

# **ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ЭКОНОМИЧЕСКИХ ПРОЦЕССОВ**

Учебное пособие  
для слушателей программы eMBA



Москва 2005

**СОСТАВИТЕЛЬ:**

кандидат экономических наук, доцент  
Н.Н. ЛЫЧКИНА

**ОТВЕТСТВЕННЫЙ РЕДАКТОР:**

заведующий кафедрой информационных систем  
доктор экономических наук, профессор  
В.В. ГОДИН

**РЕЦЕНЗЕНТЫ**

академик РАЕН, д.ф.-м. н. Д.Л. АНДРИАНОВ,  
к.т.н. В.В. ДЕВЯТКОВ

**ОБСУЖДЕН**

на заседании кафедры информационных систем  
Протокол № 5 от " 29 " декабря 2005г.

**ОБСУЖДЕН И ОДОБРЕН**

научно-методической комиссией по специальности  
"Прикладная информатика в управлении"  
Протокол № 5 от "29" декабря 2004г.

© Лычкина Н.Н., 2005

© Академия АйТи, 2005

© Государственный университет управления, 2005

## Оглавление

<b>ЛЕКЦИЯ 1 КРАТКИЙ ЭКСКУРС В СИСТЕМНЫЙ АНАЛИЗ. ПОНЯТИЕ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ. . . . .</b>	<b>5</b>
1.1 Свойства сложных систем. Сложная система, как объект моделирования. Прикладной системный анализ – методология исследования сложных систем. . . . .	5
1.2 Определение модели. Общая классификация основных видов моделирования. Компьютерное моделирование. Метод имитационного моделирования. . . . .	8
1.3 Процедурно-технологическая схема построения и исследования моделей сложных систем. Основные понятия моделирования. . . . .	11
1.4 Метод статистического моделирования на ЭВМ (метод Монте-Карло). . . . .	16
1.5 Выводы. Отличительные особенности моделей различных классов. . . . .	17
<b>ЛЕКЦИЯ 2 СУЩНОСТЬ МЕТОДА ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ. . . . .</b>	<b>19</b>
2.1 Метод имитационного моделирования и его особенности. Статическое и динамическое представление моделируемой системы. . . . .	19
2.2 Понятие о модельном времени. Механизм продвижения модельного времени. Дискретные и непрерывные имитационные модели. . . . .	22
2.3 Моделирующий алгоритм. Имитационная модель. . . . .	23
2.4 Проблемы стратегического и тактического планирования имитационного эксперимента. Направленный вычислительный эксперимент на имитационной модели. . .	24
2.5 Общая технологическая схема имитационного моделирования. . .	27
2.6 Возможности, область применения имитационного моделирования. . . . .	27
<b>ЛЕКЦИЯ 3 ТЕХНОЛОГИЧЕСКИЕ ЭТАПЫ СОЗДАНИЯ И ИСПОЛЬЗОВАНИЯ ИМИТАЦИОННЫХ МОДЕЛЕЙ. . . . .</b>	<b>29</b>
3.1 Основные этапы имитационного моделирования. Общая технологическая схема. . . . .	29

---

3.2	Формулировка проблемы и определение целей имитационного исследования. ....	30
3.3	Разработка концептуальной модели объекта моделирования. ...	34
3.4	Формализация имитационной модели. ....	38
3.5	Программирование имитационной модели. ....	40
3.6	Сбор и анализ исходных данных. ....	40
3.7	Испытание и исследование свойств имитационной модели. ....	42
3.8	Направленный вычислительный эксперимент на имитационной модели. Анализ результатов моделирования и принятие решений. ....	43
	<b>ЛЕКЦИЯ 4 БАЗОВЫЕ КОНЦЕПЦИИ СТРУКТУРИЗАЦИИ И ФОРМАЛИЗАЦИИ ИМИТАЦИОННЫХ СИСТЕМ. ....</b>	<b>44</b>
4.1	Методологические подходы к построению дискретных имитационных моделей. ....	44
4.2	Язык моделирования GPSS. ....	45
4.3	Агрегативные модели. ....	53
4.4	Сети Петри и их расширения. ....	59
4.5	Модели системной динамики. ....	67
	<b>ЛЕКЦИЯ 5 ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА АВТОМАТИЗАЦИИ МОДЕЛИРОВАНИЯ. ....</b>	<b>88</b>
5.1	Назначение языков и систем моделирования. ....	88
5.2	Классификация языков и систем моделирования, их основные характеристики. ....	91
5.3	Технологические возможности систем моделирования. ....	93
5.4	Развитие технологии системного моделирования. ....	96
5.5	Выбор системы моделирования. ....	100
	<b>ЛЕКЦИЯ 6 ИСПЫТАНИЕ И ИССЛЕДОВАНИЕ СВОЙСТВ ИМИТАЦИОННОЙ МОДЕЛИ. ....</b>	<b>102</b>
6.1	Комплексный подход к тестированию имитационной модели. ..	102
6.2	Проверка адекватности модели. ....	105
6.3	Верификация имитационной модели. ....	108
6.4	Валидация данных имитационной модели. ....	109

6.5 Оценка точности результатов моделирования. ....	110
6.6 Оценка устойчивости результатов моделирования. ....	111
6.7 Анализ чувствительности имитационной модели. ....	112
6.8 Тактическое планирование имитационного эксперимента. ....	114
<b>ЛЕКЦИЯ 7 ТЕХНОЛОГИЯ ПОСТАНОВКИ И ПРОВЕДЕНИЯ НАПРАВЛЕННОГО ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА НА ИМИТАЦИОННОЙ МОДЕЛИ. . . .</b>	<b>118</b>
7.1 Направленный вычислительный эксперимент на имитационной модели и его содержание. ....	118
7.2 Основные цели и типы вычислительных экспериментов в имитационном моделировании. ....	121
7.3 Основы теории планирования экспериментов. Основные понятия: структурная, функциональная и экспериментальная модели. ....	124
7.4 План однофакторного эксперимента и процедуры обработки результатов эксперимента. ....	130
7.5 Факторный анализ, полный и дробный факторный эксперимент и математическая модель. ....	133
7.6 Основные классы планов, применяемые в вычислительном эксперименте. ....	144
7.7 Методология анализа поверхности отклика. Техника расчета крутого восхождения. ....	148
<b>СПИСОК ЛИТЕРАТУРЫ . . . . .</b>	<b>160</b>



## ЛЕКЦИЯ

## 1

**КРАТКИЙ ЭКСКУРС В  
СИСТЕМНЫЙ АНАЛИЗ.  
ПОНЯТИЕ КОМПЬЮТЕРНОГО  
МОДЕЛИРОВАНИЯ.****1.1 Свойства сложных систем. Сложная система, как объект моделирования. Прикладной системный анализ – методология исследования сложных систем.**

В настоящее время понятие "система" в науке является до конца не определенным. Ученые приступили к исследованию сложных систем (СС).

В многочисленной литературе по системному анализу и системотехнике [11,24,38] отмечаются следующие основные **свойства сложных систем**:

**1 свойство: Целостность и членимость.**

Сложная система рассматривается как целостная совокупность элементов, характеризующаяся наличием большого количества взаимосвязанных и взаимодействующих между собой элементов.

У исследователя существует субъективная возможность разбиения системы на подсистемы, цели функционирования которых подчинены общей цели функционирования всей системы (*целенаправленность систем*). **Целенаправленность** интерпретируется, как способность системы осуществлять в условиях неопределенности и воздействия случайных факторов поведение (выбор поведения), преследующее достижение определенной цели.

**2 свойство: Связи.**

*Наличие существенных устойчивых связей (отношений) между элементами или (и) их свойствами, превосходящими по мощности (силе) связи (отношения) этих элементов с элементами, не входящими в данную систему (внешней средой).*

Под "связями" понимается некоторый виртуальный канал, по которому осуществляется обмен между элементами и внешней средой веществом, энергией, информацией.

**3 свойство: Организация.**

Свойство характеризуется наличием определенной организации – формированием существенных связей элементов, упорядоченным распределением связей и элементов во времени и пространстве. При

формировании связей складывается определенная **структура** системы, а свойства элементов трансформируются в **функции** (действия, поведение).

- При исследовании сложных систем обычно отмечают:
- Сложность функции, выполняемой системой и направленной на достижение заданной цели функционирования;
- Наличие управления, разветвленной информационной сети и интенсивных потоков информации;
- Наличие взаимодействия с внешней средой и функционирование в условиях неопределенности и воздействия случайных факторов различной природы.

#### **4 свойство: Интегративные качества.**

*Существование интегративных качеств (свойств), т.е. таких качеств, которые присущи системе в целом, но не свойственны ни одному из ее элементов в отдельности. Наличие интегративных качеств показывает, что свойства системы хотя и зависят от свойств элементов, но не определяются ими полностью.*

Примеры СС в экономической сфере многочисленны: организационно – производственная система, предприятие; социально – экономическая система, например регион; и др.

**СС, как объект моделирования**, имеет следующие характерные особенности:

- СС, как правило, уникальны. Существующие аналоги таких объектов заметно отличаются друг от друга. Следствием этого на практике является необходимость строить новые модели.
- Слабая структурированность теоретических и фактических знаний о системе. Так как изучаемые системы уникальны, то процесс накопления и систематизации знаний о них затруднен. Слабо изучены сами процессы. При идентификации сложных систем присутствует большая доля субъективных экспертных знаний о системе. СС слабопредсказуемы или контринтуитивны, как писал Форрестер.
- Рассмотренные выше интегративные качества СС предопределяют важный методологический вывод: СС не сводится к простой совокупности элементов, расчленив СС на отдельные части, изучая каждую из них в отдельности, нельзя познать свойства системы в целом. Поэтому *описание отдельных подсистем необходимо выполнять с учетом их места во всей системе в целом, и наоборот, система в целом исследуется исходя из свойств отдельных подсистем.* Одну из основных черт сложных систем составляет взаимодействие выделенных подсистем. Необходимо учитывать результат воздействия одной подсистемы на другую и их взаимодействие с внешней средой. Исследователи отмечают наличие



большого числа взаимосвязанных подсистем, многомерность СС, обусловленную большим числом связей между подсистемами, что затрудняет идентификацию моделируемых объектов. Отметим также, что расчленение СС на подсистемы зависит от целей создания системы и взглядов исследователя на нее.

- Разнородность подсистем и элементов, составляющих систему. Это определяется и многообразием природы (физической разнородностью подсистем, имеющих различную природу), и *разнородностью математических схем*, описывающих функционирование различных элементов, а также одних и тех же элементов на различных уровнях изучения.
- Присутствует необходимость исследовать систему *в динамике*, с учетом поведенческих аспектов.
- *Случайность и неопределенность факторов*, действующих в изучаемой системе. Учет этих факторов приводит к резкому усложнению задач и увеличивает трудоемкость исследований (необходимость получения представительного набора данных). Существует необходимость учета большого количества действующих в системе факторов.
- *Многокритериальность* оценок процессов, протекающих в системе. Невозможность однозначной оценки (выбора единого обобщенного критерия) диктуется следующими обстоятельствами:
  - наличием множества подсистем, каждая из которых, вообще говоря, имеет свои цели, оценивается по своим локальным критериям;
  - *множественностью показателей* (при системном подходе иногда противоречивых, — в этом случае, выбирается компромиссный вариант), характеризующих работу всей системы;
  - наличием неформализуемых критериев, используемых при принятии решений, основанных на практическом опыте лиц, принимающих решение.
- При системном подходе *процесс исследования СС носит итерационный характер*. Исходная модель усложняется путем детализации. Однако создание полной модели СС (супермодели) бесполезно, т.к. она будет столь же сложна в изучении, как и система. Следствием этого является необходимость использования *ансамбля (комплекса) моделей* при анализе системы. Различные модели могут отражать как разные стороны функционирования системы, так и разные уровни отображения исследователем одних и тех же процессов.

Рассмотренные особенности исследования сложных систем обуславливают потребность в специальных способах построения и анализа

моделей сложных систем. Традиционные аналитические модели здесь беспомощны — нужны специальные компьютерные технологии.

**Методологией исследования СС является системный анализ.** Один из важнейших инструментов прикладного системного анализа — компьютерное моделирование. Имитационное моделирование является наиболее эффективным и универсальным вариантом компьютерного моделирования в области исследования и управления сложными системами.

## **1.2 Определение модели. Общая классификация основных видов моделирования. Компьютерное моделирование. Метод имитационного моделирования.**

Определение 1. **Модель** представляет собой абстрактное описание системы (объекта, процесса, проблемы, понятия) в некоторой форме, отличной от формы их реального существования.

Определение 2. **Моделирование** представляет собой один из основных методов познания, является формой отражения действительности и заключается в выяснении или воспроизведении тех или иных свойств реальных объектов, предметов и явлений с помощью других объектов, процессов, явлений, либо с помощью абстрактного описания в виде изображения, плана, карты, совокупности уравнений, алгоритмов и программ.

Итак, в процессе моделирования всегда существует *оригинал* (объект) и *модель*, которая воспроизводит (моделирует, описывает, имитирует) некоторые черты объекта.

Моделирование основано на наличии у многообразия естественных и искусственных систем, отличающихся как целевым назначением, так и физическим воплощением, сходства или подобия некоторых свойств: геометрических, структурных, функциональных, поведенческих. Это сходство может быть полным (**изоморфизм**) и частичным (**гоморфизм**).

Моделирование появилось в человеческой деятельности со времен наскальной живописи и сооружения идолов, т.е. как только человечество стало стремиться к пониманию окружающей действительности; — и сейчас, по-существу, прогресс науки и техники находит свое наиболее точное выражение в развитии способности человека создавать модели объектов и понятий.

Исследуя современные СС, человечество придумало различные *классы моделей*. Развитие информационных технологий можно в известном смысле интерпретировать как возможность реализации моделей различного вида в рамках информационных систем различного

назначения: Информационные системы, Системы распознавания образов, Системы искусственного интеллекта, Системы поддержки принятия решений. В основе этих систем лежат модели различных типов: семантические, логические, математические и т.п.

Приведем общую **классификацию основных видов моделирования** [10]:

- **концептуальное моделирование** — представление системы с помощью специальных знаков, символов, операций над ними или с помощью естественных или искусственных языков,
- **физическое моделирование** — моделируемый объект или процесс воспроизводится исходя из соотношения подобия, вытекающего из схожести физических явлений;
- **структурно – функциональное** — моделями являются схемы (блок-схемы), графики, диаграммы, таблицы, рисунки со специальными правилами их объединения и преобразования;
- **математическое (логико-математическое) моделирование** — построение модели осуществляется средствами математики и логики;
- **имитационное (программное) моделирование** — при котором логико-математическая модель исследуемой системы представляет собой алгоритм функционирования системы, программно-реализуемый на компьютере.

Указанные виды моделирования могут применяться самостоятельно или одновременно, в некоторой комбинации (например, в имитационном моделировании используются практически все из перечисленных видов моделирования или отдельные приемы).

Доминирующей тенденцией сегодня является взаимопроникновение всех видов моделирования, симбиоз различных информационных технологий в области моделирования, особенно для сложных приложений и комплексных проектов по моделированию. Так, например, имитационное моделирование включает в себя концептуальное моделирование (на ранних этапах формирования имитационной модели) и логико-математическое (включая методы искусственного интеллекта) — для целей описания отдельных подсистем модели, а также в процедурах обработки и анализа результатов вычислительного эксперимента и принятия решений. Технология проведения и планирования вычислительного эксперимента с соответствующими математическими методами привнесена в имитационное моделирование из физического (натурного) моделирования. Наконец, структурно-функциональное моделирование используется как при создании стратифицированного описания многомодельных комплексов, так и для формирования различных диаграммных представлений при создании имитационных моделей.

Понятие компьютерного моделирования сегодня трактуется [10] шире традиционного понятия "моделирование на ЭВМ", поэтому нуждается в уточнении.

**Компьютерное моделирование** – метод решения задач анализа или синтеза сложной системы на основе использования ее компьютерной модели.

К компьютерному моделированию относят:

- *структурно-функциональное,*
- *имитационное.*

Под термином "компьютерная модель", чаще всего понимают:

- Условный образ объекта или некоторой системы объектов (или процессов), описанный с помощью взаимосвязанных компьютерных таблиц, блок-схем, диаграмм, графиков, рисунков, анимационных фрагментов, гипертекстов и т.д. и отображающих структуру и взаимосвязи между элементами объекта. Компьютерные модели такого вида мы будем называть *структурно-функциональными*;
- Отдельную программу (совокупность программ, программный комплекс) позволяющий с помощью последовательности вычислений и графического отображения их результатов, воспроизводить (имитировать) процессы функционирования объекта, системы объектов при условии воздействия на объект различных, как правило, случайных факторов. Такие модели мы будем называть *имитационными*.

Суть компьютерного моделирования заключена в получении количественных и качественных результатов на имеющейся модели. Качественные результаты анализа обнаруживают неизвестные ранее свойства сложной системы: ее структуру, динамику развития, устойчивость, целостность и др. Количественные выводы в основном носят характер анализа существующей СС или прогноза будущих значений некоторых переменных. Кстати, возможность получения не только качественных, но и количественных результатов составляет существенное отличие имитационного моделирования от структурно-функционального. Становление компьютерного моделирования связано с имитационным моделированием. Имитационное моделирование было исторически первым по – сравнению со структурно-функциональным, без компьютера никогда не существовало. Имитационное моделирование имеет целый ряд специфических черт.

**Методологией компьютерного моделирования является системный анализ** (направление кибернетики, общая теория систем). Поэтому в освоении этого метода доминирующая роль отводится системным аналитикам. Сравним с моделированием на ЭВМ (например, математическим). Методологической основой здесь чаще всего являются:

исследование операций, теория математических моделей, теория принятия решений, теория игр и многие другие.

*Центральной процедурой системного анализа является построение обобщенной модели, отражающей все факторы и взаимосвязи реальной системы.* Предметом компьютерного моделирования может быть любая сложная система, любой объект или процесс. Категории целей при этом могут быть самыми различными. Компьютерная модель должна отражать все свойства, основные факторы и взаимосвязи реальной сложной системы, критерии, ограничения.

Компьютерное моделирование сегодня предлагает совокупность методологических подходов и развитых технологических средств, используемых для подготовки и принятия решений экономического, организационного и социального или технического характера.

### **1.3 Процедурно–технологическая схема построения и исследования моделей сложных систем. Основные понятия моделирования.**

Выше мы рассмотрели различные классы моделей. Выбор метода моделирования для решения поставленной задачи, проблемы, исследования системы — является актуальной задачей, с которой системный аналитик должен уметь справляться.

С этой целью давайте уточним место имитационных моделей и их специфику среди моделей других классов. Кроме того, попробуем уточнить некоторые понятия и определения, с которыми имеет дело системный аналитик в процессе моделирования, которые не всегда и везде трактуются однозначно и достаточно корректно. С этой целью рассмотрим *процедурно-технологическую схему построения и исследования моделей сложных систем* (рис.1.3). Эта схема включает следующие этапы определения, характерные для любого метода моделирования:

1. системы (предметная, проблемная область);
2. объекта моделирования;
3. целевого назначения моделей;
4. требований к моделям;
5. формы представления;
6. вида описания модели;
7. характера реализации модели;
8. метода исследования модели.

Первые три этапа характеризуют *объект и цель исследования* и практически определяют следующие этапы моделирования. При этом важное значение приобретает корректное описание объекта и

формулировка цели моделирования из предметной области деятельности исследователя.

1) **Предметная (проблемная) область** определяется физическими, химическими, техническими, технологическими, информационными, биологическими, экологическими, экономическими, социальными и другими возможными классами систем. (Сфера применения имитационных моделей весьма обширна, имитационное моделирование используется для исследования самых разнообразных систем: экономических, производственных, социальных, транспортных, систем массового обслуживания, вычислительных, информационных, включая международную деятельность, проблемы развития городов, глобальные (мировые) проблемы и многие другие.)

2) В качестве **объекта моделирования** в процессе исследования выступает не вся система, а ее "срез" — элемент, структура, отношение, организация, функция, отдельные процессы, поведение, развитие и т.д.

3) Каждая модель должна строиться **целенаправленно**.

Целенаправленная модель представляет собой замену действительности с той степенью абстракции, которая полезнее для поставленной цели. Иначе говоря, модель прежде всего должна отражать те существенные свойства, те стороны моделируемого объекта, которые определены практической задачей.

Очень важно правильно обозначить и сформулировать проблему, четко задать **цель** исследования, проводимого с помощью моделирования. Допустим, решено, что для решения поставленной задачи необходимо использовать моделирование.

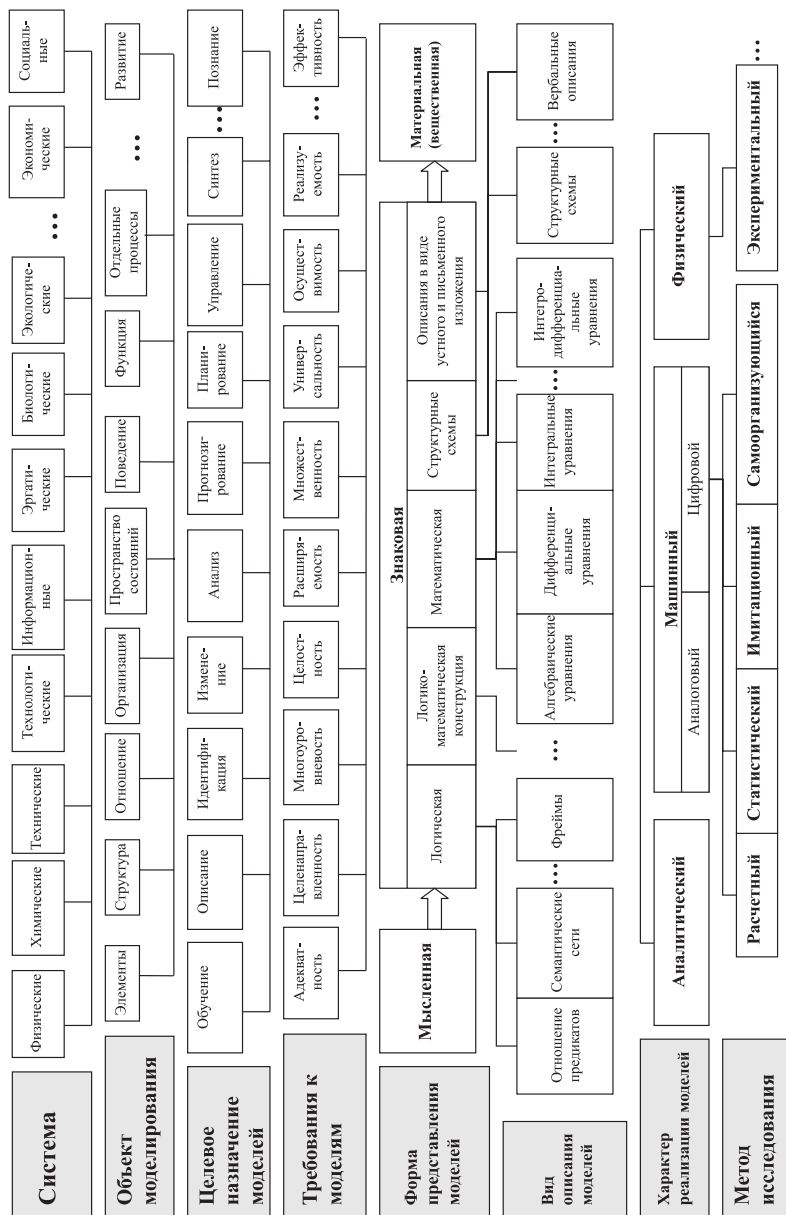


Рисунок 1.3 – Общая процедурно-технологическая схема построения и исследования сложных систем.

Моделирование успешно используется при решении большого круга **задач**: обучение, описание, сжатие, измерение, оценивание, прогнозирование, планирование, управление, синтез, идентификация, познание и др. Моделирование становится мощным средством в задачах анализа и синтеза сложных технических и экономических систем, или поиска эффективных решений (сложные разработки, проекты, новые программы).

Сложные системы необязательно требуют сложных моделей. Надо стремиться создавать простые модели, они более качественные.

Мышление исследователя, разработчика в зависимости от цели моделирования либо формирует модели реально существующих объектов в познавательной деятельности, либо создает идеальные модели еще не существующих систем в задачах проектирования. В тех и других случаях модели должны быть пригодны для решения поставленных задач.

4) *Требования к моделям.* Какими чертами должна обладать хорошая модель?

Моделирование связано с решением реальных задач и мы должны быть уверены, что результаты точно отражают истинное положение вещей — т.е. модель адекватна реальной действительности. Надо думать и о заказчике. Плоха та модель, которую не сможет использовать лицо, принимающее решение.

Чтобы получить хорошую модель, она должна удовлетворять некоторым общепринятым требованиям. Такая модель должна быть:

- адекватной;
- надежной в смысле гарантии от абсурдных ответов;
- простой и понятной пользователю;
- целенаправленной;
- удобной в управлении и обращении, т.е. общение с ней должно быть легким;
- функционально полной с точки зрения возможностей решения главных задач;
- адаптивной, позволяющей легко переходить к другим модификациям или обновлять данные;
- допускающей постепенные изменения в том смысле, что, будучи вначале простой, она может во взаимодействии с пользователем становиться все более сложной.

Это очевидные моменты с позиции здравого смысла.

В зависимости от целевой направленности модели задаются специальные требования к самим моделям, наиболее характерными являются: целостность, отражение информационных свойств,



многоуровневость, множественность (многомодельность), расширяемость, универсальность (абстрактность), осуществимость (реальная возможность построения самой модели и ее исследования), реализуемость (например, на ЭВМ) (возможность материализации модели в виде реальной системы в задачах проектирования), эффективность (затраты временных, трудовых, материальных и других видов ресурсов на построение моделей и проведение экспериментов находятся в допустимых пределах или оправданы). Значимость или приоритетность требований к модели непосредственно вытекают из назначения модели. Так, например, в познавательных задачах, задачах управления, планирования, описания важным требованием является адекватность модели объективной реальности; в задачах проектирования, синтеза уникальных систем наиболее важным является реализуемость модели, т.е. возможность материализации модели в реальную систему, например в САПР или систему поддержки принятия решений (СППР).

5) Цель моделирования и задание требований к модели, безусловно, определяют **форму представления модели**.

Любая модель (прежде чем стать объективно существующим предметом) должна быть конструктивно разработана, существовать в мысленной форме, далее переведена в знаковую форму и, наконец, материализована, т.е. можно выделить три формы представления моделей:

- мысленные (образы);
- *знаковые* (структурные схемы, описания в виде устного и письменного изложения, логические, математические, логико-математические конструкции);
- материальные (лабораторные и действующие макеты, опытные образцы).

Особое место в моделировании занимают *знаковые*, в частности логические, математические, логико-математические модели, а также модели, воссозданные на основе вербального описания, составленного экспертами. Знаковые модели используются для моделирования многообразных систем. Это направление значимо с бурным развитием вычислительных систем. Ограничимся ими в дальнейшем рассмотрении.

6) Следующий этап процедурной схемы – это выбор **вида описания и построения модели**.

Для знаковых форм такими описаниями могут быть:

- для логических – отношение и исчисление предикатов, семантические сети, фреймы и др. (более подробно их изучают методы искусственного интеллекта);
- для математических – алгебраические, дифференциальные, интегральные, интегро-дифференциальные уравнения и др.

математические схемы. Таким моделям посвящены отдельные направления математической науки.

7) **Характер реализации** знаковых моделей бывает:

- аналитический (например, система дифференциальных уравнений может быть решена математиком на листе бумаги);
- машинный (аналоговый или цифровой);
- физический (или автоматный).

8) В каждом из них в зависимости от сложности модели, цели моделирования, степени неопределенности характеристик модели могут иметь место различные по характеру способы проведения исследований (экспериментов) — **методы исследования**. Так, и при аналитическом исследовании применяются методы теорий возмущений, анализа чувствительности, устойчивости и т.п. При физическом или натурном моделировании применяется экспериментальный метод исследования.

Анализ применяемых и перспективных методов машинного экспериментирования позволяет выделить **расчетный, статистический, имитационный и самоорганизующийся методы исследований**.

*Расчетное моделирование* применяется при исследовании математических моделей и сводится к их машинной реализации при различных числовых исходных данных. Результаты этих реализаций (расчетов) выдаются в графической или табличной формах. До сих пор Вы традиционно имели дело с расчетным моделированием: например, классической схемой является машинная реализация математической модели, представленной в виде системы дифференциальных уравнений, основанная на применении численных методов, с помощью которых математическая модель приводится к алгоритмическому виду, далее программно реализуется на ЭВМ, для получения результатов на ЭВМ проводится расчет (используется расчетный метод исследования).

## **1.4 Метод статистического моделирования на ЭВМ (метод Монте–Карло).**

Под *статистическим моделированием* понимается машинное воспроизведение функционирования вероятностных моделей, либо исследование детерминированных процессов, заданных в виде математических моделей с логическими элементами с помощью статистических испытаний на ЭВМ (метод Монте-Карло). Особенностью статистического моделирования является случайное задание исходных данных с известными законами распределения и, как следствие, вероятностное оценивание характеристик исследуемых процессоров. Статистическое моделирование является эффективным методом исследования слабоорганизованных систем с несложной логикой функционирования.

Для исследования сложных логических и логико-математических моделей с неточным заданием исходных данных (заданным законом распределения, оценочными характеристиками) применяется *имитационное моделирование*. *Имитационное моделирование используется в задачах исследования сложных логико- и логико-математических моделей в результате проведения экспериментов на модели*. Поэтому в имитационном моделировании важную роль играет не только проведение, но и планирование экспериментов.

Следующим за имитационным по качественному уровню можно назвать *самоорганизующееся* моделирование, когда функция построения моделей и ее преобразования в процессе экспериментирования и поиска оптимальных моделей возлагается на ЭВМ.

## **1.5 Выводы. Отличительные особенности моделей различных классов.**

Приведенная классификация делает очевидным некорректность следующих часто допускаемых обобщений:

- Имитационное моделирование отождествляют с машинным, или с моделированием на ЭВМ, с чем, конечно, согласиться нельзя. Об этом мы уже говорили. Основное отличие находится на методологическом уровне. Методологической основой имитационного моделирования является системный анализ. Отдельные элементы, процессы в имитационной модели могут описываться сложными интегральными, дифференциальными и другими уравнениями, реализуются с помощью традиционных вычислительных процедур; т.е. аппарат имитационного моделирования включает все средства, арсенал аналитического моделирования на этапе идентификации имитационной модели. Большое место аналитическим методам отводится и в стратегическом планировании вычислительного эксперимента и при обработке его результатов. Роль аналитических методов в имитационном моделировании постоянно возрастает.
- Имитационное моделирование включает в себя идеи и приемы статистического моделирования на ЭВМ. Метод имитационного моделирования – идеально подходит для исследования стохастических систем, случайных процессов (используются на входе переменные, задаваемые известными законами распределения, можно реализовать вероятностные развития ситуаций, описать случайные процессы, проводить вероятностное оценивание характеристик модели на выходе), т.е. идеи метода Монте-Карло воплощаются в имитационном моделировании, имитационное моделирование исторически выросло из метода статистических испытаний. Однако в случае с имитационным моделированием речь

идет об исследовании сложных систем и решении сложных проблем, в котором отражается структура и динамика моделируемой системы. На ЭВМ реализуются не статистические испытания, а целенаправленные вычислительные эксперименты.

- Не всегда корректно проводится водораздел между математическими и имитационными моделями. Рассмотрим наиболее важные отличия аналитической модели от имитационной:
  - При аналитическом моделировании структура моделируемой системы и процессы ее функционирования представляются в виде некоторых (математических) выражений, отображающих зависимость определяемых характеристик системы от ее параметров и параметров внешней среды. Имитация процессов функционирования систем здесь является вырожденной, она сводится к расчетам по указанным выше выражениям. Иными словами, в аналитических моделях структура моделируемых систем и процессы их функционирования представляются в неявном виде.
  - При имитационном моделировании структура моделируемой системы адекватно отображается в модели, а процессы ее функционирования проигрываются (имитируются) на построенной модели. Построение имитационной модели заключается в описании структуры и процессов функционирования системы. Подробнее об этом пойдет речь в следующей лекции.
  - Кроме того, как отмечалось выше, метод исследования здесь имитационный, основанный на экспериментальном подходе, а не расчетный, как при математическом моделировании.

*Таким образом, имитационное моделирование отличается высокой степенью общности, создает предпосылки к созданию унифицированной модели, легко адаптируемой к широкому классу задач, выступает средством для интеграции моделей различных классов. Т.е. метод имитационного моделирования поднимает моделирование на качественно более высокий уровень.*

## ЛЕКЦИЯ

## 2

**СУЩНОСТЬ МЕТОДА  
ИМИТАЦИОННОГО  
МОДЕЛИРОВАНИЯ.****2.1 Метод имитационного моделирования и его особенности. Статическое и динамическое представление моделируемой системы.**

Определим **метод имитационного моделирования** в самом общем виде как *экспериментальный метод исследования реальной системы по ее имитационной модели, который сочетает особенности экспериментального подхода и специфические условия использования вычислительной техники.*

В этом определении подчеркивается, что имитационное моделирование является машинным методом моделирования, собственно без ЭВМ никогда не существовало, и только развитие информационных технологий привело к становлению этого вида компьютерного моделирования. В этом определении также акцентируется внимание на экспериментальной природе имитации, применяется имитационный метод исследования (осуществляется экспериментирование с моделью). Действительно, в имитационном моделировании важную роль играет не только проведение, но и планирование эксперимента на модели. Однако это определение не проясняет, что собой представляет сама имитационная модель. Попробуем в этой лекции разобраться, в чем же состоит сущность имитационного моделирования.

В процессе имитационного моделирования (рис. 2.1) исследователь имеет дело с четырьмя основными элементами:

- Реальная система;
- Логико-математическая модель моделируемого объекта;
- Имитационная (машинная) модель;
- ЭВМ, на которой осуществляется имитация – направленный вычислительный эксперимент.

Исследователь изучает реальную систему, разрабатывает логико-математическую модель реальной системы. *Имитационный характер исследования предполагает наличие логико- или логико-математических моделей, описываемых изучаемый процесс.*

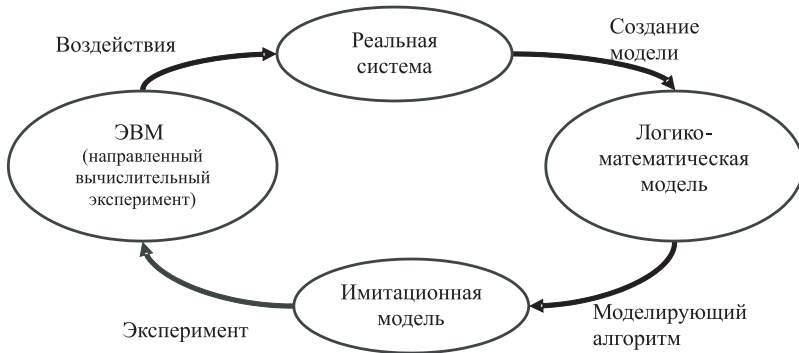


Рисунок 2.1 – Процесс имитационного исследования.

Выше мы определяли *реальную систему* как совокупность взаимодействующих элементов, функционирующих во времени.

Составной характер сложной системы диктует представление ее модели в виде тройки:

$\langle A, S, T \rangle$ , где

$A$  – множество элементов (в их число включается и внешняя среда);

$S$  – множество допустимых связей между элементами (структура модели);

$T$  – множество рассматриваемых моментов времени.

Особенностью имитационного моделирования является то, что имитационная модель позволяет воспроизводить моделируемые объекты:

- с сохранением их логической структуры,
- с сохранением поведенческих свойств (последовательности чередования во времени событий, происходящих в системе), т.е. динамики взаимодействий.

При имитационном моделировании структура моделируемой системы адекватно отображается в модели, а процессы ее функционирования проигрываются (имитируются) на построенной модели. Поэтому построение имитационной модели заключается в описании структуры и процессов функционирования моделируемого объекта или системы. В описании имитационной модели выделяют две составляющие:

- *Статическое описание системы*, которое по-существу является описанием ее структуры. При разработке имитационной модели необходимо выполнять структурный анализ моделируемых процессов.

- *Динамическое описание системы*, или описание динамики взаимодействий ее элементов. При его составлении фактически требуется построение *функциональной модели* моделируемых динамических процессов.

Идея метода, с точки зрения его программной реализации, состояла в следующем. Что если элементам системы поставить в соответствие некоторые программные компоненты, а состояния этих элементов описывать с помощью переменных состояния. Элементы, по определению, взаимодействуют (или обмениваются информацией), — значит может быть реализован алгоритм функционирования отдельных элементов — моделирующий алгоритм. Кроме того, элементы существуют во времени — значит надо задать алгоритм изменения переменных состояний. Динамика в имитационных моделях реализуется с помощью *механизма продвижения модельного времени*.

Отличительной особенностью метода имитационного моделирования является возможность описания и воспроизведения взаимодействия между различными элементами системы. Таким образом, чтобы составить имитационную модель, надо:

- представить реальную систему (процесс), как совокупность взаимодействующих элементов;
- алгоритмически описать функционирование отдельных элементов;
- описать процесс взаимодействия различных элементов между собой и с внешней средой.

Ключевым моментом в имитационном моделировании является выделение и описание *состояний* системы. Система характеризуется *набором переменных состояний*, каждая комбинация которых описывает конкретное состояние. Следовательно, путем изменения значений этих переменных можно имитировать переход системы из одного состояния в другое. Таким образом, имитационное моделирование — это представление *динамического поведения* системы посредством продвижения ее от одного состояния к другому в соответствии с хорошо определенными операционными правилами. Эти изменения состояний могут происходить либо непрерывно, либо в дискретные моменты времени. Имитационное моделирование — есть динамическое отражение изменений состояния системы с течением времени.

Итак, мы разобрались, что при имитационном моделировании логическая структура реальной системы отображается в модели, а также *имитируется* динамика взаимодействий подсистем в моделируемой системе. Это важный, но не единственный признак имитационной модели, исторически предопределивший, не совсем удачное, на мой взгляд, название методу, который серьезный исследователь чаще называют "системным моделированием".

## 2.2 Понятие о модельном времени. Механизм продвижения модельного времени. Дискретные и непрерывные имитационные модели.

Для описания динамики моделируемых процессов в имитационном моделировании реализован механизм *задания модельного времени*. Эти механизмы встроены в управляющие программы любой системы моделирования.

Если бы на ЭВМ имитировалось поведение одной компоненты системы, то выполнение действий в имитационной модели можно было бы осуществить последовательно, по пересчету временной координаты. Чтобы обеспечить имитацию параллельных событий реальной системы вводят некоторую глобальную переменную (обеспечивающую синхронизацию всех событий в системе)  $t_0$ , которую называют *модельным (или системным) временем*.

Существуют два основных способа изменения  $t_0$ :

- *пошаговый* (применяются фиксированные интервалы изменения модельного времени);
- *по-событийный* (применяются переменные интервалы изменения модельного времени, при этом величина шага измеряется интервалом до следующего события).

В случае *пошагового метода* продвижение времени происходит с минимально возможной постоянной длиной шага (*принцип  $t$* ). Эти алгоритмы не очень эффективны с точки зрения использования машинного времени на их реализацию.

*По-событийный метод (принцип "особых состояний")*. В нем координаты времени меняются только когда изменяется состояние системы. В по-событийных методах длина шага временного сдвига максимально возможная. Модельное время с текущего момента изменяется до ближайшего момента наступления следующего события. Применение по-событийного метода предпочтительно в случае, если частота наступления событий невелика, тогда большая длина шага позволит ускорить ход модельного времени. На практике по-событийный метод получил наибольшее распространение.

Способ фиксированного шага применяется:

- если закон изменения от времени описывается интегро-дифференциальными уравнениями. Характерный пример: решение интегро-дифференциальных уравнений численным методом. В подобных методах шаг моделирования равен шагу интегрирования. При их использовании динамика модели является дискретным приближением реальных непрерывных процессов;



- когда события распределены равномерно и можно подобрать шаг изменения временной координаты;
- когда сложно предсказать появление определенных событий;
- когда событий очень много и они появляются группами.

В остальных случаях применяется по-событийный метод. Он предпочтителен, когда события распределены неравномерно на временной оси и появляются через значительные временные интервалы.

Таким образом, вследствие последовательного характера обработки информации в ЭВМ, параллельные процессы, происходящие в модели, преобразуются с помощью рассмотренного механизма в последовательные. Такой способ представления носит название квазипараллельного процесса.

Простейшая классификация на основные виды имитационных моделей связана с применением двух этих способов продвижения модельного времени. Различают имитационные модели:

- *Непрерывные;*
- *Дискретные;*
- *Непрерывно-дискретные.*

В *непрерывных имитационных моделях* переменные изменяются непрерывно, состояние моделируемой системы меняется как непрерывная функция времени, и, как правило, это изменение описывается системами дифференциальных уравнений. Соответственно продвижение модельного времени зависит от численных методов решения дифференциальных уравнений.

В *дискретных имитационных моделях* переменные изменяются дискретно в определенные моменты имитационного времени (наступления событий). Динамика дискретных моделей представляет собой процесс перехода от момента наступления очередного события к моменту наступления следующего события.

Поскольку в реальных системах непрерывные и дискретные процессы часто невозможно разделить, были разработаны *непрерывно-дискретные модели*, в которых совмещаются механизмы продвижения времени, характерные для этих двух процессов.

### **2.3 Моделирующий алгоритм. Имитационная модель.**

Имитационный характер исследования предполагает наличие *логико-или логико-математических моделей*, описываемых изучаемый процесс (систему).

Логико-математическая модель сложной системы может быть как *алгоритмической*, так и *неалгоритмической* (например, система дифференциальных уравнений преобразуется в алгоритмическую с использованием численного метода интегрирования, при этом свойства модели меняются и это надо учитывать).

Чтобы быть машинно-реализуемой, на основе логико-математической модели сложной системы строится *моделирующий алгоритм*, который описывает структуру и логику взаимодействия элементов в системе.

Программная реализация моделирующего алгоритма — есть *имитационная модель*. Она составляется с применением средств автоматизации моделирования. Подробнее технология имитационного моделирования, инструментальные средства моделирования — языки и системы моделирования, с помощью которых реализуются имитационные модели, будут рассмотрены в лекции 5 и практическом курсе.

## **2.4 Проблемы стратегического и тактического планирования имитационного эксперимента. Направленный вычислительный эксперимент на имитационной модели.**

В начале этой лекции мы в общем виде дали понятие *метода имитационного моделирования*. Мы определили имитационное моделирование как *экспериментальный метод* исследования реальной системы по ее имитационной модели.

Понятие метода всегда шире понятия "имитационная модель". Не стоит его путать и с методологией. С методологией имитационного моделирования мы определились еще в 1 лекции, это — системный анализ. Именно последнее дает право, рассматриваемый вид моделирования называть "системным моделированием".

Рассмотрим особенности этого экспериментального метода. Кстати, слова "simulation", "эксперимент", "имитация" одного плана. Экспериментальная природа имитации также предопределила происхождение названия метода. Итак, цель любого исследования состоит в том, чтобы узнать как можно больше об изучаемой системе, собрать и проанализировать информацию, необходимую для принятия решения. Суть исследования реальной системы по ее имитационной модели состоит в получении (сборе) данных о функционировании системы в результате проведения эксперимента на имитационной модели (см. лекцию 1: имитационный метод исследования).

Имитационные модели — это модели прогонного типа, у которых есть вход и выход. То есть, если подать на вход имитационной модели определенные значения параметров (переменных, структурных

взаимосвязей), можно получить результат, который действителен только при этих значениях. На практике исследователь сталкивается со следующей специфической чертой имитационного моделирования. Имитационная модель дает результаты, которые действительны только для определенных значений параметров, переменных и структурных взаимосвязей, заложенных в имитационную программу. Изменение параметра или взаимосвязи означает, что имитационная программа должна быть запущена вновь. Поэтому, для получения необходимой информации или результатов необходимо осуществлять *прогон* имитационных моделей, а не решать их. Имитационная модель не способна формировать свое собственное решение в том виде, как это имеет место в аналитических моделях (см. лекцию 1: расчетный метод исследования), а может служить в качестве средства для анализа поведения системы в условиях, которые определяются экспериментатором.

Для пояснения рассмотрим 2 случая:

- детерминированный случай;
- стохастический случай.

*Стохастический случай.* Имитационная модель – удобный аппарат для исследования стохастических систем. Стохастические системы – это такие системы, динамика которых зависит от случайных факторов, входные, выходные переменные стохастической модели, как правило, описываются как случайные величины, функции, процессы, последовательности. Рассмотрим основные особенности моделирования процессов с учетом действия случайных факторов (здесь реализуются известные идеи метода статистических испытаний, метода Монте-Карло). Результаты моделирования, полученные при воспроизведении единственной реализации процессов, в силу действия случайных факторов будут реализациями случайных процессов, и не смогут объективно характеризовать изучаемый объект. Поэтому искомые величины при исследовании процессов методом имитационного моделирования обычно определяют как средние значения по данным большого числа реализаций процесса (задача оценивания). Поэтому эксперимент на модели содержит несколько реализаций, прогонов и предполагает оценивание по данным совокупности (выборки). Ясно, что (по закону больших чисел), чем больше число реализаций, тем получаемые оценки все больше приобретают статистическую устойчивость.

Итак, в случае со стохастической системой необходимо осуществлять сбор и оценивание статистических данных на выходе имитационной модели, - для этого проводить серию прогонов и статистическую обработку результатов моделирования.

*Детерминированный случай.* В этом случае достаточно провести один прогон, по определенным операционным правилам при конкретном наборе параметров.

Теперь представим, что целями моделирования являются: исследование системы при различных условиях, оценка альтернатив, нахождение зависимости выхода модели от ряда параметров и, наконец, поиск некоторого оптимального варианта. В этих случаях исследователь может проникнуть в особенности функционирования моделируемой системы, изменяя значения параметров на входе модели, при этом выполняя многочисленные машинные прогоны имитационной модели.

Таким образом, проведение экспериментов с моделью на ЭВМ заключается в проведении многократных машинных прогонов с целью сбора, накопления и последующей обработки данных о функционировании системы. Имитационное моделирование позволяет исследовать модель реальной системы, чтобы изучать ее поведение путем многократных прогонов на ЭВМ при различных условиях функционирования реальной системы.

Здесь возникают следующие проблемы: как собрать эти данные, провести серию прогонов, как организовать целенаправленное экспериментальное исследование. Выходных данных, полученных в результате такого экспериментирования, может оказаться очень много. Как их обработать? Обработка и изучение их может превратиться в самостоятельную проблему, намного сложнее задачи статистического оценивания.

В имитационном моделировании важным вопросом является не только проведение, но и планирование имитационного эксперимента в соответствии с поставленной целью исследования.

Таким образом, перед исследователем, использующим методы имитационного моделирования, всегда встает проблема *организации* эксперимента, т.е. выбора метода сбора информации, который дает требуемый (для достижения поставленной цели исследования) ее объем при наименьших затратах (лишнее число прогонов — это лишние затраты машинного времени). Основная цель — уменьшить временные затраты на эксплуатацию модели, сократить машинное время на имитацию, отражающее затраты ресурса времени ЭВМ на проведение большого количества имитационных прогонов.

Эта проблема получила название **стратегического планирования** имитационного исследования. Для ее решения используются методы регрессионного анализа, планирования эксперимента и др., которые подробно будут рассматриваться в лекции 7.

Стратегическое планирование — это разработка эффективного плана эксперимента, в результате которого либо выясняется взаимосвязь между управляемыми переменными, либо находится комбинация значений управляемых переменных, минимизирующая или максимизирующая отклик (выход) имитационной модели.

Наряду с понятием стратегического существует понятие **тактического планирования**, которое связано с определением способов проведения имитационных прогонов, намеченных планом эксперимента: как провести каждый прогон в рамках составленного плана эксперимента. Здесь решаются задачи: определение длительности прогона, оценка точности результатов моделирования и др.

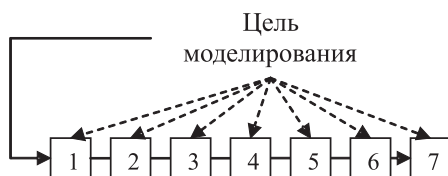
Такие имитационные эксперименты с имитационной моделью будем называть **направленными вычислительными экспериментами**.

**Имитационный эксперимент, содержание которого определяется предварительно проведенным аналитическим исследованием (т.е. являющимся составной частью вычислительного эксперимента) и результаты которого достоверны и математически обоснованы, назовем направленным вычислительным экспериментом.**

В лекции 7 мы детально рассмотрим практические вопросы организации и проведения направленных вычислительных экспериментов на имитационной модели.

## 2.5 Общая технологическая схема имитационного моделирования.

Обобщая наше рассуждение, можно в самом общем виде представить технологическую схему имитационного моделирования (рис. 2.5). В следующих лекциях мы будем более подробно рассматривать технологию имитационного моделирования.



1 – Реальная система; 2 – Построение логико-математической модели; 3 – Разработка моделирующего алгоритма; 4 – Построение имитационной (машинной) модели; 5 – Планирование и проведение имитационных экспериментов; 6 – Обработка и анализ результатов; 7 – Выводы о поведении реальной системы (принятие решений).

*Рисунок 2.5 – Технологическая схема имитационного моделирования.*

## 2.6 Возможности, область применения имитационного моделирования.

Рассмотрим возможности метода имитационного моделирования, обусловившие его широкое применение в самых различных сферах. Имитационное моделирование традиционно находит применение в

экономических исследованиях: моделировании производственных систем и логистических процессов [14,53], маркетинге, моделировании бизнес процессов; в социально-экономических исследованиях: моделировании экономических реформ, региональных процессов [15,16,17,23,28,34,51], социологии [46] и политологии; моделировании транспортных, информационных [33] и телекоммуникационных систем, наконец, глобальном моделировании мировых процессов [52].

Метод имитационного моделирования позволяет решать задачи исключительной сложности, обеспечивает имитацию любых сложных и многообразных процессов, с большим количеством элементов, отдельные функциональные зависимости в таких моделях могут описываться весьма громоздкими математическими соотношениями. Поэтому имитационное моделирование эффективно используется в задачах исследования систем со сложной структурой с целью решения конкретных проблем.

Имитационная модель содержит элементы непрерывного и дискретного действия, поэтому применяется для исследования динамических систем, когда требуется *анализ узких мест*, исследование *динамики функционирования*, когда желательно наблюдать на имитационной модели ход процесса в течение определенного времени

Имитационное моделирование — эффективный аппарат исследования *стохастических систем*, когда исследуемая система может быть подвержена влиянию многочисленных случайных факторов сложной природы (у математических моделей для этого класса систем ограниченные возможности). Имеется возможность проводить исследование *в условиях неопределенности*, при неполных и неточных данных.

Имитационное моделирование является наиболее ценным, системообразующим звеном в *системах поддержки принятия решений*, т.к. позволяет исследовать большое число альтернатив (вариантов решений), проигрывать различные сценарии при любых входных данных. Главное преимущество имитационного моделирования состоит в том, что исследователь для проверки новых стратегий и принятия решений, при изучении возможных ситуаций, всегда может получить ответ на вопрос *"Что будет, если? ..."*. Имитационная модель позволяет прогнозировать, когда речь идет о проектируемой системе или исследуются процессы развития (т.е. в тех случаях, когда реальной системы не существует).

В имитационной модели может быть обеспечен различный (в том числе и очень высокий) *уровень детализации* моделируемых процессов. При этом модель создается поэтапно, постепенно, без существенных изменений, *эволюционно*.

## ЛЕКЦИЯ

## 3

**ТЕХНОЛОГИЧЕСКИЕ ЭТАПЫ  
СОЗДАНИЯ И  
ИСПОЛЬЗОВАНИЯ  
ИМИТАЦИОННЫХ МОДЕЛЕЙ.****3.1 Основные этапы имитационного моделирования.  
Общая технологическая схема.**

Вне зависимости от типа моделей (непрерывные и дискретные, детерминированные и стохастические и т.д.) имитационное моделирование включает в себя ряд основных этапов, представленных на рисунке 3.1 и является сложным итеративным процессом:

1. **Формулировка проблемы и определение целей имитационного исследования** Документированным результатом на этом этапе является составленное *содержательное описание объекта моделирования*;
2. **Разработка концептуального описания.** Результатом деятельности системного аналитика является *концептуальная модель* (или вербальное описание) и *выбор способа формализации* для заданного объекта моделирования.
3. **Формализация имитационной модели.** Составляется *формальное описание* объекта моделирования.
4. **Программирование имитационной модели (разработка программы-имитатора).** На этапе осуществляется выбор средств автоматизации моделирования, алгоритмизация, программирование и отладка имитационной модели.
5. **Испытание и исследование модели, проверка модели.** Проводится верификация модели, оценка адекватности, исследование свойств имитационной модели и другие *процедуры комплексного тестирования* разработанной модели.
6. **Планирование и проведение имитационного эксперимента.** На данном технологическом этапе осуществляется стратегическое и тактическое планирование имитационного эксперимента. Результатом является составленный и реализованный *план эксперимента*, заданные условия имитационного прогона для выбранного плана.
7. **Анализ результатов моделирования.** Исследователь проводит интерпретацию результатов моделирования и их использование – собственно принятие решений.

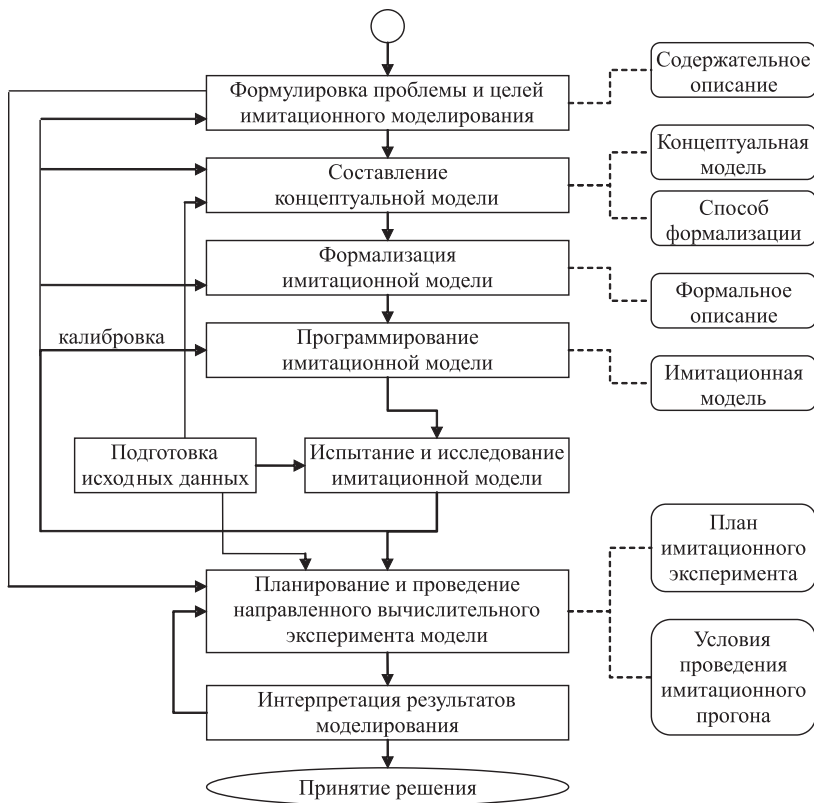


Рисунок 3.1 – Технологические этапы имитационного моделирования.

### 3.2 Формулировка проблемы и определение целей имитационного исследования.

На первом этапе формулируется проблема, стоящая перед исследователем и принимается решение о целесообразности применения метода имитационного моделирования. Затем определяются цели, которые должны быть достигнуты в результате имитации. От формулировки целей в значительной мере зависит выбор типа имитационной модели и характер дальнейшего имитационного исследования на имитационной модели.

На этом этапе определяется и детально изучается объект моделирования, те стороны его функционирования, которые представляют интерес для исследования. Результатом работ на данном этапе является *содержательное описание объекта моделирования с*



указанием целей имитации и тех аспектов функционирования объекта моделирования, которые необходимо изучить на имитационной модели. Содержательное описание составляется в терминологии реальной системы, на языке предметной области, понятном заказчику.

В ходе составления содержательного описания объекта моделирования устанавливаются границы изучения моделируемого объекта, дается описание внешней среды, с которой он взаимодействует. Формулируются также основные критерии эффективности, по которым предполагается проводить сравнение на модели различных вариантов решений, проводится генерация и описание рассматриваемых альтернатив.

Общего рецепта составления содержательного описания не существует. Успех зависит от интуиции разработчика и знания реальной системы.

Общая технология или последовательность действий на этом этапе следующая: сбор данных об объекте моделирования и составление *содержательного описания объекта моделирования*; далее следует: изучение проблемной ситуации – определение диагноза и постановка задачи; уточнение целей моделирования; обосновывается необходимость моделирования и осуществляется выбор метода моделирования. На этом этапе четко, конкретно формулируются *цели моделирования*. Цели моделирования определяют общий замысел модели и пронизывают все последующие этапы имитационного моделирования; далее осуществляется формирование концептуальной модели исследуемого объекта.

Подробнее остановимся на основном содержании деятельности системного аналитика на этих ранних этапах. Эта работа исключительно важна для всех последующих этапов имитационного моделирования, именно здесь специалист по имитационному моделированию демонстрирует себя как системный аналитик, владеющий *искусством моделирования*. Процесс, при помощи которого инженер, занимающийся системами, или ученый, исследующий вопросы управления, создает модель изучаемой системы, может быть лучше всего определен как интуитивное искусство. Искусством моделирования могут овладеть те, кто обладает оригинальным мышлением, изобретательностью, ровно как и глубокими знаниями систем, которые необходимо моделировать. При построении моделей мы должны полагаться на искусство, опыт, интуицию. Современный теоретический аппарат описания систем не может нам гарантировать что мы выполняем оптимальное для наших целей моделирование. Изучение конкретных образцов моделей не способствует развитию творческого подхода к созданию моделей.

*Структурирование исходной проблемы. Формулирование проблемы.* Прежде всего, системный аналитик должен уметь анализировать проблему. Он выполняет изучение и структурирование исходной проблемы, четкое формулирование проблемы.

Анализ проблемы необходимо начинать с детального изучения всех аспектов функционирования (здесь важно понимание деталей — поэтому надо быть специалистом в конкретной предметной области или тесно общаться с экспертами). Рассматриваемая система связана с другими системами, поэтому с позиций системного подхода надо раскрутить клубок проблем: от исходной формулировки — расширить до проблематики. Важно правильно поставить задачи. Общая задача моделирования при этом разбивается на частные.

Основное смысловое содержание системного подхода к решению проблем демонстрируется на рисунке 3.2.

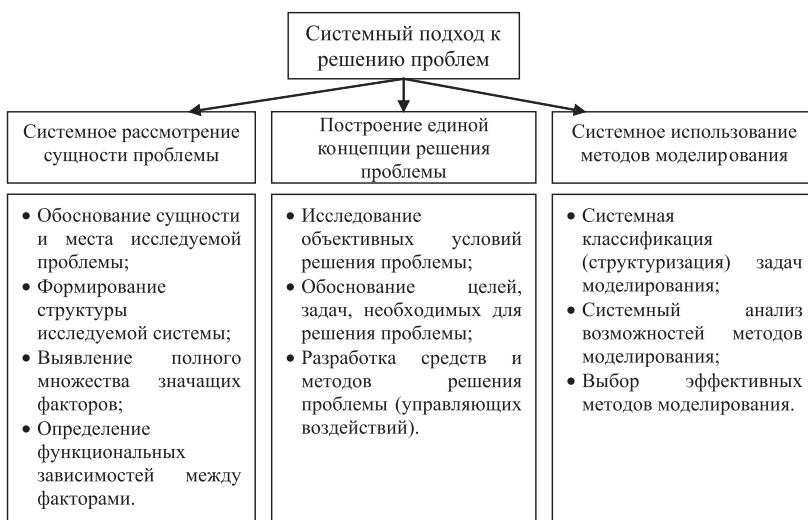


Рисунок 3.2 – Содержание системного подхода к решению проблем.

Системный подход к решению проблем предполагает:

- **Системное рассмотрение сущности проблемы:**
  - Обоснование сущности и места исследуемой проблемы;
  - Формирование общей структуры исследуемой системы;
  - Выявление полного множества значащих факторов;
  - Определение функциональных зависимостей между факторами;
- **Построение единой концепции решения проблемы:**
  - Исследование объективных условий решения проблемы;
  - Обоснование целей, задач, необходимых для решения проблемы – *структуризация задач, формализация целей*;
  - Разработка средств и методов решения проблемы: описание альтернатив, сценариев, решающих правил и управляющих

воздействий для отработки в дальнейшем на модели процедур принятия решений;

- *Системное использование методов моделирования:*
  - Системная классификация (структуризация) задач моделирования;
  - Системный анализ возможностей методов моделирования;
  - Выбор эффективных методов моделирования.

*Выявление целей.* Первый и самый важный шаг при создании любой модели состоит в определении ее целевого назначения. Цели — антиподы проблемы. Может быть применен метод декомпозиции целей, предполагающий разделение целого на части: целей — на подцели, задач — на подзадачи...и т.д. — на практике этот подход приводит к иерархическим древовидным структурам (построению дерева целей). Вся эта процедура является уделом экспертов по проблеме, специалистов. Поэтому здесь присутствует субъективный фактор, каждый эксперт сделает по-своему. Практическая задача состоит в том, насколько полно все структурировано. (Построенное в результате этой процедуры дерево целей может в дальнейшем оказаться полезным при формировании множества критериев).

Какие подводные камни поджидают начинающего системного аналитика? То, что для одного уровня цель — для другого средство, часто их путают, происходит смешение целей. Цель — есть описание желаемого будущего, поэтому здесь легко ошибиться. Для сложной системы с большим количеством подсистем цели могут быть противоречивыми. Цель редко бывает единственной: существует опасность неверного ранжирования при множестве целей.

Сформулированные и структурированные на первом этапе цели моделирования пронизывают весь ход дальнейшего имитационного исследования.

Рассмотрим наиболее употребимые *категории целей* в имитационном исследовании: [54] *оценка, прогнозирование, оптимизация, сравнение альтернатив* и др.

Эксперименты по моделированию проводятся с весьма разнообразными целями, в числе которых могут быть:

- *Оценка* — определение, насколько хорошо система предлагаемой структуры будет соответствовать некоторым конкретным критериям,
- *Сравнение альтернатив* — сопоставление конкурирующих систем, рассчитанных на выполнение определенной функции, или же на сопоставление нескольких предлагаемых рабочих принципов или методик,
- *Прогноз* — оценка поведения системы при некотором предполагаемом сочетании рабочих условий,

- *Анализ чувствительности* — выявление из большого числа действующих факторов тех, которые в наибольшей степени влияют на общее поведение системы,
- *Выявление функциональных соотношений* — определение природы зависимости между двумя или несколькими действующими факторами, с одной стороны, и откликом системы с другой,
- *Оптимизация* — точное определение такого сочетания действующих факторов и их величин, при котором обеспечивается наилучший отклик всей системы в целом.

*Формирование критериев.* Исключительно важно четкое и однозначное определение критериев. Это влияет на процесс создания и экспериментирования модели, кроме того неправильное определение критерия ведет к неправильным выводам. Различают критерии, с помощью которых оценивается степень достижения цели системой, и критерии по которым оценивается способ движения к цели (или эффективность средства достижения целей). Для многокритериальных моделируемых систем формируется набор критериев, их необходимо структурировать по подсистемам или ранжировать по важности.

### 3.3 Разработка концептуальной модели объекта моделирования.

*Концептуальная модель* — есть логико-математическое описание моделируемой системы в соответствии с формулировкой проблемы. Основное содержание этого этапа — формулировка общего замысла модели, переход от реальной системы к логической схеме ее функционирования (схематично общее содержание этого технологического перехода демонстрируется на рисунке 3.3). Здесь приводится описание объекта в терминах математических понятий и алгоритмизация функционирования ее компонент. Концептуальное описание представляет собой упрощенное алгоритмическое отображение реальной системы.

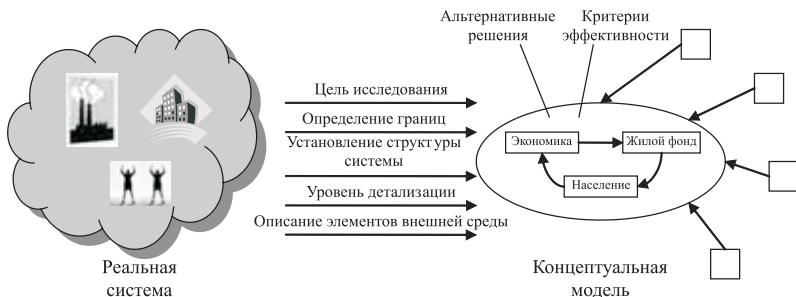


Рисунок 3.3 — Переход от реальной системы к логической схеме её функционирования.

При разработке концептуальной модели осуществляется установление *основной структуры модели*, которое включает *статическое и динамическое описание системы*. Определяются границы системы, приводится описание внешней среды, выделяются существенные элементы и дается их описание, формируются переменные, параметры, функциональные зависимости как для отдельных элементов и процессов, так и для всей системы, ограничения, целевые функции (критерии).

Результат работы на этом этапе — документированное концептуальное описание плюс выбранный способ формализации моделируемой системы. При создании небольших моделей этап совмещается с этапом составления содержательного описания моделируемой системы. На этом этапе уточняется также методика всего имитационного эксперимента.

*Построение концептуальной модели* начинается с того, что на основе цели моделирования устанавливаются границы моделируемой системы, определяются воздействия внешней среды. Выдвигаются гипотезы и фиксируются все допущения (предположения), необходимые для построения имитационной модели. Обсуждается уровень детализации моделируемых процессов.

Система есть совокупность взаимосвязанных элементов. Определение системы всегда субъективно, зависит от цели моделирования, и от того, кто именно определяет систему. На этом этапе осуществляется *декомпозиция системы*. Определяются наиболее существенные в смысле сформулированной проблемы элементы системы (выполняется *структурный анализ* моделируемой системы) и взаимодействия между ними, выявляются основные аспекты функционирования моделируемой систем (составляется *функциональная модель*), приводится описание внешней среды. Декомпозиция системы (объекта моделирования) или выделение подсистем — есть операция *анализа*. Элементы модели должны соответствовать реально- существующим фрагментам в системе. Сложная система разбивается на части, сохраняя при этом связи, обеспечивающие взаимодействие. Можно составить функциональную схему, которая прояснит специфику динамических процессов, происходящих в рассматриваемой системе. Важно определить, какие компоненты будут включены в модель, какие будут вынесены во внешнюю среду, и какие взаимосвязи будут установлены между ними.

*Описание внешней среды* выполняется по тем соображениям, что элементы внешней среды оказывают определенное влияние на элементы системы, однако влияние самой системы на них, как правило, незначительно.

При обсуждении уровня детализации модели важно понимать, что в основании всякой декомпозиции лежат два противоречивых принципа: *полнота и простота*. Обычно на начальных этапах составления модели

наблюдается тенденция к учету чрезмерно большого числа компонентов и переменных. Однако хорошая модель — простая, ведь известно, что степень понимания явления обратно пропорциональна числу переменных, фигурирующих в его описании. Модель, перегруженная деталями, может стать сложной и трудно реализуемой.

Компромисс между этими двумя полюсами, состоит в том, что в модель включаются только *существенные* (или *релевантные*) компоненты, — существенные по отношению к цели анализа. Выбор в этом тонком деле опять за экспертом.

Итак, сначала должна присутствовать “элементарность” — составляет самое простое дерево целей, упрощенная структура модели. Далее осуществляется постепенная детализация моделей. Надо стремиться делать простые модели, затем их усложнять. Необходимо следовать *принципу итеративного построения модели*, когда по мере изучения системы по модели, в ходе разработки, модель изменяется путем добавления новых или исключения некоторых ее элементов и/или взаимосвязей между ними.

Как же все-таки перейти от реальной системы к ее упрощенному описанию? *Упрощение, абстракция* — основные приемы любого моделирования. Выбранный уровень детализации должен позволять абстрагироваться от неточно определенных из-за недостатка информации аспектов функционирования реальной системы.

Под *упрощением* понимается пренебрежение несущественными деталями или принятие предположений о более простых соотношениях (например, предположение о линейной зависимости между переменными). При моделировании выдвигаются гипотезы, предположения, относящиеся к взаимосвязи между компонентами и переменными системы.

Другим аспектом анализа реальной системы является абстракция. *Абстракция* содержит или сосредоточивает в себе существенные качества поведения объекта, но не обязательно в той же форме и столь детально, как это имеет место в реальной системе.

После того как проанализированы и промоделированы части или элементы системы, мы приступаем к их объединению в единое целое. В концептуальной модели должно быть корректно отражено их взаимодействие. *Композиция* есть операция *синтеза*, агрегирование (при системном моделировании это не просто сборка компонентов). В ходе этой операции выполняется установление отношений между элементами (например, уточняется структура, приводится описание отношений, упорядочение и др.)

Системное исследование построено на сочетание операций анализа и синтеза. На практике реализуются итеративные процедуры анализа и синтеза. Лишь после этого мы можем пытаться объяснить целое —

систему, через его составляющие- подсистемы, в виде общей структуры целого.

*Критерии эффективности. Параметры, переменные модели.* В описание системы должны быть включены критерии эффективности функционирования системы и оцениваемые альтернативные решения, последние могут рассматриваться как входы модели или сценарные параметры. При алгоритмизации моделируемых процессов уточняются также основные переменные модели, участвующие в ее описании.

Каждая модель представляет собой некоторую комбинацию таких составляющих, как *компоненты, переменные, параметры, функциональные зависимости, ограничения, целевые функции (критерии)*. Приведем некоторые полезные в дальнейшем определения согласно [54].

Под *компонентами* понимают составные части, которые при соответствующем объединении образуют систему. Иногда компонентами считают также *элементы* системы или ее *подсистемы*. Система определяется как группа или совокупность объектов, объединенных некоторой формой регулярного взаимодействия или взаимозависимости для выполнения заданной функции. Изучаемая система состоит из компонент.

Параметрами являются величины, которые исследователь может выбирать произвольно, в отличие от *переменных* модели, которые могут принимать только значения, определяемые видом данной функции. В модели будем различать переменные двух видов: *экзогенные* и *эндогенные*. Экзогенные переменные называются также *входными*. Это означает, что они порождаются вне системы или являются результатом взаимодействия внешних причин. Эндогенными переменными называются переменные, возникающие в системе в результате воздействия внутренних причин. В тех случаях, когда эндогенные переменные характеризуют состояние или условия, имеющие место в системе, назовем их *переменными состояниями*. Когда же необходимо описать входы и выходы системы, мы имеем дело с *входными* и *выходными переменными*.

*Функциональные зависимости* описывают поведение переменных и параметров в пределах компоненты или же выражают соотношения между компонентами системы. Эти соотношения по природе являются либо детерминистскими, либо стохастическими.

*Ограничения* представляют собой устанавливаемые пределы изменения значений переменных или ограничивающие условия их изменений. Они могут вводиться либо разработчиком, либо устанавливаться самой системой вследствие присущих ей свойств.

*Целевая функция (функция критерия)* представляет собой точное отображение целей или задач системы и необходимых правил оценки их выполнения. Выражение для целевой функции должно быть однозначным

определением целей и задач, с которыми должны соизмеряться принимаемые решения.

### **3.4 Формализация имитационной модели.**

На третьем этапе имитационного исследования осуществляется формализация объекта моделирования. Процесс формализации сложной системы включает:

- выбор способа формализации;
- и составление формального описания системы.

В процессе построения модели можно выделить 3 уровня ее представления:

- неформализованный (этап 2) — *концептуальная модель*;
- формализованный (этап 3) — *формальная модель*;
- программный (этап 4) — *имитационная модель*.

Каждый уровень отличается от предыдущего степенью детализации моделируемой системы и способами описания ее структуры и процесса функционирования. При этом уровень абстрагирования возрастает.

*Концептуальная модель* — это систематизированное содержательное описание моделируемой системы (или проблемной ситуации) на неформальном языке. Неформализованное описание разрабатываемой имитационной модели включает определение основных элементов моделируемой системы, их характеристики и взаимодействия между элементами на собственном языке, могут использоваться таблицы, графики, диаграммы и т.д. Неформализованное описание модели необходимо как самим разработчикам (при проверке адекватности модели, ее модификации и т.д.), так и для взаимопонимания со специалистами других профилей.

Концептуальная модель содержит исходную информацию для системного аналитика, выполняющего формализацию системы и использующего для этого определенную методологию и технологию, т.е. на основании неформализованного описания осуществляется разработка более строгого и подробного формализованного описания.

Далее формализованное описание преобразуется в программ-имитатор также в соответствии с некоторой методикой (технологией программирования).

Аналогичная схема, кстати, имеет место и при выполнении имитационных экспериментов: — содержательная постановка отображается на формальную модель, после чего вносятся необходимые изменения и дополнения в методику направленного вычислительного эксперимента.



Основная задача этапа формализации — дать формальное описание сложной системы, свободное от второстепенной информации, имеющейся в содержательном описании, *алгоритмическое представление объекта моделирования*. Цель формализации — получить формальное представление логико-математической модели, т.е. алгоритмов поведения компонент сложной системы и отразить на уровне моделирующего алгоритма вопросы взаимодействия между собой этих компонент.

Может оказаться, что информации, имеющейся в содержательном описании недостаточно для формализации объекта моделирования. В этом случае — необходимо вернуться к этапу составления содержательного описания и дополнить его данными, необходимость в которых обнаруживается при формализации объекта моделирования. На практике таких возвратов может быть несколько.

Формализация полезна в определенных пределах и для простых моделей неоправданна.

Наблюдается существенное разнообразие схем (концепций) формализации и структуризации, нашедших применение в имитационном моделировании. Схемы формализации ориентируются на различные математические теории и исходят из разных представлений об изучаемых процессах — отсюда их многообразие — отсюда проблема выбора подходящей (для описания данного объекта моделирования) схемы формализации.

Для дискретных моделей, например, могут применяться процессно-ориентированные системы (process description), системы, основанные на сетевых парадигмах (network paradigms), для непрерывных — потоковые диаграммы моделей системной динамики.

Наиболее известные и широко используемые на практике концепции формализации: агрегативные системы и автоматы; сети Петри и их расширения; модели системной динамики, которые подробно мы будем изучать в лекции 4.

В рамках одной концепции формализации могут быть реализованы разнообразные алгоритмические модели. Как правило, та или иная концепция структуризации (схема представления алгоритмических моделей) или формализации на технологическом уровне закреплена в системе моделирования, языке моделирования. Концепция структуризации более или менее явно лежит в основе всех имитационных систем и поддерживается специально разработанными приемами технологии программирования. Это упрощает построение и программирование модели. Например, язык моделирования GPSS, имеет блочную концепцию структуризации, структура моделируемого процесса изображается в виде потока транзактов, проходящего через

обслуживающие устройства, очереди и другие элементы систем массового обслуживания.

В ряде современных систем моделирования, наряду с аппаратом, поддерживающим ту или иную концепцию структуризации, имеются специальные средства, обеспечивающие применение в системе определенной концепции формализации.

В основе построения имитационных моделей лежат современные методы структуризации сложных систем и описания их динамики. Широко используются в практике анализа сложных систем следующие модели и методы:

- сети кусочно-линейных агрегатов, моделирующие дискретные и непрерывно-дискретные системы;
- сети Петри (сети событий, E-сети, КОМБИ-сети и др. расширения), применяемые при структуризации причинных связей и моделировании систем с параллельными процессами, служащие для стратификации и алгоритмизации динамики дискретных и дискретно-непрерывных систем;
- потоковые диаграммы и конечно-разностные уравнения системной динамики, являющиеся моделями непрерывных систем;
- и другие.

В лекции 4 мы подробно рассмотрим некоторые полезные концепции формализации.

### **3.5 Программирование имитационной модели.**

Концептуальное или формальное описание модели сложной системы преобразуется в программу-имитатор в соответствии с некоторой методикой (дисциплиной) программирования, с применением языков и систем моделирования. Важным моментом здесь является корректный выбор инструментального средства для реализации имитационной модели. Технологическое содержание этого этапа подробно рассматривается в практическом курсе дисциплины. Назначение и классификации языков и систем моделирования, инструментальные и технологические возможности современных систем моделирования рассматривается в 5 лекции.

### **3.6 Сбор и анализ исходных данных.**

Не всегда этот этап выделяется как самостоятельный, однако выполняемая на этом этапе работа исключительно важна, трудоемка. Если программирование и трассировку имитационной модели можно выполнять на гипотетических данных, то предстоящее экспериментальное

исследование необходимо выполнять на реальном потоке данных. От этого зависит во многом точность получаемых результатов моделирования.

Здесь перед разработчиком имитационной модели встанут два вопроса:

- где и каким образом получить, собрать исходную информацию;
- как обработать собранные данные о реальной системе.

*Основные методы получения исходных данных:*

- из существующей документации на систему (это могут быть данные официальных и других отчетов, статистические сборники, — например, для социально-экономических систем, финансовая и техническая документация — для производственных систем и др.);
- физическое экспериментирование. Часто для задания исходной информации необходимо провести натурные эксперимент на моделируемой системе или ее прототипах (порой это бывают дорогостоящие эксперименты, — однако это плата за то, чтобы получить точную модель, на которой можно в дальнейшем проводить испытания). Такой подход применим для космических, военных исследований, в авиации. В более простых случаях можно проводить измерения, например хронометраж при выполнении производственных операций;
- предварительный, априорный синтез данных. Иногда исходные данные могут не существовать, и сама природа моделируемой системы исключает возможность физического экспериментирования (например, проектируемые системы, прогнозирование в социальных и политических исследованиях). В этом случае предлагают различные приемы предварительного синтеза данных. Например, при моделировании информационных систем, продолжительность выполнения информационного требования оценивается на основании трудоемкости реализуемых на ЭВМ алгоритмов. К этим методам относят различные процедуры, основанные на общем анализе проблематики, анкетировании, интервьюировании, широком применении методов экспертного оценивания.

Второй вопрос связан с проблемой *идентификации входных данных* для стохастических систем. Мы уже говорили о том, что имитационное моделирование является эффективным аппаратом исследования стохастических систем, т.е. таких систем, динамика которых зависит от случайных факторов. Входные (и выходные) переменные стохастической модели, как правило, — случайные величины, векторы, функции, случайные процессы. Поэтому возникают дополнительные трудности, связанные с синтезом уравнений относительно неизвестных законов распределения и определением вероятностных характеристик (средних

значений, дисперсий, корреляционных функций и т.п.) для анализируемых процессов и их параметров. Необходимость статистического анализа при сборе и анализе входных данных связана с задачами определения вида функциональных зависимостей, описывающих входные данные, оценкой конкретных значений параметров этих зависимостей, а также проверкой значимости параметров. Для подбора теоретических распределений случайных величин применяют известные методы математической статистики, основанные на определении параметров эмпирических распределений и проверке статистических гипотез, с использованием критериев согласия, о том, согласуются ли имеющиеся эмпирические данные с известными законами распределения (на статистически приемлемом доверительном уровне). Конечно, на вход модели можно подавать и сырые эмпирические данные, однако это неэффективно как с точки зрения программной реализации, так и с точки зрения моделирования, руководствуясь желанием получить более общие и полезные результаты на выходе имитационной модели.

### **3.7 Испытание и исследование свойств имитационной модели.**

После того, как имитационная модель реализована на ЭВМ, необходимо провести испытание, проверку достоверности модели. Это является чрезвычайно важным и ответственным моментом в имитационном моделировании. Проверка, выполненная не тщательно, может привести к неизвестным последствиям.

На этапе испытания и исследования разработанной имитационной модели организуется комплексное **тестирование модели (testing)** – **планируемый итеративный процесс, направленный главным образом на поддержку процедур верификации и валидации имитационных моделей и данных.**

Некоторые полезные процедуры тестирования можно найти в лекции 6 настоящего пособия и специальной литературе [20, 26, 33, 56].

Если в результате проведенных процедур модель окажется недостаточно достоверной, может быть выполнена *калибровка имитационной модели* (в моделирующий алгоритм встраиваются калибровочные коэффициенты) с целью обеспечения адекватности модели, или в более сложных случаях возможны многочисленные итерации на ранние этапы с целью получения дополнительной информации о моделируемом объекте или доработки имитационной модели. Наличие ошибок во взаимодействии компонент модели возвращает исследователя к этапу создания имитационной модели. Возможно, в ходе формализации исследователь слишком упростил физические явления, исключил из рассмотрения ряд важных сторон

функционирования системы, что привело к неадекватности модели объекту. В этом случае исследователь должен вернуться к этапу формализации системы. В тех случаях, когда выбор способа формализации оказался неудачным, исследователю необходимо повторить этап составления концептуальной модели с учетом новой информации и появившегося опыта. Наконец, когда у исследователя оказалось недостаточно информации об объекте, он должен вернуться к этапу составления содержательного описания системы и уточнить его с учетом результатов испытания.

### **3.8 Направленный вычислительный эксперимент на имитационной модели. Анализ результатов моделирования и принятие решений.**

В лекции 2 мы уделили внимание рассмотрению экспериментальной природы имитации, дали общее определение направленного вычислительного эксперимента. Вы уже знаете, что на заключительных этапах имитационного моделирования необходимо проводить стратегическое и тактическое планирование имитационного эксперимента. Организация направленного вычислительного эксперимента на имитационной модели предполагает выбор и применение различных аналитических методов для обработки результатов имитационного исследования. Наиболее употребимыми для целей вычислительного эксперимента аналитическими методами являются методы планирования вычислительного эксперимента, регрессионный и дисперсионный анализ, градиентные и другие методы оптимизации, о применении которых речь пойдет в 7 лекции. Очевидно, что организация и проведение эксперимента потребует от Вас серьезной аналитической подготовки и корректного применения аналитических методов. Проведенное исследование должно позволить Вам сделать некоторые выводы по полученным результатам, достаточные для принятия решений по обозначенным на ранних этапах проблемам и задачам.

Теперь Вы представляете, что процесс имитационного исследования является трудоемким итеративным процессом, требующим от разработчиков таких моделей определенных интеллектуальных затрат и технологических усилий. Разработчик имитационной модели должен быть и искусным системным аналитиком, и мастером-технологом, владеющим современными компьютерными технологиями, применяемыми при создании и исследовании имитационных моделей, обладать хорошей аналитической подготовкой, позволяющей корректно применять методы математической статистики и другие математические и вычислительные процедуры, как для целей идентификации имитационных моделей, так и обработки результатов экспериментального исследования.

## ЛЕКЦИЯ

## 4

## БАЗОВЫЕ КОНЦЕПЦИИ СТРУКТУРИЗАЦИИ И ФОРМАЛИЗАЦИИ ИМИТАЦИОННЫХ СИСТЕМ.

В этой лекции рассмотрим ряд моделей и методов, широко используемых в практике анализа сложных систем:

транзактно-ориентированный подход языка моделирования дискретного типа GPSS;

сети кусочно-линейных агрегатов, моделирующие дискретные и непрерывно-дискретные системы;

сети Петри и их расширения, применяемые при структуризации причинных связей и моделировании систем с параллельными процессами, служащие для стратификации и алгоритмизации динамики дискретных и дискретно-непрерывных систем;

поточковые диаграммы и конечно-разностные уравнения системной динамики, являющиеся моделями непрерывных систем.

### 4.1 Методологические подходы к построению дискретных имитационных моделей.

В дискретных имитационных системах изменение состава и состояния происходит в дискретные моменты времени, называемые *событиями*. Под *событием* понимается мгновенное изменение состояния модели, произошедшее в результате осуществления множества взаимодействий между компонентами модели в один и тот же момент имитационного времени.

Как можно описать функционирование дискретной имитационной модели? Взаимосвязь между понятиями поясним с помощью рисунка 4.1.

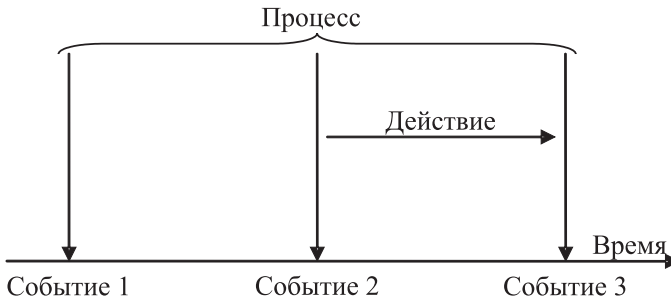


Рисунок 4.1 – Взаимосвязь между событиями, действиями и процессами.

Функционирование дискретной системы можно описать:

- определяя изменения состояния системы, происходящие в моменты свершения событий;
- описывая действия, в которых принимают участие элементы системы;
- описывая процесс, через который проходят элементы.

*Процесс* — это ориентированная во времени последовательность событий, которая может состоять из нескольких действий.

Эти представления лежат в основе трех альтернативных методологических подходов к построению дискретных имитационных моделей, называемых обычно:

- событийный;
- подход сканирования активностей (на практике получил небольшое распространение);
- процессно-ориентированный подход (включает транзактный способ имитации).

Это основные *концепции (схемы) структуризации* для дискретных имитационных моделей. Их основа закладывается в некоторые языки и системы моделирования. Примерами могут служить языки моделирования:

- GASP, SIMSCRIPT, ориентированные на события;
- язык работ SLAM;
- широко распространенные языки моделирования GPSS, SIMULA и др., предназначенные для описания параллельных процессов.

## **4.2 Язык моделирования GPSS.**

### **4.2.1 40 лет в мире информационных технологий.**

В 1961 г. Джеффи Гордон разработал язык моделирования GPSS (General Purpose Simulating System — моделирующая система общего назначения). Недавно в мире имитационного моделирования отмечалось 40-летия языка GPSS ([www.gpss.ru](http://www.gpss.ru)), методическое значение которого велико. Язык GPSS был языком, который определил современные технологические тенденции в дискретном имитационном моделировании и явился предвестником современных языков и систем моделирования дискретного типа, т.к. Extend, Arena, Process Model, Taylor, WITNESS и сотен других современных коммерческих симуляторов. Эти тенденции предопределила, прежде всего, удачно сформированная базовая схема структуризации, заложенная в GPSS, поддерживающая *блочно — ориентированный подход*, в рамках которого моделирующий блок имеет свое *функциональное назначение* и представлен соответствующими функциональными объектами (имеющими аналоги с элементами систем

массового обслуживания), а также возможности языка для описания *параллельных процессов*. Именно такой взгляд на моделируемый объект позволил реализовать идеографический режим формирования дискретной модели, когда модель конструируется из стандартных функциональных блоков, а связи на этих графических конструкциях интерпретируются как маршруты прохождения подвижных объектов в системе. Поэтому, осваивая содержание базовой концепции структуризации языка моделирования GPSS, легко понять идею и принципы работы современных коммерческих симуляторов дискретного типа и других новых интеллектуальных сред.

В настоящее время на рынке информационных технологий представлены 3 направления, поддерживающие технологическое развитие базового языка GPSS: корпорация Wolverin – GPSS/H и современное ее решение язык SLX, корпорация Minuteman Software – GPSS World, и решения Стокгольмской школы высшей экономики – Micro GPSS, Web GPSS.

#### **4.2.2 Содержание базовой концепции структуризации языка моделирования GPSS.**

В языке GPSS реализована *блочно-ориентированная концепция структуризации моделируемого процесса, разработанная с ориентацией на описание систем массового обслуживания (СМО)*.

*Структура моделируемого процесса изображается в виде потока, проходящего через обслуживающие устройства (ОУ), очереди, ключи и другие элементы СМО.*

*Модель имеет блочную структуру. Моделируемый процесс представляется как поток заявок в системе обслуживания. Блоки интерпретируются как ОУ. Заявки (транзакты) конкурируют между собой за место в ОУ, образуют очереди перед ОУ, если они заняты. Дуги на блок-схеме – потенциальные потоки заявок между ОУ. Существуют истоки и стоки этих заявок. В этом случае блок-схема модели описывает маршруты движения заявок в системе.*

Следовательно, в рамках GPSS есть специальные средства, которые являются аналогами элементов систем массового обслуживания, т.к. обслуживающие устройства, заявки, очереди.

Однако, GPSS является гибкой языковой средой, поэтому позволяет моделировать не только СМО, но и другие системы (например, склад, распределение ресурсов и многие другие).

#### **4.2.3 Системы массового обслуживания.**

*Система массового обслуживания – абстрактный объект, в котором выполняется последовательность операций, включает совокупность приборов обслуживания, которые связаны определенным логическим порядком. В соответствии с этой логикой происходит движение*



материальных носителей — заявок на обслуживание от канала (ОУ) к каналу (ОУ).

Структура систем массового обслуживания представлена на рисунке 4.2.1.

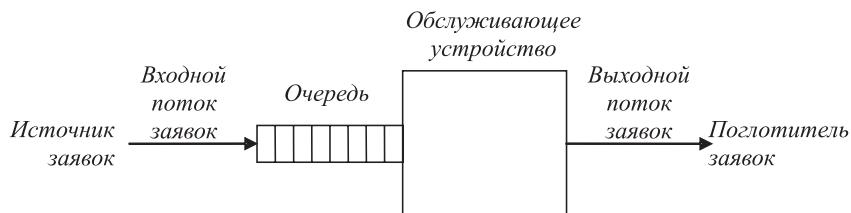


Рисунок 4.2.1 – Структура систем массового обслуживания.

Заявка характеризуется моментом появления на входе системы, статусом по отношению к другим заявкам, некоторыми параметрами, определяющими потребности во временных ресурсах на обслуживание

Постоянно поступающие заявки на обслуживание образуют *поток заявок* – совокупность заявок, распределенную во времени.

Поток заявок может быть *однородным* (с точки зрения обслуживания все заявки равноправны) и *неоднородным*.

Основной параметр потока заявок – промежуток времени между моментами поступления 2-х соседних заявок.

Поток заявок может быть *стационарным* и *нестационарным* (например, изменяться от времени суток).

Поток заявок рассматривается как *случайный процесс*, характеризующийся функцией распределения периода поступления заявок (например, простейший поток, поток Эрланга).

*Элемент системы, в котором происходят операции, называется обслуживающим устройством.* В момент выполнения операций он занят, иначе – свободен. Если ОУ (канал) свободен, то заявка принимается к обслуживанию.

Обслуживание каждой заявки каналом означает задержку в нем заявки на время, равное периоду обслуживания. После обслуживания заявка покидает прибор обслуживания. Таким образом, *ОУ характеризуется временем обслуживания заявки.*

При случайном характере поступления заявок образуются *очереди*.

Заявки принимаются к обслуживанию

- в порядке очереди (FIFO, очереди с приоритетами и др.),
- в случайном порядке в соответствии с заданными распределениями, по минимальному времени получения отказа,

- и др.

Реальный процесс функционирования СМО следует представлять в виде последовательности фаз обслуживания, выполняемых различными устройствами. Примеры многофазного обслуживания: обслуживание покупателей в магазине (прилавок, касса); производственно-технологический процесс (обработка деталей на станках). Причем эти *многофазные системы* могут иметь сложную структуру (стохастические сети), как показано на рисунке 4.2.2.

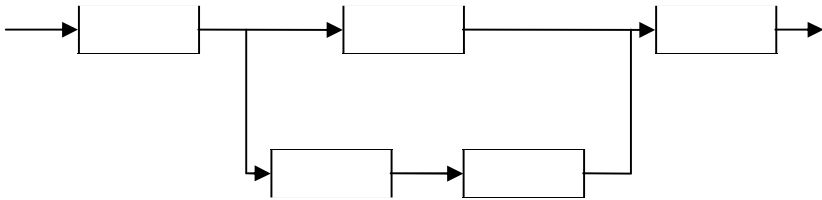


Рисунок 4.2.2 – Стохастическая сеть.

Обслуженная заявка покидает прибор обслуживания и покидает систему (*поглотитель заявок*), либо движется дальше в соответствии с технологической схемой работы системы.

*Различают следующие типы СМО:*

- *одноканальные и многоканальные – (по количеству ОУ);*
- *с ожиданием и без ожидания (с отказами);*
- *с ограничением на длину очереди (или с ограниченным ожиданием) и без ограничения;*
- *с упорядоченной очередью и с неупорядоченной очередью;*
- *с приоритетами и без приоритетов;*
- и др.

Любая модель строится для того, чтобы оценить какие-то показатели качества.

*Основные показатели качества обслуживания:*

- *общее количество обслуженных заявок за какой-либо промежуток времени;*
- *пропускная способность – среднее число заявок, обслуженных в единицу времени;*
- *доля заявок обслуженных;*
- *доля заявок, получивших отказ;*
- *время пребывания заявки в системе (от момента поступления заявки в систему до момента завершения ее обслуживания);*

- *среднее время обслуживания (функция распределения времени обслуживания);*
- *средняя длина очереди;*
- *среднее время ожидания;*
- *загрузка каналов – коэффициент использования (как доля времени, в течение которого ОУ было занято) – характеризует степень простоя ОУ;*
- *и др.*

Классические математические методы исследования СМО предложены теорией массового обслуживания. Чем аппарат аналитического моделирования СМО отличается от имитационного моделирования? Аналитические методы весьма стеснительны для решения практических задач: например, часто используется предположения о простейшем потоке заявок (однако для различных фаз обслуживания он может быть не простейшим), об однотипных устройствах и т.п. В имитационном моделировании подобные и другие ограничения снимаются: могут применяться произвольные законы распределения, различные схемы обслуживания (например, порядок обслуживания заявок из очереди и т.п.), СМО исследуется не обязательно в стационарном режиме (возможно изучение переходного режима, когда показатели отличаются от предельных асимптотических значений).

*Сущность метода имитационного моделирования для СМО состоит в следующем. Используются специальные алгоритмы, позволяющие вырабатывать случайные реализации потоков событий и моделировать процессы функционирования обслуживающих систем. Далее осуществляется многократное воспроизведение, реализация случайных процессов обслуживания и на выходе модели – статистическая обработка полученных статистических данных, оценка показателей качества обслуживания.*

#### **4.2.4 GPSS – транзактно-ориентированная система моделирования.**

*GPSS – является системой дискретного типа. Система GPSS ориентирована на класс объектов, процесс функционирования которых можно представить в виде множества состояний и правил перехода из одного состояния в другое, определяемых в дискретной пространственно-временной области. GPSS позволяет описывать процессы с дискретными событиями.*

Для регистрации изменений во времени существует таймер модельного времени. Механизм задания модельного времени: по-событийный, с переменным шагом. Изменения в реальной системе приводят к появлению событий. Событие – изменение состояния любого элемента системы. В системе происходят такие события, как:

- поступление заявки;
- постановка заявки в очередь;
- начало обслуживания;
- конец обслуживания и др.

В GPSS рассматриваются 2 класса событий:

- *основные* (те события, которые можно запланировать, то есть рассчитать момент их наступления заранее до их появления, например, момент появления заявки на входе);
- *вспомогательные* (те события, которые происходят вследствие появления основных событий. Вспомогательные события осуществляются в результате взаимодействия таких абстрактных элементов как блоки и транзакты, например, смена состояния прибора обслуживания со "свободен" на "занято").

GPSS относится к классу **процессно- (транзактно) -ориентированных систем моделирования**. GPSS является способом алгоритмизации дискретных динамических систем. Примеры моделируемых объектов: транспортные объекты, склады, производственные системы, магазины, торговые объекты, сети ЭВМ, системы передачи сообщений. Алгоритмическая схема может быть использована для оформления сложных формальных схем. Формальные модели таких объектов: СМО и стохастические сети, автоматы, сети Петри, автоматы и др.

#### 4.2.5 Функциональная структура GPSS.

Функциональная структура GPSS рассматривается на двух уровнях.

- 1 уровень определяется комбинацией основных *функциональных объектов* таких, как:
  - *устройства;*
  - *памяти;*
  - *ключи (логические переключатели);*
  - *очереди;*
  - *транзакты.*
- 2 уровень — *блок-схема модели*, составленная из типовых блоков, между которыми перемещаются транзакты.

Рассмотрим основные аппаратно-ориентированные, статистические и вычислительные объекты 1 уровня.

**Аппаратно ориентированные объекты:**

- **Транзакты** являются абстрактными подвижными элементами, которые являются аналогами различных объектов реального мира (сообщения, транспортные средства, люди, детали т.д.) Это

динамические функциональные элементы GPSS, которые отражают реальные заявки на обслуживание.

Транзакты двигаются по модели, появляются в ней с той же интенсивностью, что и реальные заявки. Транзакты могут создаваться и уничтожаться. Перемещаясь между блоками модели в соответствии с логикой моделирования, транзакты вызывают (и испытывают) различные действия:

- возможны их задержки в некоторых точках модели (связанные с обслуживанием, ожиданием в очереди);
- изменение маршрутов и направления движения;
- создание копии транзактов и т.д.

С каждым транзактом связан упорядоченный набор параметров – *атрибутов*.

При генерации транзактов резервируются 12 параметров. Обычно первые 12 параметров являются постоянными. В их набор входит:

- № транзакта;
- № блока, в котором транзакт находится в данный момент;
- № следующего блока;
- время перехода в следующий блок;
- приоритет, характеризующий очередность обработки транзактов в определенных ситуациях;
- и др.

Далее при программировании можно присвоить транзакту набор специфических параметров, выражающих свойства или характеристики моделируемых объектов (вес, скорость, цвет, время обработки и т.п.).

- **Устройства** моделируют объекты, в которых может происходить обработка транзактов, что связано с затратами времени. Устройства являются аналогами каналов СМО (каждое устройство в данный момент времени может быть занято лишь одним транзактом). Устройство может быть прервано. В GPSS существует возможность проверки состояния устройства.
- **Памяти** предназначены для моделирования объектов, обладающих емкостью. Аналогия с многоканальными СМО состоит в том, что память может обслуживать одновременно несколько транзактов. При этом транзакт занимает определённую часть памяти.
- **Логические переключатели** принимают значение включено/выключено, позволяют изменять пути следования транзактов в модели.

Устройства, памяти, логические переключатели относятся к аппаратно-ориентированным объектам GPSS.

**Статистические объекты GPSS** (используются только тогда, когда необходимо собирать статистику):

- **Очереди.** В процессе движения транзакты могут задерживаться в определенных точках модели. Если необходимо собирать информацию о длине очереди транзактов и времени задержки транзактов используют соответствующие статистические объекты.
- **Таблицы.** Таблицы обрабатывают статистическую информацию, строят гистограмму распределений по любой переменной.

**Вычислительные объекты GPSS:**

- матрицы
- функции
- переменные различных типов
- и т.п.

Рассмотрим 2 уровень. Модель на языке моделирования GPSS имеет наглядное графическое представление в виде **блок-схемы**.

**Блоки – операционные объекты GPSS.** Каждый блок имеет стандартное обозначение. Последовательность блоков – это есть последовательность операторов на языке GPSS. Любую модель на языке GPSS можно представить в виде совокупности блоков, между которыми перемещаются транзакты, они имеют вход-выход, в блоках реализуются все действия, связанные с обслуживанием транзакта (создание и уничтожение транзактов, изменение параметров транзакта, управление потоками транзактов, и т.д.). Блоки выполняются только в результате входа в них перемещающихся транзактов. GPSS является системой интерпретирующего типа с собственным языком.

Таким образом, на языке GPSS составляется и реализуется *функциональная блок-схема*.

Существуют 2 особых блока: *GENERATE*, имеющий только выход, через него транзакты входят в модель, и блок *TERMINATE*, имеющий только вход – удаляет транзакты из модели. Любой процесс на языке моделирования GPSS имеет вид:

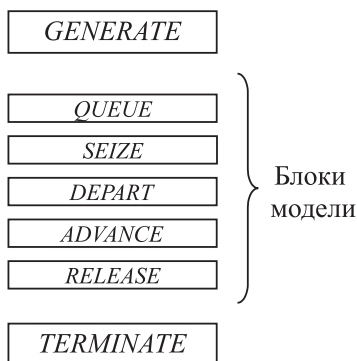


Рисунок 4.2.3 – Процесс на языке моделирования GPSS.

Описание параллельных процессов на языке GPSS представляет несколько таких цепочек блоков, взаимодействующих через общие ресурсы. Вот почему язык моделирования GPSS позволяет описывать параллельные процессы, независимые друг от друга, а взаимодействующие через общие ресурсы или переменные.

Итак, модель системы на языке GPSS представляет сеть блоков (операторов языка). Каждый блок описывает определенный этап действий в системе. Линии соединения блоков показывают направления движения подвижных элементов (транзактов) через систему или описывают некоторую последовательность событий, происходящих в моделируемой системе.

В настоящее время появились различные обобщения рассмотренной концепции структуризации, когда структура моделируемого процесса изображается в виде потока, проходящего через обслуживающие устройства и другие элементы СМО: сети очередей, графы потоков, структурно-стохастические графы и др. Дуги на графах интерпретируются как потенциальные потоки заявок между обслуживающими устройствами. Пути на графах соответствуют маршрутам движения заявок в системе обслуживания.

Подробнее изучение языка моделирования можно выполнить с помощью [5,8,12, 41,48, 55]

## 4.3 Агрегативные модели.

### 4.3.1 Кусочно-линейный агрегат.

В 60-х годах Н. П. Бусленко и И. Н. Коваленко [11, 24, 47] ввели класс моделей сложных систем, названных ими **агрегативными**. Основным элементом построения таких моделей был **кусочно-линейный агрегат**

(КЛА). Эти модели обладают рядом привлекательных свойств, позволяющих использовать их в рамках общей схемы исследования сложных систем. В работах отечественной научной школы [11] интенсивно исследовались их структурные и поведенческие свойства, создана имитационная система АИС (агрегативная имитационная система), базирующаяся на понятии агрегативной модели.

Будем рассматривать приводимые ниже определения и конструкции в форме, максимально приближенной к их программной реализации.

Дадим сначала **определение кусочно-линейный агрегата (КЛА)**.

КЛА относится к классу объектов, которые принято изображать в виде преобразователя (рис. 4.3.1), функционирующего во времени  $t \in T = [0, \infty)$  и способного воспринимать *входные сигналы*  $x$  со значениями из некоторого множества  $\mathbf{X}$ , выдавать *выходные сигналы*  $y$  со значениями из множества  $\mathbf{Y}$  и находиться в каждый момент времени в некотором *состоянии*  $z$  из множества  $\mathbf{Z}$ .

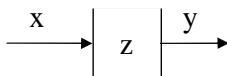


Рисунок 4.3.1 Общий вид преобразователя.

Класс КЛА отличает специфика множеств  $\mathbf{X}$ ,  $\mathbf{Y}$ ,  $\mathbf{Z}$ , допустимые формы входных и выходных сообщений (т. е. функций  $x(t)$  и  $y(t)$ ,  $t \in \mathbf{T}$ ), траекторий  $z(t)$ ,  $t \in \mathbf{T}$ , а также способ преобразования входного сообщения в выходное. Приступим к указанной выше конкретизации КЛА.

Прежде всего отметим, что **динамика** КЛА носит "событийный" характер.

В КЛА могут происходить события двух видов: *внутренние* и *внешние*. Внутренние заключаются в достижении траекторией КЛА некоторого подмножества  $\mathbf{Z}^* \subset \mathbf{Z}$  состояний; внешние — в поступлении входного сигнала.

Между событиями состояние КЛА изменяется детерминированным образом. Каждому состоянию  $z$  ставится в соответствие величина  $\tau = \tau(z)$ , трактуемая как потенциальное время до наступления очередного внутреннего события. Состояние КЛА в момент  $t^*$  — наступление события является случайным.

В момент  $t^*$  наступления внутреннего события, — выдается выходной сигнал  $y^*$ , содержание которого зависит лишь от  $z^*$ . (В частности, выходной сигнал может быть и пустым, т. е. не выдаваться). После случайного скачка  $\tau(z)$  вновь определяется время до наступления внутреннего события.



Рассмотрим теперь момент  $t^{**}$  наступления внешнего события, связанного с поступлением входного сигнала. Тогда состояние КЛА в момент  $t^{**}$  является случайным, зависящим лишь от  $x$  и  $z^{**}$ . В момент  $t^{**}$ , выдается выходной сигнал  $y^{**}$ , содержание которого определяется  $x$  и  $z^{**}$ .

(Условимся считать, что если моменты наступления внешнего и внутреннего событий совпадают, то изменение состояния осуществляется в соответствии с правилом наступления внешнего события, т. е. входные сигналы имеют *приоритет* над внутренними событиями).

Таким образом, динамику КЛА можно представить в следующем виде. Пусть в некоторый момент задано состояние КЛА. Тогда определяется время  $\tau(z)$ , через которое совершается *случайный скачок*, и меняется состояние. Начиная с момента наступления события (внешнего или внутреннего), ситуация повторяется, и динамику КЛА можно описать в виде *задания фазовой траектории изменения состояний  $z(t)$* , определенной на  $t \in T = [0, \infty)$ . Процесс функционирования КЛА полностью определяется изменениями, происходящими в особые моменты времени – моменты наступления событий (внешних или внутренних). Между особыми моментами состояние КЛА меняется *детерминированно*.

Опишем теперь КЛА более подробно.

КЛА внешне имеет вид многополюсника с  $m$  входными клеммами и  $n$  выходными клеммами (рис.4.3.2). Отметим, что в общем случае для различных КЛА  $m \neq n$ .

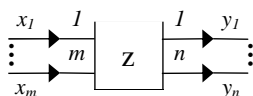


Рисунок 4.3.2 – Кусочно-линейный агрегат.

Предположим, что в состав множеств  $X_i$  и  $Y_j$  включены и фиктивные элементы  $0$ , наличие которых на входе или выходе КЛА означает отсутствие сигнала на соответствующей входной или соответственно выходной клемме.

Следовательно, входной сигнал на КЛА имеет вид

$$x = (x_1, x_2, \dots, x_m),$$

а выходной

$$y = (y_1, y_2, \dots, y_n).$$

Рассмотрим, на чем основана программная реализация агрегативных моделей. Не фиктивными входными  $x_i$  или выходными  $y_j$  сигналами, а также состояниями  $z$  КЛА являются *данные*. Данными считаются: элементарные данные; списки данных; массивы данных; структуры

данных. *Элементарными данными* считаются: целые числа; действительные числа; символьные переменные. Здесь термины "список", "массив" употребляются в их обычном смысле. Понятие структуры данных соответствует дереву, на корнях которого размещены данные. Каждое данное имеет свое имя. Рассматриваемые данные хорошо отображают содержательные представления, существующие у исследователя относительно реальных объектов, и существенно облегчают процесс построения модели. Эти данные удобны как с математической, так и с программной точек зрения.

Пусть состояние  $z$  КЛА определено как некоторая структура данных. Тем самым фиксирован вид дерева, представляющего эту структуру.

Дерево базируется в конечном счете на элементарных данных. Обозначим через  $\mathbf{I}_z$  элементарные данные, входящие в состояние  $z$  и имеющие тип целых чисел и символов, а через  $\mathbf{R}_z$  — элементарные данные, имеющие действительный тип. Предположим, что значения и состав элементарных данных могут меняться лишь в особые моменты времени, а между ними остаются постоянными. Разобьем множество  $\mathbf{R}_z$  на два подмножества  $\mathbf{R}_z = \mathbf{R}_z^+ \cup \mathbf{R}_z^-$ , где  $\mathbf{R}_z^+$  состоит из положительных величин, а  $\mathbf{R}_z^-$  — из неположительных. Будем считать, что данные из подмножества остаются неизменными между особыми моментами времени и что моменты наступления внутренних событий определяются лишь данными из  $\mathbf{R}_z^+$ . Это отвечает обычно используемой "энергетической интерпретации" причин наступления внутренних событий в моменты, когда исчерпывается некоторый ресурс, оканчивается операция и т. д. Таким образом, внутреннее событие происходит, когда хотя бы один из положительных элементов множества обращается в нуль.

Аналогично задается реакция КЛА на входной сигнал  $x = (x_1, x_2, \dots, x_m)$ .

#### 4.3.2 Схема сопряжения. Агрегативная система.

Формализуем понятие *структуры сложной системы* [47].

Структура сложной системы — есть формализованное множество КЛА.

Введем понятие агрегативной системы: *агрегативная система представляется либо как КЛА, либо как объединение конечного числа агрегативных систем. Это объединение описывается через схему сопряжения, где КЛА связаны через каналы связи, по которым передаются сигналы.*

Пусть  $A = (A_1, A_2, \dots, A_N)$  — множество КЛА,  $N$  — фиксированное число.

Обозначим через  $\mathbf{I}_i$  — множество входных клемм КЛА  $A_i$ , а через  $\mathbf{O}_i$  — множество

Пусть  $\mathbf{I} = \bigcup_{i=1}^N \mathbf{I}_i$ ;  $\hat{\mathbf{I}} = \bigcup_{i=1}^N \mathbf{O}_i$ , — множества входных и выходных клемм

всех рассматриваемых КЛА.

Рассмотрим множество  $R$  всех возможных отображений  $R, \mathbf{I} \rightarrow \mathbf{O}$ , которое можно интерпретировать как множество потенциальных соединений между собой КЛА, входящих в  $A$ .

Именно, каждой входной клемме из  $\mathbf{I}$  ставится в соответствие выходная клемма, с которой на входную клемму поступает сигнал. При этом допускается, что некоторым из входных клемм не ставятся в соответствие никакие выходные, т. е. на эти входные клеммы не поступают никакие сигналы. Точно так же некоторые выходные клеммы могут быть "висячими": сигналы с этих клемм могут никуда не поступать. Таким образом, на каждую входную клемму подаются сигналы не более чем с одной выходной клеммы. В то же время с одной выходной клеммы сигналы могут идти на несколько различных входных клемм.

Это допустимое отображение  $R \in R$  называется *схемой сопряжения* (рис. 4.3.3). Схема сопряжения указывает адресацию сигналов в системе, состоящей из КЛА.

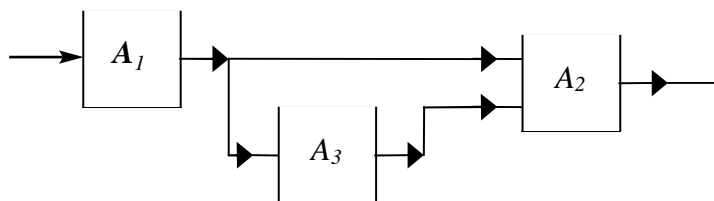


Рисунок 4.3.3 – Схема сопряжения  $R$ .

Если  $(j, r)$  – выход является значением отображения  $R(i, l)$  – входа и при этом отображение  $R$  является схемой сопряжения, то будем говорить, что между  $(j, r)$  – выходом и  $(i, l)$  – входом проложен канал связи.

Таким образом, состав агрегатов  $A$  и схема сопряжения  $R$  определяют агрегативную систему.

Однако задать состав агрегатов  $A$  и схему сопряжения  $R$  еще недостаточно для однозначного определения динамики получившейся системы. Вводят 2 следующих предположения:

*Предположение 1.* Каналы связи в системе, состоящей из КЛА, являются идеальными, т. е. передающими сигналы мгновенно и без искажений.

Данное предположение весьма удобно как с математической, так и с программной точек зрения. Отметим, что хотя предположение 1 сужает круг рассматриваемых моделей, мы можем учесть возможные задержки и искажения, присутствующие в реальной системе, введя дополнительные КЛА, отображающие эти особенности реальных каналов связи.

Но и предположения 1 еще недостаточно для определения динамики модели. Поясним причину этого.

Пусть, по предположению, каждый из агрегатов в ответ на поступление любого входного сигнала мгновенно выдает выходной сигнал. Тогда налицо неопределенность. Как должен реагировать агрегат  $A$ . Он может сначала отреагировать на сигнал и затем (через нулевой промежуток времени) выдать второй выходной сигнал в ответ на поступление сигнала. В этом случае реакция  $A$  состоит в выдаче пары выходных сигналов. Возникает "сопоставительная" ситуация. Разрешить эту неопределенность и данное противоречие можно с помощью следующего предположения, которое отвечает принципу причинности.

*Предположение 2.* Сигналы в системе из взаимодействующих КЛА обрабатываются следующим образом. Упорядочивают выходные сигналы по номерам агрегатов и номерам выходных клемм. В пределах одного агрегата нумерация сигналов соответствует нумерации клемм.

На основании схемы сопряжения  $R$  определим адреса сигналов и последовательно найдем реакции на них. Если при этом будут выданы выходные сигналы, то назовем их сигналами второй стадии и т. д. Все указанные реакции осуществляются в один и тот же момент системного времени, но по стадиям. Таким образом, упорядочивают номера агрегатов, номера выходных клемм и последовательно находят реакции на сигналы.

Схема сопряжения в совокупности с предположениями 1 и 2 полностью определяет *динамику агрегативной системы*.

Мы описали агрегативную систему с постоянной схемой сопряжения. Легко видеть, что в приведенных построениях ничего не изменится, если предположить, что схема сопряжения  $R$  меняется всякий раз при наступлении очередного системного события и ее вид зависит лишь от состояния системы непосредственно перед наступлением события. При этом получим *систему с переменной структурой*.

Определенные выше агрегативные системы обладают важным свойством *замкнутости*, заключающимся в том, что **агрегативная система в целом может быть описана в виде КЛА** [11]. Данный факт интуитивно понятен и здесь не доказывается.

Следовательно, и **объединение конечного числа агрегативных систем также является агрегативной системой**.

*Это является одной из основ проведения структурных преобразований с агрегативными системами и их моделирующими алгоритмами.*

### **4.3.3 Оценка агрегативных систем как моделей сложных систем.**

Введенное понятие агрегативной системы дает возможность описания самых разнообразных объектов реального мира в агрегативном виде.

Агрегативные системы служат определенным обобщением таких хорошо известных схем, как автоматы и модели массового обслуживания. Нетрудно понять, что в агрегативном виде также могут быть представлены сети Петри и практически любые численные методы решений дифференциальных уравнений (обыкновенных и в частных производных). Таким образом, агрегативные системы охватывают широкий класс различных моделей, используемых при изучении сложных систем. В агрегативном виде можно также представить модели, имеющие вид "черного ящика". Структуры данных, описывающие состояния и сигналы агрегативных систем, помогают формализовать концептуальное представление, которое существует у пользователя относительно элементов сложной системы.

Эти описания достаточно естественны и удобны, что можно объяснить следующими причинами.

Агрегативные модели соответствуют основным концепциям описания сложных систем:

В основу понятия агрегативной модели положено структурное представление системы в виде взаимодействующих элементов – КЛА. Это соответствует одной из основных концепций сложных систем, рассмотренных нами в 1 лекции.

Динамика агрегативной системы полностью определяется последовательностью событий, происходящих в ней, что отвечает концепции алгоритмической модели динамической сложной системы.

В соответствии с алгоритмической моделью и понятием агрегативной системы возможны реализации разнообразных машинных моделей сложных систем. Одним из преимуществ является удобство реализации агрегативных систем на ЭВМ. Примером может являться агрегативная имитационная система.

Таким образом, можно констатировать, что агрегативная формализация удовлетворяет основным требованиям, предъявляемым к моделям сложных систем.

## **4.4 Сети Петри и их расширения.**

### **4.4.1 Описание структур моделируемых проблемных ситуаций в виде сетей Петри.**

Сети Петри и их обобщения представляют собой математические модели, построенные в рамках определенной концепции структуризации.

Концепция структуризации базируется на возможности представления моделируемых систем в виде совокупности параллельных процессов, взаимодействующих на основе синхронизации событий или распределения общих для нескольких процессов ресурсов. Каждый процесс в рамках этой концепции представляется в виде логически обусловленных не упорядоченных по времени причинно-следственных цепочек условий и событий.

Рассмотрим общие подходы к описанию структур моделируемых проблемных ситуаций в виде сетей Петри. Сети Петри – удобный аппарат моделирования параллельных процессов, т.е. процессов, протекающих независимо один от другого.

При разработке структур моделей дискретных систем в качестве базовой информации можно использовать данные о логической взаимосвязи наблюдаемых в системе событий и условий, предопределяющих наступление этих событий.

В некоторых реальных системах нельзя точно предсказать момент времени наступления событий. Наступление событий предвещает сложная система причин и следствий. Точное знание моментов времени реализации событий в системе часто можно игнорировать, поскольку такие сведения о событиях, происходящих в реальных (или проектируемых) системах, либо просто отсутствуют, либо их нельзя считать достоверными. Это объясняется многообразием предвещающих события условий, невозможностью полного их учета и верного описания, а также действием сложной и запутанной системы причин и следствий, определение которых на временной шкале часто не представляется возможным.

Вводятся базовые понятия "Условие" и "Событие", которые могут быть связаны отношением типа "Выполняется после" (рис. 4.4.1).

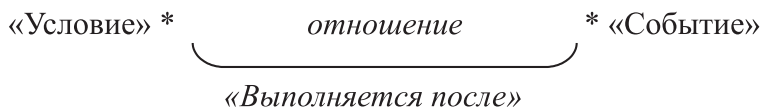


Рисунок 4.4.1 – Описание структур моделируемых дискретных систем в виде сетей Петри.

События выражают действия, реализация которых управляет состояниями системы. Состояния задаются в виде сложных условий, формулируемых как предикаты с переменными в виде простых условий. Только при достижении определенных состояний (в этом случае соответствующие предикаты принимают истинное значение) обеспечивается возможность действий (наступления событий). Условия, с фактами выполнения которых связана истинность предиката и, следовательно,

возможность реализации события, называют "до-условиями" (предпосылками наступления события).

В результате действия, совершившегося при реализации события, объявляются истинными все простые условия, непосредственно связанные с данным событием отношением "Выполняется после". Эти условия рассматриваются как "пост-условия" (прямые следствия событий).

Только после выполнения всех "до-условий" для некоторого события это событие может быть выполнено. После того как событие имело место, истинными становятся все "пост-условия" данного события, которые затем, в свою очередь, могут быть "до-условиями" каких-либо других событий и т. д. Таким образом, оформляется логическая взаимосвязь событий и условий, предопределяющих эти события – *в виде логически обусловленных причинно-следственных цепочек условий и событий*. Построение полной структуры таких отношений для моделируемой проблемной ситуации составляет цель и задачу *формирования структуры модели*.

В таблице 4.4 рассматривается фрагмент моделируемой производственной системы, отражающий некоторый процесс обслуживания деталей на станке. При структуризации объекта моделирования были выделены следующие события и условия:

До-условия	События	Пост-условия
$p_1 - i$ - деталь готова к обработке, находится на месте комплектации $p_2$ - станок свободен	$t_1$ - начинается программа обработки детали на станке	$p_3$ - освободилось место в накопителе деталей $p_4$ - выполняется программа обработки детали
$p_4$	$t_2$ - выполнена программа обработки $i$ -детали	$p_5 - i$ - деталь обработана $p_2$ - станок свободен

Таблица 4.4 – Структуризация производственной системы (фрагмент).

#### 4.4.2 Формальное и графическое представление сетей Петри.

Рассмотренная концепция структуризации моделируемой проблемной ситуации поддерживается формальными средствами, разработанными в теории сетей Петри.

В сетях Петри условия моделируются позициями, а события — переходами.

Формально сеть Петри представляет собой набор:

$C = (P, T, E)$ , где

$P$  — непустое конечное множество позиций сети;

$T$  — непустое конечное множество переходов;

$E = (P \times T) \cup (T \times P)$  — отношение инцидентности позиций и переходов (множество дуг сети) — логически обусловленные причинно-следственные связи между событиями и условиями.

Также могут быть заданы:

- $W: F \rightarrow N$  — функция кратности дуг (каждой дуге ставится в соответствие  $n > 0$  — кратность дуг);
- $M: P \rightarrow N$  — функция начальной разметки.

В различных расширениях сетей Петри используются графические представления — графы, орграфы, диграфы — в общем виде некоторые сетевые представления.

**Графически ординарные сети Петри** представляются двудольными орграфами:

$C = (P, T, E)$ .

Множество вершин в таких орграфах состоит из непересекающихся подмножеств позиций

$P = \{p_i\}$ ,  $i = \overline{1, |P|}$  и переходов  $T = \{t_j\}$ ,  $j = \overline{1, |T|}$ , а множество дуг  $E$  разделяется на два подмножества  $\{(p_i, t_j)\}$  и  $\{(t_j, p_i)\}$ . Дуги  $(p_i, t_j)$  ориентированы от позиций к переходам, а дуги  $(t_j, p_i)$  — от переходов к позициям.

В изображении графов, представляющих ординарные сети Петри, позиции принято обозначать кружками, а переходы — барьерами (планками) следующим образом:



Рисунок 4.4.2 — Обозначения основных элементов сетей Петри.

Для примера рассмотрим фрагмент сети Петри, моделирующей структуру процессов функционирования производственной системы, соответствующий примеру, приведенному в таблице 4.4.



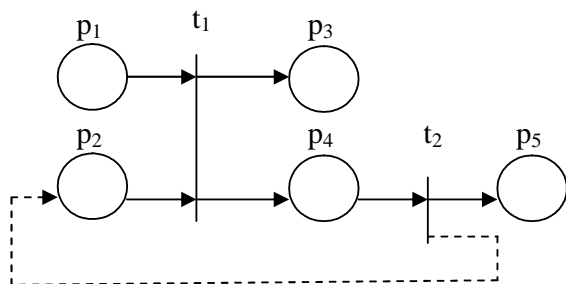


Рисунок 4.4.3 – Фрагмент сети Петри.

Реальный процесс может иметь более сложное ветвящееся графическое изображение (рис. 4.4.4).

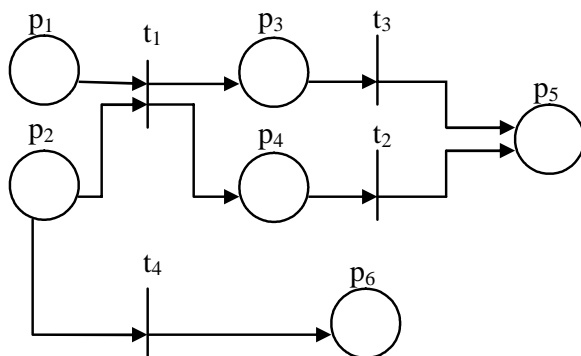


Рисунок 4.4.4 – Фрагмент сети Петри.

#### 4.4.3 Динамика сетей Петри.

Динамика поведения моделируемой системы отражается в функционировании сети в виде совокупности действий, называемых *срабатыванием переходов*.

Динамика сетей Петри обусловлена соглашениями относительно *правил срабатывания переходов*. Действующие в сетях Петри соглашения о правилах выполнения переходов выражают логические взаимосвязи между условиями и событиями в моделируемой системе. Переход может сработать (*срабатывание перехода*), если выполнены все условия реализации соответствующего события. Последовательная реализация событий в системе отображается в сети в виде последовательного срабатывания ее переходов.

*Механизм маркировки позиций.* Изменение состояния сети связано с механизмом изменения маркировок позиций. Начальное состояние сети

Петри задается с помощью маркировки ее позиций. *Маркировка сети (разметка позиции-места)* заключается в присвоении позициям числовых значений (меток, маркеров, фишек). Метки представляют собой набор атрибутов (числа, переменные). Например, начальную разметку можно записать в виде вектора  $M_0(p) = (00000110100)$ , где 1- соответствует условиям, которые должны быть выполнены.

Когда емкости позиций невелики, в качестве меток на графических изображениях сетей Петри используют не числа, а *фишки, маркеры*. Выполнение условия изображается разметкой (маркировкой) соответствующего места, а именно помещением числа  $n$  или  $n$ - фишек (маркеров) в эту позицию. Изменение состояния сети связано с изменением маркировок позиций, в этом случае выполняется возбужденный переход, т.е. переход с ненулевыми метками. Переход может сработать, если выполнены все условия реализации соответствующего события. Для этого задаются специальные правила или *процедуры перехода*. На рисунке 4.4.5 (а, б, в) демонстрируется, как в сетях Петри реализуется механизм маркировки.

Таким образом, срабатывание перехода — это неделимое действие, изменяющее разметку: из каждого входного места (позиции) изымается по одной фишке, а в каждое выходное место (позицию) добавляется по одной фишке. Тем самым реализация события, изображенного переходом, изменяет состояние связанных с ним условий (уменьшается емкость до-условий, увеличивается емкость пост-условий).

Применяются следующие правила изменения маркировок:

- выполняется только возбужденный переход, т.е. такой переход, во всех входных позициях которого имеются ненулевые метки;
- срабатывание перехода может наступить через любой конечный промежуток времени после его возбуждения;
- если в некотором состоянии сети возбужденным оказывается сразу несколько переходов, то всегда выполняется только какой-то один (любой) из них;
- в результате срабатывания перехода метки в каждой позиции перехода уменьшаются на единицу, а метки во всех его выходных позициях увеличиваются на единицу;
- выполнение перехода — неделимый акт, изменение разметки входных и выходных позиций перехода при его выполнении осуществляется мгновенно.

Всякому возможному варианту выполнения сети будет отвечать определенная последовательность срабатываний переходов и последовательность получающих после каждого очередного срабатывания маркировок вида:  $M_0 > M_1 > M_2 > \dots M_k$ .

Сети Петри формализуют понятие *абстрактной асинхронной системы динамической структуры из событий и условий*. В сетях Петри не моделируется ход времени, события упорядочиваются по отношению "Выполняется после".

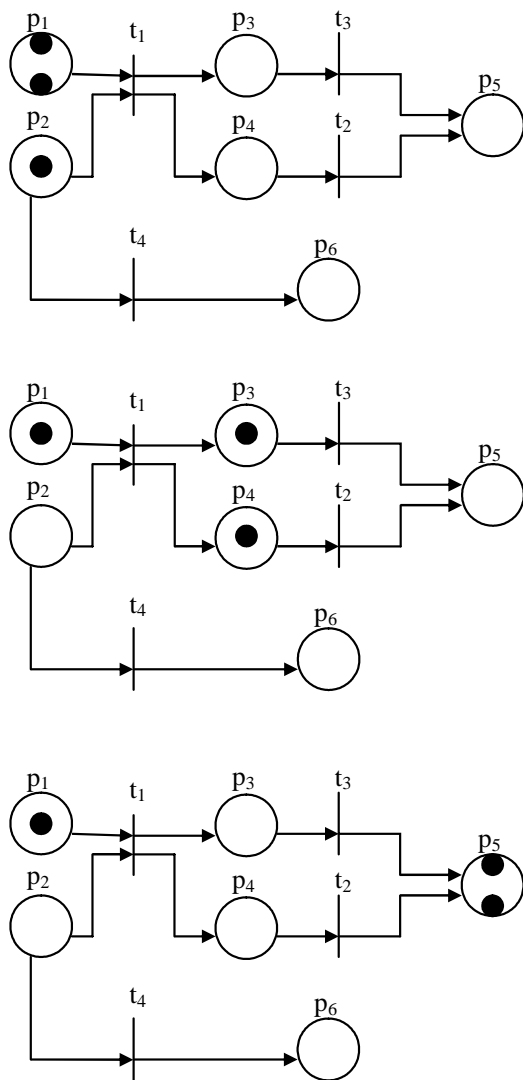


Рисунок 4.4.5 – Реализация механизма маркировки в сетях Петри.

#### 4.4.5 Различные обобщения и расширения сетей Петри.

Сети Петри моделируют широкий класс систем, но для некоторых распространенных специальных классов систем удобно применять сети Петри не общего вида, а некоторые их подклассы или расширения (иерархические сети, раскрашенные сети Петри, сети событий, временные сети событий, КОМБИ-сети, ЕД-диаграммы), более адекватные рассматриваемым системам.

**Регулярные сети.** Основным свойством регулярных сетей является наличие определенной алгебры, которая обеспечивает проведение аналитического представления процесса топологического построения и расчленения процесса анализа сетей на совокупность этапов, на каждом из которых достаточно иметь дело с более простыми фрагментами сети.

В свою очередь обобщением регулярных сетей являются **иерархические сети**, предназначенные для адекватного моделирования иерархических динамических систем. Иерархическая сеть функционирует, переходя от разметки к разметке, с некоторыми отличиями от работы регулярных сетей. Данные отличия связаны с присутствием составных переходов, срабатывание которых является не мгновенным событием, как в традиционных сетях Петри, а составным действием. Поэтому составной переход не срабатывает, а работает, т.е. находится некоторое время в активном состоянии.

При моделировании систем сетями Петри, часто возникают ситуации, при которых фишки позиций (мест) должны быть не "безлики", а должны соответствовать объектам, передаваемым от компонента к компоненту (от перехода к переходу). Причем, как правило, эти объекты имеют дополнительные атрибуты, которые позволяют различать их и использовать эти различия для управления функционированием системы. Для адекватного описания подобных ситуаций был разработан подкласс **раскрашенных сетей Петри**. В рассмотренных нами ранее сетях все метки предполагались одинаковыми. Механизм функционирования сетей был связан только лишь с количествами меток во входных позициях переходов и определялся общими для всех меток условиями возбуждения переходов и правилами изменения разметки позиций при выполнении сети. Появление раскрашенных сетей Петри связано с концепцией использования различных меток. В таких сетях каждая метка получает свой цвет. Условия возбуждения и правила срабатывания переходов для меток каждого цвета задаются независимо. В данных сетях фишкам приписываются некоторые признаки, например различные цвета (переменные), а кратности дуг интерпретируются как функции от этих переменных. Условия срабатывания переходов и правила изменения разметки задаются специальной таблицей, учитывающей цвета фишек.

Дальнейшим расширением раскрашенных сетей явились **предикатные сети**. Данные сети позволили связывать с переходами сетей логические формулы (предикаты или защиту), представляющие классы возможных разметок во входных и выходных позициях в соответствии с метками дуг. Эти выражения задают условия отбора необходимых цветов для срабатывания переходов.

#### **4.4.6 Технология разработки моделей.**

При разработке имитационной модели из класса расширений сетей Петри, как впрочем и любой другой имитационной модели, выделяются четыре основные этапа: структуризации, формализации и алгоритмизации, программирования модели, а далее проведения имитационных экспериментов с моделью.

При *структуризации* определяются и неоднократно уточняются:

- действующие в системе процессы и используемые ресурсы;
- множество позиций (отображают в модели состояния процессов и ресурсов) и множество переходов (событий);
- подмножество синхронизирующих (для описания параллельных процессов) переходов.

При *формализации и алгоритмизации* элементов модели для каждой позиции определяются атрибуты меток. Переход считается формально описанным, если известны:

- множества смежных с этим переходом позиций;
- условий возбуждения перехода;
- схема выполнения;
- процедура перехода.

*Условия возбуждения перехода* — есть некоторый предикат, принимающий истинное значение, если реализуется некоторая разметка позиций множества  $E$  (проверяются атрибуты меток). *Схема выполнения* определяет изменение разметки позиций сети при срабатывании перехода. *Процедура перехода* представляет собой правила вычисления атрибутов или добавления меток.

*Программирование модели* связано с описанием позиций и переходов сети, оформляемых с помощью некоторых языков программирования или моделирования, например GPSS.

### **4.5 Модели системной динамики.**

Теперь нам предстоит познакомиться с одним из способов формализации непрерывных систем.

Под *непрерывной системой* обычно понимается система, состояние которой изменяется непрерывным образом в зависимости от некоторых независимых переменных (обычно времени).

Языки имитационного моделирования непрерывных систем предназначены для моделирования динамических объектов с непрерывным фазовым пространством и непрерывным временем. Как правило, такие объекты описываются с помощью систем дифференциальных (интегро-дифференциальных) уравнений. Однако, в имитационном моделировании формализация явлений в виде математических схем — больше дань традициям, сегодня находят применение подходы, основанные на графической технике структуризации моделируемых динамических процессов.

Классическим примером языков моделирования такого типа явился язык *DYNAMO*, предложенный Дж. Форрестером. Дж. Форрестер — крупнейший специалист, разработчик концепции системной динамики, основные его работы: "Основы кибернетики предприятия" (1961г.); "Динамика развития города" (1970 г.); "Мировая динамика" (1974 г.) В этих работах изложен метод системной динамики или концепция потокового подхода в имитации, а также исследуются динамика предприятия, урбанизированной территории, проблемы развития человеческой цивилизации на основе предложенной концепции.

Методы и техника построения моделей системной динамики оказали большое влияние на формирование технологии имитационного моделирования. Сейчас появилось много систем моделирования, т.к. *iThink*, *Vensim*, *Powersim* в которых возможности моделирования непрерывных и нелинейных динамических систем дополнены удобными графическими интерфейсами. Сейчас много фирм разрабатывают системы моделирования, реализующие принципы системной динамики, предлагают консалтинговые услуги по моделированию систем на основе инструментальных средств, поддерживающих нормативные подходы методов системной динамики. Существует международное научное общество, которое развивает и совершенствует методы системного анализа и моделирования систем на основе концепции системной динамики.

Модели системной динамики получили широкое распространение в задачах исследования сложных систем из сферы производства и экономики, торговли и городского хозяйства, из области социальных проблем, проблем экологии и охраны окружающей среды [15,17, 28, 51-53]. Эти модели были первыми машинными моделями, положившими начало новому направлению в системных исследованиях — так называемому *глобальному моделированию*, охватывающему проблемы мирового развития — моделирование мировой экономической системы [52].

#### 4.5.1 Общая структура моделей системной динамики. Содержание базовой концепции структуризации.

В основе этого класса моделей лежит концепция системной динамики, ориентированная на моделирование систем и процессов *на высоком уровне агрегирования*, где отображения отдельных элементов процессов, т.е. их дискретности, становится ненужным (например, экономика отдельных стран и регионов, транспортные системы страны, и т.п., проблемы мирового развития). *В основе концепции системной динамики лежит представление о функционировании системы, как совокупности потоков информации, энергии, промышленной продукции, денежных средств и т.п.*

*Математической (формальной) основой методов системной динамики являются дифференциальные модели, в которых используются представления динамических процессов в пространстве состояний. Модели такого вида – это системы дифференциальных уравнений:*

$$X' = f(x, u, t), \text{ где} \quad (4.5.1)$$

$X = (x_1, x_2, \dots, x_m)^T$  – вектор состояний;

$x_1, \dots, x_m$  – переменные состояния;

$u = (u_1, \dots, u_p)^T$  – вектор входов;

$t$  – символ времени (в дальнейшем для краткости  $t$  опускается).

Дифференциальные модели, применяемые в математической теории систем, включают кроме уравнений (4.5.1), называемых уравнениями состояния, еще и уравнение:

$$y = H(x, u), \quad (4.5.2)$$

в котором переменная  $y = (y_1, \dots, y_q)^T$  – вектор выходов моделируемых процессов.

При составлении дифференциальных моделей производится выбор переменных состояния, и устанавливаются связи между этими переменными в виде функций правых частей уравнений состояния. Как правило, сформулировать такие зависимости только с использованием переменных состояния бывает очень трудно. Более продуктивным оказывается другой подход, основанный на детальном описании цепочек причинно-следственных связей между факторами, отображаемыми в модели с помощью переменных состояния.

Модели системной динамики разрабатываются на основе предоставляемых экспертами сведений об объектах исследования и в зависимости от целей изучения этих объектов. При построении моделей указанные сведения подвергаются тщательному анализу, в ходе которого находятся решения двух основных взаимосвязанных вопросов: *выбора и*

интерпретации переменных состояния модели; выявления причинно-следственных отношений между переменными состояния и описанием этих отношений в форме структурированных функциональных зависимостей вида  $f$  и  $H$ . По мнению экспертов, динамика состояний моделируемого объекта зависит от ряда факторов. Эксперт анализирует цепочки причинно-следственных связей этих факторов, которые должны обязательно учитываться при описании динамики состояний моделируемого объекта. А далее технология процесса структуризации информации о проблеме позволяет структурировать функции  $f$  и  $H$ .

В качестве общей структурной схемы при описании вектор-функций  $f(x, u)$  можно использовать граф, вершинам которого соответствуют переменные модели, а дугам — непосредственные функциональные связи между этими переменными. Поясним на примере (рис.4.5.1.), о каких графах идет речь.

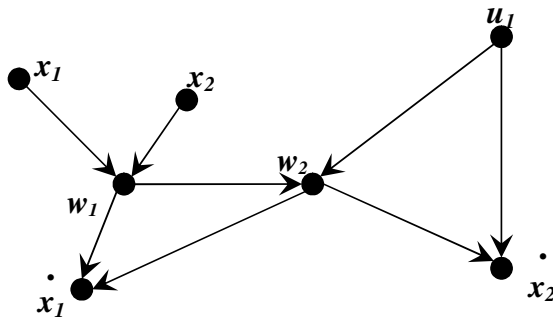


Рисунок 4.5.1 — Граф функциональных зависимостей переменных дифференциальной модели.

Требуется описать структуру правых частей уравнений состояния:

$$\begin{aligned}\dot{X}_1 &= f_1(x_1, x_2, u_1); \\ \dot{X}_2 &= f_2(x_1, x_2, u_1).\end{aligned}$$

После факторизации правых частей этих уравнений были получены структурирующие соотношения вида:

$$\begin{aligned}f_1(x_1, x_2, u_1) &= F_1(w_1, w_2); \\ w_1 &= F_2(x_1, x_2); \\ w_2 &= F_3(w_1, u_1); \\ f_2(x_1, x_2, u_1) &= F_4(w_2, u_1).\end{aligned}$$

Этому случаю будет соответствовать граф непосредственных функциональных зависимостей между переменными, представленный на рисунке 4.5.1. Данный граф имеет четырехъярусную структуру:



Вершины графа	Ярусы графа
$x_1, x_2, u_1$	0
$w_1$	1
$w_2$	2
$\dot{x}_1, \dot{x}_2$	3

Структуры правых частей уравнений дифференциальных моделей всегда могут быть представлены многоярусными графами, не имеющими контуров. Разработка таких цепочек причинно-следственных связей потребовала введения некоторых *дополнительных* переменных. Обозначим через  $w = (w_1, \dots, w_s)^T$  векторную переменную, компоненты которой образуют в совокупности множество включаемых при структуризации правых частей уравнений состояния (4.5.1) дополнительных переменных.

Алгоритмические формы записи дифференциальных моделей системной динамики имеют следующие характерные особенности:

Общая схема построения структур правых частей уравнений состояния этих моделей могут быть представлены в виде графовых представлений следующим образом:

Вершины таких графов образуют какой-то один ярус графа:

- нулевой ярус образуют вершины, отображающие на графе переменные состояния и входные переменные модели;
- последний ярус (с наибольшим номером) составляют вершины, представляющие на графе переменные левых частей уравнений состояния (4.5.1);
- вершинам всех промежуточных ярусов соответствуют переменные состояния и дополнительные переменные дифференциальной модели.

Нормативной схемой решения уравнений в моделях системной динамики является одношаговая схема первого порядка. Уравнения состояния моделей составляются в форме разностных уравнений вида:

$$X_{t+1} = X_t + hf(x_t, u_t).$$

Здесь  $h$  – шаг дискретизации, а  $t = 0, 1, 2, \dots$

Рассмотрим *содержание базовой концепции структуризации*.

**Говоря в целом о методах разработки моделей системной динамики, их можно охарактеризовать как способы структуризации дифферен-**

циальных моделей, базирующиеся на концепции потоковой стратификации систем.

В общей структуре моделей системной динамики выделяются две части:

#### I – “Сеть потоков”:

Отметим, что переменные состояния и переменные скорости изменения состояния (соответственно уровни и темпы в терминах системной динамики), определяющие состояние модели, задаются в сети потоков системами разностных уравнений и в описании присутствуют в неявном виде.

#### II – “Сеть информации”:

С помощью неё осуществляется структуризация функциональных зависимостей, иначе дается структурированное представление функции  $f(x_t, u_t)$ .

Это как бы *основные образы моделируемых процессов в системной динамике*.

Поясним сначала первую составляющую этой структурной схемы.

### 4.5.2 Основные понятия. Потоковая стратификация.

Как описывается динамика объекта моделирования в виде потоковых сетей?

Модель рассматривается в качестве *сети потоков материальных ингредиентов модели*. Каждая компонента этой сети соответствует какой-то одной совокупности однородных ингредиентов (например, предметы труда, население, денежные средства и т. п.), динамика которых учитывается в модели.

Например, рассматривая предприятие, с позиций системного подхода, в основе него можно выделить совокупность таких потоков, как финансовые, материальные, людские (кадры), потоки оборудования и др. (рис. 4.5.2).

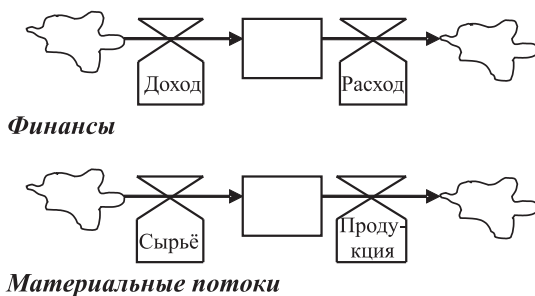


Рисунок 4.5.2 – Поток на предприятии.

Сеть имеет узлы и дуги.

Узлы компонент сети потоков (за исключением нулевого узла) изображают наиболее существенные с точки зрения разработчиков модели *состояния выделенных ингредиентов*, а дуги сети задают возможные переходы их элементов из одного состояния в другое (рис. 4.5.3).

Распределение элементов по состояниям меняется с течением времени. Эти изменения для системной динамики являются нормативными образами моделируемых процессов.

В качестве характеристик распределения элементов входящих в модель ингредиентов по состояниям  $X_1, \dots, X_m$  рассматриваются переменные  $x_1, \dots, x_m$  уравнений состояния модели. Переменные  $v_1, \dots, v_n$  принимаются за характеристики *интенсивностей (скоростей)*, с которыми совершаются переходы элементов из состояния в состояние по дугам  $V_1, \dots, V_n$  сети.

Узлы сети изображаются в виде прямоугольников (рис. 4.5.4а). На рисунке 4.5.4б использованы специальные символы потоковых сетей. В моделях системной динамики нуль сети принято обозначать специальным знаком "Озеро".

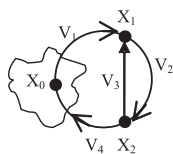


Рисунок 4.5.3 – Сеть потоков

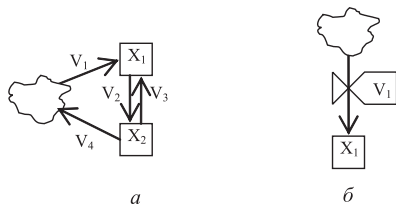


Рисунок 4.5.4 – Основные символы потоковых сетей: а – узлы сетей; б – потоковая дуга с символом темпа.

Такая интерпретация напоминает структурные формы задания автоматных моделей дискретных процессов. Однако модели системной динамики — это дифференциальные модели, переменные состояния которых непрерывны. Поэтому приведенной здесь структурной концепции уравнений состояния моделей системной динамики обычно дается более естественное объяснение, основанное на *гидравлической интерпретации потоковых сетей*.

В этой интерпретации каждый узел  $X_i \in X$  сети рассматривается как резервуар, уровень наполнения которого в момент  $t$  равен  $x_i(t)$ ,  $t=1,2,\dots, m$ . Дуги  $V_j \in V$  сети соответствуют потокам жидкости между резервуарами. Они указывают направление потоков, темпы которых характеризуются объемными расходами жидкости  $v_j(t)$ ,  $j = 1, 2, \dots, n$  на интервале времени  $[th, (t+1)h]$ . Темпы потоков символически изображаются в виде вентилях на дугах потоковых сетей (рис. 4.5.4б).

Наглядность гидравлической интерпретации обусловила ее важную роль при структуризации моделей, специальную "потоковую" терминологию и общую трактовку математических схем моделей системной динамики.

В соответствии с терминологией системной динамики переменные состояния  $x_1, \dots, x_m$  называются **уровнями** модели, а переменные  $v_1, \dots, v_n$  — **темпами**. *Уровни и темпы — основные переменные моделей*. Все остальные переменные, называются *вспомогательными (или дополнительными)*. Вспомогательные переменные, как мы уже знаем, используются при структуризации функциональных зависимостей  $f$  темпов от уровней и входов, а также функциональных зависимостей  $H$  выходов  $u$  от уровней  $x$  и входов  $i$ .

Таким образом, получается следующая картина при описании динамической системы: моделируемые процессы отображаются в виде некоторой фиксированной структуры, состоящей из *накопителей* — *уровней*, соединенных взаимосвязанными *потоками*, которые, перетекая по всей системе, изменяют значение уровней. *Уровни* характеризуют возникающее накопление внутри системы и являются величинами, которые определяются как переменные состояния системы (например: для банка — это сальдо, для склада — текущий уровень запасов на складе). Уровни описывают величины, непрерывные по диапазону значений, но дискретные во времени. *Потоки* изменяют значение уровней. В экономике, производстве потоками можно управлять. Потоки регулируются управленческими решениями, которые можно определить как скорости изменения потоков, т.е. *темпы*. *Темп показывает, как изменяются уровни за интервал времени, равный шагу моделирования. Темп характеризует динамику моделируемой системы* (попробуйте остановить систему: уровни будут значимы, а темпы неразличимы).

Потоковые сети являются *неявной формой описания* состояний системы, *основных переменных модели* – переменных состояния и скоростей изменения состояний, – в форме разностных уравнений. Основные переменные модели: уровни, темпы, вспомогательные (или дополнительные) описываются с помощью следующих *разностных уравнений*:

*Уровень* – переменная, закон изменения которой во времени выражается конечно-разностными уравнениями:

$$X(t+h) = x(t) + h * V(t), \text{ где} \quad (4.5.3)$$

$t$  – модельное (системное) время;

$h$  – изменение (приращение) времени – шаг моделирования (интегрирования);

$x(t), x(t+h)$  – значение уровня в моменты времени;

$V(t)$  – скорость изменения уровня, т.е. величина его изменения за единицу времени.

Уровнями имитируют такие характеристики моделируемой системы, которые определяют ее состояние в конкретный момент времени.

*Темп* – это переменная, равная скорости изменения уровня. В (4.5.3)  $V(t)$  является темпом. Закон изменения темпа задается функциональной зависимостью:

$$V(t) = F(p_1(t), p_2(t), \dots, p_k(t)), \text{ где} \quad (4.5.4)$$

$t$  – модельное (системное) время;

$V(t)$  – темп на момент времени  $t$ ;

$F$  – произвольная функция от  $k$  – аргументов;

$p_i(t)$  – любые переменные (уровни, темпы, дополнительные переменные), значения которых в момент  $t$  известны.

Темп характеризует динамику моделируемой системы.

*Вспомогательные переменные* введены для описания сложных функциональных зависимостей, их можно использовать для более удобной записи уравнений темпов.

$$W(t) = F'(p_1(t) \dots p_k(t)), \text{ где} \quad (4.5.5)$$

$t$  – модельное (системное) время;

$W(t)$  – значение вспомогательной переменной на момент  $t$ ;

$p_i(t)$  – любые переменные, значения которых на момент  $t$  известны;

$F'$  – некоторая произвольная функция  $k$  – аргументов.

Алгоритм имитации, таким образом, реализуется на основе этих разностных уравнений следующим образом:

- Устанавливаются *параметры системного времени* (начальное значение, шаг интегрирования, длина интервала моделирования), задаются *начальные условия* (значения уровней в начальный момент системного времени);
- По уравнениям (4.5.4), (4.5.5) на данный момент системного времени  $t$  рассчитываются значения всех темпов и вспомогательных переменных;
- $t + h$ , системное время увеличивается на шаг моделирования (интегрирования);
- по уравнениям (4.5.3) рассчитываются значения всех уровней на данный момент системного времени;
- и т.д. выполняются итерации по этой схеме, пока не пройдет весь интервал моделирования.

**Информационная сеть.** Вернемся к общей структурной схеме моделей системной динамики, ко II составляющей структурных схем моделей системной динамики – *информационной сети*.

**Структуризация функциональных зависимостей  $f$  и  $H$  завершается построением информационной сети модели.** Подобно структурированным описаниям вектор-функций дифференциальных моделей, рассмотренных вначале, описания информационных сетей в моделях системной динамики представляют собой *многоярусные графы*. Ясно, что такие графы легко построить, если уравнения модели уже известны. Для сложных слабоструктурированных систем такие уравнения, как правило, не известны. *В моделях системной динамики большая роль отводится эксперту. Модели создаются путем структуризации экспертной информации.* Поэтому используется подход, основанный на анализе цепочек причинно-следственных связей факторов, подлежащих, по мнению экспертов, обязательному учету при описании динамики состояний моделируемого объекта. Вспомогательные переменные главным образом используются для построения логически ясных, хорошо интерпретируемых структур взаимосвязей переменных, с помощью которых в моделях отображаются представляемые экспертами разнородные сведения об объекте моделирования.

**Рассмотрим общую схему структуризации информации о причинно-следственных взаимосвязях динамических процессов в объектах моделирования:**

- модели системной динамики разрабатываются на основе представляемых экспертами сведений об объектах исследования в форме вербального описания;

- эксперты проводят анализ и выявление всех факторов, необходимых для описания динамики состояний моделируемого объекта (выбираются и интерпретируются переменные состояния модели). При построении моделей проводится тщательный анализ этих сведений;
- выявляются причинно-следственные отношения между переменными, для этого детально описываются цепочки причинно-следственных связей между факторами, отображаемыми в модели с помощью переменных состояний;
- описываются эти переменные в структурированном виде с помощью графовых представлений (это требует введения вспомогательных переменных);
- результатом являются структурированные описания вектор-функций правых частей уравнения состояний.

Таким образом, в моделях системной динамики реализуется **удобная и простая схема сбора и формализации информации, получаемой от эксперта в процессе построения моделей**. Модели системной динамики используют графическую технику при структуризации экспертной информации о проблеме.

#### **4.5.3 Диаграммы причинно-следственных связей.**

Уточним, какие способы описания структурированного экспертного знания предлагает концепция системной динамики.

**Диаграмма причинно-следственных связей (ДПСС).** *Диаграмма причинно-следственных связей – это эффективный способ структуризации экспертной информации, который демонстрирует простые приемы качественного описания взаимосвязей факторов, учет которых признается необходимым для отображения в модели принципиальных моментов развития моделируемых процессов.*

Диаграмма причинно-следственных связей достаточно хорошо известна как схема конструирования математических моделей. (Этот подход давно применялся при конструировании моделей, получил и самостоятельное развитие при структуризации моделируемых объектов вне метода системной динамики).

На дугах графов причинно-следственных диаграмм обычно расставляются знаки "+" и "-" (отсюда название – "знаковые оргграфы"), с помощью которых фиксируется предполагаемый или же эмпирически обоснованный характер влияния переменных, соответствующих конечным вершинам дуг. Наличие знака "+" на дуге ( $a, b$ ), направленной из вершины  $a$  графа в вершину  $b$ , означает, что при прочих равных условиях рост переменной  $a$  приводит к росту переменной  $b$ . Знак "-" описывает противоположный эффект влияния  $a$  на  $b$ : с ростом  $a$  величина  $b$  убывает. Проставив веса на дугах такого графа – мы получаем

взвешенные оргграфы, а произвольную функцию — функциональные оргграфы. Очевидно, диаграммы причинно-следственных связей являются предварительным способом анализа сложной системы

Технология составления диаграммы причинно-следственных связей следующая. На основе вербального описания моделируемых процессов выделяют фазовые переменные; используя логику описания, их попарно классифицируют по критерию "причина-следствие", причина со следствием соединяется стрелками, — таким образом, выявляются все причинно-следственные отношения; над стрелкой ставятся знаки: "плюс — минус" — факторы; в результате такой процедуры могут быть обозначены контуры с обратной связью.

*Такая схема структуризации информации о причинно-следственных взаимосвязях динамических процессов в объектах моделирования — схема формирования моделей на базе "плюс — минус" — факторов называется **знаковым оргграфом**, известным также под названием **причинно-следственной диаграммы**. Она позволяет получить общую структуру дифференциальной модели системной динамики.*

Построению и анализу знаковых оргграфов при разработке моделей системной динамики уделяется большое внимание.

#### **4.5.4 Системные потоковые диаграммы моделей.**

Рассматривая характерные особенности моделей системной динамики, мы неоднократно обращались к *различным графовым представлениям*: сетям потоков, информационным сетям, причинно-следственным диаграммам. Для дальнейшего изложения важно заметить, что:

- сети потоков и ярусные информационные сети описывают структуру уравнений моделей системной динамики только по частям;
- а причинно-следственные диаграммы, напротив, отображают взаимосвязи переменных моделей только в целом, не разделяя их по типам.

Сеть потоков — является неявной формой (в виде разностных уравнений), а сеть информации — явной формой описания одних и тех же переменных. Все они предлагают формы *графового описания* зависимостей переменных состояния моделируемых систем.

Разработчики дифференциальных моделей системной динамики используют *особую технику графического описания структур моделируемых систем*. Основу этой техники составляет применение для описания структур моделей так называемых *системных потоковых диаграмм*.

**В потоковую диаграмму объединяются графовые конструкции сетей потоков и информации, в результате чего обеспечивается целостность представления структуры уравнений (4.5.1) и (4.5.2).**

В потоковой диаграмме используются нормативные структуры потоковых и информационных сетей.



Популярность методов системной динамики обусловлена во многом тем, что для описания структуры модели *используются графические формы* представления информации, получаемой от экспертов в ходе сбора и анализа сведений о моделируемых процессах, — предлагается особая, графическая *техника структуризации информации*, полезная в любой технологии системного моделирования.

Модели системной динамики используют особую технику графического описания структур моделируемых систем: системные потоковые диаграммы. Специальная техника графического представления предлагает *развитую графическую символику диаграмм*: оснащение графов — (сетей потоков) и ярусных графов функциональных зависимостей темпов — (сетей информации) специальной выразительной символикой. Основные символы, их назначение и условия использования представлены в таблице 4.5.

Отметим важность языковой функции потоковых диаграмм как технологического средства структуризации информации. При использовании описанной символики рассматриваемые графы превращают потоковые диаграммы в *средство наглядного отображения информации* о динамике моделируемых процессов или, другими словами, — в *язык общения экспертов по проблеме* и системных аналитиков. Системные потоковые диаграммы доступны и наглядны, это делает их удобным средством для проведения совместных экспертных ревизий. Совмещение в конструкции потоковых диаграмм явной (сеть информации) и неявной (сеть потоков) форм графового описания зависимостей переменных состояния моделируемых систем, а также развитая графическая символика диаграмм приводят к тому, что потоковые диаграммы дают значительную часть той же информации, что и системы уравнений модели (4.5.1) и (4.5.2), но в иной, более наглядной форме.

Построение потоковых диаграмм оказывается непосредственно связанным с решением задач предмодельного анализа исследуемой проблемы, служит своего рода итогом исходной содержательной проработки информационной базы процесса моделирования. *Методы системной динамики предлагают эффективный способ структуризации знаний эксперта.*

Системные потоковые диаграммы — эффективное средство системного анализа, позволяет осуществлять декомпозицию сложной системы с последующей композицией. Такой язык определяет форму выражения обсуждаемых вопросов, выступает в качестве *средства разделения на части* задач анализа причинно-следственной структуры моделируемой системы и последующей "сборки" их результатов в *целостную картину* организации процессов развития системы.

По-видимому, без большого преувеличения можно сказать, что концепция потоковой стратификации систем, на которой базируются

методы системной динамики, без языка потоковых диаграмм вряд ли оказалась такой привлекательной и популярной для специалистов, применявших и применяющих системную динамику в различных областях исследования сложных систем.

Таблица 4.5 – Основные символы потоковых диаграмм моделей системной динамики.

Название	Условное обозначение		Назначение и условия использования
	буквенное	графическое	
Озеро	0		Нулевой узел потоковых сетей. Обозначает истоки и стоки потоковой сети
Потоковая связь	1		Дуга потоковой сети диаграммы. Может соединять уровни с уровнями, истоками и стоками. Проходит через темп
Информационная связь	F		Дуга информационной сети диаграммы. Может соединять входы (параметры), вспомогательные переменные и уровни с темпами, вспомогательными переменными и выходами
Уровень	L		Узел потоковой сети диаграммы. Обозначает переменную состояния модели
Запаздывание	L		Уровень специального вида. Характеризуется временем и порядком. Темп выходного потока определяется значением уровня, временем и порядком запаздывания
Темп	R		Обозначает скорость потока, проходящего по соответствующей дуге потоковой сети. Единица измерения темпа равна единице измерения уровня, деленной на время. Темпы не зависят непосредственно друг от друга
Вспомогательная переменная	A		Если идентификатор переменной заключен в дефисы сверху и снизу, то это означает, что переменная находится с помощью табличной функции
Выход (дополнительная переменная)	S		Переменные модели, характер изменения которых во времени интересует исследователя
Вход	C		Экзогенная переменная модели либо константа

*Нормативные схемы формирования общей структуры моделей.* В рамках рассмотренной концепции системная динамика предлагает две нормативные схемы формирования общей структуры моделей:

*Схема 1.* Сначала разрабатывается *причинно-следственная диаграмма модели*. В число учитываемых при разработке модели факторов и связей включаются все те из них, которые используются экспертами при содержательном описании моделируемого объекта. Затем выполняется анализ зафиксированных в разработанной диаграмме цепочек причинно-следственных связей и определяются факторы, которые описываются в модели *уровнями и темпами.*, т.е. выделяются переменные уровней и темпов. В результате формируется, прорисовывается на эскизах *сеть потоков модели*.

А далее выделяется и *уточняется* в качестве структуры, дополняющей сеть потоков в причинно-следственной диаграмме, *информационная сеть модели*.

*Схема 2.* Сначала выделяется множество основных материальных ингредиентов, динамику которых необходимо отобразить в модели. Для каждой выделенной совокупности однородных элементов определяется множество их возможных состояний и устанавливается структура переходов элементов ингредиентов из состояния в состояние. В результате *формируется сеть потоков модели*.

А затем устанавливается структура причинно-следственных связей между уровнями и темпами сети потоков, т. е. *разрабатывается структура информационной сети модели*. При таком подходе с помощью информационной сети "как бы" сшиваются потоковые представления.

Обе нормативные схемы являются лишь общими правилами структуризации в рамках единой концепции системной динамики и применяются в зависимости от класса решаемых задач.

#### 4.5.5 Основные этапы технологии системной динамики.

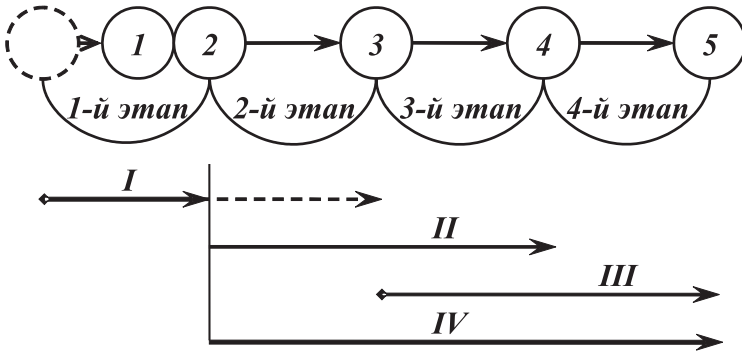


Рисунок 4.5.5 – Основные этапы технологии системной динамики:

*I* – концептуализация; *II* – структуризация; *III* – параметризация; *IV* – формализация;

*1* – вербальная модель; *2* – знаковый оргграф модели; *3* – потоковая диаграмма модели;

*4* – дифференциальные (разностные) уравнения модели (составление уравнений темпов);

*5* – машинная модель.

При нормативном подходе к разработке динамических моделей предполагается систематизация форм причинно-следственных описаний моделируемых процессов и соблюдение определенного порядка построения этих описаний. Основные этапы технологии моделирования в рамках рассмотренных нами схем поясняются на рисунке 4.5.5.

Рассмотрим, каким образом могут быть использованы рассмотренные средства структуризации моделей на первых этапах технологии моделирования.

*1 этап: Концептуализация проблемной ситуации.* Начало разработки модели системной динамики обычно определяют как этап построения **“вербальной модели”** исследуемой проблемной ситуации. Такой этап работы, безусловно, входит в любую технологию моделирования. В процессе построения вербальной модели осуществляются постановка проблемы, анализ исходной информации, формулировка целей моделирования и т. п. Этот этап, пожалуй, наиболее ответственный и сложный. Успех его выполнения во многом зависит от уровня подготовки и опыта системных аналитиков, участвующих в создании модели.

Составление вербального описания предполагает систематизацию причинно-следственных описаний моделируемых динамических процессов. Вербальное описание может содержать эскизы потоковых диаграмм и диаграммы причинно-следственных связей. При составлении вербального описания необходимо выполнить:

- анализ исходной информации, выявляемой в ходе интенсивных дискуссий с экспертами и специалистами;
- постановку проблемы, формулировку целей моделирования;
- формулирование наиболее принципиальных гипотез, которые впоследствии должны найти отражение в модели;
- выявление границ моделируемой системы (исходя из принципа замкнутости);
- детальное обоснование и установление основной структуры модели (состава взаимодействующих компонентов, динамика которых определяет наиболее существенные (важные) аспекты поведения и состояния системы);
- обсуждение воздействия на систему внешних факторов;
- выявление основных факторов и процессов, отображение которых является обязательным для достижения поставленной цели моделирования;
- описание всей структуры отображаемых в модели причинно-следственных взаимосвязей между факторами в форме вербального описания.

При разработке вербальной модели должны быть также выявлены:

- альтернативы (основные ситуации, варианты, стратегии), экспериментальное исследование которых предполагается проводить с помощью имитационной модели;
- критерии оценки поведения модели;
- временные параметры имитации (шаг интегрирования, время моделирования).

В результате выполнения первого этапа должно быть составлено вербальное описание, сформулированное в виде четких словесных конструкций, содержащее предварительное описание всей структуры отображаемых в модели причинных взаимосвязей, и зафиксированное специальными диаграммными представлениями, например, с помощью эскизов диаграмм потоков и/или диаграммы причинно-следственных связей.

*2 этап: Построение системных потоковых диаграмм.* Этап разработки потоковой диаграммы является основным при структуризации модели

системной динамики. Нормативное содержание данного этапа – переход от причинно-следственной диаграммы разрабатываемой модели к ее потоковой диаграмме. Такой переход в соответствии с 1 нормативным подходом связан с выделением вершин (и дуг) орграфа причинно-следственной диаграммы в соответствии с основными типами переменных и аксиомами системной динамики. Выполнение аксиом обеспечивает в последующем разработку и алгоритмизацию по диаграмме дифференциальных (разностных) уравнений модели.

В целом рассматриваемый переход неформален и, как правило, опирается на содержательные суждения о характере и причинах взаимодействия факторов, представленных в виде переменных модели.

*3 этап: Параметризация модели.* Предположим, что в нашем распоряжении уже имеется потоковая диаграмма модели, на которой указана структура основных причинных зависимостей темпов модели от уровней. Параметризация модели представляет собой процесс перевода вербальных описаний взаимозависимостей факторов моделируемой проблемной ситуации на язык четких количественных соотношений.

Системная динамика представляет разработчикам общие приемы, использование которых, как правило, облегчает и упрощает выбор и обоснование производящих функций темпов. Рассмотрим здесь *два основных приема*.

Первый из них основан на том, что *темпы потоков можно рассматривать в качестве функций принятия решений*. Использование такой концепции особенно удобно и естественно при моделировании производственных и экономических систем, когда производящие функции темпов фактически представляют собой количественные описания *решающих правил*, действующих в механизме управления системой.

При разработке, с этой точки зрения, рекомендуется выделять и в явном виде отображать в производящих функциях темпов следующие структурные элементы:

- желаемое состояние потокового сектора, в котором действует определенный темп (задание цели решающего правила);
- существующее (текущее) состояние сектора;
- количественное выражение различия между указанными состояниями потокового сектора;
- соотношение для выработки корректирующего воздействия на темп, которое обеспечивает перевод сектора в желаемое состояние.

Например, в производственной системе рассогласование между сетью материалов и сетью оборудования такой системы, связанные с отклонениями от требуемой экономической программы, позволяют в

информационной сети модели формировать соответствующие регуляторы для достижения требуемого состояния производственной системы.

Второй методический прием, широко используемый при разработке уравнений темпов моделей системной динамики, — это способ задания производящей функции темпа в виде произведения “нормального темпа” и корректирующих множителей, определяющих его зависимость от переменных состояния (уровней) модели.

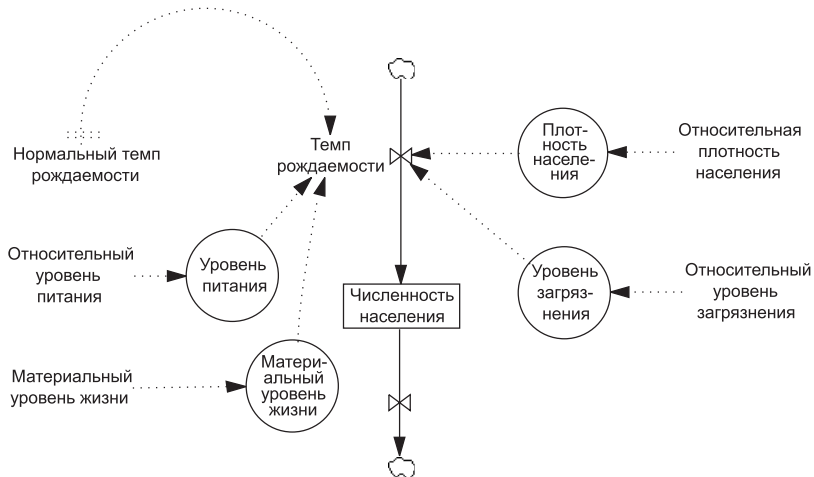


Рисунок 4.5.6 — Фрагмент потоковой диаграммы модели мировой динамики.

Рассмотрим фрагмент потоковой диаграммы (рис. 4.5.6) модели мировой динамики Дж. Форрестера [52]. Темп рождаемости населения определяется здесь как произведение численности населения (состояние), нормального темпа рождаемости и следующих сомножителей, отображающих зависимости темпа рождаемости от:

- материального уровня жизни;
- плотности населения;
- уровня питания;
- уровня загрязнения.

Каждый из четырех перечисленных выше множителей представляет собой нелинейную функцию, вид которой отражает реальные совокупности данных о характере описываемой причинной связи, или же задает экспертную оценку (гипотезу) такой связи.

Использование описанной формы уравнений темпов обеспечивает наглядность и простоту их содержательной интерпретации, является удобным средством для экспертной оценки информации о наиболее

трудно формализуемых аспектах причинных связей компонентных процессов моделируемых систем.

Важным обстоятельством, способствующим преемственности различных форм конструирования моделей при использовании рассмотренных нами приемов параметризации моделей, является четкое соответствие структур уравнений темпов строению информационной сети потоковой диаграммы модели системной динамики. Например, структура представления производящих функций темпов в виде произведений нормальных темпов и корректирующих множителей непосредственно соотносится структуре причинно-следственных зависимостей, заданных диаграммами разрабатываемой модели. Каждый из корректирующих множителей ставится в соответствие дуге информационной сети потоковой диаграммы модели. При этом вид (возрастающая или убывающая функция) будет отвечать знаку, определенному для данной дуги в причинно-следственном графе модели. Таким образом, используя второй из рассмотренных приемов, специалист по системной динамике на основе знакового орграфа модели может записать общее выражение для любого темпа модели и качественно охарактеризовать вид образующих его функциональных сомножителей.

Сегодня существует большое количество систем моделирования, таких как Vensim, iThink, Powersim и др., поддерживающих нормативные схемы системной динамики, предлагающих *эффективные инструментальные средства программной поддержки техники, процедур и методов системной динамики*. С помощью этих систем системные потоковые диаграммы создаются на идеографическом уровне, параметризация модели осуществляется в режиме подменю, с использованием средств ввода формульной, табличной и графической информации, в процессе диалогового взаимодействия разработчиков модели и системы моделирования, поддерживающей методы системной динамики. Системы моделирования обеспечивают прием задаваемых экспертами спецификаций указанных стандартных описаний, их контроль на непротиворечивость, преобразование полученной информации в текст на языке моделирования. Такие системы моделирования имеют развитые средства для анализа результатов вычислительных экспериментов и проведения сценарных расчетов.

*Основные особенности моделей экономической динамики.* Модели системной динамики широко применяются в моделировании экономической динамики. На практике используются знаковые графы в моделировании сценариев развития макроэкономических процессов, выборе вариантов экономической политики. Модели системной динамики



в сочетании с балансовыми методами находят широкое применение в моделировании социально-экономических процессов, в моделях ресурсного типа, при исследовании процессов воспроизводства в региональных, макроэкономических системах [16,17,23,28].

Исследуемые с помощью этих методов *задачи являются слабоструктурированным*; отсутствие теоретических знаний, *качественный характер знаний* о системе с *большой долей экспертных знаний* не позволяет применять точные нормативные модели. При исследовании такого класса систем присутствует низкий уровень точности исходных данных, *внешняя и внутренняя неопределенность*, связанная с присутствием большого количества факторов, находящихся под слабым контролем лиц, принимающих решения.

Особенностью динамического моделирования является то, что *решения здесь носят качественный характер*, по результатам моделирования в основном судят о направлении развития динамических процессов, проводят анализ устойчивости динамических процессов: исследуется устойчивость или скачкообразность, степень энтропии процессов, протекающих во времени.

**ЛЕКЦИЯ**  
**5****ИНСТРУМЕНТАЛЬНЫЕ  
СРЕДСТВА АВТОМАТИЗАЦИИ  
МОДЕЛИРОВАНИЯ.****5.1 Назначение языков и систем моделирования.**

В процессе построения имитационной модели мы выделяли три уровня ее представления: *концептуальную модель, формализованное или алгоритмическое описание, программу-имитатор.*

Формальное или концептуальное описание модели, преобразуется в программу-имитатор в соответствии с технологией программирования. В принципе, имитационную модель можно реализовать на любом универсальном языке моделирования. Однако для облегчения написания и работы с программой-имитатором созданы специальные системы автоматизации моделирования.

**Языки и системы моделирования упрощают построение программ-имитаторов и проведение имитационных экспериментов за счет частичной или полной автоматизации переходов от одного уровня представления модели к другому.** В этом состоит *основное назначение языков моделирования*, именно здесь и проявляется их главное преимущество перед универсальными алгоритмическими языками.

Общепризнанными являются следующие *преимущества языков и систем моделирования* по сравнению с универсальными языками и системами программирования:

**Концептуальная выразительность.** Языки моделирования обеспечивают более строгое следование выбранной концепции построения модели.

Язык моделирования содержит абстрактные конструкции, непосредственно отражающие понятия, в которых представлена формализованная модель, или близкие концептуальному уровню описания моделируемой системы, с помощью которых четко классифицируют элементы моделируемой системы, элементы различных классов различают по характеристикам и свойствам, описываются связи между элементами системы и внешней среды, позволяющие изменять структуру модели.

Это упрощает программирование программы-имитатора, позволяет автоматизировать выявление, диагностику ошибок в программах;

**Автоматизация стандартных функций моделирования** (функций управляющей программы):

реализация механизма модельного времени — системы моделирования имеют эффективный встроенный *механизм продвижения*

моделного времени (календарь событий, методы интегрирования и др.), средства разрешения временных узлов;

- языки моделирования, как правило, содержат встроенные датчики случайных чисел, *генераторы случайных чисел* и других типовых воздействий;
- в языках моделирования автоматизирован сбор *стандартной статистики* и других результатов моделирования. Имеются средства автоматизации выдачи этих результатов в табличной или графической форме;
- *управление процессом моделирования* (анализ ошибочных ситуаций и т.д.)
- языки моделирования имеют средства, упрощающие *программирование имитационных экспериментов* (в частности, автоматизирующие установку начального состояния и перезапуск модели.) и другие интерактивные и технологические возможности, используемые при проведении имитационных исследований;
- и др.

Вместе с тем пользователи нередко отмечают такие недостатки языков и систем моделирования, как недостаточная распространенность языков и систем моделирования, необходимость дополнительного обучения языкам и систем моделирования и, как следствие, недостаток программистов, хорошо владеющих современными языками и системами моделирования; слабые технологические возможности некоторых систем моделирования; высокая стоимость систем моделирования; отсутствие гибкости и широких возможностей, присущих универсальным языкам программирования.

Множество языков моделирования можно разделить на две группы:

1) **методо-ориентированные** языки моделирования, поддерживающие определенный класс формализованных или алгоритмических описаний;

2) **проблемно-ориентированные** языки моделирования — языки моделирования конечного пользователя, позволяющие формулировать задачи моделирования непосредственно на концептуальном уровне. Связь с пользователем в такой системе моделирования на уровне программного интерфейса осуществляется через набор понятий непосредственно из предметной области исследований. Для этого в проблемно-ориентированные системы моделирования включаются абстрактные элементы, языковые конструкции и наборы понятий, взятые непосредственно из предметной области исследований. Примерами таких решений могут служить системы моделирования:

Simular, Simflex — управление материальными потоками в производственной системе;

- MAST- моделирование гибких производственных систем (применяется блочная концепция структуризации);
- TOMAS — технологическая подготовка производственных систем (используемые формальные схемы — автоматы);
- SIRE — календарное планирование производственных процессов (сети с очередями);
- COMNET — телекоммуникации;
- MEDMODEL — медицинское обслуживание.

За годы становления технологии имитационного моделирования наблюдалось большое разнообразие специализированных языков моделирования. В известном в 80-е годы обзоре Киндлера [19] упоминалось более 500 языков и систем моделирования.

Рост числа языков моделирования свидетельствует о необходимости использования средств автоматизации моделирования. Вместе с тем *разнообразие языков моделирования обусловлено* следующими факторами: существует большое число схем формализации и алгоритмизации моделируемых систем: агрегаты, сети, автоматы, процессы, системы массового обслуживания, дифференциальные уравнения, аналоговые блок-схемы, графы связей и др. Любая из этих схем может служить основой для разработки нового языка моделирования. Еще большее разнообразие возможно на уровне концептуальных моделей. Все это — существенный стимул для появления новых языков.

Далее рассматриваются только методо-ориентированные языки и системы, которые для краткости называются просто языками и системами моделирования.

Язык моделирования предоставляется пользователем как часть системы моделирования. **Система моделирования — это совокупность языковых и программных средств, которая включает:**

- собственно язык моделирования;
- язык управления системой моделирования — язык команд интерактивного взаимодействия с пользователем;
- управляющая программа — программные средства, обеспечивающие трансляцию модели и другие стандартные функции системы моделирования (продвижение модельного времени, генерацию случайных чисел, сбор статистической информации, вывод результатов и т. д.).

Системы моделирования проблемно-ориентированные включают также *средства разработки языков конечного пользователя.*

Среди большого числа языков моделирования довольно сложно выделить какое-то базовое подмножество языков, покрывающих основные потребности пользователей в средствах автоматизации моделирования. На

практике существует проблема выбора системы моделирования, подходящей для поставленной задачи. С этой целью, в дальнейшем рассмотрим основные характеристики систем моделирования.

## **5.2 Классификация языков и систем моделирования, их основные характеристики.**

Классификация языков и систем моделирования позволяет упорядочить сведения о существующих многочисленных языках моделирования, а также более обоснованно выбирать подходящую систему моделирования.

В основу классификации положим следующие общепризнанные характеристики языков и систем моделирования:

- **Класс моделируемых систем;**
- **Средства описания моделируемых систем;**
- **Инструментально-технологические возможности систем моделирования.**

Принято выделять *три класса моделируемых объектов: дискретные, непрерывные, дискретно-непрерывные (комбинированные)*. Соответственно различают три класса языков моделирования. Области применения языков непрерывного и дискретного моделирования практически не пересекаются. Языки комбинированного моделирования используются при работе с моделями объектов и непрерывного, и дискретного класса. Примерами классических языков и систем непрерывного типа являются: DYNAMO, Vensim и др., поддерживающие методы системной динамики. Широко известные в свое время языки непрерывно-дискретного моделирования: GASP, SLAM, языки дискретного моделирования: SMPL, SIMULA, GPSS, АИС (агрегативная имитационная система) и др.

Средства описания моделируемых систем включают базовую для языка моделирования схему алгоритмизации; альтернативные схемы алгоритмизации; синтаксическую основу языка моделирования; средства проблемной ориентации в языке моделирования

Под *базовой схемой алгоритмизации* подразумевается совокупность понятий, которые используются для алгоритмизации (формализации) моделируемой системы и непосредственно представлены в языке моделирования. В настоящее время используется большое число различных способов алгоритмизации. В языках непрерывного моделирования широко применяются системы дифференциальных и интегро-дифференциальных уравнений, структурные схемы, графы связей. Среди дискретного моделирования различают: языки событий, языки работ, языки процессов, языки транзактов, языки, основанные на автоматных и сетевых представлениях, и др. Языки комбинированного моделирования могут

основываться на агрегатах, КОМБИ-сетях, использовать комбинированные схемы алгоритмизации языков непрерывного и дискретного моделирования. Базовая схема алгоритмизации предполагает определенную организацию работы управляющей системы моделирования.

В качестве доминирующих базовых концепций формализации и структуризации в современных системах моделирования используются:

- для дискретного моделирования – системы, основанные на описании процессов (process description) или на сетевых концептах (network paradigms), – (Extend, Arena, ProModel, Witness, Taylor, Gpss/H-Proof, ARIS и др.);
- для систем, ориентированных на непрерывное моделирование – модели и методы системной динамики, – (Powersim, Vensim, Dynamo, Stella, Ithink и др.).

Кроме базовой схемы алгоритмизации при построении имитационной модели могут быть применены *альтернативные схемы*, близкие к базовой, либо приводимые к ней. Например, язык структурных схем можно использовать для моделирования систем дифференциальных уравнений, дискретные сети можно моделировать на языке транзактов или процессов. Как правило, в современных системах, с целью расширения их функциональности присутствуют альтернативные концепции формализации. Так, например, в системах Powersim, Ithink встроены аппарат дискретного моделирования, и, наоборот, в системах Extend, ProcessModel реализована поддержка, правда, довольно слабая, непрерывного моделирования. Часто проще и целесообразнее разработать методику программирования для выбранной схемы алгоритмизации, чем найти язык, основанный на этой схеме. Таким образом, один и тот же язык моделирования можно применять для реализации различных способов алгоритмизации моделей.

Актуальной задачей сегодня является разработка систем моделирования, интегрирующих различные подходы в имитационном моделировании. Примером является система моделирования AnyLogic, совмещающая различные подходы в описании динамических процессов: транзактно-ориентированный способ описания дискретных систем, динамическое моделирование, диаграммы состояний универсального языка моделирования UML, агентный подход в моделировании.

*По синтаксической основе* различают языки моделирования: вложенные в базовый язык моделирования или расширяющие его, а также с собственным синтаксисом. Классическим языком с собственным синтаксисом является язык моделирования GPSS.

Наличие *средств проблемной ориентации* в языке моделирования позволяет разрабатывать языки конечных пользователей, вводить макропонятия для упрощения программирования. Средства проблемной

ориентации могут быть реализованы различными способами. Весьма распространенный способ — использование макросов. В универсальном языке моделирования SIMULA для проблемной ориентации языка служат классы. В процедурно-ориентированных языках могут использоваться подпрограммы или процедуры.

Далее рассмотрим 3 группу характеристик: *инструментально-технологические возможности систем моделирования.*

### **5.3 Технологические возможности систем моделирования.**

Технология системного моделирования — основа целенаправленной деятельности, смысл которой в обеспечении возможности эффективного выполнения на ЭВМ исследований функционирования сложной системы. Имитационная модель — специфическое, сложное программное изделие, ее разработка должна вестись с применением высокотехнологичных систем моделирования. Действия исследователя организуются на всех этапах имитационного моделирования, начиная с изучения предметной области и выделения моделируемой проблемной ситуации и кончая построением и реализацией планов машинных экспериментов и обработкой результатов.

Технология системного моделирования объединяет современные методы и средства, используемые на всех этапах имитационного моделирования, и включает современные способы формализации моделируемых процессов, средства программирования и трассировки моделей, технологию проведения испытаний, поддержку имитационного эксперимента. Начальные этапы имитационного моделирования трудно поддаются автоматизации. Самая большая степень автоматизации традиционно приходится на этап программирования (и этап эксплуатации) имитационной модели. Принципиальным недостатком систем моделирования прошлых лет было то, что автоматизации подлежал только этап программирования имитационной модели. Технология имитационного моделирования должна охватывать весь цикл системного моделирования, от постановки проблемы и формирования концептуальной модели — до анализа результатов вычислительного эксперимента и принятия решения. Актуально, для современного уровня развития технологии системного моделирования автоматизировать этап проведения испытаний имитационной модели и обработки результатов эксперимента.

Опыт эксплуатации различных систем моделирования показывает, что функциональные свойства любой системы моделирования определяется наличием развитых средств, поддерживающих реализацию следующих основных функций:

- Подготовка, редактирование и модификация в ходе трассировки программы-имитатора и основных модельных характеристик;
- Интерактивное взаимодействие системы с исследователем в процессе имитации;
- Управление направленным вычислительным экспериментом на имитационной модели.

Рассмотрим с помощью, каких инструментов реализуются в системах моделирования эти основные функции.

*Подготовка, редактирование и модификация в ходе трассировки программы-имитатора и основных модельных характеристик.* Для поддержки этих функций в системе моделирования, как правило, представлены:

- поддержка создания формализованных описаний;
- создание модели (языки плюс графический интерфейс);
- развитые средства визуализации и редактирования программы и модельных характеристик;
- эффективные средства трансляции, компилирования программы-имитатора;
- средства трассировки (отладки) и диагностики: выдача сообщений в процессе составления модели и по ходу имитации;
- трассировка (в реальном времени), верификация по ходу имитации, ручная имитация;
- развитые средства визуализации модельных характеристик по ходу и по окончании прогона;
- средства сбора и выдачи результатов моделирования, возможность управления ими. Присутствуют как стандартные формы вывода выходной статистики, так и средства для создания нестандартных средств обработки и вывода результатов моделирования. (Как правило, все языки моделирования имеют стандартные средства сбора и вывода результатов моделирования в табличной или графической форме. Эти средства не всегда достаточно гибки и не во всех случаях удовлетворяют пользователя. Поэтому должна быть предусмотрена возможность подключения нестандартных средств обработки и вывода результатов моделирования, разработанных пользователем);
- передача выходных данных в другие среды (в том числе в базы данных);
- средства для анализа входных данных (Input Analyzer);
- средства для реализации арифметических, математических, логических функций, расширяющие алгоритмические возможности языка моделирования;



- и другие.

*Интерактивное взаимодействие системы с исследователем в процессе имитации:*

- интерактивные средства взаимодействия с пользователем: удобный графический интерфейс, выдача стандартной и формируемой пользователем выходной статистики и др. модельных характеристик в том числе по ходу имитации (в реальном времени) и в процессе трассировки;
- возможность доступа пользователя к модельным характеристикам, доступ к графикам, таблицам, модельным характеристикам с промежуточной статистикой,
- управление ходом имитации, управление прогоном (задание начальных условий, условий завершения имитационного прогона, возобновление имитации и др.) (Средства управления имитационными экспериментами позволяют автоматизировать установку начального состояния модели, определить условия завершения прогона, выбрать режимы работы управляющей программы, хранить и восстанавливать промежуточные состояния программы-имитатора, проводить анализ чувствительности. Наличие таких средств значительно упрощает проведение имитационных экспериментов с моделями);
- наличие специальных средств испытания имитационных программ;
- анимация (animation) (современные графические системы, позволяющие наблюдать за поведением "реальной" системы на модели);
- и т.п.

*Управление направленным вычислительным экспериментом на имитационной модели.* Для управления имитационным исследованием в системе моделирования должно присутствовать: развитое математическое и программное обеспечение (библиотеки, макросы), средства для организации статистических исследований, средства вторичного моделирования: файловые структуры, базы данных результатов моделирования, выход в другие программные среды или собственные инструменты для реализации и поддержки направленных вычислительных экспериментов.

Как правило, наборы этих инструментов ориентированы на:

- анализ чувствительности (проведение многократных прогонов с различными входными данными, регулировка параметров, а также сбор и обработка выходной статистики);
- статистическую поддержку моделирования (макросы, библиотеки и т.п.);

- накопление данных по серии прогонов и организация их в файловые структуры или базы данных, организации связи с системой управления базой данных;
- оптимизация (Output Analyzer – инструмент реализации направленного вычислительного эксперимента);
- транспорт данных в другие интеллектуальные среды.

## **5.4 Развитие технологии системного моделирования.**

Совершенствования технологических возможностей систем автоматизации моделирования, развитие технологии системного моделирования в историческом ракурсе шло по следующим основным направлениям:

- математическая и программная поддержка процесса имитационного моделирования;
- технология Data Base Simulation (использование баз данных в процессе имитационного исследования);
- создание машин имитационного моделирования (стохастические машины, распределенные системы моделирования, современные решения по этому направлению – это взаимодействие технологии имитационного моделирования со Всемирной паутиной);
- создание проблемно-ориентированных имитационных систем;
- появление и развитие концепции моделирующих центров [47] (технология решения сложных задач системного анализа на ЭВМ): создание единой, целостной многофункциональной системы программных средств автоматизации моделирования;
- работы по комплексной автоматизации моделирования в России были связаны с реализацией систем типа СПРИНТ, СПЛАВ, СПРУТ, САПФИР; были выдержаны в свое время на мейнфреймовском направлении и поэтому не получили широкого распространения;
- построения имитационных систем принятия решений [16,28,49]. Разработка информационных систем поддержки принятия решения осуществляется на основе синтеза новейших информационных технологий, таких как Хранилище данных, OLAP – технология, компьютерное моделирование и интеллектуальный анализ данных. Имитационные модели в таких системах выступают в качестве системообразующего и наиболее ценного звена в принятии решений.

Предпосылки совершенствования технологии системного моделирования были связаны, с одной стороны, – с общим развитием информационных технологий (графических оболочек, мультимедийных средств, объектно-ориентированного программирования и т.д.), а с другой

— с комплексным, многоаспектным исследованием сложных систем, таких как социально-экономические, производственно-технологические, созданием человеко-машинных систем поддержки принятия решения в различных областях научно-исследовательской деятельности. По данным обзоров, публикуемых в Internet, куда информация предоставляется компаниями — производителями программного обеспечения для имитационного моделирования, сегодня на рынке информационных технологий фигурирует более 100 программных продуктов аналитического типа, ориентированных на имитационное моделирование. Диапазон и разнообразие такого программного обеспечения продолжает расти, отражая тенденцию устойчивого спроса на него.

Сегодня имитационное моделирование становится все более зрелой компьютерной технологией. Кратко отметим основные направления совершенствования технологических возможностей современных систем моделирования:

- создание проблемно-ориентированных систем моделирования в различных областях исследований;
- наличие удобного и легко интерпретируемого графического интерфейса, когда блок-схемы дискретных моделей и системные потоковые диаграммы непрерывных реализуются на идеографическом уровне, параметры моделей определяются через подменю;
- объектно-ориентированное моделирование;
- использование развитой двух- и трех-мерной анимации в реальном времени;
- применение структурно-функционального подхода, многоуровневых иерархических, вложенных структур и других способов представления моделей на разных уровнях описания (см. понятие "стратифицированное описание");
- совершенствование инструментов для проведения сценарных расчетов;
- информационная (доступ к базам данных) и математическая поддержка процедур анализа входных данных, анализа чувствительности и широкого класса вычислительных процедур, связанных с планированием, организацией и проведением направленного вычислительного эксперимента на имитационной модели;
- применение интерактивного распределенного моделирования, разработки в области взаимодействия имитационного моделирования со Всемирной паутиной и др.

Остановимся на некоторых из перечисленных свойствах высокотехнологичных систем моделирования.

*Идеографический режим построения моделей.* Большинство систем моделирования имеют удобный, легко интерпретируемый графический интерфейс, системные потоковые диаграммы или блок-схемы реализуются на идеографическом уровне, т.е. рисуются, параметры моделей определяются через подменю (pull-down menus). Сохраняются элементы программирования (на языках общего назначения или объектно-ориентированных) для отдельных элементов модели или создания специализированных блоков подготовленным пользователем, так называемое авторское моделирование (например, в системе Extend существует встроенный язык Modl для создания специализированных блоков).

*Анимация.* Системы моделирования имеют развитые средства мультипликации (animation), подчас мультипликация весьма сложная, трехмерная с использованием стандарта в области виртуальной реальности языка VRML, *в реальном времени*, как, например, в TAYLOR (и может быть перенесена в другие среды – например, в форме апплета для презентаций). Имитационное моделирование – это “компьютерный экстрасенс”, который может создавать картины будущего.

*Проблемная ориентация.* Имитационные системы становятся все более проблемно-ориентированными. Известны системы моделирования производственных систем различного назначения (TOMAC, SIRE и др.), медицинского обслуживания (MEDMODEL), в области телекоммуникаций (COMNET) и др. Для этого в проблемно-ориентированные системы моделирования включаются абстрактные элементы, языковые конструкции и наборы понятий, взятые непосредственно из предметной области исследований. Определенные преимущества имеют системы моделирования, декларирующие свою проблемную ориентацию, например, пакет Rethink, ориентирующийся на моделирование бизнес-процессов, реинжиниринг. Все это, конечно, влияет на доступность и привлекательность имитационного моделирования.

*Стратификация.* В современных системах моделирования появляется некоторый инструментарий для создания стратифицированных моделей [21]. Стратификация систем, являясь общим принципом системного моделирования, реализуется в технологии имитационного моделирования либо путем детализации, итерационной процедуры эволюции имитационной модели, – либо путем создания комплекса взаимосвязанных моделей, с развитыми информационными и имплицитными связями между моделями. Стратифицированные модели представляют собой машинно-ориентированные понятия, предполагающие конструирование баз данных и знаний, над которыми определены вычислительные процессы решения задач системного анализа и принятия решения.

Разработчики систем моделирования используют различные подходы для реализации стратифицированных моделей. Ряд программных продуктов, такие как AUTOMOD, ProModel, TAYLOR, WITNESS и др. поддерживают интеграцию моделей на основе создания *вложенных структур*. В системах Arena, Extend реализован подход к стратификации, основанный на построении *иерархических многоуровневых структур*. Наиболее перспективным является *структурно-функциональный подход*, реализованный, например, в системах моделирования Ithink, Rethink, базирующийся на методологии структурного анализа и проектирования. При такой технологии есть возможность для реализации нескольких уровней представления моделей, — высоко-уровневое представление в виде блок-схем, с использованием CASE-средств, а на нижнем уровне модели могут отображаться, например, потоковыми схемами и диаграммами.

*Поддержка направленного вычислительного эксперимента на имитационной модели.* Новая методология научного исследования в компьютерном моделировании, предполагающая организацию и проведение вычислительного эксперимента на имитационной модели, требует серьезной математической и информационной поддержки процесса системного моделирования, особенно в части вычислительных процедур, связанных с планированием эксперимента, оптимизацией, организации работы с большим объемом данных в процедурах принятия решений. Имитационные модели в зависимости от решаемых задач применяются совместно с дифференциальными уравнениями балансового типа, в сочетании с принципами и методами логистики, основанными на оптимизации, управлении, интеграции потоков в сложных системах. Перспективно применение компьютерного моделирования в сочетании с другими методами принятия решений, интеллектуальными технологиями, экспертными процедурами [30], реализация имитационно-оптимизационных вычислительных процедур [13] на основе компенсационных подходов.

*Средства интеграции с другими средами.* Многие системы моделирования обеспечены средствами для интеграции с другими программными средами, осуществляют доступ к процедурным языкам, связанным с кодом имитационной модели, для реализации специальных вычислений (в электронную таблицу, например), доступа к базам данных. В более мощных пакетах осуществляется интеграция через дополнительное программное обеспечение (add-on software) со специализированными блоками различного назначения. Это могут быть блоки *анализа входных данных (моделирование входных потоков -Input Distribution Fitting)*, гибкие средства *анализа чувствительности* (контроль экспериментов, анализ результатов -*Output Analysis Support*), позволяющие осуществлять многократные прогоны с различными входными данными (в системах GPSS/H-PROOF, ProModel и др.). Перспективно создание систем

моделирования с функционально широкими, ориентированными на специфику имитационного моделирования, блоками *оптимизации* (факторные эксперименты, выбор оптимального варианта и т.п.). В этом смысле показательны системы WITNESS, TAYLOR, Arena.

Реализуемый в ряде систем *многопользовательский режим*, применение интерактивного *распределенного моделирования*, разработки в области *взаимодействия имитационного моделирования со Всемирной паутиной*, расширяют возможности и сферу применения имитационного моделирования, позволяя обрабатывать совместные или конкурирующие стратегии различным компаниям.

Более подробную информацию о современных системах моделирования можно найти в [32].

## 5.5 Выбор системы моделирования.

Анализ перечисленных характеристик позволяет составить достаточно полное представление о языке и поддерживающей его системе моделирования. При выборе системы и языка моделирования необходимо учитывать ряд факторов, определяющих выбор системы моделирования, подходящей для Вашего исследования, среди которых:

- *Область применения моделей.*

Необходимо прежде всего учитывать требования к моделям со стороны заказчика, определяемые условиями ее эксплуатации. Многое зависит от того, в контур каких информационных систем и проектов должна интегрироваться модель: СППР, САПР и т.п.

- *Пригодность языка моделирования для описания объекта моделирования.*

Уточните, насколько пригодно мировоззрение языка моделирования для описания объекта моделирования, насколько подходит концептуально тот способ описания динамических процессов, схема формализации, принятая в языке моделирования для идентификации реальных процессов, подлежащих моделированию.

Ваши рассуждения должны строиться примерно следующим образом: Материальные объекты движутся в сетях? Или это непрерывные информационные потоки? В рассматриваемой системе можно выделить агрегированное множество материальных объектов? Если так, то скорее всего Ваш выбор будет за непрерывной системой моделирования, поддерживающей методы системной динамики.

А если в моделируемой динамической системе наблюдается дискретный характер изменений, перемещаются единичные объекты и эти изменения проходят неравномерно, связаны с появлением событий – выбор, очевидно за системой дискретного типа. Если система характеризуется большим числом взаимодействующих процессов – надо

выбирать процессно- (транзактно) -ориентированные системы моделирования типа GPSS.

- *Прагматические соображения.*

Такие как, совместимость с другими программно-техническими платформами, наличие или отсутствие опыта работы с данной системой моделирования, технологические возможности системы моделирования, отвечающие требованиям настоящего проекта по системному моделированию, гибкость языковой среды, и многие другие соображения. Безусловно важной, но внешней по отношению к системе моделирования характеристикой является качество сопровождения системы моделирования, наличие четкой и полной документации, желательно на русском языке.

## ЛЕКЦИЯ

## 6

**ИСПЫТАНИЕ И  
ИССЛЕДОВАНИЕ СВОЙСТВ  
ИМИТАЦИОННОЙ МОДЕЛИ.****6.1 Комплексный подход к тестированию имитационной модели.**

После того, как имитационная модель реализована на ЭВМ, необходимо провести испытание, проверку модели. Это является чрезвычайно важным и ответственным моментом в имитационном моделировании. Проверка, выполненная не тщательно, может привести к неизвестным последствиям.

Поскольку моделирование связано с решением реальных задач, мы должны, прежде всего, быть уверены, что конечные результаты моделирования точно *отражают истинное положение вещей*, что модель не *абсурдна*, не дает нелепых ответов, оценить насколько модель и данные, полученные на ней, *полезны* и могут быть использованы при принятии решений, насколько точна разработанная модель.

Считается, что имитационные модели обладают высокой степенью изоморфизма (изоморфизм — сходство модели с объектом), т.к. существует взаимно однозначное соответствие между элементами модели и реального объекта моделирования, а также сохраняется характер взаимодействия между элементами (вы знаете, что имитационная модель призвана отражать структуру и внутренние связи моделируемой системы). Считается, что область пригодности модели тем обширнее, чем ближе структура модели к структуре системы и чем выше уровень детализации. Однако, большинство моделей все же гомоморфны. При построении любой модели используют упрощения, абстракции реальной системы, поэтому модель не является абсолютно точной в смысле однозначного соответствия её реальной системе. Кроме того, при описании системы, несмотря на наше стремление к объективности, действует субъективный фактор. Основной вопрос — насколько модель может быть в известной степени гомоморфной, и в то же время точной.

Таким образом, на этапе исследования имитационной модели мы должны *укрепить свое доверие к модели*, убедиться, что модель *функционально надежна*, оценить ее *достоверность*. Исследователь должен провести серию проверок, и в процессе проверки модели достигнуть приемлемого уровня уверенности, что выводы, сделанные на основе моделирования, будут правильными и применимыми для реальной системы



Это сложная, *философская* проблема. Моделирование является методом научного познания окружающей действительности. Исторически существует несколько направлений в науке об отношениях к методам научного познания: *эмпиризм, рационализм, абсолютный прагматизм*.

На одном конце философского спектра находится *эмпиризм*, критерий которого — практика. Эмпирик считает, что должна быть проведена эмпирическая проверка любой испытуемой гипотезы (с помощью эксперимента, или на основе анализа эмпирических данных).

*Рационализм* основывается на применении методов формальной логики. Приверженцы рационализма считают, что модель есть совокупность правил логической дедукции (типа "если, то"), которые могут привести от предпосылок к объективным выводам. Получается, что согласиться с правильностью модели, значит согласиться с основными предпосылками и логикой построения такой модели. (А вдруг исходные предпосылки необоснованны, ошибочны? Что, если мы моделируем всего лишь собственные мысли, имеющие мало отношения к реальной действительности? Такие вопросы всегда задает себе начинающий системный аналитик).

Третий философский подход — философия *абсолютного прагматизма*, у которого основной критерий — полезность, утверждает, что, модель должна с определенной точностью позволять достигать некоторых целей и давать полезные результаты, поэтому прагматика не интересуется внутренность модели — "черного ящика", его интересуют лишь соотношения между входами и выходами модели.

Но истина, как говорят философы, посередине. Поэтому должна быть произведена всесторонняя проверка пригодности модели, *комплексная оценка модели*, отражающая все философские точки зрения.

В мировой практике имитационного моделирования к настоящему времени сформировались определённые концепции и сложились вполне устоявшиеся подходы к решению проблемы оценки достоверности имитационных моделей. Следует заметить, что оценка достоверности относится к числу "вечных" проблем имитационного моделирования. Такое положение обусловлено прежде всего спецификой применения имитационного моделирования как инструментария исследования, который в отличие от классических методов математического моделирования не обеспечивает проектировщиков и исследователей сложных систем соответствующими формализованными средствами определения (описания) таких систем. Однако нелишне заметить, что простота реализации некоторых процедур исследования в имитационном моделировании, например, анализа чувствительности, делают метод имитационного моделирования привлекательным и доступным.

Исследованию различных аспектов проблемы оценки достоверности уделяли и уделяют много внимания известные учёные и специалисты в области имитационного моделирования Р.Шеннон [54], Н.П.Бусленко [11], Дж.Клейнен [20], В.В.Калашников [18], А.А.Вавилов, С.В.Емельянов [47], и др.

В середине 90-х годов прошлого века американские специалисты выполнили цикл работ применительно к проблеме оценки достоверности. Наиболее известными являются методологические схемы О.Балчи и Р.Сэджента [1-3,7] сформулированные как своего рода методологические стандарты решения указанной проблемы, согласно которым реализация задачи оценки достоверности есть многоэтапный итерационный процесс получения доказательства правильности и корректности выводов (или, по крайней мере, достижение приемлемого уровня уверенности в правильности таких выводов) относительно поведения исследуемой (проектируемой) системы. Вопросы практического использования этих стандартов рассматриваются в [4, 6, 56].

На практике выделяют 3 основные категории оценки:

- **Оценка адекватности или валидация модели.**

В общем случае валидации предполагает проверку соответствия между поведением имитационной модели и исследуемой реальной системы. *Валидация модели (validation)* есть подтверждение того, что модель в пределах рассматриваемой области приложений ведет себя с удовлетворительной точностью в соответствии с целями моделирования.

- **Верификация модели.**

Это проверка на соответствие поведения модели замыслу исследователя и моделирования. Т.е. процедуры верификации проводят, чтобы убедиться, что модель ведет себя так, как было задумано. Для этого реализуют формальные и неформальные исследования имитационной модели.

Верификация имитационной модели предполагает доказательство возможности использования создаваемой программной модели в качестве машинного аналога концептуальной модели на основе обеспечения максимального сходства с последней. Цель процедуры верификации – определить уровень, на котором это сходство может быть успешно достигнуто.

Валидация и верификация имитационной модели связаны с обоснованием *внутренней структуры модели*, в ходе этих процедур проводятся испытания внутренней структуры и принятых гипотез, исследуется внутренняя состоятельность модели.

- **Валидация данных.**

*Валидация данных (data validity)* направлена на доказательство того, что все используемые в модели данные, в том числе входные, обладают удовлетворительной точностью и не противоречат исследуемой системе, а значения параметров точно определены и корректно используются.

Эти проверки связаны с проблемным анализом, т.е. анализом и интерпретацией полученных в результате эксперимента данных. *Проблемный анализ* – это формулировка статистически значимых выводов на основе данных, полученных в результате эксперимента на имитационной модели. Проверяется правильность интерпретации полученных с помощью модели данных, оценивается насколько могут быть справедливы статистические выводы, полученные в результате имитационного эксперимента. С этой целью проводят **исследование свойств имитационной модели**: оценивается *точность, устойчивость, чувствительность результатов моделирования*. Эти проверки связаны с выходами модели, сама имитационная модель рассматривается как черный ящик.

Таким образом, на этапе испытания и исследования разработанной имитационной модели организуется комплексное **тестирование модели (testing)** – **планируемый итеративный процесс, направленный главным образом на поддержку процедур верификации и валидации имитационных моделей и данных.**

Некоторые полезные процедуры тестирования рассмотрим ниже. Более широкое изложение методов тестирования имитационных моделей можно найти в специальной литературе [20, 33, 56].

## **6.2 Проверка адекватности модели.**

При моделировании исследователя прежде всего интересует, насколько хорошо модель представляет моделируемую систему (объект моделирования). Модель, поведение которой слишком отличается от поведения моделируемой системы, практически бесполезна.

Различают *модели существующих и проектируемых систем*.

Если реальная система (или ее прототип) существует, дело обстоит достаточно просто. Поэтому для моделей существующих систем исследователь должен выполнить проверку адекватности имитационной модели объекту моделирования, т.е. проверить соответствие между поведением реальной системы и поведением модели.

На *реальную систему* воздействуют **переменные  $G^*$** , которые можно измерять, но нельзя управлять, **параметры  $X^*$** , которые исследователь

может изменять в ходе натуральных экспериментов. На выходе системы возможно измерение *выходных характеристик*  $Y^*$ .

При этом существует некоторая неизвестная исследователю зависимость между ними  $Y^*=f^*(X^*, G^*)$ .

Имитационную модель можно рассматривать как преобразователь входных переменных в выходные. В любой имитационной модели различают составляющие: компоненты, переменные, параметры, функциональные зависимости, ограничения, целевые функции. *Модель* системы определяется как совокупность компонент, объединенных для выполнения заданной функции  $Y=f(X, G)$ . Здесь  $Y, X, G$  – векторы соответственно результата действия модели системы *выходных переменных, параметров моделирования, входных переменных модели*. Параметры модели  $X$  исследователь выбирает произвольно,  $G$  – принимают только те значения, которые характерны для данных объекта моделирования.

Очевидный подход в оценке адекватности состоит в *сравнении выходов модели и реальной системы при одинаковых* (если возможно) *значениях входов*. И те, и другие данные (данные, полученные на выходе имитационной модели и данные, полученные в результате эксперимента с реальной системой) – статистические. Поэтому *применяют методы статистической теории оценивания и проверки гипотез*.

Используя соответствующий статистический критерий для двух выборок, мы можем проверить статистические гипотезы ( $H_0$ ) о том, что выборки выходов системы и модели являются выборками из различных совокупностей или ( $H_1$ ), что они "практически" принадлежат одной совокупности.

Могут быть рекомендованы два основных подхода к оценке адекватности:

- *1 способ: по средним значениям откликов модели и системы.*

Проверяется гипотеза о близости средних значений каждый  $n$ -й компоненты откликов модели  $Y_n$  известным средним значениям  $n$ -й компоненты откликов реальной системы  $Y^*_n$ .

Проводят  $N_1$  опытов на реальной системе и  $N_2$  опытов на имитационной модели (обычно  $N_2 > N_1$ ).

Оценивают для реальной системы и имитационной модели математическое ожидание и дисперсию,  $Y'_n, D'_n$  и  $Y_n, D_n$  соответственно.

Гипотезы о средних значениях проверяются с помощью критерия  $t$ -Стьюдента, можно использовать параметрический критерий Манна-Уитни и др.

Например, продемонстрируем использование  $t$ -статистики. Основой проверки гипотез является  $E_n = (Y_n - Y'_n)$ , оценка её дисперсии:

$$D_{pn} = \frac{(N_1 - 1)D_n + (N_2 - 1)D_n'}{N_1 + N_2 - 2}$$

$t$ -статистика:

$$t_n = (Y_n - Y_n') \sqrt{\frac{N_1 N_2}{D_{pn} (N_1 + N_2)}}$$

Берут таблицу распределения  $t$ -статистики с числом степеней свободы:  $\gamma = N_1 + N_2 - 2$  (обычно с уровнем значимости  $\alpha = 0.05$ ). По таблицам находят критическое значение  $t_{кр}$ . Если  $t_n \leq t_{кр}$ , гипотеза о близости средних значений  $n$ -й компоненты откликов модели и системы принимается. И т.д. по всем  $n$ -компонентам вектора откликов.

- 2 способ: по дисперсиям отклонений откликов модели от среднего значения откликов систем.

Сравнение дисперсии проводят с помощью критерия  $F$  (проверяют гипотезы о согласованности), с помощью критерия согласия  $\chi^2$  (при больших выборках,  $n > 100$ ), критерия Колмогорова-Смирнова (при малых выборках, известны средняя и дисперсия совокупности), Кохрена и др.

Проверяется гипотеза о значимости различий оценок двух дисперсий:  $D_n^*$  и  $D_n$ .

Составляется  $F$ -статистика:  $F = D_n / D_n^*$  (задаются обычно уровнем значимости  $\alpha = 0,05$ , при степенях свободы  $\gamma_1 = \gamma_2 = N_1 = N_2$ ), по таблицам Фишера для  $F$ -распределения находят  $F_{кр}$ . Если  $F > F_{кр}$ , гипотеза о значимости различий двух оценок дисперсий принимается, значит — отсутствует адекватность реальной системы и имитационной модели по  $n$ -ой компоненте вектора отклика.

Процедура повторяется аналогичным образом по всем компонентам вектора отклика. Если хотя бы по одной компоненте адекватность отсутствует, то модель неадекватна. В последнем случае, если обнаружены незначительные отклонения в модели, может проводиться *калибровка имитационной модели* (вводятся поправочные, калибровочные коэффициенты в моделирующий алгоритм), с целью обеспечения адекватности.

А если не существует реальной системы (что характерно для задач проектирования, прогнозирования)? Проверку адекватности выполнить в этом случае не удастся, поскольку нет реального объекта. Для целей исследования модели иногда проводят специальные испытания (например, так поступают при военных исследованиях). Это позволяет убедиться в точности модели, полезности ее на практике, несмотря на сложность и дороговизну проводимых испытаний.

Могут использоваться и другие подходы к проведению валидации имитационной модели [56], кроме статистических сравнений между

откликами реальной системы и модели. В отдельных случаях полезна валидация внешнего представления, когда проверяется насколько модель выглядит адекватной с точки зрения специалистов, которые с ней будут работать, так называемый *тест Тьюринга* (установление экспертами различий между поведением модели и реальной системы). В процессе валидации требуется постоянный контакт с заказчиком модели, дискуссии с экспертами по системе. Рекомендуется также проводить эмпирическое тестирование допущений модели, в ходе которого может осуществляться графическое представление данных, проверка гипотез о распределениях, анализ чувствительности и др. Важным инструментом валидации имитационной модели является графическое представление промежуточных результатов и выходных данных, а также анимация процесса моделирования. Наиболее эффективными являются такие представления данных, как гистограммы, временные графики отдельных переменных за весь период моделирования, графики взаимозависимости, круговые и линейчатые диаграммы. Методика применения статистических технологий зависит от доступности данных по реальной системе.

### 6.3 Верификация имитационной модели.

**Верификация** модели — есть доказательство утверждений соответствия алгоритма ее функционирования замыслу моделирования и своему назначению. На этапе верификации устанавливается *верность логической структуры модели*, реализуется комплексная отладка с использованием средств трассировки, ручной имитации, в ходе которой проверяется правильность реализации моделирующего алгоритма.

Комплексные процедуры верификации включают неформальные и формальные исследования программы-имитатора. Неформальные процедуры могут состоять из серии проверок следующего типа:

- проверка преобразования информации от входа к выходу;
- трассировка модели на реальном потоке данных (при заданных  $G$  и  $X$ ):
  - $X$  изменяется по всему диапазону значений — контролируется  $Y$ ;
  - можно посмотреть, не будет ли модель давать абсурдные ответы, если ее параметры будут принимать предельные значения;
  - "проверка на ожидаемость", когда в модели заменяют стохастические элементы на детерминированные и др.

Полезным при решении указанных задач могут быть также следующие приёмы [56]:

- обязательное масштабирование временных параметров в зависимости от выбранного шага моделирования (валидация данных);
- валидация по наступлению "событий" в модели и сравнение (если возможно) с реальной системой;

- тестирование модели для критических значений и при наступлении редких событий;
- фиксирование значений для некоторых входных параметров с последующим сравнением выходных результатов с заранее известными данными;
- вариация значениями входных и внутренних параметров модели с последующим сравнительным анализом поведения исследуемой системы;
- реализация повторных прогонов модели с неизменными значениями всех входных параметров;
- оценка фактически полученных в результате моделирования распределений случайных величин и оценок их параметров (математическое ожидание и дисперсия) с априорно заданными значениями;
- сравнение исследователями поведения и результатов валидируемой модели с результатами уже существующих моделей, для которых доказана достоверность;
- для существующей реальной исследуемой системы предсказание её будущего поведения и сравнение прогноза с реальными наблюдениями.

*Формальные процедуры связаны с проверкой исходных предположений (выдвинутых на основе опыта, теоретических знаний, интуитивных представлений, на основе имеющейся информации). Общая процедура включает:*

- построение ряда гипотез о поведении системы и взаимодействии ее элементов;
- проверка гипотез с помощью статистических тестов: используют методы статистической теории оценивания и проверки гипотез (методы проверки с помощью критериев согласия ( $\chi^2$ , Колмогорова-Смирнова, Кокрена и др.), непараметрические проверки и т.д., а также дисперсионный, регрессионный, факторный, спектральный анализы).

## **6.4 Валидация данных имитационной модели.**

*Валидация данных имитационной модели предполагает исследование свойств имитационной модели, в ходе которого оценивается точность, устойчивость, чувствительность результатов моделирования и другие свойства имитационной модели.*

Наиболее существенные процедуры исследования свойств модели:

- *оценка точности результатов моделирования;*
- *оценка устойчивости результатов моделирования;*

- *оценка чувствительности имитационной модели.*

Получить эти оценки в ряде случаев бывает весьма сложно. Однако без успешных результатов этой работы, доверия к модели не будет, невозможно будет провести корректный проблемный анализ и сформулировать статистически значимые выводы на основе данных, полученных в результате имитации.

## **6.5 Оценка точности результатов моделирования.**

Экспериментальная природа имитации требует, чтобы мы учитывали случайную вариацию оценок, получаемых на модели характеристик. В ходе этих испытаний исследователь интересуется выходами модели, прежде всего его интересует:

- какой разброс данных на выходе имитационной модели или точность имитации;
- и какие выводы можно сделать по полученным результатам моделирования.

*Точность имитации явлений представляет собой оценку влияния стохастических элементов на функционирование модели сложной системы.*

Всем стохастическим элементам имитационной модели присущи флуктуации. Данные одного имитационного прогона представляют собой единичную выборку, т.е. результаты прогонов на имитационной модели могут рассматриваться как случайные величины — и к ним должны быть применены все определения и правила математической статистики. Ясно, для того, чтобы увеличить точность имитации (уменьшить разброс данных) необходимо выборку сделать большой (представительной). Поэтому, для достижения заданной точности результатов эксперимента, либо повторяют эксперимент несколько раз, либо имитируют более продолжительный период времени и оценивают полученные результаты (см. основные задачи тактического планирования: определение продолжительности имитационного прогона, анализ установившегося состояния). Для повышения точности используют специальные методы понижения дисперсии.

*Степень точности определяется величиной флуктуации случайного фактора (дисперсией). Мерой точности является доверительный интервал.*

Для определения точности результатов имитации оцениваем доверительные интервалы. Если мы имеем оценку  $\bar{X}$  истинного среднего  $\mu$  совокупности, мы определяем верхнюю и нижнюю границы интервала, так, чтобы вероятность попадания истинного среднего в интервал, заключенный между этими границами, равнялась некоторой заданной величине ( $\alpha$  — доверительная вероятность) следующим образом:



$$P\{\mu - d < X < \mu + d\} = 1 - \alpha, \text{ где} \quad (6.5)$$

$\bar{X}$  – выборочное среднее,

$1 - \alpha$  – вероятность того, что интервал  $\mu \pm d$  содержит  $X$ .

## 6.6 Оценка устойчивости результатов моделирования.

*Под устойчивостью результатов имитации будем понимать степень нечувствительности ее к изменению условий моделирования.* Универсальной процедуры для такой проверки не существует.

*Устойчивость результатов моделирования* характеризуется сходимостью контролируемого параметра моделирования к определенной величине при увеличении времени моделирования варианта сложной системы.

На практике, рекомендуется устойчивость результатов моделирования оценивать дисперсией значений отклика (по выбранной компоненте). Если эта дисперсия при увеличении времени моделирования  $T_{\text{мог}}$  не увеличивается, значит, результаты моделирования устойчивы.

Может быть рекомендована следующая *методика оценки устойчивости*. В модельном времени с шагом  $t$  контролируются выходные параметры  $Y$ . Оценивается амплитуда изменений параметра  $Y$ . Рост разброса контролируемого параметра от начального значения при изменении  $t + \Delta t$  указывает на неустойчивый характер имитации исследуемого процесса.

Для проверки статистической гипотезы о равенстве дисперсий значений откликов имитационной модели ( $\sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2$ ) для испытаний с различными длительностями прогонов может быть использован критерий Бартлетта:

$$B = \frac{1}{c} \left( \int \ln S^2 - \sum_{i=1}^k f_i \ln S_i^2 \right), \text{ где}$$

$$S^2 = \frac{1}{f} \sum_{i=1}^k f_i S_i^2$$

$$f = \sum_{i=1}^k f_i, f_i = n - 1$$

Методика несмещенной оценки  $k$ -дисперсий нормальных генеральных совокупностей:

1. Устанавливается длительность прогона ( $0, t^{\text{мог}}$ )

2. Выбирается контролируемая компонента вектора отклика  $y_i$

3. Задается шаг  $\Delta t$ ,

На каждом шаге контролируется  $y_i$ , оценивается дисперсия и т.д.

Формулируется нулевая статистическая гипотеза: о равенстве дисперсий и проверяется с помощью критерия Бартлетта.

$V_{расч}$  сравнивается с тестовой. Если  $V > \chi^2$ , то  $H_0$  принимается. Считается, что модель устойчива по  $i$ - компоненте вектора отклика.

и т.д. по всем компонентам

В случае удачной проверки, считается, что модель устойчива по всему вектору выходных переменных.

## 6.7 Анализ чувствительности имитационной модели.

*Анализ чувствительности модели* определяет оценку влияния колебаний значений входных переменных на отклики (выходные переменные) модели. Необходимо установить, при каком разбросе входных данных сохраняется справедливость основных выводов, сделанных по результатам моделирования.

Под *анализом чувствительности* понимаем определение чувствительности наших окончательных результатов моделирования к изменению используемых значений параметров. Анализ означает, как меняется выходная переменная  $Y$  при небольших изменениях различных параметров модели или ее входов  $X$ .

Простота проведения анализа чувствительности в имитационном моделировании — одно из преимуществ этого метода. Оценка чувствительности является исключительно важной процедурой и подготовительным этапом перед планированием имитационного эксперимента.

Дело в том, что величины параметров систематически варьируются в некоторых представляющих интерес пределах ( $X_{min} - X_{max}$ ) и наблюдается влияние этих вариаций на характеристики системы ( $Y_{min} - Y_{max}$ ). Если при незначительных изменениях величин некоторых параметров результаты меняются очень сильно, то это — основание для затраты большого количества времени и средств с целью получения более точных оценок. И наоборот, если конечные результаты при изменении величин параметров в широких пределах не изменяются, то дальнейшее экспериментирование в этом направлении бесполезно и неоправданно. Поэтому очень важно определить степень чувствительности результатов относительно выбранных для исследования величин — параметров.

Исследование чувствительности является *предварительной процедурой перед планированием эксперимента* и позволяет определить стратегию планирования экспериментов на имитационной модели. Этой информа-

ции бывает достаточно для ранжирования компонент вектора параметров модели  $X$  по значению чувствительности вектора отклика модели. Если модель оказывается малочувствительной по какой-либо  $q$ -й компоненте вектора параметров модели  $X_q$ , то зачастую не включают в план имитационного эксперимента изменение  $X_q$ , чем достигается экономия ресурса времени моделирования.

Анализ чувствительности поможет также *внести коррективы в разрабатываемую модель* — упростить, например, перейти от использования закона распределения к использованию среднего значения переменной, а некоторые подсистемы вообще отбросить (или процессы не детализировать). И наоборот, анализ чувствительности может показать, какие части модели было бы полезно разработать более детально.

*Чувствительность имитационной модели* представляется величиной минимального приращения выбранного критерия качества, вычисляемого по статистикам моделирования, при последовательном варьировании параметров моделирования на всем диапазоне их изменения.

*Методика* (процедура) *оценки чувствительности* [33]:

По каждому фактору  $X$  определяется интервал изменения ( $\min X_q$ ,  $\max X_q$ ). Остальные компоненты вектора  $X$  не изменяются и соответствуют центральной точке. Проводят пару модельных экспериментов и получают отклики модели ( $\min Y$ ,  $\max Y$  соответственно). Для оценки чувствительности используют абсолютные значения или относительные. В последнем случае вычисляют приращение вектора параметров

$$\delta X_q^0 = \frac{(\max X_q - \min X_q)^2}{(\max X_q + \min X_q)} 100\%$$

и вычисляют приращение вектора отклика

$$\delta Y_q = \frac{(\max Y_q - \min Y_q)^2}{(\max Y_q + \min Y_q)} 100\%$$

Выбирают .  $\delta Y_q^0 = \max \{\delta Y_n\}$

Итак, чувствительность модели по  $q$ - компоненте вектора параметров  $X$  определяют парой значений  $(\delta X_q^0, \delta Y_q^0)$ .

*Все рассмотренные процедуры в комплексе дают необходимую информационную базу обеспечения доверия к разработанной имитационной модели и перехода к следующим этапам работы с моделью.*

## 6.8 Тактическое планирование имитационного эксперимента.

Процедуры верификации и валидации собственно имитационной модели и её программного кода требуют проведения широкого спектра тестовых имитационных экспериментов согласно сценариям, разработанным в процессе как тактического, так и стратегического планирования. Стратегическое планирование (подробно рассматривается в следующей лекции) направлено на решение задач анализа чувствительности имитационной модели и определение комбинации оптимизирующих исследуемую систему параметров.

*Тактическое планирование* позволяет определиться с условиями проведения каждого прогона в рамках составленного плана эксперимента и связано с вопросами эффективности и определением способов проведения испытаний (прогонов), намеченных планом экспериментов.

*Тактическое планирование* направлено на решение проблемы точностного оценивания имитационных моделей и связано с тем, что в условиях стохастической модели, чтобы достигнуть заданной точности результатов экспериментов стремятся повторять реализации (проводить многочисленные прогоны). Время на серию машинных прогонов сложного модельного эксперимента может быть большим, а выделенное на эксперимент машинное время ограничивается имеющимися временными и машинными ресурсами. Поэтому необходимо стремиться к получению максимальной информации с помощью небольшого числа прогонов. Основное *противоречие* — *между точностью результатов и ограничением на ресурс* (затратами на машинное время и на проведение серии экспериментов). На практике ищется компромисс. Выше мы обсуждали, что экспериментатор должен не только получить данные, но и оценить их точность, т.е. степень доверия к тем выводам, которые будут сделаны на основе этих результатов. Поэтому он стремится увеличить продолжительность прогона или число прогонов (размер выборки), т.к. от этого зависит точность результатов.

*Основные вопросы (задачи) тактического планирования*, которые решаются в связи с этим:

- Определение продолжительности имитационного прогона или требуемого числа повторений каждого прогона (размера выборки), обеспечивающего заданную точность результатов моделирования;
- Определение длительности переходного режима (анализ установившегося состояния), задание начальных условий (начального состояния).

Здесь решаются также такие задачи:

- выбор корректного шага моделирования, поскольку именно от шага моделирования зависит точность воспроизведения в имитационной модели имеющих место в реальной системе цепочек событий;
- контроль повторяемости результатов;
- установление правил остановки,
- уменьшении дисперсии выходов (используются специальные методы понижения дисперсии),
- снижение погрешности имитации, обусловленной наличием в имитационной модели генераторов псевдослучайных чисел,
- и многие другие.

Рассмотрим основные задачи тактического планирования.

*Определение необходимого числа прогонов.*

*Основные методы организации прогонов: повторные прогоны, метод подинтервалов* (прогоны делятся на группы, вычисляется среднее). Чтобы сделать статистический анализ по всей последовательности моделируемого случайного процесса, либо повторяют имитацию несколько раз (метод повторных прогонов), либо имитируют более продолжительное время (метод удлинненных прогонов).

*Основные методы задания продолжительности имитационного прогона:*

- Часто задается момент времени завершения моделирования;
- Метод, управляющий размером выборки (применяются правила автоматической остановки):
  - задание определенного числа компонентов, поступающих на вход модели,
  - задание числа компонентов, обрабатываемых в системе
  - и др.

Чтобы результаты, полученные на имитационной модели, были статистически значимы, стремятся повысить точность результатов моделирования, повторяя эксперимент и усредняя полученные результаты. Этот способ неэффективен, т.к.:

$$\sigma \sim \frac{1}{\sqrt{n}}, \text{ где}$$

$n$  – число повторений.

Вопрос: как много выборочных значений следует взять, чтобы обеспечить статистическую значимость результатов моделирования. *Задача состоит в том, чтобы определить при заданной точности*

необходимое число прогонов. Задача является обратной рассмотренной выше задачи оценивания точности методом доверительных интервалов.

Задача состоит в определении необходимого для выполнения (6.5) объема выборки  $n$ .

В предположении нормальности распределения выборочных значений из генеральной совокупности можно показать, что

$$n = (\sigma Z_{\alpha/2})^2 / d^2 \text{ где}$$

$Z_{\alpha/2}$  -двусторонняя станд.нормальная статистика.

Для решения  $n$  необходимо знать  $\sigma$ ,  $Z_{\alpha/2}$ ,  $d$ .

На практике проводят пробные прогоны и поступают следующим образом:

$$n \cong Z_{\alpha/2}^2 \frac{\sigma^2}{d^2}$$

$$n \cong n_0 \frac{d_0^2}{d^2}, \text{ где}$$

$n$  – необходимое число прогонов;

$n_0$  – количество пробных прогонов;

$d_0$  – доверительный интервал, оцененный по результатам пробного прогона;

$d$  – требуемая точность.

*Анализ установившегося состояния. Определение участка разгона (разогрева) модели для исключения неустановившихся режимов функционирования системы.*

*Установившимся (стационарным) называется такое состояние модели, когда последовательные наблюдения отклика в установившемся состоянии имеют некоторое предельное стационарное распределения вероятностей, не зависят от времени.*

Стационарность режима моделирования характеризует собой некоторое установившееся равновесие процессов в модели системы. Часто говорят, что динамическая система находится в равновесии (стационарном состоянии), если её функционирование происходит в соответствии с параметрами предельного стационарного распределения, которое не зависит от времени (т.е. если имитировать и дальше, то новой информации не получишь и продолжение имитации будет являться бессмысленной тратой времени). Обычно имитационные модели применяются для изучения системы в типичных условиях. Установившееся состояние обычно характерно для типичных условий функционирования (для систем

массового обслуживания не естественны начальные состояния  $(0,0)$  – “пуст и свободен”), например, процессы обслуживания в аэропорту, на транспорте, в больнице.

В некоторых стохастических моделях требуется некоторое время для достижения моделью необходимого установившегося состояния, имитационная модель не сразу выходит в стационарный режим работы. Необходимо определить момент достижения стационарного режима моделирования и позаботиться об уменьшении влияния начального периода моделирования или его исключения из результатов моделирования. Очевидно, что если исследовать только установившийся режим, то качество статистических оценок повысится. Чтобы исключить влияние начального периода на результаты моделирования можно предложить:

- Использование длинных прогонов модели (если позволяет машинное время) для достижения статистического равновесия. Недостаток такого подхода – требуется много машинного времени.
- Исключение из рассмотрения начального периода прогона. Продумывают процедуры отсечения, когда имитационная модель выходит в установившийся режим, в этот момент уничтожают предыдущую статистику. Недостаток – надо контролировать результаты моделирования, при этом сложно определить установившееся состояние в условиях стохастической модели, кроме того машинное время тратится не на имитацию, а на анализ установившегося режима.
- Выбор такого начального условия, которое ближе всего к типичному, тем самым достигается существенное уменьшение длительность переходного режима в модели. Для этого проводят пробный прогон, чтобы определить момент выхода системы в стационарный режим.

Однако рассмотренной процедурой надо пользоваться аккуратно, т.к. для некоторых исследуемых процессов переходный режим может представлять самостоятельный интерес.

Обсуждение других задач тактического планирования можно найти в специальной литературе [20,33].

## ЛЕКЦИЯ

## 6

**ТЕХНОЛОГИЯ ПОСТАНОВКИ И ПРОВЕДЕНИЯ НАПРАВЛЕННОГО ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА НА ИМИТАЦИОННОЙ МОДЕЛИ.****7.1 Направленный вычислительный эксперимент на имитационной модели и его содержание.**

Мы уже обсуждали с Вами, что компьютерное моделирование рассматривает вычислительный эксперимент как новую методологию научного исследования. В лекции 2 мы определяли метод имитационного моделирования как **экспериментальный метод**, предполагающий экспериментирование с помощью модели с целью получения информации о реально действующей системе. Вычислительный эксперимент здесь рассматривается как *целенаправленное* исследование, организованное на имитационной модели, которое позволяет получить информацию, необходимую для принятия решений.

Имитационные модели, отмечали мы там же, не расчетные, а *прогонного типа*. Одна реализация исследования на имитационной модели, эксперимент состоит из серии прогонов, в ходе которого осуществляется оценка функционирования системы при заданном (фиксированном) наборе условий, при определенном наборе исходных данных и управляемых параметров (здесь мы временно отвлечемся от стохастичной природы имитации и будем подразумевать детерминированный случай). Характерной чертой имитационной модели является то, что каждый машинный прогон имитационной модели дает результаты, которые действительны только при определенных значениях параметров, переменных и структурных взаимосвязей, заложенных в имитационную программу, т.е. для различных исследуемых вариантов могут изменяться параметры, переменные, операционные правила, структурные отношения, которые характеризуют определенный вариант. Цель имитационных исследований – снабжать данными при изменении входных условий.

*Имитационный эксперимент, содержание которого определяется:*

- *предварительно-проведенным аналитическим исследованием (являющимся составной частью вычислительного эксперимента),*
- *результаты которого достоверны и математически обоснованы,*  
назовем **направленным вычислительным экспериментом.**



Приведенное выше определение отражает две части и основные задачи исследователя при организации и проведении вычислительного эксперимента на имитационной модели. Эти задачи включают:

- Стратегическое планирование вычислительного эксперимента;
- Выбор (математического) метода анализа (обработки) результатов вычислительного эксперимента.

Рассмотрим первую задачу. *Проблемы стратегического планирования* в имитационном исследовании мы обсуждали ранее. Стратегическое планирование вычислительного эксперимента — это организация вычислительного эксперимента, выбор метода сбора информации, который дает требуемый (для данной цели моделирования, для принятия решения) ее объем *при наименьших затратах*. Т.е. основная цель стратегического планирования — получить желаемую информацию для изучения моделируемой системы при минимальных затратах на экспериментирование, при наименьшем числе прогонов.

Перед началом исследования необходимо спланировать эксперимент — разработать *план проведения эксперимента на модели*. Цель этого планирования двоякая:

1) Планирование эксперимента позволяет выбрать конкретный *метод сбора* необходимой для получения обоснованных выводов *информации*, т.е. план задает схему исследования. Таким образом, *план эксперимента служит структурной основой процесса исследования*.

2) Достигнуть цели исследования *эффективным* образом, т.е. уменьшить число экспериментальных проверок (прогонов).

Действительно, если в процессе имитационного исследования рассматривается большое число вариантов (для каждого варианта могут меняться параметры, переменные, структурные отношения), то число прогонов растет, растут и затраты машинного времени.

Допустим, число уровней, принимаемых значений переменной всего 2. В случае 3-х двухуровневых факторов необходимо проводить прогонов  $N = 2^3 = 8$ , при 7 факторов требуемое число прогонов возрастает до  $2^7 = 128$ .

Проблема выбора ограниченного числа прогонов может быть решена с помощью *статистических методов планирования эксперимента*, которые мы будем рассматривать ниже.

Вторая задача при организации и проведении направленного вычислительного эксперимента на имитационной модели: *выбор метода анализа результатов*. В зависимости от целей и задач вычислительного эксперимента могут применяться различные математические методы для

обработки результатов эксперимента. На данном технологическом этапе имитационного моделирования, имитационная модель представляется исследователю в виде черного ящика, как показано на рисунке 7.1. Взаимосвязь  $F$  между входом  $X$  и выходом  $Y$  имитационной модели, должна быть промоделирована с помощью некоторой вторичной модели, отвечающей стратегическим требованиям. В простейшем случае — это может быть некоторая линейная регрессионная модель. В задачах интерполяции ищется функция  $F$ , в задачах оптимизации — экстремум функции  $F$ .

$$Y = F(x_1, x_2, \dots, x_k).$$

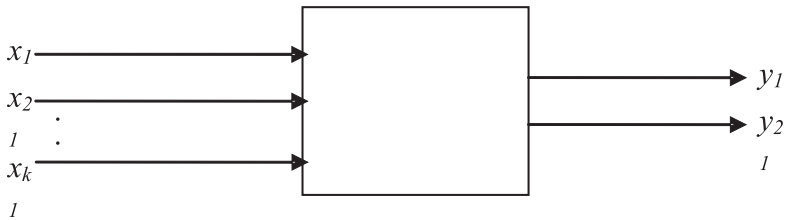


Рис. 7.1 — Представление имитационной модели в виде "черного ящика".

Выбор метода анализа результатов исследования зависит в основном от цели и характера исследования. На практике, выбор аналитического метода анализа результатов исследования (вторичной математической модели) определяется целью исследования и методом статистического анализа его результатов, необходимым для достижения этой цели.

Итак, решение основных задач рассмотренного этапа имитационного моделирования предполагает составление такого плана компьютерного эксперимента, позволяющего достигать поставленные цели эксперимента эффективным образом, с учетом ограничений на ресурсы, а также выбрать математический метод анализа (истолкования) результатов моделирования. Технологическая цепочка такого экспериментального исследования на имитационной модели представляет последовательность следующих действий: в соответствии с целью исследования осуществляется постановка математической задачи (определяется содержание направленного вычислительного эксперимента); в соответствии с поставленной задачей исследования разрабатывается план эксперимента; по плану проводится серия экспериментов, в ходе которого собирается информация (результаты экспериментов); далее в зависимости от цели эксперимента выбирается метод анализа результатов, позволяющий в конечном итоге принимать исследователю решение по результатам моделирования.

Далее мы будем изучать анонсируемые выше методы, полезные в вычислительном эксперименте на имитационной модели.

## 7.2 Основные цели и типы вычислительных экспериментов в имитационном моделировании.

В соответствии с наиболее употребимыми целями моделирования, рассмотренными в лекции 3, *целями вычислительного эксперимента* могут быть:

оценка выходных переменных функционирования сложной системы при заданных параметрах системы;

выбор на множестве альтернатив;

получение знаний о влиянии управляемых параметров на результаты эксперимента;

определение тех значений входных параметров и переменных, при которых достигается оптимальный выход (отклик).

Наиболее широко на практике распространены следующие *типы вычислительных экспериментов*, представленные в таблице 7.2.:

Таблица 7.2 – Основные типы направленных вычислительных экспериментов

Цели вычислительного эксперимента	Тип направленного вычислительного эксперимента	Применяемые математические модели и методы
1. Оценка выходных переменных при заданных параметрах системы. 2. Сравнение альтернатив (или выбор на множестве альтернатив)	<i>1 тип:</i> <b>Оценка и сравнение средних и дисперсий различных альтернатив</b>	Статистические методы оценивания, проверки гипотез; однофакторный дисперсионный анализ; методы множественного ранжирования и сравнения; процедуры ранжирования, отбора, эвристические приемы и др.
3. Получение знаний о влиянии управляемых параметров на результаты эксперимента	<i>2 тип:</i> <b>Анализ чувствительности (задача интерполяции)</b>	Методы планирования эксперимента, дисперсионный, регрессионный анализ; и др.
4. Определение тех значений входных параметров и переменных, при которых достигается оптимальный выход	<i>3 тип:</i> <b>Поиск оптимума на множестве возможных значений переменных (задача оптимизации)</b>	Последовательные методы планирования эксперимента (методология анализа поверхности отклика).
5. Вариантный синтез	<b>Многокритериальная оптимизация, выбор</b>	Итерационные имитационно-оптимизационные процедуры, методы принятия решений

1 тип: *Оценка и сравнение средних и дисперсий различных альтернатив.*

2 тип: *Анализ чувствительности* (параметрический анализ) системы к изменению параметров. Основным содержанием такого эксперимента является определение влияния управляемых параметров, переменных (факторов) на результаты экспериментов (отклик). В эксперименте 2 типа ставится математическая *задача интерполяции* и осуществляется построение интерполяционных формул. Например, модель  $F$  на рис.7.1.1. может быть аппроксимирована полиномиальной функцией, например некоторой линейной регрессионной моделью. В задачах интерполяции необходимо *найти функцию  $F$ .*

3 тип: Решается задача оптимизации: *поиск оптимальных значений на некотором множестве возможных значений переменных.* В задачах оптимизации необходимо *найти экстремум функции  $F$ .*

4 тип: *Вариантный синтез*, это более сложный класс вычислительных экспериментов, как правило, связанный с многокритериальной оптимизацией, реализацией итерационных имитационно-оптимизационных процедур [13], выбором и принятием решения в широком смысле слова. Рассмотрение этих методов в этой лекции мы не будем проводить.

Рассмотрим *основные математические модели и методы*, применяемые в первых трех типах вычислительных экспериментов и общие схемы по их организации и проведению.

Эксперименты первого типа довольно просты и обычно являются так называемыми однофакторными экспериментами, подробнее рассматриваются в разделе 7.4. Основные вопросы, встающие перед экспериментатором при их проведении, — это вопросы о размере выборки, начальных условиях, наличие или отсутствии автокорреляции и другие задачи тактического планирования машинного эксперимента, которые рассматриваются подробнее в соответствующем разделе учебника.

Основные математические методы, применяемые и рекомендуемые в этом эксперименте:

- *статистические методы оценивания* путем использования таких величин, как среднее значение, стандартное отклонение, коэффициент корреляции др.;
- процедуры *проверки гипотез* с использованием стандартной тестовой статистики ( $t$ ,  $F$ ,  $2$  и др.), *однофакторный дисперсионный анализ*;
- при сравнении и выборе альтернатив Клейнен [20] рекомендует статистические процедуры ранжирования (веса) и отбора: методы множественного ранжирования и методы множественного сравнения;

- в более сложных случаях могут быть полезны различные эвристические приемы.

Эксперименты второго типа предполагают обычно широкое использование *методов планирования эксперимента*, которые мы подробно будем изучать в следующем разделе учебника. Основными методами истолкования результатов этих экспериментов являются *дисперсионный и регрессионный анализы*. Для исследования динамических рядов (в моделях системной динамики) рекомендуется *спектральный анализ*.

В терминах теории планирования экспериментов вход модели называется фактором, конкретные значения фактора — уровнями, выход модели — откликом. План эксперимента определяет комбинацию уровней и для каждой комбинации задает число повторных прогонов модели. Выбирают и осуществляют план, далее, используя данные эксперимента, определяют параметры регрессионной модели.

Общая схема исследования здесь следующая:

- Выбор ограниченного числа прогонов вариантов системы решается с помощью статистических методов планирования экспериментов. Используют полные и дробные факторные планы;
- В ходе обработки результатов эксперимента получают параметры регрессионной модели;
- Исследователь выполняет анализ модели (регрессионной зависимости).

3 тип вычислительного эксперимента, ориентированный на решение задачи оптимизации (определяются такие значения управляемых параметров и переменных, которые максимизируют или минимизируют заданную целевую функцию), предполагает использование *последовательных или поисковых методы построения экспериментов*. Полезно применение *методологии анализа поверхности отклика*, рассматриваемой в последующих разделах этой лекции, комбинирующей эти методы планирования и итерационные имитационно-оптимизационные вычислительные процедуры. Принципы оптимизации здесь те же, что и для аналитических моделей, однако выход имитационной модели содержит случайную составляющую, — поэтому необходимо в вероятностной форме задавать ограничения на отклики и осуществлять статистическую интерпретацию значений целевой функции.

### **7.3 Основы теории планирования экспериментов. Основные понятия: структурная, функциональная и экспериментальная модели.**

Проблема выбора ограниченного числа экспериментов для анализа вариантов моделируемой системы может быть решена с помощью статистических методов планирования экспериментов.

Теория планирования эксперимента традиционно используется в химии, физике, сельском хозяйстве. В экономике натуральный эксперимент исключен, так как труден и дорогостоящ, однако машинный эксперимент, когда все факторы находятся под управлением исследователя – возможен.

Теория и практика использования методов планирования разработана в настоящее время достаточно хорошо, – существует большое число работ, специальные справочники, в которых некоторые типы планов стандартизованы, поэтому можно использовать готовые проекты, как выкройки для готовой одежды. Подобно тому, как иногда необходимо подогнать портному выкройку, так и начинающему симуляционисту необходимо изучить основы теории планирования эксперимента, чтобы сделать правильный выбор для своего проекта. Можно рекомендовать следующую специальную литературу, в которой обсуждаются методы планирования машинных экспериментов [16,20, 54].

Машинный эксперимент имеет целый ряд преимуществ по сравнению с физическим:

- Машинный эксперимент управляемый, активный. Существует возможность управления условиями проведения эксперимента. Можно выбирать уровни факторов заведомо постоянные, а не случайные, т.е. строить модели постоянных эффектов. Это упрощает методы планирования экспериментов, снимает проблемы рандомизации и разбиения на блоки, анализа результатов;
- В условиях машинного эксперимента выполняется требование воспроизводимости эксперимента. Присутствует легкость воспроизведения условий проведения экспериментов, легкость прерывания и возобновления эксперимента. Это позволяет, например, на ЭВМ реализовать одну и ту же последовательность событий, что полезно в экспериментах при сравнении альтернатив. Кроме того, можно создавать такие условия в компьютерных экспериментах, которые позволяют выполнять допущения и предположения дисперсионного и регрессионного анализов, например, т.к. независимость отклика, однородность дисперсии и некоторые другие;

Еще одно преимущество – при машинных экспериментах можно использовать последовательные или эвристические методы планирования,

которые могут оказаться нереализуемыми при экспериментах с реальными системами. В компьютерном эксперименте можно прервать эксперимент, выполнить анализ результатов, а дальше принять решение об изменении параметров модели или продолжить эксперимент с теми же параметрами.

В машинном эксперименте возникают и некоторые трудности:

- существует "чистая ошибка опыта", вносимая программными датчиками случайных чисел, правда ее можно оценить на стадии определения пригодности модели;
- большая роль отводится случайным внешним (экзогенным) факторам,
- трудность определения понятия выборочной точки (брать среднее значение отклика в прогоне, или моделируемый интервал времени разбивать и усреднять) и некоторые другие.

### **Основные понятия теории планирования экспериментов.**

Чтобы оказать помощь в самостоятельном освоении теории планирования экспериментов введем некоторые термины и понятия.

При планировании и построении модельных экспериментов мы имеем дело с двумя типами переменных, которые будем называть в этой лекции *факторами* и *откликами*. Для выяснения различий между ними рассмотрим простой эксперимент, в котором рассматриваются лишь две переменные  $x$  и  $y$  и цель которого состоит в ответе на вопрос: как при изменении  $x$  изменяется  $y$ ? В этом случае  $x$  — фактор, а  $y$  — отклик. В литературе факторы называют независимыми переменными, а отклики-выходами или зависимыми переменными. Ранее, при разработке имитационной модели мы использовали термины экзогенный (вход) и эндогенный (выход или состояние) соответственно для фактора и отклика.

Итак, термины *фактор*, *режим*, *независимая переменная*, *входная переменная* и *экзогенная переменная* эквивалентны, так же как и термины *отклик*, *выход*, *зависимая переменная*, *выходная переменная*, *переменная состояния*, *эндогенная переменная*.

*Уровни* — это значения квантования каждого фактора.

Планирование эксперимента по имитационному моделированию, как и другие проблемы планирования, требует систематического подхода.

Для выбора плана эксперимента следует:

- определить критерии планирования эксперимента. В качестве основных *критериев планирования* рассматриваются: отклик, число варьируемых факторов, число уровней, необходимое число измерений переменной отклика;
- синтезировать экспериментальную модель;

- сравнить полученную модель с существующими моделями, со стандартными планами и выбрать оптимальный план.

Процесс построения плана эксперимента разбивается обычно на три этапа:

- построение структурной модели;
- построение функциональной модели;
- построение экспериментальной модели.

*Структурная модель* характеризуется

- числом факторов;
- числом уровней для каждого фактора.

Структурная модель выбирается исходя из того, что должно быть сделано, а функциональная – из того, что может быть сделано.

Выбор этих параметров определяется целями эксперимента, точностью измерений факторов, интересом к нелинейным эффектам и т. п. На этот выбор не должна влиять ограниченность числа возможных измерений, возникающая вследствие ограниченности ресурсов. Подобные ограничения существенны для выбора функциональной модели.

Структурная модель эксперимента, следовательно, имеет вид:

$$N_s = (q_1), (q_2), (q_3) \dots (q_k), \text{ где}$$

$N_s$ -число элементов эксперимента;

$k$ -число факторов эксперимента;

$q_i$ -число уровней  $i$ -го фактора,  $i=1, 2, \dots, k$ .

Мы называем *элементом* основной структурный блок эксперимента, определяемый как простейший эксперимент в случае одного фактора и одного уровня, т. е.  $k = 1, q = 1, N_s = 1$ .

*Функциональная модель* определяет количество элементов структурной модели, которые должны служить действительными измерителями отклика, т. е. определять, сколько необходимо иметь различных информационных точек. Подобные функциональные модели могут быть либо совершенными, либо несовершенными. Функциональная модель называется совершенной, если в измерении отклика участвуют все ее элементы, т. е.  $N_f = N_s$ . Функциональная модель называется несовершенной, если число имеющих место откликов меньше числа элементов, т.е.  $N_f < N_s$ .

В идеале – когда структурная модель совпадает с функциональной, однако в имитационном эксперименте существует ограничение на ресурс. Функциональная модель должна позволить установить компромисс между имеющимися ресурсами и желаниями:

$$N = pq^k, \text{ где}$$



$p$  – число повторений экспериментов;

$q$  – число уровней факторов;

$k$  – число факторов (входных параметров и переменных).

С учетом ограничений на ресурсы нужно определить  $q, k, p$ .

Вид *экспериментальной модели* определяется должным образом подобранными критериями планирования. Разработка плана эксперимента включает ряд шагов, в ходе которых экспериментатор должен ответить на ряд важных вопросов.

Шаг первый состоит в *выборе переменной отклика* (целевой функции, параметра оптимизации), который зависит от цели исследования. Это означает, что мы должны решить, какие отклики интересуют нас в первую очередь, т. е. какие величины необходимо измерить, чтобы получить искомые ответы. Например, при моделировании информационно-поисковой системы нас может интересовать время ответа системы на запрос. В то же время нас может интересовать и максимальное число обслуженных запросов за данный промежуток времени или какие-либо другие характеристики моделируемой системы. При рассмотрении методов планирования в этой лекции мы пока будем иметь дело с однокритериальными задачами.

Основные требования к параметру оптимизации:

- он должен быть эффективным с точки зрения достижения цели;
- универсальным;
- количественным;
- статистически эффективным (наиболее точным);
- имеющим физический смысл, простым и легко вычисляемым;
- существующим (при различных состояниях, ситуациях).

Шаг второй: *выделение существенных факторов*.

После выбора интересующих нас переменных откликов мы должны определить факторы, которые могут влиять на эти переменные. Обычно число таких факторов довольно велико, и потому необходимо выделять среди них несколько наиболее существенных. К сожалению, чем меньше мы знакомы с системой, тем больше таких факторов, которые, как нам представляется, способны, влиять на отклики. Известно, что, как правило, степень понимания явления обратно пропорциональна числу переменных, фигурирующих в его описании. Большинство систем работает в соответствии с принципом Парето, который гласит, что с точки зрения характеристик системы существенны лишь некоторые из множества факторов. Действительно, в большинстве систем 20% факторов определяют 80% свойств системы, а остальные 80% факторов определяют лишь 20% ее свойств. Наша задача – *выделить существенные факторы*.

Предварительная процедура в имитационном моделировании, которая упрощает эту задачу — анализ чувствительности имитационной модели.

После определения переменных отклика и выделения существенных факторов необходимо классифицировать эти факторы в соответствии с тем, как они войдут в будущий эксперимент. Каждый фактор может входить в эксперимент тремя способами:

- 1) фактор может быть постоянным и тем самым играть роль граничных условий эксперимента (в имитационной модели это входные переменные);
- 2) фактор может быть переменным, но неуправляемым и вносить тем самым вклад в ошибки эксперимента (в имитационной модели это, как правило, внешние, экзогенные переменные);
- 3) фактор может быть измеряемым и управляемым. Для построения плана эксперимента важны факторы третьего вида. В имитационной модели это — параметры.

Основные требования к факторам: управляемость (это позволяет реализовать активный эксперимент) и однозначность.

Требования к совокупности факторов:

- выбранное множество должно быть достаточно полным;
- точность фиксации факторов должна быть достаточно высокой;
- совместимость и отсутствие линейной корреляции, независимость факторов, т.е. возможность установления факторов на любом уровне, вне зависимости от уровней других факторов.

Необходимо понимать важность проводимых на этой стадии процесса моделирования рассматриваний. Исследователю необходимо знать, какие переменные ему понадобятся измерять и контролировать в процессе проектирования и проведения эксперимента.

Следующий шаг разработки плана эксперимента состоит в *определении уровней*, на которых следует измерять и устанавливать данный фактор. На это влияет точность измерения, интерес к нелинейным эффектам.

Минимальное число уровней фактора, не являющегося постоянным, равно двум. Очевидно, что число уровней следует выбирать минимально возможным и в то же время достаточным для достижения целей эксперимента. Каждый дополнительный уровень увеличивает стоимость эксперимента, и следует тщательно оценивать необходимость его введения. Выбор для каждого фактора одинакового числа уровней (в особенности если уровней всего два-три) дает определенные аналитические преимущества. Такие структурные модели *симметричны* и имеют вид

$$N=q^k.$$

Уровни могут быть:

- *качественные или количественные;*
- *фиксированные или случайные.*

*Количественной* называется переменная, величина которой может быть измерена с помощью некоторой интервальной или относительной шкалы. Примерами могут служить доход, загрузка, цена, время и т. п. *Качественной* же называется переменная, величина которой не может быть измерена количественно, а упорядочивается методами ранжирования. Примерами качественных переменных могут служить машины, политика, географические зоны, организации, решающие правила, тип очереди в системах массового обслуживания, стратегии в системах принятия решений и т.п. Качественный фактор по своей сути принимает ряд возможных уровней, например стратегий в системах принятия решений. Хотя мы для удобства обозначаем уровни качественного фактора цифрами 1, 2, 3 или буквами *A, B, C*, мы должны помнить, что подобное упорядочение произвольно, так как качественные уровни нельзя измерять с помощью количественной шкалы.

Для количественного фактора необходимо выделить интересующую нас область его изменения и определить степень нашей заинтересованности нелинейными эффектами. Если нас интересуют только линейные эффекты, достаточно выбрать два уровня количественной переменной на концах интервала области ее изменения. Если же исследователь предполагает изучать квадратичные эффекты, он должен использовать три уровня. Соответственно для кубического случая необходимы четыре уровня и т. д. *Число уровней равно минимальному числу точек, необходимых для восстановления полиномиальной функции.*

Анализ данных существенно упрощается, если сделать уровни равноотстоящими друг от друга. Такое расположение позволяет рассматривать *ортогональное разбиение* и тем самым упрощает определение коэффициентов полиномиальной функции. Поэтому обычно две крайние точки интересующей нас области изменения количественной переменной выбирают как два ее уровня, а остальные уровни располагают так, чтобы они делили полученный отрезок на равные части.

Термин *фиксированные уровни* означает, что мы управляем уровнями квантования или устанавливаем их. Если уровни квантования выбираются случайно (например, с помощью метода Монте-Карло), то уровни называются *случайными*. Если используемая для построения эксперимента математическая модель имеет фиксированные параметры, она называется *жесткой моделью*. Если факторы модели могут изменяться случайным образом, она называется *вероятностной моделью*. Если модель содержит

как фиксированные, так и случайные факторы, она называется *смешанной моделью*.

Чтобы определить необходимое число измерений переменной отклика экспериментатор должен ответить также на ряд вопросов:

а) Следует ли выявить взаимное влияние различных факторов?

*Эффектом взаимодействия* можно назвать комбинированное влияние на отклик двух или более факторов, проявляющееся помимо индивидуального влияния всех этих факторов по отдельности. На практике это приводит к нелинейным эффектам при построении регрессионных моделей;

б) Каков характер имеющихся ограничений: ограничено время на исследования, ограничены средства или машинное время на проведение машинных прогонов?

в) Какова требуемая точность?

## 7.4 План однофакторного эксперимента и процедуры обработки результатов эксперимента.

Наиболее прост в планировании так называемый *однофакторный эксперимент*, в котором изменяется лишь единственный фактор. (Уровни исследуемого фактора могут быть качественными или количественными, фиксированными или случайными). Уровнями фактора могут быть различные стратегии работы, различные конфигурации системы и различные уровни входной переменной. Число наблюдений или прогонов для каждого уровня режима или фактора определяется допустимыми затратами, желаемой мощностью проверки или статистической значимостью результатов.

Рассматриваемую ситуацию можно представить в виде следующей математической модели:

$$X_{ij} = \mu + T_j + \varepsilon_{ij}, \text{ где}$$

$X_{ij}$  обозначает  $i$ -е наблюдение ( $i=1, 2, \dots, n$ ) на  $j$ -м уровне ( $j=1, 2, \dots, k$  уровней). Например,  $X_{42}$  обозначает четвертое наблюдение или прогон на втором уровне фактора;

$\mu$  – общее влияние всего эксперимента;

$T_j$  – влияние  $j$ -го уровня,

$\varepsilon_{ij}$  – случайная ошибка  $i$ -го наблюдения на  $j$ -м уровне. В большинстве рассматриваемых в литературе экспериментальных моделей предполагается нормально распределенной случайной величиной с нулевым средним и дисперсией <sup>2</sup>, одинаковой для всех  $j$ .

В более сложных случаях в правую часть приведенного выше уравнения модели включают дополнительные переменные, позволяющие учесть влияние других факторов и условий задачи.

В таблице показан типичный план (макет) однофакторного эксперимента с  $k$  уровнями фактора.

Таблица 7.4 – План однофакторного ДАНа.

<i>Уровень фактора</i>	<i>1</i>	<i>2 ...</i>	<i>j ...</i>	<i>k</i>
	$x_{11}$	$x_{12}$	$x_{1j}$	$x_{1k}$
	$x_{21}$	$x_{22}$	$x_{2j}$	$x_{2k}$
	...			
<i>Среднее</i>	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_j$	$\bar{x}_k$

*Основные методы анализа результатов:*

1. Простейший (процедура ANOVA). Если режим или фактор имеют лишь два уровня, то можно использовать процедуры прямой проверки гипотез с использованием стандартных критериев ( $t$ ,  $F$ ,  $h^2$  или отношений).

2. Если фактор или режим имеет более двух уровней, то обычно используется однофакторный дисперсионный анализ (ДАН) с нулевой гипотезой  $T_j = 0$  для всех  $j$ . Количество наблюдений или прогонов не обязательно одинаково для различных уровней фактора. Если нулевая гипотеза верна, то наблюдение  $X_{ij}$  не зависит от уровня фактора и имеет среднее  $\mu$  и случайную ошибку.

Большинство описанных в литературе классических экспериментальных планов основано на использовании *дисперсионного или регрессионного анализа* после сбора данных. Обычно при наличии качественных факторов используется дисперсионный анализ, а в случае, когда все факторы количественные, – регрессионный анализ. Рассмотрение соотношения между регрессионным и дисперсионным анализом (рис.7.4.).

Дисперсионный анализ приспособлен как к качественным, так и к количественным факторам. Если факторы количественны, то можно использовать ANOVA (ДАН) для проверки, есть ли эффект некоторого фактора, без уточнения (в виде регрессионной кривой) того, как меняется отклик при варьировании фактора во всей области экспериментирования. Если надо оценить отклик в некоторой точке экспериментальной области – следует строить регрессионную кривую.

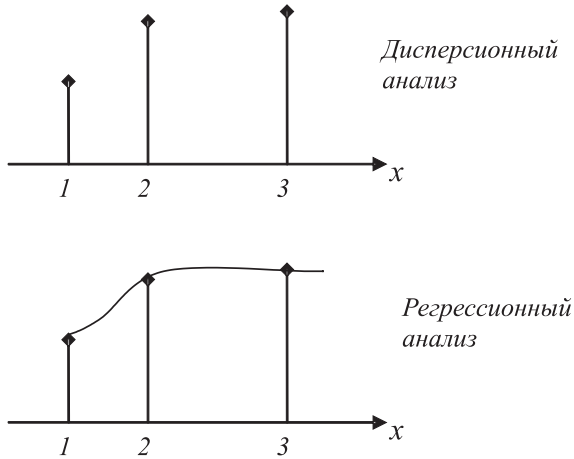


рис 7.4 – Соотношение между дисперсионным и регрессионным анализом.

3. Методы множественного ранжирования. Методы множественных сравнений.

Эти методы применяются при сравнении нескольких вариантов систем, когда изучаем  $k$  – систем, каждой системе соответствует одна частная комбинация уровней факторов, варьируемых в эксперименте.

*Методы множественного ранжирования* – статистические методы полного (для всей совокупности) и неполного ранжирования, в литературе [20] называются методами принятия решений. Эти методы позволяют определить число наблюдений, которые надо взять из каждой  $k > 2$  совокупностей, чтобы выбрать наилучшую совокупность. Наилучшая, обычно, – та совокупность, которая имеет наибольшее (или наименьшее – в зависимости от задачи) – среднее. (Критерием отбора может быть и дисперсия). Большинство процедур ранжирования последовательны. Обзор, эффективность, робастность (устойчивость) методов множественного ранжирования подробно рассматривается в [20].

*Методы множественных сравнений.* Если число экспериментов задано, тогда можно произвести различные типы сравнений между средними совокупностей:

- сравнение со стандартным  $M_0$  средним, соответствующим имеющейся эталонной системе ( $M_i - M_0$ )  $i=1, \dots, k-1$ ;
- все попарные сравнения ( $M_i - M_j$ ),  $i \neq j$ ;
- построить линейные контрасты  $\sum_i C_i M_i$  (и сравнить со стандартной системой и т.п.);

- выделить подгруппу, содержащую наилучшую совокупность, а потом ее дополнительно исследовать экспериментально.

Если все факторы количественны — более эффективен регрессионный анализ, чем методы множественных сравнений.

## **7.5 Факторный анализ, полный и дробный факторный эксперимент и математическая модель.**

В предыдущем разделе нас интересовало влияние на отклик одного фактора. Рассмотрим теперь случай наличия двух и более факторов, влияние которых на отклик должен исследовать экспериментатор.

Возможны следующие методы исследования:

### **Классический метод "один фактор в каждый момент времени".**

Один из традиционных методов исследований многофакторных экспериментов состоит в фиксации всех факторов, кроме одного, на некоторых уровнях и вариации уровней этого фактора. При такой схеме факторы изменяются и исследуются поочередно. Известно, что эксперимент с одним фактором редко обладает достаточной информативностью, если он насчитывает в себе менее 8 выборочных точек на каждом уровне. (Можно привести подобный традиционный 2-х факторный эксперимент с двумя уровнями каждого фактора, выборка имеет объем 32.)

### **Симметричный полный факторный эксперимент.**

Однако можно построить этот эксперимент и как симметричный полный факторный эксперимент, план которого приведен в таблице 7.5.1. *Факторным экспериментом называется такой эксперимент, в котором все уровни данного фактора комбинируются со всеми уровнями всех других факторов.* Под "симметричностью" понимается одинаковое количество уровней для всех факторов.

Основные достоинства факторного анализа:

- простота применения и интерпретации;
- максимальная эффективность метода исследования (факторный анализ позволяет получить требуемую информацию при заданной точности с меньшими затратами, т.е. количество требуемых экспериментов меньше);
- если имеют место взаимодействия между факторами — то их можно правильно идентифицировать и интерпретировать эти взаимодействия (важно в задачах интерпретации);
- результаты справедливы, как правило, в более широком диапазоне условий (т.к. влияние фактора оценивается при нескольких уровнях других факторов).

Если число уровней каждого фактора 2, то имеем полный факторный эксперимент типа  $2^k$  — он прост в планировании. Требуемое количество машинных прогонов  $N = 2^k$ ,  $k$  — число факторов, 2- число уровней.

В планировании эксперимента используют кодированные значения факторов: +1, -1 (1 опускают для простоты). Условия эксперимента описывают в виде таблицы — *матрицы планирования* эксперимента: вектор — строки матрицы соответствуют № прогона, вектор — столбцы — значениям факторов.

Таблица 7.5.1 — Матрица планирования эксперимента

№ опыта	$X_1$	$X_2$	$Y$
1	- 1	- 1	$Y_1$
2	+ 1	- 1	$Y_2$
3	- 1	+ 1	$Y_3$
4	+ 1	+ 1	$Y_4$

Геометрическая интерпретация полных факторных планов в кодированном двухфакторном пространстве имеет вид:

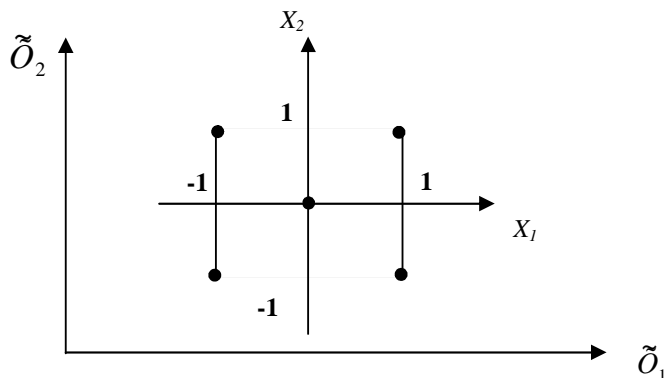


Рисунок 7.5.1 — Геометрическая интерпретация полных факторных планов  $2^2$ .



В области определения факторов ( $\tilde{O}_1, \tilde{O}_2$ ), найдем точку, соответствующую основному уровню, и проведем оси координат. Вершины квадрата соответствуют опытам, каждая сторона равно двум интервалам. Площадь, ограниченная квадратом называется областью определения эксперимента. В задачах интерполяции – это область предсказываемых значений  $y$ .

*План  $2^2$  задается координатами вершин квадрата.*

Геометрической интерпретацией полного факторного эксперимента  $2^3$  служит куб, координаты вершин которого задают условия прогонов.

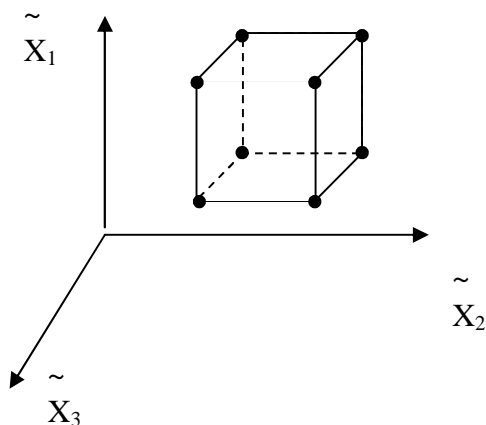


Рисунок 7.5.2 – Геометрическая интерпретация полных факторных планов  $2^3$ .

При  $k > 3$  – план – координаты вершин гиперкуба.

### Свойства полного факторного эксперимента.

Полный факторный эксперимент типа  $2^k$  обладает свойствами: симметричности, нормировки, ортогональности, ротатабельности.

Свойства 1, 2 вытекают из построения матрицы планирования.

1. Симметричность относительно центра эксперимента:

Алгебраическая сумма элементов вектор-столбца каждого фактора равна 0.

$$\sum_{i=1}^N X_{ji} = 0, \text{ где}$$

$j=1, \dots, k$  – номер фактора,

$N$  – число опытов.

2. *Условие нормировки*: сумма квадратов элементов каждого столбца равна числу опытов, или  $N$

$$\sum_{i=1}^N X_{ji}^2 = N,$$

(т.к. значения факторов в матрице задаются  $+1, -1$ )

1 и 2 – это свойства отдельных столбцов матрицы планирования. Теперь остановимся на свойствах совокупности столбцов.

3. *Ортогональность* матрицы планирования: сумма почленных произведений любых двух вектор-столбцов матрицы равна 0.

$$\sum_{i=1}^N X_{ji} X_{ui} = 0, j \neq u, j, u = 0, 1, 2, \dots, k$$

4. *Ротатабельность* (для линейной модели), т.е. точки в матрице планирования подбираются так, что точность предсказания значений параметра оптимизации одинакова на равных расстояниях от центра эксперимента и не зависит от направления.

### **Полный факторный эксперимент и его математическая модель.**

Выход (отклик) имитационной модели системы является временным рядом.

Пусть, выход содержит только одну выходную переменную. Для множественных выходов рассматриваемая процедура должна быть применена к каждой переменной отдельно.

Чтобы сравнивать различные варианты системы, целый временной ряд характеризуется посредством одной или нескольких величин (среднее, стандартное отклонение и т.д.). Назовем такую величину переменной отклика  $-Y$ .

Если  $X = (x_1, \dots, x_k)$  – факторы, то отклик  $Y$  является функцией факторов  $X$ .

$$Y = f(x_1, \dots, x_k)$$

$Y$  – стохастическая переменная, является функцией от  $k$  факторов  $x_j$  ( $j=1, k$ );  $f$  – и есть результат действия имитационной модели. Эта модель может быть аппроксимирована некоторой (например, линейной) регрессионной моделью (внутри некоторой экспериментальной области  $E$ ).

Простейшая (линейная) регрессионная модель для выражения эффектов от  $k$ -факторов имеет вид:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \dots + \beta_k X_{ik} + e_i \quad (i=1, N)$$

где в  $i$ -м имитационном прогоне ( $i$ -наблюдение) фактор  $j$  имеет значение  $X_{ij}$  ( $j=1, k$ ) и  $e_i$  представляет ошибку в регрессионной модели и по предположению имеет нулевое математическое ожидание.

Эта модель подразумевает, что изменение в  $X_j$  имеет постоянный эффект на ожидаемый отклик  $Y$ :

$$\partial Y / \partial x_j = \beta_j \quad (j=1, k)$$

Более общая регрессионная модель постулирует, что эффект от фактора  $j$  зависит также от значения других факторов (с этим связан тот или иной вид нелинейности в регрессионных моделях).

Например, для случая  $k=3$  имеет место уравнение:

$$Y_i = \beta_0 + (\beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i3}) + \\ + \beta_{12} X_{i1} X_{i2} + \beta_{13} X_{i1} X_{i3} + \beta_{23} X_{i2} X_{i3} + \beta_{123} X_{i1} X_{i2} X_{i3} + e_i \quad (i=1, N),$$

где

вектор коэффициентов  $\beta$  содержит *суммарное среднее, главные эффекты, и эффекты взаимодействия*.

#### **Техника регрессионного анализа.**

Эксперимент, содержащий конечное число прогонов позволяет получить выборочные оценки для коэффициентов уравнения. Параметры полинома могут быть оценены и проверены на их значимость с помощью известной техники регрессионного анализа путем применением *обычного или обобщенного метода наименьших квадратов*.

Техника регрессионного анализа реализуется с помощью ряда процедур, включающих:

- *Проверку постулатов регрессионного анализа* (таких как проверка статистической гипотезы, о том что параметр оптимизации  $y$  – есть случайная величина с нормальным законом распределения, проверка однородности дисперсий и др.);

- *Расчет коэффициентов регрессии;*

- *Проверку адекватности модели*. Клейнен [20] рекомендует статистические проверки, основанные на использовании теста Стьюдента и F-теста. Если предполагаемая модель в результате окажется неприемлемой, то можно попробовать усложнить исходную модель путем добавлений двух- трех -факторных взаимодействий, или использовать преобразование исходных факторов, сужение экспериментальной области  $E$ , изменение интервалов варьирования и другие подходы;

- *Проверку значимости коэффициентов регрессии;*

– *Анализ коэффициентов регрессии.* Коэффициенты указывают на силу влияния факторов. Эффект фактора численно равен удвоенному коэффициенту, т.е. чем больше численная величина коэффициента, тем большее влияние оказывает фактор. Величина коэффициента соответствует вкладу данного фактора в величину параметра оптимизации при переходе фактора с нулевого уровня на верхний или на нижний. Если коэффициент "+", – то с увеличением значения фактора параметр оптимизации увеличивается, если "-", – то уменьшается.

Вы знаете, что модель может быть нелинейной. Следующий вопрос – как оценить нелинейность, пользуясь полным факторным экспериментом.

Часто встречающийся вид нелинейности связан с тем, что эффект одного фактора зависит от уровня, на котором находится другой фактор – так называемый *эффект взаимодействия*. При оптимизации стремятся сделать эффекты взаимодействия меньше, при интерполяции – важно их выявить.

Полный факторный эксперимент позволяет количественно оценивать эффекты взаимодействия, пользуясь правилом перемножения столбцов – получают столбец произведения двух факторов.

Матрица планирования полного факторного эксперимента  $2^2$  с учетом эффекта взаимодействия будет иметь вид:

*Таблица 7.5.2 – Матрица планирования полного факторного эксперимента  $2^2$  с учетом эффекта взаимодействия.*

N опыта	$X_1$	$X_2$	$(X_1 X_2)$	$y$
1	-1	-1	+1	$Y_1$
2	+1	-1	-1	$Y_2$
3	-1	+1	-1	$Y_3$
4	+1	+1	+1	$Y_4$

Все свойства матрицы планирования сохраняются, теперь модель выглядит:

$$Y = b_0 + b_1x_1 + b_2x_2 + b_{12}x_1x_2$$

$b_{12}$  вычисляется обычным образом.

Необходимо оценить 4 параметра (эффекта). Матрица планирования эксперимента составлена с помощью приема, развитого в теории

планирования эксперимента: последняя колонка получена умножением соответствующих элементов в столбцах  $X_1$  и  $X_2$ .

Таблица 7.5.3 – Полный факторный эксперимент  $2^3$ .

№ опыта	$X_1$	$X_2$	$X_3$	$X_1X_2$	$X_1X_3$	$X_2X_3$	$X_1X_2X_3$	$Y$
1	-	-	+	+	-	-	+	$Y_1$
2	+	-	-	-	-	+	+	$Y_2$
3	-	+	-	-	+	-	+	$Y_3$
4	+	+	+	+	+	+	+	$Y_4$
5	-	-	-	+	+	+	-	$Y_5$
6	+	-	+	-	+	-	-	$Y_6$
7	-	+	+	-	-	+	-	$Y_7$
8	+	+	-	+	-	-	-	$Y_8$

Теперь не составляет труда составить матрицу планирования для полного факторного эксперимента  $2^3$ : значения получаются перемножением столбцов (таблица 7.5.3.).

Эффект взаимодействия 2-х факторов называется эффектом взаимодействия 1-го порядка (парные эффекты взаимодействия); порядок на единицу меньше числа факторов. Здесь присутствуют тройные эффекты взаимодействия  $x_1x_2x_3$ .

### Неполный факторный анализ.

В полном факторном эксперименте разность между числом опытов и числом коэффициентов велика. Надо уменьшать число опытов. Эксперимент  $2^7$  уже содержит 128 прогонов (и это без повторений):  $2^7 = 128$

Таблица 7.5.4

k	2	3	4	5	6	7	8	9	10	...
N	4	8	16	32	64	128	256	512	1024	

С ростом числа факторов  $k$  число комбинаций переменных растет, растет число взаимодействий высокого порядка. Это демонстрируется в таблицах 7.5.4 и 7.5.5. (Полное число взаимодействий равно  $C_m^k$ , где  $k$  - число факторов,  $m$  - число элементов во взаимодействии).

Таблица 7.5.5

k	2 k	Число главных эффектов	Число взаимодействий				
			1й порядок	2й порядок	3й порядок	4..	5...
5	32	5	10	10	5	1	
6	64	6	15	20	15	6	1
7	128	7	21	35	35	21	7
8	256	8	28	56	70	56	28

При больших  $k$  появляются взаимодействия очень высоких порядков. Часто на основе априорных и общих соображений их предполагают пренебрежимо малыми. Т.к. взаимодействия высоких порядков соответствуют членам высокого порядка в регрессионном полиноме — члены высокого порядка полагаются равными нулю, и считается, что полином низкого порядка дает адекватное регрессионное уравнение. А если некоторые эффекты предполагаются равными нулю, то мы не обязаны делать наблюдение во всех 2 экспериментальных точках, можно сделать только часть точек — т.е. поставить *грубный факторный эксперимент*.

Оказывается, что если нас не интересуют взаимодействия высокого порядка (пренебрегаем некоторыми эффектами взаимодействия, в этом случае — члены полинома высокого порядка можем принять за 0), мы можем получить достаточное количество информации с помощью исследования лишь некоторой части (1/2, 1/4, 1/8 и т.д.) всех возможных комбинаций (реализовать часть экспериментальных точек или *дробный факторный эксперимент*).

*Неполным факторным планом* называется план эксперимента, если в факторном эксперименте производится лишь часть всех возможных повторений. Такой эксперимент называется *дробным факторным экспериментом*, а его матрица планирования — *дробной реликвой*.

Всякий раз, когда мы используем выборку меньшую, чем этого требует полный факторный план, мы платим за это риском смешивания эффектом. Под смешиванием мы понимаем то, что статистик, измеряя один эффект, в то же время измеряет, возможно, и некоторый другой эффект. Например, если главный эффект смешивается с взаимодействиями более высокого порядка, то эти два эффекта уже невозможно отделить друг от друга. Т.е., если наш анализ показывает наличие некоторого эффекта, то мы не можем с уверенностью сказать, главный ли это эффект, или эффект взаимодействия, или некоторая аддитивная комбинация этих эффектов.

Поэтому экспериментатору необходимо выбирать эффективную стратегию экспериментирования. При построении неполного факторного плана экспериментатор должен определить эффекты, смешивание которых он может допустить. Вообще говоря, лучше спутать взаимодействия высокого порядка, чем главные эффекты. Обычно, можно надеяться, что взаимодействия высокого порядка отсутствуют, и можно получить разумную информацию о главных эффектах или взаимодействиях низкого порядка. Эффективная стратегия экспериментирования или успех неполного факторного эксперимента достигается в случае, если его план позволяет не смешивать ни один главный эффект с другим.

Когда 2 или более эффекта смешиваются, то говорят, что они являются *совместными*.

Таблица 7.5.6 – Пример.

$X_1$	$X_2$	$X_3 = (X_1 X_2)$	$Y$
-	-	+	$Y_1$
+	-	-	$Y_2$
-	+	-	$Y_3$
+	+	+	$Y_4$

Пользуясь планом (табл.7.5.6.), можно вычислить 4 коэффициента и представить результаты:

$$Y = b_0 + b_1 x_1 + b_2 x_2 + b_{12} x_1 x_2 \quad (7.5)$$

Если предположить и считать, что в выбранных интервалах варьирования процесс может быть описан линейной моделью, то достаточно определить 3 коэффициента:  $b_0, b_1, b_2$ .

Оставшуюся степень свободы – используем для минимизации числа опытов. При  $b_{12} \rightarrow 0$  столбец  $x_1 x_2$  используется для нового фактора  $x_3$ . Но теперь оценки будут смешанные. Оценки смешаются следующим образом:

$$b_1 \rightarrow \beta_1 + \beta_{23}$$

$$b_2 \rightarrow \beta_2 + \beta_{13}$$

$$b_3 \rightarrow \beta_3 + \beta_{12}$$

Это так называемые *смешанные эффекты* или *эффекты, оцениваемые совместно* (вместе). Однако основные эффекты оцениваются раздельно друг от друга. Не огорчайтесь! Мы постулируем линейную модель, значит все парные взаимодействия незначимы.

$$b_1 \cong \beta_1 ; b_2 \cong \beta_2 ; b_3 \cong \beta_3.$$

Главное, что мы минимизировали число опытов – вместо 8 опытов для 3 факторов можно поставить 4. При этом матрица планирования не

потеряла своих свойств (ортогональность, ротатабельность и т.п.) – можете самостоятельно убедиться.

*Правило:* Чтобы сократить число опытов, нужно новому фактору присвоить вектор-столбец матрицы, принадлежащий взаимодействию, которым можно пренебречь. Тогда значение нового фактора в условиях опытов определяется знаками этого столбца.

Т.о., поставив 4 опыта для оценки влияния 3-х факторов, мы воспользовались половиной полного факторного эксперимента  $2^3$ , т.е. "полуреplikой".

Если  $x_3 = -x_1x_2$ , то получили бы 2-ую половину матрицы  $2^3$  (см. табл.7.5.3.), объединение этих 2-х полуреplik есть полный факторный эксперимент  $2^3$ . Есть две полуреплики  $2^{3-1}$ : 1 случай – когда  $x_3$  приравниваем к  $x_1x_2$ ; 2 случай –  $x_3$  приравниваем к  $-x_1x_2$ .

Для обозначения *гробных релик*, в которых  $p$  линейных эффектов приравнены к эффектам взаимодействия, используют условные обозначения  $2^{k-p}$ . Например, полуреплика от  $2^6$  обозначается  $2^{6-1}$ . Бывают реплики большей дробности (см. табл.7.5.7.). Например, при 15 факторах можно в 2048 раз сократить число опытов, применяя реплику большей дробности (поставить 16 опытов вместо 32768)

Таблица 7.5.7 – Примеры *гробных релик*.

		Число опытов	
		Дробная реплика	Полный факторный эксперимент
1/2 реплики от $2^4$	$2^{4-1}$	8	16
1/2 реплики от $2^5$	$2^{5-1}$	16	32
1/4 реплики от $2^6$	$2^{6-2}$	16	64
1/8 реплики от $2^6$	$2^{6-3}$	8	64
1/2048 реплики от $2^{15}$	$2^{15-11}$	16	32768

Символическое обозначение произведения столбцов, равного +1 или -1, называется *определяющим контрастом*.

Контраст помогает определять *смешанные эффекты*. Для того, чтобы определить какой эффект смешан с данным, нужно помножить обе части определяющего контраста на столбец, соответствующий данному эффекту.

Так, если  $I = x_1x_2x_3$ , то



$$X_1 = x_1^2 x_2 x_3 = x_2 x_3 \text{ (т.к. } x_1^2 = 1).$$

$$X_2 = x_1 x_2^2 x_3 = x_1 x_3$$

$$X_3 = x_1 x_2 x_3^2 = x_1 x_2$$

Это значит, что коэффициенты линейного уравнения будут оценками

$$b_1 \rightarrow \beta_1 + \beta_{23}$$

$$b_2 \rightarrow \beta_2 + \beta_{13}$$

$$b_3 \rightarrow \beta_3 + \beta_{12}$$

Соотношение, показывающее, с каким из эффектов смешан данный эффект, называется *генерирующим соотношением*.

Полуреплики, в которых основные эффекты смешаны с двухфакторными взаимодействиями, носят название планов с *разрешающей способностью III*. (по наибольшему числу факторов (символов) в определяющем контрасте). Обозначают такие планы  $2^{3-1}_{III}$ . Разрешающая способность плана равна наименьшему числу символов в коде определяющего контраста.

Неполный факторный анализ используется на начальной стадии исследования в случае наличия больше 4 факторов, когда необходимо выявить наиболее существенные переменные. Если число переменных меньше 4 – ставят полный факторный эксперимент.

Итак, резюмируем наше рассуждение:

- Дробные реплики применяются при получении линейных моделей.
- Эффективность применения дробных реплик зависит от удачного выбора системы смешивания линейных эффектов с эффектами взаимодействия.
- При построении дробных реплик используют *правило*: для того, чтобы сократить число опытов при введении в планирование нового фактора, нужно поместить этот фактор в вектор столбец матрицы, принадлежащий взаимодействию, которым можно пренебречь.
- Реплики, которые используются для сокращения опытов в  $2^m$  раз, где  $m = 1, 2, 3, 4, \dots$  называются *регулярными* и пользуются популярностью, т.к. расчет коэффициентов производится также просто, как в случае полного факторного эксперимента (см. техника регрессионного анализа).
- При применении дробных реплик линейные эффекты смешиваются с эффектами взаимодействия. Чтобы определить систему смешивания, надо знать *определяющие контрасты и генерирующие соотношения*.

- Эффективность реплики зависит от системы смешивания реплики, у которых линейные эффекты смешаны с взаимодействиями наивысшего порядка, являются наиболее эффективными, т.к. обладают наибольшей разрешающей способностью.

## 7.6 Основные классы планов, применяемые в вычислительном эксперименте.

Приведем краткий обзор планов, используемых в имитационном эксперименте.

*По методу анализа и виду математической модели различают:*

- планы дисперсионного анализа (однофакторный, многофакторный);
- планы регрессионного анализа;
- планы ковариационного анализа.

*Планы многофакторного анализа:*

- двухуровневые,
- многоуровневые;
- симметричные,
- несимметричные.

В практике машинного эксперимента полезны следующие виды планов:

### 1. Планы многофакторного анализа:

*Планы типа  $2^{k-p}$ .* Все  $k$  факторов имеют 2 уровня, используется часть всех комбинаций (дробная реплика) — существует возможность оценить главные эффекты факторов и взаимодействия низкого порядка.

Определяют следующие типы планов:

- *Планы разрешающей способности III:*

ни один главный эффект не смешан ни с каким другим главным эффектом, но главные эффекты смешаны с двухфакторными взаимодействиями, которые смешаны друг с другом.

- *Планы разрешающей способности IV:*

ни один главный эффект не смешан с другим главным эффектом или взаимодействием двух факторов, но эти взаимодействия смешаны друг с другом.

- *Планы разрешающей способности V:*

ни один главный эффект и ни одно взаимодействие 2-х факторов не смешаны с другими главными эффектами или двухфакторными взаимодействиями, но эти взаимодействия смешаны с взаимодействиями трех факторов.

В общем случае *разрешающая* способность плана равна наименьшему числу символов в коде определяющего контраста.

## 2. Планы отсеивающего эксперимента:

Если  $k$  велико, число комбинаций все-таки остается большим даже при неполном факторном плане ( $2^{k-p}$ ), -тогда используются *планы отсеивающих экспериментов*. "Отсеивающий эксперимент" предполагает предварительное отсеивание, определение наиболее важных (существенных) факторов и используется на стадии предварительного исследования.

### 2.1. Случайные планы.

Комбинации уровней факторов случайно (рандомизация) отбираются среди всех возможных комбинаций. Число комбинаций  $N$  может быть определено независимо от числа факторов и уровней. ( $N$  может быть, например,  $< k$ ). Могут быть получены "хорошие" оценки индивидуальных эффектов, не требуется специальных методов анализа – используются традиционные ДАН и регрессионный анализ.

Если очень много факторов – можно использовать сверхнасыщенные планы:

2.2. *Сверхнасыщенные планы*. Если реплики от планов  $2^k$  насыщены, то они содержат как раз столько опытов, сколько эффектов надо оценить. Сверхнасыщенные планы – когда число комбинаций  $N$  меньше, чем число факторов  $k$ , и комбинации отбираются так, чтобы (для данных  $N$  и  $k$ ) оценки эффектов были достаточно "хорошими".

2.3. *Планы группового отсеивания (последовательного отсеивания)*.  $k$  факторов разбиваются на  $q$  групп ( $q \ll k$ ) и эти  $q$  групп факторов испытываются в плане типа  $2^{k-p}$  или в сверхнасыщенном плане. Та группа факторов, которая оказалась наиболее важной, разбивается на несколько групп меньшей размерности (в конечном итоге до групп размерности 1).

## 3. Планы для изучения поверхности отклика

В имитационном моделировании используется технология *последовательного планирования*.

На практике, например, может быть использована следующая схема проведения эксперимента: допустим, сначала реализуется план  $2^{k-p}$ -используется малая часть эксперимента; если оказывается его недостаточно (часть эксперимента слишком мала для оценки всех эффектов) – необходимо уменьшить дробность плана, дополнить план эксперимента ("методом перевала") – расширить эксперимент до полного факторного эксперимента- и т. д.

Аналогичные типы планов, основанные на последовательном планировании используются в методологии анализа поверхности отклика (см. раздел 7.7.), которая состоит в отыскании оптимальной комбинации уровней  $k$  количественных факторов.

Рассмотрим некоторые планы, полезные для анализа поверхности отклика.

Дело в том, что в почти стационарной области желательно аппроксимировать поверхность отклика по меньшей мере полиномом второго порядка. Для этой цели проводится эксперимент *квадратичного приближения*. Ниже приведен общий вид квадратичного полинома (второго порядка) для случая двух независимых переменных

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_{11} x_1^2 + a_{22} x_2^2 + a_{12} x_1 x_2$$

Для оценивания коэффициентов регрессии в этой модели необходимо измерить каждый фактор или переменную по меньшей мере на *трех* уровнях. Это означает необходимость использования полного  $3^n$ -факторного эксперимента или же неполного факторного эксперимента. К сожалению, в случае более чем трех переменных число необходимых прогонов может стать слишком большим даже при использовании неполного факторного эксперимента. Кроме того, эксперимент с  $3^n$  комбинациями дает довольно низкую точность оценок коэффициентов регрессии.

Для квадратичных полиномов используют специальные способы построения эксперимента. Из них наиболее полезны *центральные композиционные*, или *ротатабельные*, построения, которые получаются посредством добавления дополнительных точек к данным, полученным из  $2^n$ -факторного эксперимента. Для ротатабельных построений стандартная ошибка одинакова для равноудаленных от центра области точек. Такие построения существуют для любого числа факторов и представляют собой регулярные и полурегулярные геометрические фигуры с центральными точками.

Весьма простое и полезное построение, называемое *куб плюс звезда* плюс *центральные точки*, приведено для случая двух переменных на рисунке 7.6.1. Отметим, что это  $2^2$  эксперимент (куб), плюс четыре дополнительные точки (звезда), расположенные на равном расстоянии друг от друга в плоскости по окружности с центром  $(0,0)$ , плюс центральные точки. Тот же самый принцип, но применительно к 3-факторному эксперименту, иллюстрируется на рис. 7.6.2.

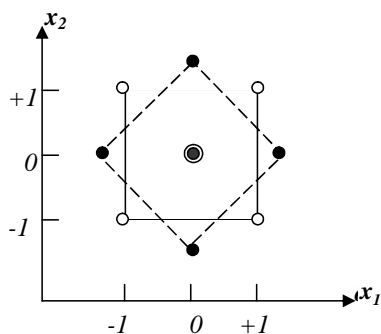


Рисунок 7.6.1 – 2<sup>2</sup>-факторный план плюс звезда и центральные точки.

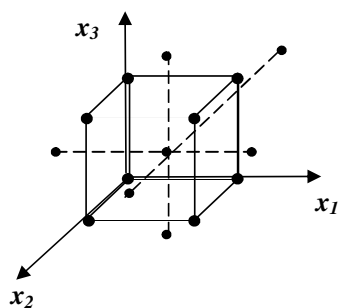


Рисунок 7.6.2 – 3<sup>2</sup>-факторный план плюс звезда и центральные точки.

Другое полезное ротатабельное построение, называемое *гексагональ плюс центральные точки*, приведено на рисунке 7.6.3.

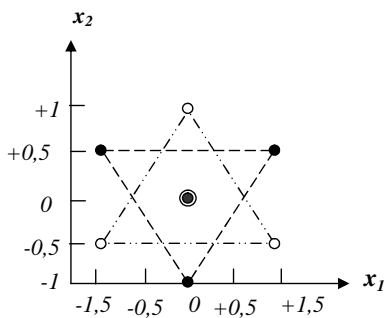


Рисунок 7.6.3 – Гексагональный план.

Известны и другие типы планов: латинские квадраты, квадраты Юдена, сбалансированные блоки и др.

На практике может быть полезен план Бокса В2 в кодированном двухфакторном пространстве, иллюстрированный на рисунке 7.6.4.

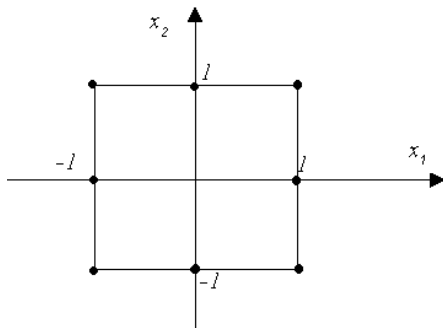


Рисунок 7.6.4 – План Бокса В2.

## 7.7 Методология анализа поверхности отклика. Техника расчета крутого восхождения.

Методология анализа поверхности отклика полезна для реализации 3 типа вычислительного эксперимента, связанного с отысканием оптимальных условий.

Во многих случаях целью моделирования является отыскание таких величин или уровней независимых переменных, при которых отклик или зависимая переменная достигает оптимальных (максимальных или минимальных) значений. Если зависимая и независимая переменные количественны и непрерывны, то для решения задачи поиска оптимума обычно используется *методология поверхности отклика* (RSO), которая состоит в отыскании оптимальной комбинации уровней  $k$ -количественных факторов.

Если обозначить зависимую переменную через  $y$ , а независимые переменные через  $(x_1, x_2, \dots, x_k)$ , где  $k$  – число факторов, и предположить, что все переменные *количественны, непрерывны* и измеримы, то уравнение поверхности отклика можно записать в следующем виде:

$$y = F(x_1, x_2, \dots, x_k).$$

Очень полезно иметь геометрическое представление поверхности отклика. На рисунке 7.7.1 изображена поверхность отклика,

изображающая зависимость отклика  $y$  от двух независимых переменных  $x_1$  и  $x_2$  в прямоугольной системе координат.

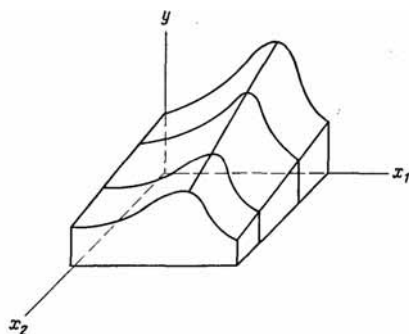


Рисунок 7.7.1 – Поверхность отклика.

Другое полезное представление поверхности отклика можно получить, изображая на плоскости  $x_1$   $x_2$  проекции на нее линий постоянного отклика  $y=y_0$ , проведенных на поверхности отклика (контуры отклика). Ряд примеров подобных представлений приведен на рисунках 7.7.2, 7.7.3, 7.7.4, 7.7.5.

(Такое изображение поверхностей отклика аналогично изображению на топографических картах рельефа местности посредством контуров равной высоты или изображению на картах погоды изобар, показывающих распределение атмосферного давления на поверхности Земли.)

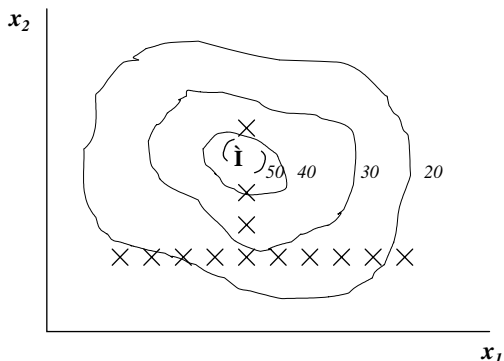


Рисунок 7.7.2 -Типичные контуры отклика: "Холм".

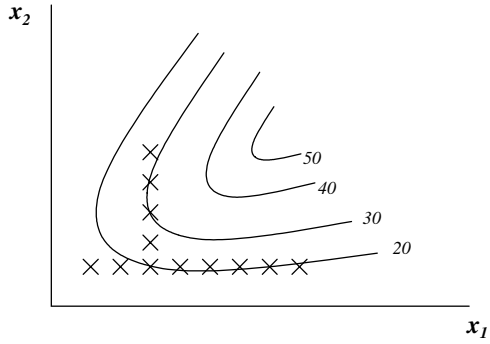


Рисунок 7.7.3 – Типичные контуры отклика: "Поднимающийся гребень".

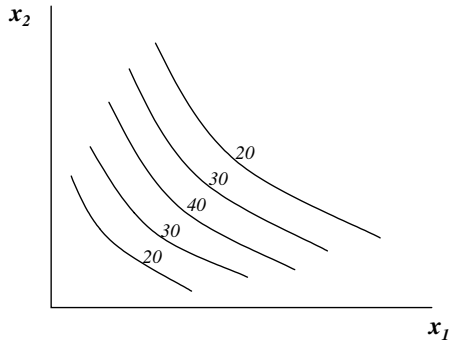


Рисунок 7.7.4 – Типичные контуры отклика: "Гребень постоянной высоты".

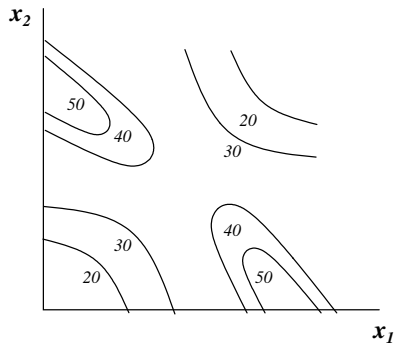


Рисунок 7.7.5 – Типичные контуры отклика: "Минимум", или "Седло".



На рисунке 7.7.2. точка  $M$  — та оптимальная точка, которую мы ищем. Каждая линия соответствует постоянному значению параметра оптимизации и является проекцией сечений поверхности отклика на плоскость. Такая линия называется *линией равного отклика*.

Безусловно, геометрические представления поверхности отклика ограничены трехмерными изображениями. Однако геометрическое представление трехмерных поверхностей отклика вида  $y=f(x_1, x_2)$  помогает понять, что происходит в более общем случае — при  $k$  независимых переменных. Поэтому дальше мы ограничимся рассмотрением трехмерных поверхностей отклика.

*Методология поверхности отклика* обычно основана на исследовании поверхности отклика с помощью ряда небольших полных или неполных факторных экспериментов. В имитационном моделировании используется технология *последовательного планирования*. Такого рода последовательные планы используются в методологии анализа поверхности отклика (*RSO*), которая состоит в отыскании оптимальной комбинации уровней  $k$  количественных факторов, где реализуется пошаговая процедура движения к точке оптимума.

Сделаем небольшое пояснение. Если известно точное математическое выражение функции отклика, то отыскание оптимальной точки можно сравнительно просто осуществить аналитическими методами. Так как обычно мы не знаем вид поверхности отклика, то необходимо использовать в качестве аппроксимации какую-либо гибкую, плавно изменяющуюся функцию. В качестве такой функции обычно используют полином первого порядка или полином второго порядка, где коэффициенты этого полинома оцениваются с помощью эксперимента.

В этом случае наша задача состоит в том, чтобы

- как можно быстрее выйти в близкую к оптимуму область,
- а затем воспользоваться аналитическими методами локального представления этой функции в окрестности точки оптимума.

Эти эксперименты могут использоваться для разрешения *двух вопросов*.

Первый из них связан с выбором такого направления перемещения для проведения следующего эксперимента, чтобы приблизиться к оптимальной точке. К точке оптимума (вернее в область оптимума) надо придти наискорейшим образом — в *RSO* для этого используется метод наискорейшего подъема (или *Крутое восхождение по поверхности отклика*). С этой целью реализуются планы  $2^k$  или  $2^{k-p}$ , используется линейная регрессия.

Второй вопрос возникает тогда, когда мы уже находимся в достаточной близости к точке экстремума. Этот второй вопрос — *вопрос о виде уравнения поверхности отклика вблизи оптимальной точки*. Можно воспользоваться аналитическими методами для локального представления этой функции в окрестности точки оптимума. Вблизи точки оптимума линейная аппроксимация перестает быть удовлетворительной, и используется полином более высокого порядка. Для этого необходимо использовать планы  $Z^*$  и др., — и квадратичные полиномы.

Предлагаемая концепция оптимизации условно включает 2 этапа:

*1 этап оптимизации* — крутое восхождение с целью скорейшего достижения области оптимума. Используется линейное планирование и реализуется пошаговая итерационная процедура движения в направлении градиента.

Однако область оптимума не может быть описана линейным приближением.

*2 этап* — описание области оптимума методами нелинейного планирования и исследование почти стационарной области.

Давайте подробнее рассмотрим эти 2 задачи.

### **Крутое восхождение по поверхности отклика.**

Одним из традиционных методов поиска оптимума (максимума) является *метод покоординатного подъема*. Как показано на рисунке 7.7.2, если зафиксировать  $x_2$  и варьировать  $x_1$ , то мы можем найти максимум по  $x_1$  при фиксированном значении  $x_2$ . Зафиксировав  $x_1$  в найденной точке максимума, будем искать максимум по  $x_2$ . В случае поверхности отклика как на рисунке 7.7.2, этот метод за два шага приведет в близкую к оптимуму точку. Однако в случае поверхности вида рисунка 7.7.3 этот метод не приведет в точку оптимума. Обычно вид поверхности отклика неизвестен, поэтому желательно иметь такой метод поиска оптимума, который работал бы при неизвестной форме поверхности отклика.

Наиболее часто используется в МПО так называемый *метод наискорейшего подъема* — *крутого восхождения*. Наиболее короткий путь к вершине — направление градиента функции отклика, на рисунке 7.7.6. — это направление  $Q$ , перпендикулярное линиям уровня. Это так называемый градиентный метод поиска оптимума. Вы знаете, что, градиент непрерывной однозначной функции

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \dots + \beta_k X_{ik} + e_i \quad (i=1, N)$$

есть частная производная функции  $Y$

$$\partial Y / \partial x_j = \beta_j \quad (j=1, k)$$

Составляющие градиента суть частные производные функции отклика, оценками которых являются коэффициенты регрессии.

*Основная идея метода* состоит в построении линейной аппроксимации поверхности отклика в окрестности данной точки с помощью простого факторного эксперимента. По построенной линейной функции определяется направление наискорейшего подъема к точке оптимума — в направлении градиента функции отклика. По этому направлению делается небольшой шаг, затем повторяется процедура определения направления наискорейшего подъема и т.д.

Этот метод не позволяет определять длину шага (длина шага определяется экспериментатором, она не должна быть очень большой, чтобы не проскочить оптимум, и не должна быть очень маленькой — чтобы не проводить большое количество шагов), однако показывает направление движения.

Рассмотрим пример, приведенный на рисунке 7.7.6.

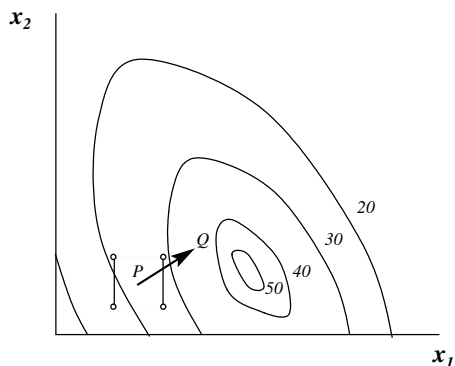


Рисунок 7.7.6 — Контуры отклика с изображением  $2^2$ -факторного плана.

Предположим, что исследователь провел в точке  $P$  эксперимент с  $2^2$  комбинациями плюс два наблюдения в центре. Эксперимент позволил оценить  $a_0$ ,  $a_1$  и  $a_2$ , определяющие наклон плоскости аппроксимации, и вычислить направление наискорейшего возрастания или максимального наклона плоскости. Это направление показывает относительные величины изменения факторов, обеспечивающие увеличение отклика. Продвинувшись по этому направлению до некоторой точки  $Q$ , следует повторить всю процедуру.

Такая пошаговая процедура позволяет достигать все больших и больших откликов.

Итак, изменяя независимые переменные пропорционально коэффициентам регрессии (с учетом их знаков), мы будем двигаться в

направлении градиента функции отклика по самому крутому пути. Поэтому процедура движения к почти стационарной области называется *крутым восхождением*.

#### Техника расчета крутого восхождения.

Технику расчета крутого восхождения удобно рассмотреть для простейшего случая — одного фактора.

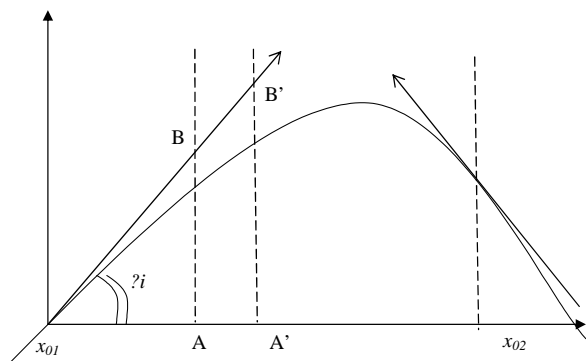


Рисунок 7.7.7 — Расчет координат точек в направлении градиента.

Значение коэффициента регрессии равно  $\operatorname{tg}$  угла наклона между линией регрессии и осью данного фактора. Если  $\operatorname{tg}$  умножить на интервал варьирования  $OA$ , то получим катет  $AB$ , где  $B$  — координаты точки, лежащие на градиенте. Обобщение на  $k$ -факторов делается механически, коэффициенты умножаются на соответствующие интервалы. Получаются точки, лежащие на том же градиенте. Различны только шаги — они получаются, если к нулевому уровню последовательно прибавлять величины, пропорциональные составляющим градиента.

Шаг движения по градиенту выбирается экспериментатором, если сделаете его минимальным — придется проводить много опытов при движении к оптимуму, максимальным — можно проскочить оптимум.

Величины составляющих градиента определяются формой поверхности отклика и тем, как выбран параметр оптимизации, нулевая точка и интервалы варьирования. Знак составляющих градиента зависит от формы поверхности отклика и положения нулевой точки  $x_0$ .

Итак, если для данной поверхности отклика (предполагается, что один оптимум)

- выбраны нулевая точка и интервалы варьирования,
- проведен эксперимент,
- оценены коэффициенты регрессии,

- то направление градиента задается однозначно и является единственным.

**Правило:** *расчет крутого восхождения.* В крутом восхождении независимые переменные изменяют пропорционально величинам коэффициентов регрессии и с учетом их знаков. Составляющие градиента однозначно получаются умножением коэффициентов регрессии на интервалы варьирования по каждому фактору. Серия опытов (называются мысленными) в направлении градиента рассчитывается последовательным прибавлением к основному уровню факторов величин, пропорциональных составляющим градиента.

Далее в эксперименте реализуются мысленные опыты (обычно только те, условия которых выходят за область эксперимента хотя бы по одному фактору). Потом выполняется анализ на достижение оптимума, а далее:

- окончание исследования (канонический анализ), если оптимум найден, или
- последовательный поиск (еще раз делается шаг крутого восхождения).

*Крутое восхождение эффективно*, если реализация мысленных опытов на стадии крутого восхождения улучшило значение параметра оптимизации по-сравнению с самым хорошим результатом в матрице планирования. *Если крутое восхождение эффективно*, то можно:

- или окончить исследование (в зависимости от того как поставлена задача оптимизации),
- или совершить следующий цикл крутого восхождения;
- или исследовать почти стационарную область (область оптимума) методами нелинейного планирования (построить план 2-го порядка для исследования области оптимума).

Если *крутое восхождение неэффективно*, — можно построить линейный план нового цикла.

Итак, сначала исследование проводим внутри малой подобласти. Выбираем некоторую точку в факторном пространстве. И в данной точке факторного пространства планируем эксперимент — далее интерполяция с помощью линейного полинома. А затем экстраполируем (вне нашей подобласти), делаем небольшой шаг и проводим эксперимент, и цикл повторяется до тех пор, пока движение по градиенту не перестанет быть эффективным. Это значит, что мы попали в область, близкую к оптимуму. Такая область называется *“почти стационарной”*.

Таким образом, процедура поиска оптимума основана на шаговом принципе: проводятся короткие (небольшие, насколько возможно) серии прогонов, по их результатам строят математическую модель (линейную регрессию). Полученную модель используют для оценки градиента

(направления движения к оптимуму). А далее делается небольшой шаг в направлении градиента и ставят новые эксперименты в этом направлении и т.д. Получается циклический процесс, который заканчивается при попадании в область, близкую к оптимуму ("почти стационарную" область).

Многошаговая итерационная процедура, реализуемая методологией анализа поверхности отклика условно включает следующие этапы:

1. Планирование эксперимента, т.е. выбор комбинаций уровней факторов. Методика использует полные и дробные типа  $2^{k-p}$  планы эксперимента и некоторые специальные планы.
2. Подбор уравнения регрессии по наблюдениям. Отклик рассматривается как функция независимых переменных. Уравнение линейной регрессии описывает поверхность отклика в некоторой точке (нулевой или центральной).
3. "Влезание" по поверхности отклика к вершине. Для нахождения направления увеличения отклика служит метод "крутого восхождения". В направлении, где ожидается увеличение отклика, этапы 1,2,3 повторяются до тех пор, пока не будет достигнута область максимума.
4. Исследование почти стационарной области с помощью полиномов 2-го порядка (см. в разделе 7.6. специальные планы для анализа поверхности отклика). Канонический анализ в области максимума функции отклика. Канонический анализ показывает, имеется ли один максимум, несколько максимумов, седловая точка или гребень.

Давайте теперь подробнее остановимся на заключительных этапах, связанных с анализом "почти стационарной" области.

Хотя последовательное использование эксперимента с  $2^n$  комбинациями и измерениями в центральной точке не дает возможность точно определить точку оптимума, оно, однако, позволит исследователю быстро найти окрестность точки оптимума, которую можно назвать *почти стационарной областью*. Однако вблизи точки оптимума описанная пошаговая процедура уже не эффективна. Вблизи точки оптимума линейная аппроксимация перестает быть удовлетворительной, и для аппроксимации необходимо брать полином более высокого порядка. По достижении такой области необходимо исследовать вид в ней поверхности отклика, чтобы определить более точно оптимальную точку. Для этой цели обычно используются аппроксимации полиномами второго и более высокого порядка. Переход к полиномам высокого порядка вызывается существенной потерей точности аппроксимации полиномом первого порядка.

*Канонический анализ.* Тот факт, что достигнута почти стационарная область, не означает, что мы находимся изблизи точки максимума, как это имеет место на рисунке 7.7.2 (также см. рис. 7.7.8). Мы можем при этом находиться на медленно поднимающемся гребне, как на рисунке 7.7.3 (также см. рис. 7.7.9), или на гребне постоянной высоты, как на рисунке 7.7.4 (также см. рис. 7.7.10). При движении по градиенту возможны различные ситуации. Наиболее благоприятны случаи:

- параметр оптимизации проходит через максимум (рис.7.7.8.),
- параметр оптимизации все время возрастает (рис.7.7.9.),
- более сложный случай: нарушена предпосылка одноэкстремальности.

Возможно, что мы достигли окрестности минимаксной точки (иногда называемой седловой точкой), которая является точкой максимума по одним направлениям и точкой минимума по другим (рис. 7.7.5). Появление седловой точки может сообщить нам полезную информацию о том, что наша поверхность отклика имеет два или более максимума, и, следуя методу скорейшего подъема, мы можем найти не глобальный, а некоторый локальный, второстепенный максимум (также см. рис. 7.7.11).

Метод наискорейшего подъема не гарантирует сходимость в точку глобального максимума. В какую именно точку максимума сойдется построенная с его помощью последовательность, зависит от начальных условий эксперимента. Если предполагается наличие нескольких максимумов, то один из основных способов отыскания глобального максимума — это *повторное применение метода наискорейшего подъема с меняющимися в широкой области начальными условиями*. То есть надо продвигнуться в другую точку факторного пространства, построить линейный план нового цикла крутого восхождения. Хотя такой случай и возможен, в реальных условиях обычно имеется единственный максимум, и это упрощает дело.

Неэффективны случаи:

- наблюдается уменьшение параметра оптимизации (возможно есть ошибки в расчетах),
- все опыты на градиенте имеют одно и то же значение (рис. 7.7.10.). Поверхность отклика имеет вид *постоянного гребня*.

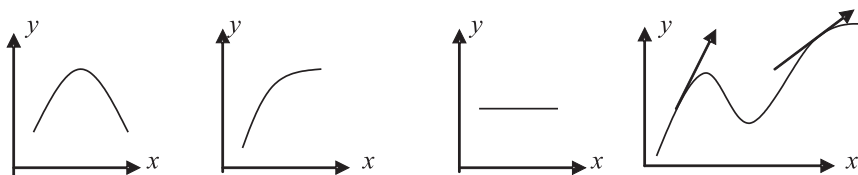


Рисунок 7.7.8. Рисунок 7.7.9. Рисунок 7.7.10 Рисунок 7.7.11.

## СПИСОК ЛИТЕРАТУРЫ.

1. Balci O. Credibility Assessment of Simulation Results//Proceedings of the 1986 Winter Simulation Conference. – 1986. – pp. 39-44.
2. Balci O. (1994) Validation, Verification and Testing Techniques Throughout the Life Cycle of a Simulation Study, *Annals of Operation Research*.
3. Balci O. Verification, validation and accreditation//Proceedings of the 1998 Winter Simulation Conference.-1998. – pp. 41-48.
4. Carson J.S. Model verification and validation//Proceedings of the 2002 Winter Simulation Conference. – 2002. – pp. 52-58.
5. Henriksen J. An Introduction to SLX Proceedings of the 1997 Winter Simulation Conference IEEE 1997 p. 593-599.
6. Law A.M., McComas, M.G. How to build valid and credible simulation models//Proceedings of the 2001 Winter Simulation Conference.- 2001. – pp. 22-29.
7. Sargent R.G. Some approaches and paradigms for verifying and validating simulation models//Proceedings of the 2001 Winter Simulation Conference. – 2001.- pp, 106-114.
8. Schriber T. Introduction to Simulation Using GPSS/H N4 John Willey & Sons 1991.
9. W. David Kelton, Randall P. Sadowski, Deborah A. Sadowski Simulation with Arena – WCB/McGraw-Hill, 1998.
10. Бахвалов Л.А. Компьютерное моделирование: долгий путь к сияющим вершинам, Компьютерра. 1997, № 40. с.26-36.
11. Бусленко Н.П. Моделирование сложных систем. – М.: Наука, 1978, 400 с.
12. Варжапетян А.Г. Имитационное моделирование на GPSS/H. – М.: Вузовская книга, 2004.
13. Имитационное моделирование в задачах синтеза структуры сложных систем (оптимизационно-имитационный подход)./Цвиркун А.Д., Акинфиев В.К. и др. – М.: Наука, 1985.- 176 с.
14. Имитационное моделирование производственных систем/Под общей редакцией А.А. Вавилова. – М.: Машиностроение; Берлин: Техника, 1983.- 416 с.
15. Имитационное моделирование социально-экономических систем. Алексеев Ю.Н., Биткова Г.В. и др. – 1986.
16. Имитационные системы принятия экономических решений/К.А. Багриновский и др. – М.: Наука, 1989.



17. Имитационный анализ регионального воспроизводства./Ответственный редактор доктор физ.-мат. наук Булавский В.Г. — Новосибирск: Наука, сибирское отделение, 1987г. -176 с.
18. Калашников В.В. Организация моделирования сложных систем. — М.: Знание, 1982, 62 с.
19. Киндлер Е. Языки моделирования, пер. с чешск. М.: Энергоатомиздат, 1985. 288 с.
20. Клейнен Дж. Статистические методы в имитационном моделировании — 1,2 т.- М.: Статистика, 1978.
21. Клир Дж. Системология. Автоматизация решения системных задач. — М.: Радио и связь, 1990.
22. Коблев Н.Б. Основы имитационного моделирования сложных экономических систем. — М.: Дело, 2003.
23. Кугаенко А.А. Основы теории и практики динамического моделирования социально-экономических объектов и прогнозирования их развития. — М.: Вузовская книга 1998. — 392с.
24. Лекции по теории сложных систем. Бусленко Н.П., Калашников В.В. и др. — М.: Сов. радио, 1973.- 440 с.
25. Лоу А.М., Кельтон В.Д. Имитационное моделирование. Классика CS — 3-е издание — Питер, 2004.
26. Лычкина Н.Н. Методические указания к курсовому проектированию по дисциплине "Имитационное моделирование экономических процессов", ГУУ, — М.: 2004.
27. Лычкина Н.Н. Имитационное моделирование социально-экономического развития регионов/Первая всероссийская научно-практическая конференция "Опыт практического применения языков и программных систем имитационного моделирования в промышленности и прикладных разработках" ИММОД — 2003, Санкт-Петербург, 2003.
28. Лычкина Н.Н. Компьютерное моделирование социально-экономического развития регионов в системах поддержки принятия решений,/III Международная конференция "Идентификация систем и задачи управления" SICPRO'04, М., 2004.
29. Лычкина Н.Н. Моделирование социально-экономического развития регионов./Материалы научно-практического семинара кафедры информационных систем./Под ред. Ю.М. Черкасова; ГУУ, М., 2001.
30. Лычкина Н.Н. Системы принятия решений в задачах социально-экономического развития регионов. Компьюлог, № 2 (32), М., 1999.

31. Лычкина Н.Н. Современные тенденции в имитационном моделировании – “Вестник университета”, серия “Информационные системы управления” № 2, ГУУ, М., 2000.
32. Лычкина Н.Н. Технологические возможности современных систем моделирования./Банковские технологии, Выпуск 9, М., 2000.
33. Максимей И.В. Имитационное моделирование на ЭВМ. – М.: Радио и связь, 1988.- 232с.
34. Методы и модели управления проектами в металлургии. Смирнов В.С., Власов С.А. и др. – М.: СИНТЕГ, 2001.- 176 с.
35. Нейлор Т. Машинные имитационные эксперименты с моделями экономических систем./Пер. с англ. – М.: Мир, 1975.- 502 с.
36. Перегудов Ф.И. Тарасенко Ф.П. Введение в системный анализ: учеб. пособие для ВУЗов. – М.: Высш.шк., 1989.- 367 с.
37. Планирование экспериментов при поиске оптимальных условий./Адлер Ю.П. и др. – М.: “Наука”, 1971.
38. Прангишвили И.В. Системный подход и общесистемные закономерности. – М.: СИНТЕГ, 2000.- 528 с.
39. Прицкер А. Введение в имитационное моделирование и язык СЛАМ II/пер. с англ. – М: Мир, 1987.- 646 с.
40. Ресин В.И., Попков Ю.С. Развитие больших городов в условиях переходной экономики. Системный подход. – М.: Эдиториал УРСС, 2000.
41. Руководство пользователя по GPSS World./Перевод с английского/. – Казань: Изд. “Мастер Лайн”, 2002. – 384 с.
42. Рыжиков Д.И. Имитационное моделирование систем массового обслуживания. – Л.: ВИККИ им А.Ф. Можайского, 1991.- 111 с.
43. Системный анализ и проблемы развития городов. Попков Ю.С., Посохин М.В. и др. – 1983.
44. Советов Б.Я., Яковлев С.А. Моделирование систем.- М.: Высшая школа, 2003, 320с.
45. Советов Б.Я., Яковлев С.А. Моделирование систем: учебник для ВУЗов. – М.: Высшая школа, 1985 (1999).- 224 с.
46. Социальные системы. Формализация и компьютерное моделирование: учебное пособие./А.К. Гуц, В.В. Коробицын и др. – Омск: Омский Государственный Университет, 2000. – 160 с.
47. Технология системного моделирования./Под общей редакцией Емельянова С.В. – М.: Машиностроение, 1998.
48. Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. – М: Бестселлер, 2003. – 416 с.

49. Трахтенгерц Э.А. Компьютерная поддержка принятия решений: Научно-практическое издание. Серия "Информатизация России на пороге XXI века". - М.: СИНТЕГ, 1998.
50. Трахтенгерц Э.А. Субъективность в компьютерной поддержке управленческих решений. — М.: СИНТЕГ, 2001.- 256 с.
51. Форрестер Дж. Динамика развития города. — М.: Прогресс, 1974.
52. Форрестер Дж. Мировая динамика. — М.: Наука, 1978.
53. Форрестер Дж. Основы кибернетики предприятия (индустриальная динамика)/пер. с англ., общая редакция Д.М. Гвишиани — М: Прогресс, 1971.- 340 с.
54. Шеннон Р. Имитационное моделирование систем — искусство и наука. — М.: Мир, 1978
55. Шрайбер Т. Дж. Моделирование на GPSS — М.: Машиностроение, 1980.- 592 с.
56. Яцкив И.В. Проблема валидации имитационной модели и ее возможные решения. — Материалы конференции ИММОД -2003, 2003г. с. 211-217.

