своим законом распределения:

,

где *f(y)*– плотность распределения вероятностей, а *F(y)-* функция распределения вероятностей. Доказано, что случайная величина



распределена равномерно на интервале (0,1).

Отсюда следует, что искомое значение y может быть определено из уравнения:

 (6.8)

которое эквивалентно уравнению:

 (6.9)

где y – значение случайной величины Y, a x – значение СВ X. Решение уравнения (4.9) можно записать в общем виде через обратную функцию



Основной недостаток метода заключается в том, что интеграл (4.8) не всегда является берущимся, а уравнение (4.9) не всегда решается аналитическими методами.

##### 6.4.2. Метод Неймана (режекции)

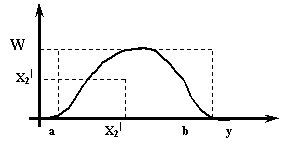
Метод Неймана, так же как метод обратной функции, является методом, позволяющим получить значения СВ в соответствии с заданным законом распределения. Этот метод является достаточно универсальным он применим для моделирования всех СВ, значения которых не выходят за пределы ограниченного интервала (a,b), а также для СВ, законы распределения которых можно аппроксимировать усеченными.

Метод Неймана состоит в следующем:

С помощью датчика случайных чисел получают пару чисел, распределенных равномерно на (0,1) x1 и x2.

Путем преобразований (по методу обратной функции получают два числа x1\* и x2\*, равномерно распределенных соответственно на интервалах (a,b) и (o,w), то есть

 и , где 



Из точек с координатами x1\* и x2\* выбирают те, которые попали “под колокол” функции *f(x)*, то есть те точки, для которых f(x1\*)<x2\*.

Если условие выполнено, то искомое значение y полагают равным x1\*.

##### 6.4.3. Алгоритм получения значения нормально распределенной случайной величины.

Нормальное распределение является наиболее часто встречающимся. Функция плотности распределения вероятностей для него имеет вид:



где *m* – математическое ожидание, а σ2– дисперсия. Согласно центральной предельной теореме теории вероятностей



распределена асимптотически нормально, если распределены одинаково.

Для практического получения значений *X* в качестве и выбирают равномерно распределенные случайные величины. При этом наиболее часто используют преобразование

 (4.10)

где *xi* – равномерно распределенные на (0,1) случайные числа. При к=12 формула приобретает вид наиболее удобной для расчетов, но она дает достаточно точные результаты уже для k=3,4. Формула (4.10) верна для центрированной (m=0) и нормированной ( =1) случайной величины.

Для получения y\*, распределенного нормально с произвольными m и σ, пользуются дополнительно преобразованием

y\*=m+σy (4.11)

#### 6.5. Алгоритмы получения значений систем случайных величин (случайных векторов).

##### 6.5.1. Метод аналитических преобразований.

Пусть системы непрерывных случайных величин (*x1, x2, …, xn*) задана условными законами распределения *xi* (*i*=1,*n*). По теореме умножения плотностей распределения: совместная функция плотности распределения вероятностей

*f(x1, x2, . . . xn)=f1(x1) f2(x2|x1) f3(x3|x1x2) . . . f1(xn| x1,x2, . . ., xn-1).*

Для системы двух случайных величин (*x1,x2*), алгоритм получения вектора ее значений сводится к следующему:

Вычисление частной функции плотности для *x1*:



Получение значения *X1* в соответствии с *f1*(*x*1) согласно любому методу, например, одному из описанных в предыдущем разделе.

Вычисление частной функции плотности для второй компоненты *x2* системы. Она может быть получена на основании теоремы умножения законов распределения:



Получение x2 – значения СВ X2 любым известным методом в соответствии с найденным законом ее распределения.

Алгоритм может быть обобщен для любого *n*. Однако, практические работы, выполняемые по этому методу, связаны с большими вычислительными трудностями, за исключением тех редких случаев, когда интегралы берутся. Поэтому разработаны другие методы, позволяющие решать задачу получения значений системы непрерывных случайных величин.

##### 6.5.2. Метод разложения по координатным случайным величинам.

Пусть СНСВ задана в рамках теории корреляций: математическими ожиданиями компонент (m1, m2, . . . mn) и матрицей корреляционных моментов:

, 

Доказано, что  можно получить с помощью их разложения по координатам СВ xi:

 (4.12)

где *xi* - некоррелированные, центрированные, нормированные нормально распределенные СВ.

Коэффициенты  могут быть достаточно просто получены решением системы уравнений:

 (4.13)

Алгоритм получения значений СНСВ сводится к следующему:

* Решение системы нелинейных уравнений (4.13).
* Получение *n* значений *yi* нормированных, центрированных СВ, распределенных нормально.
* Вычисление xi i=(1,…,n) значений СВ, образующих систему непрерывных случайных величин в соответствии с (4.12).

##### 6.5.3. Алгоритм получения значений системы дискретных случайных величин

Дискретный двумерный вектор CDCB задается двумерным законом распределения, т.е.

а) матрицей вероятностей , где Pij – вероятность совместного появления i-ого и j-ого значений соответственной первой и второй компоненты, причем:

.

б) двумя векторами возможных значений первой и второй компоненты {Ai}, {Bi}, .

Получение значений двумерной дискретной системы случайных величин может осуществляться по следующему алгоритму.

Вычисляют суммы , , .

Если *Х* - равномерно распределенное случайное число из интервала (0,1) такое, что , то считают, что *x1* компонента двумерной дискретной случайной величины получила k-ое значение.

Выбирают k-ую строку , вычисляют .

Если вновь полученное с помощью датчика случайных чисел *Х* такое, что вторая компонента получила S-е значение.

Замечание: В алгоритме используется правило “розыгрыша по жребию”, однако надо иметь в виду, что .

### Тема 7. ПЛАНИРОВАНИЕ МОДЕЛЬНЫХ ЭКСПЕРИМЕНТОВ

При планировании эксперимента необходимо учитывать следующие факторы.

* необходимо определить, какой режим работы системы исследуется: стационарный (установившийся) или нестационарный;
* необходимо знать, в течение какого промежутка времени следует наблюдать за поведением (функционированием) системы;
* необходимо знать, какой объем испытаний (т.е повторных экспериментов) сможет обеспечить требуемую точность оценок ( в статическом смысле) исследуемых характеристик системы.

Планирование модельных экспериментов преследует две основные цели:

* сокращение общего объема испытаний при соблюдении требований к достоверности и точности их результатов;
* повышение информативности каждого из экспериментов в отдельности.

Поиск плана эксперимента производится в так называемом факторном пространстве.

***Факторное пространство*** – это множество внешних и внутренних параметров модели, значения которых исследователь может контролировать в ходе подготовки и проведения модельного эксперимента.

Поскольку факторы могут носить как количественный, так и качественный характер (например, отражать некоторую стратегию управления), значения факторов обычно называют уровнями. Если при проведении эксперимента исследователь может изменять уровни факторов, эксперимент называется активным, в противном случае – пассивным.

Введем еще несколько терминов, используемых в теории планирования эксперимента.

Каждый из факторов имеет верхний и нижний уровни, расположенные симметрично относительно некоторого нулевого уровня. Точка в факторном пространстве, соответствующая нулевым уровням всех факторов, называется центром плана.

***Интервалом варьирования фактора*** называется некоторое число J, прибавление которого к нулевому уровню дает верхний уровень, а вычитание – нижний.

Как правило, план эксперимента строится относительно одного (основного) выходного скалярного параметра Y, который называется наблюдаемой переменной. Если моделирование используется как инструмент принятия решения, то в роли наблюдаемой переменной выступает показатель эффективности.

При этом предполагается, что значение наблюдаемой переменной, полученное в ходе эксперимента, складывается из двух составляющих:

y = f(x) + e(x),

где f(x) – функция отклика (неслучайная функция факторов);

e(x) – ошибка эксперимента (случайная величина) ;

x – точка в факторном пространстве (определенное сочетание уровней факторов);

Очевидно, что y является случайной переменной, так как зависит от случайной величины e(x).

Дисперсия Dy наблюдаемой переменной, которая характеризует точность измерений, равна дисперсии ошибки опыта: Dy = De.

Dy называют дисперсией воспроизводимости эксперимента. Она характеризует качество эксперимента. Эксперимент называется идеальным при Dy = 0.

Существует два основных варианта постановки задачи планирования имитационного эксперимента:

Из всех допустимых выбрать такой план, который позволили бы получить задачи планирования имитационного эксперимента:

1. Из всех допустимых выбрать такой план, который позволил бы получить наиболее достоверное значение функции отклика f(x) при фиксированном числе опытов.
2. Выбрать такой допустимый план, при котором статистическая оценка функции отклика может быть получена с заданной точностью при минимальном объеме испытаний.

Решение задачи планирования в первой постановке называется стратегическим планирование эксперимента, во второй – тактическим планирование.

#### 7.1.Стратегическое планирование имитационного эксперимента.

Итак, цель методов стратегического планирования имитационных экспериментов получение максимального объема информации об исследуемой системе в каждом эксперименте (наблюдении). Другими словами, стратегическое планирование позволяет ответить на вопрос, при каком сочетании уровней внешних и внутренних факторов может быть получена наиболее полная и достоверная информация о поведении системы.

При стратегическом планировании эксперимента должны быть решены две основные задачи:

* Идентификация факторов;
* Выбор уровней факторов.

Под идентификацией факторов понимается их ранжирование по степени влияния

на значение наблюдаемой переменной (показателя эффективности).

По итогам идентификации целесообразно разделить все факторы на две группы – первичные и вторичные. Первичные – это те факторы, в исследовании влияния которых экспериментатор заинтересован непосредственно. Вторичные - факторы, которые не являются предметом исследования, но влиянием которых нельзя пренебречь.

Выбор уровней факторов производится с учетом двух противоречивых требований:

* Уровни фактора должны перекрывать (заполнять) весь возможный диапазон его изменения;
* Общее количество уровней по всем факторам не должно приводить к чрезмерному объему моделирования.

Отыскание компромиссного решения, удовлетворяющего этим требованиям, и является задачей стратегического планирования эксперимента.

Эксперимент, в котором реализуются все возможные сочетания уровней факторов, называется *полным факторным экспериментом* (ПФЭ).

Общее число различных комбинаций уровней в ПФЭ для k факторов можно вычислить так: *N = l1\* l2\*…\*lk,*

где *li* – число уровней i-го фактора.

Если число уровней для всех факторов одинаково, то N = Lk (L – число уровней).

Недостаток ПФЭ – большие временные затраты на подготовку и проведение.

Например, если в модели отражены 4 фактора ,влияющие на значение выбранного показателя эффективности, каждый из которых имеет 3 возможных уровня (значения), то план проведения ПФЭ будет включать 81 эксперимент (N=34). Если при этом каждый из них длится хотя бы одну минуту (с учетом времени на изменение значений факторов), то на однократную реализацию ПФЭ потребуется более часа.

Поэтому использование ПФЭ целесообразно только в том случае, если в ходе имитационного эксперимента исследуется взаимное влияние всех факторов, фигурирующих в модели.

Если такие взаимодействия считают отсутствующими или их эффектом пренебрегают, проводят частичный факторный эксперимент (ЧФЭ).

Известны и применяются на практике различные варианты построения планов ЧФЭ. Мы рассмотрим только некоторые из них.

1. *Рандомизированный план* – предполагает выбор сочетания уровней для каждого прогона случайным образом.
2. *Латинский план («латинский квадрат»)* - используется в том случае, когда проводится эксперимент с одним первичным фактором и несколькими вторичными. Суть такого планирования состоит в следующем. Если первичный фактор А имеет *l* уровней, то для каждого вторичного фактора также выбирается *l* уровней. Выбор комбинации уровней факторов выполняется на основе специальной процедуры, которую мы рассмотрим на примере.

Пусть в эксперименте используется первичный фактор А и два вторичных фактора – В и С; число уровней факторов *l* равно 4.

Соответствующий план можно представить в виде квадратной матрицы размером *l\*l(4\*4)*относительно уровней фактора А. При этом матрица стоится таким образом, чтобы в каждой строке и в каждом столбце данный уровень фактора А встречался только один раз (табл. 2.2):

Таблица 2.3. Пример латинского плана

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Значение  фактора В | Значение фактора С | | | |
| С1 | С2 | С3 | С4 |
| B1 | A1 | A2 | A3 | A4 |
| B2 | A2 | A3 | A4 | A1 |
| B3 | A3 | A2 | A1 | A4 |
| B4 | A4 | A1 | A2 | A3 |

В результате имеем план, требующий 4\*4 = 16 прогонов, в отличие от ПФЭ, для которого нужно 43 = 64 прогона.

1. *Эксперимент с изменением факторов по одному*.

Суть его состоит в том, что один из факторов «пробегает» все *l* уровней, а остальные *n – 1* факторов поддерживаются постоянными. Такой план обеспечивает исследование эффектов каждого фактора в отдельности. Он требует всего *N = l1+ l2+…+ln* прогонов (*li* – число уровней i – го фактора).

Для рассмотренного выше примера (3 фактора, имеющие по 4 уровня) *N = 4+4+4=12.*

Ещё раз подчеркнем, что такой план применим (как и любой ЧФЭ) только при отсутствии взаимодействия между факторами.

1. *Дробный факторный эксперимент*.

Каждый фактор имеет два уровня – нижний и верхний, поэтому общее число вариантов эксперимента N=2k, k – число факторов. Матрицы планов для k = 2 и k=3 приведены ниже.

Планы, построенные по такому принципу, обладают определенными свойствами (симметричности, нормированности, ортогональности и ротабельности), обеспечивающими повышение качества проводимых экспериментов.

Таблица 2.4 Матрица плана дробного факторного эксперимента для k=2

|  |  |  |
| --- | --- | --- |
| Номер  эксперимента | Значение факторов | |
| х1 | х2 |
| 1 | 0 | 0 |
| 2 | 0 | 1 |
| 3 | 1 | 0 |
| 4 | 1 | 1 |

Таблица 2.5 Матрица плана дробного факторного эксперимента для k=3

|  |  |  |  |
| --- | --- | --- | --- |
| Номер  эксперимента | Значение факторов | | |
| х1 | х2 | х3 |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 1 |
| 3 | 0 | 1 | 0 |
| 4 | 0 | 1 | 1 |
| 5 | 1 | 0 | 0 |
| 6 | 1 | 0 | 1 |
| 7 | 1 | 1 | 0 |
| 8 | 1 | 1 | 1 |

#### 7.2.Тактическое планирование эксперимента.

Совокупность методов установления необходимого объема испытаний относят к тактическому планированию экспериментов.

Поскольку точность оценок наблюдаемой переменной характеризуется её дисперсией, то основу тактического планирования эксперимента составляют так называемые методы понижения дисперсии.

Поскольку имитационное моделирование представляет собой статистический эксперимент, то при его проведении необходимо не только получить достоверный результат, но и обеспечить его «измерение» с заданной точностью. Т.е необходимо выбрать такой объем испытаний, при котором доверительный интервал при заданном значении доверительной вероятности не превышал требуемый.

В общем случае объем испытаний (величина выборки), необходимый для получения оценок наблюдаемой переменной с заданной точностью, зависит от следующих факторов:

* Вида распределения наблюдаемой переменной y (напомним, что при статистическом эксперименте она является случайной величиной);
* Коррелированности между собой элементов выборки;
* Наличия и длительности переходного режима функционирования моделируемой системы.

Если исследователь не обладает перечисленной информацией, то у него имеется единственный способ повышения точности оценок истинного значения наблюдаемой переменной – многократное повторение прогонов модели для каждого сочетания уровней факторов, выбранного на этапе стратегического планирования эксперимента. Такой подход получил название «формирование простой случайной выборки» (сокращенно - ПСВ). Другими словами, при использовании ПСВ каждый «пункт» стратегического плана просто выполняется повторно определенное число раз, и затем полученные результаты усредняются (вычисляются математическое ожидание и дисперсия наблюдаемой переменной). При таком подходе общее число прогонов модели, необходимое для достижения цели моделирования, равно произведению Nc\*x\*Nt (Nc  - число сочетаний уровней факторов по стратегическому плану; Nt – число прогонов модели для каждого сочетания, вычисленное при тактическом планировании).

Поэтому даже при использовании ПСВ до начала испытаний необходимо определить тот минимальный объем выборки, который обеспечит требуемую точность результатов.

Рассмотрим несколько основных вариантов вычисления необходимого объема испытаний(величину Nt).

1.Если случайные значения наблюдаемой переменной не коррелированны и их распределение не изменяется от прогона к прогону, то выборочное среднее можно считать нормально распределенным.

В этом случае число прогонов Nt, необходимое для того, чтобы истинное среднее y лежало в интервале y ± b c вероятностью (1 - α), определяется следующим образом:

Nt = (Z2\*Dy)/b2

где Z – значение нормированного нормального распределения, которое определяется по справочной таблице при заданном уровне значимости a/2;

Dy – дисперсия;

b – доверительный интервал.

Если требуемое значение дисперсии Dy до начала эксперимента неизвестно, целесообразно выполнить пробную серию из L прогонов и вычислить на её основе выборочную дисперсию D.

2. Если наблюдаемая переменная – вектор, то оценку необходимого числа прогонов выполняют отдельно для каждой компоненты вектора. Наибольшее и полученных значений М принимают в качестве числа прогонов Nt.

Основной недостаток методов планирования, основанных на использовании простой случайной выборки – медленная сходимость выборочных средних к истинным средним с ростом объема выборки Nt (прапорционально значению Nt1/2). Это приводит к необходимости использования методов уменьшения ошибок, не требующих увеличения Nt. Такие методы называются методами понижения дисперсии и делятся на три группы:

* *Активные*(предусматривают формирование выборки специальным образом);
* *Пассивные*(применяются после того, как выборка уже сформирована);
* *Косвенные*( в которых для получения оценок наблюдаемой переменной используются значения некоторых величин).

Активных методов понижения дисперсии известно достаточно много. Выбор конкретного метода определяется, как правило, спецификой модели и целями эксперимента. Рассмотрим те из них, которые направлены на снижение влияния переходного периода.. Выбор объясняется тем, что наличие и длительность переходного режима оказывает существенное влияние на качество результатов моделирования (в смысле точности). Вместе с тем, большинство ИМ используется для изучения функционирования системы в установившемся режиме.

Существует три основных метода уменьшения ошибок, обусловленных наличием переходного периода:

1. Значительное увеличение длительности прогона.
2. Исключение и рассмотрения переходного периода
3. Инициализация модели при некоторых специально выбранных начальных условиях.

На практике снижения влияния переходного периода обычно добиваются одним из следующих способов:

* + Методом повторений
  + Методом подинтервалов
  + Методом циклов

Метод повторения

При использовании этого метода каждое наблюдение получается при помощи отдельного прогона модели, причем все прогоны начинаются при одних и тех же начальных условиях, но используются различные последовательности случайных чисел

Преимуществом метода является статистическая независимость получаемых наблюдений. Недостаток состоит в том, что наблюдения могут оказаться сильно смещенными под влиянием начальных условий.

Метод подинтервалов

Данный метод основан на разбиении каждого прогона модели на равные промежутки времени. Начало каждого интервала совпадает с началом очередного этапа наблюдений.

Достоинство метода состоит в том, что влияние переходных условий со временем уменьшается и наблюдение точнее отражает поведении системы в стационарном режиме. Недостаток в том, что значения наблюдаемых переменных, полученных в начале очередного интервала, зависят от конечных условий предыдущего интервала (т.е. между интервалами существует автокорреляция).

Метод циклов

При использовании метода циклов влияние автокорреляции уменьшается за счет выбора интервалов таким образом, чтобы в их начальных точках условия были одинаковыми. Например, в качестве таких условий можно рассматривать длину очереди заявок на обслуживание. В этом случае удобно выбрать начало очередного интервала совпадающим с моментом, когда длина очереди становится равной нулю. Недостатком метода является меньшее по сравнению с методом подинтервалов число получаемых наблюдений.

**Пассивные методы** влияют на подготовку и проведение эксперимента, но реализуются на этапе обработки и анализа результатов моделирования. Их довольно много; рассмотрим наиболее простой и распространенный – метод стратифицированной выборки.

Суть метода состоит в следующем.

Выборка разделяется на части, называемые слоями (стратами). При этом необходимо, чтобы значения элементов выборки как можно меньше различались внутри одного слоя и как можно больше – между различными слоями. Внутри каждого слоя производят случайный отбор элементов и вычисляют среднее значение слоя *yi*. Полученные оценки используют для вычисления математического ожидания по выборке в целом:

y = (∑ Ni yi) / N

где N, Ni – объем всей выборки и i-го слоя соответственно. Если считать, что оценки yi независимы, то дисперсия по выборке и в целом равна:

Dy = (∑ Ni Di) / N

где Dyi - дисперсия для i-го слоя.

При удачном выборе слоев величины Di будут малы, а значит, и выборочная дисперсия Dy будет предпочтительнее, чем для оценки, полученной методами простой случайной выборки.

Косвенные методы понижения дисперсии основаны на том, что зачастую некоторые из выходных характеристик можно получить (вычислить) легче, чем другие. Их использование предполагает не только весьма глубокое знание сущности процессов, протекающих в системе, но и наличие формального описания взаимной зависимости параметров модели.

### Тема 8. ОБРАБОТКА И АНАЛИЗ РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ

Решения, принимаемые исследователем по результатам имитационного моделировании, могут быть конструктивными только при выполнении двух основных условий:

* полученные результаты обладают требуемой точностью и достоверностью;
* исследователь способен правильно интерпретировать полученные результаты.

Возможность выполнения первого условия закладывается, в основном, еще на этапе разработки модели и частично – на этапе планирования эксперимента. Достоверность результатов моделирования предполагает, что модель, с помощью которой они получены, не только является «правильной», но и отвечает и некоторым дополнительным требованиям, предъявляемым к имитационным моделям. Эти требования и методы оценки соответствия их созданной модели рассматриваются ниже.

Способность исследователя правильно интерпретировать полученные результаты и принимать на их основе важные решения существенно зависит от степени соответствия формы представления результатов целям моделирования.

Если разработчик модели уверен, что полученные результаты будут использоваться в соответствии с одной, четко определенной целью, форма их представления может быть определена заранее. В этом случае преобразование экспериментальных данных к требуемому виду может производиться либо в ходе эксперимента, либо сразу после его завершения.

Если же заранее конкретизировать цель моделирования сложно или целей несколько, данные должны накапливаться в базе данных и затем уже выдаваться в требуемой форме по запросу пользователя. Как правило, по такому принципу строятся системы автоматизации моделирования.

При правильной организации обработки экспериментальных данных могут быть получены дополнительные сведения о моделируемой системе.

#### 8.1 Оценка качества имитационной модели

Оценка качества модели является завершающим этапом ее разработки и преследует две цели:

* проверить соответствие модели ее предназначению (целям исследования);
* оценить достоверность и статистические характеристики результатов, получаемых при проведении модельных экспериментов.

При аналитическом моделировании достоверность результатов определяется двумя основными факторами:

* корректным выбором математического аппарата, используемого для описания исследуемой системы;
* методической ошибкой, присущей данному математическому методу.

При имитационном моделировании на достоверность результатов влияет целый ряд дополнительных факторов, основными из которых являются:

* моделирование случайных факторов, основанное на использовании датчиков СЧ, которые могут вносить «искажения» в поведение модели;
* наличие нестационарного режима работы модели;
* использование нескольких разнотипных математических методов в рамках одной модели;
* зависимость результатов моделирования от плана экспериментов;
* необходимость синхронизации работы отдельных компонентов модели;
* наличие модели рабочей нагрузки, качество которой зависит, в свою очередь, от тех же факторов.

Пригодность имитационной модели для решения задач исследования характеризуется тем, в какой степени она обладает так называемыми целевыми свойствами. Основными из них являются:

* адекватность;
* устойчивость;
* чувствительность.

Ниже рассмотрены некоторые способы проведения оценки модели по каждому из них.

##### 8.1.1. Оценка адекватности модели.

В обще случае под адекватностью понимают степень соответствия модели тому реальному явлению или объекту, для описания которого она строится.

Вместе с тем, создаваемая модель ориентирована, как правило, на исследование определенного подмножества свойств этого объекта. Поэтому можно считать, что адекватность модели определяется степенью ее соответствия не столько реальному объекту, сколько целям исследования. В наибольшей степени это утверждение справедливо относительно моделей проектируемых систем (т.е. в ситуациях, когда реальная система вообще не существует).

Тем не менее, во многих случаях полезно иметь формальное подтверждение (или обоснование) адекватности разработанной модели. Один из наиболее распространенных способов такого обоснования - использование методов математической статистики. Суть этих методов заключается в проверке выдвинутой гипотезы (в данном случае - об адекватности модели) на основе некоторых статистических критериев.

*Замечание: при проверке гипотез методами математической статистики необ­ходимо иметь в виду, что статистические критерии не могут доказать ни одной ги­потезы: они могут лишь указать на отсутствие опровержения.*

Процедура оценки основана на сравнении измерений на реальной системе и результатов экспериментов на модели и может проводиться различными способами. Наиболее распространенные из них:

* по средним значениям откликов модели и системы;
* по дисперсиям отклонений откликов модели от среднего значения откликов системы;
* по максимальному значению относительных отклонений откликов  
  модели от откликов системы.

Названные способы оценки достаточно близки по сути, поэтому ограничимся рассмотрением первого из них.

При этом способе проверяется гипотеза о близости среднего значения наблюдаемой переменной У среднему значению отклика реальной системы Y\*.

В результате No опытов на реальной системе получают выборку Y\*. Выполнив NM экспериментов на модели, также получают множество значений наблюдаемой переменной Y.

Затем вычисляются оценки математического ожидания и дисперсии откликов модели и системы, после чего выдвигается гипотеза о близости средних значений величин У\* и У (в статистическом смысле). Основой для проверки гипотезы является t-статистика (распределение Стьюдента). Ее значение, вычисленное по результатам испытаний, сравнивается с критическим значением £кр, взятым из справочной таблицы. Если выполняется неравенство t < £кр, то гипотеза принимается.

*Необходимо еще раз подчеркнуть, что статистические методы применимы только в том случае, если оценивается адекватность модели существующей системе. На проектируемой системе провести измерения, естественно, не представляется возможным. Единственный способ преодолеть это препятствие заключается в том, чтобы принять в качестве эталонного объекта концептуальную модель проектируемой системы. Тогда оценка адекватности программно реализованной модели заключается в проверке того, насколько корректно она отражает концептуальную модель. Данная проблема сходна с проверкой корректности любой компьютерной программы, и ее можно решать соответствующими методами, например с помощью тестирования.*

##### 8.1.2. Оценка устойчивости модели.

При оценке адекватности модели как существующей, так и проектируемой системе реально может быть использовано лишь ограниченное подмножество всех возможных значений входных параметров (рабочей нагрузки и внешней среды). В связи с этим для обоснования достоверности получаемых результатов моделирования большое значение имеет проверка устойчивости модели. В теории моделирования это понятие трактуется следующим образом.

*Устойчивость модели* — это ее способность сохранять адекватность при исследовании эффективности системы на всем возможном диапазоне рабочей нагрузки, а также при внесении изменений в конфигурацию системы.

Универсальной процедуры проверки устойчивости модели не существует. Разработчик вынужден прибегать к частичным тестам и здравому смыслу. Часто бывает полезна апостериорная проверка. Она состоит в сравнении результатов моделирования и результатов измерений на системе после внесения в нее изменений. Если результаты моделирования приемлемы, уверенность в устойчивости модели возрастает.

В общем случае можно утверждать, что чем ближе структура модели структуре системы и чем выше степень детализации, тем устойчивее модель.

Устойчивость результатов моделирования может быть также оценена методами математической статистики. Здесь уместно вспомнить основную задачу математической статистики. В данном случае устойчивость результатов моделирования можно рассматривать как признак, подлежащий оценке. Для проверки гипотезы об устойчивости результатов может быть использован критерий Уилкоксона.

Критерий Уилкоксона служит для проверки того, относятся ли две выборки к одной и той же генеральной совокупности (т. е. обладают ли они одним и тем же статистическим признаком). Например, в двух партиях некоторой продукции измеряется определенный признак, и требуется проверить гипотезу о том, что этот признак имеет в обеих партиях одинаковое распределение; другими словами, необходимо убедиться, что технологический процесс от партии к партии изменяется несущественно.

При статистической оценке устойчивости модели соответствующая гипотеза может быть сформулирована следующим образом: при изменении входной (рабочей) нагрузки или структуры ИМ закон распределения результатов моделирования остается неизменным.

Проверку указанной гипотезы Н проводят при следующих исходных данных:

* есть две выборки X = (х1 ..., хn) и Y = (у1 ..., уm), полученные для различных значений рабочей нагрузки; относительно законов распределения X и Y никаких предположений не делается.
* значения обеих выборок упорядочиваются вместе по возрастанию. Затем анализируется взаимное расположение xi и yj В случае уj < xi говорят, что пара значений (xi yj) образует инверсию.

(*Например, пусть для n = m = 3. После упорядочивания получилась такая последовательность значений: у1 х1 у3, х2, у2, х3. Тогда имеем инверсии: (х1, у1), (х2, у1), (х2, у3), (х3, у1), (х3, у2), (х3, у3)*).

* подсчитывают полное число инверсий U. Если гипотеза верна, то U должно сильно отклоняться от своего математического ожидания М:

*М=(п\* т)/2*

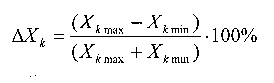
От гипотезы отказываются, если | U - M | > Ukp (Ukp определяют по таблице для заданного уровня значимости).

##### 8.1.3. Оценка чувствительности ИМ.

Очевидно, что устойчивость является положительным свойством модели. Однако если изменение входных воздействий или параметров модели (в некотором заданном диапазоне) не отражается на значениях выходных параметров, то польза от такой модели невелика (ее можно назвать «бесчувственной»). В связи с этим возникает задача оценивания чувствительности модели к изменению параметров рабочей нагрузки и внутренних параметров самой системы.

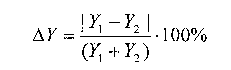
Такую оценку проводят по каждому параметру Хk в отдельности. Основана она на том, что обычно диапазон возможных изменений параметра известен. Одна из наиболее простых и распространенных процедур оценивания состоит в следующем.

1) вычисляется величина относительного среднего приращения параметра Хk:

****

2) проводится пара модельных экспериментов при значениях Хk =Хkmax и Хk = Xkmin и средних фиксированных значениях остальных параметров. Определяются значения отклика модели Y1=f(Xkmax) и Y2=(Xkmin);

3)вычисляется ее относительное приращение наблюдаемой переменной У:



В результате для k-го параметра модели имеют пару значений (∆Хк, ∆Ук), характеризующую чувствительность модели по этому параметру.

Аналогично формируются пары для остальных параметров модели, которые образуют множество {∆Хк,∆Ук}.

Данные, полученные при оценке чувствительности модели, могут быть использованы, в частности, при планировании экспериментов: большее внимание должно уделяться тем параметрам, по которым модель является более чувствительной.

##### 8.1.4. Калибровка модели.

Если в результате проведенной оценки качества модели оказалось, что ее целевые свойства не удовлетворяют разработчика, необходимо выполнить ее калибровку, т. е. коррекцию с целью приведения в соответствие предъявляемым требованиям.

Как правило, процесс калибровки носит итеративный характер и состоит из трех основных этапов:

* глобальные изменения модели (например, введение новых процессов, изменение типов событий и т. д.);
* локальные изменения (в частности, изменение некоторых законов  
  распределения моделируемых случайных величин);
* изменение специальных параметров, называемых калибровочными.

Целесообразно объединить оценку целевых свойств ИМ и ее калибровку в единый процесс. Именно такая стратегия принята в статистическом методе калибровки, описанном ниже.

Процедура калибровки состоит из трех шагов, каждый из которых является итеративным (рис. 2.13).

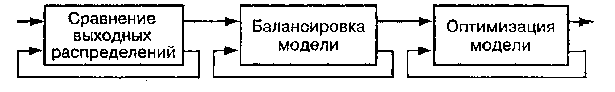


Рис 2.13. Схема процесса калибровки ИМ

**Шаг** 1. Сравнение выходных распределений.

Цель - оценка адекватности ИМ. Критерии сравнения могут быть различны. В частности, может использоваться величина разности между средними значениями откликов модели и системы.

Устранение различий на этом шаге основано на внесении глобальных изменений.

**Шаг** 2. Балансировка модели.

Основная задача - оценка устойчивости и чувствительности модели. По его результатам, как правило, производятся локальные изменения (но возможны и глобальные).

Шаг 3. **Оптимизация модели.**

Цель этого этапа — обеспечение требуемой точности результатов. Здесь возможны три основных направления работ:

* дополнительная проверка качества датчиков СЧ;
* снижение влияния переходного режима;
* применение специальных методов понижения дисперсии.

#### 8.2. Подбор параметров распределений

В некоторых случаях имитационная модель сложной системы может быть реализована в виде набора отдельных моделей ее подсистем. При проведении экспериментов с такой моделью в целях сокращения затрат времени бывает необходимо заменять моделирование работы одной из подсистем некоторым числовым параметром (вспомните принцип параметризации), либо случайной величиной, распределенной по заданному закону. Чтобы такая замена была выполнена корректно, исследователь должен располагать описанием зависимости данного числового параметра от времени и других факторов, фигурирующих в модели.

При имитационном моделировании подбор законов распределений выполняется на основе статистических данных, полученных в ходе эксперимента.

В основе процедуры отыскания закона распределения некоторой величины по экспериментальным данным лежит проверка статистических гипотез.

Статистическая гипотеза — это утверждение относительно значений одного или боле параметров распределения некоторой величины или о самой форме распределения.

Обычно выбирают две исходные гипотезы: основную—Н0 и альтернативную ей – H1

Статистическая проверка гипотезы - это процедура выяснения, следует ли принял основную гипотезу Н0 или отвергнуть ее.

Если в результате проверки гипотеза Н0 ошибочно отвергается, то имеет место ошибка 1-го рода (характеризующаяся более тяжелыми последствиями); если гипотеза Н0 принимается при истинности H1 — это ошибка второго рода.

Вероятности ошибок I и II рода (α и β) зависят от критерия, на основание которого будет выбираться одна из гипотез. Очевидно, что вероятности этих двух ошибок взаимосвязаны, то есть чем больше значение α, тем меньше βи наоборот.

Обычное решение этой дилеммы состоит в том, что выбирают некоторое фиксированное значение α(как правило 0.05, 0.01, 0.001) и надеются, что β будет также мало. Фиксированное значение а называется **уровнем значимости.**

Для выбранного значения α определяется так называемая критическая область В, удовлетворяющая условию

****

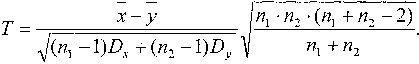
где Z— контрольная величина (критерий), представляющая собой некоторую функцию от выборки (результатов эксперимента).

Проверка гипотезы состоит в следующем. Производится выборка (проводится эксперимент), на основании чего вычисляется z — частное значение критерия Z. Если , то от гипотезы Н0 отказываются. Если z не лежит в В, то говорят, что полученные наблюдения **не противоречат принятой гипотезе.**

Для наиболее часто используемых статистических гипотез разработаны критерии, позволяющие проводить их проверку с наибольшей достоверностью. Рассмотрим основные из них.

***t-критерий***служит для проверки гипотезы о равенстве средних значений двух нормально распределенных СВ (X и Y) в предположении, что дисперсии их равны (хотя и неизвестны). Сравниваемые выборки могут иметь разный объем.

В качестве критерия используют величину

****

Величина Т подчиняется t-распределению Стьюдента.

Критическое значение для t-критерия определяется по таблице для выбранного значения α и числа степеней свободы k = n1+n2-2.

Если вычисленное значение Т превышает tкр, найденное по таблице, то ги­потезу Н0 отвергают.

По отношению к предположению о «нормальной распределенности» величин x и у t-критерий не очень чувствителен. Его можно применять, если распределения СВ не имеют нескольких вершин и не слишком асимметричны.

***F-критерий***служит для проверки гипотезы о равенстве дисперсий Dx и Dy при условии, что x и у распределены нормально.

Гипотезы такого рода имеют большое значение в технике, так как дисперсия есть мера таких характеристик, как погрешности измерительных приборов, точность технологических процессов, точность наведения при стрельбе и так далее.

В качестве контрольной величины используется отношение дисперсий F= Dx / Dy (или Dy / Dx - большая дисперсия должна быть в числителе).

Величина F подчиняется F-распределению (Фишера) с (m1, m2) степенями свободы (m1=n1-1, m2=n2 - l). Проверка гипотезы состоит в следующем.

Для величины γ = α/2 и величин m1 m2 по таблице F-распределения выбирают значения Fγ,m1,m2. Если статистика F\*, вычисленная по выборке, больше этого критического значения, гипотеза должна быть отклонена с вероятностью ошибки α.

Критерии согласия — это критерии, с помощью которых проверяют, удовлетворяет ли рассматриваемая СВ данному закону распределения.

*Критерий согласия Пирсона* **(формула)**и *Критерий Колмогорова* — *Смирнова* были подробно рассмотрены в курсе ТВиМС.

#### 8.3.Оценка влияния и взаимосвязи факторов

Как правило, количественная оценка степени влияния того или иного фактора на значения наблюдаемой переменной (показателя эффективности) вызывает значительную сложность, особенно при наличии взаимного влияния факторов. Наиболее простой и доступный способ решения этой проблемы состоит в использовании результатов оценки чувствительности модели.

Однако эти результаты сложно представить в форме аналитической зависимости. Такое представление может оказаться весьма полезным для многих практических задач, связанных как с разработкой моделей (речь опять-таки идет о принципе параметризации), так и непосредственно с принятием решений по экспериментальным данным.

Отыскание аналитических зависимостей, связывающих между собой различные параметры, фигурирующие в модели, может быть основано на совместном использовании группы методов математической статистики: дисперсионного, корреляционного и регрессионного анализа. Подробному и строгому описанию соответствующих процедур посвящено огромное количество книг учебного, научного и справочного характера. Поэтому основная цель изложения последующего материала сводится к тому, чтобы показать роль и место указанных методов при проведении анализа данных, полученных в ходе имитационного эксперимента.

##### 8.3.1. Однофакторный дисперсионный анализ

Его суть сводится к определению влияния на результат моделирования одного выбранного фактора.

Пусть, например, исследователя интересует средняя интенсивность отказов компьютера, и в созданной им модели учтены следующие факторы: интенсивность поступления заданий пользователей, интенсивность обращений в оперативную память, временные характеристики решаемых задач и интенсивность обращений к жесткому диску. Если предварительные данные говорят о том, что основной причиной отказов является ненадежная работа жесткого диска, то в качестве анализируемого фактора целесообразно выбрать интенсивность обращений к нему. Задача факторного анализа в данном случае состоит в том, чтобы оценить влияние указанного фактора на среднее число отказов.

Формально постановка задачи однофакторного дисперсионного анализа состоит в следующем. Пусть интересующий нас фактор х имеет 1 уровней. Для каждого из них получена выборка значений наблюдаемой переменной у: уj(1), уj(2), ... yj(l), j=l,...,n, n — объем выборки (число наблюдений).

Необходимо проверить гипотезу Н0 о равенстве средних значений выборок (т.е. о независимости значений у от значений исследуемого фактора x). Уравнение однофакторного дисперсионного анализа имеет вид:

*yij = m+ai+eij,*

где yij - j-e значение у в i-й серии опытов,

m - генеральное среднее случайной величины у,

а. - неизвестный параметр, отражающий влияние фактора х («эффект» i-гo значения фактора х),

eij - ошибка измерения у.

Для проверки гипотезы Н0 используют F-критерий и переходят от проверки значимости различий средних к проверке значимости различий двух дисперсий:

* генеральной (обусловленной погрешностями измерений) - D0;
* факторной (обусловленной изменением фактора х) - Dx.

Значение F-критерия вычисляется как отношение Dx / Do или Do / Dx (в числителе должна стоять большая из дисперсий); затем но по таблице F-распределений находят его критическое значение fкр для заданного уровня значимости и числа степеней свободы.

Если F > Ркр, то гипотезу Н0 отвергают, т. е. различия являются значимыми (фактор x влияет на значения у).

##### 8.3.2. Многофакторный дисперсионный анализ

Многофакторный дисперсионный анализ (МДА) позволяет оценивать влияние на наблюдаемую переменную уже не одного, а произвольного числа факторов. Точнее, МДА позволяет выбрать ив группы факторов, участвующих в эксперименте, те, которые действительно влияют на его результат.

Методику проведения многофакторного дисперсионного анализа рассмотрим применительно к частичному факторному эксперименту, проводимому в соответствии с латинским планом.

Пусть в эксперименте рассматриваются один первичный фактор и два вторичных, каждый из которых имеет n уровней (т. е. объем испытаний N = n2).

Обозначим через yijk результат эксперимента при условии, что фактор ***а***находился на уровне i, фактор ***b***- на уровне j, фактор ***с***- на уровне к. Множество значений, которые может принимать упорядоченная тройка (i, j, к), обозначим через L.

В этом случае уравнение дисперсионного анализа выглядит следующим образом:

*yijk =m+ai+bj+gk+eijk,*

где m - генеральное среднее случайной величины у,

ai, bj, gk - неизвестные параметры («эффекты» соответствующих факторов).

Решение задачи дисперсионного анализа заключается в проверке гипотез о независимости результатов измерений от факторов ***a, b, c****:*

*Ha: ai=0, i=1, …, n;*

*Hb: bj=0, j=1, …, n;*

*Hc: ck=0, k=1,…, n;*

Для этого по методу наименьших квадратов (МНК) находят оценки параметров m, ai, bj, gk, минимизируя по указанным переменным (поочередно) функцию



Затем по каждому фактору вычисляется f-статистика. Величина F есть мера потерь при принятии гипотезы H0. Чем больше F, тем хуже модель, отвергающая влияние соответствующего фактора. Таким образом, если вычисленное значение F больше Fкр, найденного по таблице для некоторого уровня значимости, то гипотеза отвергается.

Необходимо отметить, что дисперсионный анализ может использоваться для оценки влияния факторов, имеющих как количественный характер, так и качественный, поскольку в уравнении дисперсионного анализа фигурируют не сами факторы, а только их «эффекты».

В том случае, если все факторы носят количественный характер, взаимосвязь между ними и наблюдаемой переменной может быть описана с помощью уравнения регрессии.

##### 8.3.3. Корреляционный и регрессионный анализ.

Это два близких метода, которые обычно используются совместно для исследования взаимосвязи между двумя или более непрерывными переменными.

Методы корреляционного анализа позволяют делать статистические выводы о степени зависимости между переменными.

Величина линейной зависимости между двумя переменными измеряется посредством простого коэффициента корреляции, величина зависимости от нескольких - посредством множественного коэффициента корреляции.

В корреляционном анализе используется также понятие частного коэффициента корреляции, который измеряет линейную взаимосвязь между двумя переменными без учета влияния других переменных.

Если корреляционный анализ позволил установить наличие линейной зависимости наблюдаемой переменной от одной или более независимых, то форма зависимости может быть уточнена методами регрессионного анализа.

Для этого строится так называемое уравнение регрессии, которое связывает зависимую переменную с независимыми и содержит неизвестные параметры. Если уравнение линейно относительно параметров (но необязательно линейно относительно независимых переменных), то говорят о линейной регрессии, в противном случае регрессия нелинейна.

Рассмотрим простой корреляционный анализ, т. е. метод определения взаимосвязи между двумя переменными.

Обозначим их x и у. Независимо от способа получения выборки имеются два предварительных шага для определения существования и степени линейной зависимости между x и у. Первый шаг заключается в графическом отображении точек (xi уi) на плоскости (х, у) — т. е. в построении диаграммы рассеяния. Анализируя диаграмму рассеяния, можно решить, допустимо ли предположение о линейной зависимости между x и у (см.рис.8.1).

Если rxy не равен нулю, то на втором шаге вычисляется его точное значение.

Чем больше по абсолютному значению rxy, тем сильнее линейная зависимость между переменными. При r =1 имеет место функциональная линейная зависимость между x и у вида у=ах+b, причем если г положительно, то говорят о положительной корреляции, т.е. большие значения одной величины соответствуют большим значениям другой; при r = -1 имеет место отрицательная корреляция; при 0 < r < 1 вероятна либо линейная корреляция с рассеянием (рис.8.1, в), либо нелинейная корреляция (рис.8.1, г).

При анализе результатов ИМ необходимо иметь в виду, что если даже удалось установить тесную зависимость между двумя переменными, это еще не является прямым доказательством их причинно-следственной связи. Возможно, имеет место стохастическая зависимость, обусловленная, например, коррелированностью последовательностей псевдослучайных чисел, используемых в имитационной модели.

Поэтому результаты корреляционного анализа целесообразно уточнить, проведя регрессионный анализ.

Регрессионный анализ позволяет решать две задачи:

1. устанавливать наличие возможной причинной связи между переменными;
2. предсказывать значения зависимой переменной по значениям независимых переменных. Эта возможность особенно важна в тех случаях, когда прямые измерения зависимой переменной затруднены.

Если предполагается линейная зависимость между x и y, то она может быть описана уравнением вида yi=b0+b1x+ei, которое называется простой линейной регрессией y по оси x.

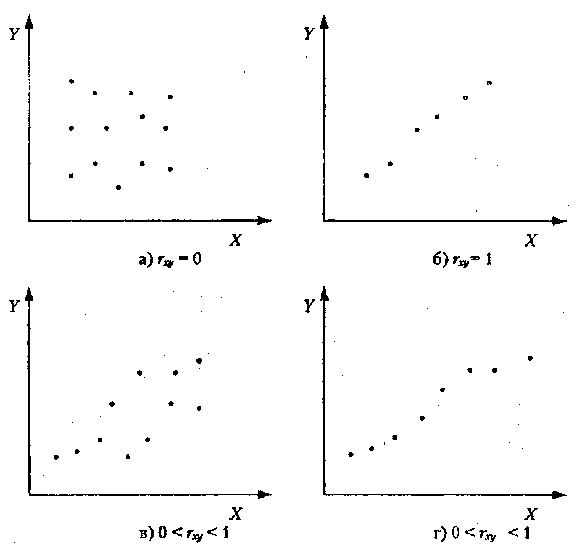


Рис. 8.1 Графическое отображении корреляции переменных

Величины b0 и b1 являются неизвестными параметрами, а еi — случайные ошибки испытаний.

Цель регрессионного анализа — найти наилучшие в статистическом смысле оценки параметров b0 и bi (величину bi обычно называют коэффициентом регрессии).

Разница между наблюдаемым и оцененным значением у при х = xi называется отклонением (или остатком). Величины отклонений могут быть использованы для проверки адекватности полученной модели. Для этого строится график отклонений d = f(y) или d=f(x) (см. рис.8.2), и по его виду делается предварительное заключение о степени адекватности регрессионной модели.

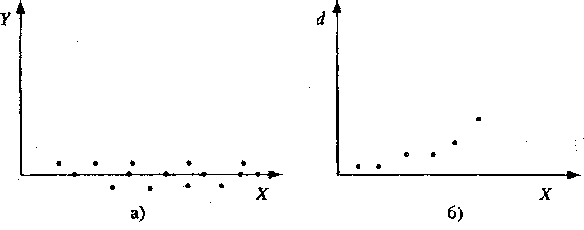


Рис. 2.16 Графическое представление функций отклонений: а – модель адекватна; б – необходимо введение дополнительной независимой переменной.

В случае нескольких независимых переменных имеет место ***множественная линейная регрессия****:*

y=b0+b1x1+b2x2+ ...+bkxk+e.

В этом случае для отыскания оценок bi также используется метод наименьших квадратов.

В случае нелинейной регрессии основой для построения регрессионной модели опять-таки является МНК. Однако в этом случае для отыскания оценок b строится система нелинейных уравнений (относительно bi), а для ее решения используются различные итерационные методы.

# ПРАКТИЧЕСКИЙ РАЗДЕЛ

Рабочей программой дисциплины «Математическое моделирование» предусмотрено выполнение двух ИПР с ИКТ и одной контрольной работы.

Лабораторная работа должна быть представлена в виде исполняемых файлов и текстов программ. Отчет по лабораторной работе должен содержать задание, пример выполнения программы.

Контрольная работа должна быть оформлена в соответствии с общеустановленными нормами и правилами, предъявляемыми к выполнению контрольных работ.

**ИНДИВИДУАЛЬНЫЕ ПРАКТИЧЕСКИЕ РАБОТЫ, ИХ ХАРАКТЕРИСТИКА**

### ИПР 1. ПОСТРОЕНИЕ И ИССЛЕДОВАНИЕ ХАРАКТЕРИСТИК ДАТЧИКОВ БАЗОВЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

***Базовой*** случайной величиной (БСВ) в статистическом моделировании называют непрерывную случайную величину *z*, равномерно распределенную на интервале (0,1). Ее плотность распределения вероятностей (**п.р.в**.) имеет вид:

*f* (*t*) = 1, 0 < *t* < 1,

Математическое ожидание (м.о.) и дисперсия БСВ составляют



соответственно.

БСВ моделируется на ЭВМ с помощью *датчиков* БСВ. Датчик БСВ  это устройство или программа, выдающая по запросу одно или несколько независимых значений *z1* , ..., *zn* БСВ.

Датчики БСВ могут быть трех типов: табличные, физические и программные.

***Программный*** датчик БСВ обычно вычисляет значения *z1*, *z2*,..., по какой-либо рекуррентной формуле типа

*zi = f ( zn),*

при заданном стартовом значении *z0*.

Заданное значение *z0* полностью определяет всю последовательность реализаций *z1*, *z2*,..., поэтому *z* часто называют *псевдослучайной* величиной. Но ее статистические свойства идентичны свойствам "чисто случайной" последовательности, что и обеспечивает успех статистического моделирования.

Имея датчик БСВ *z*, можно промоделировать любые случайные факторы: непрерывные или дискретные случайные величины (как простые, так и многомерные), случайные события, случайные процессы и поля и т.д. Для этого достаточно соответствующим образом преобразовать последовательность *z1*, *z2*, ... . Поэтому БСВ *z* и называют базовой.

Теоретически в качестве базовой можно было бы взять почти любую случайную величину (**с.в**.). Использование **с.в**. *z* с распределением обусловлено технологическими соображениями: простотой и экономичностью датчика, простотой преобразования *z* в другие случайные факторы, относительной простотой тестирования датчика.

#### Метод середины квадрата

Метод середины квадрата предложен для получения псевдослучайных чисел Д. фон Нейманом в 1946 г. Вот один из вариантов этого метода.

* 1. Возьмем произвольное 4-значное число.
  2. Возведем полученное число в квадрат и, если необходимо, добавим к результату слева нули до 8-значного числа.
  3. Возьмем четыре цифры из середины 8-значного в качестве нового случайного 4-значного числа.
  4. Если нужны еще случайные числа, то перейдем к 2.

Например, если взять в качестве начального числа 1994, то из него получается следующая последовательность псевдослучайных чисел: 9760 2576 6357 4114 9249 5440 5936 2360 5696 4444 7491 1150 3225 4006 0480 2304 3084 5110 1121 2566 ...

Сам по себе метод середины квадрата не получил широкого распространения, так как выдает "больше чем нужно малых значений". Но открытый в нем принцип используется во многих, если не во всех, более поздних датчиках БСВ. Этот принцип состоит в вырезании нескольких цифр из результата какой-либо операции над числами.

#### Мультипликативный конгруэнтный метод

Так называемый мультипликативный конгруэнтный датчик БСВ задается двумя параметрами: модулем *m* и множителем *k*. Обычно это достаточно большие целые числа.

При заданных *m*, *k* числа *z1*, *z2*, ..., вычиcляются по рекуррентной формуле:

*Ai* = (*kAi -1*) mod *m*, *i* = 1, 2,...,

*zi = Ai / m*,

где *m * модуль, *k * множитель, *A0*  начальное значение, mod  операция вычисления остатка от деления *kAi -1* на *m*.

Таким образом, *A1* определяется как остаток от деления  *kA0* на *m*; *A2* - как остаток от деления *kA1* на *m* и т.д. Поскольку все числа *Ai*  это остатки от деления на *m*, то 0   *Ai* < *m*. Разделив последнее неравенство на *m*, видим, что 0  *Ai / m*< 1, т. е. 0  *zi* <1.

Из неравенства 0  *Ai* < *m* вытекает также, что датчик (2.5) дает периодическую последовательность *Ai*. Действительно, число всех возможных остатков от 0 до *m* - 1 равно *m* и, рано или поздно, на каком-то шаге *i* обязательно появится значение *Ai*, уже встречавшееся ранее. С этого момента последовательность *Ai* “зациклится".

Длина периода *T* будет не больше *m -* 1. Например, если встретится остаток *Ai*= 0, то далее, согласно (2.5), будет *Ai+ 1* = 0, *Ai+ 2* = 0, ... , т.е. длина периода *T* = 1. Ненулевых же остатков в интервале 0 *Ai* < *m* всего *m -* 1, и, если все они войдут в период, будет *T* = *m -* 1. Это имеет место, например, при *m* = 13, *k* = 7; в этом случае ряд *Ai* выглядит так:

1, 7, 10, 5, 9, 11, 12, 6, 3, 8, 4, 2,            1, 7,... .

\\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

*T* = *m -* 1 = 12

Поскольку в качестве случайной можно использовать лишь подпоследовательность *Ai* внутри одного периода, то параметры датчика выбирают так, чтобы длина периода *T* была максимальной. С учетом ограничения *T m -* 1 модуль *m* берут максимально возможным. Чтобы упростить вычисление остатков по (2.5), для двоичных ЭВМ часто берут *m* = 2*n*. Рекомендуется также брать достаточно большой множитель *k*, причем взаимно простой с *m*.

Существуют подробные рекомендации по выбору параметров *m*, *k* и начального значения *A0* . Заметим, однако, что в настоящее время не известны правила, которые гарантировали бы высокое качество датчика без его специального статистического тестирования.

Датчик называют мультипликативно-конгруэнтным потому, что он использует две основные операции  умножение (англ. multiplication) и вычисление остатка (в теории чисел  получение конгруэнтного числа). Можно было бы поэтому перевести его название и как "множительно-остатковый датчик".

Обратим внимание также и на то, что операция вычисления остатка воплощает здесь упоминавшийся в п. 2.2 неймановский принцип вытаскивания цифр. Это становится очевидным, если записывать числа в системе счисления с основанием *m*. Тогда операция *X* mod *m* означает выбор последней цифры из числа *X*. Для *m* = 2*n* операция *X* mod *m* .

#### Тестирование равномерности

Обозначим равномерное распределение вероятностей на интервале (0,1) через *R*[0,1]. Тогда утверждение, что БСВ *z* имеет распределение *R*[0,1], можно кратко записать в виде *z* ~ *R*[0,1].

С помощью статистических тестов проверяют два свойства датчика, делающих его точной моделью идеальной БСВ,  это *равномерность* распределения чисел *zi*, выдаваемых датчиком на интервале (0,1), и их статистическая *независимость*. При этом числа *zi* рассматривают как реализации некоторой **с.в.**, т.е. как статистическую выборку.

Достаточно простым методом проверки равномерности распределения является частотный тест. Он основан на законе больших чисел и выполняется по следующему алгоритму.

1. Разобьем интервал (0,1) на *K* равных отрезков (например, *K* = 10).

2. Сгенерируем *n* чисел *z1*,..., *zn* с помощью тестируемого датчика БСВ (например, *n* = 100).

3. Подсчитаем, сколько чисел попало в каждый из *k* отрезков, т.е. найдем числа попаданий *n1*,...,*nk*.

4. Рассчитаем относительные частоты попаданий в отрезки: 

5. Построим гистограмму частот на *K* отрезках интервала (0,1).

6. Повторим действия (2)  (5) для большего значения *n* (например, для *n* =10 000).

7. Оценим по полученным гистограммам сходимость каждой частоты к вероятности *p* = 1/*K* того, что БСВ попадет в *i*-й отрезок. Согласно закону больших чисел должно быть



Это значит, что высоты столбиков во второй гистограмме должны в целом быть ближе к уровню 1/*K*, чем в первой.

Тестирование датчика на равномерность можно совместить с оцениванием **м.о**. и дисперсии **с.в.** Оценки  и для **м.о.** и дисперсии рассчитываются соответственно по формулам:

С ростом *n* оценки  и  должны сходиться по вероятности к точным значениям *M*(*z*) = 1/2, *D*(*z*) = 1/12 = 0.08333... .

#### Тестирование независимости

Простейшую проверку статистической независимости реализаций *z1*, *z2*, ..., можно осуществить, оценивая корреляцию между числами *zi* и *zi+ s*, отстоящими друг от друга на шаг *s* >1.

Для вывода формулы, по которой можно рассчитать коэффициент корреляции чисел *zi* и *zi+ s* , рассмотрим две произвольные СВ *x*, *y*. Коэффициент корреляции определяется для них формулой:



С ростом *n* оценка *R*' должна приближаться к нулю, в противном случае датчик БСВ не отвечает требованию независимости.

Конечно, если *R*' сходится к нулю, то это еще не гарантирует наличие независимости, но все же один из тестов оказывается успешно выдержанным. При желании всегда можно продолжить испытания датчика другими методами.

#### ЗАДАНИЕ

Написать программы, реализующие рассмотренные методы построения датчиков случайных величин (разрядность чисел – не менее 8)

Выполнить статистическое исследование датчиков.

Сравнить результаты.

### ИПР 2. ИМИТАЦИЯ НЕПРЕРЫВНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН (МЕТОД ОБРАТНЫХ ФУНКЦИЙ)

Случайные числа с заданным законом распределения вероятностей, как правило, формируются в результате преобразования случайных равномерно распределенных чисел R[0,1]. В настоящее время известно много процедур, позволяющих имитировать непрерывные и дискретные вероятностные распределения. Рассмотрим одину наиболее распространенную процедуру.

Пусть имеется непрерывная случайная величина X, распределенная с постоянной плотностью в интервале (0,1), которая описывается плотностью распределения

****

Требуется путем функционального преобразования *Y=φ(X)* получить случайную величину с заданной функцией распределения *G(y)*. Покажем, что для этого надо подвергнуть равномерно распределенную случайную величину X функциональному преобразованию:

,

где *G-1 –* функция, обратная требуемой функции распределения *G(y).*

Поскольку функция распределения непрерывна и монотонна, то и обратная функция *G-1* также непрерывна и монотонна. В этом случае функция распределения случайной величины Y определяется так:

.

Следовательно, для получения значения *y* непрерывной случайной величины Y нужно выполнить следующее:

1. Получить значение случайной величины *X*, распределенной равномерно на интервале (0, 1).
2. Найти обратную функцию *G-1(x)*по отношению к требуемой функции распределения *G(y)* и вычислить значение случайной величины Y по формуле:



#### ЗАДАНИЕ

Получить у преподавателя вид типового распределения непрерывной случайной величины.

Написать программу реализующую метод формирования непрерывной случайной величины.

Выполнить статистическое исследование (построение гистограммы, точечных, интервальных оценок)

Проверить гипотезы о соответствии закона распределения полученной случайной величины требуемому.

**КОНТРОЛЬНАЯ РАБОТА**

### ПОСТРОЕНИЕ ИМИТАЦИОННЫХ МОДЕЛЕЙ

#### Структура модели СМО (Q-схемы)

Для детального ознакомления с технологией машинной имитации рассмотрим Q-схему достаточно общего вида (рис.КР.1). Q-схема содержит три фазы обслуживания и источник заявок*.*

Первая фаза содержит 2 однотипных канала К11 и К12 и общий входной накопитель заявок Н1. В случае заполнения накопителя Н1 заявки источника получают отказ (дисциплина отказа заявкам на входе фазы I).

Вторая фаза также содержит два однотипных канала К21,К22 и общий входной накопитель Н2. В случае заполнения накопителя Н2 заявки блокируются в первой фазе. Это означает, что если какой -либо канал К21 или К22 в некоторый момент модельного времени завершил обслуживание заявки и в этот момент каналы второй фазы заняты и накопитель заполнен, то обслуженная заявка не покидает систему, что имеет место в случав отказа, а блокируется в канале первой фазы. Заявка сохраняется каналом первой фазы до тех пор, пока в накопителе Н2 не освободится по крайней мере одна позиция.

Третья фаза содержит только один канал К31 и накопитель Н3 емкостью, равной нулю. При занятом канале К31 заявки блокируются во второй фазе.

Для описания имитационной модели Q-схемы введем следующие переменные*:*

tn - текущее значение модельного времени;

tm -время появления очередной заявки на выходе источника;

tkj -время окончания обслуживания каналом j k-й фазы К очередной заявки;

Zkj (tn) - состояние канала j фазы k в момент t;

Li - емкость накопителя i-й фазы;

Zi - состояние накопителя i-й фазы;

N1 - количество потерянных заявок;

N3 - количество обслуженных системой заявок;

P - вероятность отказа (потери) заявки системой;

∆t - интервал продвижения модельного времени в сплошном моделировании.

Каждый из каналов Q-схемы может находиться в следующих состояниях:

1) канал свободен (0);

2) канал занят обслуживанием (I);

3) канал заблокирован (хранит уже обслуженную заявку)(2). Текущее состояние Z накопителя Н равно количеству заявок , хранящемуся в накопителе в текущий момент модельного времени t.

#### Алгоритм моделирования Q-схемы

Процедура моделирования начала обслуживания заявки каждым элементарным каналом Кij - сводится к следующему.

Выполняется обращение к генератору случайных чисел. Генератор формирует интервал обслуживания заявки каналом Кij, закон распределения , длительности которого должен соответствовать закону F распределения времени обслуживания заявок каналом Кij. Вычисляется время окончания обслуживания tij=tn+τij, где tn - текущий момент модельного времени. Канал Кij переходит в состояние "занят обслуживанием".

Когда модельное время достигает значения tij, соответствующего моменту завершения обслуживания каналом Кij, моделируется процесс передачи заявки с выхода канала Кij в накопитель Нi+1 следующей фазы или в каналы фазы i+1, если ёмкость Li+1 накопителя Н1+1 равна нулю. Если фаза i+1 может принять заявку, то канал Kij переводится в состояние "свободен". В этом случае количество заявок в накопителе Нi+1 фазы i+1 увеличивается на 1, а канал Кij может принять заявку из накопителя Нi своей фазы. Канал Кij переходит в состояние "занят обслуживанием", а количество заявок в накопителе Нi уменьшается на единицу. Если фаза i+1 заявку принять не может (накопитель и каналы заняты обслуживанием заявок), канал Кij переводится в состояние "заблокирован".

Укрупнённая схема алгоритма моделирования Q-схемы, построенного по принципу последовательного просмотра состояний модели через фиксированный временной интервал ∆t, представлена на рис.КР.2. Такой метод управления модельным временем называется моделированием с постоянным шагом и состоит в том, что после каждого просмотра состояния модели, модельное время tn увеличивается на интервал ∆t. Наращивание модельного времени tn = tn + ∆t выполняется блоком 10. Момент завершения моделирования Q-схемы может быть зафиксирован по числу просмотров N, по длине интервала времени моделирования Т или по количеству обслуженных заявок N1. Проверка соответствующих условий выполняется блоком 3.

Работа вспомогательных блоков - ввода исходных данных *I,* установки начальных условий 2, обработки *II* и вывода результатов моделирования 12 - не отличается по своей сути от аналогичных блоков, используемых в алгоритмах вычислений на ЭВМ. Поэтому остановимся более детально на работе той части моделирующего алгоритма, которая отражает специфику моделирования подхода (блоки 4-9). Детализированные схемы алгоритмов этих блоков приведены на рис. КР.3 - рис КР.8. На этих и последующих схемах моделирующих алгоритмов Q-схем приняты следующие обозначения:

ZN(1) = z , Z(i, J) = zij, TM=tm, TN=tn, T(i,J) = tij, LO(I) =Li, PO = P.

Процедура формирования времени завершения обслуживания заявок каналами Кij оформлена в виде подрограммы WORK (T(K,J)). Процедура генерирует, tkj - длительность интервала обслуживания очередной заявки и формирует время завершения обслуживания t(K,j ) = tn+ Tkj. Окончание обслуживания заявки в некотором канале Кij в момент времени tn может вызвать процесс распространения изменений состояний элементов ("особых состояний") системы в направлении противоположном движению заявок в системе, поэтому все Н (накопители) и К (каналы) системы должны просматриваться при моделировании, начиная с обслуживающего канала последней фазы по направлению к накопителю 1-й фазы (см. рис. . ).

#### Алгоритм формирования очередного состояния Q-схемы в дискретный фиксированный момент модельного времени

Рассмотрим реализацию основных блоков моделирующего алгоритма. Это блоки 9, 8..... 4, которые имитируют формирование заявок источником и их обслуживание в каналах 1-й, 2-й и 3-й фаз модели.

Рассмотрим состояние модели Q-схемы на стационарном участке моделирования. Пусть после очередного выполнения блока 10 модельное время приняло значение tn.

Блок 4 (рис.КР.3) имитирует завершение обслуживания заявок каналом К31 третьей фазы. Блок 4.1 проверяет состояние канала К31 и, если канал находится в состоянии "занят обслуживанием" ("I"), то в блоке 4.2 проверяется время Т31 завершения обслуживания каналом К31. Если это время меньше или совпадает с текущим модельным временем tn, то это означает, что в момент tn на выходе К31 появляется очередная заявка. В этом случае в блоке 4.3 увеличивается на 1 количество обслуженных заявок N3, а в блоке 4.4 канал К31 переводится в состояние "свободен" ("О").

Блок 5 (рис.КР.4), имитирует завершение обслуживания заявок каналами 2-й фазы и передачу обслуженных заявок в 3-ю фазу. Блоки 5.1, 5.9 и 5.10 составляют цикл просмотра каналов 2-й фазы. Блоки 5.2 и 5.З проверяют состояние и время завершения обслуживания заявки каждым из каналов. Если для некоторого канала j его состояние Z2j=0, т.е. он находится в состоянии "занят обслуживанием" или "заблокирован", T2j<=Tn, то это означает, что канал K2j хранит ранее заблокированную заявку ( Z2j = 2 и Т2j < Тn) или именно в момент Тn он завершил обслуживание (T2j = Tn, Z2j < 1). B этих случаях блок 5.4 проверяет состояние канала 3.1 3-й фазы. Если этот канал не свободен (Z31≠0), то блок 5.5 переводит канал Kij в состояние "заблокирован" (или подтверждает ранее установленное состояние "заблокирован"). Если Z31=0, то блок 5.6 формирует новое время завершения обслуживания заявки каналом К31, блок 5.7 переводит канал К31 в состояние "занят обслуживанием", а блок 5.8 освобождает канал K2j.

Блок 6 (рис.КР.5) имитирует процесс передачи заявок из накопителя Н2 второй фазы в каналы К21, К22. Блоки 6.1, 6.7, 6.8 составляют цикл просмотра состояния каналов второй фазы. Блок 6.2 проверяет состояние накопителя Н2. Если накопитель Н2 содержит хотя бы одну заявку (ZN(2)≠0), выполняется переход к блоку 6.3, который проверяет состояние очередного канала 2-й фазы. Если j-й канал свободен (Z(2,j)=0), то в блоке 6.4 вычисляется время завершения обслуживания заявки каналом K2j, блок 6.5 переводит канал К2j в состояние "занят обслуживанием", а блок 6.6 уменьшает на единицу количество заявок в накопителе Н2. Если при выполнении блока 6.2 оказывается, что накопитель Н2 заявок не содержит (ZN(2)=0), то выполняется переход к блоку 7.

Блок 7 (рис.КР.6) воспроизводит процесс передачи заявок из каналов 1-й фазы в накопитель и каналы 2-й фазы. Блоки 7.1, 7.15, 7.16 составляют цикл просмотра состояния каналов 1-й фазы. Если при выполнении блоков 7.2,7.3 оказывается, что некоторый канал К1j хранит заявку в состоянии "заблокирован" или выработал заявку в момент tn, выполняется переход к блокам 7.4, 7.5, 7.6, 7.7, составляющим цикл просмотра состояния каналов 2-й фазы. Если в результате выполнения в цикле блока 7.5 находится некоторый канал К2i в состоянии "свободен" (Z(2,i)=0), то выполняются блоки 7.8, 7.9, 7.10. Эти блоки формируют время завершения обслуживания заявки каналом К2i, канал К2i переводится в состояние "занят обслуживанием", а канал К1j переводится в состояние "свободен".

Если в результате просмотра каналов 2-Й фазы все каналы оказываются занятыми, в блоке 7.11 проверяется состояние накопителя Н2 .Если накопитель содержит свободные позиции (ZN(2)<L(2)), выполняются блоки 7.12, 7.14, увеличивающие на 1 количество заявок в накопителе Н2 и переводящие канал Кij в состояние "свободен". Если накопитель Н2 полностью заполнен, выполняется блок 7.18, переводящий канал Kij в состояние "заблокирован".

Детальный алгоритм блока 8 приведен на рис. КР.7. Блок имитирует процесс передачи заявок из накопителя Н1-й фазы в каналы 1-й фазы. Структура алгоритма полностью аналогична блоку 6.

Блок 9 (рис.КР.8) воспроизводит поступление заявок из источника U на вход 1-й фазы. Если при выполнении 9.1 удовлетворяется Tm ≤ Тn, то это означает, что в момент tn на выходе источника сформирована очередная заявка. Блоки 9.2, 9.6, 9.7 составляют цикл просмотра состояния каналов 1-й фазы. Если в результате просмотра блок 9.3 обнаружит свободный канал К1j, выполняются блоки 9.4, 9.5. Они формируют время завершения обслуживания заявки Tij каналом Кij и переводят канал Кij в состояние "занят обслуживанием".

Если свободных каналов в 1-й фазе нет, то анализируется состояние накопителя Н (9.7). Если накопитель содержит свободную позицию (ZМ(1)<L(1)), блок 9.9 увеличивает на 1 количество заявок в накопителе. Если накопитель заполнен, блок 9.10 увеличивает на 1 количество заявок получивших отказ. Во всех случаях в блоке 9.11 вычисляется момент времени t поступления очередной заявки источника на вход системы.

**H1**

**N1**

**H2**

**I**

**II**

**III**

**N3**

Рис. КР.1. Структура Q-схемы

**ВХОД**

**ВВОД**

**ИСХОДНЫХ ДАННЫХ**

**УСТАНОВКА НАЧАЛЬНЫХ УСЛОВИЙ**

**ПРОВЕРКА ОКОНЧАНИЯ МОДЕЛИР.**

**Обслуживание заявки каналом 3-й фазы**

**Переход заявки из 2-й фазы в 3-ю**

**Обслуживание заявки каналом 2-й фазы**

**Переход заявки из 1-й фазы в накопитель 2-й фазы**

**Обслуживание заявки каналом 1-й фазы**

**Поступление заявки на вход Q-схемы**

**Переход к следующему моменту времени**

**Обработка результатов**

**Вывод результатов**

**ВЫХОД**

**ДА**

**НЕТ**

**1**

**2**

**3**

**4**

**5**

**6**

**7**

**8**

**9**

**10**

**11**

**12**

Рис. КР.2. Укрупненный алгоритм модели Q-схемы

**4**

**Z(3,1)=1**

**N3=N3+1**

**Z(3,1)=0**

**T(3,1)=TN**

**5**

**НЕТ**

**НЕТ**

**ДА**

**ДА**

**4.1**

**4.2**

**4.3**

**4.4**

Рис. КР.3 Блок-схема алгоритма блока 4

**5**

**WORK(T(3,1)**

**J(2,J)=2**

**Z(2,J)≠0**

**6**

**НЕТ**

**5.1**

**J=1**

**T(2,J)≤TN**

**Z(3,1)=0**

**Z(3,1)=1**

**Z(2,J)=0**

**J≥2**

**J=J+1**

**5.10**

**НЕТ**

**ДА**

**5.9**

**ДА**

**5.6**

**5.7**

**5.8**

**5.4**

**ДА**

**НЕТ**

**5.5**

**5.2**

**ДА**

**5.3**

Рис. КР.4. Блок-схема алгоритма блока 5

**6**

**WORK(K(2,J))**

**ZN(2)>0**

**7**

**НЕТ**

**6.1**

**J=1**

**Z(2,J)=0**

**Z(2,J)=1**

**ZN(2)=ZN(2)-1**

**J≥2**

**J=J+1**

**6.8**

**НЕТ**

**ДА**

**6.7**

**6.5**

**6.6**

**6.4**

**ДА**

**НЕТ**

**6.2**

**ДА**

**6.3**

Рис. КР.5. Блок-схема алгоритма блока 6

**7**

**J=1**

**Z(1,J)≠0**

**Z(1,J)≤TN**

**I=1**

**Z(2,I)≠0**

**I≥2**

**ZN(2)<L(2)**

**ZN(2)=ZN(2)+1**

**Z(1,J)=0**

**J≥2**

**WORK(K(2,I))**

**Z(2,I)=1**

**Z(1,J)=0**

**I=I+1**

**8**

**Z(1,J)=2**

**J=J+1**

**7.1**

**7.2**

**7.3**

**7.4**

**7.5**

**7.6**

**7.7**

**7.8**

**7.9**

**7.10**

**7.11**

**7.12**

**7.14**

**7.13**

**7.16**

**7.15**

**ДА**

**ДА**

**ДА**

**ДА**

**НЕТ**

**НЕТ**

**НЕТ**

**НЕТ**

**НЕТ**

**НЕТ**

**ДА**

**ДА**

Рис. КР.6. Блок-схема алгоритма блока 7

**8**

**J=1**

**ZN(1)≠0**

**Z(1,J)≠0**

**WORK(K(1,J))**

**Z(1,J)=1**

**ZN(1)=ZN(1)-1**

**J=J+1**

**J≥2**

**9**

**ДА**

**НЕТ**

**НЕТ**

**НЕТ**

**8.1**

**8.2**

**8.3**

**8.4**

**8.5**

**8.6**

**8.8**

**8.7**

Рис. КР.7. Блок-схема алгоритма блока 8.

**9**

**J=1**

**TM=TN**

**Z(1,J)=0**

**J=J+1**

**ДА**

**НЕТ**

**НЕТ**

**9.1**

**9.3**

**WORK(K(1,J))**

**Z(1,J)=1**

**J≥2**

**ДА**

**ZN(1)<L(1)**

**N1=N1+1**

**ZN(1)=ZN(1)+1**

**D(TM)**

**10**

**НЕТ**

**НЕТ**

**ДА**

**9.1**

**9.4**

**9.5**

**9.6**

**9.8**

**9.9**

**9.11**

**9.10**

**9.7**

Рис. КР.8. Блок-схема алгоритма блока 9.

**ЗАДАНИЕ**

1. Построить аналитические модели компонентов СМО.

2. Построить имитационную модель СМО.

3. Сравнить результаты моделирования, полученные аналитическим и имитационным методами