

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Федоров О.В.

**МОДЕЛИРОВАНИЕ
ЭНЕРГОМЕХАНИЧЕСКИХ СИСТЕМ**

конспект лекций

Р а с с м о т р е н о
на заседании кафедры
«Энергомеханические системы».
Протокол № 8 от 27.04.2017 г.

Донецк – 2017

Федоров О.В. Конспект лекций по дисциплине «Моделирование энергомеханических систем» /О.В. Федоров. – Донецк: ГОУВПО «ДОННТУ», 2017. – 60 с.

Рассмотрены вопросы моделирования технических систем, свойства моделей и требования к ним, элементы теории подобия, выбора параметров и факторов моделей. Изложены основные методы моделирования: разработка модели, ее реализация и верификация. Проанализированы различные подходы и типы моделей, используемые в моделировании технических систем. Рассмотрено имитационное моделирование технических систем.

Составитель: О. В. Федоров, доцент, к.т.н.

Рецензент:

Отв. за выпуск: А. П. Кононенко, проф, д.т.н., заведующий кафедрой «Энергомеханические системы»

Модели и моделирование

1. Общие сведения о моделировании

Модель - это упрощенное представление исследуемого объекта (системы). Она должна отражать наиболее существенные (с точки зрения цели исследования) свойства изучаемого объекта. Объект, для которого создается модель, принято называть оригиналом. Не всегда есть возможность или целесообразность исследовать непосредственно оригинал. Часто исследования проводят на модели оригинала, а затем результаты исследования переносят на оригинал. В зависимости от цели исследования для одного и того же оригинала может быть создано несколько моделей.

Моделирование можно рассматривать как замещение исследуемого объекта (оригинала) его условным образом, описанием или другим объектом, именуемым *моделью* и обеспечивающим адекватное с оригиналом поведение в рамках некоторых допущений и приемлемых погрешностей. Моделирование обычно выполняется с целью познания свойств оригинала путем исследования его модели, а не самого объекта. Разумеется, моделирование оправдано в том случае, когда оно проще создания самого оригинала или когда последний по каким-то причинам лучше вообще не создавать.

Реальная польза от моделирования может быть получена при выполнении **двух главных условий**:

1) **модель должна быть адекватной оригиналу** в том смысле, что должна с достаточной точностью отображать интересующие исследователя характеристики оригинала;

2) **модель должна устранять проблемы**, связанные с физическим измерением каких-то физических величин или характеристик оригинала.

Следует отметить, что в большинстве случаев моделирование вовсе не заменяет реальный объект и не отменяет необходимости в его разработке и натурном испытании. Оно просто значительно уменьшает объем работ по проектированию и исследованию объектов. В тех же случаях, когда это не так, стоимость моделирования может оказаться вполне сравнимой со стоимостью разработок и натурных испытаний изделий.

Итак, моделирование - это процесс создания модели, её исследование и обработка полученных результатов. Чаще всего

моделированием занимаются при анализе существующих объектов (систем) или в процессе проектирования таких объектов (систем).

Моделирование позволяет анализировать даже ситуации, в которых реальный объект не может функционировать. Можно моделировать, например, катастрофы, редчайшие случаи и даже такие явления и процессы, которых не существует в реальности.

Сейчас методы моделирования широко используют во всех сферах деятельности человека – от конструирования моделей технических, технологических и организационных систем до разрешения проблем развития человечества и Вселенной.

2. Исходные предпосылки моделирования

Научной основой моделирования является *теория подобия*. Основным в этой теории есть понятие *аналогии*, т.е. подобие объектов за некоторыми признаками. Подобные объекты называются *аналогами*.

Аналогия между объектами может устанавливаться по качественным, по количественным признакам или по тем и другим.

Основным видом количественной аналогии является *математическое подобие*. Объекты в этом случае описываются аналогичными уравнениями или функциями, отличающимися только значениями коэффициентов или констант.

Другим видом количественной аналогии можно считать *физическое подобие*. В этом случае объекты не могут быть описаны математически, и их подобие определяется соотношением физических параметров, которые характеризуют исследуемый процесс в оригинале и на модели.

Любая модель отражает свойства оригинала лишь частично. И чем больше свойств оригинала отражено в модели, тем ближе модель к оригиналу, тем она точнее. Но одновременно с повышением точности модели растет и её сложность. Исследователю постоянно

приходится искать компромисс между желаемой точностью модели и её сложностью.

На степень соответствия между объектом и моделью указывают два понятия: *изоморфизм* и *гомоморфизм*.

Объект и его модель изоморфны, если существует взаимнооднозначное соответствие между ними, благодаря которому можно преобразовать одно представление на другое. Строго доведённый изоморфизм для объектов разной природы дает возможность переносить знания с одной области в другую.

Однако существуют и менее тесные связи между объектом и моделью. Это так называемые гомоморфные связи. Они устанавливают однозначное соответствие только в одну сторону - от модели к объекту. На рис.1 схематично изображена разница изоморфной зависимостей между объектом и моделью для пространства состояний объекта Z_0 и модели Z_M .

ИЗОМОРФИЗМ и ГОМОМОРФИЗМ (греч. *isos* — одинаковый, *homoios* — подобный и *morphe* — форма) — понятия, характеризующие соответствие между структурами объектов. Две системы, рассматриваемые отвлеченно от природы составляющих их элементов, являются изоморфными друг другу, если каждому элементу первой системы соответствует лишь один элемент второй и каждой связи в одной системе соответствует связь в другой и обратно. Такое взаимнооднозначное соответствие называется ИЗОМОРФИЗМ. Полный ИЗОМОРФИЗМ может быть лишь между абстрактными, идеализированными объектами, напр., соответствие между геометрической фигурой и ее аналитическим выражением в виде формулы. ИЗОМОРФИЗМ связан не со всеми, а лишь с некоторыми фиксированными в познавательном акте свойствами и отношениями сравниваемых объектов, которые в других своих отношениях могут отличаться. ГОМОМОРФИЗМ отличается от ИЗОМОРФИЗМА тем, что соответствие объектов (систем) однозначно лишь в одну сторону. Поэтому ГОМОМОРФНЫЙ образ есть неполное, приближенное отображение структуры оригинала.

Таково, напр., отношение между картиной и местностью, между грамзаписью и ее оригиналом — звуковыми колебаниями воздушной среды. Понятия ИЗОМОРФИЗМ и ГОМОМОРФИЗМ широко применяются в математической логике и кибернетике (Филос. словарь).

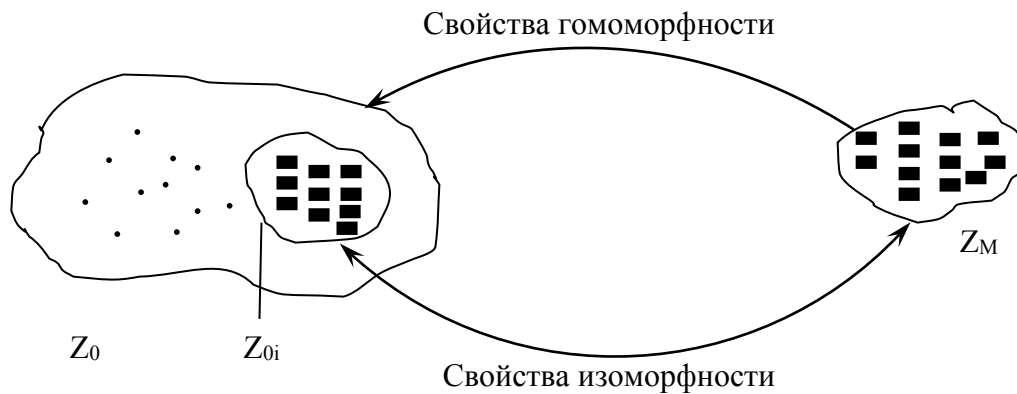


Рис. 1

Современные машины и оборудование относятся к *сложным техническим системам*, которые представляют собой иерархически организованную и целенаправленно функционирующую совокупность большого числа закономерно связанных и взаимодействующих элементов. Под *элементами* подразумевают неделимые конструктивные составляющие.

По определению основателя общей теории систем У.Эшби. **сложные технические системы - это системы, перенасыщенные внутренними связями элементов и внешними связями со средой.**

Проектирование сложных технических систем (т.с.) как единого материального объекта представляет значительные трудности. Поэтому их обычно формализуют путем разделения на отдельные подсистемы различных уровней, связанные между собой. Эта процедура называется **декомпозицией или структуризацией**.

Подсистемами верхнего уровня являются наиболее крупные, условно автономные, составные части машин и оборудования, например, механическая, гидравлическая, пневматическая системы. За подсистемы нижнего уровня часто принимают отдельные элементы технической системы.

В идеальном случае целесообразно выделение подсистем до таких границ, когда удастся применять те или иные физические

законы. Это подтверждает вывод У.Эшби о том, что *«теория систем должна по сути дела представлять собой науку упрощения»*.

Аналитическое проектирование т.с. базируется на моделировании и оптимизации. По результатам математического моделирования производится оценка свойств т.с. и прогнозируется их поведение в различных условиях.

Физическое моделирование позволяет проверить на практике потенциальные возможности т.с. Соответствующие контрольные испытания могут проводиться как на установках, изготовленных на основе критериев подобия, так и на опытных образцах в натуральном измерении.

Использование методов оптимизации дает возможность с помощью специфических математических моделей определять наилучшие характеристики проектируемых т.с.

Применение математического моделирования и оптимизации потребовало серьезной научной разработки ряда формализованных методов и приемов.

Выбор параметров математической модели

Независимая переменная, соответствующая одному из возможных способов воздействия на м.с, называется **фактором**. Функционирование любой т.с. зависит от ряда внешних и внутренних факторов. *Те факторы, которые варьируют в определенном диапазоне, принимают за параметры*. Наиболее важные из них считаются *определяющими*.

Процесс выделения совокупности параметров, которые должны быть учтены при разработке математической модели, называется параметризацией.

Традиционно при аналитическом проектировании м.с. выделяют следующие 5 групп параметров.

1. Входные параметры $x^*(i = 1, 2, n)$. Они могут быть измерены, но возможность воздействия на них отсутствует (например, контролируемые, но нерегулируемые климатические условия).

2. Управляющие параметры $u (i = 1, 2, \dots, m)$. Они являются контролируемыми и регулируемыми (температура, влажность, давление, исходная масса и др.). От них в первую очередь, зависят условия эффективного функционирования т.с.

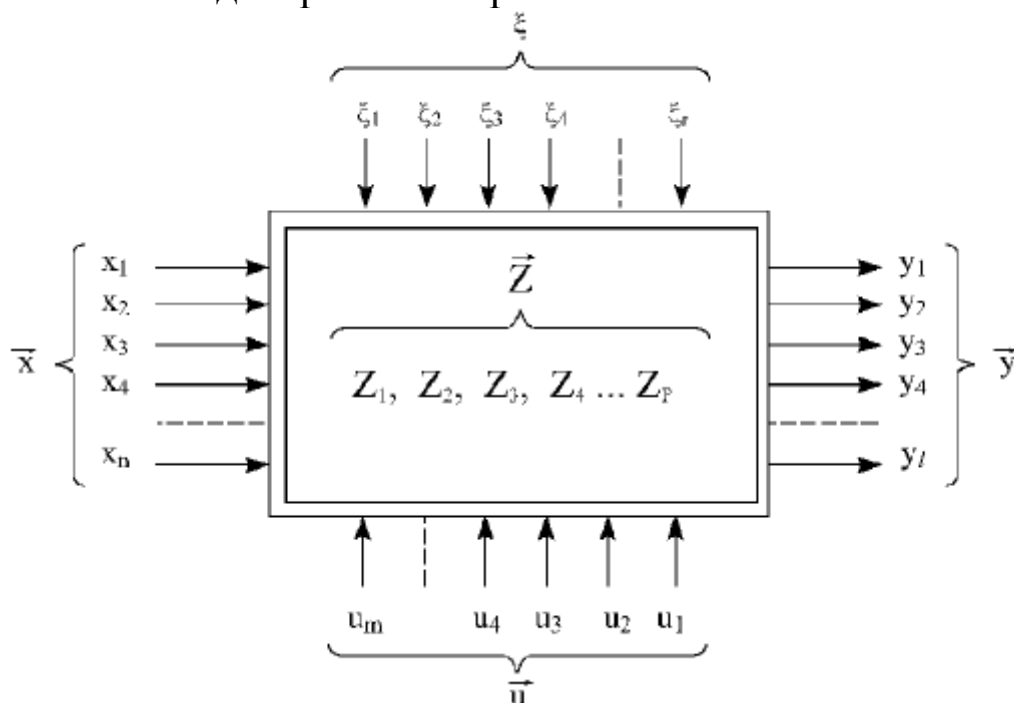
3. Возмущающие параметры ϵ ($i = 1, 2, \dots, r$). Они характеризуют внешний «шум», могут изменяться случайным образом с течением времени и недоступны для измерения (например, неконтролируемый износ кинематических пар, различные помехи).

4. Внутренние параметры s ($i = 1, 2, \dots, p$). Они характеризуют состояние м.с. и определяют логику ее функционирования (например, амплитудно-частотные параметры вибрационного грохота).

5. Выходные параметры y ($i = 1, 2, \dots, l$). Они являются интегральными характеристиками т.с. определяют ее потенциальные возможности и особенности взаимодействия с окружающей средой. Условия, при которых эти параметры достигают экстремальных значений, называют **критериями качества** или **критериями оптимальности**. Зависимость выходного параметра от других параметров представляет собой **уравнение связи**, а соотношение для критерия оптимальности – **целевую функцию**.

Если состояние т.с. характеризуется непрерывными физическими величинами (температура, давление, скорость и т.п.), то она имеет непрерывные переменные параметры.

Если параметры т.с. принимают лишь целочисленные значения, то их относят к дискретным переменным величинам.



Информационная модель системы

В некоторых случаях при построении математических моделей м.с. внутренние параметры s вообще не рассматриваются, а

моделируются только связи между входными и выходными процессами. Получаемые при этом зависимости не раскрывают особенностей процессов, протекающих в самом моделируемом объекте, но их вполне достаточно для оценки функционирования.

При формировании математических моделей т.с. выбранные параметры могут использоваться двояко:

при теоретических исследованиях они входят в состав фундаментальных закономерностей, характеризующих свойства объекта-оригинала;

при экспериментальных исследованиях они фигурируют в процессе идентификации моделируемого объекта

Этап параметризации является одним из важнейших при математическом моделировании и оптимизации. Поэтому для выявления определяющих параметров т.с. используются различные методы:

экспертные оценки,

планируемые наблюдения за функционированием реальных объектов,

специально поставленные эксперименты,

другие информационные источники.

1.3. Общая характеристика математических моделей

Введем понятие математической модели технического объекта. **Математическая модель** представляет собой идеализированную схему технического объекта (или его составных частей), построенную путем отображения в ней наиболее существенных свойств и «элементарных» процессов с помощью комплекса математических зависимостей и логических соотношений.

Такая модель позволяет решать поставленные задачи обобщенно, обеспечивает краткость и четкость фиксации свойств и отношений объекта - оригинала, дает его знаковую интерпретацию.

Формирование математической модели осуществляется на основе выделенного комплекса параметров, а также различных уровней абстрагирования и упрощения реального технического объекта. Тем не менее, такая идеализированная модель позволяет получать довольно точные результаты. Это объясняется тем, что для определения основных характеристик исследуемого объекта при

моделировании достаточно учесть относительно небольшое число определяющих параметров. Сложность, однако, состоит в том, что они должны быть правильно выбраны и между ними должны быть установлены объективные связи; необходимо также решать вопрос о полноте модели. При этом важно не столько знание математики, сколько глубокое понимание сущности поведения объекта-оригинала, где тесно переплетаются и знание теории, и опыт, и интуиция.

К настоящему времени установится общий порядок построения математических моделей, при котором сначала исходят из простых условий, а затем шаг за шагом по мере увеличения глубины анализа и накопления необходимой информации поднимаются по ступеням иерархической градации, переходя к постепенному усложнению моделей [18].

Качество получаемых моделей нельзя оценить ни по структуре, ни по форме. Единственным критерием такой оценки может служить лишь достоверность полученных на модели прогнозов поведения реального объекта.

Любая математическая модель должна рассматриваться в совокупности трех ее аспектов - смыслового, аналитического и вычислительного.

Смысловая сторона модели - содержательное описание - представляет собой начальную стадию математической формализации. При этом производится определение границ исследуемого объекта, обобщаются сведения о физической природе протекающих в нем процессов и соответствующих количественных характеристиках, выделяется степень взаимодействия его элементов, выбираются управляемые переменные и критерии оптимальности, учитываются необходимые ограничения и принятые допущения.

Аналитическая сторона модели - математическое описание - является выражением содержательного описания на языке математики в виде некоторой системы уравнений связи. При этом вся совокупность используемых параметров представляется в абстрактных терминах, все качественные зависимости между ними переводятся в функциональные или иные соотношения, устанавливаются граничные и начальные условия.

Применяемый математический аппарат должен «вписываться» в моделируемый объект и достоверно отражать специфику его структурных и динамических особенностей. По своей форме

математическое описание модели обычно состоит из зависимостей, отражающих общие физические законы: уравнений, описывающих «элементарные» процессы (например, движение материальной точки по поверхности рабочего органа машины); различных эмпирических и полуэмпирических соотношений, полученных в результате статистической обработки экспериментальных данных.

Примерный состав математического описания модели представлен на рис. 1.2.

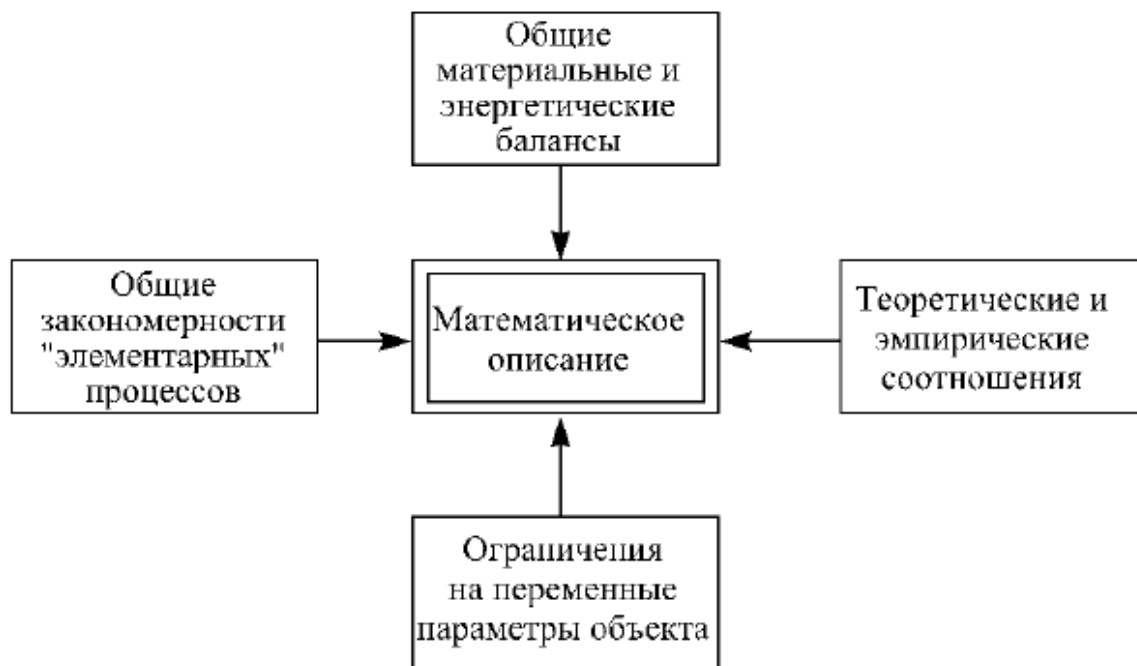


Рис.1.2. Состав математического описания модели

Вычислительный аспект математической модели - алгоритм решения с использованием ЭВМ - определяется как упорядоченная последовательность операций, которые надо выполнить над уравнениями связи, чтобы получить искомые результаты самым эффективным путем.

При разработке алгоритма должны быть установлены размерности всех используемых величин, определены допустимые границы, в которых будут изменяться параметры, и задана предельная ошибка вычислений.

Во многих случаях ряд операций, заложенных в алгоритм, осуществляется с помощью *стандартных процедур* программного обеспечения ЭВМ.

Основными требованиями к математическим моделям являются универсальность, математическая строгость, точность и экономичность.

Универсальность математической модели определяется возможностью ее использования для различных технических объектов. В настоящее время получили широкое распространение так называемые модульные (блочные) модели, которые описываются известными физическими закономерностями. При разработке математической модели какого-либо конкретного технического объекта модульные модели соединяются между собой по принципу подчинения. Образующаяся многоуровневая цепочка моделей обладает рядом преимуществ, т.к. на каждом уровне решаются свои задачи, а на вышестоящие и нижестоящие уровни по уравнениям связи (каналам связи) передается минимальный объем информации.

Использование модульных моделей существенно сокращает затраты труда при моделировании технических объектов, позволяет применять унифицированные формы ввода в ЭВМ данных об их свойствах и систематизированные процедуры обработки полученных результатов.

Математическая строгость связана с качеством идеализации объекта оригинала. *Под идеализацией понимается выделение основных и отбрасывание второстепенных* (в условиях поставленной задачи) свойств и характеристик указанного объекта.

На практике идеализация осуществляется различными путями:

- переходом от распределенных параметров к сосредоточенным;
- сокращением числа независимых переменных;
- снижением размерности решаемой задачи (от трехмерной к двухмерной и одномерной):
 - заменой переменных константами;
 - изменением принимаемых ограничений;
 - усреднением свойств по объему и направлению (идеальное перемешивание; гипотезы плоских сечений и т.п.).

В любом случае математическая строгость соответствует принятой степени идеализации.

Точность математической модели оценивается ее способностью отображать значения искомых параметров моделируемого объекта с ошибкой, не более заданной.

Экономичность математической модели характеризуется затратами вычислительных ресурсов ЭВМ (машинного времени и памяти) на ее реализацию, а также количеством параметров, используемых в модели.

Требования по универсальности, математической строгости и экономичности моделей противоречивы. Необходимо иметь удачное компромиссное решение. По этой причине в каждом случае следует располагать не одной, а несколькими математическими моделями.

Математические модели отличаются одна от другой по назначению, структуре, степени детализации свойств моделируемого объекта, по способу получения. Поэтому их можно классифицировать по целому ряду признаков.

Так, по назначению выделяют модели **функционирования** и **оптимизационные** модели. В первом случае модели предназначены для выявления характерных зависимостей между параметрами технического объекта в процессе его функционирования. Во втором случае модели служат для определения наилучших, оптимальных значений регулируемых параметров.

Разделение моделей по признаку **предсказуемости** на вероятностные и детерминированные является наиболее фундаментальным элементом классификации.

Если в модели внешние воздействия и внутренние возмущения принимаются случайными, т.е. являются непредсказуемыми, то модель называют **вероятностной**. Ее решение формируется в виде распределения вероятностей.

Если же указанные воздействия и возмущения носят закономерный характер - а реакция на них предсказуема, то модель называют **детерминированной**. Ее решение формируется в виде числа.

В зависимости от того, **входит время** в качестве независимой переменной в математическое описание модели или нет, все модели принято *разделять на динамические и статические*.

В **динамических моделях** отображается инерционность исследуемых объектов. При этом значения выходных параметров в определенный момент времени зависят не только от значений остальных параметров, но и от предшествующих воздействий, т.е. от предыстории. Обычно интервал времени, относящийся к

предыстории, является небольшим. Он называется «памятью» объекта и характеризует его запаздывание, т.е. последствие.

Если в модели последствие и текущее время не учитываются, то она называется **статической**.

Весьма существенно деление математических моделей на **линейные и нелинейные**. Модель называется **линейной**, если для нее **выполняется принцип суперпозиции** (наложения). При этом каждый выходной параметр связан линейной зависимостью с другими параметрами.

Модель является **нелинейной**, если реакция на два различных входных возмущения не эквивалентна сумме реакций на каждое из этих возмущений в отдельности, т.е. когда нарушается принцип суперпозиции.

Линейные модели значительно проще нелинейных. Поэтому их широко используют при моделировании. И хотя большинство процессов в технических объектах являются нелинейными, при моделировании их стремятся линеаризовать даже за счет некоторого снижения точности.

По **способу получения** математические модели подразделяются на 4 большие группы:

- аналитические;
- эмпирические;
- стохастические;
- имитационные.

В последующих разделах будет дан подробный анализ таких моделей.

3. Уровни детализации математических моделей

Когда проводятся исследования по разработке имитационной модели реальной системы, невозможно моделировать сразу всю систему в целом. Вначале моделируются подсистемы, которые имеют отношение к исследуемой проблеме. Они включают части системы на различных уровнях детализации. Графически это можно представить в виде пирамиды управления (рис. 1.3). Коллекция выделенных областей формирует те части системы, которые включены в модель.

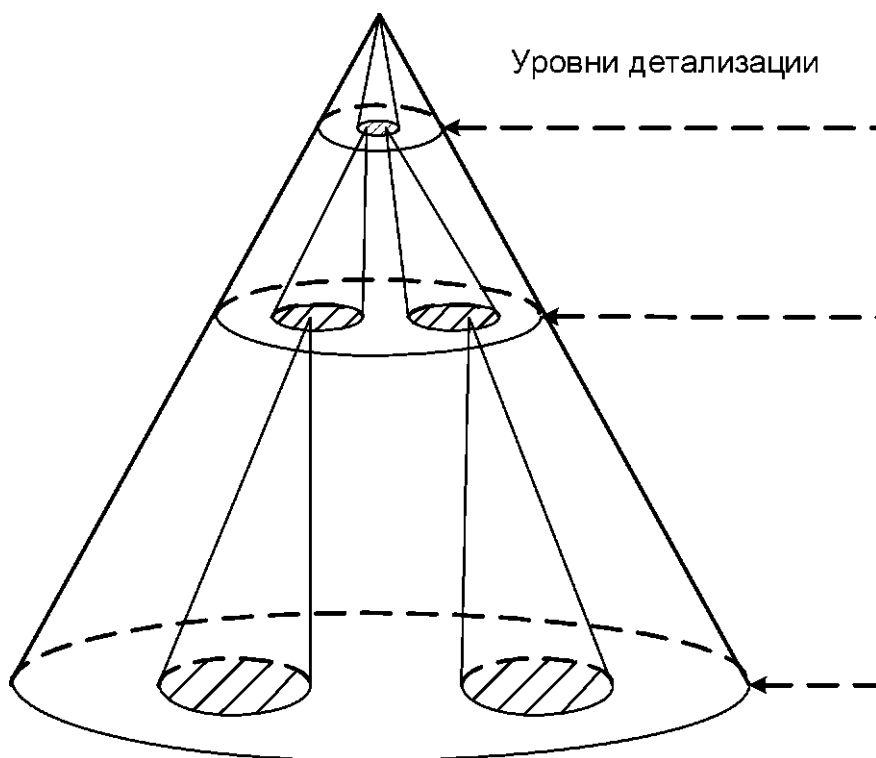


Рис. 1.3. Пирамида управления моделированием

При проектировании технических объектов используют множество видов математических моделей, в зависимости от степени декомпозиции системы, аспекта исследования, стадии и этапа проектирования.

На любом уровне иерархии объект проектирования представляют в виде некоторой системы, состоящей из элементов. В этой связи различают **ММ элементов и систем**.

Всякая система – это сложный объект, который не может быть сведен просто к совокупности всех своих частей благодаря наличию связей, объединяющих различные части в одно целое. Причем отдельные части оказываются связаны друг с другом не только непосредственно, но и через посредников. Следовательно, поведение некоторых частей системы может приобретать новое качество, отсутствовавшее у таких частей по отдельности. В целом система может вести себя таким образом, что предсказать ее поведение путем анализа отдельных свойств ее объектов, не возможно. Такое свойство системы называется **«эмерджентность»** или более известным словосочетанием **«системный эффект»**.

По мере усложнения систем не увидеть системный эффект обходится очень дорого.

При переходе к более высокому уровню система низшего уровня становится элементом нового уровня и, наоборот, при переходе к низшему уровню элемент становится системой. Обычно, чем ниже уровень, тем более детально описание его физических свойств. Следовательно, на низших уровнях используют наиболее сложные математические модели. На высших уровнях с успехом могут быть применены более простые модели.

В зависимости от степени абстрагирования при описании физических свойств технической системы различают три основных технических уровня: верхний или **метауровень**; средний или **макроуровень**; нижний или **микроуровень**.

Метауровень соответствует начальные стадиям проектирования, задачей которого является научно-технический поиск и прогнозирование, разработка концепции и технического решения, разработка технического предложения. Обычно на этом уровне используются методы теории графов, математической логики, теории автоматического управления, теории массового обслуживания, теории конечных аппаратов.

На **макроуровне** объект рассматривается как динамическая система с сосредоточенными параметрами. Математические модели этого уровня обычно представляют собой системы обыкновенных дифференциальных уравнений. Эти модели используются обычно для определения параметров технического объекта и его функциональных элементов.

На **микроуровне** объект представляется как сплошная среда с распределенными параметрами. Для математического описания процессов функционирования обычно используются дифференциальные уравнения в частных производных. На микроуровне проектируются неделимые по функциональному признаку элементы, которые называются **базовыми**.

1.4. Реализация математической модели и ее верификация

Основные методы решения задач моделирования

На этапе программной реализации модели и реализации плана экспериментов необходим выбор методов решения задач моделирования. При этом используются три основные группы методов:

1) *графические* — оценочные приближенные методы, основанные на построении и анализе графиков;

2) *аналитические* — решения, полученные строго в виде аналитических выражений (пригодны для узкого круга задач);

3) *численные* — основной инструмент для решения сложных математических задач, основанный на применении различных численных методов.

Аналитическое решение удается получить редко и чаще всего лишь при упрощенной формулировке задачи моделирования в линейном приближении. Основным средством моделирования является алгоритмический подход, реализующий вычислительный эксперимент на ЭВМ или в наши дни на *персональном компьютере* (ПК).

Ветви 1 на [рис. 12.1](#) соответствует постановка задачи, относящейся к *микроуровню*, как краевой, чаще всего в виде ДУЧП. Численные методы решения ДУЧП основаны на *дискретизации* переменных и *алгебраизации* задачи.

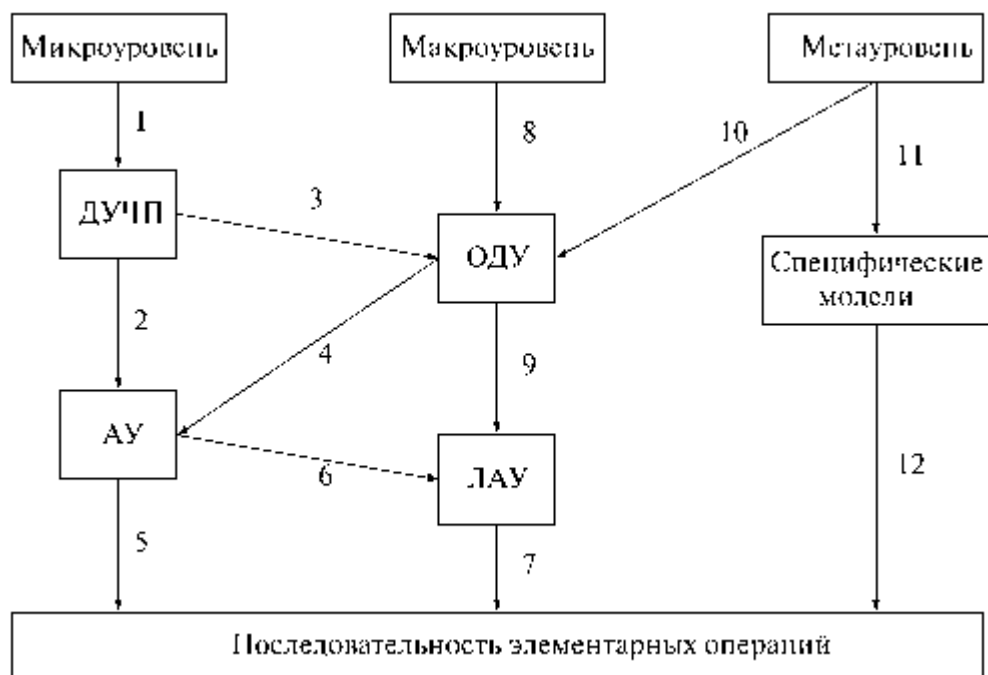


Рис. 12.1. Процесс преобразования ММ. ДУЧП — дифференциальные уравнения с частными производными; ОДУ — обыкновенные дифференциальные уравнения; АУ — алгебраические уравнения; ЛАУ — линейные алгебраические уравнения; 1...12 — взаимно направленные пути дискретизации переменных в ММ

Дискретизация заключается в замене непрерывных переменных конечным множеством их значений в заданных для исследования пространственном и временном интервалах; **алгебраизация** — в замене производных алгебраическими соотношениями.

Получаемое численными методами решение почти всегда содержит некоторую *погрешность*. Абсолютная погрешность

$$\varepsilon = x - x_u$$

есть разность между приближенным x и точным или идеальным x_u значениями результата, а относительная погрешность определяется как

$$\delta = \frac{\varepsilon}{x_u}$$

Наличие погрешности решения обусловлено рядом причин. Перечислим основные источники погрешности.

1. Математическая модель является лишь приближенным описанием реального процесса (погрешность модели).

2. Исходные данные, как правило, содержат погрешности, поскольку являются результатами приближенных экспериментов (измерений) или решениями вспомогательных задач (погрешность данных).

3. Применяемые для решения задачи методы в большинстве случаев являются приближенными (погрешность метода).

4. При вводе исходных данных в ЭВМ, выполнении операций производятся округления (вычислительная погрешность).

Погрешности 1 и 2 — неустранимые на данном этапе решения, для их уменьшения приходится возвращаться вновь к построению математической, а иногда и концептуальной модели, проводить дополнительное экспериментальное уточнение условий задачи.

Оценка обусловленности вычислительной задачи — еще одно обязательное требование при выборе метода решения и построении математической модели.

Пусть вычислительная задача корректна. Теоретически устойчивость задачи означает, что ее решение может быть найдено со сколь угодно малой погрешностью, если только гарантировать достаточно малую погрешность входных данных. Однако на практике их точность ограничена (и величиной гораздо большей, чем $\varepsilon_m = 2^{-P+1}$ - машинная точность, P - порядок, округление производится усечением).

Как влияют малые, но конечные погрешности входных данных на решение? Насколько сильно они искажают результат? Ответ на это дает понятие *обусловленности* задачи, т. е. чувствительности решения вычислительной задачи к малым погрешностям входных данных.

Задачу называют *хорошо обусловленной*, если малым погрешностям входных данных отвечают малые погрешности решения, и *плохо обусловленной*, если возможны сильные изменения решения. Часто возможно ввести количественную оценку степени обусловленности - *число обусловленности*. Его можно интерпретировать как коэффициент возможного возрастания погрешности в решении по отношению к вызвавшей их погрешности входных данных. Если установлено неравенство между этими погрешностями, то можно пользоваться следующими выражениями:

$$\Delta y^* \leq v_{\Delta} \Delta(x^*) \text{ и } \delta y^* \leq v_{\delta} \delta(x^*)$$

где v_{Δ} — абсолютное число обусловленности v_{δ} — относительное число обусловленности. Для плохо обусловленных задач $v_{\delta} \gg 1$, неустойчивость соответствует $v_{\delta} = \infty$.

При каких значениях v_{δ} можно считать задачу плохо обусловленной? Это зависит от требований к точности решения и от уровня обеспечиваемой точности исходных данных.

Если требуется найти решение с точностью 0.1 %, а входная информация задается с точностью в 0.02%, то при $v_{\delta} = 10$ уже будет плохая обусловленность.

Однако если исходные данные задаются с $\delta(x^*) < 0.0001$ %, то при $v_{\delta} = 10^3$ - задача хорошо обусловлена ($\delta(y^*) = 0.1$ %).

Вычислительные методы, как правило, преобразуются к виду, удобному для программной реализации. Можно выделить следующие классы численных методов:

- *Метод эквивалентных преобразований* - исходную задачу заменяют другой, имеющей то же решение: нахождение корня нелинейного уравнения $f(x) = 0$ сводят к поиску точек глобального минимума $\Phi(x) = (f(x))^2$.
- *Методы аппроксимации* - заменяют исходную задачу другой, решение которой близко к решению исходной задачи.
- *Методы конечно-разностные*; основанные на замене производных конечными разностями, например $f'(x) \approx \frac{f(x+h) - f(x)}{h}$.
- *Прямые (точные) методы* - решение может быть получено за конечное число элементарных операций (арифметические и извлечение корня). Многие прямые методы не годятся к применению в ЭВМ из-за чувствительности к ошибкам округления.
- *Итерационные методы* - методы последовательных приближений к решению задачи. задается начальное приближение решения, строится итерационная последовательность приближений к решению. Если эта последовательность сходится к решению, то говорят что итерационный процесс сходится. Множество начальных приближений, для которых метод сходится, называются областью сходимости метода.

- *Методы статистических испытаний (Монте-Карло)* - основаны на моделировании случайных величин и построении статистических оценок решений задач (для моделирования больших систем). Для реализации этих методов используются генераторы случайных чисел.

Численные методы группируются вокруг типичных математических задач: задач анализа, алгебры, оптимизации, решения дифференциальных и интегральных уравнений, обратных задач (синтез). Этот этап решения заканчивается выбором и обоснованием конкретных численных методов решения, разработкой алгоритма, которые могут быть программно реализованы средствами компьютерной техники.

Контроль правильности модели

Для контроля правильности полученной модели может использоваться ряд приемов.

- *Анализ размерности* - величины в левой и правой части выражения, отдельные слагаемые в каждой из частей должны иметь одинаковую размерность.

- *Проверка порядков и характеров зависимостей* - параметры и переменные, которые в данной задаче выражены величинами большего порядка малости, могут быть исключены из рассмотрения как несущественные, что часто позволяет значительно упростить модель и ее анализ. Характер изменения значений моделируемых величин должен соответствовать их реальному смыслу, не противоречить наблюдаемым данным.

- *Исследование предельных случаев* - результаты моделирования при крайних значениях параметров модели, равных, как правило, нулю или бесконечности, не должны противоречить смыслу (например, энергия реальной физической системы не может оказаться бесконечно большой, время протекания процесса - отрицательным и т. п.). Модель в этом случае существенно упрощается и легче для понимания.

- *Проверка замкнутости и корректности математической задачи* - система математических соотношений должна иметь единственное решение.

Задача называется *корректной*, если она удовлетворяет трем требованиям:

- 1) ее решения существует при любых допустимых входных данных;
- 2) это решение единственно (однозначно определено);
- 3) решение непрерывно зависит от данных задачи - устойчиво по отношению к малым возмущениям входных данных.

Решение вычислительной задачи называется *устойчивым* по входным данным x , если оно зависит от входных данных непрерывным образом; т. е. для любого $\varepsilon > 0$ существует $\delta = \delta(\varepsilon) > 0$ такое, что всяким исходным данным x , удовлетворяющим условию $\Delta(x^*) < \delta$, отвечает приближенное решение y^* для которого $\Delta(y^*) < \varepsilon$.

Далеко не все встречающиеся на практике задачи являются корректными. К ним, например, нельзя отнести обратные задачи геофизики, астрофизики, спектрографии, распознавания образов, синтез и многие другие важные прикладные проблемы. Свойство корректности задачи имеет большое значение для выбора метода решения. К некорректным задачам неприменимы обычные численные методы вычислительной математики. Строгий анализ корректности во многих случаях математически сложен, и ограничиваются проверкой соответствия количества неизвестных и связывающих их уравнений в модели.

Заключительная трехстадийная оценка качества математической модели и интерпретация полученных при ее решении численных результатов называется верификацией.

На первой стадии устанавливается достоверность, т.е. адекватность модели. Это очень ответственная, трудоемкая и неформальная процедура. Основной путь оценки адекватности - проведение тестовых исследований, при которых сравниваются результаты расчетов по модели с результатами опытов на реальном объекте.

При анализе адекватности модели возможна ее корректировка путем внесения некоторых уточнений и дополнений.

На второй стадии верификации полученное решение математической модели подвергается качественной оценке, т.е. объясняется с позиций технологии и механики происходящих явлений.

На третьей стадии решение анализируется в зависимости от изменения диапазонов варьирования параметров.

После получения адекватной математической модели т.с. следует оценить возможности ее использования в более широком плане, а именно:

для изыскания таких параметров т.с. которые обеспечивали бы оптимальное преобразование входных воздействий в выходные координаты:

для построения алгоритма эффективного управления т.с; для расширения объема исследований за счет реализации большого числа вариантов и увеличения номенклатуры используемых показателей:

для прогнозирования поведения т.с. при различных условиях функционирования:

для изучения других технических объектов со сходным математическим описанием.

2. Разработка математической модели

Что касается математического моделирования, то следует отметить, что при его использовании очень многое зависит от способа подачи, как модели, так и результатов моделирования. Рассмотрим простой пример. Пусть на некотором предприятии для водоснабжения используются резервуар ёмкостью W тысяч литров. Уровень потребления воды предприятием – V_{Π} тысяч литров в сутки. Скорость наполнения резервуара – $V_{\text{Н}}$ тысяч литров в сутки. Нужно найти время T , за которое будет заполнен резервуар.

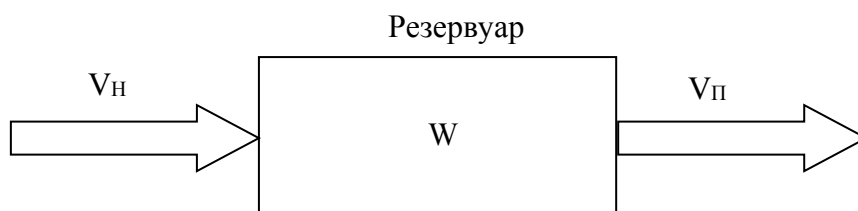


Рис. 2

Время заполнения резервуара находится легко:

$$T = \frac{W}{V_H - V_{II}} \quad (1)$$

Эта математическая модель наполнения резервуара является чрезвычайно идеализированной. Все её параметры полагаются неизменными во времени, скорости наполнения и потребления постоянными, влияние внешних факторов на систему не учитываются. Такая идеализированная модель легко решается аналитически. Однако с помощью такой модели можно получить ответ только на один конкретный вопрос – за какое время будет заполнен резервуар? Если задачу усложнить, приблизить её к практике, то строя модель, нужно учитывать, что потребности предприятия в водообеспечении постоянно меняются, более того, возможны перебои в работе насосов во время подачи воды. Решение задачи в частично замкнутом виде можно записать как:

$$W = (V_H - V_{II})T \quad (2)$$

Время заполнения резервуара объемом W зависит от параметров модели V_H и V_{II} при разных соотношениях между параметрами V_H и V_{II} и при разных начальных условиях можно получить разные результаты. Можно даже построить график наполнения резервуара (рис.3).

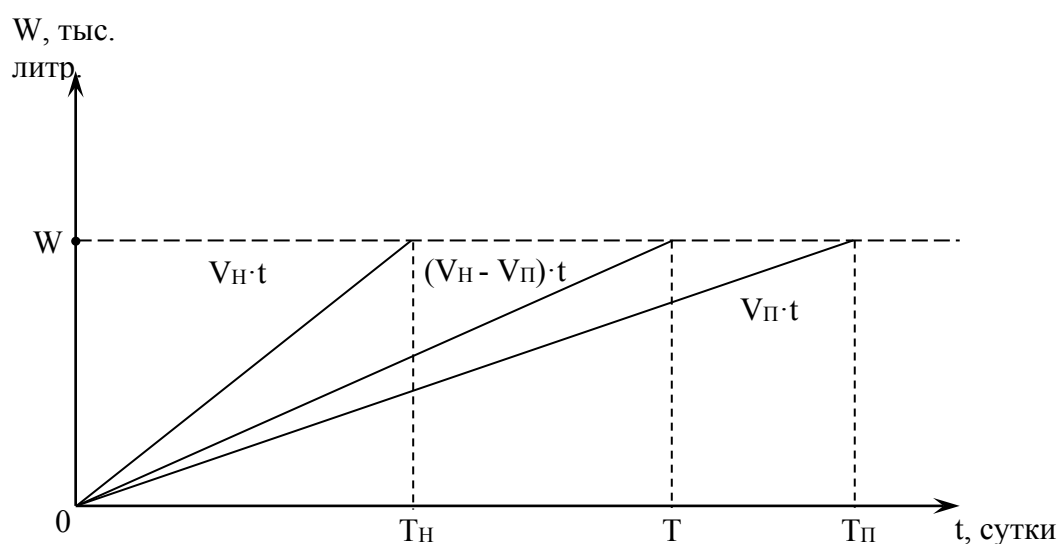


Рис. 3

Реализовать эту модель можно с помощью численных методов. Меняя в формуле (2) значения t от 0 с некоторым интервалом Δt (шаг

моделирования) до того значения, когда будет выполняться равенство (2), получим динамическую характеристику заполнения резервуара. Чем меньше шаг Δt , тем точнее результат будет получен. Правда, тем дольше будет решаться задача моделирования.

Имитационное моделирование – это метод конструирования модели и проведения экспериментов. Вся информация про имитационную модель в общем имеет логико-математический характер и представляется в виде совокупности алгоритмов, которые описывают процесс функционирования объекта. В большинстве имитационной моделью есть её программная реализация на компьютере, а имитационное моделирование сводится к проведению экспериментов с моделью путём многократного прогона программы с некоторым множеством данных.

Если вернуться к процессу наполнения резервуара, то при помощи имитационной модели весь процесс можно воспроизвести с использованием равенства (2). Обозначим через W_i текущее состояние резервуара, которое воспроизводится в определенные моменты модельного времени, которое изменяется с постоянным шагом Δt :

$$W_i = (V_H - V_{II})t_i, \quad (3)$$

где $t_i = t_{i-1} + \Delta t$ ($i = 1, 2, 3, \dots$) $t_0 = 0$

W , тыс. литров

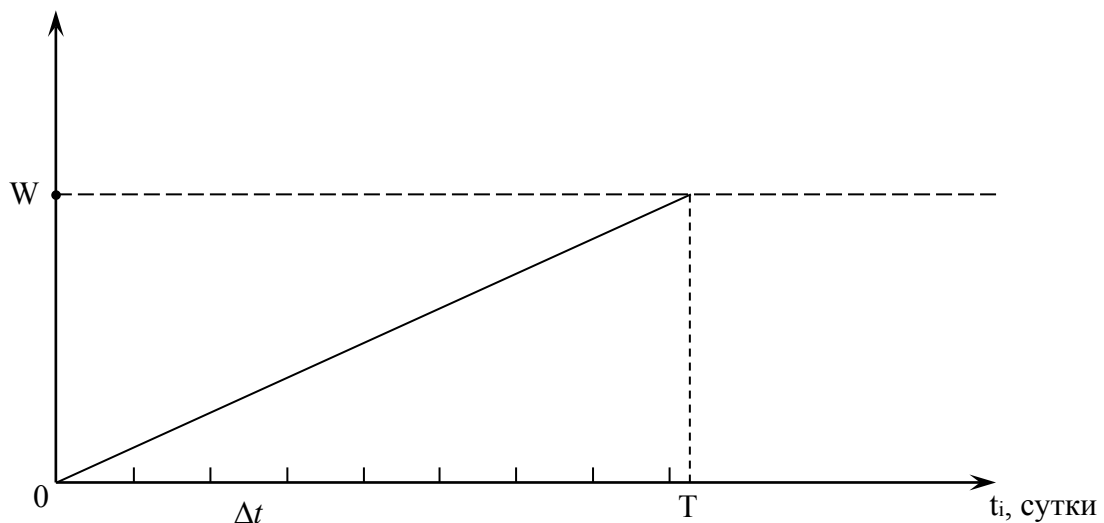


Рис. 4

Такая модель является детерминированной. Процесс моделирования заканчивается, если на некотором шаге выполняется условие $W_i \geq W$. Другими словами решение получается за один прогон имитационной модели. Точность результатов будет зависеть от значения Δt .

Если в модели нужно учитывать влияние случайных факторов, то возникает необходимость статистической оценки результатов моделирования.

Статистическое моделирование – есть самостоятельным видом моделирования. Оно включается в имитационное моделирование только при необходимости моделирования вероятности систем и процессов.

Построим более реальную модель системы водоснабжения предприятия. Допустим, что уровень потребления воды на предприятии имеет вероятностный характер и изменяется в соответствии с равномерным распределением вероятностей в пределах $V_{\Pi} \pm V_{\Pi}$. Тогда значения V_{Π} в некоторый момент времени t_i будет определять как:

$$2\Delta V_{\Pi} z_i + V_{\Pi} - \Delta V_{\Pi}$$

где z_i - случайное число, равномерно распределённое в интервале $[0.1]$.

Результаты работы имитационной модели даны на рис.5:

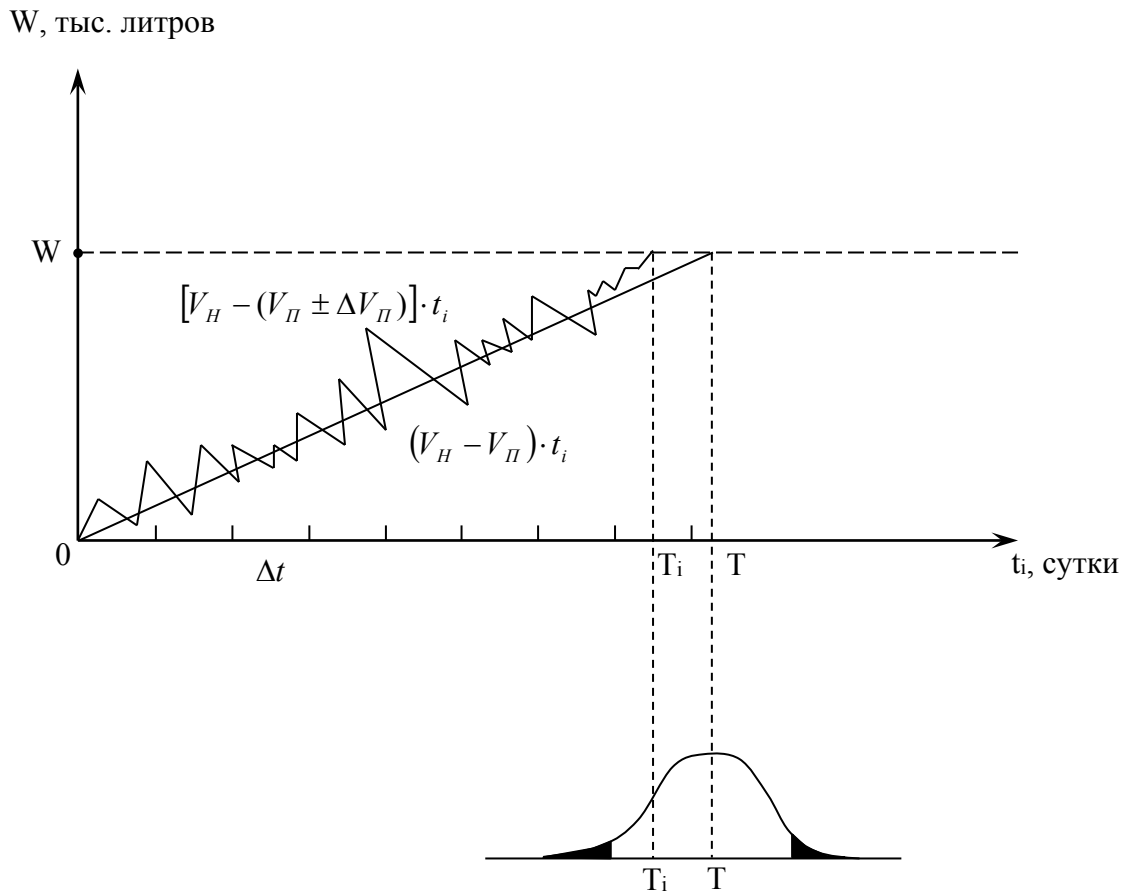


Рис. 5

После каждого прогона модели получается случайное значение T_j (j -количество прогонов). Для каждого прогона задается своя последовательность случайных чисел z_i .

Как видно на рис.5, полученные значения T_j будут отличаться от среднего значения T , найденного при помощи детерминированной модели.

Чтобы оценить время наполнения резервуара, нужно задаться точностью оценивания $E = T$ и уровень доверия. Обычно = 0,95, т.е. есть гарантия того, что в 95 случаях из 100 среднее значение времени T будет лежать в пределах $T + \Delta T$.

Математические схемы описания технических систем

Каждая конкретная система S характеризуется набором свойств, под которыми понимаются величины, отражающие поведение моделируемого объекта и учитывающие условия ее функционирования во взаимодействии с внешней средой E . При

построении математической модели системы необходимо решить вопрос о ее полноте. Полнота модели регулируется выбором границы «система S -внешняя среда E ». При этом необходимо выделить основные свойства и и отбросить второстепенные. Причем разделение свойств на основные и второстепенные существенно зависит от цели моделирования системы.

Типовые схемы

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

Непрерывно–детерминированные модели (D – схемы)

Математические схемы такого вида отражают *динамику* процессов, протекающих во времени в системе. Поэтому они называются *D–схемы (dynamic)*.

Любая модель физического процесса должна удовлетворять закону сохранения ... движения, импульса, массы, энергии и т.д. Можно так сказать – законы сохранения – это опоры, основа любого модельного описания.

К примеру, рассмотрим процесс, математическая модель которого представляет дифференциальное уравнение вида:

для линейного движения

$$m\ddot{x} + \beta\dot{x} + cx = F(t)$$

для кругового (вращательного) движения

$$J\ddot{\gamma} + \beta_k\dot{\gamma} + c_k\gamma = M(t).$$

Здесь:

m, J - инерционный параметр системы – масса, момент инерции;

β, β_k - величины, характеризующие изменение силы (момента), обусловленные изменением скорости (частоты);

c, c_k - величины, характеризующие жесткость системы;

$F(t), M(t)$ - внешняя нагрузка – сила, момент, как функции времени.

Отметим, что такими моделями описывается довольно широкий круг практических инженерных и исследовательских задач, да и задач исследовательского характера.

Что отражают эти зависимости? Первое слагаемое в приведенных уравнениях – это сила (момент), обусловленная ускорением элемента или машины в целом, согласно второму закону Ньютона. Тогда несложно уяснить, что поскольку суммироваться могут только физически одинаковые величины, и все остальные слагаемые в левой части уравнений есть силы или моменты для второго уравнения, обусловленные, соответственно, изменением скорости (частоты) и деформацией системы, правая часть уравнений – сила или момент внешнего воздействия на систему.

Приведем эти уравнения к так называемому *стандартному* виду. В дальнейшем, не нарушая общности задачи и с целью уменьшения объема изложения, примем для анализа второе дифференциальное уравнение, т.е. математическую модель, описывающую протекающий процесс или, другими словами, формирования момента в элементе рассматриваемой системы. Поскольку $J \neq 0$, разделим все члены уравнения и его правую часть на эту величину, и обозначив

$$\beta_k / J \text{ через } 2n, \quad c_k / J \text{ через } \omega^2,$$

$$M(t) / J \text{ через } m(t),$$

приведенное выше уравнение примет вид

$$\ddot{\gamma} + 2n\dot{\gamma} + \omega^2\gamma = m(t).$$

Представленная в таком виде математическая модель рассматриваемого процесса есть известное вам еще из курса высшей математики линейное дифференциальное второго порядка с правой частью. Поэтому, говорят, физическая модель и соответствующая ей ***система есть линейная второго порядка динамическая система с возмущением.***

Рассмотрим эту модель более подробно, хотя она, как увидим далее, и является частным случаем, но широко применяемой в инженерных расчетах и исследованиях.

Модель является линейной, т.к. коэффициенты при первой производной $\dot{\gamma}$ и деформацией γ - величины постоянные. Геометрически – это угловые коэффициенты, характеризующие угол наклона линии к оси абсцисс. Раз они постоянные, то линии – прямые.

Динамические системы, допускающие описание протекающих в них процессов, называют ***линейными второго порядка.***

Для установления физического смысла и соответствующего наименования составляющих математической модели запишем еще раз эту модель в ее первоначальном виде, а именно:

$$J\ddot{\gamma} + \beta_k \dot{\gamma} + c_k \gamma = M(t).$$

В этой системе, как не трудно видеть, первый член представляет собой момент, обусловленный ускорением, второй член – момент, обусловленный скоростью, и третий – момент, обусловленный деформацией динамической системы или ее отдельного элемента. Правая часть математической модели – это внешнее возмущение, точнее, момент внешнего возмущения.

Таким образом, все три параметра, которыми характеризуется движение тела, имеют место в уравнении. Это ускорение, скорость (здесь частота или, говорят, угловая скорость) и координата (угол закручивания или деформация). Можно было бы вывести это

уравнение и другим путем, скажем, методом Лагранжа – сравнением кинетической и потенциальной энергии системы.

Иногда указанные составляющие математической модели называют первую – инерционной, вторую – скоростной, третью – восстанавливающей составляющей или восстанавливающим моментом. При этом, следует отметить, *при постоянном коэффициенте при втором слагаемом в левой части уравнения*, что свидетельствует о линейной связи между моментом и скоростью (частотой), *динамические системы называют системами с вязким сопротивлением*.

В зависимости от того, каковы коэффициенты по характеру и каково по характеру внешнее возмущение математические модели и соответствующие им динамические системы могут быть:

1. Линейными второго (или более высокого порядка в зависимости от порядка старшей производной) порядка, если все коэффициенты – величины постоянные, которые, напомним еще раз, являются по своей сути угловыми коэффициентами, а соответствующие им динамические системы – линейными динамическими системами второго (или более высокого) порядка.

2. При переменном моменте инерции (массе, что хорошо может быть иллюстрировано на летящей ракете, взлетная масса которой несколько сот тонн, в конце полета – несколько сот килограмм). Математическая модель в этом случае представляет собой дифференциальное уравнение второго (или более высокого) порядка с переменными коэффициентами, а соответствующая этой модели динамическая система – система второго или более высокого порядка с переменными параметрами.

3. Если коэффициент $\beta_k = \beta(\dot{\gamma})$, т.е. является зависящим, например, от скорости, математическая модель будет нелинейной, а соответствующая ей динамическая система также не линейна. Такие системы хорошо изучены голландским механиком и математиком Ван дер Полем и носят в его честь название «вандерполевских» или автоколебательных систем, с которыми мы ниже познакомимся.

4. Если восстанавливающий момент – величина не линейная, т.е. $c_k = c(\gamma)$ или другого параметра, такие модели тоже являются не линейными и соответствующие им динамические системы также являются нелинейными и называются нелинейными динамическими системами с нелинейным восстанавливающим моментом или не линейной восстанавливающей силой. Такие математические модели, т.е. дифференциальные уравнения обычно называют по имени тех математиков, которые нашли их решение, например, уравнение Матье, Дуффинга и т.д. Динамические системы, которые допускают описание этими уравнениями, могут также называться именами этих же математиков.

5. Если правая часть математической модели есть детерминированная функция, то такие динамические системы называют детерминированными динамическими системами (линейными или не линейными в зависимости от математической модели).

6. Если правая часть математической модели – величина случайная или является случайной функцией, а в случае, когда случайная функция есть временная, т.е. случайная во времени, случайным процессом, динамическая система называется системой (линейной или нелинейной) со случайным возмущением или, иногда, со случайной правой частью.

7. Если коэффициенты математической модели или хотя бы один из них является величиной случайной, а правая часть – случайным процессом, такие математические модели носят название *стохастических* дифференциальных уравнений, а соответствующие им динамические системы – стохастическими, т.е. случайными во времени.

8. Кроме указанных математических моделей, могут быть модели с запаздывающим аргументом. В качестве аргумента, скажем, в правой части уравнения выступает не время t , а $t - \tau$ - аргумент со смещением или с «запаздыванием».

Приведенные математические модели и соответствующие им динамические системы относятся к классу динамических систем с **сосредоточенными массами**, т.е. к таким системам, в которых массы (моменты инерции) сосредоточены в отдельных элементах их конструкции (шестернях, роторах двигателей и т.д.). Моменты инерции (массы) связей (валов) по сравнению с ними незначительны и ими можно или пренебречь или по известным законам их можно привести к сосредоточенному моменту инерции, например, к моменту инерции ротора двигателя.

Вместе с тем существуют и такие динамические системы, у которых моменты инерции (массы) связей (тех же валов, пружин, цепей и т.д.) соизмеримы с моментами инерции (массами) других элементов конструкции. Такие динамические системы носят название динамических систем с **распределенными параметрами или массами**. Примерами таких динамических систем могут служить редукторы привода с так называемыми тяжелыми валами, например, паровых турбин генераторов тепловых электростанций. Эти валы представляют собой конструкцию примерно 10-12 м длиной и 500-600 мм диаметром. Другой пример, цепи корабельного типа, которые используются в качестве тягового органа очистных комбайнов, стругов и т.д. Погонная масса таких цепей достигает 25-30 кг. При длине цепи, например, 200 м масса такой цепи достигает 6000 кг.

Такие динамические системы описываются дифференциальными уравнениями в частных производных, т.е. математические модели таких динамических систем представляют либо дифференциальные уравнения в частных производных, либо системы дифференциальных уравнений в частных производных. С такими математическими моделями имеют дело при описании тепловых процессов, колебаний на тяжелых нитях и др. процессов. В зависимости от того, какие процессы описываются этими дифференциальными уравнениями, последние и носят название дифференциальных уравнений гиперболического, теплового или параболического типа. Ниже рассмотрим и некоторые из этих

уравнений и соответствующие им процессы. С уравнениями в частных производных мы уже встречались при описании процессов, протекающих в жидкостях. Примером тому может служить известный вам оператор Лапласа, представляющий собой сумму вторых частных производных параметра по координатам выбранной системы координат.

Дискретно–детерминированные модели (F – схемы)

Основным видом дискретно–детерминированных моделей является *конечный автомат*.

Конечным автоматом называют дискретный преобразователь информации, способный под воздействием входных сигналов переходить из одного состояния в другое и формировать сигналы на выходе. Это автомат *с памятью*. Для организации памяти в описание автомата вводят автоматное время и понятие *состояние автомата*.

Понятие «*состояние*» автомата означает, что выходной сигнал автомата зависит не только от входных сигналов в данный момент времени, но и учитывает входные сигналы, поступающие ранее. Это позволяет устранить время как явную переменную и выразить выходные сигналы как функцию состояний и входных сигналов.

Всякий переход автомата из одного состояния в другое возможен не ранее, чем через дискретный интервал времени. Причем сам переход считается, происходит мгновенно, то есть не учитывают переходные процессы в реальных схемах.

Существует два способа введения автоматного времени по которому автоматы делятся на *синхронные* и *асинхронные*.

В *синхронных* автоматах моменты времени, в которых фиксируются изменения состояний автомата, задаются специальным устройством – генератором синхросигналов. Причем сигналы поступают через равные интервалы времени – Δt . Частота тактового

генератора выбирается такой, чтобы любой элемент автомата успел закончить свою работу до появления очередного импульса.

В *асинхронном* автомате моменты перехода автомата из одного состояния в другое заранее не определены и зависят от конкретных событий. В таких автоматах интервал дискретности является переменным.

Также существуют *детерминированные* и *вероятностные* автоматы.

В *детерминированных* автоматах поведение и структура автомата в каждый момент времени однозначно определены текущей входной информацией и состоянием автомата.

В *вероятностных* автоматах они зависят от случайного выбора.

Абстрактно конечный автомат можно представить как математическую схему (F – схему), которая характеризуется шестью видами переменных и функций:

- 1) конечное множество $x(t)$ входных сигналов (входной алфавит);
- 2) конечное множество $y(t)$ выходных сигналов (выходной алфавит);
- 3) конечное множество $z(t)$ внутренних состояний (алфавит состояний);
- 4) начальное состояние автомата z_0 , $z_0 \in Z$;
- 5) функция переходов $\varphi(z, x)$ автомата из одного состояния в другое;
- 6) функция выходов $\Psi(z, x)$ автомата.

Абстрактный конечный автомат имеет один вход и один выход. В каждый дискретный момент времени $t=0, 1, 2, \dots$ F– автомат находится в определенном состоянии $z(t)$ из множества Z – состояний автомата, причем в начальный момент времени $t=0$ он всегда находится в начальном состоянии $z(0)=z_0$. В момент t , будучи в состоянии $z(t)$, автомат способен воспринять на входном канале сигнал $x(t) \in X$ и выдать на выходном канале сигнал $y(t) = \Psi[z(t), x(t)]$, переходя в состояние

$$z(t+1) = \varphi[z(t), x(t)] \text{ , где } z(t) \in Z, x(t) \in X \text{ .}$$

Абстрактный конечный автомат реализует некоторое отображение множества слов входного алфавита X на множество слов выходного алфавита Y , то есть, если на вход конечного автомата, установленного в начальное состояние z_0 , подавать в некоторой последовательности буквы входного алфавита $x(0), x(1), x(2), \dots$, которые составляют входное слово, то на выходе автомата будут последовательно появляться буквы выходного алфавита $y(0), y(1), y(2), \dots$ образуя выходное слово.

Следовательно, работа конечного автомата происходит по следующей схеме: на каждом t -ом такте на вход автомата, находящегося в состоянии $z(t)$, подается некоторый сигнал $x(t)$, на который автомат реагирует переходом на $(t+1)$ -ом такте в новое состояние $z(t+1)$ и выдачей некоторого выходного сигнала.

В зависимости от способа определения выходного сигнала синхронные абстрактные конечные автоматы подразделяются на два типа:

– F – автомат первого рода, также называется **автомат Мили**:

$$\begin{cases} z(t+1) = \varphi[z(t), x(t)], t = 0, 1, 2, \dots; \\ y(t) = \Psi[z(t), x(t)], t = 0, 1, 2, \dots; \end{cases}$$

– F – автомат второго рода:

$$\begin{cases} z(t+1) = \varphi[z(t), x(t)], t = 0, 1, 2, \dots; \\ y(t) = \Psi[z(t), x(t-1)], t = 1, 2, 3, \dots; \end{cases}$$

Автомат второго рода, для которого

$$y(t) = \Psi[z(t)], t = 0, 1, 2, \dots$$

называется **автомат Мура** – функция выходов не зависит от входной переменной $x(t)$.

Чтобы задать конечный F – автомат, необходимо описать все элементы множества $F = \langle z, x, y, \varphi, \Psi, z_0 \rangle$.

Существует несколько способов задания работы F – автоматов среди которых наибольшее применение нашли табличный, графический и матричный.

Дискретно – непрерывные модели

Процессы в линейных импульсных и цифровых системах автоматического управления описываются дискретно – разностными уравнениями вида:

$$b_k \Delta^k y(n) + b_{k-1} \Delta^{k-1} y(n) + \dots + b_0 y(n) = x(n), \quad (1)$$

где $x(n)$ – решетчатая функция входного сигнала; $y(n)$ – решетчатая функция выходного сигнала, которая определяется решением уравнения (1.2); b_k – постоянные коэффициенты; $\Delta^k y(n)$ – разность k – го порядка; $t=nT$, где nT – n –ый момент времени, T – период дискретности (в выражении (1) он условно принят за единицу).

Уравнение (1) можно представить в другом виде:

$$a_k y(n-k) + a_{k-1} y(n-k+1) + \dots + a_0 y(n) = x(n) \quad (2)$$

Уравнение (2) представляет собой рекуррентное соотношение, которое позволяет вычислить любой $(i+1)$ –й член последовательности по значениям предыдущих её членов $i, i-1, \dots$ и значению $x(i+1)$.

Основным математическим аппаратом моделирования цифровых автоматических систем является Z – преобразование, которое базируется на дискретном преобразовании Лапласа. Для этого необходимо найти импульсную передаточную функцию системы, задаться входной переменной и, варьируя параметрами системы, можно найти лучший вариант проектируемой системы.

Дискретно – стохастические модели (Р - схемы)

К дискретно – стохастической модели относится *вероятностный автомат*. В общем, виде вероятностный автомат является дискретным потактным преобразователем информации с памятью, функционирование которого в каждом такте зависит только от состояния памяти в нем и может быть описано статистически. Поведение автомата зависит от случайного выбора.

Применение схем вероятностных автоматов имеет важное значение для проектирования дискретных систем, в которых проявляется статистически закономерное случайное поведение.

Для Р – автомата вводится аналогичное математическое понятие, как и для F – автомата. Рассмотрим множество G, элементами которого являются всевозможные пары (x_i, z_s) , где x_i и z_s элементы входного подмножества X и подмножества состояний Z соответственно. Если существуют две такие функции φ и Ψ , что с их помощью осуществляется отображение $G \rightarrow Z$ и $G \rightarrow Y$, то говорят, что $F = \langle Z, X, Y, \varphi, \Psi \rangle$ определяет автомат детерминированного типа.

Функция переходов вероятностного автомата определяет не одно конкретное состояние, а распределение вероятностей на множестве состояний (автомат со случайными переходами). Функция выходов также есть распределение вероятностей на множестве выходных сигналов (автомат со случайными выходами).

Для описания вероятностного автомата введем в рассмотрение более общую математическую схему. Пусть Φ – множество всевозможных пар вида (z_k, y_j) , где y_j – элемент выходного подмножества Y. Далее потребуем чтобы любой элемент множества G индуцировал на множестве Φ некоторый закон распределения следующего вида:

элементы из $\Phi \dots (z_1, y_1)(z_1, y_2) \dots (z_k, y_{j-1})(z_k, y_j)$

$(x_i, z_s) \dots b_{11} \quad b_{12} \quad \dots \quad b_{k(j-1)} \quad b_{kj}$

где b_{kj} – вероятности перехода автомата в состояние z_k и появления на выходе сигнала y_j , если он был в состоянии z_s и на его вход в этот момент времени поступал сигнал x_i .

Число таких распределений, представленных в виде таблиц равно числу элементов множества G. Если обозначить это множество таблиц через B, то тогда четверку элементов $P = \langle Z, X, Y, B \rangle$ называют

вероятностным автоматом (Р – автоматом). При этом $\sum_{k=1}^k \sum_{j=1}^j b_{kj} = 1$.

Частным случаем Р– автомата, задаваемого как $P = \langle Z, X, Y, B \rangle$ являются автоматы, у которых либо переход в новое состояние, либо выходной сигнал определяются детерминировано (Z – детерминированный вероятностный автомат, Y – детерминированный вероятностный автомат соответственно).

Очевидно, что с точки зрения математического аппарата задание Y – детерминированного Р – автомата эквивалентно заданию некоторой марковской цепи с конечным множеством состояний. В связи с этим аппарат марковских цепей является основным при использовании Р– схем для аналитических расчетов. Подобные Р– автоматы используют генераторы марковских последовательностей при построении процессов функционирования систем или воздействий внешней среды.

Марковские последовательности, согласно теореме Маркова, – это последовательность случайных величин, для которой справедливо выражение

$$\lim_{N \rightarrow \infty} \left(\frac{1}{N^2} \right) D \left[\sum_{i=1}^N x_i \right] = 0,$$

где N – количество независимых испытаний; D – дисперсия.

Такие Р– автоматы (Р– схемы) могут быть использованы для оценки различных характеристик исследуемых систем как для аналитических моделей, так и для имитационных моделей с использованием методов статистического моделирования.

Y – детерминированный Р– автомат можно задать двумя таблицами: переходов (табл.1) и выходов (табл.2).

Таблица 1

Z_k	Z_k				
	Z_1	Z_2	...	Z_{k-1}	Z_k
Z_1	P_{11}	P_{12}	...	$P_{1(k-1)}$	P_{1k}
Z_2	P_{21}	P_{22}	...	$P_{2(k-1)}$	P_{2k}
...
Z_k	P_{k1}	P_{k2}	...	$P_{k(k-1)}$	P_{kk}

Таблица 2

Z...	z ₁	z ₂	...	z _{k-1}	z _k
Y...	y _{i1}	y _{i2}	...	y _{i(k-1)}	y _{ik}

Где P_{ij} – вероятность перехода Р– автомата из состояния z_i в состояние z_j , при этом $\sum_{j=1}^k P_{ij} = 1$.

Таблицу 1 можно представить в виде квадратной матрицы размерности $k \times k$. Такую таблицу будем называть *матрицей переходных вероятностей* или просто *матрицей переходов Р-автомата*, которую можно представить в компактной форме:

$$P_p = \begin{pmatrix} P_{11} & P_{12} & \dots & P_{1k} \\ P_{21} & P_{22} & \dots & P_{2k} \\ \cdot & \cdot & \cdot & \cdot \\ P_{k1} & P_{k2} & \dots & P_{kk} \end{pmatrix}$$

Для описания Y– детерминированного Р–автомата необходимо задать начальное распределение вероятностей вида:

$$\begin{array}{cccccc} Z\dots & z_1 & z_2 & \dots & z_{k-1} & z_k \\ D\dots & d_1 & d_2 & \dots & d_{k-1} & d_k \end{array}$$

где d_k – вероятность того, что в начале работы Р– автомат находится в состоянии z_k , при этом $\sum_{k=1}^K d_k = 1$.

И так, до начала работы Р– автомат находится в состоянии z_0 и в начальный (нулевой) такт времени меняет состояние в соответствии с распределением D. После этого смена состояний автомата определяется матрицей переходов P. С учетом z_0 размерность матрицы P_p следует увеличить до $(k+1) \times (k+1)$, при этом первая строка матрицы будет $(d_0, d_1, d_2, \dots, d_k)$, а первый столбец будет нулевым.

Пример. Y– детерминированный Р– автомат задан таблицей переходов:

Таблица 3

$$P_p = \begin{vmatrix} 0 & 0,50 & 0 & 0 & 0,50 \\ 0 & 0 & 0 & 1,00 & 0 \\ 0 & 0 & 0,75 & 0 & 0,25 \\ 0 & 0 & 0,40 & 0 & 0,60 \\ 0 & 1,00 & 0 & 0 & 0 \end{vmatrix}$$

и таблицей выходов

Таблица 4

Z	z ₀	z ₁	z ₂	z ₃	z ₄
Y	0	0	1	1	0

С учетом таблицы 3 граф переходов вероятностного автомата представлен на рис.14.

Требуется оценить суммарные финальные вероятности пребывания этого автомата в состоянии z₂ и z₃, т.е. когда на выходе автомата появляются единицы.

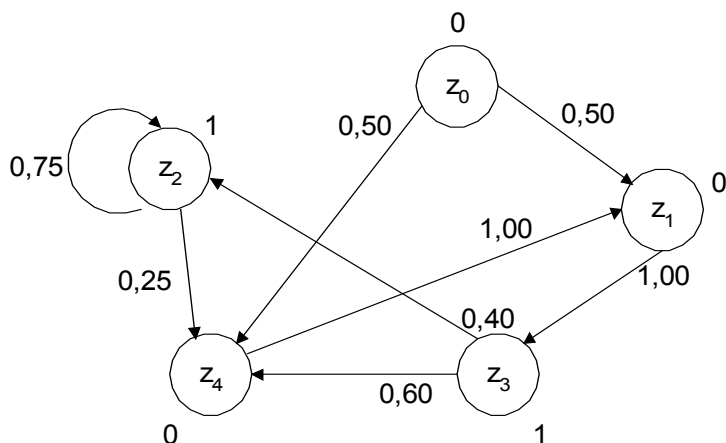


Рис. 14. Граф переходов

При аналитическом подходе можно использовать известные соотношения из теории марковских цепей и получить систему уравнений для определения финальных вероятностей. Причем начальное состояние можно не учитывать в виду того, что начальное распределение не оказывает влияние на значения финальных вероятностей. Тогда таблица 1.3 примет вид:

$$\vec{C} = \vec{C} \begin{vmatrix} 0 & 0 & 1,00 & 0 \\ 0 & 0,75 & 0 & 0,25 \\ 0 & 0,40 & 0 & 0,60 \\ 1,00 & 0 & 0 & 0 \end{vmatrix}$$

где $\vec{C} = (c_k) = (c_1, c_2, c_3, c_4)$ – финальная вероятность пребывания Y– детерминированного P– автомата в состоянии z_k .

В результате получаем систему уравнений:

$$\begin{cases} c_1 = c_4 \\ c_2 = 0,75c_2 + 0,40c_3 \\ c_3 = c_1 \\ c_4 = 0,25c_2 + 0,60c_3 \end{cases} \quad (3)$$

К данной системе следует добавить условие нормировки:

$$c_1 + c_2 + c_3 + c_4 = 1 \quad (4)$$

Теперь решая систему уравнений (3) совместно с (4), получаем:

$$c_1 = \frac{5}{23}; c_2 = \frac{8}{23}; c_3 = \frac{5}{23}; c_4 = \frac{5}{23}.$$

Таким образом, при бесконечной работе заданного автомата на его выходе будет формироваться двоичная последовательность с вероятностью появления единицы, равной: $c_2 + c_3 = \frac{13}{23} = 0,5652$.

Кроме аналитических моделей в виде P– схем можно применять и имитационные модели, реализуемые, например, методом статистического моделирования.

Непрерывно–стохастические модели (Q– схемы)

Такие модели рассмотрим на примере использования в качестве типовых математических схем систем массового обслуживания, которые называют **Q– схемами**. Такие Q– схемы применяются при формализации процессов функционирования систем, которые по своей сути являются процессами *обслуживания*.

К *процессам обслуживания* можно отнести: потоки поставок продукции некоторому предприятию, потоки деталей и комплектующих изделий на сборочном конвейере цеха, заявки на

обработку информации ЭВМ от удаленных терминалов сети ЭВМ. Характерным признаком для функционирования таких систем или сетей является случайное появление заявок на обслуживание. Причем в любом элементарном акте обслуживания можно выделить две основные составляющие: ожидание обслуживания и, собственно, сам процесс обслуживания заявки. Представим это в виде некоторого i -го прибора обслуживания Π_i (рис.15), состоящего из накопителя заявок H_i , в котором может находиться одновременно λ_i заявок; K_i – канал обслуживания заявок.

На каждый элемент прибора Π_i поступают потоки событий, в накопитель H_i поток заявок λ_i , на канал K_i – поток обслуживания I_i .

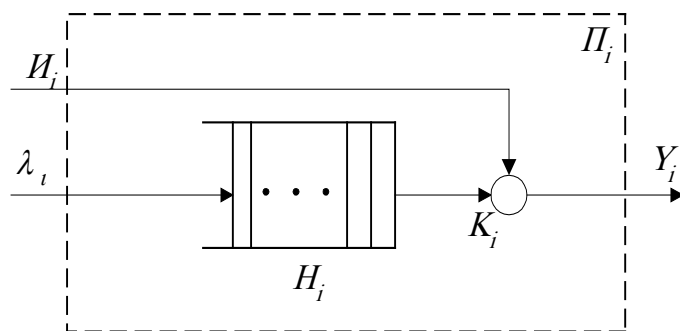


Рис.15. Прибор обслуживания

Потоки событий могут быть *однородными*, если он характеризуется только последовательностью поступления этих событий ($0 \leq t_1 \leq t_2 \leq \dots \leq t_n$), или *неоднородными*, если он характеризуется набором признаков события, например таким набором признаков: источник заявок, наличие приоритета, возможность обслуживания тем или иным типом канала и т.п.

Обычно при моделировании различных систем применительно к каналу K_i можно считать, что поток заявок λ_i на входе K_i образует подмножество неуправляемых переменных, а поток обслуживания I_i – образует подмножество управляемых переменных.

Те заявки, которые по различным причинам не обслуживаются каналом K_i , образуют выходной поток Y_i .

Стохастические минимаксные модели

Эти модели можно отнести к оптимальным стохастическим моделям.

Во многих случаях при построении модели не все условия заранее известны. Эффективность нахождения модели здесь будет зависеть от трех факторов:

- заданных условий x_1, x_2, \dots, x_n ;
- неизвестных условий y_1, y_2, \dots, y_k ;
- зависящих от нас факторов u_1, u_2, \dots, u_m , которые необходимо найти.

Показатель эффективности решения такой задачи имеет вид:

$$W = W(x_1, x_2, \dots, x_n; y_1, y_2, \dots, y_k; u_1, u_2, \dots, u_m)$$

Наличие неизвестных факторов y_i переводит задачу оптимизации в задачу о выборе решения в условиях неопределенности. Задача становится чрезвычайно сложной.

Особенно задача осложняется для случаев, когда величины y_i не обладают статистической устойчивостью, то есть неизвестные факторы y_i нельзя изучить с помощью статистических методов. Их законы распределения либо не могут быть получены, либо вовсе не существуют.

В этих случаях рассматриваются комбинации всевозможных значений $Y: \alpha = (y_1^*, y_2^*, \dots, y_k^*)$ таким образом, чтобы получить как «наилучшее», так и «наихудшее» сочетания значений переменных y_i .

Тогда в качестве критерия оптимизации рассматривается:

$$W = W(x, \alpha, u)$$

$$x = x(x_1, x_2, \dots, x_n), \alpha = \alpha(\alpha_1, \alpha_2, \dots, \alpha_k)$$

$$u = u(u_1, u_2, \dots, u_m)$$

Критерий Вальда

При использовании этого критерия всегда ориентируются на то, что случайные факторы сложатся в наихудшую комбинацию. Такой критерий называют *пессимистическим*:

$$\max_u \min_\alpha W(x, \alpha, u).$$

Пример. Имеется четыре стратегии игрока: u_1, u_2, u_3, u_4 ; три варианта условий: y_1, y_2, y_3 и известные значения $x_{ij} \ i=1,2,3,4, j=1,2,3$.

Требуется найти оптимальное решение с использованием критерия Вальда.

Составим матрицу игры наихудших результатов (табл.5). Согласно критерия Вальда $\alpha_i = \alpha_3 = 0,25$.

Таблица 5

$u_i \setminus y_i$	y_1	y_2	y_3	α_i
u_1	0,20	0,30	0,15	0,15
u_2	0,75	0,20	0,35	0,20
u_3	0,25	0,80	0,25	0,25
u_4	0,85	0,05	0,45	0,05

Следовательно, оптимальной будет стратегия u_3 .

Критерий Гурвица

При этом критерии предполагается, что с вероятностью β сложатся наилучшая, а с вероятностью $(1-\beta)$ наихудшая комбинации случайных факторов:

$$\max_u [\beta \max_\alpha W(x, \alpha, u) + (1 - \beta) \min_\alpha W(x, \alpha, u)] \quad (1.6)$$

$$0 < \beta < 1$$

При $\beta=0$ критерий Гурвица переходит в критерий Вальда, при $\beta=1$ получается так называемый критерий «здорового оптимизма». Коэффициент β выбирается из субъективных соображений. Если ситуация очень опасная, то его выбирают ближе к 1.

Пример. Берется тот же пример с табл.1. Принимается $\beta=0,4$. Дистраивается табл.1.5 столбцом, соответствующим оптимальной оценке (табл.1.6). Максимальное значение $\eta_3=0,47$ соответствует стратегии u_3 .

Таблица 6.

$u_i \setminus y_i$	y_1	y_2	y_3	α_i	ω_i	η_i
u_1	0,20	0,30	0,15	0,15	0,30	0,21
u_2	0,75	0,20	0,35	0,20	0,75	0,42
u_3	0,25	0,80	0,25	0,25	0,80	0,47
u_4	0,85	0,05	0,45	0,05	0,85	0,37

По критерию (1.6) получаем: $\eta_i = 0,4\omega_i + 0,6\alpha_i$.

4. Имитационное моделирование технических систем

Имитационное моделирование есть процесс конструирования модели реальной системы и постановки экспериментов на основе этой модели с целью улучшения функционирования системы.

Под *имитацией* с использованием ЭВМ мы будем понимать численный метод проведения на вычислительных машинах экспериментов с математическими моделями, описывающих поведение сложной системы в течение продолжительного периода времени.

Принципиальное отличие имитационного эксперимента от натурального эксперимента состоит в том, что в процессе имитации эксперимент проводится с моделью реальной системы, а не с самой системой.

Достоинства и недостатки имитационного моделирования

Все имитационные модели представляют собой модели типа черного ящика. Поэтому для получения нужной информации или результатов необходимо задаться входными величинами и осуществить "прогон" имитационных моделей на ЭВМ, а не "решать" их. Имитационные модели не способны формировать свое собственное решение в том виде, в каком это имеет место в аналитических моделях, а могут лишь служить в качестве средства для анализа поведения системы в условиях, которые определяются экспериментом. Имитационное моделирование скорее относится не к теории, а к методологии решения проблем. И поэтому имитационное

моделирование следует считать одним из методов решения сложных задач. К имитационному моделированию прибегают обычно в тех случаях, когда:

- нельзя решить задачу аналитическим способом с помощью физического эксперимента;

- натурный эксперимент очень дорог;

- натурный эксперимент может нарушить ход производственного процесса или порядок работы фирмы;

- составной частью системы являются люди (люди будут чувствовать, что за ними наблюдают, вследствие чего могут изменить свое поведение);

- оказывается сложным поддержание одних и тех же рабочих условий при каждом повторном эксперименте или в течении всего времени проведения серии экспериментов;

- при экспериментировании с реальными системами может оказаться невозможным исследование многих альтернативных вариантов.

Благодаря своей простоте идея имитационного моделирования интуитивно привлекательна и для исследователей и для руководителей. И хотя ученые и специалисты с высокой математической подготовкой имитационный подход представляют грубым силовым приемом или последним средством в то же время этот метод является самым распространенным инструментом в руках ученого и специалиста занимающихся проблемами управления и исследования операций. Так среди методов наиболее часто используемых во внутрифирменном планировании в США уже в 1970 г. всех чаще использовался метод имитационного моделирования - 29%. Затем линейное программирование - 21%; сетевые методы - 14% и на последнем месте - теория массового обслуживания - 3%.

Однако следует отметить недостатки имитационного моделирования.

1. Разработка эффективной имитационной модели часто обходится дорого и требует много времени. Так даже для малой

фирмы создание имитационной модели эффективного функционирования может потребоваться более трех лет. Кроме того, требуются высококвалифицированные специалисты.

2. Полученная имитационная модель может оказаться не адекватной, т.е. она может не отражать реальный ход процесса или событий.

3. Имитационная модель, если говорить о её корректности, то она в принципе не точна и нет способов измерить степень её точности. Хотя частично степень точности может быть получена в результате анализа чувствительности модели к изменению определенных параметров.

Приведенные соображения показывают, что не смотря на широкое применение имитационного моделирования для решения сложных задач, этот метод не является панацеей для решения всех проблем управления. Имитационное моделирование может принести пользу, если исследователь наделен изобретательностью, интуицией и находчивостью.

Процесс имитации

Можно выделить следующие этапы процесса имитации.

1. Определение системы - установление границ ограничений и эффективности системы, подлежащей изучению.

2. Формирование модели - переход от реальной системы к некоторой логической системе (абстрагирование).

3. Подготовка данных - отбор данных необходимых для построения модели и представление их в соответствующей форме.

4. Трансляция модели - описание модели на языке, приемлемом для использования на ЭВМ.

5. Оценка адекватности - повышенная до приемлемого уровня степень уверенности, с которой можно судить относительно корректировки выводов о реальной системе.

6. Стратегическое планирование - планирование эксперимента.

7. Tактическое планирование - определение способа проведения каждой серии испытаний, представленных планом эксперимента.

8. Экспериментирование - процесс осуществления имитации с целью получения желаемых данных и анализа чувствительности.

9. Интерпретация - построение выводов по данным, полученным путем имитации.

10. Реализация - практическое использование модели.

11. Документирование - регистрация хода осуществления проекта и его результатов.

Моделирующий алгоритм

Для моделирования процесса на ЭВМ необходимо преобразовать математическую модель в специальный моделирующий алгоритм.

Существуют различные способы представления моделирующего алгоритма.

Исторически первым из таких способов является запись алгоритмов при помощи операторных схем. Затем появились языки программирования, пакеты прикладных программ и специальные языки моделирования.

Среди важнейших операторов моделирующих алгоритмов выделим следующие.

1. Вычислительные операторы. Это в основном арифметические операторы. В операторных схемах моделирующих алгоритмов вычислительный оператор может описывать любую сколь угодно сложную и громоздкую группу операций.

2. Операторы формирования реализаций случайных чисел, величин, событий и процессов.

Для имитации действия различных случайных факторов, сопровождающих исследуемый процесс, при моделировании возникает необходимость формировать реализации случайных событий, случайных величин и случайных функций.

Для реализации всех этих перечисленных случайных групп необходимо иметь случайные числа. Операторы формирования реализаций случайных процессов решают задачу преобразования случайных чисел стандартного вида в реализации случайных процессов с заданными свойствами.

3.Операторы формирования неслучайных величин. Они нужны при моделировании детерминированных математических моделей для реализации различных констант и неслучайных функций времени.

Принципы построения моделирующих алгоритмов

В большинстве имитационных моделей имитируется поведение системы на некотором отрезке времени. Поэтому при создании модели и выборе языка программирования важной задачей является определение механизма системного времени. Это необходимо для корректирования временной координаты состояния системы и для обеспечения согласованности различных блоков и событий в системе.

Модель функционирует в искусственном времени и необходимо обеспечить появление событий в определенном порядке с надлежащими временными интервалами между ними.

Принцип фиксированного шага. Принцип переменного шага

Существует два основных метода задания времени: с помощью фиксированных и переменных интервалов времени. Их иногда еще называют метод фиксированного шага Δt и шага до следующего события. По методу фиксированного временного шага отсчет системного времени ведется через заранее определенные временные интервалы. При использовании метода переменного шага состояние моделируемой системы обновляется с появлением следующего события.

В непрерывных моделях используются механизмы фиксированных приращений временных интервалов. В большинстве моделей с дискретным изменением событий используется метод отсчета времени до следующего события.

Принцип особых состояний

Бывают случаи, когда соотношения математических моделей систем удастся преобразовать таким образом, что можно

определить последующее особое состояние системы по предыдущему особому состоянию или нескольким предыдущим состояниям. Для таких случаев моделирующий алгоритм может быть построен по так называемому принципу "особых состояний". Он отличается от принципа Δt только тем, что включает в себя процедуру определения момента времени, соответствующего следующему особому состоянию по известным характеристикам данного или предыдущего состояния.

Языки моделирования

Для имитационного моделирования используются как универсальные, так и специальные языки. Универсальные языки дают большие возможности программисту в смысле гибкости разработки, отладки и использования модели. Однако это требует больших усилий, затрачиваемых на программирование. В этом случае усложняется порядок выполнения операций, отсчет системного времени и контроль.

Специализированный язык отличается от универсального своими специфическими свойствами.

К этим свойствам или требованиям можно отнести:

- * способность генерировать случайные числа;
- * возможность генерировать случайные величины и процессы;
- * возможность "продвигать" время либо на одну единицу Δt , либо до следующего события;
- * способность накапливать выходные данные;
- * способность проводить статистический анализ накапливаемых данных;
- * способность распределять выходные данные по заранее заданным форматам;
- * способность выполнять идентификацию конкретных событий.

Некоторые из языков имитационного моделирования являются и описательными языками. Они близки к естественному языку, поэтому имитационные модели, написанные на таком языке, легче воспринимаются руководителями и специалистами, не имеющими непосредственного отношения к программированию.

Специальные языки цифрового имитационного моделирования делятся на две группы, соответствующие двум видам имитации: дискретных и непрерывных процессов.

Для моделирования непрерывных процессов используются языки:

– Динамо - для аппроксимации непрерывных процессов используют дифференциальные уравнения первого порядка;

– CSMP, Midas и др., которые помимо блочного построения, применяемого в языке Динамо, еще обладают мощностью и удобством алгебраической и логической алгоритмизацией.

Языки для моделирования дискретных процессов можно разбить на четыре категории:

- 1) языки, ориентированные на действия (CSL, Форсим IV и др.);
- 2) языки, ориентированные на события (Симскрипт, Симком и др.);
- 3) языки, ориентированные на процессы (Симула, SOL);
- 4) языки, ориентированные на потоки сообщений (GPSS, BOSS).

В первой группе действия представлены в модельном времени как мгновенные. В этих языках нет регламентации действиям. Вместо этого исполнительные программы просматривают набор всех условий, от которых зависит появление какого-либо события. И только тогда, когда выполняются все контролируемые условия происходит изменение состояния и сдвиг времени в этой части программы.

Некоторые задачи удобнее программировать на языке, ориентированном на события. При этом событие регламентировано. Регламентация обеспечивает наступление события именно в тот

момент времени, когда динамическое состояние показывает, что сложились условия для его появления.

Языки, ориентированные на процессы, объединяют достоинства первых двух языков, т.е. краткость языков, ориентированных на действия и эффективность языков, ориентированных на события. Написанная на этом языке программа работает как несколько независимых программ: одна посредством просмотра действий, другая посредством регламентирования событий.

Что касается четвертой группы языков, то они на самом деле являются языками процессов и отличаются только схемным построением.

Формы представления моделирующих алгоритмов

Удобной формой представления логической структуры моделей процессов функционирования систем и машинных программ является схема. На различных этапах моделирования составляются обобщенные и детальные логические схемы моделирующих алгоритмов, а также схемы программ.

Обобщенная (укрупненная) схема моделирующего алгоритма задает общий порядок действий при моделировании системы без каких-либо уточняющих деталей. Обобщенная схема показывает, что необходимо выполнить на очередном шаге моделирования, например, обратиться к датчику случайных чисел.

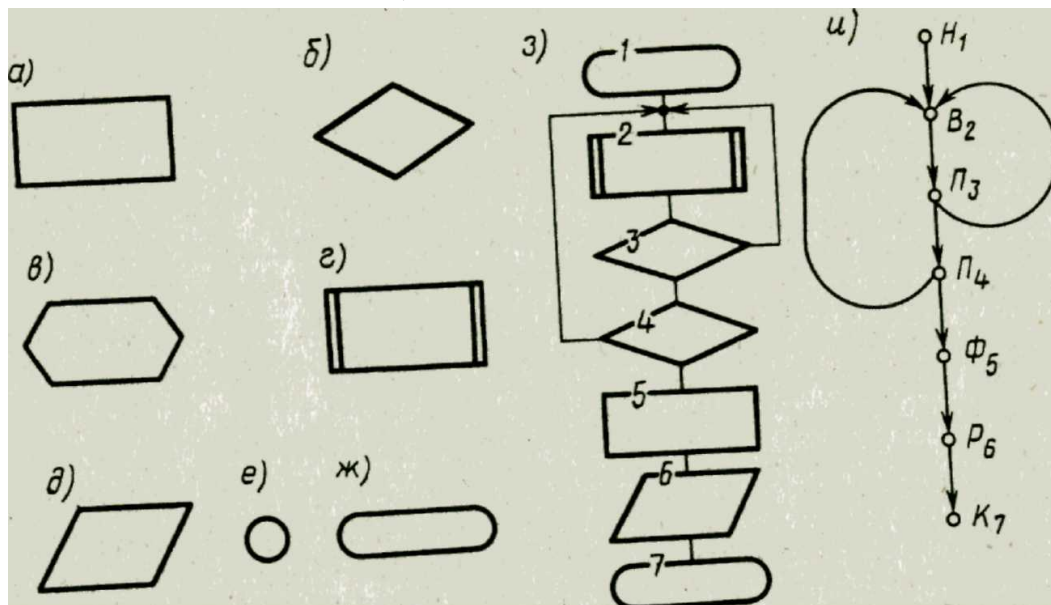
Детальная схема моделирующего алгоритма содержит уточнения, отсутствующие в обобщенной схеме. Детальная схема показывает не только, что следует выполнить на очередном шаге моделирования системы, но и как это выполнить.

Логическая схема моделирующего алгоритма представляет собой логическую структуру модели процесса функционирования системы. Логическая схема указывает упорядоченную во времени последовательность логических операций, связанных с решением задачи моделирования.

Схема программы отображает порядок программной реализации моделирующего алгоритма с использованием конкретного математического обеспечения. Схема программы представляет собой интерпретацию логической схемы моделирующего алгоритма разработчиком программы на базе конкретного алгоритмического языка. Различие между этими схемами заключается в том, что логическая схема отражает логическую структуру модели процесса функционирования системы, а схема программы — логику машинной реализации модели с использованием конкретных программно-технических средств моделирования.

Логическая схема алгоритма и схема программы могут быть выполнены как в укрупненной, так и в детальной форме. Для начертания этих схем используется набор символов, определяемых ГОСТ 19.701 — 90 (ИСО 5807 — 85) «Единая система программной документации. **Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения**». Некоторые наиболее употребительные в практике моделирования на ЭВМ символы показаны на рис. 8.3, где изображены основные, специфические и специальные символы процесса. К ним относятся: основной символ: *a* — процесс — символ отображает функцию обработки данных любого вида (выполнение определенной операции или группы операций, приводящее к изменению значения, формы или размещения информации или к определению, по которому из нескольких направлений потока следует двигаться); специфические символы процесса: *b* — решение — символ отображает решение или функцию переключательного типа, имеющую один вход и ряд альтернативных выходов, один и только один из которых может быть активизирован после вычисления условий, определенных внутри этого символа (соответствующие результаты вычисления могут быть записаны по соседству с линиями, отображающими эти пути); *v* — подготовка — символ отображает модификацию команды или группы команд с целью воздействия на некоторую последующую функцию (установка переключателя, модификация индексного

регистра или инициализация программы); z — predetermined процесс — символ отображает predetermined процесс, состоящий из одной или нескольких операций или шагов программы, которые определены в другом месте (в подпрограмме, модуле); d — ручная операция — „символ отображает любой процесс, выполняемый человеком; специальные символы: e — соединитель — символ отображает выход в часть схемы и вход из другой части этой схемы и используется для обрыва линии и продолжения ее в другом месте (соответствующие символы-соединители должны содержать одно и то же уникальное обозначение); $ж$ — терминатор — символ отображает выход во внешнюю среду и вход из внешней среды (начало или конец схемы алгоритма, внешнее использование или пункт назначения данных).



3.3. Символы и схемы моделирующих алгоритмов. Пример изображения схемы моделирующего алгоритма показан на рис. 8.3, з.

Рис. Обычно схема является наиболее удобной формой представления структуры моделирующих алгоритмов. В ряде случаев используются и другие формы представления моделирующих алгоритмов, например форма *граф-схем* (рис. 3.3, и). Здесь H — начало, K_2 — конец, B_2 — вычисление, Φ — формирование, Π — проверка условия, C — счетчик, P — выдача результата, $l=1, \dots, g$, где g — общее число операторов моделирующего алгоритма. В качестве пояснения к граф-схеме алгоритма в тексте дается раскрытие

содержания операторов, что позволяет упростить представление алгоритма, но усложняет работу с ним.

Моделирующие алгоритмы могут быть также представлены в виде операторных схем. Обозначения операторов на такой схеме соответствуют обозначениям для граф-схем. Для рассмотренного примера операторная схема алгоритма имеет вид

$I^3 \ ^4B, ПЗП4Ф5P6K7$.

Подэтапы второго этапа моделирования. Рассмотрим подэтапы, выполненные при алгоритмизации модели системы и ее машинной реализации, обращая основное внимание на задачи каждого подэтапа и методы их решения.

2.1. Построение логической схемы модели. Рекомендуется строить модель по блочному принципу, т. е. в виде некоторой совокупности стандартных блоков. Построение модели систем \mathcal{L} из таких блоков обеспечивает необходимую гибкость в процессе ее эксплуатации, особенно на стадии машинной отладки. При построении блочной модели проводится разбиение процесса функционирования системы на отдельные достаточно автономные подпроцессы. Таким образом, модель функционально подразделяется на подмодели, каждая из которых в свою очередь может быть разбита на еще более мелкие элементы. Блоки такой модели бывают двух типов: основные и вспомогательные. Каждый основной блок соответствует некоторому реальному подпроцессу, имеющему место в моделируемой системе B , а вспомогательные блоки представляют собой лишь составную часть машинной модели, они не отражают функции моделируемой системы и необходимы лишь для машинной реализации, фиксации и обработки результатов моделирования.

2.2. Получение математических соотношений. Одновременно с выполнением подэтапа построения логической схемы модели необходимо получить, если это возможно, математические соотношения B в виде явных функций, т. е. построить аналитические модели. Этот подэтап

соответствует неявному заданию возможных математических соотношений на этапе построения концептуальной модели. При выполнении первого этапа еще не может иметься информации о конкретном виде таких математических соотношений, а на втором этапе уже необходимо получить эти соотношения. Схема машинной модели M_m должна представлять собой полное отражение заложенной в модели концепции и иметь:

- а) описание всех блоков модели с их наименованиями;
- б) единую систему обозначений и нумерацию блоков;
- в) отражение логики модели процесса функционирования системы;
- г) задание математических соотношений в явном виде.

Таким образом, в общем случае построенная машинная модель M_u системы будет иметь комбинированный характер, т. е. отражать аналитико-имитационный подход, когда часть процесса в системе описана аналитически, а другая часть имитируется соответствующими алгоритмами.

2.3. Проверка достоверности модели системы. Эта проверка является первой из проверок, выполняемых на этапе реализации модели. Так как модель представляет собой приближенное описание процесса функционирования реальной системы S , то до тех пор, пока не доказана достоверность модели M_m , нельзя утверждать, что с ее помощью будут получены результаты, совпадающие с теми, которые могли бы быть получены при проведении натурального эксперимента с реальной системой S . Поэтому определение достоверности модели можно

считать наиболее важной проблемой при моделировании систем. От решения этой проблемы зависит степень доверия к результатам, полученным методом моделирования. Проверка модели на рассматриваемом подэтапе должна дать ответ на вопрос, насколько логическая схема модели системы и используемые математические соотношения отражают замысел модели, сформированный на первом этапе. При этом проверяются:

- а) возможность решения поставленной задачи;
- б) точность отражения замысла в логической схеме;
- в) полнота логической схемы модели;
- г) правильность используемых математических соотношений.

Только после того, как разработчик убеждается путем соответствующей проверки в правильности всех этих положений, можно считать, что имеется логическая схема модели системы S , пригодная для дальнейшей работы по реализации модели на ЭВМ.

2.4. Выбор инструментальных средств для моделирования.

На этом подэтапе необходимо окончательно решить вопрос о том, какую вычислительную машину (ЭВМ, АВМ, ГВК) и какое программное обеспечение целесообразно использовать для реализации модели системы S . Вообще, выбор вычислительных средств может быть проведен и на предыдущих подэтапах, но рассматриваемый подэтап является последним, когда этот выбор должен быть сделан окончательно, так как в противном случае возникнут трудности в проведении дальнейших работ по реализации модели. Вопрос о выборе ЭВМ сводится к обеспечению следующих требований:

- а) наличие необходимых программных и технических средств;
- б) доступность выбранной ЭВМ для разработчика модели;
- в) обеспечение всех этапов реализации модели;
- г) возможность своевременного получения результатов.

2.5. Составление плана выполнения работ по программированию.

Такой план должен помочь при программировании модели, учитывая оценки объема программы и трудозатрат на ее составление. План при использовании универсальной ЭВМ должен включать в себя: а) выбор языка (системы) программирования модели; б) указание типа ЭВМ и необходимых для моделирования устройств; в) оценку примерного объема необходимой оперативной и внешней памяти; г) ориентировочные затраты машинного времени на моделирование; д)

предполагаемые затраты времени на программирование и отладку программы на ЭВМ.

2.6. Спецификация и построение схемы программы. Спецификация программы — формализованное представление требований, предъявляемых к программе, которые должны быть удовлетворены при ее разработке, а также описание задачи, условия и эффекта действия без указания способа его достижения. Наличие логической блок-схемы модели позволяет построить схему программы, которая должна отражать:

- а) разбиение модели на блоки, подблоки и т. д.;
- б) особенности программирования модели;
- в) проведение необходимых изменений;
- г) возможности тестирования программы;
- д) оценку затрат машинного времени;
- е) форму представления входных и выходных данных.

Построение схемы программы представляет собой одну из основных задач на этапе машинной реализации модели. При этом особое внимание должно быть уделено особенностям выбранного для реализации модели языка: алгоритмического языка общего назначения или языка моделирования (например, *MATLAB*, *SIMULA*, *SJMSCRIPT*, *GPSS*).

2.7. Верификация и проверка достоверности схемы программы.

Верификация программы — доказательство того, что поведение программы соответствует спецификации на программу. Эта проверка является второй на этапе машинной реализации модели системы. Очевидно, что нет смысла продолжать работу по реализации модели, если нет уверенности в том, что в схеме программы, по которой будет вестись дальнейшее программирование, допущены ошибки, которые делают ее неадекватной логической схеме модели, а следовательно, и неадекватной самому объекту моделирования. При этом проводится проверка соответствия каждой операции,

представленной в схеме программы, аналогичной ей операции в логической схеме модели.

2.8. Проведение программирования модели.

При достаточно подробной схеме программы, которая отражает все операции логической схемы модели, можно приступить к программированию модели. Если имеется адекватная схема программы, то программирование представляет собой работу только для программиста без участия и помощи со стороны разработчика модели. При использовании пакетов прикладных программ моделирования проводится непосредственная генерация рабочих программ для моделирования конкретного объекта, т. е. программирование модели реализуется автоматизированном режиме.

2.9. Проверка достоверности программы.

Эта последняя проверка на этапе машинной реализации модели, которую необходимо проводить:

- а) обратным переводом программы в исходную схему;
- б) проверкой отдельных частей программы при решении различных тестовых задач;
- в) объединением всех частей программы и проверкой ее в целом на контрольном примере моделирования варианта системы 5.

На этом подэтапе необходимо также проверить оценки затрат машинного времени на моделирование. Полезно также получить достаточно простую аналитическую аппроксимацию зависимости затрат машинного времени от количества реализаций, что позволит разработчику модели (заказчику) правильно сформулировать требования к точности и достоверности результатов моделирования.