

ГОУ ВПО Уральский государственный технический
университет – УПИ
Кафедра «Автоматика и управление в технических системах»
Специальность 210100 – Управление и информатика в
технических системах

МОДЕЛИРОВАНИЕ СИСТЕМ

конспект лекций
(часть 2)

Преподаватель: Дружинина О.Г., доц., к.т.н.

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА МОДЕЛИРОВАНИЯ СИСТЕМ

Систематизации языков имитационного моделирования

Алгоритмические языки при моделировании систем служат вспомогательным аппаратом разработки, машинной реализации и анализа характеристик моделей.

Выбрав для решения задачи моделирования процесса функционирования системы конкретный язык, исследователь получает в распоряжение тщательно разработанную систему абстракций, предоставляющих ему основу для формализации процесса функционирования исследуемой системы S .

Высокий уровень проблемной ориентации языка моделирования значительно упрощает программирование моделей, а специально предусмотренные в нем возможности сбора, обработки и вывода результатов моделирования позволяют быстро и подробно анализировать возможные исходы имитационного эксперимента с моделью M_M .

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА МОДЕЛИРОВАНИЯ СИСТЕМ

Систематизации языков имитационного моделирования

Качество языков моделирования характеризуется:

- удобством описания процесса функционирования системы S;
- удобством ввода исходных данных моделирования и варьирования структуры, алгоритмов и параметров модели;
- реализуемостью статистического моделирования;
- эффективностью анализа и вывода результатов моделирования;
- простотой отладки и контроля работы моделирующей программы;
- доступностью восприятия и использования языка.

Достоинства и недостатки языков имитационного моделирования

Преимущества языков имитационного моделирования (ЯИМ) по сравнению с универсальными языками общего назначения (ЯОН) следующие:

- 1) язык моделирования содержит абстрактные конструкции, непосредственно отражающие понятия, в которых представлена формализованная модель, и близкие концептуальному уровню описания моделируемой системы. Это упрощает программирование имитатора, позволяет автоматизировать выявление многих ошибок в программах;
- 2) языки моделирования имеют эффективный встроенный механизм продвижения модельного времени (календарь событий, методы интегрирования и др.), средства разрешения временных узлов;
- 3) языки моделирования, как правило, содержат встроенные датчики случайных чисел, генераторы других типовых воздействий;

Достоинства и недостатки языков имитационного моделирования

Преимущества языков имитационного моделирования (продолжение)

- 4) в языках моделирования автоматизирован сбор стандартной статистики и других результатов моделирования, имеются средства автоматизации выдачи этих результатов в табличной или графической форме;
- 5) языки моделирования имеют средства, упрощающие программирование имитационных экспериментов, в частности, автоматизирующие установку начального состояния и перезапуск модели.

Подходы к разработке языков моделирования

Различают два подхода к разработке языков моделирования: непрерывный и дискретный - отражающие основные особенности исследуемых методом моделирования систем.

Поэтому ЯИМ делятся на две группы: для имитации непрерывных и дискретных процессов.

Для моделирования *дискретных* процессов используется ЭВМ, обладающая большей надежностью в эксплуатации, позволяющая получить высокую точность результатов и обеспечивающая адекватность дискретных моделей реальным процессам, протекающим в системах S .

Для моделирования *непрерывных* процессов могут быть использованы не только АВМ, но и ЭВМ, последние имитируют различные непрерывные процессы. Моделирование осуществляется с помощью численных решений дифференциальных уравнений при использовании некоторого стандартного пошагового метода.

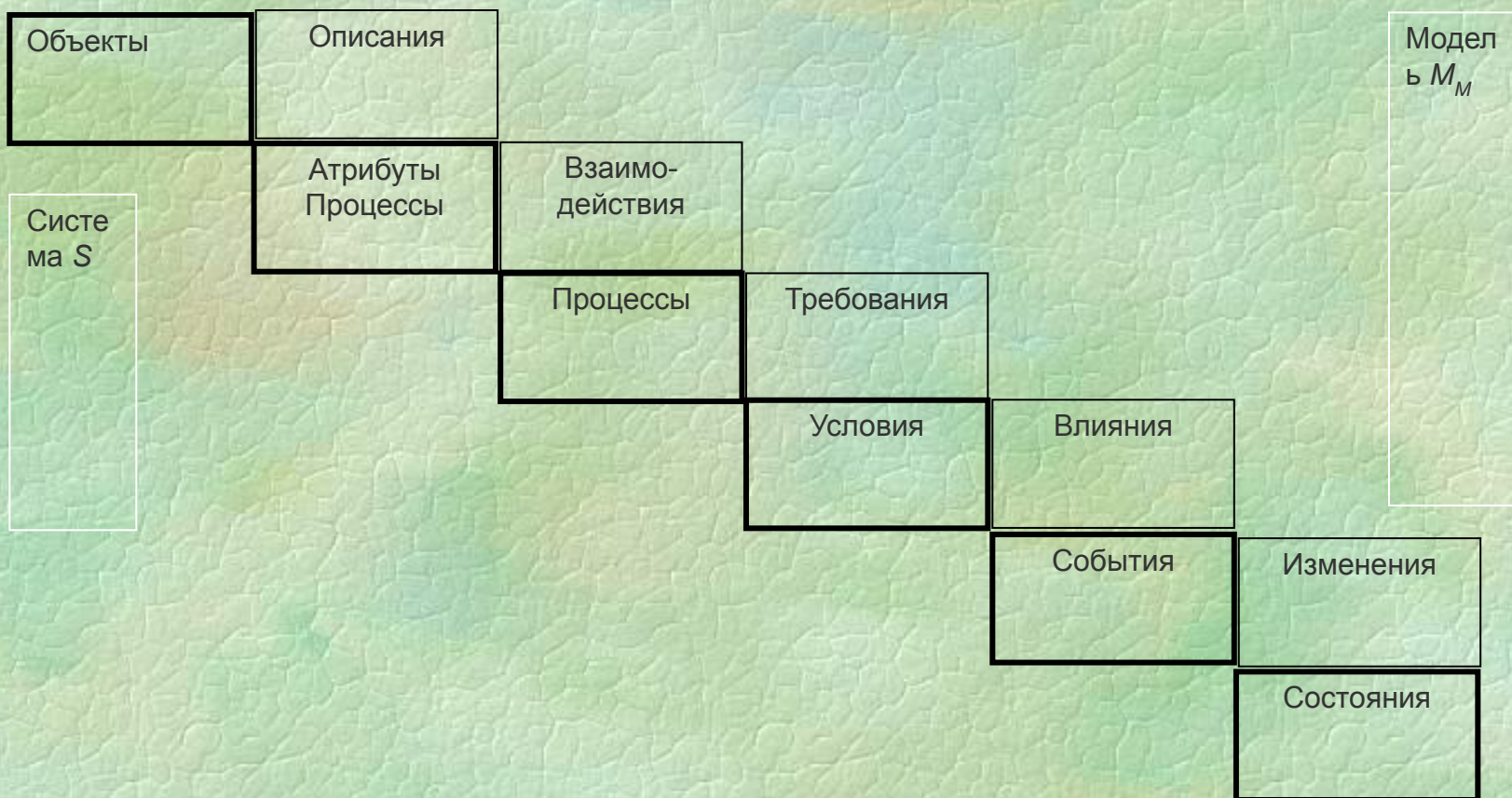
Архитектура языков моделирования

Архитектуру ЯИМ, т.е. концепцию взаимосвязей элементов языка как сложной системы, и технологию перехода от системы S к ее машинной модели M_M можно представить следующим образом:

- объекты моделирования (системы S) описываются (отображаются в языке) с помощью некоторых атрибутов языка;
 - атрибуты взаимодействуют с процессами, адекватными реально протекающим явлениям в моделируемой системе S ;
 - процессы требуют конкретных условий, определяющих логическую основу и последовательность взаимодействия этих процессов во времени;
 - условия влияют на события, имеющие место внутри объекта моделирования (системы S) и при взаимодействии с внешней средой E ;
- события изменяют состояния модели системы M в пространстве и во времени.

Архитектура языков моделирования

Типовая схема архитектуры ЯИМ и технология его использования при моделировании систем показана на рис. 5.1.



Задание времени в машинной модели

Функционирование модели M_M должно протекать в *искусственном* (не в реальном и не в машинном) времени, обеспечивая появление событий в требуемом логикой работы исследуемой системы порядке и с надлежащими временными интервалами между ними.

При этом надо учитывать, что элементы *реальной* системы S функционируют одновременно (*параллельно*), а компоненты машинной модели M_M действуют *последовательно*, так как реализуются с помощью ЭВМ последовательного действия.

Поскольку в различных частях объекта моделирования события могут возникать одновременно, то для сохранения адекватности причинно-следственных временных связей в ЯИМ имеется “механизм” задания времени для синхронизации действий элементов модели системы.

Основные *функции* этого механизма:

- 1) корректировка временной координаты состояния системы (“продвижение” времени, организация “часов”);
- 2) обеспечение согласованности различных блоков и событий в системе (синхронизация во времени, координация с другими блоками).

Задание времени в машинной модели

Существует два основных подхода к заданию времени: с помощью постоянных и переменных интервалов времени, которым соответствуют два принципа реализации моделирующих алгоритмов, т.е. “принцип Δt ” и “принцип dz ”.

Рассмотрим соответствующие способы управления временем в модели на примере, показанном на рис. 5.2, где по оси реального времени отложена последовательность событий в системе $\{s_i\}$ во времени, причем события s_4 и s_5 происходят одновременно (рис. 5.2, *a*).

Под действием событий s_i изменяются состояния модели z_i в момент времени t_{z_i} , причем такое изменение происходит скачком dz .

Задание времени в машинной модели

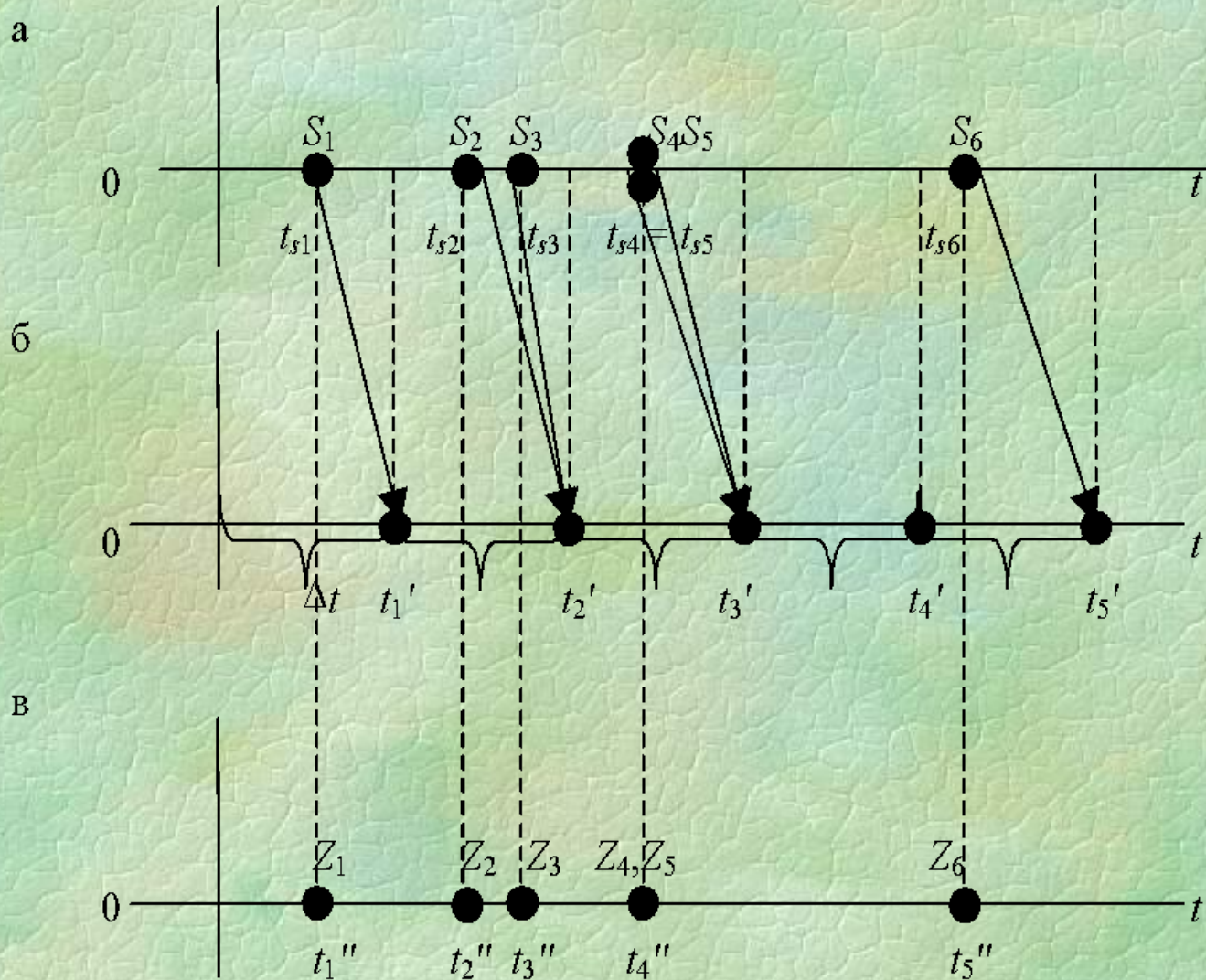


Рис. 5.2. Способы управления временем в модели системы

Задание времени в машинной модели

В модели, построенной по “принципу Δt ” (рис. 5.2, б), моменты системного времени будут последовательно принимать значения $t_1'=\Delta t, t_2'=2\Delta t, t_3'=3\Delta t, t_4'=4\Delta t, t_5'=5\Delta t$.

Эти моменты системного времени $t_j'(\Delta t)$ никак не связаны с моментами появления событий s_i , которые имитируются в модели системы.

Системное время при этом получает постоянное приращение, выбираемое и задаваемое перед началом имитационного эксперимента.

В модели, построенной по “принципу dz ” (рис. 5.2, в), изменение времени наступает в момент смены состояния системы, и последовательность моментов системного времени имеет вид:

$$t_1''=t_{z1}, t_2''=t_{z2}, t_3''=t_{z3}, t_4''=t_{z4}, t_5''=t_{z5},$$

т.е. моменты системного времени $t_k''(dz)$ непосредственно связаны с моментами появления событий в системе s_i .

Задание времени в машинной модели

У каждого из этих методов есть свои преимущества с точки зрения адекватного отражения реальных событий в системе S и затрат машинных ресурсов на моделирование.

При использовании “принципа dz ” события обрабатываются последовательно, и время смещается каждый раз вперед до начала следующего события.

В модели, построенной по “принципу Δt ”, обработка событий происходит по группам, пакетам или множествам событий.

При этом выбор Δt оказывает существенное влияние на ход процесса и результаты моделирования, и если Δt задана неправильно, то результаты могут получиться недостоверными, так как все события появляются в точке, соответствующей верхней границе каждого интервала моделирования.

При применении “принципа dz ” одновременная обработка событий в модели имеет место только тогда, когда эти события появляются одновременно и в реальной системе. Это позволяет избежать необходимости искусственного введения ранжирования событий при их обработке в конце интервала Δt .

Задание времени в машинной модели

При моделировании по “принципу Δt ” можно добиться хорошей аппроксимации: для этого Δt должно быть малым, чтобы два неодновременных события не попали в один и тот же временной интервал.

Но уменьшение Δt приводит к увеличению затрат машинного времени на моделирование, так как значительная часть тратится на корректировку “часов” и отслеживание событий, которых в большинстве интервалов может и не быть.

При этом даже при сильном “сжатии” Δt два неодновременных события могут попасть в один и тот же временной интервал Δt , что создает ложное представление об их одновременности.

Требования к языкам имитационного моделирования

В ЯИМ должен быть предусмотрен следующий набор программных средств и понятий, которые не встречаются в обычных ЯОН.

Совмещение. Параллельно протекающие в реальных системах S процессы представляются с помощью последовательно работающей ЭВМ. Языки моделирования позволяют обойти эту трудность путем введения понятия системного времени, используемого для представления упорядоченных во времени событий.

Размер. Большинство моделируемых систем имеет сложную структуру и алгоритмы поведения, а их модели велики по объему. Поэтому используют динамическое распределение памяти, когда компоненты модели системы M_M появляются в оперативной памяти ЭВМ или покидают ее в зависимости от текущего состояния. Важным аспектом реализуемости модели M_M на ЭВМ в этом случае является блочность ее конструкции, т.е. возможность разбиения модели на блоки, подлежащие и т.д.

Требования к языкам имитационного моделирования

Изменения. Динамические системы связаны с движением и характеризуются развитием процесса, вследствие чего пространственная конфигурация этих систем претерпевает изменения по времени. Поэтому во всех ЯИМ предусматривают обработку списков, отражающих изменения состояний процесса функционирования моделируемой системы S .

Взаимосвязанность. Условия, необходимые для свершения различных событий в модели M_M процесса функционирования системы S , могут оказаться весьма сложными из-за наличия большого количества взаимных связей между компонентами модели. Для разрешения связанных с этим вопросов трудностей в большинстве ЯИМ включают соответствующие логические возможности и понятия теории множеств.

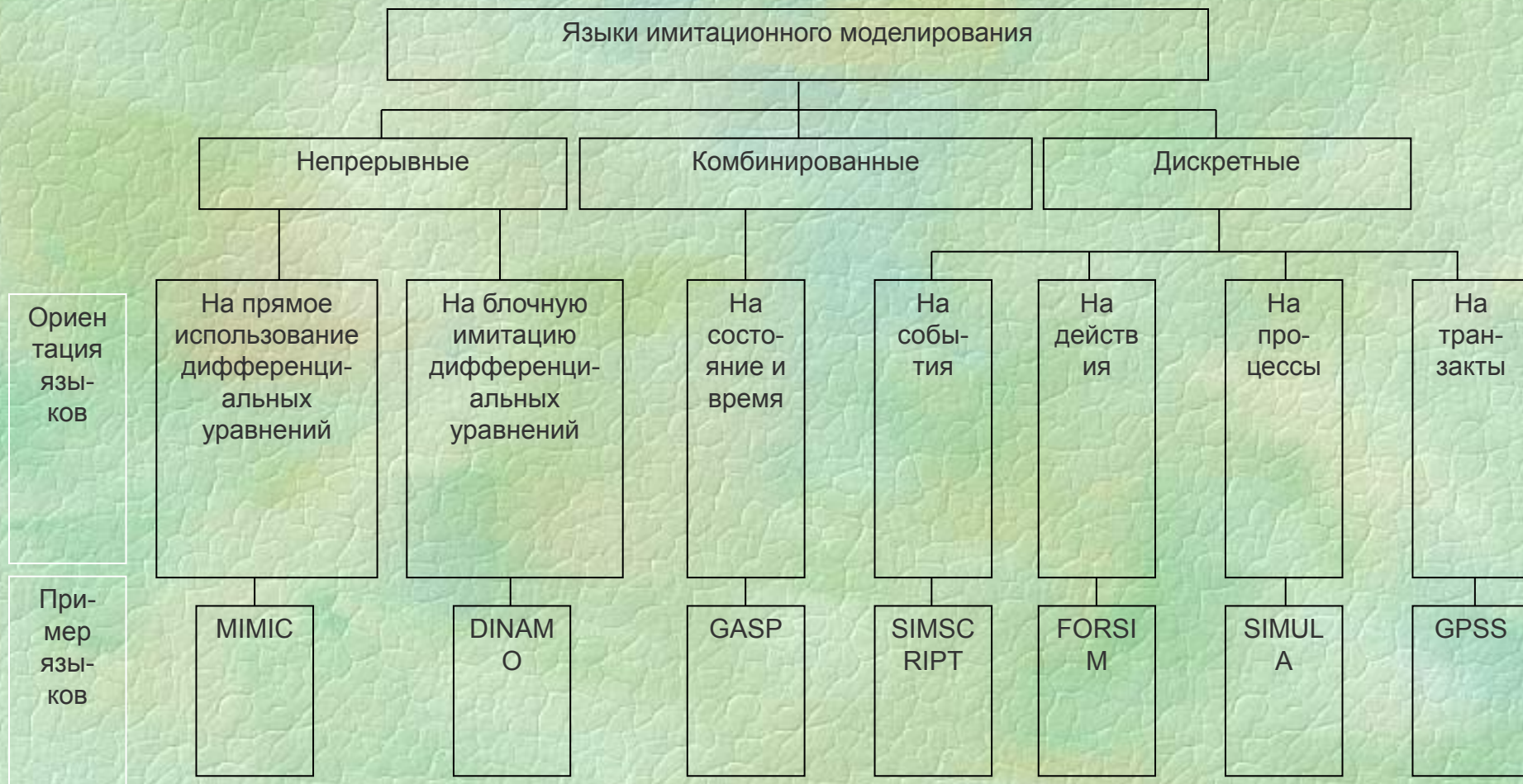
Требования к языкам имитационного моделирования

Стохастичность. Для моделирования случайных событий и процессов используют специальные программы генерации последовательностей псевдослучайных чисел, квазиравномерно распределенных на заданном интервале, на основе которых можно получить стохастические воздействия на модель M_M имитируемые случайными величинами с соответствующим законом распределения.

Анализ. Для получения наглядного и удобного в практическом отношении ответа на вопросы, решаемые методом машинного моделирования, необходимо получать статистические характеристики процесса функционирования модели системы. Поэтому предусматривают в языках моделирования способы статистической обработки и анализа результатов моделирования.

Классификации языков моделирования

Классификация языков для программирования моделей систем приведена на рис. 5.3.



Классификации языков моделирования

Для моделирования систем используются как универсальные и процедурно-ориентированные ЯОН, так и специализированные ЯИМ.

Имеющиеся ЯИМ можно разбить на три основные группы, соответствующие трем типам математических схем: непрерывные, дискретные и комбинированные.

Непрерывное представление системы S сводится к составлению дифференциальных уравнений, с помощью которых устанавливается связь между эндогенными и экзогенными переменными модели.

Например, это реализовано в языке MIMIC. Когда экзогенные переменные модели принимают дискретные значения, уравнения являются разностными. Такой подход реализован, например, в языке DYNAMO.

Классификации языков моделирования

Представление системы S в виде типовой схемы, в которой участвуют как непрерывные, так и дискретные величины, называется *комбинированным*.

Примером языка, реализующего комбинированный подход, является GASP.

Состояние модели системы описывается набором переменных, часть которых меняется во времени непрерывно.

Законы изменения непрерывных компонент заложены в структуру, объединяющую дифференциальные уравнения и условия относительно переменных.

Классификации языков моделирования

Предполагается, что в системе могут наступать события двух типов:

- 1) события, зависящие от состояния z_i ,
- 2) события, зависящие от времени t_i .

Для событий **первого** типа процесс моделирования состоит в продвижении системного времени от момента наступления события до следующего аналогичного момента.

События **второго** типа наступают в результате выполнения условий, относящихся к законам изменения непрерывных переменных.

События приводят к изменениям состояния модели системы и законов изменения непрерывных компонент.

Классификации языков моделирования

В рамках *дискретного подхода* можно выделить несколько принципиально различных групп ЯИМ.

Первая группа ЯИМ подразумевает наличие списка событий, отличающих моменты начала выполнения операций. Продвижение времени осуществляется по событиям, в моменты наступления которых производятся необходимые операции, включая операции пополнения списка событий.

Примером языка событий является язык SIMSCRIPT.

Модель M_M состоит из элементов, с которыми происходят события, представляющие собой последовательность предложений, изменяющих состояния моделируемой системы в различные моменты времени.

Классификации языков моделирования

Моделирование с помощью языка SIMSCRIPT включает в себя следующие этапы:

- элементы моделируемой системы S описываются и вводятся с помощью карт определений;
- вводятся начальные условия;
- фиксируются и вводятся исходные значения временных параметров;
- составляются подпрограммы для каждого события;
- составляется перечень событий и указывается время свершения каждого эндогенного события.

Классификации языков моделирования

Команды языка SIMSCRIPT группируются следующим образом: операции над временными объектами, арифметические и логические операции и команды управления, команды ввода-вывода, специальные команды обработки результатов.

К центральным понятиям языка SIMSCRIPT относятся обработка списков с компонентами, определяемыми пользователем, и последовательность событий в системном времени.

При этом имеются специальные языковые средства для работы с множествами.

Классификации языков моделирования

При использовании ЯИМ **второй** группы после пересчета системного времени просмотр действий с целью проверки выполнения условий начала или окончания какого-либо действия производится непрерывно.

Просмотр действий определяет очередность появления событий.

Языки данного типа имеют в своей основе поисковый алгоритм, и динамика системы S описывается в терминах действий.

Примером языка действий (работ) является ЯИМ FORSIM, представляющий собой пакет прикладных программ, который позволяет оперировать только фиксированными массивами данных, описывающих объекты моделируемой системы. С его помощью нельзя имитировать системы переменного состава.

Язык FORSIM удобен для описания систем с большим числом разнообразных ресурсов, так как он позволяет записывать условия их доступности в компактной форме.

Классификации языков моделирования

Третья группа ЯИМ описывает системы, поведение которой определяется процессами, т.е. последовательностью событий, связь между которыми устанавливается с помощью набора специальных отношений.

Пример языка процессов - язык SIMULA, в котором осуществляется блочное представление моделируемой системы S .

Процесс задается набором признаков, характеризующих его структуру, и программой функционирования.

Функционирование каждого процесса разбивается на этапы, протекающие в системном времени.

Классификации языков моделирования

В четвертую группу выделены ЯИМ типа GPSS, хотя принципиально их можно отнести к группе языков процессов.

Язык GPSS представляет собой интерпретирующую языковую систему, применяющуюся для описания пространственного движения объектов.

Такие динамические объекты в языке GPSS называются *транзактами* и представляют собой элементы потока.

В процессе имитации транзакты “создаются” и “уничтожаются”. Функцию каждого из них можно представить как движение через модель M_M с поочередным воздействием на ее блоки. GPSS-программа генерирует и передает транзакты из блока в блок в соответствии с правилами, устанавливаемыми блоками.

Каждый переход транзакта приписывается к определенному моменту системного времени.

Сравнение эффективности языков

При анализе эффективности использования для моделирования конкретной системы S того или иного ЯИМ (или ЯОН) выделяют несколько важных свойств языков:

- возможность описания структуры и алгоритмов поведения исследуемой системы S в терминах языка;
- простота применения для построения модели M , ее машинной реализации и обработки результатов моделирования;
- предпочтение пользователя, обычно отдаваемое языку, который ему более знаком или который обладает большей степенью универсальности, и т.д.

Классификации языков моделирования

Результаты сравнения различных языков при моделировании сведены в табл. 5.1.

Языки даны в порядке уменьшения их эффективности.

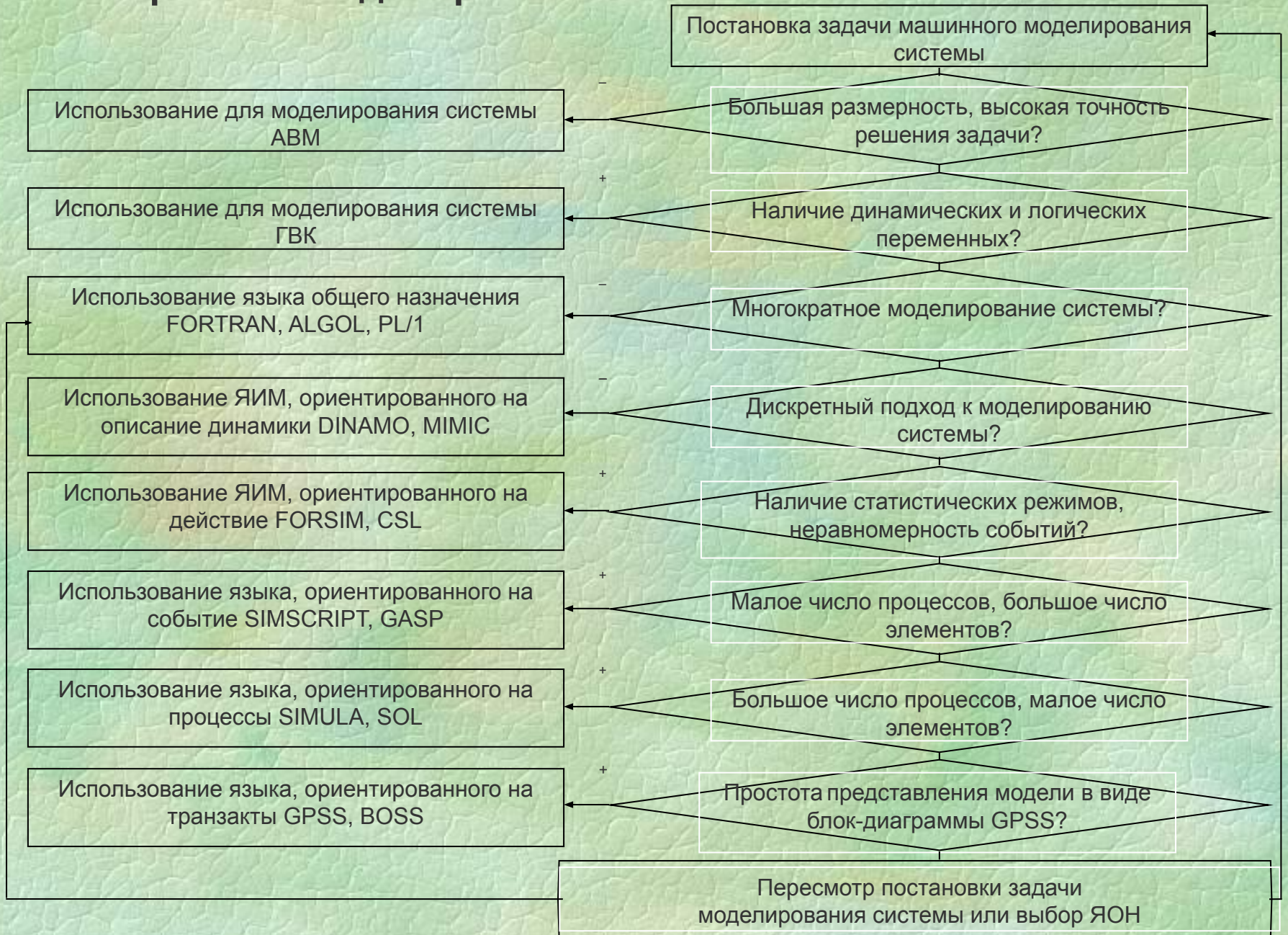
Таблица 5.1

Возможности языка пользователя	Простота применения	Предпочтение
SIMULA	GPSS	SIMSCRIPT
SIMSCRIPT	SIMSCRIPT	GPSS
GPSS	SIMULA	SIMULA

Выбор языка моделирования системы

Основываясь на классификации языков (рис. 5.3) и исходя из оценки эффективности (табл. 5.1), выбор языка для решения задачи машинного моделирования конкретной системы S можно представить в виде *дерева решений* (рис. 5.4).

Выбор языка моделирования системы



Выбор языка моделирования системы

Исходя из постановки задачи машинного моделирования конкретной системы S , поставленных целей, выбранных критериев оценки эффективности и заданных ограничений (блок 1),

можно сделать вывод о размерности задачи моделирования и требуемой точности и достоверности ее решения (блок 2).

Для задач большой размерности моделирование на АВМ (блок 3) позволяет получить достаточно высокую точность.

При этом АВМ позволит наглядно выявить компромисс между сложностью и точностью модели M ,

проиллюстрирует влияние изменения параметров и переменных на характеристики модели системы и т.п.

Выбор языка моделирования системы

Если в модели M при моделировании системы S имеют место как непрерывные, так и дискретные переменные, отражающие динамику системы и логику ее поведения (блок 4), то рекомендуется использовать для моделирования ГVK (блок 5).

Если моделирование конкретной системы S представляет собой единичный акт (блок 6), то, вероятно, в ущерб концептуальной выразительности модели M_K и отладочным средствам для проверки логики машинной модели M_M следует выбрать более распространенные и более гибкие ЯОН (блок 7).

Если при моделировании на универсальной ЭВМ выбран непрерывный подход (блок 8), то следует остановить выбор на одном из языков, позволяющих отразить динамику системы при наличии обратных связей (блок 9).

При этом могут быть приняты языки непрерывного типа DYNAMO, MIMIC либо комбинированные (дискретно-непрерывные) — GASP.

Выбор языка моделирования системы

Если в основу модели M положена дискретная математическая схема и в ней при построении моделирующего алгоритма используется “принцип Δt ” или “принцип dz ”, причем имитируются взаимодействующие элементы статической природы при неравномерности событий во времени (блок 10),

то рационально воспользоваться ЯИМ, ориентированным на действия, например FORSIM, CSL.

Если в модели M описывается малое число взаимодействующих процессов и имеется большое число элементов (блок 12),

то целесообразно выбрать для построения моделирующих алгоритмов “принцип Δt ” и остановиться на ЯИМ событий (блок 13), например SIMSCRIPT, GASP и т. п.

Выбор языка моделирования системы

Если для программирования модели более эффективен ЯИМ, позволяющий описать большое число взаимодействующих процессов (блок 14), то следует использовать языки процессов (блок 15), которые не связаны с использованием блоков только определенных типов, например в транзактных языках.

Наиболее распространенными языками описания процессов являются языки SIMULA и SOL.

И наконец, если предпочтение отдается блочной конструкции модели M при наличии минимального опыта в программировании (блок 16),

то следует выбирать ЯИМ транзактов типа GPSS, BOSS (блок 17), но при этом надо помнить, что они негибки и требуют большого объема памяти и затрат машинного времени на прогон программ моделирования.

Выбор языка моделирования системы

Если перечисленные средства по той или иной причине не подходят для целей моделирования конкретной системы S (блок 18), то надо снова провести модификацию модели M либо попытаться решить задачу с использованием ЯОН на универсальной ЭВМ.

Выбор языка моделирования системы

Примеры дискретных, непрерывных и комбинированных ЯИМ приведены в табл. 5.2.

Таблица 5.2

Название языка	Тип	Примечание
ACSL	Непрерывный	Advanced Continuous Simulation Language
ALSIM	Дискретный	ALgol SIMulation
ANDSIM	Непрерывный	ANALog Digital SIMulator
APLIS	Комбинированный	APL Simulation
CDL	Дискретный	Computer Design Language
COSMO	Непрерывный	Compartment System MOdeling
CSL	Дискретный	Control and Simulation Language
CSSL	Непрерывный	Continuous System Simulation Language
DEMOS	Дискретный	Discrete Event MOdeling in Simula
DIANA	Комбинированный	DIgital-ANALog simulator
DISCO	Комбинированный	DISrete-COntinuous
DYNAMO	Непрерывный	DYNAmic MOdels

Выбор языка моделирования системы

Примеры дискретных, непрерывных и комбинированных ЯИМ приведены в табл. 5.2.

Окончание табл. 5.2

Название языка	Тип	Примечание
FORSIM	Непрерывный	Forran ORiented Simulator
GASP	Комбинированный	General Activity Simulation Program
GEMS	Непрерывный	General Equation Modeling System
GPSS	Дискретный	General Processing Simulation System
IPSS	Дискретный	Information processing Simulation System
MIDAS	Непрерывный	Modified Integration Digital Analog Simulator
SAM	Непрерывный	Simulation of Analogue Methods
SIMSCRIPT	Дискретный	SIMulation SCRIPTure
SIMULA	Дискретный	SIMUlation LAnguage
SOL	Дискретный	Simulation Oriented Language
VANS	Дискретный	Value Added Network Simulator

Пакеты прикладных программ моделирования систем

Автоматизированная система моделирования (АСМ) позволяет повысить эффективность выполнения пользователем следующих этапов имитационного моделирования:

- преобразование к типовым математическим схемам элементов моделируемой системы S и построение схем сопряжения;
- обработка и анализ результатов моделирования системы S ;
- реализация интерактивного режима с пользователем в процессе моделирования системы S .

Пакеты прикладных программ моделирования (ППМ) – это комплекс взаимосвязанных программ моделирования и средств системного обеспечения (программных и языковых), предназначенных для автоматизации решения задач моделирования.

Весь круг работ, связанных с разработкой алгоритмов и программ моделирования, а также с подготовкой и проведением машинных экспериментов, называется *автоматизацией моделирования* и реализуется в виде конкретных АСМ.

Структура пакетов прикладных программ моделирования систем

В структуре ППМ можно выделить три основных компонента:

- функциональное наполнение,
- язык заданий
- системное наполнение.

Функциональное наполнение пакета.

Функциональное наполнение ППМ отражает специфику предметной области применительно к конкретному объекту моделирования, т. е. системе S , и представляет собой совокупность модулей.

Под модулем здесь понимается конструктивный элемент, используемый на различных стадиях функционирования пакета. Язык (языки), на котором записываются модули функционального наполнения, будем называть *базовым языком* ППМ.

Структура пакетов прикладных программ моделирования систем

Функциональное наполнение пакета

Состав функционального наполнения пакета, его мощность или полнота охвата им предметной области отражают объем прикладных знаний, заложенных в ППМ, т.е. потенциальный уровень тематической квалификации пакета.

Одной из ключевых проблем разработки ППМ является *модуляризация*, т.е. разбиение функционального наполнения пакета на модули. Различают программные модули, модули данных и модули документации.

Структура пакетов прикладных программ моделирования систем

Язык заданий пакета.

Язык заданий ППМ является средством общения пользователя (разработчика или исследователя машинной модели M_M процесса функционирования системы S) с пакетом.

Он позволяет описывать последовательность выполнения различных операций, обеспечивающих решение задачи моделирования, или постановку задачи моделирования, по которой эта последовательность строится автоматически.

Именно через язык заданий пользователь воспринимает и оценивает, какие “вычислительные услуги” предоставляет АСМ и насколько удобно их использование.

Структура пакетов прикладных программ моделирования систем

Язык заданий пакета.

Общая структура и стиль языка заданий ППМ в значительной степени зависят от режима работы, принятой в обслуживаемой пакетом предметной области.

Можно выделить два основных (в определенном смысле противоположных) **режима проведения моделирования**:

- *активный*, предусматривающий при создании конкретных рабочих программ модели M_M модификацию и настройку имеющихся модулей функционального наполнения, а также разработку новых модулей;
- *пассивный*, предусматривающий проведение машинных экспериментов с моделью M_M без модификации функционального наполнения ППМ.

Структура пакетов прикладных программ моделирования систем

Активный режим работы предназначен для специалистов, создающих программное обеспечение АСМ.

Пассивный режим – для деятельности так называемых конечных пользователей, т.е. специалистов, которые не обязательно имеют высокий уровень подготовки в области программирования.

Характерной особенностью языков заданий пакетов, обслуживающих проведение моделирования в *активном* режиме, является их направленность на описание схем программ моделирования процессов, причем центральное место в таких языках (их обычно называют языками сборки) занимают средства:

- конструирования схем программ, в которых указывается порядок выполнения и взаимодействия модулей при моделировании конкретной системы S ;
- развития или модификации функционального наполнения ППМ;
- управления процессами генерации и исполнения рабочей программы, реализующей задание пользователя.

Структура пакетов прикладных программ моделирования систем

Главная цель разработки языка заданий ППМ, обеспечивающего решение задач моделирования в *пассивном* режиме, заключается в том, чтобы “спрятать” от конечного пользователя основную массу алгоритмических подробностей моделирования конкретной системы.

Такие языки, называемые *языками запросов*, ориентированы обычно на формулирование содержательных постановок задач, т.е. запросов, указывающих, “что необходимо получить”, без явного задания того, “как это получить”.

Язык заданий ППМ может быть реализован как в форме *самостоятельного* языка, так и в форме *встроенного* языка, т.е. расширения существующего языка программирования.

Структура пакетов прикладных программ моделирования систем

Системное наполнение пакета.

Системное наполнение ППМ представляет собой совокупность программ, которые обеспечивают выполнение заданий и взаимодействие пользователя с пакетом.

Системное наполнение организует использование потенциала знаний, заложенных в функциональном наполнении в соответствии с возможностями, предусмотренными в языке заданий ППМ. Реализация функций системного наполнения ППМ осуществляется на основе согласованного использования:

- штатных общецелевых средств системного обеспечения;
- средств системного наполнения, расширяющих и сопрягающих возможности компонентов штатного обеспечения;
- специальных средств системного наполнения, выполняющих управляющие, архивные и обрабатывающие процедуры, с учетом специфики моделирования процесса функционирования системы *S*.

Структура пакетов прикладных программ моделирования систем

Язык, на котором пишутся программы системного наполнения пакета, называется *инструментальным языком* ППМ.

Традиционными составляющими системного наполнения пакета являются:

- резидентный монитор, осуществляющий интерфейс как между отдельными компонентами системного наполнения, так и между ними и штатным программным обеспечением;
- транслятор входных заданий, формирующий внутреннее представление заданий и реализуемый обычно в виде макрогенератора или препроцессора;
- интерпретатор внутреннего представления задания;
- архив функционального наполнения (подсистема хранения программного материала);
- банк данных об объекте моделирования и машинном эксперименте;
- монитор организации процесса машинного моделирования (взаимодействия модулей по данным и управлению);
- планировщик процесса машинного моделирования, который определяет последовательность выполнения модулей, реализующую задание ППМ;
- монитор организации интерактивного взаимодействия с пользователем (исследователем системы S).

Программные средства ППМ

ППМ – комплекс программных средств и документов, предназначенных для реализации функционального завершеного алгоритма моделирования процесса функционирования системы S и обеспечивающих автоматизацию управления ведением эксперимента с моделью M_M на ЭВМ.

ППМ не является набором готовых программ для проведения машинных экспериментов с моделью M_M , а представляет собой набор средств для разработки конкретных, удовлетворяющих требованиям пользователя рабочих программ моделирования, служащих для автоматизации определенных функций при построении модели, машинном эксперименте и обработке результатов моделирования системы S .

Программные средства ППМ

К программным средствам ППМ относятся:

- 1) набор программных модулей (тело пакета), из которых в соответствии с требованиями пользователя по заданному алгоритму набирается конкретная рабочая программа моделирования заданного объекта;
- 2) управляющая программа, представляющая собой аналог супервизора ЭВМ;
- 3) средства генерации рабочих программ для конкретного применения при решении задач моделирования систем.

Программные средства ППМ

При *динамической* генерации заранее оговариваются все варианты рабочей программы моделирования системы S , которые могут потребоваться пользователю при машинном эксперименте с моделью M_m .

При решении конкретной задачи моделирования, т.е. перед каждым новым прогоном программы в ходе машинного эксперимента, вводится специальная параметрическая карта, определяющая требуемый на этом прогоне вариант программы.

Монитор пакета собирает необходимые модули и помещает их в оперативную память ЭВМ для решения задачи моделирования. Условия проведения машинного эксперимента при динамической генерации являются более гибкими, но при этом увеличиваются затраты машинных ресурсов на моделирование (увеличивается необходимый объем памяти и время моделирования каждого варианта модели системы S).

Программные средства ППМ

Кроме использования программных модулей, входящих в тело ППМ, пользователь имеет возможность подключать свои *собственные* программы моделирования в точках пользователя. Имеется также возможность замены имеющихся модулей ППМ на собственные, что еще больше расширяет возможности моделирования различных вариантов систем.

Кроме программных средств ППМ содержит комплект *документов*. В состав комплекта документов ППМ входят проектная документация, являющаяся документацией разработчиков пакета, и пользовательская, необходимая для эксплуатации пакета при решении конкретных задач моделирования.

Структура АСМ

Структурно АСМ можно разбить на следующие комплексы программ:

- 1) формирования базы данных об объекте моделирования (БДО);
- 2) формирования базы данных о машинном эксперименте (БДЭ);
- 3) моделирования процесса функционирования объекта;
- 4) расширения возможностей ППМ;
- 5) организации различных режимов работы ППМ.

Структура АСМ

Комплекс программ *формирования БДО* реализует все работы по созданию в АСМ сведений о моделируемом объекте, т.е. системе *S*. Информация об объекте может корректироваться. Для формирования БДО требуются следующие программы:

- ввода данных об объекте (сведения об элементах системы, типовых математических схемах и операторах их сопряжения);
 - корректировки введенной информации;
 - перевода в стандартную форму;
 - диспетчеризации процедур ввода;
- формирования БДО (расположения информации во внешней памяти).

Структура АСМ

В результате работы комплекса программ *формирования БДЭ* в АСМ формируется база данных, т.е. сведения, достаточные для проведения конкретных экспериментов с машинной моделью объекта M_m . Для формирования БДЭ необходимы следующие программы:

- ввода данных о планируемом эксперименте (сведений о факторах, реакциях, начальных состояниях и т.п.);
- формирования БДЭ (выделения сведений из БДО, необходимых и достаточных для реализации конкретного машинного эксперимента с моделью M_m);
- корректировки введенной информации о машинном эксперименте;
- расположения информации в архивах во внешней и оперативной памяти ЭВМ.

Структура АСМ

Комплекс программ моделирования процесса реализует план ведения машинных экспериментов, их организацию на ЭВМ и обработку промежуточных данных и результатов эксперимента, взаимодействие с пользователем. Для решения задачи моделирования требуются следующие программы:

- управления машинным экспериментом,
- реализация стратегии эксперимента и его диспетчеризация;
- машинной имитации, включая организацию вычислений и взаимосвязь модулей модели M_M ;
- обработки и выдачи результатов моделирования системы S в различных режимах взаимодействия с пользователем.

Структура АСМ

Комплекс программ расширения возможностей ППМ призван обеспечить пользователя средствами генерации новых программ моделирования при различных перестройках

(объекта моделирования, машинного эксперимента, обработки результатов и т.п.).

Комплекс программ организации различных режимов работы ППМ, кроме основной работы по диспетчеризации процесса функционирования ППМ, призван организовать его работу в режиме диалога с пользователем как на этапе ввода данных об объекте моделирования и эксперименте, корректировки БДО и БДЭ, так и непосредственно в ходе машинного эксперимента с моделью M_m использования мультимедиа технологий.

Необходимо также обеспечить режим коллективного пользования пакетом.

Диалоговая система и банк данных моделирования

Одно из самых *перспективных* направлений в области машинного моделирования – оснащение существующих АСМ машин графикой и диалоговыми средствами.

Системы моделирования с интерактивной графикой могут существенно повысить эффективность исследования систем, формализуемых в виде непрерывных (детерминированных и стохастических) схем,

когда пользователю выдаются на дисплей выходные характеристики (переменные) модели системы S и имеется возможность визуального сравнения с помощью буферизации на выходе старых и новых решений.

Диалоговая система и банк данных моделирования

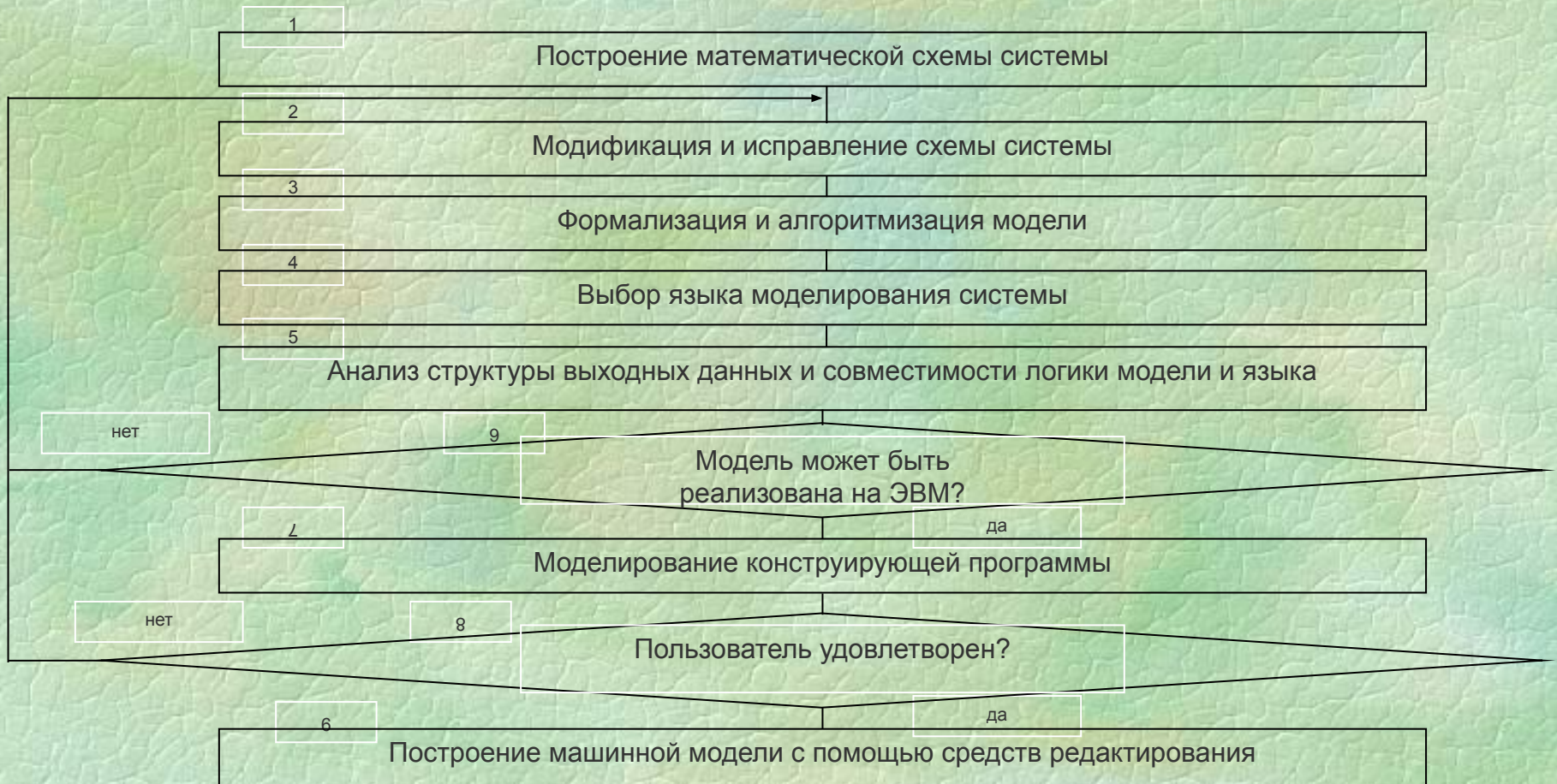
Особенности диалоговых систем моделирования коллективного пользования:

- возможность одновременной работы многих пользователей, занятых разработкой одной системы S ;
- доступ пользователей к программно-техническим ресурсам системы моделирования, включая распределенные банки данных и пакеты прикладных программ моделирования;
- обеспечение диалогового режима работы с различными вычислительными машинами и устройствами, включая цифровые и аналоговые вычислительные машины, установки физического моделирования, элементы реальных систем и т.п.;
- диспетчирование работ в АСМ и оказание различных услуг пользователям, включая обучение работе с диалоговой системой моделирования;
- использование сетевых технологий.

Диалоговая система и банк данных моделирования

Рассмотрим организацию интерактивного процесса создания моделирующей программы.

Схема построения машинной модели с помощью программного генератора приведена на рис. 5.5.



Диалоговая система и банк данных моделирования

После построения на основе концептуальной модели M_k математической схемы моделируемой системы S (блок 1) у пользователя имеется возможность ее дальнейшей модификации и исправлений, вносимых в структуру, алгоритмы поведения и параметры (блок 2).

Затем с помощью средств ППМ проводятся формализация и алгоритмизация модели M (блок 3).

Далее пользователь имеет возможность выбрать, исходя из особенностей модели и имеющихся в его распоряжении программно-технических средств, язык моделирования ЯОН или ЯИМ (блок 4),

а в интерактивном режиме проанализировать структуру входных данных моделирования и проверить совместимость логики модели системы и выбранного для моделирования языка (блок 5).

Диалоговая система и банк данных моделирования

В зависимости от этого решается вопрос о реализуемости модели на ЭВМ (блок 6):

если модель требует внесения изменений, связанных с возможностью ее машинной реализации,

то проводится ее модификация и доработка схемы (блок 2), а если модель реализуют на ЭВМ, то конструируют программу моделирования системы (блок 7).

И если такая программа по своим свойствам удовлетворяет пользователя (блок 8), то в режиме диалога реализуется окончательное построение рабочей программы машинной модели M_m с помощью имеющихся средств редактирования (блок 9).

В том случае, когда, по мнению пользователя, конструкция программы не является удовлетворительной, имеется возможность дальнейших модификаций и исправления схемы модели системы (блок 2).

Банк данных моделирования

Банк данных моделирования –

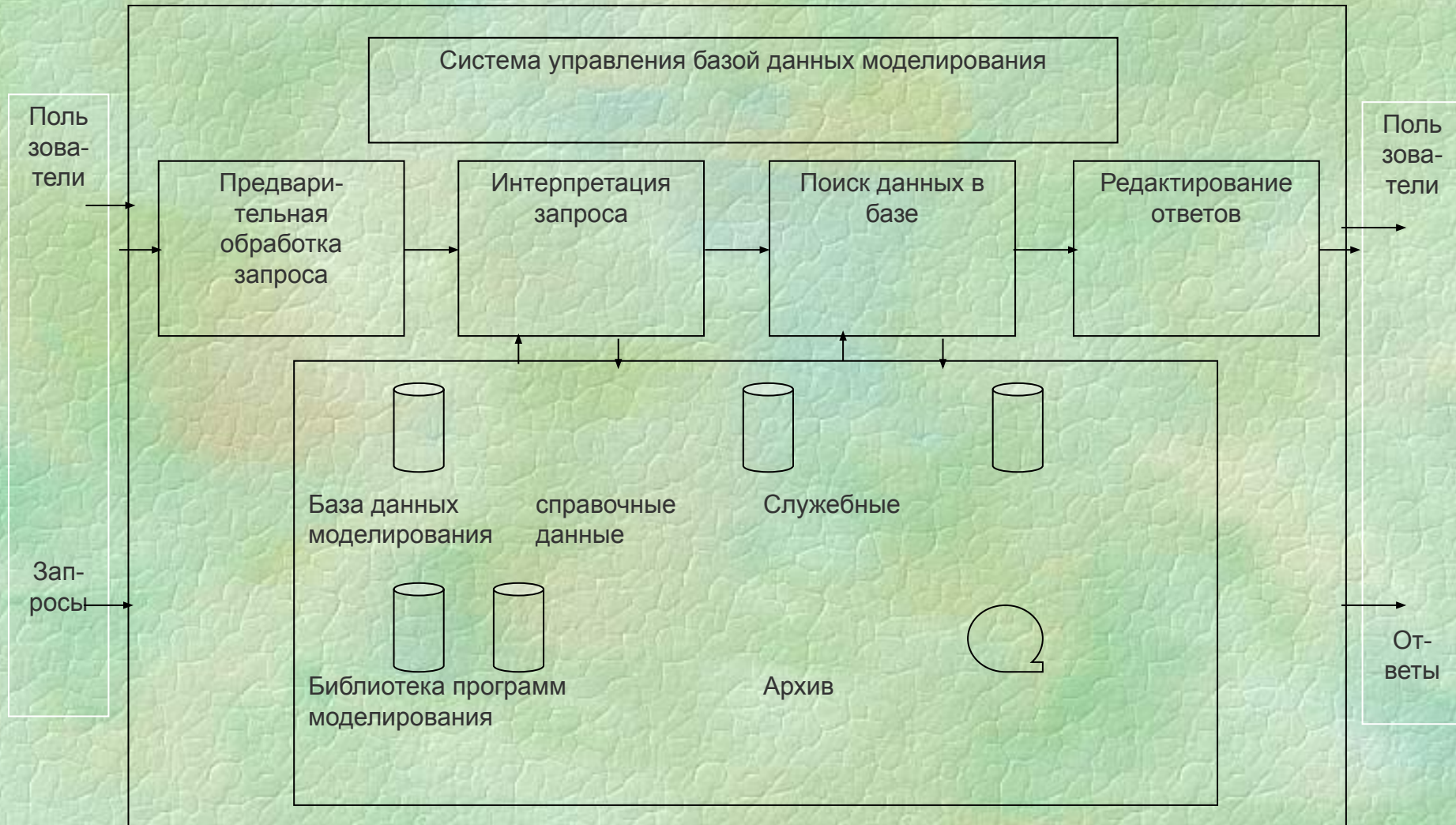
организационно-программно-техническая система, представляющая собой совокупность БДО и БДЭ, программных и технических средств формирования и ведения этих баз и коллектива специалистов, обеспечивающих функционирование банка.

Банк данных моделирования включает в себя следующие *структурные элементы*:

- информационный фонд – организованную совокупность данных моделирования, т.е. базы данных объекта и эксперимента;
- специализированное математическое обеспечение для управления банком данных;
- информационные языки для описания и манипулирования с данными об объекте моделирования и машинном эксперименте;
- администратор банка данных;
- справочные и служебные данные, необходимые для нормального функционирования банка данных;

Банк данных моделирования

Структура банка данных АСМ показана на рис. 5.6.



Банк данных моделирования

Пользователи взаимодействуют с базой данных моделирования в диалоговом режиме с помощью набора специальных языков.

Для обращения пользователей к базе данных необходим *информационный язык запросов*.

Для записи схем баз данных применяется *язык описания данных*.

Общее управление работой банка данных осуществляется либо с помощью *языка управления* заданиями операционной системы, либо с помощью специально для этой цели создаваемого языка управления.

Администратор банка данных осуществляет внешнюю координацию всей работы банка и выполняет операции, не поддающиеся формализации.

В его функции входят создание баз данных, согласование требований пользователей, управление восстановлением при сбоях, анализ статистики, оценка и обеспечение эффективности работы с базой данных, управление загрузкой баз данных, реорганизация банка (изменение схем и баз данных), генерация и развитие СУБД.

Банк данных моделирования

При *поиске* данных происходят обращение к рабочей области в памяти ЭВМ и выборка искомых данных по объекту и эксперименту или корректировка данных в базе.

Найденные в базе данные контролируются и анализируются, а затем редактируются *ответы* пользователю, выдаваемые на печать или устройство отображения.

Таким образом, база данных моделирования, имеющаяся в банке данных, позволяет создать единое (интегрированное) информационное обеспечение АСМ,

т.е. в ней хранится как информация, необходимая для построения моделей различных объектов, так и информация, необходимая для планирования и проведения машинного эксперимента.

Моделирующие комплексы

В машинном моделировании систем используется вычислительная техника трех типов: ЭВМ, АВМ и ГВК. ГВК называются гибридными или *аналого-цифровыми моделирующими комплексами* (АЦМК).

Рассмотрим **достоинства** и **недостатки** этих трех типов вычислительных средств (АВМ, ЭВМ и ГВК) применительно к машинному моделированию систем.

АВМ значительно уступают ЭВМ по точности и логическим возможностям,
но по быстродействию, схемной простоте, сопрягаемости с датчиками внешней информации не уступают им.

Моделирующие комплексы

Характерные черты для АВМ:

- зависимые переменные модели системы S представляются в непрерывном виде;
- точность результатов моделирования определяется качеством компонентов электрических схем АВМ;
- возможность одновременного выполнения параллельных вычислительных операций;
- возможность выполнения операций в реальном или ускоренном масштабе времени;
- операции сложения, вычитания, умножения, дифференцирования, интегрирования, генерирования непрерывных функций выполняются весьма эффективно, но имеются ограниченные возможности выполнения логических действий, накопления цифровых данных, обеспечения длительных задержек, обработки информации;

Моделирующие комплексы

Характерные черты для АВМ: (продолжение)

- технология программирования состоит в основном в замещении элементами АВМ (такими, как операционные усилители, интеграторы и т. п.) соответствующих элементов моделируемой системы S ;
- к АВМ можно подключить блоки реальной системы S при комбинированном моделировании;
- пользователь имеет возможность в ходе машинного эксперимента на АВМ изменять значения установок, т.е. коэффициентов, устанавливаемых на АВМ, что обеспечивает более наглядное проведение эксперимента с моделью системы S .

Моделирующие комплексы

Характерные черты ЭВМ:

- вся обработка промежуточной и результирующей информации в процессе моделирования системы S реализуется в дискретном виде;
- все операции по работе с машинной моделью M_m выполняются последовательно;
- точность результатов моделирования системы S определяется главным образом выбранными численными методами решения задачи и формой представления чисел;
- время решения определяется сложностью задачи моделирования системы S , т.е. числом операций, необходимых для получения результатов моделирования;
- наличие компромисса между временем решения и точностью результатов моделирования системы S ;

Моделирующие комплексы

Характерные черты ЭВМ: (продолжение)

- применяется ограниченное число арифметических операций (сложение, вычитание, умножение и деление), но с помощью численных методов можно в модели на базе этих исходных операций реализовать и более сложные, например дифференцирование, интегрирование и т.д.;
- для выполнения логических операций и принятия решений в процессе моделирования используются как цифровые, так и нецифровые данные;
- предусматриваются операции с плавающей запятой, что устраняет трудности масштабирования модели;
- методы программирования базируются как на ЯОН, так и на ЯИМ.

Моделирующие комплексы

Современные ГВК представляют собой попытку объединить все лучшее, присущее цифровой и аналоговой технике, и избежать их недостатков.

Аналоговая часть ГВК позволяет увеличить скорость вычислений и распараллелить процессы.

Цифровая часть ГВК дает возможность:

- управлять аналоговой частью машинной модели M_m при высоком быстродействии;
- использовать устройства запоминания и хранения данных моделирования;
- обеспечивать более высокую точность вычислений и применения логических операций при моделировании системы.

Моделирующие комплексы

Преимущества ГВК:

- сочетает быстродействие АВМ и точность ЭВМ, что позволяет расширить класс моделируемых объектов;
- в процессе машинного моделирования позволяет использовать реальные технические средства и части исследуемой конкретной системы S;
- обеспечивает гибкость аналогового моделирования благодаря использованию логики и памяти ЭВМ;
- увеличивает быстродействие ЭВМ за счет использования аналоговых подпрограмм;
- делает возможной обработку входной информации о модели системы, представленной частично в дискретной и непрерывной формах.

При использовании ГВК существенно упрощается вопрос взаимодействия с датчиками, установленными на реальных объектах, что позволяет, в свою очередь, проводить комбинированное моделирование с использованием аналого-цифровой части модели и натурной части объекта.

Моделирующие комплексы

Аналого-ориентированные комплексы используются в тех случаях, когда не требуется высокая точность результатов и когда моделируемая система S реализуема аналоговыми средствами. Системы такого класса исследуются на АЦМК, в которых цифровые средства необходимы на этапе подготовки модели для автоматизации набора задачи, накопления и обработки результатов моделирования. Сама же модель системы S реализуется исключительно на аналоговом вычислителе (аналоговое моделирование). Наряду с указанными функциями ЭВМ может выполнять задачи управления АВМ в процессе реализации модели. АЦМК с цифровым управлением и цифровой логикой способны воспроизводить более сложные модели по сравнению со стандартными АВМ. К аналого-ориентированным АЦМК относятся также комплексы, в которых ЦВМ применяются в качестве периферийного оборудования.

Моделирующие комплексы

К *цифроориентированным комплексам* можно отнести универсальные ЭВМ, где для отображения и регистрации результатов используются аналоговые средства – осциллографы, самописцы и т.д.

В таких АЦМК модель M_m полностью реализуется цифровыми методами. Возможны варианты построения АЦМК для полунатурного моделирования, когда реальная аппаратура стыкуется с ЭВМ через аналоговый вычислитель.

В *цифроориентированных АЦМК* может иметь место распараллеливание отдельных вычислительных процедур в процессе работы с цифровой моделью M_m за счет реализации их аналоговыми средствами.

Моделирующие комплексы

Сбалансированные (универсальные) комплексы являются самым мощным средством для решения задач аналого-цифрового моделирования.

В их состав входят средства, с помощью которых могут эффективно решаться не только аналого-цифровые задачи, но и задачи аналоговые с цифровым управлением, а также задачи цифрового моделирования.

На комплексах такого типа широко используется диалог “оператор – машинный эксперимент”, т.е. могут запоминаться, отображаться и регистрироваться результаты решений, оперативно вносятся изменения в модель M_m и может осуществляться ее запуск.

Другими словами, имеется возможность реализовать итеративный процесс исследования, сводящийся к получению искомого результата, что особенно важно при автоматизации проектирования системы S на базе машинного моделирования.

ПЛАНИРОВАНИЕ МАШИННЫХ ЭКСПЕРИМЕНТОВ С МОДЕЛЯМИ СИСТЕМ

Машинный эксперимент

Цель машинного эксперимента с моделью системы – получение информации о характеристиках процесса функционирования объекта.

Основная задача планирования машинных экспериментов – получение необходимой информации об исследуемой системе S при ограничениях на ресурсы (затраты машинного времени, памяти).

Частные задачи планирования машинных экспериментов – уменьшение затрат машинного времени на моделирование, увеличение точности и достоверности результатов моделирования, проверка адекватности модели и т. д.

План эксперимента определяет объем и порядок проведения вычислений на ЭВМ, приемы накопления и статистической обработки результатов моделирования системы S .

Таким образом, при машинном моделировании рационально планировать и проектировать не только саму модель M_m системы S , но и процесс ее использования, т. е. проведение с ней экспериментов с использованием устройств центрального ЭВМ.

ПЛАНИРОВАНИЕ МАШИННЫХ ЭКСПЕРИМЕНТОВ С МОДЕЛЯМИ СИСТЕМ

Машинный эксперимент

При планировании эксперимента важное значение имеют следующие моменты:

- простота повторения условий эксперимента на ЭВМ с моделью M_m системы S ;
- возможность управления экспериментом с моделью M_m , включая его прерывание и возобновление;
- легкость варьирования условий проведения эксперимента (воздействий внешней среды E);
- наличие корреляции между последовательностью точек в процессе моделирования;
- трудности, связанные с определением интервала моделирования $(0, T)$.

ПЛАНИРОВАНИЕ МАШИННЫХ ЭКСПЕРИМЕНТОВ С МОДЕЛЯМИ СИСТЕМ

Машинный эксперимент

Преимуществами машинных экспериментов перед натурным являются:

- возможность полного воспроизведения условий эксперимента с моделью исследуемой системы S для сравнения двух и более альтернатив;
- простота прерывания и возобновления машинных экспериментов для анализа результатов и принятия решений об его дальнейшем ходе.

Недостатком машинных экспериментов является наличие корреляции в выходных последовательностях, т.е. результаты одних наблюдений зависят от результатов одного или нескольких предыдущих, и поэтому в них содержится меньше информации, чем в независимых наблюдениях.

Основные понятия планирования экспериментов

Если цель эксперимента – изучение влияния переменной x на переменную y , то x – фактор, а y – реакция.

В экспериментах с машинными моделями M_m системы S фактор является экзогенной или управляемой (входной) переменной, реакция – эндогенной (выходной) переменной.

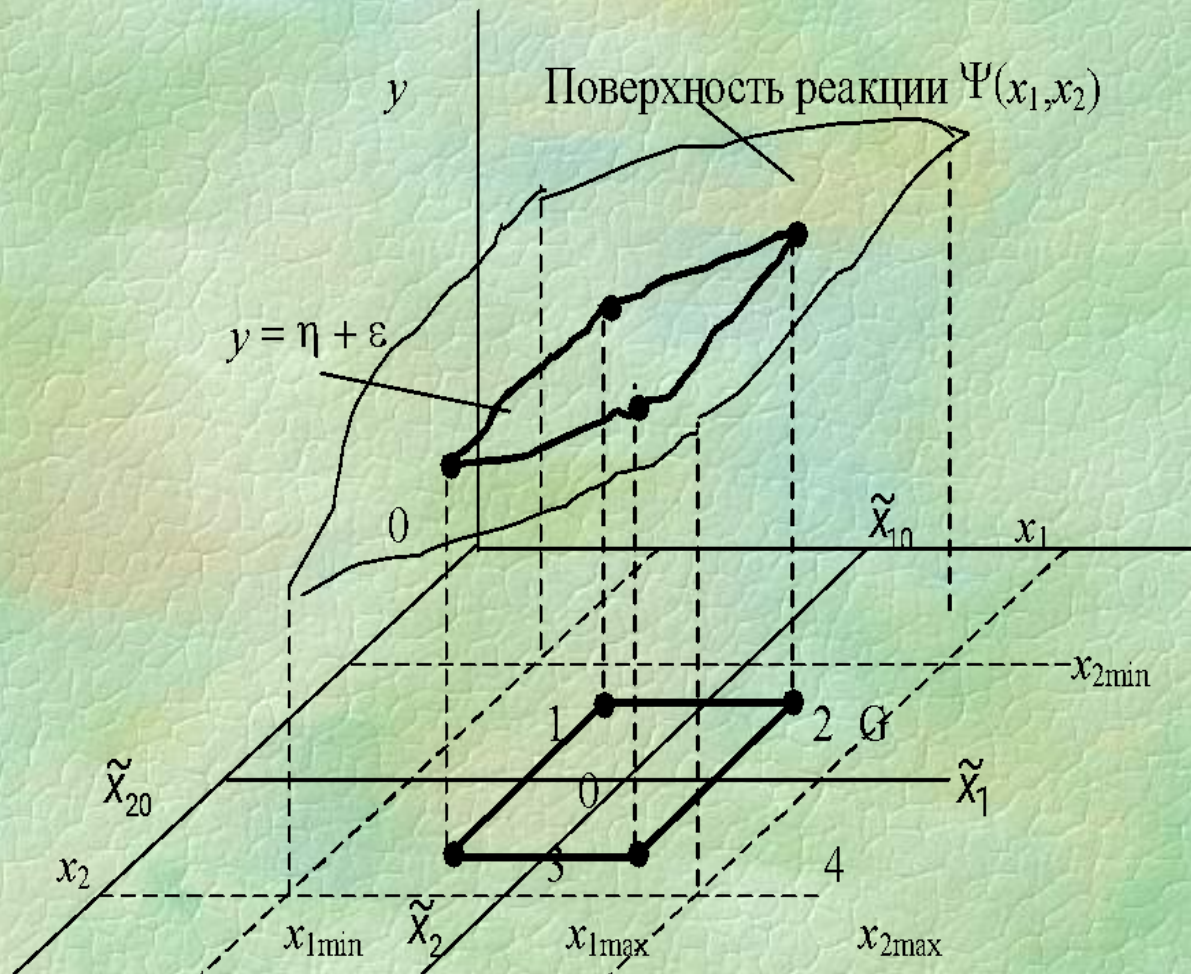
Каждый фактор x_i , $i=1, k$, может принимать в эксперименте одно из нескольких значений, называемых *уровнями*.

Фиксированный набор уровней факторов определяет одно из возможных состояний рассматриваемой системы.

Каждому фиксированному набору уровней факторов соответствует определенная точка в многомерном пространстве, называемом *факторным пространством*.

Эксперименты не могут быть реализованы во всех точках факторного пространства, а реализуется лишь в точках, принадлежащих допустимой области, как, например, это показано для случая двух факторов x_1 и x_2 на рис. 6.1 (плоскость $x_1 O x_2$).

Геометрическое представление поверхности реакции



Основные понятия планирования экспериментов

Связь между уровнями факторов и реакцией (откликом) системы представим в виде соотношения

$$y_l = \Psi_l(x_1, x_2, \dots, x_k), \quad l=1, m.$$

Функцию Ψ_l , связывающую реакцию с факторами, называют *функцией реакции*, а геометрический образ, соответствующий функции реакции, – *поверхностью реакции*.

Исследователю заранее не известен вид зависимостей Ψ_l , $l=1, m$, поэтому используют приближенные соотношения:

$$\tilde{y}_l = \phi_l(x_1, x_2, \dots, x_k), \quad .$$

Зависимости ϕ_l находятся по данным эксперимента.

Основные понятия планирования экспериментов

При планировании экспериментов необходимо определить основные свойства факторов.

Факторы при проведении экспериментов могут быть:

- управляемыми и неуправляемыми,
- наблюдаемыми и ненаблюдаемыми,
- изучаемыми и неизучаемыми,
- количественными и качественными,
- фиксированными и случайными.

Фактор называется *управляемым*, если его уровни целенаправленно выбираются исследователем в процессе эксперимента.

Основные понятия планирования экспериментов

Фактор называется *наблюдаемым*, если его значения наблюдаются и регистрируются. Обычно в машинном эксперименте наблюдаемые факторы совпадают с управляемыми, так как нерационально управлять фактором, не наблюдая его. Но неуправляемый фактор также можно наблюдать.

Например, на этапе проектирования конкретной системы S нельзя управлять заданными воздействиями внешней среды E , но можно наблюдать их в машинном эксперименте. Наблюдаемые неуправляемые факторы называются *сопутствующими*.

Фактор относится к *изучаемым*, если он включен в модель M_m для изучения свойств системы S , а не для вспомогательных целей, например для увеличения точности эксперимента.

Основные понятия планирования экспериментов

Фактор будет *количественным*, если его значения — числовые величины, влияющие на реакцию, а в противном случае фактор называется *качественным*. Например, в модели системы, формализуемой в виде схемы массового обслуживания (Q -схемы), количественными факторами являются интенсивности входящих потоков заявок, интенсивности потоков обслуживания, емкости накопителей, количество обслуживающих каналов и т.д., а качественными факторами — дисциплины постановки в очередь, выбора из очереди, обслуживания заявок каналами и т.д. Качественным факторам в отличие от количественных соответствует условная порядковая шкала, а не числовая.

Фактор называется *фиксированным*, если в эксперименте исследуются все интересующие экспериментатора значения фактора, а если экспериментатор исследует только некоторую случайную выборку из совокупности интересующих значений факторов, то фактор называется *случайным*.

Основные понятия планирования экспериментов

Основными требованиями, предъявляемыми к факторам, являются требование управляемости фактора и требование непосредственного воздействия на объект.

При планировании эксперимента обычно одновременно изменяются несколько факторов. Основные требования, которые предъявляются к совокупности факторов, – совместимость и независимость. *Совместимость* факторов означает, что все их комбинации осуществимы, а *независимость* соответствует возможности установления фактора на любом уровне независимо от уровней других.

Основные понятия планирования экспериментов

При проведении машинного эксперимента с моделью M_m необходимо выявить влияние факторов, находящихся в функциональной связи с искомой характеристикой. Для этого необходимо:

- отобрать факторы $x_i, i=1, k$, влияющие на искомую характеристику, и описать функциональную зависимость;
- установить диапазон изменения факторов $x_{i\min} \div x_{i\max}$;
- определить координаты точек факторного пространства $\{x_1, x_2, \dots, x_k\}$, в которых следует проводить эксперимент;
- оценить необходимое число реализации и их порядок в эксперименте.

Модели планирования экспериментов

Существуют различные методы (модели) планирования.

Для экстремального планирования экспериментов наибольшее применение нашли модели в виде алгебраических полиномов. Предполагаем, что изучается влияние k количественных факторов x_i , $i=1, k$, на некоторую реакцию η в отведенной для экспериментирования локальной области факторного пространства G , ограниченной $x_{i\min} - x_{i\max}$, $i=1, k$ (см. рис. 6.1 для случая $k = 2$). Допустим, что функцию реакции $\phi(x_1, x_2, \dots, x_k)$ можно с некоторой степенью точности представить в виде **полинома** степени d от k переменных, который содержит C_{k+d}^k коэффициентов.

$$\eta = \tilde{b}_0 + \sum_{i=1}^k \tilde{b}_i x_i + \sum_{i=1}^k \sum_{j=i+1}^k \tilde{b}_{ij} x_i x_j + \sum_{i=1}^k \sum_{j=i+1}^k \sum_{l=j+1}^k \tilde{b}_{ijl} x_i x_j x_l + \dots$$

Данный полином является частным случаем и применим для полного факторного эксперимента 2^k , где k – количество факторов. Для оценки коэффициентов можно применить методы линейной регрессии.