МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ПРИБОРОСТРОЕНИЯ И ИНФОРМАТИКИ

Русаков Алексей Михайлович
RusakovAM.ru

Исследование и моделирование сложных систем

Москва 2014

Оглавление

ВВЕДЕНИЕ

МОДЕЛИРОВАНИЕ КАК МЕТОД НАУЧНОГО ПОЗНАНИЯ

Методологическая основа моделирования. Все то, на что направлена человеческая деятельность, называется *объектом* (лат. *objection* - предмет). Выработка методологии направлена на упорядочение получения и обработки информации об объектах, которые существуют вне нашего сознания и взаимодействуют между собой и внешней средой.

В научных исследованиях большую роль играют *гипотезы,* т. е. определенные предсказания, основывающиеся на небольшом количестве опытных данных, наблюдений, догадок. Быстрая и полная проверка выдвигаемых гипотез может быть проведена в ходе специально поставленного эксперимента. При формулировании и проверке правильности гипотез большое значение в качестве метода суждения имеет аналогия.

Аналогией называют суждение о каком-либо частном сходстве двух объектов, причем такое сходство может быть существенным и несущественным. Необходимо отметить, что понятия существенности и несущественности сходства или различия объектов условны и относительны. Существенность сходства (различия) зависит от уровня абстрагирования и в общем случае определяется конечной целью проводимого исследования. Современная научная гипотеза создается, как правило, по аналогии с проверенными на практике научными положениями. Таким образом, аналогия связывает гипотезу с экспериментом.

Гипотезы и аналогии, отражающие реальный, объективно существующий мир, должны обладать наглядностью или сводиться к удобным для исследования логическим схемам; такие логические схемы, упрощающие рассуждения и логические построения или позволяющие проводить эксперименты, уточняющие природу явлений, называются *моделями.* Другими словами, ***модель*** (лат. *modulus* — мера) — это объект-заместитель объекта-оригинала, обеспечивающий изучение некоторых свойств оригинала.

Модель

- Материальная
  - Геометрические
  - Физические
  - Предметно-математические
- Идеальная
  - Мысленные
  - Логико-математические

Компьютерная модель – это программная реализация математической модели, дополненная различными служебными программами (например, рисующими и изменяющими графические образы во времени). Компьютерная модель имеет две составляющие – программную и аппаратную. Программная составляющая так же является абстрактной знаковой моделью. Это лишь другая форма абстрактной модели, которая, однако, может интерпретироваться не только математиками и программистами, но и техническим устройством – процессором компьютера.

Моделированием называется замещение одного объекта другим с целью получения информации о свойствах объекта-оригинала путем изучения объекта-модели.

Таким образом, моделирование может быть определено как представление объекта моделью для получения информации об этом объекте путем проведения экспериментов с его моделью. Теория замещения одних объектов (оригиналов) другими объектами (моделями) и исследования свойств объектов на их моделях называется *теорией моделирования*.

ГЛАВА 1
ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МОДЕЛИРОВАНИЯ СИСТЕМ

В настоящее время при анализе и синтезе сложных (больших) систем получил развитие *системный подход,* который отличается от классического (или индуктивного - путем перехода *от частного к общему* и синтезирует (конструирует) систему путем слияния ее компонент, разрабатываемых раздельно) подхода. В отличие от этого *системный подход* предполагает последовательный переход *от общего к частному*, когда в основе рассмотрения лежит цель, причем исследуемый объект выделяется из окружающей среды.

Понятие системы и элемента системы. Специалисты по проектированию и эксплуатации сложных систем имеют дело с системами управления различных уровней, обладающими общим свойством - стремлением достичь некоторой цели. Эту особенность учтем в следующих определениях системы.

Система S — целенаправленное множество взаимосвязанных элементов любой природы.

Внешняя среда E — множество существующих вне системы элементов любой природы, оказывающих влияние на систему или находящихся под ее воздействием.

Понятие модели. Модель – представление объекта, системы или понятия, в некоторой форме, отличного от их реального существования.

Моделирование – во-первых, построение модели, во-вторых, изучение модели, в-третьих, анализ системы на основе данной модели.

При системном подходе к моделированию систем необходимо прежде всего четко определить *цель моделирования*. Применительно к вопросам моделирования цель возникает из требуемых задач моделирования, что позволяет подойти к выбору критерия и оценить, какие элементы войдут в создаваемую модель M. Поэтому необходимо иметь критерий отбора отдельных элементов в создаваемую модель.

Цели моделирования:

1) оценка – оценить действительные характеристики проектируемой или существующей системы, определить насколько система предлагаемой структуры будут соответствовать предъявляемым требованиям.

2) сравнение – произвести сравнение конкурирующих систем одного функционального назначения или сопоставить несколько вариантов построения одной и той же системы.

3) прогноз – оценить поведение системы при некотором предполагаемом сочетании рабочих условий.

4) анализ чувствительности – выявить из большого числа факторов, действующих на систему тем, которое в большей степени влияют на ее поведение и определяют ее показатели эффективности.

5) оптимизация – найти или установить такое сочетание действующих факторов и их величин, которое обеспечивает наилучшие показатели эффективности системы в целом.

1-4 задачи анализа, 5 - задача синтеза.

Подходы к исследованию систем. Важным для системного подхода является определение *структуры системы* — совокупности связей между элементами системы, отражающих их взаимодействие.

При *структурном подходе* выявляются состав выделенных элементов системы S и связи между ними. Совокупность элементов и связей между ними позволяет судить о структуре системы. Последняя в зависимости от цели исследования может быть описана на разных уровнях рассмотрения. Наиболее общее описание структуры — это топологическое описание, позволяющее определить в самых общих понятиях составные части системы и хорошо формализуемое на базе теории графов.

Менее общим является функциональное описание, когда рассматриваются отдельные функции, т. е. алгоритмы поведения системы, и реализуется *функциональный подход,* оценивающий функции, которые выполняет система, причем под функцией понимается свойство, приводящее к достижению цели.

Простой подход к изучению взаимосвязей между отдельными частями модели предусматривает рассмотрение их как отражение связей между отдельными подсистемами объекта. Такой классический подход может быть использован при создании достаточно простых моделей. Процесс синтеза модели M на основе классического (индуктивного) подхода представлен на рис. 1.1, *а.* Реальный объект, подлежащий моделированию, разбивается на отдельные подсистемы, т. е. выбираются исходные данные *Д* для моделирования и ставятся цели *Ц,* отображающие отдельные стороны процесса моделирования. По отдельной совокупности исходных данных *Д* ставится цель моделирования отдельной стороны функционирования системы, на базе этой цели формируется некоторая компонента K будущей модели. Совокупность компонент объединяется в модель M.

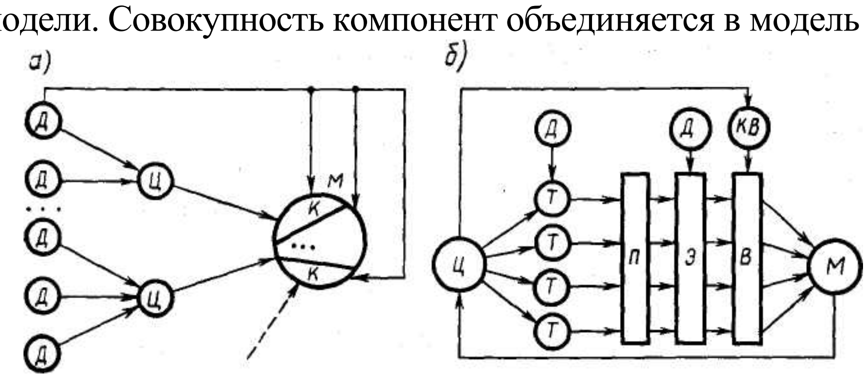


Рис. 1.1. Процесс синтеза модели на основе классического *(а)* и системного (б) подходов

Таким образом, разработка модели *М* на базе классического подхода означает суммирование отдельных компонент в единую модель, причем каждая из компонент решает свои собственные задачи и изолирована от других частей модели. Поэтому классический подход может быть использован для реализации сравнительно простых моделей, в которых возможно разделение и взаимно независимое рассмотрение отдельных сторон функционирования реального объекта. Для модели сложного объекта такая разобщенность решаемых задач недопустима, так как приводит к значительным затратам ресурсов при реализации модели на базе конкретных программно-технических средств. Можно отметить две отличительные стороны классического подхода: наблюдается движение от частного к общему, создаваемая модель (система) образуется путем суммирования отдельных ее компонент и не учитывается возникновение нового системного эффекта.

Процесс синтеза модели *М* на базе системного подхода условно представлен на рис. 1.1, *б.* На основе исходных данных *Д,* которые известны из анализа внешней системы, тех ограничений, которые накладываются на систему сверху либо исходя из возможностей ее реализации, и на основе цели функционирования формулируются исходные требования *Т* к модели системы *S.* На базе этих требований формируются ориентировочно некоторые подсистемы *П,* элементы Э и осуществляется наиболее сложный этап синтеза — выбор *В* составляющих системы, для чего используются специальные критерии выбора *КВ*.

Стадии разработки моделей. На базе системного подхода может быть предложена и некоторая последовательность разработки моделей, когда выделяют две основные стадии проектирования: *макропроектирование* и *микропроектирование*.

На стадии *макропроектирования* на основе данных о реальной системе *S* и внешней среде *Е* строится модель внешней среды, выявляются ресурсы и ограничения для построения модели системы, выбирается модель системы и критерии, позволяющие оценить адекватность модели *М* реальной системы *S.*

Стадия *микропроектирования* в значительной степени зависит от конкретного типа выбранной модели. В случае имитационной модели необходимо обеспечить создание информационного, математического, технического и программного обеспечений систем моделирования.

Независимо от типа используемой модели *М* при ее построении необходимо руководствоваться рядом *принципов системного подхода*:

1) пропорционально-последовательное продвижение по этапам и направлениям создания модели;

2) согласование информационных, ресурсных, надежностных и других характеристик;

3) правильное соотношение отдельных уровней иерархии в системе моделирования;

4) целостность отдельных обособленных стадий построения модели.

КЛАССИФИКАЦИЯ ВИДОВ МОДЕЛИРОВАНИЯ СИСТЕМ

Классификация видов моделирования систем *S* приведена на рис. 1.2.

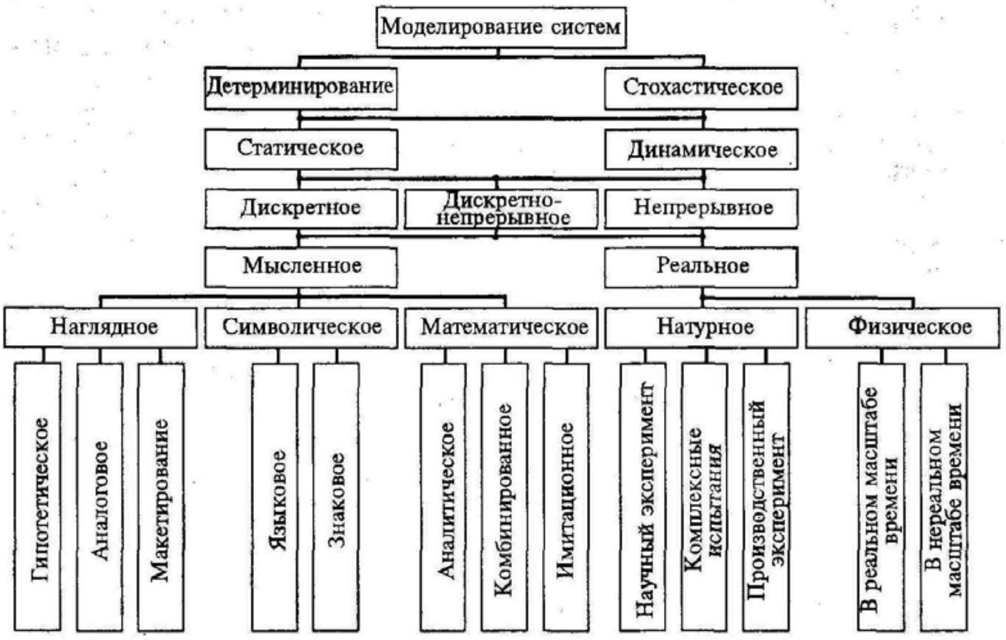


Рис. 1.2. Классификация видов моделирования систем

Детерминированное моделирование отображает процессы, в которых предполагается отсутствие всяких случайных воздействий; *стохастическое моделирование* отображает вероятностные процессы и события. В этом случае анализируется ряд реализаций случайного процесса и оцениваются средние характеристики, т. е. набор однородных реализаций. *Статическое моделирование* служит для описания поведения объекта в какой-либо момент времени, а *динамическое моделирование* отражает поведение объекта во времени. *Дискретное моделирование* служит для описания процессов, которые предполагаются дискретными, соответственно непрерывное моделирование позволяет отразить непрерывные процессы в системах, а *дискретно-непрерывное моделирование* используется для случаев, когда хотят выделить наличие как дискретных, так и непрерывных процессов.

В зависимости от формы представления объекта (системы S) можно выделить мысленное и реальное моделирование.

Мысленное моделирование часто является единственным способом моделирования объектов, которые либо практически нереализуемы в заданном интервале времени, либо существуют вне условий, возможных для их физического создания. Например, на базе мысленного моделирования могут быть проанализированы многие ситуации микромира, которые не поддаются физическому эксперименту. Мысленное моделирование может быть реализовано в виде *наглядного, символического* и *математического*.

При *наглядном моделировании* на базе представлений человека о реальных объектах создаются различные наглядные модели, отображающие явления и процессы, протекающие в объекте. В основу *гипотетического моделирования* исследователем закладывается некоторая гипотеза о закономерностях протекания процесса в реальном объекте, которая отражает уровень знаний исследователя об объекте и базируется на причинно-следственных связях между входом и выходом изучаемого объекта. Гипотетическое моделирование используется, когда знаний об объекте недостаточно для построения формальных моделей.

Аналоговое моделирование основывается на применении аналогий различных уровней. Наивысшим уровнем является полная аналогия, имеющая место только для достаточно простых объектов. С усложнением объекта используют аналогии последующих уровней, когда аналоговая модель отображает несколько либо только одну сторону функционирования объекта.

Существенное место при мысленном наглядном моделировании занимает *макетирование.* Мысленный макет может применяться в случаях, когда протекающие в реальном объекте процессы не поддаются физическому моделированию, либо может предшествовать проведению других видов моделирования. Если ввести условное обозначение отдельных понятий, т. е. знаки, а также определенные операции между этими знаками, то можно реализовать *знаковое моделирование* и с помощью знаков отображать набор понятий — составлять отдельные цепочки из слов и предложений. Используя операции объединения, пересечения и дополнения теории множеств, можно в отдельных символах дать описание какого-то реального объекта.

В основе *языкового моделирования* лежит некоторый тезаурус. Последний образуется из набора входящих понятий, причем этот набор должен быть фиксированным. Следует отметить, что между тезаурусом и обычным словарем имеются принципиальные различия. Тезаурус — словарь, в котором каждому слову может соответствовать лишь единственное понятие, хотя в обычном словаре одному слову могут соответствовать несколько понятий.

Символическое моделирование представляет собой искусственный процесс создания логического объекта, который замещает реальный и выражает основные свойства его отношений с помощью определенной системы знаков или символов.

Математическое моделирование. Под *математическим моделированием* будем понимать процесс установления соответствия данному реальному объекту некоторого математического объекта, называемого математической моделью, и исследование этой модели, позволяющее получать характеристики рассматриваемого реального объекта. Вид математической моли зависит как от природы реального объекта, так и задач исследования объекта и требуемой достоверности и точности решения этой задачи.

Для *аналитического моделирования* характерно то, что процессы функционирования элементов системы записываются в виде некоторых функциональных соотношений (алгебраических, интегродиференциальных, конечно-разностных и т.п.) или логических условий.

Имитационное моделирование позволяет по исходным данным получить сведения о состоянии процесса в определенные моменты времени, дающие возможность оценить характеристики системы S.

МАТЕМАТИЧЕСКИЕ МЕТОДЫ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ

Основные этапы построения математической модели:

1. составляется описание функционирования системы в целом;
2. составляется перечень подсистем и элементов с описанием их функционирования, характеристик и начальных условий, а также взаимодействия между собой;
3. определяется перечень воздействующих на систему внешних факторов и их характеристик;
4. выбираются показатели эффективности системы, т.е. такие числовые характеристики системы, которые определяют степень соответствия системы ее назначению;
5. составляется формальная математическая модель системы;
6. составляется машинная математическая модель, пригодная для исследования системы на ЭВМ.

<u>Требования к математической модели:</u>

Требования определяются прежде всего ее назначением, т.е. характером поставленной задачи:

"Хорошая" модель должна быть:

1. целенаправленной;
2. простой и понятной пользователю;
3. достаточной с точки зрения возможностей решения поставленной задачи;
4. удобной в обращении и управлении;
5. надежной в смысле защиты от абсурдных ответов;
6. допускающей постепенные изменения в том смысле, что, будучи вначале простой, она при взаимодействии с пользователями может становиться более сложной.

Математическая модель, в широком смысле, это приближенное описание какого-либо класса явлений внешнего мира, выраженное с помощью математической символики. Применительно к задачам исследования качества системы математическая модель должна обеспечивать адекватное описание влияния параметров и условий функционирования на показатели ее качества. Что касается точности модели, то ее уровень должен обеспечивать достоверное сравнительное оценивание и ранжирование по уровню качества альтернативных вариантов

В основе изучения и моделирования процессов функционирования технических систем всегда лежит эксперимент - реальный или логический. Суть реального эксперимента состоит в непосредственном изучении конкретного физического объекта. В ходе логического эксперимента свойства объекта исследуются не на самом объекте, а с помощью его математической или содержательной (словесной) модели, изоморфной объекту с точки зрения изучаемых эксперименте свойств.

Подавая на вход системы различные входные процессы и измеряя процесс на ее выходе, исследователь получает возможность установить и записать математически существующую между ними связь в виде уравнения, связывающего для каждого интервала времени значения входных и выходных воздействий и потому называемого уравнением «вход-выход». Кроме того, для адекватного отражения связи между входом и выходом системы в системотехнике вводится понятие «состояние». По своему смыслу состояние $z(\tau)$ представляет собой совокупность существенных свойств (характеристик) системы, знание которых в настоящем (в момент времени τ) позволяет определить ее поведение в будущем (в моменты времени $t > \tau$). Благодаря этому понятию, уравнение "вход-выход"-состояние принимает вид:

$$\boldsymbol{Y_T = A(T,\ z(\tau),\ X_T)}, \quad (2.1)$$

где X_T, Y_T – входной и выходной процесс на интервале времени T;

$A(*)$ - оператор выходов.

Согласно (2.1), выходной процесс полностью определяется входным процессом и начальным состоянием и не зависит от того, каким образом система была переведена в это состояние. Отсюда ясно, что уравнение (2.1) ограничивает класс рассматриваемых систем только такими системами, функционирование которых в настоящем не зависит от того, как они функционировали в прошлом.

Для полного описания процесса функционирования системы необходимо задать условия определения состояния системы. Для этого вводится понятие уравнения состояния:

$$\boldsymbol{z(t) = B(\tau t,\ z(\tau),\ X_{\tau t})}, \quad (2.2)$$

где

$B(*)$ - оператор, устанавливающий однозначную зависимость $z(t)$ от пары $(z(\tau),\ X_{\tau t})$, которая задана на интервале $\tau\ t$, и называемый оператором перехода.

Уравнения (2.1) и (2.2) имеют достаточно логичное обобщение и на многомерный случай, когда каждая из компонент уравнений имеет векторный вид:

$$X \to \vec{X}, Y \to \vec{Y}, Z \to \vec{Z}$$

Таким образом, модель функционирования системы должна обеспечивать прогнозирование процесса функционирования на всем интервале функционирования $\boldsymbol{T}$ (множество времени) по заданному вектору начального состояния $\vec{Z}(\tau)$ записанном в векторном виде входному процессу $\vec{X}(T)$.

Согласно изложенному выше, для решения этой задачи достаточно задать множества допустимых значений входных $\boldsymbol{X}$ и выходных $\boldsymbol{Y}$ процессов, а также множество возможных состояний системы $\boldsymbol{Z}$ и операторы выхода $\boldsymbol{A}$ и перехода $\boldsymbol{B}$. Модель функционирования системы без предыстории представляет собой кортеж

$$\boldsymbol{MF = \langle T,\ X,\ Y,\ Z,\ A,\ B\rangle}. \quad (2.3)$$

Если все компоненты в (2.3) известны, модель функционирования полностью определена и может быть использована для описания и изучения свойственных системе процессов функционирования. Множества и операторы, составляющие *общесистемную модель* (2.3), могут обладать различными свойствами, совокупность которых позволяет конкретизировать характер функционирования системы:

N – непрерывность;

L – линейность;

C – стационарность;

P – стохастичность (вероятность).

Наделяя систему теми или иными свойствами общесистемная модель конкретизируется до *системной модели.*

Системные свойства:

1). Если интервал функционирования системы $\boldsymbol{T} = [\varepsilon,\eta]$ представляет отрезок оси действительных чисел, заданный началом ε и концом η, то система функционирует в непрерывном времени. Если, кроме того непрерывны операторы $\boldsymbol{A}$ и $\boldsymbol{B}$, то система наз. непрерывной.

2). С т.зр. реакции на внешнее воздействие объекты подразделяют на линейные и нелинейные. Линейными наз. такой объект, реакция которого на совместное воздействие 2-х любых внешних возмущений равно сумме реакций на каждое из этих воздействий, приложенных к системе порознь.

$F^0(x_1(t)+x_2(t)) = F^0(x_1(t)) + F^0(x_2(t))$ - принцип суперпозиции,

$F^0(0)=0$ (начальное состояние системы),

где F^0 - оператор объекта, устанавливает связь входа и выхода.

Для линейных систем выполняется принцип суперпозиции.

3). Поскольку стационарная система при фиксированном начальном состоянии $Z(t_0)$ одинаково реагирует на эквивалентные, отличающиеся только сдвигом по времени, входные воздействия, то наложение интервала $t_0,\ t$ на оси времени не оказывает влияния на процесс функционирования системы. Модель M для стационарных систем не содержит в явном виде интервал функционирования T.

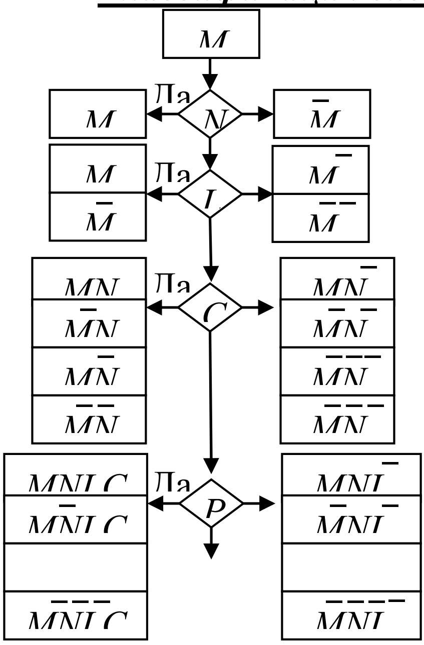
4) Если в модели M операторы A и B каждой паре $(X,\ V,\ Z(t_0))$ (вход, состояние) ставят в соответствие *единственные* значения Y и Z, описываемая этой моделью система называется *детерминированной*. Для стохастической (вероятностной) системы Y и Z, *случайные* величины, заданные законами распределения.

Общесистемная и системные модели функционирования (в дальнейшем термин «модель функционирования» для краткости может заменяться термином «модель» с сохранением исходного смысла) обладают исключительно высокой степенью общности. Они, безусловно, необходимы для теоретических исследований и полезны, так как выявляют общие закономерности, присущие весьма широкому классу систем. Но в повседневной практической деятельности инженеры традиционно используют так называемые *конструктивные модели* - гораздо менее общие, но позволяющие производить конкретные вычисления. ***Конструктивные модели*** в сущности представляют собой алгоритмы, пользуясь которыми, можно определить значения одних переменных, характеризующих данную систему, по заданным или измеренным значениям других переменных. Однако между системными и конструктивными моделями нет противоречия. По мере накопления знаний о системе, уточнения и конкретизации ее свойств и характеристик системная модель естественным образом преобразуется в конструктивную. Следовательно, конструктивная модель может и должна закономерно вырастать из более общей системной модели. Такой - истинно системотехнический подход – представляется более обоснованным, чем априорное задание конструктивной модели исследователем, использующим для этого лишь свою интуицию и субъективные представления о возможностях тех или иных математических схем.

Таким образом, наиболее важные и принципиальные этапы построения модели функционирования системы определяются процессом реализации системотехнической цепочки преобразований **«общесистемная модель → системная модель → конструктивная модель → машинная модель»**.

Моделирование процессов функционирования конкретной системы должно начинаться с записи всех компонент общесистемной модели (2.3), определения их содержательного смысла и областей изменения. Согласно модели (2.3), необходимо определить: интервал времени, на котором нас интересует функционирование системы; множество входных и выходных воздействий и области их возможных изменений; множество характеристик состояния системы и область их возможных изменений.

<u>Классификация системных моделей</u>



$MNLC\bar{P}$ - легко мат.описание

$M\bar{N}\bar{L}\bar{C}P$ - нет адекватного мат.описания (трудно)

Инверсия ($\bar{N}$) *– данное свойство не выполняется, например нет свойства непрерывности*

Общесистемная и системные модели обладая высшей степенью общности устанавливают закономерности, которые присущи всем или достаточно широкому классу систем. В инженерной практике используют так называемые <u>конструктивные модели,</u> пригодные для инженерных расчетов.

<u>*КМ*</u> – алгоритмы, пользуясь которыми можно определить значения одних переменных, характеризующих систему по заданным или измеренным

значениям других переменных.

КМ – может и должна вырастать из большой общей системной модели путем конкретизации ее свойств.

При построении моделей функционирования систем применяют следующие подходы:

1) непрерывно-детерминированный подход (дифференцированные уравнения);
2) дискретно-детерминированный (конечные автоматы);
3) дискретно-стохастический подход (вероятностные автоматы);
4) непрерывно-стохастический подход (системы СМО)
5) обобщенный / универсальный подход (агрегитивные системы)

Основы теории массового обслуживания

Теория массового обслуживания составляет один из разделов теории вероятностей. В этой теории рассматриваются **вероятностные** задачи и математические модели (до этого нами рассматривались детерминированные математические модели). Напомним, что:

Детерминированная математическая модель отражает поведение объекта (системы, процесса) с позиций **полной определенности** в настоящем и будущем.

Вероятностная математическая модель учитывает влияние случайных факторов на поведение объекта (системы, процесса) и, следовательно, оценивает будущее с позиций вероятности тех или иных событий.

Т.е. здесь как, например, в теории игр задачи рассматриваются **в условиях неопределенности**.

Рассмотрим сначала некоторые понятия, которые характеризуют «стохастическую неопределенность», когда неопределенные факторы, входящие в задачу, представляют собой случайные величины (или случайные функции), вероятностные характеристики которых либо известны, либо могут быть получены из опыта. Такую неопределенность называют еще «благоприятной», «доброкачественной».

Понятие случайного процесса

Строго говоря, случайные возмущения присущи любому процессу. Проще привести примеры случайного, чем «неслучайного» процесса. Даже, например, процесс хода часов (вроде бы это строгая выверенная работа – «работает как часы») подвержен случайным изменениям (уход вперед, отставание, остановка). Но до тех пор, пока эти возмущения несущественны, мало влияют на интересующие нас параметры, мы можем ими пренебречь и рассматривать процесс как детерминированный, неслучайный.

Пусть имеется некоторая система S (техническое устройство, группа таких устройств, технологическая система – станок, участок, цех, предприятие, отрасль промышленности и т.д.). В системе S протекает **случайный процесс**, если она с течением времени меняет свое состояние (переходит из одного состояния в другое), причем, заранее неизвестным случайным образом.

Примеры: 1. Система S – технологическая система (участок станков). Станки время от времени выходят из строя и ремонтируются. Процесс, протекающий в этой системе, случаен.

2. Система S – самолет, совершающий рейс на заданной высоте по определенному маршруту. Возмущающие факторы – метеоусловия, ошибки экипажа и т.д., последствия – «болтанка», нарушение графика полетов и т.д.

Марковский случайный процесс

Случайный процесс, протекающий в системе, называется **Марковским**, если для любого момента времени t_0 вероятностные характеристики процесса в будущем зависят только от его состояния в данный момент t_0 и не зависят от того, когда и как система пришла в это состояние.

Пусть в настоящий момент t_0 система находится в определенном состоянии S_0. Мы знаем характеристики состояния системы в настоящем $t_0 \to S_0$ и все, что было при $t < t_0$ (предысторию процесса). Можем ли мы предугадать (предсказать) будущее, т.е. что будет при $t > t_0$? В точности – нет, но какие-то вероятностные характеристики процесса в будущем найти можно. Например, вероятность того, что через некоторое время τ система S окажется в состоянии S_1 или останется в состоянии S_0 и т.д.

Пример. Система S – группа самолетов, участвующих в воздушном бою. Пусть x – количество «красных» самолетов, y – количество «синих» самолетов. К моменту времени t_0 количество сохранившихся (не сбитых) самолетов соответственно – x_0, y_0. Нас интересует вероятность того, что в момент времени $(t_0+\tau)$ численный перевес будет на стороне «красных». Эта вероятность зависит от того, в каком состоянии находилась система в момент времени t_0, а не от того, когда и в какой последовательности погибали сбитые до момента t_0 самолеты.

На практике Марковские процессы в чистом виде обычно не встречаются. Но имеются процессы, для которых влиянием «предистории» можно пренебречь. И при изучении таких процессов можно применять Марковские модели (в теории массового обслуживания рассматриваются и не Марковские системы массового обслуживания, но математический аппарат, их описывающий, гораздо сложнее).

В исследовании операций большое значение имеют Марковские случайные процессы с дискретными состояниями и непрерывным временем.

Процесс называется **процессом с дискретным состоянием**, если его возможные состояния S_1, S_2, … можно заранее определить, и переход системы из состояния в состояние происходит «скачком», практически мгновенно.

Процесс называется **процессом с непрерывным временем**, если моменты возможных переходов из состояния в состояние не фиксированы заранее, а неопределенны, случайны и могут произойти в любой момент.

Далее рассматриваются только процессы с дискретным состоянием и непрерывным временем.

Пример. Технологическая система (участок) S состоит из двух станков, каждый из которых в случайный момент времени может выйти из строя (отказать), после чего мгновенно начинается ремонт узла, тоже продолжающийся заранее неизвестное, случайное время. Возможны следующие состояния системы:

S_0 - оба станка исправны;

S_1 - первый станок ремонтируется, второй исправен;

S_2 - второй станок ремонтируется, первый исправен;

S_3 - оба станка ремонтируются.

Переходы системы S из состояния в состояние происходят практически мгновенно, в случайные моменты выхода из строя того или иного станка или окончания ремонта.

При анализе случайных процессов с дискретными состояниями удобно пользоваться геометрической схемой – **графом состояний**. Вершины графа – состояния системы. Дуги графа – возможные переходы из состояния в

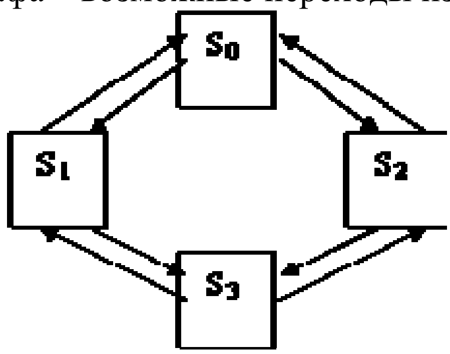


Рис.1. Граф состояний системы

состояние. Для нашего примера граф состояний приведен на рис.1.

Примечание. Переход из состояния S_0 в S_3 на рисунке не обозначен, т.к. предполагается, что станки выходят из строя независимо друг от друга. Вероятностью одновременного выхода из строя обоих станков мы пренебрегаем.

Потоки событий

Поток событий – последовательность однородных событий, следующих одно за другим в какие-то случайные моменты времени.

В предыдущем примере – это поток отказов и поток восстановлений. Другие примеры: поток вызовов на телефонной станции, поток покупателей в магазине и т.д.

Поток событий можно наглядно изобразить рядом точек на оси времени $O\ t$ – рис. 2.

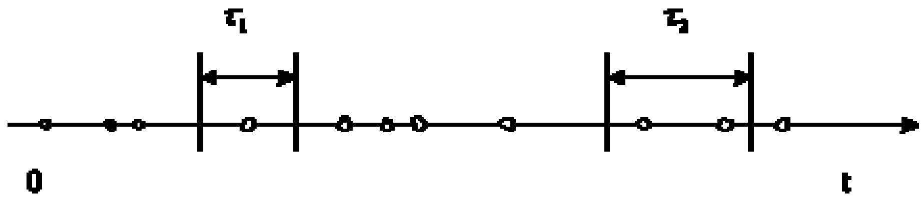


Рис.2. Изображение потока событий на оси времени

Положение каждой точки случайно, и здесь изображена лишь какая-то одна реализация потока.

Интенсивность потока событий (λ) – это среднее число событий, приходящееся на единицу времени.

Рассмотрим некоторые свойства (виды) потоков событий.

Поток событий называется **стационарным**, если его вероятностные характеристики не зависят от времени.

В частности, интенсивность λ стационарного потока постоянна. Поток событий неизбежно имеет сгущения или разрежения, но они не носят закономерного характера, и среднее число событий, приходящееся на единицу времени, постоянно и от времени не зависит.

Поток событий называется **потоком без последствий**, если для любых двух непересекающихся участков времени τ_1 и τ_2 (см. рис.2) число событий, попадающих на один из них, не зависит от того, сколько событий попало на другой. Другими словами, это означает, что события, образующие поток, появляются в те или иные моменты времени **независимо друг от друга** и вызваны каждое своими собственными причинами.

Поток событий называется **ординарным**, если события в нем появляются поодиночке, а не группами по нескольку сразу.

Поток событий называется **простейшим (или стационарным пуассоновским),** если он обладает сразу тремя свойствами: 1) стационарен, 2) ординарен, 3) не имеет последствий.

Простейший поток имеет наиболее простое математическое описание. Он играет среди потоков такую же особую роль, как и закон нормального распределения среди других законов распределения. А именно, при наложении достаточно большого числа независимых, стационарных и ординарных потоков (сравнимых между собой по интенсивности) получается поток, близкий к простейшему.

Для простейшего потока с интенсивностью λ интервал T между соседними событиями имеет так называемое **показательное (экспоненциальное) распределение** с плотностью

$$f(t) = \lambda e^{-\lambda t},$$

где λ - параметр показательного закона.

Для случайной величины T, имеющей показательное распределение, математическое ожидание m_T есть величина, обратная параметру, а среднее квадратичное отклонение σ_T равно математическому ожиданию

$$m_T = \sigma_T = 1/\lambda$$

Уравнения Колмогорова для вероятностей состояний. Финальные вероятности состояний

Рассматривая Марковские процессы с дискретными состояниями и непрерывным временем, подразумевается, что все переходы системы S из состояния в состояние происходят под действием простейших потоков событий (потоков вызовов, потоков отказов, потоков восстановлений и т.д.). Если все потоки событий, переводящие систему S из состояния в состояние простейшие, то процесс, протекающий в системе, будет Марковским.

Итак, на систему, находящуюся в состоянии S_i, действует простейший поток событий. Как только появится первое событие этого потока, происходит «перескок» системы из состояния S_i в состояние S_j (на графе состояний по стрелке $S_i \rightarrow S_j$).

Для наглядности на графе состояний системы у каждой дуги проставляют интенсивности того потока событий, который переводит систему по данной дуге (стрелке). λ_{ij}- интенсивность потока событий, переводящий систему из состояния S_i в S_j. Такой граф называется **размеченным**. Для нашего примера размеченный граф приведен на рис.3.

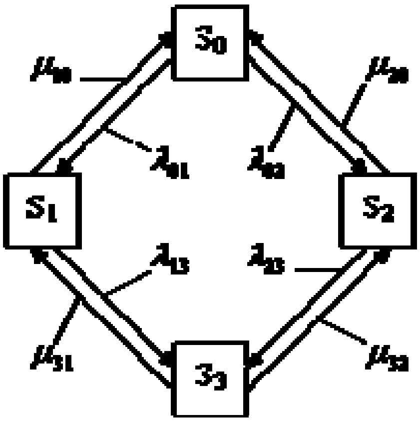


Рис.3. Размеченный граф состояний системы

На этом рисунке λ_{ij}- интенсивности потока отказов; μ_{ij}- интенсивности потока восстановлений.

Предполагаем, что среднее время ремонта станка не зависит от того, ремонтируется ли один станок или оба сразу. Т.е. ремонтом каждого станка занят отдельный специалист.

Пусть система находится в состоянии S_0. В состояние S_1 ее переводит поток отказов первого станка. Его интенсивность равна

$$\lambda_{01} = 1/T_{ср.раб.1}, ед.времени^{-1}.$$

где $T_{ср.раб.1}$- среднее время безотказной работы первого станка.

Из состояния S_1 в S_0 систему переводит поток «окончаний ремонтов» первого станка. Его интенсивность равна

$$\mu_{10} = 1/T_{ср.рем.1}, ед.времени^{-1}.$$

где $T_{ср.рем.1}$- среднее время ремонта первого станка.

Аналогично вычисляются интенсивности потоков событий, переводящих систему по всем дугам графа. Имея в своем распоряжении размеченный граф состояний системы, строится **математическая модель** данного процесса.

Пусть рассматриваемая система S имеет n- возможных состояний $S_1, S_2, ..., S_n$. Вероятность i- го состояния $p_i(t)$ - это вероятность того, что в момент времени t система будет находиться в состоянии S_i. Очевидно, что для любого момента времени сумма всех вероятностей состояний равна единице:

$$\sum_{i=1}^{n} p_i(t) = 1.$$

Для нахождения всех вероятностей состояний $p_i(t)$ как функций времени составляются и решаются **уравнения Колмогорова** – особого вида уравнения, в которых неизвестными функциями являются вероятности состояний. Правило составления этих уравнений приведем здесь без доказательств. Но прежде, чем его приводить, объясним понятие **финальной вероятности состояния**.

Что будет происходить с вероятностями состояний при $t \to \infty$? Будут ли $p_1(t), p_2(t), \ldots$ стремиться к каким-либо пределам? Если эти пределы существуют и не зависят от начального состояния системы, то они называются **финальными вероятностями состояний**.

$$\lim_{t \to \infty} p_i(t) = p_i, i = \overline{1,n},$$

где n - конечное число состояний системы.

Финальные вероятности состояний – это уже не переменные величины (функции времени), а постоянные числа. Очевидно, что

$$\sum_{i=1}^{n} p_i = 1.$$

Финальная вероятность состояния S_i – это по – существу среднее относительное время пребывания системы в этом состоянии.

Например, система S имеет три состояния S_1, S_2 и S_3. Их финальные вероятности равны соответственно 0,2; 0,3 и 0,5. Это значит, что система в предельном стационарном состоянии в среднем 2/10 времени проводит в состоянии S_1, 3/10 – в состоянии S_2 и 5/10 – в состоянии S_3.

Правило составления системы уравнений Колмогорова: в каждом уравнении системы **в левой его части** стоит финальная вероятность данного состояния p_i, умноженная на суммарную интенсивность всех потоков, **ведущих из данного состояния**, а **в правой его части** – сумма произведений интенсивностей всех потоков, **входящих в** i**- е состояние**, на вероятности тех состояний, из которых эти потоки исходят.

Пользуясь этим правилом, напишем систему уравнений **для нашего примера**:

$$\begin{cases} (\lambda_{01} + \lambda_{02})p_0 = \mu_{10}p_1 - \mu_{20}p_2 \\ (\mu_{10} + \lambda_{13})p_1 = \lambda_{01}p_0 + \mu_{31}p_3 \\ (\mu_{20} + \lambda_{23})p_2 = \lambda_{02}p_0 + \mu_{32}p_3 \\ (\mu_{31} + \mu_{32})p_3 = \lambda_{13}p_1 + \lambda_{23}p_2 \end{cases}$$

Эту систему четырех уравнений с четырьмя неизвестными p_0, p_1, p_2, p_3, казалось бы, можно вполне решить. Но эти уравнения однородны (не имеют свободного члена), и, значит, определяют неизвестные только с точностью до произвольного множителя. Однако можно воспользоваться нормировочным условием

$$p_0 + p_1 + p_2 + p_3 = 1$$

и с его помощью решить систему. При этом одно (любое) из уравнений можно отбросить (оно вытекает как следствие из остальных).

Продолжение примера. Пусть значения интенсивностей потоков равны:

$$\lambda_{01} = \lambda_{23} = 1, \lambda_{13} = \lambda_{02} = 2, \mu_{10} = \mu_{32} = 2, \mu_{20} = \mu_{31} = 3.$$

Четвертое уравнение отбрасываем, добавляя вместо него нормировочное условие:

$$\begin{cases} 3p_0 = 2p_1 + 3p_2 \\ 4p_1 = p_0 + 3p_3 \\ 4p_2 = 2p_0 + 2p_3 \\ p_0 - p_1 + p_2 + p_3 = 1 \end{cases}$$

$$p_0 = 6/15 = 0{,}4; p_1 = 3/15 = 0{,}2; p_2 = 4/15 \cong 0{,}27; p_3 = 2/15 \cong 0{,}13.$$

Т.е. в предельном, стационарном режиме система S в среднем 40 % времени будет проводить в состоянии S_0 (оба станка исправны), 20 % - в состоянии S_1 (первый станок ремонтируется, второй работает), 27 % - в состоянии S_2 (второй станок ремонтируется, первый работает), 13% - в состоянии S_3 (оба станка ремонтируются). Знание этих финальных вероятностей может помочь оценить среднюю эффективность работы системы и загрузку ремонтных органов.

Пусть система S в состоянии S_0 (полностью исправна) приносит в единицу времени доход 8 условных единиц, в состоянии S_1 – доход 3 условные единицы, в состоянии S_2 – доход 5 условных единиц, в состоянии S_3 – не приносит дохода. Тогда в предельном, стационарном режиме средний доход в единицу времени будет равен $W = 0{,}4*8 + 0{,}2*3 + 0{,}27*5 = 5{,}15$ условных единиц.

Станок 1 ремонтируется долю времени, равную $p_1 + p_3 = 0{,}2 + 0{,}13 = 0{,}33$. Станок 2 ремонтируется долю времени, равную $p_2 + p_3 = 0{,}27 + 0{,}13 = 0{,}4$. Возникает **задача оптимизации**. Пусть мы можем уменьшить среднее время ремонта первого или второго станка (или обоих), но это нам обойдется в определенную сумму. Спрашивается, окупит ли увеличение дохода, связанное с ускорением ремонта, повышенные расходы на ремонт? Нужно будет решить систему четырех уравнений с четырьмя неизвестными.

Задачи теории массового обслуживания

Примеры систем массового обслуживания (СМО): телефонные станции, ремонтные мастерские, билетные кассы, справочные бюро, станочные и другие технологические системы, системы управления гибких производственных систем и т.д.

Каждая СМО состоит из какого – то количества обслуживающих единиц, которые называются **каналами обслуживания** (это станки, транспортные тележки, роботы, линии связи, кассиры, продавцы и т.д.). Всякая СМО предназначена для обслуживания какого – то **потока заявок** (требований), поступающих в какие – то случайные моменты времени.

Обслуживание заявки продолжается какое – то, вообще говоря, случайное время , после чего канал освобождается и готов к приему следующей заявки. Случайный характер потока заявок и времени обслуживания приводит к тому, что в какие – то периоды времени на входе СМО скапливается излишне большое количество заявок (они либо становятся в очередь, либо покидают СМО необслуженными). В другие же периоды СМО будет работать с недогрузкой или вообще простаивать.

Процесс работы СМО – случайный процесс с дискретными состояниями и непрерывным временем. Состояние СМО меняется скачком в моменты появления каких - то событий (прихода новой заявки, окончания обслуживания, момента, когда заявка, которой надоело ждать, покидает очередь).

Предмет теории массового обслуживания – построение математических моделей, связывающих заданные условия работы СМО (число каналов, их производительность, правила работы, характер потока заявок) с интересующими нас характеристиками – показателями эффективности СМО. Эти показатели описывают способность СМО справляться с потоком заявок. Ими могут быть:

среднее число заявок, обслуживаемых СМО в единицу времени; среднее число занятых каналов; среднее число заявок в очереди; среднее время ожидания обслуживания и т.д.

Математический анализ работы СМО очень облегчается, если процесс этой работы Марковский, т.е. потоки событий, переводящие систему из состояния в состояние – простейшие. Иначе математическое описание процесса очень усложняется и его редко удается довести до конкретных аналитических зависимостей. На практике не Марковские процессы с приближением приводятся к Марковским. Приведенный далее математический аппарат описывает Марковские процессы.

Классификация систем массового обслуживания

Первое деление (по наличию очередей):

1. СМО с отказами;
2. СМО с очередью.

В СМО с отказами заявка, поступившая в момент, когда все каналы заняты, получает отказ, покидает СМО и в дальнейшем не обслуживается.

В СМО с очередью заявка, пришедшая в момент, когда все каналы заняты, не уходит, а становится в очередь и ожидает возможности быть обслуженной.

СМО с очередями подразделяются на разные виды в зависимости от того, как организована очередь – **ограничена или не ограничена**. Ограничения могут касаться как длины очереди, так и времени ожидания, «дисциплины обслуживания».

Итак, например, рассматриваются следующие СМО:

- СМО с нетерпеливыми заявками (длина очереди и время обслуживания ограничено);
- СМО с обслуживанием с приоритетом, т.е. некоторые заявки обслуживаются вне очереди и т.д.

Кроме этого СМО делятся на открытые СМО и замкнутые СМО.

В открытой СМО характеристики потока заявок не зависят от того, в каком состоянии сама СМО (сколько каналов занято). **В замкнутой СМО** – зависят. Например, если один рабочий обслуживает группу станков, время от времени требующих наладки, то интенсивность потока «требований» со стороны станков зависит от того, сколько их уже исправно и ждет наладки.

Классификация СМО далеко не ограничивается приведенными разновидностями, но этого достаточно.

Математические модели простейших систем массового обслуживания

Ниже будут рассмотрены примеры простейших систем массового обслуживания (СМО). Понятие «простейшие» не означает «элементарные». Математические модели этих систем применимы и успешно используются в практических расчетах.

Одноканальная СМО с отказами

Дано: система имеет один канал обслуживания, на который поступает простейший поток заявок с интенсивностью λ. Поток обслуживаний имеет интенсивность μ. Заявка, заставшая систему занятой, сразу же покидает ее.

Найти: абсолютную и относительную пропускную способность СМО и вероятность того, что заявка, пришедшая в момент времени t, получит отказ.

Система при любом $t > 0$ может находиться в двух состояниях: S_0 – канал свободен; S_1 – канал занят. Переход из S_0 в S_1 связан с появлением заявки и немедленным началом ее обслуживания. Переход из S_1 в S_0 осуществляется, как только очередное обслуживание завершится (рис.4).

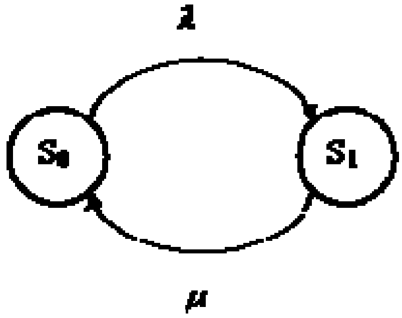


Рис.4. Граф состояний одноканальной СМО с отказами

Выходные характеристики (характеристики эффективности) этой и других СМО будут даваться без выводов и доказательств.

Абсолютная пропускная способность (среднее число заявок, обслуживаемых в единицу времени):

$$A=\frac{\lambda\mu}{\lambda+\mu}, \text{шт / ед.времени},$$

где λ – интенсивность потока заявок (величина, обратная среднему промежутку времени между поступающими заявками - $\overline{t_з}$);

μ – интенсивность потока обслуживаний (величина, обратная среднему времени обслуживания $\overline{t_{об}}$)

Относительная пропускная способность (средняя доля заявок, обслуживаемых системой):

$$Q=\frac{\mu}{\lambda+\mu}.$$

Вероятность отказа (вероятность того, что заявка покинет СМО необслуженной):

$$P_{отк}=\frac{\lambda}{\lambda+\mu}.$$

Очевидны следующие соотношения: $Q=1-P_{отк}$ и $P_{отк}=1-Q$.

Пример. Технологическая система состоит из одного станка. На станок поступают заявки на изготовление деталей в среднем через 0,5 часа $(\overline{t_з}=0,5ч)$. Среднее время изготовления одной детали равно $\overline{t_{об}}=0,6ч$. Если при поступлении заявки на изготовление детали станок занят, то она (деталь) направляется на другой станок. Найти абсолютную и относительную пропускную способности системы и вероятность отказа по изготовлению детали.

Решение.

$$\lambda = 1/\bar{t}_з = 1/0,5 = 2ч^{-1}; \mu = 1/\bar{t}_{об} = 1/0,6 \cong 1,67ч^{-1};$$

$$A = \frac{\lambda\mu}{\lambda+\mu} = \frac{2\cdot 1,67}{2+1,67} = 0,9\ дет/ч; Q = \frac{\mu}{\lambda+\mu} = \frac{1,67}{2+1,67} = 0,455 \cong 0,46.$$

Т.е. в среднем примерно 46 % деталей обрабатываются на этом станке.

$$P_{отк} = \frac{\lambda}{\lambda-\mu} = \frac{2}{2+1,67} \cong 0,54.$$

Т.е. в среднем примерно 54 % деталей направляются на обработку на другие станки.

N – канальная СМО с отказами (задача Эрланга)

Это одна из первых задач теории массового обслуживания. Она возникла из практических нужд телефонии и была решена в начале 20 века датским математиком Эрлангом.

Дано: в системе имеется *n* – каналов, на которые поступает поток заявок с интенсивностью λ. Поток обслуживаний имеет интенсивность μ. Заявка, заставшая систему занятой, сразу же покидает ее.

Найти: абсолютную и относительную пропускную способность СМО; вероятность того, что заявка, пришедшая в момент времени *t*, получит отказ; среднее число заявок, обслуживаемых одновременно (или, другими словам, среднее число занятых каналов).

Решение. Состояние системы *S* (СМО) нумеруется по максимальному числу заявок, находящихся в системе (оно совпадает с числом занятых каналов):

- S_0 – в СМО нет ни одной заявки;
- S_1 – в СМО находится одна заявка (один канал занят, остальные свободны);
- S_2 – в СМО находится две заявки (два канала заняты, остальные свободны);
- . . .
- S_n – в СМО находится *n* – заявок (все *n* – каналов заняты).

Граф состояний СМО представлен на рис. 5

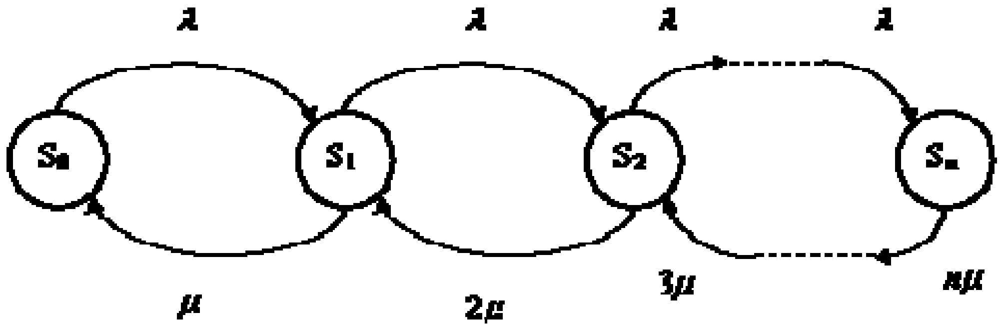


Рис.5 Граф состояний для n – канальной СМО с отказами

Почему граф состояний размечен именно так? Из состояния S_0 в состояние S_1 систему переводит поток заявок с интенсивностью λ (как только приходит заявка, система переходит из S_0 в S_1). Если система находилась в состоянии S_1 и пришла еще одна заявка, то она переходит в состояние S_2 и т.д.

Почему такие интенсивности у нижних стрелок (дуг графа)? Пусть система находится в состоянии S_1 (работает один канал). Он производит μ обслуживаний в единицу времени. Поэтому дуга перехода из состояния S_1 в состояние S_0 нагружена интенсивностью μ. Пусть теперь система находится в состоянии S_2 (работают два канала). Чтобы ей перейти в S_1, нужно, чтобы закончил обслуживание первый канал, либо второй. Суммарная интенсивность их потоков равна 2μ и т.д.

Выходные характеристики (характеристики эффективности) данной СМО определяются следующим образом.

Абсолютная пропускная способность:

$$A = \lambda\left[1-\left(\frac{\lambda}{\mu}\right)^n \frac{p_0}{n!}\right], \text{заяв / ед.времени},$$

где n – количество каналов СМО;

p_0 – вероятность нахождения СМО в начальном состоянии, когда все каналы свободны (финальная вероятность нахождения СМО в состоянии S_0);

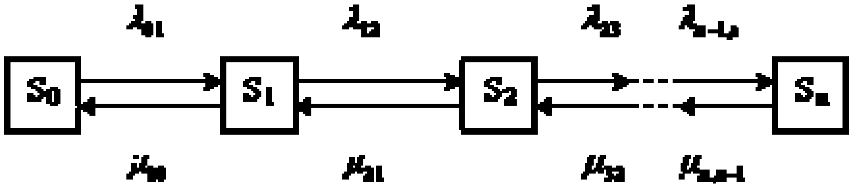


Рис.6. Граф состояний для схемы «гибели и размножения»

Для того, чтобы написать формулу для определения p_0, рассмотрим рис.6

Граф, представленный на этом рисунке, называют еще графом состояний для схемы «гибели и размножения». Напишем сначала для p_0 общую формулу (без доказательства):

$$p_0 = \left(1+\frac{\lambda_{01}}{\mu_{10}}+\frac{\lambda_{12}\lambda_{01}}{\mu_{21}\mu_{10}}+\frac{\lambda_{23}\lambda_{12}\lambda_{01}}{\mu_{32}\mu_{21}\mu_{10}}+\ldots+\frac{\lambda_{n-1,n}\times\ldots\times\lambda_{23}\lambda_{12}\lambda_{01}}{\mu_{n,n-1}\times\ldots\times\mu_{32}\mu_{21}\mu_{10}}\right)^{-1}.$$

Кстати, остальные финальные вероятности состояний СМО запишутся следующим образом.

Вероятность того, что СМО находится в состоянии S_1, когда один канал занят:

$$p_1 = \frac{\lambda_{01}}{\mu_{10}}p_0.$$

Вероятность того, что СМО находится в состоянии S_2, т.е. когда два канала заняты:

$$p_2 = \frac{\lambda_{12}}{\mu_{21}}p_1 = \frac{\lambda_{12}\lambda_{01}}{\mu_{21}\mu_{10}}p_0.$$

Вероятность того, что СМО находится в состоянии S_n, т.е. когда все каналы заняты.

$$p_n = \frac{\lambda_{n-1,n}\times\ldots\times\lambda_{23}\lambda_{12}\lambda_{01}}{\mu_{n,n-1}\times\ldots\times\mu_{32}\mu_{21}\mu_{10}}p_0.$$

Теперь для n – канальной СМО с отказами

$$p_0 = \left(1+\frac{\lambda}{\mu}+\frac{\lambda^2}{1\times2\mu^2}+\frac{\lambda^3}{1\times2\times3\mu^3}+\ldots+\frac{\lambda^n}{n!\mu^n}\right)^{-1}.$$

При этом

$$p_1 = \frac{\lambda}{\mu}p_0;$$

$$p_2 = \frac{\lambda^2}{2\mu^2} p_0;$$

$$p_n = \frac{\lambda^n}{n!\mu^n} p_0.$$

Относительная пропускная способность:

$$Q = 1 - \left(\frac{\lambda}{\mu}\right)^n \frac{p_0}{n!}.$$

Напомним, что это средняя доля заявок, обслуживаемых системой. При этом

$$A = \lambda Q;$$

$$Q = 1 - P_{отк}.$$

Вероятность отказа:

$$P_{отк} = \left(\frac{\lambda}{\mu}\right)^n \frac{p_0}{n!}.$$

Напомним, что это вероятность того, что заявка покинет СМО необслуженной. Очевидно, что $P_{отк} = 1 - Q$.

Среднее число занятых каналов (среднее число заявок, обслуживаемых одновременно):

$$\bar{k} = \frac{\lambda}{\mu}\left[1 - \left(\frac{\lambda}{\mu}\right)^n \frac{p_0}{n!}\right].$$

При этом

$$\bar{k} = \frac{\lambda}{\mu} Q.$$

Пример. Имеется технологическая система (участок), состоящая из трех одинаковых станков. В систему поступают для обработки детали в среднем через 0,5 часа ($\bar{t}_з$). Среднее время изготовления одной детали $\bar{t}_{об} = 0{,}6ч$. Если при поступлении заявки на изготовление детали все станки заняты, то деталь направляется на другой участок таких же станков. Найти финальные вероятности состояний системы и характеристики (показатели эффективности) данной СМО.

$$\lambda = \frac{1}{\bar{t}_з} = \frac{1}{0{,}5} = 2,$$

т.е. в среднем две заявки на обработку деталей в час.

$$\mu = \frac{1}{\bar{t}_{об}} = \frac{1}{0{,}6} \cong 1{,}67.$$

Граф состояний системы представлен на рис.7

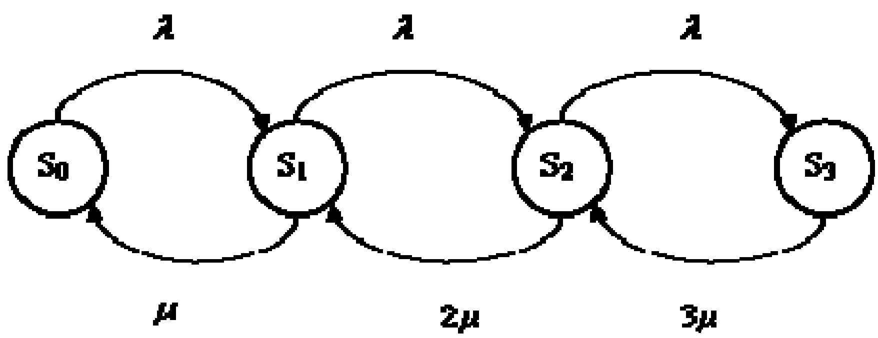


Рис.7Граф состояний для рассматриваемого примера

Возможные состояния системы:

S_0 – в СМО (на участке) нет ни одной заявки;

S_1 – в СМО (на участке) одна заявка;

S_2 – в СМО (на участке) две заявки;

S_3 – в СМО (на участке) три заявки (заняты все три станка).

Вероятность того, что все станки свободны:

$$p_0=\left(1+\frac{\lambda}{\mu}+\frac{\lambda^2}{2\mu^2}+\frac{\lambda^3}{2x3\mu^3}\right)^{-1}=\left(1+\frac{2}{1,67}+\frac{2^2}{2x1,67^2}+\frac{2^3}{2x3x1,67^3}\right)^{-1}=\frac{1}{3,21}\cong 0,31$$

Вероятность того, что один станок занят:

$$p_1=\frac{\lambda}{\mu}p_0=\frac{2}{1,67}0,31\cong 0,37.$$

Вероятность того, что два станка заняты:

$$p_2=\frac{\lambda^2}{2\mu^2}p_0=\frac{2^2}{2\cdot 1,67^2}0,31\cong 0,22$$

Вероятность того, что все три станка заняты:

$$p_3=\frac{\lambda^3}{3!\mu^3}p_0=\frac{2^3}{2\cdot 3\cdot 1,67^3}0,31\cong 0,09.$$

$$A=\lambda\left[1-\left(\frac{\lambda}{\mu}\right)^n\frac{p_0}{n!}\right]=2\left[1-\left(\frac{2}{1,67}\right)^3\frac{0,31}{1\cdot 2\cdot 3}\right]\cong 1,82\text{дет/ч}.$$

$$Q=1-\left(\frac{\lambda}{\mu}\right)^n\frac{p_0}{n!}=1-\left(\frac{2}{1,67}\right)^3\frac{0,31}{1\cdot 2\cdot 3}\cong 0,91;P_{отк}=\left(\frac{\lambda}{\mu}\right)^n\frac{p_0}{n!}=1-Q\cong 0,09.$$

$$\bar{k}=\frac{\lambda}{\mu}\left[1-\left(\frac{\lambda}{\mu}\right)^n\frac{p_0}{n!}\right]=\frac{2}{1,67}0,91\cong 1,09.$$

Т.е. в среднем в этой системе обрабатывается 1,82 дет/ч (примерно 91 % направляемых деталей), при этом примерно 9 % деталей направляется для обработки на другие участки. Одновременно в среднем работает в основном один станок ($\bar{k}=1,09$). Но из–за случайных характеристик потока заявок иногда работают одновременно все три станка ($p_3=0,09$), отсюда 9 % отказов.

Возможные постановки задач оптимизации n – канальных СМО с отказами

1. Определить оптимальное число каналов, обеспечивающее минимум затрат на систему, при условии достижения требуемого уровня ее безотказной работы.

Пример. Пусть $\frac{\lambda}{\mu} = 1, P_{отк} \le 0{,}03 (т.е. \le 3\%)$. Целевая функция (затраты на СМО) запишется: $y = cn \to \min$, где $c - const$. Найти: $n_{опт}$.

Решение:

$$P_{отк} = \left(\frac{\lambda}{\mu}\right)^n \frac{P_0}{n!};$$

$$\frac{\lambda}{\mu} = 1 \Rightarrow P_{отк} = \frac{P_0}{n!}$$

$$P_{отк} < 0{,}03 \Rightarrow \frac{P_0}{n!} < 0{,}03$$

или

$$\frac{n!}{P_0} \ge 33.$$

По другому можно записать:

$$n!\left(1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \ldots + \frac{1}{n!}\right) \ge 33.$$

Последнее равенство начинает выполняться при $n_{опт} = 4$, т.к.

$$n = 1 \to 1\left(1 + \frac{1}{1}\right) = 2 < 33; n = 2 \to 1 \cdot 2\left(1 + \frac{1}{1} + \frac{1}{1 \cdot 2}\right) = 5 < 33;$$

$$n = 3 \to 1 \cdot 2 \cdot 3\left(1 + \frac{1}{1} + \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3}\right) = 16 < 33;$$

$$n = 4 \to 1 \cdot 2 \cdot 3 \cdot 4\left(1 + \frac{1}{1} + \frac{1}{1 \cdot 2} + \frac{1}{1 \cdot 2 \cdot 3} + \frac{1}{1 \cdot 2 \cdot 3 \cdot 4}\right) = 65 > 33.$$

2. Определить оптимальное число каналов, обеспечивающее максимум прибыли от эксплуатации СМО в единицу времени.

Содержание каждого канала в единицу времени обходится в какую–то сумму. Чем больше каналов, тем больше затраты на эксплуатацию СМО. Вместе с тем, чем больше каналов (при λ и $\mu - const$), тем больше доля обслуживаемых заявок. А каждая обслуженная заявка дает определенный (пусть постоянный) доход в единицу времени. При увеличении числа каналов растут доходы *D*, но растут и расходы на эксплуатацию СМО – *R*. Чтобы решить эту задачу, необходимо найти оптимальное число каналов $n_{опт}$, обеспечивающее максимум целевой функции $P = D - R \to \max$, т.е. нужно максимизировать прибыль в единицу времени.

Сетевые модели (N-схемы). Сети Петри

1. Теоретические основы сетей Петри: принципы построения, алгоритмы поведения.

Сети Петри были разработаны и используются для моделирования систем, которые содержат взаимодействующие параллельные компоненты, например аппаратное и программное обеспечение ЭВМ, гибкие производственные системы, а также социальные и биологические системы. Впервые сети Петри предложил Карл Адам Петри в своей докторской диссертации "Связь автоматов" в 1962 году. Работа Петри привлекла внимание группы исследователей, работавших под руководством Дж. Денниса над проектом MAC в Массачусетском Технологическом институте. Эта группа стала источником значительных исследований и публикаций по сетям Петри. Полная оценка и понимание современной теории сетей Петри требуют хорошей подготовки в области математики, формальных языков и автоматов. Современный инженер - системотехник должен иметь квалификацию, необходимую для проведения исследований с помощью сетей Петри.

1.1 Введение в теорию комплектов.

Сети Петри - инструмент исследования систем. Сети Петри делают возможным моделирование системы математическим представлением ее в виде сети Петри. Математическим аппаратом сетей Петри является теория комплектов. Теория комплектов представляет собой естественное расширение теории множеств. Как и множество, комплект является набором элементов из некоторой области. Однако в отличие от множества комплекты допускают наличие нескольких экземпляров одного и того же элемента. В отличие от множества, где элемент либо является элементом множества, либо нет, в комплект элемент может входить заданное число раз. Пусть область представляет собой {a,b,c,d}, тогда комплекты над этой областью будут иметь вид:

B1={a,b,c} B2={a} B3={a,b,c,c}
B4={a,a,a} B5={b,c,b,c} B6={c,c,b,b}
B7={a,a,a,a,a,a,b,b,b,b,b,c,d,d,d,d,d}

Основным понятием теории комплектов является функция числа экземпляров. Обозначение #(x,B) число x в B т.е. число экземпляров элемента x в B. Если ограничить число элементов в комплекте так, что $0 <= \#(x,B) <= 1$, то получим теорию множеств.

Элемент x является членом комплекта B, если $\#(x,B) > 0$. Аналогично, если $\#(x,B) = 0$ то x не принадлежит B.

Определим пустой комплект 0, не имеющий членов (для всех x : $\#(x,0) = 0$). Под мощностью |B| комплекта B понимается общее число экземпляров в комплекте $|B| = \Sigma_x \#(x,B)$.

Комплект A является подкомплектом комплекта B (обозначается $A \subseteq B$), если каждый элемент A является элементом B по крайней мере не больше число раз, т.е. $A \subseteq B$ тогда и только тогда, когда $\#(x,A) <= \#(x,B)$ для всех x.

Два комплекта равны (A = B), если $\#(x,A) = \#(x,B)$.

Комплект A строго включен в комплект B ($A \subseteq B$), если $A \subseteq B$ и A не равно B. Над комплектами определены 4 операции. Операции для двух комплектов A и

В:

1 объединение $A \cup B$: $\#(x, A \cup B) = \max(\#(x,A), \#(x,B))$;
2 пересечение $A \cap B$: $\#(x, A \cap B) = \min(\#(x,A), \#(x,B))$;
3 сумма $A + B$: $\#(x, A + B) = \#(x,A) + \#(x,B)$;
4 разность $A - B$: $\#(x, A - B) = \#(x,A) - \#(x,B)$;

Назовем множество элементов, из которых составляются комплекты, областью D. Пространство комплектов D^n есть множество всех таких комплектов, что элементы их принадлежат D и ни один из элементов не входит в комплект более n раз. Иначе говоря, для любого $B \in D^n$:

а) из $x \in B$ следует $x \in D$;
б) для любого x $\#(x,B) <= n$.

Множество D^∞ есть множество всех комплектов над областью D, без какого либо ограничения на число экземпляров элемента в комплекте.

1.2 Структура сети Петри.

Сеть Петри состоит из 4 компонентов, которые и определяют ее структуру:

- множество позиций P,
- множество переходов T,
- входная функция I,
- выходная функция O.

Входная и выходная функции связаны с переходами и позициями. Входная функция I отображает переход t_j в множество позиций $I(t_j)$, называемых входными позициями перехода. Выходная функция O отображает переход t_j в множество позиций $O(t_j)$, называемых выходными позициями перехода. Т.е.

$$(I : T \to P^\infty)$$
$$(O : T \to P^\infty).$$

Определение 1. Сеть Петри С является четверкой С = (P,T,I,O) где

$P=\{p1,p2,...,p_n\}$ конечное множество позиций, n>=0.
$T=\{t1,t2,...,t_m\}$ конечное множество переходов, m>=0.

Множества позиций и переходов не пересекаются.

$I : T \to P^\infty$ является входной функцией - отображением из переходов в комплекты позиций.

$O : T \to P^\infty$ выходная функция - отображение из переходов в комплекты позиций.

Мощность множества P есть число n, а мощность множества T есть число m. Произвольный элемент P обозначается символом p_i, i=1...n; а произвольный элемент T - символом t_j, j=1...m.

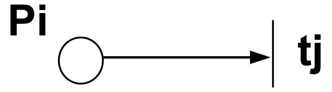


рис. 1

Позиция p_i является входной позицией перехода t_j, в том случае, если $p_i \in I(t_j)$; p_i является выходной позицией перехода, если $pi \in O(t_j)$.

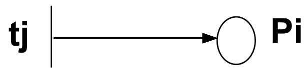


рис. 2

Входы и выходы переходов представляют комплекты позиций. Кратность входной позиции для перехода t_j есть число появлений позиции во входном комплекте перехода $\#(p_i,I(t_j))$. Аналогично, кратность выходной позиции p_i для перехода t_j есть число появлений позиции в выходном комплекте перехода $\#(p_i,O(t_j))$.

Определим, что переход t_j является входом позиции p_i, если p_i есть выход t_j (рис. 2). Переход t_j есть выход позиции p_i, если p_i есть вход t_j (рис. 1).

Определение 2. Определим расширенную входную функцию I и выходную функцию O таким образом, что $\#(t_j,I(p_i)) = \#(p_i,O(t_j))$; $\#(t_j,O(p_i)) = \#(p_i,I(t_j))$;

1.3 Графы сетей Петри.

Для иллюстрации понятий теории сетей Петри гораздо более удобно графическое представление сети Петри. Теоретико - графовым представлением сети Петри является двудольный ориентированный мультиграф. В соответствии с этим граф сети Петри обладает двумя типами узлов:

кружок O является позицией,

планка | является переходом.

Ориентированные дуги соединяют позиции и переходы. Дуга направленная от позиции p_i к переходу t_j определяет позицию, которая является входом перехода t_j. Кратные входы в переход указываются кратными дугами из входных позиций в переход. Выданая позиция указывается дугой от перехода к позиции. Кратные входы также представлены кратными дугами.

Определение 3. Граф G сети Петри есть двудольный ориентированный мультиграф G=(V,A) где

$V = \{v1,V2,...,v_s\}$ - множество вершин

$A = \{a1,a2,...,a_r\}$ - комплект направленных дуг,

$a_i=\{v_j,v_k\}$ где $v_j,v_k \in V$.

Множество V может быть разбито на 2 непересекающихся подмножества P и T, таких что $P \cap T = 0$, и если $a_i = (v_j,v_k)$, тогда либо $v_j \in P$ и $v_k \in T$, либо $v_j \in T$ и $v_k \in P$.

Сеть Петри есть мультиграф, т.к. он допускает существование кратных дуг от одной вершины к другой. Т.к. дуги направлены, то это ориентированный мультиграф. Граф является двудольным, т.к. он допускает существование вершин двух типов: позиций и переходов.

1.4 Пример. Представление сети Петри в виде графа и в виде структуры сети Петри.

Пусть задана следующая структура сети Петри: C = (P,T,I,O), n=5, m=4

P = {p1,p2,p3,p4,p5}   T = {t1,t2,t3,t4}
I(t1)={p1}   O(t1)={p2,p3,p5}
I(t2)={p2,p3,p5}   O(t2)={p5}
I(t3)={p3}   O(t3)={p4}
I(t4)={p4}   O(t4)={p2,p3}

Для сети, изображенной на рис. 3 расширенными входной и выходной функциями являются:

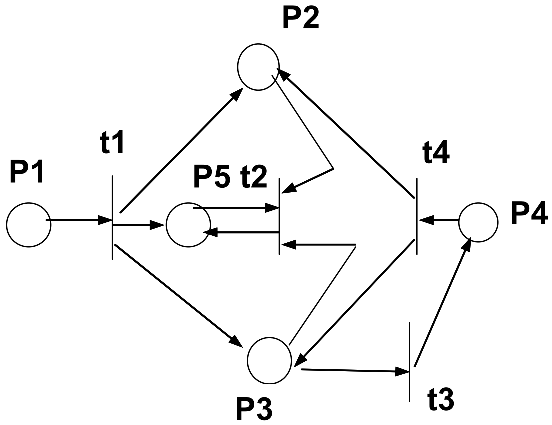


рис. 3

I(p1)={}   O(p1)={t1}
I(p2)={t1,t4}   O(p2)={t2}
I(p3)={t1,t4}   O(p3)={t2,t3}
I(p4)={t3}   O(p4)={t4}
I(p5)={t1,t2}   O(p5)={t2}

Пример 2. Пусть задана следующая структура сети Петри: C = (P,T,I,O)
P={p1,p2,p3,p4,p5,p6}   T={t1,t2,t3,t4,t5} n=6, m=5.
I(t1)={p1}   O(t1)={p2,p3}
I(t2)={p3}   O(t2)={p3,p5,p5}
I(t3)={p2,p3}   O(t3)={p2,p4}
I(t4)={p4,p5,p5,p5}   O(t4)={p4}
I(t5)={p2}   O(t5)={p6}

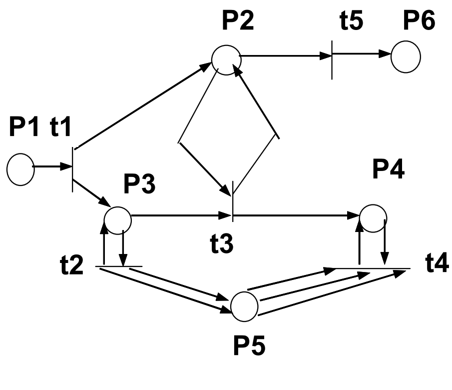


рис. 4

Заметим, что оба представления сети Петри - в виде структуры и в виде графа - эквивалентны. Их можно преобразовать друг в друга.

1.4 Маркировка сетей Петри.

Маркировка μ есть присвоение фишек позициям сети Петри. Фишка - это одна из компонент сети Петри (подобно позициям и переходам). Фишки присваиваются позициям. Их количество при выполнении сети может изменяться. Фишки используются для отображения динамики системы.

Маркированная сеть Петри есть совокупность структуры сети Петри С = (P,T,I,O) и маркировки μ и может быть записана в виде М = (P,T,I,O, μ). На графе сети Петри фишки изображаются крупными точками в кружке, который представляет позицию сети Петри. Количество фишек (точек) для каждой позиции не ограничено и, следовательно, в целом для сети существует бесконечно много маркировок. Множество всех маркировок сети, имеющей n позиций, является множеством всех n векторов, т.е. N^n. Очевидно, что хотя это множество и бесконечно, но оно счетно. Когда маркировка превышает 4 или 5 фишек, то в кружках удобнее не рисовать фишки, а указывать их количество как на рис. 3.7.

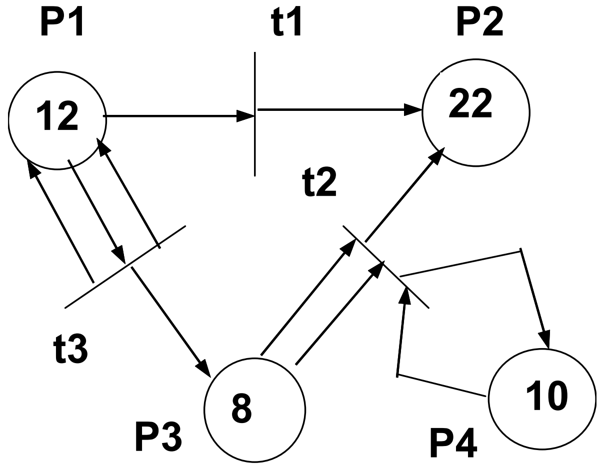


рис. 5

Маркировка μ=(12,22,8,10) - как вектор. Может оказаться, что структура остается неизменной, а маркировка иная, например вектор маркировки будет иметь вид μ = (13,22,9,10)

1.5 Правила выполнения сетей Петри.

Выполнением сети Петри управляют количество и распределение фишек в сети. Сеть Петри выполняется посредством запусков переходов. Переход запускается удалением фишек из его входных позиций и образованием новых фишек, помещаемых в его выходные позиции.

Переход запускается, если он разрешен. Переход называется разрешенным, если каждая из его входных позиций имеет число фишек по крайней мере равное числу дуг из позиции в переход. Фишки во входной позиции, которые разрешают переход, называются его разрешающими фишками. Например, если позиции p1 и p2 служат входами для перехода t1, тогда t1 разрешен, если p1 и p2 имеют хотя бы по одной фишке. Для перехода t3 с входным комплектом {p3,p3,p3} позиция p3 должна иметь не менее 3 фишек для разрешения перехода t3 (рис. 6).

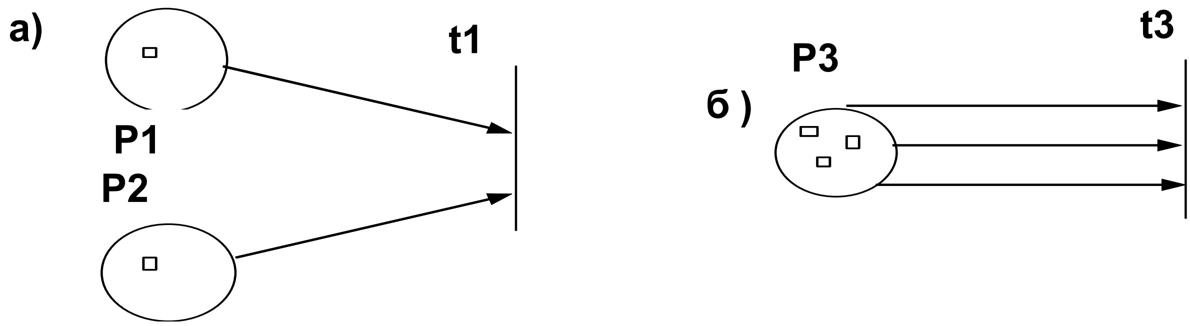


рис. 6

Определение 3.9. Переход $t_j, \in T$ маркированной сети Петри C = (P,T,I,O,μ) с маркировкой μ, разрешен, если для всех $p_i, \in P$, $\mu(p_i)>=\#(p_i,I(t_j))$.
Переход запускается удалением разрешающих фишек, из всех его выходных позиций (количество удаленных фишек для каждой позиции соответствует числу дуг, идущих из этой позиции в переход), с последующим помещением фишек в каждую из его выходных позиций (количество помещаемых фишек в позицию соответствует количеству дуг входящих в данную позицию из перехода).
Переход t3 I(t3) = {p2} и O(t3) = {p3,p4} разрешен каждый раз, когда в p2 будет хотя бы одна фишка. Переход t3 запускается удалением одной фишки из позиции p2 и помещением одной фишки в позицию p3 и p4 (его выходы). Переход t4, в котором I(t4) = {p4,p5} и O(t4) = {p5,p6,p6} запускается удалением по одной фишке из позиций p4 и p5, при этом одна фишка помещается в p5 и две в p6 (рис. 7).

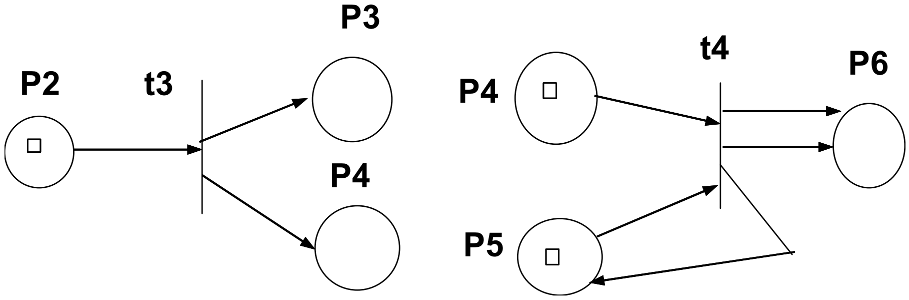


рис. 7

.

Определение 3.10. Переход t_j в маркированной сети Петри с маркировкой μ может быть запущен всякий раз, когда он разрешен. В результате запуска разрешенного перехода t_j образуется новая маркировка μ':

$$\mu'(p_i) = \mu\ (p_j)-\#(p_i,I(t_j)+\#(p_i,O(t_j))$$

.

2. Сети Петри для моделирования систем: способы реализации.

2.1 События и условия.

Представление системы сетью Петри базируется на двух понятиях: событиях и условиях. Под событием понимается действие, имеющее место в системе. Появление события определяет состояние системы, которое может быть описано множеством условий. Условие - это предикат или логическое описание состояния системы. При этом условие может принимать либо значение "истина", либо значение "ложь".

Для того, чтобы событие произошло, необходимо выполнение соответствующих условий, которые называются предусловиями события. Возникновение события может привести к появлению постусловий.

В сети Петри условия моделируются позициями, события - переходами. При этом входы перехода являются предусловиями соответствующего события, выходы - постусловиями. Возникновение события равносильно запуску соответствующего перехода. Выполнение условия представляется фишкой (маркером) в позиции, соответствующей этому условию. Запуск перехода удаляет разрешающие маркеры, представляющие выполнение предусловий и образует новые маркеры, которые представляют выполнение постусловий.

Построение моделей систем в виде сетей Петри связано со следующими обстоятельствами:

1. Моделируемые процессы (явления) совершаются в системе, описываемой множеством событий и условий, которые эти события определяют, а также причинно - следственными отношениями, устанавливаемыми на множестве "события - условия".

2. Определяются события - действия, последовательность наступления которых управляется состоянием системы. Состояния системы задаются множеством условий. Условия формулируются в виде предикатов. Количественные условия характеризуются емкостью. Емкость условий выражается числами натурального ряда.

3. Условия (предикаты) могут быть выполнены или не выполнены. Только выполнение условий обеспечивает возможность наступления событий (предусловия).

4. После наступления события обеспечивается выполнение других условий, находящихся с предусловиями в причинно - следственной связи (постусловия). После того, как событие имело место, реализуются постусловия, которые в свою очередь являются предусловиями следующего события и т.д.

В качестве примера рассмотрим задачу моделирования работы автомата по производству какого либо изделия. Автомат находится в состоянии ожидания до появления заготовки, которую он обрабатывает и посылает в накопитель, т.е. событиями для такой системы являются:

1. заготовка поступила;
2. автомат начинает обработку;
3. автомат заканчивает обработку;
4. деталь посылается в накопитель.

Условиями для системы являются:

1. автомат ждет;
2. заготовка загружена;
3. автомат выполняет обработку;
4. деталь обработана.

В сети Петри условия моделируются позициями, а события - переходами. При этом входы перехода являются предусловиями соответствующего события, а выходы - постусловиями. Выполнение условия представляется фишкой (маркером) в позиции, соответствующей этому условию. Запуск перехода удаляет разрешающие фишки, представляющие выполнение предусловий и образуют новые маркеры, которые представляют выполнение постусловий.

Сеть Петри рассматриваемого автомата имеет вид (рис.8):

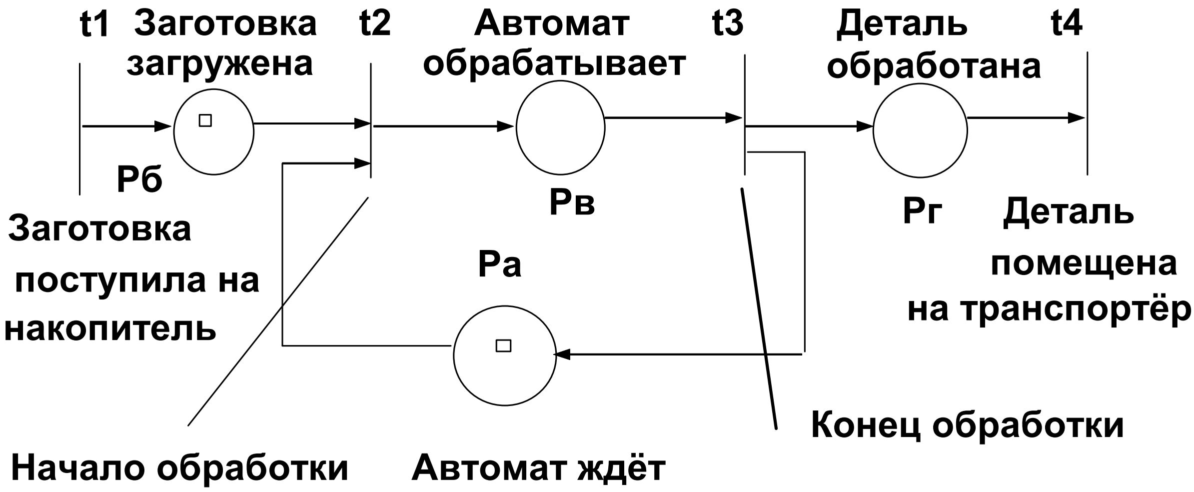


рис.8

Аналогичный пример можно привести для вычислительной системы, которая обрабатывает задания, поступающие с устройства ввода и выводит результаты на устройство вывода. Задание поступает на устройство ввода. Когда процессор свободен и в устройстве ввода есть задание, процессор начинает обработку задания. Когда задание выполнено, оно посылается на устройство вывода; процессор либо продолжает обрабатывать другое задание, если оно есть, либо ждет прихода задания. Эта система может быть промоделирована сетью Петри, изображенной на рис.9

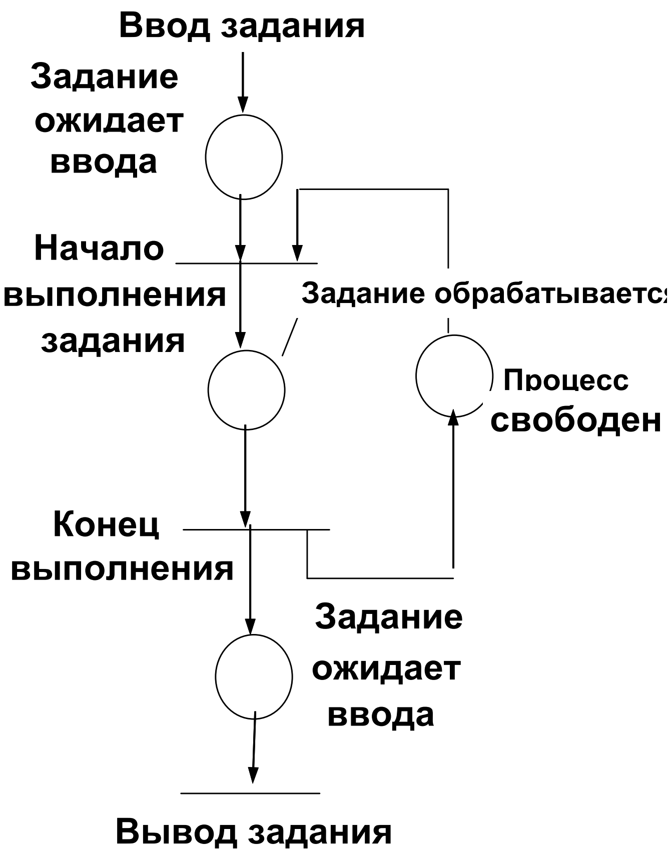


Рис. 9

2.2 Одновременность и конфликт.

Одной из особенностей сетей Петри и их моделей является параллелизм или одновременность. В модели сети Петри два разрешенных взаимодействующих события могут происходить независимо друг от друга, но при необходимости их легко синхронизировать. Т.о. сети Петри представляются идеальными для моделирования систем с распределенным управлением, в которых несколько процессов выполняются одновременно.

Другая важная особенность сетей Петри - их асинхронная природа. В сети Петри отсутствует измерение времени или течение времени. Структура сетей такова, что содержит в себе информацию для определения возможных последовательностей событий. В этих моделях нет никакой информации, связанной с количеством времени, необходимым для выполнения событий.

Выполнение сети Петри рассматривается как последовательность дискретных событий. Обычно запуск перехода рассматривается как мгновенное событие, занимающее нулевое время и одновременное возникновение двух событий невозможно. Моделируемое таким образом событие называется примитивным, примитивные события мгновенны и неодновременны.

Непримитивными называются события, длительность которых отлична от нуля. Однако это не приводит к возникновению проблем при моделировании систем. Непримитивное событие может быть представлено в виде двух примитивных: "начало непримитивного события", "конец непримитивного события" и условия когда «непримитивное» событие происходит".

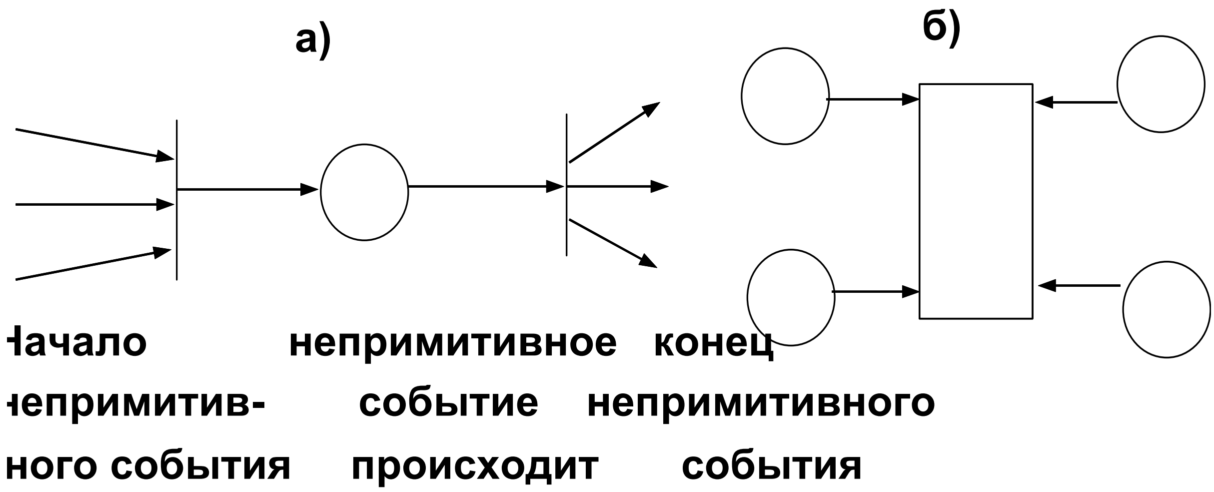


рис.10

В сетях Петри предложено представлять непримитивные события в виде прямоугольника (рис.10), а примитивные события планками. Прямоугольник может иметь существенное значение при моделировании сложных систем на нескольких иерархических уровнях, т.к. он позволяет выделить в отдельный элемент сети целые подсети. Наличие прямоугольника в некотором смысле подобно понятию подпрограммы в блочном программировании и может оказаться в некоторых приложениях весьма полезным.

Если в какой либо момент времени разрешено более одного перехода, то любой из них может стать “следующим”. Выбор запускаемого перехода осуществляется недетерминированным образом, то есть случайно и зависит от воли моделирующего систему. Недетерминорованность и неодновременность запусков переходов в моделировании паралельной системы показывается двумя

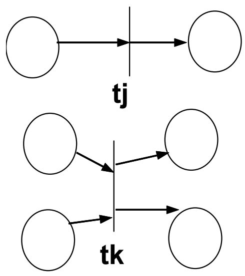


рис 11

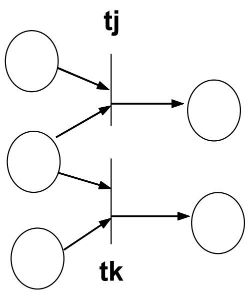


рис 12

способами. Одна из них представлена на рисунке 11. В этой ситуации два разрешённых перехода t_j и t_k не влияют друг на друга. В число возможных последовательностей событий входит последовательность, в которой первым срабатывает один переход и последовательность в которой первым срабатывает другой переход. Эти два перехода могут быть запущены в любом порядке, это называется недетерминированностью и неодновременностью, переход t_k (рис 12) может быть запущен в любом порядке, но обязательно при помощи маркеров в обеих позициях. Это называется одновременностью. Другая ситуация, в которой одновременное выполнение затруднено и которая характеризуется невозможностью одновременного запуска показана на рисунке 10. Здесь переходы t_j и t_k находятся в конфликте, так как запуск одного из них удаляет маркёр из p_i и тем самым завершает другой переход. Эта ситуация называется конфликтом и в моделируемых системах отображает борьбу за общие ресурсы.

Существуют определённые области, в которых сети Петри являются идеальным инструментом для моделирования: это области, в которых события происходят синхронно и независимо. Одной из таких областей является использование сетей Петри для моделирования аппаратного и програмного обеспечения ЭВМ и других систем.

ОБОЩЕННЫЕ МОДЕЛИ (А-СХЕМЫ)

Обобщенный подход базируется на понятии *агрегативной системы* (от англ, aggregate system), представляющей собой формальную схему общего вида, которую будем называть *А-схемой.* Этот подход позволяет описывать поведение непрерывных и дискретных, детерминированных и стохастических систем

Комплексное решение проблем, возникающих в процессе создания и машинной реализации модели, возможно лишь в случае, если моделирующие системы имеют в своей основе единую формальную математическую схему, т. е. *А-схему. А-схема* должна выполнять несколько функций:

- являться адекватным математическим описанием объекта моделирования;
- позволять в упрощенном варианте (для частных случаев) проводить аналитические исследования.

Представленные требования несколько противоречивы, но в рамках обобщенного подхода на основе *А-схем* удается найти между ними компромисс.

При агрегативном подходе первоначально дается формальное определение объекта моделирования — агрегативной системы. При агрегативном описании сложный объект (система) разбивается на конечное число частей (подсистем), сохраняя при этом связи, обеспечивающие их взаимодействие. В случае сложной организации полученных подсистем, подсистемы декомпозируются до уровней в которых они могут быть удобно математически описаны. В результате сложная система представляется в виде многоуровневой конструкции из взаимосвязанных элементов, объединенных в подсистемы различных уровней.

Элементом *А-схемы* является агрегат. Связь между агрегатами (внутри системы S и с внешней средой E) осуществляется с помощью оператора сопряжения R. Агрегат может рассматриваться как *А-схема,* т. е. может разбиваться на элементы (агрегаты) следующего уровня.

Характеристиками агрегата являются множества моментов времени T, входных X и выходных Y сигналов, состояний Z в каждый момент времени t.

Пусть переход агрегата из состояния $z(t_1)$ в состояние $z(t_2) \neq z(t_1)$ происходит за малый интервал времени δz. Переходы из состояния $z(t_1)$ в $z(t_2)$ определяются внутренними параметрами агрегата $h(t) \in H$ входными сигналами $x(t) \in X$.

В начальный момент времени t_0 состояния z имеют значения, равные z°, т. е. z°=z(t_0), которые задаются законом распределения $L\ [z(t_0)]$. Пусть изменение состояния агрегата при входном сигнале x_n описывается случайным оператором V. Тогда для момента времени $t_n \in T$ при поступлении входного сигнала x_n состояние определяется (1)

$$z(t_n+0) = V\,[t_n,\ z(t_n),\ x_n]. \quad (1)$$

Если на интервале времени $(t_n,\ t_n+i)$ нет поступления сигналов, то для $t \in (t_n, t_{n+1})$

состояние агрегата определяется случайным оператором U, можно записать (2)

$$z(t)=U\,[t,\ t_n,\ z(t_n+0)].\ (2)$$

Так как на оператор U не накладываются ни какие ограничения, то допустимы скачки состояний δz в моменты времени, не являющимися моментами поступления входных сигналов x.

Моменты скачков δz называются особыми моментами времени t_s, состояния $z(t_s)$ — особыми состояниями *А-схемы.* Для описания скачков состояний δz в особые моменты времени t_s используется случайный оператор W, который представляет собой частный случай оператора U (3).

$$z(t_\delta+0)=W\,[t_\delta,\ z(t_\delta)].\ (3)$$

На множестве состояний Z выделяется такое подмножество $Z^{(Y)}$, что если $z(t_\delta)$ достигает $Z^{(Y)}$, то это состояние является моментом выдачи выходного сигнала. Выходной сигнал можно описать оператором выходов (4)

$$y=G\,[t_\delta,\ z(t_\delta)].\ (4)$$

Агрегатом будем понимать любой объект, который описывается следующим образом (5).

$$A_n = < T, X, Y, Z, Z^{(Y)}, H, V, U, W, G > \ (5)$$

<u>Структура агрегативной системы</u>

Рассмотрим *А-схему,* структура которой приведена на рис.1.

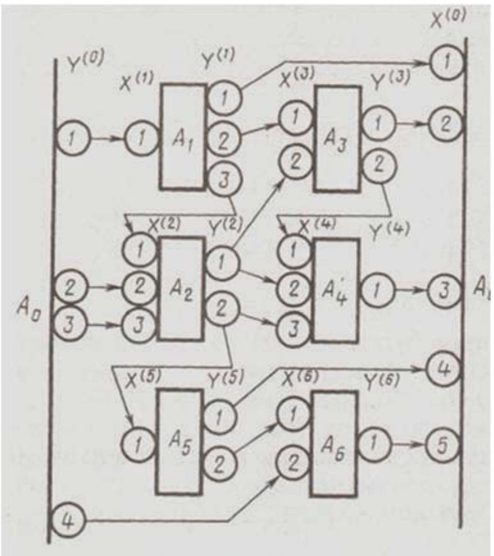


Рис. 2.11. Структура агрегативной системы

Рис. 1

Функционирование *А-схемы* связано с переработкой информации, передача последней на схеме показана стрелками. Вся информация, циркулирующая в *А-схеме,* делится на внешнюю и внутреннюю. Внешняя информация поступает от внешних объектов, внутренняя информация вырабатывается агрегатами самой *А-схемы.* Обмен информацией между *А-схемой* и внешней средой *Е* происходит через агрегаты, называющиеся *полюсами А-схемы. Р*азличают входные полюсы на которые поступают *х-сообщения* (агрегаты $A_t A_2, A_6$), и выходные полюсы *А-схемы,* выходная информация которых является *у- сообщениями* (агрегаты $A_1 A_3, A_4, A_5, A_6$). Агрегаты, не являющиеся полюсами, называются *внутренними.*

Каждому агрегату *А-схемы* A_n подводятся входные контакты (I_n) с элементарными входными сигналами $x_i(t)$, $i = 1..I_n$, и выходные контакты (J_n)с сигналами $y_j(t)$, $j = 1...J_n$.

Введем ряд предположений:

1) взаимодействие между *А-схемой* и внешней средой *Е,* а также между отдельными агрегатами внутри системы *S* осуществляется при передаче сигналов;

2) для описания сигнала достаточно некоторого конечного набора характеристик;

3) элементарные сигналы мгновенно передаются в *А-схеме* независимо друг от друга по элементарным каналам;

4) к входному контакту любого элемента *А-схемы* подключается не более чем один элементарный канал, к выходному контакту — любое конечное число элементарных каналов при условии, что ко входу одного и того же элемента А-схемы направляется не более чем один из упомянутых элементарных каналов.

Взаимодействие *А-схемы* с внешней средой *Е* рассматривается как обмен сигналами между внешней средой *Е* и элементами *А-схемы*, поэтому внешняя среда является фиктивным элементом системы A_0, вход которого содержит I_0 входных контактов и выход — J_0 выходных контактов. Можем записать контакты (6):

$$X_i^{(0)} : i = \overline{1..I_0}, Y_j^{(0)} : \overline{j = 1..J_0} \quad (6)$$

Каждый агрегат, в т.ч. A_n можно охарактеризовать множеством входных контактов $X_1^{(n)}, X_2^{(n)} ..., X_{In}^{(n)} = \{X_i^{(n)}\}$, и множеством выходных контактов $Y_1^{(n)}, Y_2^{(n)} ..., Y_J^{(n)} = \{Y_j^{(n)}\}$,где n=0, N_A.

Пара множеств $\{X_i^{(n)}\}$, $\{Y_j^{(n)}\}$ представляют математическую модель агрегата, которая описывает сопряжения его с прочими элементами *А-схемы* и внешней средой *Е.*

В силу предположения о независимости передачи сигналов каждому

входному контакту $X_l^{(n)} \in \bigcup\limits_{n=0}^{N_A} \{X_l^{(n)}\}$

соответствует не более чем один выходной контакт

$$Y_l^* \in \bigcup_{n=0}^{N_A} \{Y_j^{(n)}\},$$

Введем оператор сопряжения *R*: оператор $Y^*=R(X_i^{(n)})$ с областью определения в множестве $\{X_i^{(n)}\}$ и областью значений $\{Y_j^{(n)}\}$, сопоставляющий входному контакту X_i^n выходной контакт Y_j^n\ связанный с ним элементарным каналом.

Совокупность множеств $\{X_i^{(n)}\}$, $\{Y_j^{(n)}\}$ и оператор *R* представляют схему сопряжения элементов в А-схему. Это есть одноуровневая система сопряжения.

Оператор сопряжения *R* можно задать в виде таблицы, в которой на пересечении строк с номерами элементов (агрегатов) *n* и столбцов с номерами контактов *i* располагаются пары чисел *k, l*, указывающие номер элемента *k* и номер контакта *l*, с которым соединен контакт $X_i^{(n)}$. (таб.1)

n	*i*				
	1	2	3	4	5
0	1.1	3.1	4.1	5.1	6.1
1	0.1				
2	1.3	0.2	0.3		
3	1.2	2.1			
4	3.2	2.1	2.2		
5	2.2				
6	5.2	0.4			

Если столбцы и строки такой таблицы пронумеровать парами n,i и k,l соответственно и на пересечении помещать 1 для контактов n,i и k,l, соединенных элементарным каналом и 0 в противном случае, то получим матрицу смежности ориентированного графа, вершинами которого являются контакты агрегатов, а дугами — элементарные каналы *А-схемы.*

В более сложных случаях могут быть использованы многоуровневые иерархические схемы сопряжения. Схема сопряжения агрегата, определяемая оператором *R,* может быть использована для описания весьма широкого класса объектов.

Упорядоченную совокупность конечного числа агрегатов *An*, $n = N_A$ и оператора *R* можно представить А-схемой при следующих условиях:

1) каждый элементарный канал, передающий сигналы во внешнюю среду должен начинается в одном из выходных каналов первого агрегата А-схемы; каждый элементарный канал, передающий сигналы из внешней среды должен заканчиваться на одном из выходных каналов А-схемы;
2) сигналы в А-схеме передаются непосредственно от одного агрегата к другому без устройств, которые способны отсеивать сигналы, по каким-либо признакам;
3) согласование функционирования агрегатов А-схемы во времени;
4) сигналы между агрегатами предаются мгновенно, без искажений и перекодирования, изменяющего структуру сигнала.

Формализация и алгоритмизация информационных процессов

С развитием вычислительной техники наиболее эффективным методом исследования больших систем стало машинное моделирование, без которого невозможно решение многих крупных народнохозяйственных проблем. Поэтому актуальными задачами являются освоение теории и методов математического моделирования с учетом требований системности, анализ динамики и возможности управления машинным экспериментом с моделью, анализ адекватности моделей исследуемых систем.

Общие методологические аспекты широкого класса математических моделей позволяют исследовать механизм явления, протекающие в реальном объекте с большими или малыми скоростями, когда в натурных экспериментах с объектом трудно (или невозможно) проследить за изменениями, происходящими в течение короткого времени. или когда получение достоверных результатов сопряжено с длительным экспериментом. При необходимости машинная модель «растягивает» или «сжимает» реальное время, так как машинное моделирование связано с понятием системного времени, отличного от реального. Кроме того, с помощью машинного моделирования можно обучать персонал АСОИУ принятию решений в управлении объектом.

Сущность машинного моделирования системы состоит в проведении на ЭВМ эксперимента с моделью, которая представляет собой некоторый программный комплекс, описывающий формально и (или) алгоритмически поведение элементов системы *S* в процессе ее функционирования, т. е. в их взаимодействии друг с другом и внешней средой *E.*

Требованиями пользователя к модели М процесса функцинирования системы S являются:

1. Полнота модели должна предоставлять пользователю возможность получения необходимого набора оценок характеристик системы с требуемой точностью и достоверностью.
2. Гибкость модели должна давать возможность воспроизведения различных ситуаций при варьировании структуры, алгоритмов и параметров системы.
3. Длительность разработки и реализации модели большой системы должна быть по возможности минимальной при учете ограничений на имеющиеся ресурсы.
4. Структура модели должна быть блочной, т. е. допускать возможность замены, добавления и исключения некоторых частей без переделки всей модели.
5. Информационное обеспечение должно предоставлять возможность эффективной работы модели с базой данных систем определенного класса.
6. Программные и технические средства должны обеспечивать эффективную (по быстродействию и памяти) машинную реализацию модели и удобное общение с ней пользователя.
7. Должно быть реализовано проведение целенаправленных (планируемых) машинных экспериментов с моделью системы с использованием аналитико-имитационного подхода при наличии ограниченных вычислительных ресурсов.

Моделирование систем с помощью ЭВМ можно использовать в следующих случаях: а) для исследования системы *S* до того, как она спроектирована, с целью определения чувствительности характеристики к изменениям структуры, алгоритмов и пара метров объекта моделирования и внешней среды; б) на этапе проектирования системы *S* для анализа и синтеза различных вариантов системы и выбора среди конкурирующих такого вариантах; в) при эксплуатации системы, для получения информации, дополняющей результаты натурных испытаний (эксплуатации) реальной системы, и получения прогнозов развития системы во времени.

1.1 Концептуальные модели

Первым этапом машинного моделирования является построение *концептуальной модели* **М,** системы *S* и ее формализация, т. е. основным назначением этого этапа является переход от содержательного описания объекта к его математической модели. Наиболее ответственными и наименее формализованными моментами в этой работе являются проведение границы между системой *S* и внешней средой *E,* упрощение описания системы и построение сначала концептуальной, а затем формальной модели системы. Модель должна быть адекватной, иначе невозможно получить положительные результаты моделирования. Под адекватной моделью понимается модель, которая с определенной степенью приближения на уровне понимания моделируемой системы *S* разработчиком модели отражает процесс ее функционирования во внешней среде *E.*

Наиболее рационально строить модель функционирования системы по блочному принципу. Могут выделяться три автономные группы блоков такой модели:

1 группа: представляют собой имитатор воздействий внешней среды *E* на систему *S*;

2 группа: является собственно моделью процесса функционирования исследуемой системы *S;*

3 группа: служит для машинной реализации блоков двух первых групп, а также для фиксации и обработки результатов моделирования.

После перехода от описания | моделируемой системы *S* к ее модели *M,* построенной по блочному принципу, строятся математические модели процессов, происходящих в различных блоках. Математическая модель представляет собой совокупность соотношений (например, уравнений, логических условий, операторов), определяющих характеристики процесса функционирования системы *S* в зависимости от структуры системы, алгоритмов поведения, параметров системы, воздействий внешней среды *E,* начальных условий и времени

Формализации процесса функционирования любой системы S должно предшествовать изучение составляющих его явлений. Результатом является описание процесса, в котором изложены закономерности, характерные для исследуемого процесса, и постановку прикладной задачи. Содержательное описание является исходным материалом для последующих этапов формализации. Для моделирования процесса функционирования системы на ЭВМ необходимо преобразовать математическую модель процесса в соответствующий моделирующий алгоритм и машинную программу.

Последовательность построения концептуальной модели *M*, системы и ее формализации:

1. Постановка задачи машинного моделирования системы.
2. Анализ задачи моделирования системы.
3. Определение требований к исходной информации об объекте моделирования и организация ее сбора.
4. Выдвижение гипотез и принятие предположений.
5. Определение параметров и переменных модели.
6. Установление основного содержания модели.
7. Обоснование критериев оценки эффективности системы.
8. Определение процедур аппроксимации;
9. Описание концептуальной модели системы.
10. Проверка достоверности концептуальной модели.
11. Составление технической документации по первому этапу.

2. Алгоритмизация моделей

Вторым этапом моделирования является этап алгоритмизации модели и ее машинная реализация. Этот этап представляет собой этап, направленный на реализацию идей и математических схем в виде машинной модели *M* процесса функционирования систем *S*.

Процесс функционирования системы *S* можно рассматривать как последовательную смену ее состояний $\vec{z} = z(z_1(t), z_2(t), ..., z_k(t))$ в k-мерном пространстве. Задачей моделирования процесса функционирования исследуемой системы *S* является построение функций *z,* на основе которых можно провести вычисление интересующих характеристик процесса функционирования системы. Для этого необходимы соотношения, связывающие функции *z* с переменными, параметрами и временем, а также начальные условиями $\vec{z}^{\,0} = z(z_1(t_0), z_2(t_0), ..., z_k(t_0))$ в момент времени $t = t_0$.

Существуют два типа состояний системы:

1) особые, присущие процессу функционирования системы только в некоторые моменты времени;

2) неособые, в которых процесс находится все остальное время. В этом случае функция состояния $z_i(t)$ могут изменяться скачкообразно, а между особыми – плавно.

Моделирующие алгоритмы могут быть построены по «принципу особых состояний». Обозначим скачкообразное (релейное) изменение состояния *z* как δz, а «принцип особых состояний» — как *принцип* δz.

«*Принцип* δz» дает возможность для ряда систем существенно уменьшить затраты машинного времени на реализацию моделирующих алгоритмов.

Удобной формой представления логической структуры моделей процессов функционирования систем и машинных программ является схема. На различных этапах моделирования составляются следующие схемы моделирующих алгоритмов и программ:

Обобщенная (укрупненная) схема моделирующего алгоритма задает общий порядок действий при моделировании системы без каких-либо уточняющих деталей.

Детальная схема моделирующего алгоритма содержит уточнения, отсутствующие в обобщенной схеме.

Логическая схема моделирующего алгоритма представляет собо логическую структуру модели процесса функционирования систем *S*.

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.

Этапы алгоритмизации модели и ее машинной реализации:

1. Построение логической схемы модели.
2. Получение математических соотношении.
3. Проверка достоверности модели системы.
4. Выбор инструментальных средств для моделирования.
5. Составление плана выполнения работ по программированию.
6. Спецификация и построение схемы программы.
7. Верификация и проверка достоверности схемы программы.
8. Проведение программирования модели.
9. Проверка достоверности программы.
10. Составление технической документации по второму этапу.

3. Общая характеристика метода статистического моделирования

Статистическое моделирование представляет собой метод получения с помощью ЭВМ статистически данных о процессах, происходящих в моделируемой системе.

Сущность метода статистического моделирования сводится к построению для процесса функционирования исследуемой системы *S* некоторого моделирующего алгоритма, имитирующего поведение и взаимодействие элементов системы с учетом случайных входных воздействий и воздействий внешней среды *E*, и реализации этого алгоритма с использованием программно-технических средств ЭВМ.

Метод применяется:

1) для изучения стохастических систем;

2) для решения детерминированных задач.

Особенностью применения метода заключается во втором методе. А именно замена детерминированной задачи эквивалентной схемой некоторой стохастической системы, выходные характеристики последней совпадают с результатом решения детерминированной задачи.

В результате статистического моделирования системы *S* получается серия частных значений искомых величин или функций, статистическая обработка которых позволяет получить сведения о поведении реального объекта или процесса в произвольные моменты времени. Если количество реализации *N* достаточно велико, то полученные результаты моделирования системы приобретают статистическую устойчивость и с достаточной точностью могут быть приняты в качестве оценок искомых характеристик процесса функционирования системы *S*.

Теоретической основой метода статистического моделирования систем на ЭВМ являются *предельные теоремы теории вероятностей*. Множества случайных явлений (событий, величин) подчиняются определенным закономерностям, позволяющим не только прогнозировать их поведение, но и количественно оценить некоторые средние их характеристики, проявляющие определенную устойчивость.

Примеры статистического моделирования. Методом статистического моделирования найти оценки выходных характеристик стохастической системы $S_{R.,}$ функционирование которой описывается следующими соотношениями:

$x = 1 - e^{-\lambda}$ - входное воздействие;

$v = 1 - e^{-\varphi}$ - воздействие внешней среды;

λ и φ - случайные величины, для которых известны функции распределения.

Целью моделирования является оценка математического ожидания *M[y]* величины $y = \sqrt{x^2 + v^2}$

В качестве оценки математического ожидания *M* [у], как следует из приведенных теорем теории вероятностей, может выступать среднее арифметическое, вычисленное по формуле

$$\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i,$$

где y_i — случайное значение величины *y; N* — число реализации мат. ожиданий, которое достаточно для статистической устойчивости результатов.

Структурная схема системы S_R показана на рис. 1.

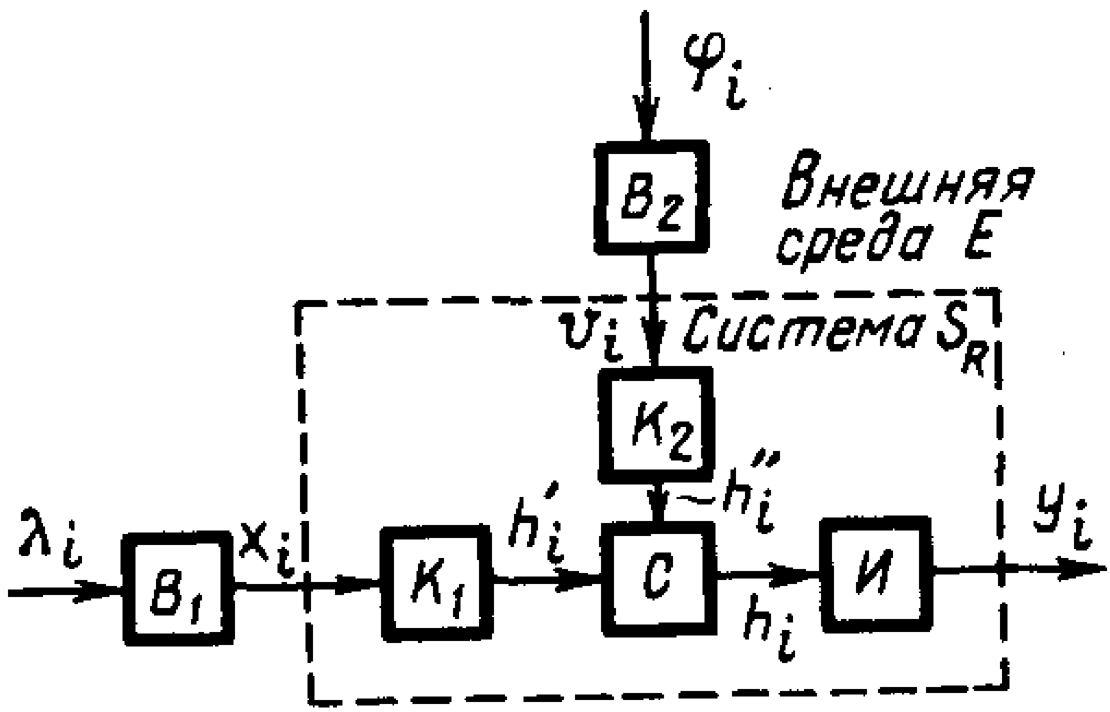


Рис. 1. Структурная схема системы S_R

Здесь элементы выполняют следующие функции:

вычисление

B_1, B_2 на выходе $x_i=1-e^{-\lambda_i}$ и B_2: $v_i=1-e^{-\varphi_i}$;

K1 и K2: $h_i'=(1-e^{-\lambda_i})^2$; K_2: $h_i''=(1-e^{-\varphi_i})^2$;

суммирование C: $h_i=(1-e^{-\lambda_i})^2+(1-e^{-\varphi_i})^2$;

извлечение квадратного корня И $y_i=\sqrt{(1-e^{-\lambda_i})^2+(1-e^{-\lambda_i})^2}$.

Схема алгоритма, реализующего метод статистического моделирования для оценки *M[y]* системы S_R, приведена на рис. 2.

Здесь *LA* и *FI* — функции распределения случайных величин λ и φ;

N — заданное число реализации;

I=i — номер текущей реализации;

LAT = λ_I;

FII = φ_I;

EXP = e;

MY = M[y] ;

$SY = \sum_{y=1}^{N} y_i$

ВИД [...], ГЕН [...], ВРМ[...]—процедуры ввода исходных данных, генерации псевдослучайных последовательностей и выдачи результатов моделирования соответственно.

Таким образом, данная модель позволяет получить методом статистического моделирования на ЭВМ статистическую оценку математического ожидания выходной характеристики *M[y]* рассмотренной стохастической системы S_R**.** Точность и достоверность результатов взаимодействия в основном будут определяться числом реализации *N*.

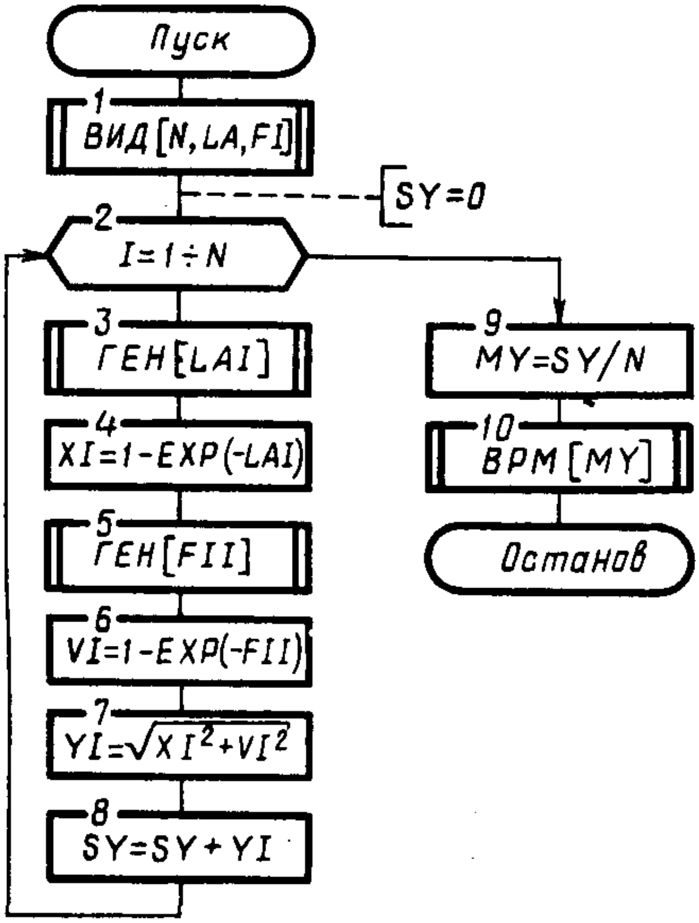


Рис. 2. Схема моделирующего алгоритма системы S_R

ПСЕВДОСЛУЧАЙНЫЕ ПОСЛЕДОВАТЕЛЬНОСТИ И ПРОЦЕДУРЫ ИХ МАШИННОЙ ГЕНЕРАЦИИ

При статистическом моделировании систем одним из основных вопросов является учет стохастических воздействий. Количество случайных чисел, используемых для получения статистически устойчивой оценки характеристики процесса функционирования системы S при реализации моделирующего алгоритма на ЭВМ. Количество случайных чисел колеблется в достаточно широких пределах в зависимости от:

1 класса объекта моделирования;

2.вида оцениваемых характеристик;

3необходимой точности и достоверности результатов моделирования. Результаты статистического моделирования существенно зависят от качества исходных (базовых) последовательностей случайных чисел.

На практике используются три основных способа генерации случайных чисел:

– аппаратный (физический);
– табличный (файловый);
– алгоритмический (программный).

Аппаратный способ. Генерация случайных чисел вырабатываются специальной электронной приставкой — генератором (датчиком) случайных чисел,— служащей в качестве одного из внешних устройств ЭВМ. Реализация этого способа генерации не требует дополнительных вычислительных операций ЭВМ по выработке случайных чисел, а необходима только операция обращения к внешнему устройству (датчику). В основе лежит физический эффект, лежащего в основе таких генераторов чисел, чаще всего используются шумы в электронных и полупроводниковых приборах, явления распада радиоактивных элементов и т. д.

Достоинства:

Запас чисел не ограничен;

Расходуется мало операций;

Не занимается место в памяти .

Недостатки:

Требуется периодическая проверка;

Нельзя воспроизводить последовательности;

Используется специальное устройство;

Необходимы меры по обеспечению стабильности.

Табличный способ. Случайные числа, представленные в виде таблицы, помещаются в память ЭВМ. Этот способ получения случайных чисел обычно используют при сравнительно небольшом объеме таблицы и файла чисел.

Достоинства:

Требуется однократная проверка;

Можно воспроизводить последовательности.

Недостатки:

Запас чисел ограничен;

Много места в ОЗУ;

Необходимо время для обращения к памяти.

Алгоритмический способ. Способ получения последовательности случайных чисел основанный на формировании случайных чисел в ЭВМ с помощью специальных алгоритмов и реализующих их программ. Каждое случайное число вычисляется с помощью соответствующей программы по мере возникновения потребностей при моделировании системы на ЭВМ.

Достоинства:

Требуется однократная проверка;

Многократная воспроизводимость последовательности чисел;

Мало места в памяти и нет внешних устройств.

Недостатки:

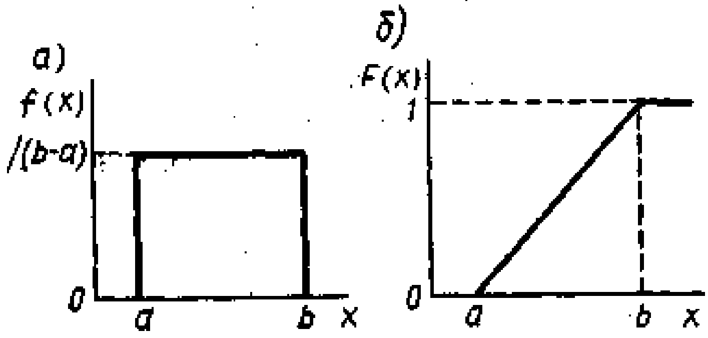


Рис. 4.9. Равномерное распределение случайной величины

Запас чисел ограничен периодом последовательности;

Затраты машинного времени.

Программная имитация случайных воздействий сводится к генерированию некоторых стандартных процессов и их последующего функционального преобразования. В качестве базового может быть принят любой удобный для моделирования конкретной системы *S* процесс (например, пуассоновский поток при моделировании Q-схемы). При дискретном моделирований базовым процессом является последовательность чисел $\{x_i\} = x_0, x_1, x_3, ..., x_N$, которые представляют реализации независимых, равномерно распределенных на интервале (0, 1) случайных величин $\{\xi_i\} = \xi_1, \xi_2, \xi_3, ..., \xi_N$. В статистических терминах - повторная выборка из равномерно распределенной на интервале (0, 1) генеральной совокупности значений величины ξ.

Непрерывная случайная величина ξ имеет равномерное распределение в интервале (а, Ь), если ее функции плотности (а) и функция распределения (б) примет вид (Рис. 1):

$$f(x)=\begin{cases}1/(b-a), & a\leqslant x\leqslant b,\\ 0, & x<a,\ x>b;\end{cases}$$

$$F(x)=\begin{cases}0, & x<a,\\ (x-a)/(b-a), & a\leqslant x\leqslant b,\\ 1, & x>b.\end{cases}$$

Рис.1

Числовые характеристики случайной величины ξ, принимающей значения х— это математическое ожидание, дисперсия и среднее квадратическое отклонение соответственно:

$$M|\xi|=\int_a^b xf(x)\,dx=\int_a^b x dx/(b-a)=(a+b)/2;$$

$$D|\xi|=\int_a^b (x-M|\xi|)^2 f(x)\,dx=(b-a)^2/12;$$

$$\sigma_\xi=+\sqrt{D|\xi|}=(b-a)/(2\sqrt{3}).$$

При моделировании систем на с случайными числами интервала (0, 1), где границы интервала соответственно а=0 и б = 1. Частным случаем равномерного распределения является функция плотности и функция распределения, соответственно имеющие вид:

$$f(x)=\begin{cases}1, & 0\leqslant x\leqslant 1,\\ 0, & x<0,\ x>1;\end{cases} \qquad F(x)=\begin{cases}0, & x<0,\\ x, & 0\leqslant x\leqslant 1,\\ 1, & x>1.\end{cases}$$

Такое распределение имеет математическое ожидание M [ξ] = 1/2 и дисперсию D[ξ] = 1/12.

Это распределение требуется получить на ЭВМ. Но получить его на цифровой ЭВМ невозможно, так как машина оперирует с п-разрядными числами. Поэтому на ЭВМ вместо непрерывной совокупности равномерных случайных чисел интервала (0, 1) используют дискретную последовательность 2" случайных чисел того же интервала. Закон распределения такой дискретной последовательности называют *квазиравномерным распределением*.

Случайная величина ξ, имеющая квазиравномерное распределение в интервале (0, 1), принимает значения $x_i = i/(2^n - 1)$ с вероятностями $p_i = \frac{1}{2}^n$, $i = \overline{0,2^n - 1}$.

Математическое ожидание и дисперсия квазиравномерной случайной величины соответственно имеют вид

$$M[\xi]=\sum_{i=0}^{2^n-1}\frac{i}{2^n-1}\frac{1}{2^n}=\frac{1}{(2^n-1)2^n}\sum_{i=0}^{2^n-1}i=\frac{(2^n-1)2^n}{(2^n-1)2^n2}=\frac{1}{2};$$

$$D[\xi]=\sum_{i=0}^{2^n-1}\frac{1}{2^n}\left[\frac{i}{2^n-1}-\frac{1}{2}\right]^2=\frac{1}{2^n}\sum_{i=0}^{2^n-1}\left(\frac{i^2}{(2^n-1)^2}-\frac{i}{2^n-1}+\frac{1}{4}\right)=$$

$$=\frac{1}{2^n}\left(\frac{(2^n-1)\,2^n(2^{n+1}-1)}{6(2^n-1)^2}-\frac{(2^n-1)2^n}{2(2^n-1)}+\frac{1}{4}\right)=\frac{1}{12}\frac{2^n+1}{2^n-1}.$$

На ЭВМ невозможно получить идеальную последовательность случайных чисел хотя бы потому, что на ней можно оперировать только с конечным множеством чисел. Кроме того, для получения значений x случайной величины ξ используются формулы (алгоритмы). Поэтому такие последовательности, являющиеся по своей сути детерминированными, называются псевдослучайными.

Требования к генератору случайных чисел.

Требованиями, к идеальному генератору случайных чисел формулируются следующим образом. Полученные с помощью идеального генератора псевдослучайные последовательности чисел должны:

- состоять из квазиравномерно распределенных чисел;
- содержать статистически независимые числа;
- быть воспроизводимыми;
- иметь неповторяющиеся числа;
- получаться с минимальными затратами машинного времени;
- занимать минимальный объем машинной памяти.

В практике моделирования применяются генерации последовательностей псевдослучайных чисел находят алгоритмы вида (1)

(1)

$$x_{i+1}=\Phi(x_i),$$

Данные алгоритмы представляют *рекуррентные соотношения* первого порядка, для которых начальное число x_0 и постоянные параметры уже заданы.

Метод серединных квадратов

Пусть имеется 2n-разрядное число, меньшее 1:

$x_i = a_1, a_2, ..., a_{2n}$

1.Возведем его в квадрат:

$x_i^2 = 0, b_1, b_2, ..., b_{4n}$

2. Отберем средние 2n разрядов $x_{i+1} = 0, b_{n+1}, b_{n+2}, ..., b_{3n}$ которые будут являться очередным числом псевдослучайной последовательности.

Пример, если начальное число x_0=0,2152, то $(x_0)^2$=0,04631104, т. е. Xj=0,6311, затем $(x_1)^2$=0,39828721, т. е. x_2=0,8287, и т. д.

Недостаток метода:

Наличие корреляции между числами последовательности, в некоторых случаях может отсутствовать.

Конгруэнтные процедуры генерации.

Конгруэнтные процедуры представляют собой арифметические операции, в основе которых лежит фундаментальное понятие конгруэнтности.

Два целых числа α и β конгруэнтны или сравнимы по модулю m, m — целое число, тогда и только тогда, когда существует такое целое число k, что $\alpha - \beta = km$ т. е. разность $\alpha - \beta$ делится на m и числа α и β дают одинаковые остатки от деления на абсолютную величину числа m.

Например,

$$1984 \equiv 4(\mathrm{mod}10)$$

$$1984 - 4 = 1984 \times 10$$

$$ост(\frac{1984}{10}) = ост(\frac{4}{10}) = 4$$

Конгруэнтные процедуры являются чисто детерминированными, так как описываются в виде рекуррентного соотношения (1), и имеют вид.

$$X_{i+1} = \lambda X_i + \mu\,(\mathrm{mod}\,M), \qquad (2)$$

где X_i, λ, μ, M — неотрицательные целые числа.

Раскроем рекуррентное соотношение (2):

$$\begin{aligned}
&X_1 = \lambda X_0 + \mu\,(\mathrm{mod}\,M);\\
&X_2 = \lambda X_1 + \mu = \lambda^2 X_0 + (\lambda+1)\mu(\mathrm{mod}\,M);\\
&X_3 = \lambda X_2 + \mu = \lambda^3 X_0 + (\lambda^2+\lambda+1)\mu = \lambda^3 X_0 + (\lambda^3 - 1)\mu/(\lambda-1)(\mathrm{mod}\,M);\\
&\cdots\\
&X_i = \lambda^i X_0 + (\lambda^i - 1)\mu/(\lambda-1)(\mathrm{mod}\,M). \qquad (3)
\end{aligned}$$

Если заданы начальные числа X_0, λ, μ, M (3) последовательность целых чисел *{Xi}*, составленную из остатков от деления на *M* членов последовательности

$$\left\{\frac{\lambda^i X_0 + \mu(\lambda^i - 1)}{(\lambda - 1)}\right\}$$

Таким образом, для любого i>=1 справедливо неравенство $X_t<M$, получится последовательность рациональных чисел из единичного интервала (0,1) $\{x_i\} = \{x_i / M\}$

Мультипликативный метод.

Задается последовательность неотрицательных целых чисел *{Xt}*, не превосходящих *M*, рассчитанных по формуле

$$X_{t+1} = \lambda X_t (\bmod M), \qquad (4)$$

т. е. это частный случай соотношения (2) при $\mu=0$.

В силу детерминированности метода получаются воспроизводимые последовательности..

В машинной реализации наиболее удобна версия $M=p^g$, где *p* — число цифр в системе счисления в ЭВМ; *g* -— число битов в машинном слове. Тогда вычисление остатка от деления на *M* сводится к выделению *g* младших разрядов делимого. Преобразование целого числа X_t в рациональную дробь из интервала $x_i \in (0,1)$ осуществляется подстановкой слева от X_i двоичной или десятичной запятой.

Алгоритм построения последовательности для двоичной машины $M=p^g$ сводится к выполнению таких операций:

1. Выбрать в качестве X_0 произвольное нечетное число.
2. Вычислить коэффициент $\lambda = 8t \pm 3$ где t — любое целое положительное число.
3. Найти произведение λX_0, содержащее не более 2g значащих разрядов.
4. Взять g младших разрядов в качестве первого члена последовательности X_1 а остальные отбросить.
5. Определить дробь $x_1 = \frac{X_1}{2^g}$ из интервала (0, 1).
6. Присвоить $X_0 = X_1$.
7. Вернуться к п. 3.

Смешанный метод.

Позволяет вычислить последовательность неотрицательных целых чисел *{Xi}*, не превосходящих *M*, по формуле

$$X_{t+1} = \lambda X_t + \mu (\bmod M),$$

Отличием от мультипликативного метода является $\mu \neq 0$.

С вычислительной точки зрения смешанный метод генерации сложнее мультипликативного на одну операцию сложения. При этом возможность выбора дополнительного параметра позволяет уменьшить возможную корреляцию получаемых чисел.

Моделирование случайных воздействий

В моделировании систем методами имитационного моделирования, существенное внимание уделяется учету случайных факторов и воздействий на систему. Для их формализации используются случайные события, дискретные и непрерывные величины, векторы, процессы. Формирование реализации случайных объектов любой природы сводится к генерации и преобразованию последовательностей случайных чисел.

В практике имитационного моделирования систем на ЭВМ ключевым факторам является оптимизация алгоритмов работы со случайными числами.

Таким образом, наличие эффективных методов, алгоритмов и программ формирования, необходимых для моделирования конкретных систем последовательностей случайных чисел, во многом определяет возможности практического использования машинной имитации для исследования и проектирования систем.

Моделирование случайных событий.

Простейшими случайными объектами при статистическом моделировании систем являются случайные события..

1. Пусть имеются случайные числа x_i т. е. возможные значения случайной величины ξ, равномерно распределенной в интервале (0, 1). Необходимо реализовать случайное событие *A*, наступающее с заданной вероятностью *p*. Определим *A* как событие, как сосотоящее в том, что выбранное значение x_i случайной величины ξ удовлетворяет неравенству

$$x_i \leqslant p. \qquad (1)$$

Тогда вероятность наступления события *A* будет $P(A) = \int_0^p dx = p$

Противоположное событие $\overline{A}$ состоит в том, что $x_i > p$. Тогда $P(\overline{A}) = 1—p$.

Процедура моделирования состоит в выборе значений x_i и сравнении их с p. Если условие (1) выполняется, то исходом испытания является событие A.

2. Пусть A1, A2, ..., A, — событий, наступающих с вероятностями p_1, p_2, ..., p. Определим A_m как событие, состоящее в том, что выбранное значение x_i, случайной величины удовлетворяет неравенству

$$l_{m-1} < x_i \leq l_m,$$

где $l_r = \sum_{i=1}^{r} p_i$. **Тогда**

$$P(A_m) = \int_{l_{m-1}}^{l_m} dx = p_m. \quad (2)$$

Процедура моделирования испытаний в последовательном сравнен... случайных чисел x_i со значениями l_r. Исходом испытания называется событие *Am*, если выполняется условие (2). Эту процедуру называют определением исхода испытания по жребию в соответствии с вероятностями $p_1, p_2, ..., p$

Пусть, независимые события *A и B*, поступающие с вероятностями p_A и p_B .Возможными исходами совместных испытаний будут события $AB, \overline{A}B, A\overline{B}, \overline{AB}$ с вероятностями $p_A p_B, (1-p_B)p_A, (1-p_A)p_B, (1-p_B)(1-p_A)$

В моделировании испытаний можно использовать два варианта расчетов:

1) последовательную проверку условия (2);

2) определение одного из исходов $AB, \overline{A}B, A\overline{B}, \overline{AB}$ по жребию с соответствующими вероятностями.

Для первого варианта необходима пара чисел x_i, для выполнения условия (1). Во втором варианте необходимо одно число x_i, но сравнений может потребоваться больше.

Пусть события *A* и *B* являются зависимыми. События наступают с вероятностями p_A и p_B. *P(B/A)* - условная вероятность наступления события *B* при что событие *A* произошло. Считается, что условная вероятность *P(B/A)* задана.

Из последовательности случайных чисел { x_i } извлекается число x_m, удовлетворяющее $x_m<p_л$. Если этой неравенство справедливо, то наступило событие *A*. Дальше из совокупности чисел *{x,}* берется очередное число x_{m+1} и проверяется условие $x_{m+1} \leq P(B/A)$. Возможный исход испытания являются *AB* или $A\overline{B}$.

Если условие $x_m < p_A$ не выполняется, то наступило событие *A*. Для испытания, связанного с событием *B,* необходимо определить вероятность

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

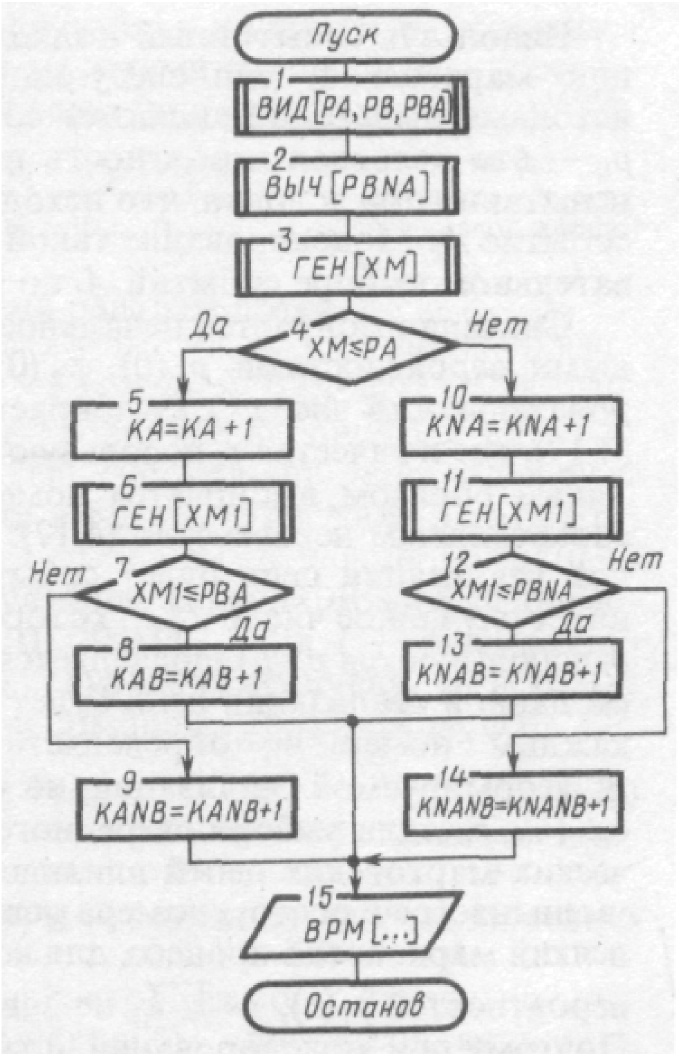
Выберем из совокупности *{x,}* число x_{m+1}, проверим справедливость неравенства $x_{m+1} \leq P(B/A)$. В зависимости от того, выполняется оно или нет, получим исходы испытания *A B* или *A B.*

<u>Схема моделирующего алгоритма для зависимых событий</u>

Алгоритм включает следующие процедуры:

ВИД [...]-процедура ввода исходных данных;

ГЕН [...] — генератор равномерно распределенных случайных чисел;
$XM = x_m$;
$XMI = x_{m+1}$;
$PA = p_A$ $PB = p_B$;
$PBA = P(B/A)$;
$PBNA = P(B/A)$;
KA, KNA, KAB, KANB, KNAB, KNANB — число событий $A, \overline{A}, AB, A\overline{B}, \overline{A}B, \overline{AB}$;
ВРМ [...] — процедура выдачи результатов моделирования.



Моделирование Марковских цепей

Пусть простая однородная марковская цепь определяется матрицей переходов

$$\mathbf{P} = \begin{Vmatrix} p_{11} & p_{12} \cdots p_{1k} \\ p_{21} & p_{22} \cdots p_{2k} \\ \cdots & \cdots \\ p_{k1} & p_{k2} \cdots p_{kk} \end{Vmatrix}, \quad 0 \leqslant p_{ij} \leqslant 1,$$

где P_{IJ} — вероятность перехода из состояния z_i, в состояние z_j.

Матрица переходов P полностью описывает марковский процесс. Так как сумма элементов каждой строки равна 1, то данная матрица является стохастической, т. е. $\sum_{j=1}^{k} p_{ij} = 1; i = \overline{1,k}$

Пусть $p_i(n)$, $i = \overline{1,k}$ - вероятность, что система будет находиться в состоянии z_i после n переходов. По определению $\sum_{i=1}^{k} p(n) = 1$.

Пусть возможными исходами испытаний являются события $A_1, A_2, .., A_k$. P_{IJ} — это условная вероятность наступления события A_J в данном испытании при условии, что исходом предыдущего испытания было событие A_I.

Моделирование такой цепи Маркова состоит в последовательном выборе событий A_J по жребию с вероятностями p_{ij}. Последовательность действий следующая:

1. выбирается начальное состояние z_0, задаваемое начальными вероятностями $p_1(0), p_2(0),...., p_k(0)$. Из последовательности чисел $\{x_i\}$ выбирается число x_m и сравнивается с (2). p_i - это значения $p_1(0), p_2(0),...., p_k(0)$. Выбирается номер m_0, удовлетворяющий неравенству (2). Начальным событием данной реализации цепи будет событие A_{m0}.
2. выбирается следующее случайное число x_{m+1}, которое сравнивается с l_τ. В качестве pi используются p_{m0j} . Определяется номер m_1. Следующим событием данной реализации цепи будет событие A_{m1} и т. д.

Каждый номер m_i, определяет не только очередное событие A_{mi} но и распределение вероятностей $p_{mi1}, p_{mi2}, ... p_{mik}$ для определения очередного номера m_{i+1}. Для эргодических марковских цепей влияние начальных вероятностей быстро уменьшается с ростом номера испытаний.

Эргодический марковский процесс - это всякий марковский процесс, для которого предельное распределение вероятностей $p_i(n)$, $i = \overline{1,k}$, не зависит от начальных условий $p_i(0)$. Поэтому можно принимать, что

$$p_1(0) = p_2(0) = ... = p_k(0) = 1/k.$$

Моделирование дискретных случайных величин.

Дискретная случайная величина η принимает значения $y_1 \le y_2 \le .... \le y_j$ с вероятностями $p_1,p_2,...,p_j$ составляющими дифференциальное распределение вероятностей

$$\begin{array}{ccc} y & y_1 & y_2 \dots y_j \dots, \\ P(\eta = y) & p_1 & p_2 \dots p_j \dots \end{array} \quad (3)$$

Интегральная функция распределения

$$F_\eta(y) = P(\eta \leqslant y) = \sum_{j=1}^{m} p_j;\ y_m \leqslant y \leqslant y_{m+1};\ m = 1, 2, ...;$$

$$F_\eta(y) = 0;\ y < y_1. \quad (4)$$

Для получения дискретных случайных величин используется метод обратной функции. Если ξ случайная величина, распределенная на интервале (0,1), то случайная $\eta = F_\eta^{-1}(\xi)$, величина η получается с помощью преобразования (5) где F_n^{-1} — функция, обратная F_n.

Алгоритм вычисления (3) и (4) сводится к выполнению следующих действий:

$$\begin{array}{l}
\text{если } x_1 < p, \text{ то } \eta = y_1, \text{ иначе} \\
\text{если } x_2 < p_1 + p_2, \text{ то } \eta = y_2, \text{ иначе,} \\
\dots\dots\dots\dots\dots\dots\dots\dots \\
\text{если } x_j < \sum\limits_{j=1}^{m} p_j, \text{ то } \eta = y_m, \text{ иначе,} \\
\dots\dots\dots\dots\dots\dots\dots\dots
\end{array} \qquad (6)$$

При счете по (6) среднее число циклов сравнения $\overline{\mu} = \sum\limits_{j=1}^{\infty} jp_j$.

<u>Моделирование непрерывных случайных величин</u>

Непрерывная случайная величина η задана функцией распределения

$$F_\eta(y) = P(\eta \leqslant y) = \int\limits_{-\infty}^{y} f_\eta(y)\,dy,$$

где $f_\eta(y)$ — плотность вероятностей.

Для получения непрерывных случайных величин используется метод обратной функции. Взаимно однозначная монотонная функция $\eta = F^{-1\eta}(\xi)$ преобразует случайную величину ξ, равномерно распределена на интервале (0,1) в случайную величину η с требуемой функцией плотности $f_\eta(y)$. Чтобы получить числа из последовательности {y_i}, имеющие функцию плотности $f_\eta(y)$, необходимо разрешить относительно y_i уравнение $\int\limits_{-\infty}^{y_i} f_\eta(y)dy = x_i$ (3)

Пример 1. Получить случайные числа с показательным законом распределения:

$$\begin{cases} f_\eta(y) = \lambda e^{-\lambda y}, y > 0 \\ f_\eta(y) = 0, y, 0 \end{cases}$$

В силу соотношения (3) получим

$$\lambda \int\limits_{0}^{\eta_i} e^{-\lambda y}\,dy = x_i,$$

где x_i — случайное число, имеющее равномерное распределение в интервале (0, 1). Тогда

$$1-e^{-\lambda y_j}=x_i,\ y_j=-\frac{1}{\lambda}\ln(1-x_i).$$

$1-\xi$ - случайная величина, распределенная на интервале (0, 1), поэтому

$$f_\eta(y)=\lambda(1-\lambda_y/2),\ 0\leqslant y\leqslant 2/\lambda.$$

можно записать $y_i=-\dfrac{1}{\lambda}\ln x_i$

Приближенные способы преобразования

В практике моделирования систем приближенные способы преобразования случайных чисел классифицируются следующим образом:

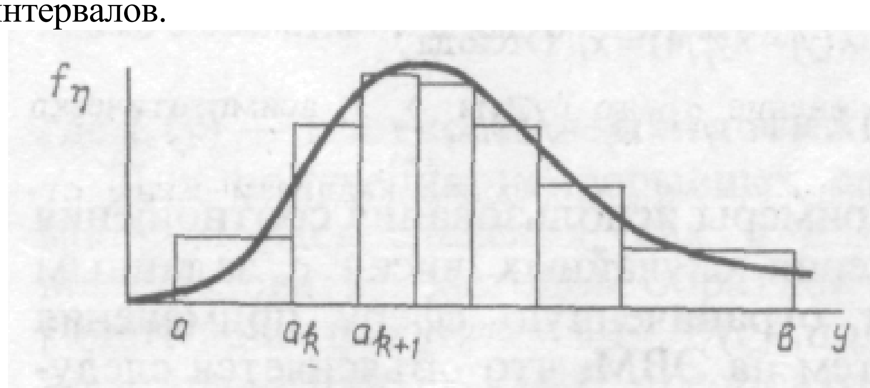
а) универсальные способы, с помощью которых можно получать случайные числа с законом распределения любого вида;

б) неуниверсальные способы, пригодные для получения случайных чисел с конкретным законом распределения.

Универсальный способ

Универсальный способ получения случайных чисел, базируется на кусочной аппроксимации функции плотности.

Пусть требуется получить последовательность случайных чисел $\{y_i\}$ с функцией плотности $f_n(y)$, возможные значения которой лежат в интервале (a, b). Представим $f_n(y)$ в виде кусочно-постоянной функции, т. е. разобьем интервал (a, b) на m интервалов.



Будем считать, что функция плотности на каждом интервале постоянна. Тогда случайную величину η можно представить в виде

$$\eta=ak+\eta_k^*$$

где ak— абсцисса левой границы k-го интервала;

η_k^* — случайная величина, возможные значения которой располагаются равномерно внутри k-го интервала.

На участке (a_k, a_{k+1}) случайная величина η_k^* распределена равномерно. Целесообразно разбить (a, b) на интервалы так, чтобы вероятность попадания случайной величины η_k^* в любой интервал (a_k, a_{k+1}) была постоянной и не зависела от номера интервала .

Для вычисления *ak* воспользуемся следующим соотношением:

$$\int_{a_k}^{a_{k+1}} f_\eta(y)\, dy = 1/m. \qquad (1)$$

Алгоритм машинной реализации этого способа получения случайных чисел сводится к выполнению следующих действий:

1) генерируется случайное равномерно распределенное число x_i из интервала (0, 1);

2) с помощью этого числа случайным образом выбирается интервал (a_k, a_{k+1});

3) генерируется число x_{i+1} и масштабируется с целью приведения его к интервалу (a_k, a_{k+1}), т. е. домножается на коэффициент $(a_{k+1} - a_k)x_{i+1}$

4) вычисляется случайное число $y_i = a_k + (a_{k+1} - a_k)x_{i+1}$ с требуемым законом распределения.

В п.2 целесообразно для этой цели построить таблицу (сформировать массив), в которую предварительно поместить номера интервалов *k* и значения коэффициента масштабирования, которые получаются из соотношения (1) для приведения числа к интервалу *(а, Ь)*. Получив из генератора случайное число x_i , с помощью таблицы сразу определяем абсциссу левой границы a_k и коэффициент масштабирования $(a_{k+1} - a_k)$.

Достоинства способа: При реализации на ЭВМ требуется небольшое количество операций для получения каждого случайного числа, так как операция масштабирования выполняется только один раз перед моделированием.

Не универсальные способы преобразования

Рассмотрим способы преобразования последовательности равномерно распределенных случайных чисел $\{x_i\}$ в последовательность с заданным законом распределения $\{y_j\}$ на основе предельных теорем теории вероятностей. Такие способы ориентированы на получение последовательностей чисел с конкретным законом распределения, т. е. не являются универсальными.

Пусть требуется получить последовательность случайных чисел имеющих распределение Пуассона.

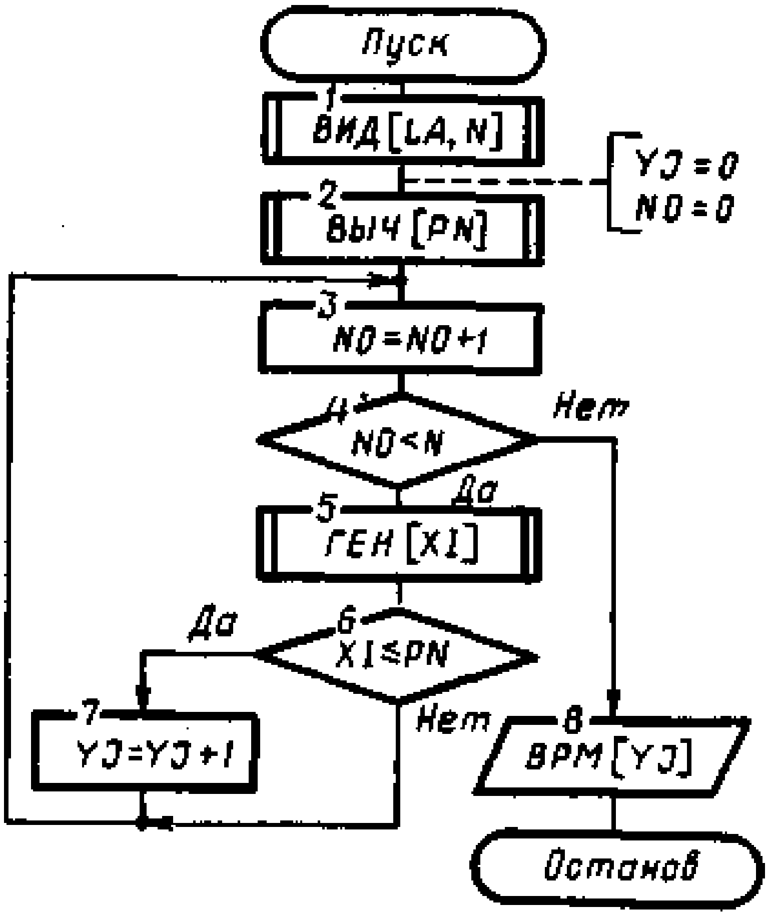
$$p(m) = \frac{\lambda^m e^{-\lambda}}{m!}$$

Воспользуемся предельной теорией Пуассона.

Если p- вероятность наступления события A в одном из испытаний, то вероятность наступления m событий в N независимых испытаниях при $N \to \infty, p \to 0, Np = \lambda$ ассимтотически равняется *p(m)*. выберем достаточно бостаточно большое количество испытаний N, такое что $p = \frac{\lambda}{N} < 1$.

Будем проводить серии из N независимых испытаний, в каждом из которых событие A наступает с вероятностью p. Будем подсчитывать число случаев y_j фактического наступления события A в серии с номером j. Число y_j будет приближенно следовать закону Пуассона. Практически номер выбирается таким образом, что $p \le 0.1 - 0.2$

Алгоритм



Алгоритм генерации последовательности случайных чисел y_p имеющих пуассоновское распределение.

$LA \equiv \lambda, N \equiv N, PN \equiv p, XI \equiv x_i$ — случайные числа последовательности, равномерно распределенной в интервале (0, 1);

$YJ \equiv y_j$:

NO — вспомогательная переменная;

ВИД [...] — процедура ввода исходных данных;

ВЫЧ [...] — процедура вычисления;

ГЕН [...] — процедура генерации случайных чисел;

ВРМ [...] — процедура выдачи результатов моделирования.

Моделирование случайных векторов.

При решении задач исследования характеристик процессов функционирования систем методом статистического моделирования на ЭВМ возникает необходимость в формировании реализаций случайных векторов, которые обладают заданными вероятностными характеристиками. Случайный вектор можно задать проекциями на оси координат, эти проекции являются случайными величинами, и описываются совместным законом распределения.

Случайные вектора можно задать проекциями на оси координат. В двухмерном случае, когда вероятность распределения на плоскости XOY, он может быть задан совместным законом распределения его проекций ξ и η на оси Ox и Oy.

Моделирование дискретных векторов

Пусть имеется дискретный случайный процесс. Двухмерная случайная величина (ξ,η) является дискретной. Ее составляющая ξ принимает возможные значения $x_1, x_2, ...x_n$. η принимает значения $y_1, y_2, ..., y_n$.

Каждой паре (x_i, y_i) соответствует вероятность pi . Возможному значению x_i случайной величины ξ, будет соответствовать

$$p_i = \sum_{j=1}^{n} p_{ij}.$$

В соответствии распределением вероятностей можно определить конкретное значение x_t случайной величины ξ и из значений p_{ij} выбрать последовательность

$$p_{i_1 1}, p_{i_1 2}, \dots, p_{i_1 n}, \tag{2}$$

которая описывает условное распределение величины η при условии $\xi = x_i$. Тогда конкретное значение y_i случайной величины η будет определяться в соответствии с распределением вероятностей (2). Пара чисел (x_i, y_i) будет первой реализацией моделируемого случайного вектора. Далее аналогичным образом определяем возможные значения x_{i_2}, выбираем последовательность

$$p_{i_2 1}, p_{i_2 2}, \dots, p_{i_2 n} \tag{3}$$

и находим д в соответствии с распределением (3). Это дает реализацию вектора (x_{i2}, y_{i2}) и т. д.

Моделирование непрерывных случайных векторов

Пусть величины ξ и η являются составляющими случайного вектора. В этом случае двухмерная случайная величина (ξ,η) описывается совместной функцией плотности *f(x, y)*.

$$f_\xi(x) = \int_{-\infty}^{\infty} f(x, y)\, dy.$$

С помощью функции плотности *f*ξ*(x)* находится случайное число x_t. При условии $\xi = x_i$ определяется условное распределение случайной величины η:

$$f_\eta(y/\xi = x_i) = f(x, y)/f_\xi(x_i).$$

По функции плотности определяется случайное число y_t. Пара чисел (x_i, y_i) будет являться искомой реализацией вектора (ξ,η).

В условиях многомерных векторов объем вычислений существенно увеличивается, что создает препятствия к использованию этого способа в практике моделирования систем.

В пространстве с числом измерений больше двух доступным оказывается формирование случайных векторов в рамках корреляционной теории. Рассмотрим случайный вектор с математическими ожиданиями $a_1, a_2, ..., a_n$ и корреляционной матрицей

$$K = \begin{Vmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{Vmatrix},$$

где $k_{ij} = k_{ji}$.

Пример. Рассмотрим трехмерный случай реализации трехмерного случайного вектора с составляющими (ξ,η,φ) и имеющего нормальное распределение с математическими ожиданиями $M[\xi] = a_1, M[\eta] = a_2, M[\varphi] = a3$ и корреляционной матрицей *K,* элементы которой являются дисперсиями случайных величин $k_{11} = D[\xi], k_{22} = D[\eta], k_{33} = D[\varphi]$. Элементы $k_{12} = k_{21}, k_{13} = k_{31}, k_{23} = k_{32}$ представляют собой соответственно корреляционные моменты ξ и η, ξ и η, η и φ.

Пусть имеется последовательность некорреляционных случайных чисел *{vi}*, имеющих одномерное нормальное распределение с параметрами a и σ. Выберем три числа v_1, v_2, v_3, преобразуем так, что они имеют характеристики a_1, a_2, a_3 и *K*.

Искомые составляющие случайного вектора (ξ,η,φ) обозначим как x, y, z и представим в виде линейного преобразования случайных величин v_i:

$$x=c_{11}(v_1-a)+a_1,\ y=c_{12}(v_1-a)+c_{22}(v_2-a)+a_2,$$

где c_{ij} — некоторые не известные коэффициенты. Для вычисления этих коэффициентов воспользуемся элементами корреляционной матрицы *K*. Велечины v_1, v_2, v_3 независимы между собой, то $M[(v_i-a)(v_j-a)]=0$ при $i\neq j$ В итоге имеем:

$$k_{11}=c_{11}^2\sigma^2,\quad k_{22}=c_{12}^2\sigma^2+c_{22}^2\sigma^2,$$
$$k_{33}=c_{13}^2\sigma^2+c_{33}^2\sigma^2,\quad k_{12}=c_{11}c_{12}\sigma^2,$$
$$k_{13}=c_{11}c_{13}\sigma^2,\quad k_{23}=c_{12}c_{13}\sigma^2+c_{23}^2\sigma^2.$$

Решая эту систему уравнения относительно c_{ij} получим

$$c_{11}=\sqrt{k_{11}}/\sigma,\quad c_{12}=k_{12}/(\sigma\sqrt{k_{11}}),\quad c_{13}=k_{13}/(\sigma\sqrt{k_{11}}),$$
$$c_{22}=\sqrt{k_{11}k_{22}-k_{12}^2}/(\sigma\sqrt{k_{11}}),$$
$$c_{23}=\sqrt{k_{11}k_{22}-k_{12}k_{13}}/(\sigma\sqrt{k_{11}}),$$
$$c_{33}=\sqrt{k_{11}k_{33}-k_{13}^2-k_{11}k_{23}+k_{12}k_{13}}/(\sigma\sqrt{k_{11}}).$$

Вычислив коэффициенты c_{ij} три последовательных случайных числа v_i i:=1, *2,* 3, преобразуются в составляющие случайного вектора (x_i, y_i, z_i).

Требуется хранить в памяти ЭВМ $n(n+1)/2$ корреляционных моментов k_{ij} и *n* математических ожиданий a_i. При больших *n* могут встречаться сложности, связанные с большим объемом вычислений.

Имитационное моделирование

Процедура имитационного моделирования.

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

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

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

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

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

Для решения задач такого вида применяется следующий алгоритм:

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

Имитация функционирования системы.

Предположим, исследуется вычислительная система (ВС), состоящая из процессора 1 с основной памятью, устройство вода перфокарт 4, АЦПУ 2 и дисплея 3 (рис. 4.1.).

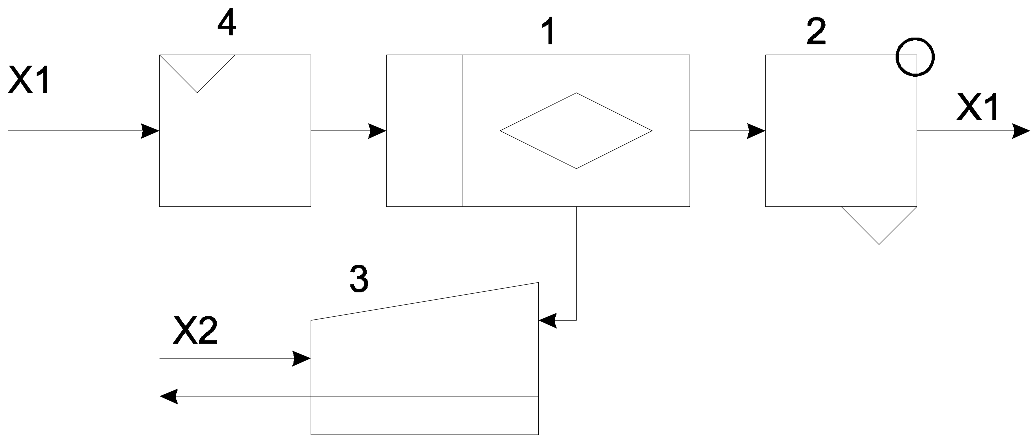


Рис. 4.1. Упрощённая схема моделируемой системы.

Через устройство 4 поступает поток заданий Х1. Процессор обрабатывает задания и результаты выдаёт на АЦПУ 2. Одновременно с этим ВС используется, например, как информационно-справочная система. Оператор-пользователь, работающий за дисплеем, посылает в систему запросы Х2, которые обрабатываются процессором и ответы выводятся на экран дисплея. Процессор работает в 2-х программном режиме: в одном разделе обрабатываются задания Х1, в другом, с более высоким относительным приоритетом запросы Х2. Представим данную ВС в упрощённом варианте в виде стохастической сети из 4-х СМО. Потоки заданий и запросы будем называть потоками заявок. Считаем потоки Х1 и Х2 независимыми. Известны ф.р. периодов следования заявок τ_1 и τ_2 и длительность обслуживания T_{1K}, T_{2K} заявок в к-ом устройстве. Требуется определить времена загрузки каждого устройства и времена реакции по каждому из потоков.

Вначале определяется момент поступления в систему 1-ой заявки потока Х1 по результатам случайного испытания в соответствии с ф.р. периода следования заявок.

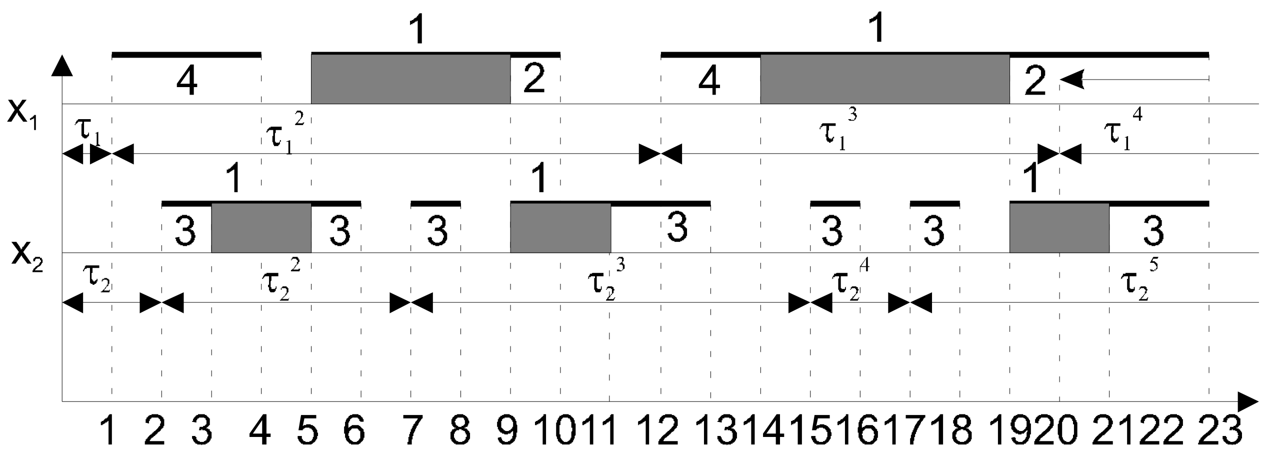


Рис. 4.2. Временная диаграмма функционирования ВС.

На рис. 2 это момент времени $t_1=0+\tau_1^1$ (здесь и далее верхний индекс обозначает порядковый номер заявки данного потока). То же самое делается для потока X2. На рис.2 момент поступления 1-ой заявки потока X2 $t_2=0+\tau_2^1$. Затем находится минимальное время, т.е. наиболее раннее событие. В примере это время t_1. Для 1-ой заявки потока X1определяется время обслуживания устройством ввода перфокарт $T^1{}_{14}$ методом случайного испытания и отмечается момент окончания обслуживания $t_4=t_1+ T^1{}_{14}$. На рис. показан переход устройства 4 в состояние "занято". Одновременно определяется момент поступления следующей заявки потока X1: $t_{12}=t_1+\tau_1^2$. Следующее минимальное время это момент поступления заявки потока X2 - t_2. Для этой заявки находится время обслуживания на дисплее $T^1{}_{23}$ и отслеживается время окончания обслуживания $t_3=t_2+ T^1{}_{23}$. Определяется момент поступления второй заявки потока X2: $t_7=t_2+\tau_2^2$. Снова выбирается минимальное время — это t_3. В этот момент заявка потока X2 начинает обрабатываться процессором. По результату случайного испытания определяется время её обслуживания $T^1{}_{21}$ и отмечается момент $t_5=t_3+ T^1{}_{21}$ окончания обслуживания. Следующее минимальное время t_4 - момент завершения обслуживания заявки потока X1 устройством 4. С этого момента заявка может начать обрабатываться процессором, но он занят обслуживанием потока X2. Тогда заявка потока X1 переходит в состояние ожидания, становиться в очередь. В следующий момент времени t_5 освобождается процессор. С этого момента процессор начинает обрабатывать заявку потока X1, а заявка потока X2 переходит на обслуживание дисплеем, т.е. ответ на запрос пользователя передаётся из основной памяти в буферный накопитель дисплея. Далее определяются соответствующие времена обслуживания: $T^1{}_{11}$ и $T^1{}_{23}$ и отмечаются моменты времени $t_9=t_5+ T^1{}_{11}$ и $t_6=t_5+ T^1{}_{23}$. В момент t_6 полностью завершается обработка первой заявки потока X2. По разности времени t_6 и t_2 вычисляется время реакции по этой заявке $u^1{}_2= t_6- t_2$. Следующий минимальный момент t_7 - это наступление 2-ой заявки потока X2. Определяет время поступления очередной заявки этого потока $t_{15}= t_7+\tau_2^3$. Затем вычисляется время обслуживания 2-ой заявки на дисплее $T^2{}_{23}$ и отмечается момент $t_8=t_7+ T^2{}_{23}$, после чего заявка становится в очередь, т.к. процессор занят. Эта заявка поступит на обслуживание в процессор только после его освобождения в момент t_9 . В этот момент заявка потока X1 начинает обслуживаться в АЦПУ. Определяются времена обслуживания $T^2{}_{21}$ и $T^1{}_{12}$ по результатам случайных испытаний и отмечаются моменты окончания обслуживания $t_{11}= t_9+T^2{}_{23}$ и $t_{10}= t_9+T^1{}_{12}$. В момент времени t_{10} завершается полное обслуживание 1-ой заявки потока X1. Разность между этим моментом и моментом времени t_1 даёт 1-ое значение времени реакции по потоку X1 $u^1{}_1= t_{10}- t_1$.

Указанные процедуры выполняются до истечения времени моделирования. В результате получается некоторое количество (выборка) случайных значений времени реакции (u_1) и (u_2) по 1-ому и 2-ому потокам. По этим значениям могут быть определены эмпирические функции распределения и вычислены количественные вероятностные характеристики времени реакции. В процессе моделирования можно суммировать продолжительности занятости каждого устройства обслуживанием всех потоков. Например, на рис. 2 занятость процессора 1 выделена заштрихованными ступеньками. Если результаты суммирования разделить на время моделирования, то получатся коэффициенты загрузки устройств.

Можно определить время ожидания заявок в очереди, обслуженных системой, среднюю и максимальную длину очереди заявок к каждому устройству, требуемая ёмкость памяти и др.

Имитация даёт возможность учесть надёжностные характеристики ВС. В частности, если известны времена наработки на отказ и восстановления всех входящих в систему устройств, то определяются моменты возникновения отказов устройств в период моделирования и моменты восстановления. Если устройство отказало, то возможны решения:

- снятие заявки без возврата;
- помещение заявки в очередь и дообслуживание после восстановления;
- поступление на повторное обслуживание из очереди;

Моделирование систем и языки программирования.

Большое значение при реализации модели на ЭВМ имеет вопрос правильного выбора языка программирования.

Язык программирования должен отражать внутреннюю структуру понятий при описании широкого круга понятий. Высокий уровень языка моделирования значительно упрощает программирование моделей.

Основными моментами при выборе ЯМ является:

- проблемная ориентация;
- возможности сбора, обработки, вывода результатов;
- быстродействие;
- простота отладки;
- доступность восприятия.

Этими свойствами обладают процедурные языки высокого уровня. Для моделирования могут быть использованы языки Имитационного моделирования (ЯИМ) и общего назначения (ЯОМ).

Более удобными являются ЯИМ. Они обеспечивают:

- удобство программирования модели системы;
- проблемная ориентация.

Недостатки ЯИМ:

- неэффективность рабочих программ;
- сложность отладки;
- недостаток документации.

Основные функции языка программирования:

- управление процессами (согласование системного и машинного времени);
- управление ресурсами (выбор и распределение ограниченных средств описываемой системы).

Как специализированные языки, ЯИМ обладают некоторыми программными свойствами и понятиями, которые не встречаются в ЯОН. К ним относятся:

Совмещение. Параллельно протекающие в реальных системах S процессы представляются с помощью последовательно работающей ЭВМ. ЯИМ позволяют обойти эту трудность путём введения понятий системного времени.

Размер. ЯИМ используют динамическое распределение памяти (компоненты модели системы M появляются в ОЗУ и исчезают в зависимости от текущего состояния. Эффективность моделирования достигается так же использованием блочных конструкций: блоков, подблоков и т.д.

Изменения. ЯИМ предусматривают обработку списков, отражающих изменения состояний процесса функционирования моделируемой системы на системном уровне.

Взаимосвязь. Для отражения большого количества между компонентами модели в статике и динамике ЯИМ включаем системно организованные логические возможности и реализации теории множеств.

Стохастичность. ЯИМ используют специальные программные генерации последовательностей случайных чисел, программы преобразования в соответствующие законы распределения.

Анализ. ЯИМ предусматривают системные способы статистической обработки и анализа результатов моделирования.

Наиболее известными языками моделирования являются SIMULA, SIMSCRIPT, GPSS, SOL, CSL.

Для языков, используемых в задачах моделирования, можно составить классификацию следующего вида. (см. рис. 9.1.)

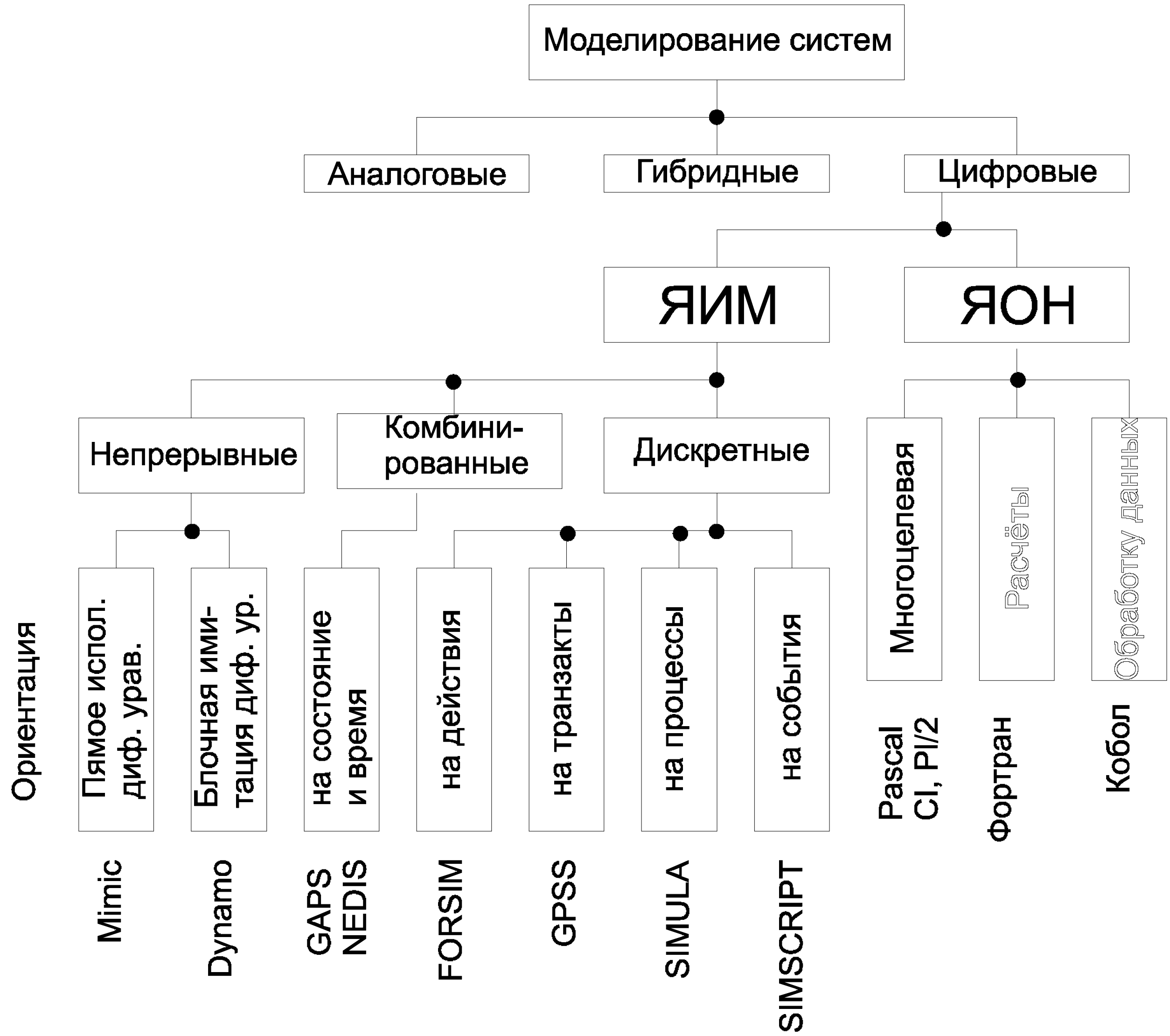


Рис. 9.1. Классификация языков моделирования.

Язык DYNAMO используется для решения разностных уравнений.

Представление системы S в виде типовой схемы, в которой участвуют как дискретные, так и непрерывные величины, называются комбинированными. Предполагается, что в системе могут наступать события двух видов: 1) события, от состоянии Z_i; 2) события, зависящие от времени t. При использовании языка GAPS на пользователь возлагается работа по составлению на яз. FORTRAN подпрограмм, в которых описываются условия наступления событий, законы изменения непрерывной величины, правил перехода из одного состояния в другое. SIMSCRIPT - язык событий, созданный на базе языка FORNRAN. Каждая модель M_j состоит из элементов, с которыми происходят события, представляющие собой последовательность формул, изменяющих состояние моделируемой системы с течением времени. Работа со списками, определяемые пользователем, последовательность событий в системном времени, работа с множествами. FORSIT - пакет ПП на языке FORNRAN позволяет оперировать только фиксированными массивами данных, описывающих объекты моделируемой системы. Удобен для описания систем с большим числом разнообразных ресурсов. Полное описание динамики модели можно получить с помощью ПП.

SIMULA - расширение языка ALGOL. Блочное представление моделируемой системы. Функционирование процесса разбивается на этапы, происходящие в системном времени. Главная роль в языке SIMULA отводится понятию параллельного оперирования с процессами в системном времени, универсальной обработки списков с процессами в роли компонент.

GPSS- интегрирующая языковая система, применяющаяся для описания пространственного движения объектов. Такие динамические объекты в языке GPSS называются транзактами и представляют собой элементы потока. Транзакты "создаются" и "уничтожаются". Функцию каждого из них можно представить как движение через модель М с поочерёдным воздействием на её блоки. Функциональный аппарат языка образуют блоки, описывающие логику модели, сообщая транзактам, куда двигаться и что делать дальше. Данные для ЭВМ подготавливаются в виде пакета управляющих и определяющих карт, которым составляется по схеме модели, набранной из стандартных символов. Созданная программа GPSS, работая в режиме интерпретации, генерирует и передаёт транзакты из блока в блок. Каждый переход транзакта приписывается к определенному моменту системного времени.

При моделировании предпочтение отдают языку, который более знаком, универсален. Вместе с увеличением числа команд возрастают трудности использования ЯИМ. Получены экспертные оценки ЯИМ по степени их эффективности.

Баллы	Возможности	Простота применения	Предпочтение пользователя
5	SIMULA	GPSS	SIMSCRIPT
4	SIMSCRIPT	SIMSCRIPT	GPSS
3	GPSS	SIMULA	SIMULA

Суммарный бал:

SIMULA -11
SIMSCRIPT -13
GPSS -12

Если предпочтение отдаётся блочной конструкции модели при наличии минимального опыта в моделировании, то следует выбрать язык GPSS, но при этом следует помнить, что он негибок, требует большого объёма памяти и затрат машинного времени для счёта.

Методы определения характеристик моделируемых систем.

Измеряемые характеристики моделируемых систем.

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

Для всей системы производится подсчёт поступивших в систему заявок, полностью обслуженных и покинувших систему заявок без обслуживания по тем или иным причинам. Соотношения этих величин характеризует производительность системы при определённой рабочей нагрузке.

По каждому потоку заявок могут вычисляться времена реакций и ожидания, количества обслуженных и потерянных заявок. По каждому устройству определяется время загрузки при обслуживании одной заявки м число обслуженным устройством заявок, время простоя устройства в результате отказов и количество отказов, возникших в процессе моделирования, дины очередей и занимаемые ёмкости памяти.

При статистическом моделировании большая часть характеристик — это случайные величины. По каждой такой характеристике **у** определяется N значений, по которым строится гистограмма относительных частот, вычисляется математическое ожидание, дисперсия и моменты более высокого порядка, определяются средние по времени и максимальные значения. Коэффициенты загрузки устройств вычисляются по формуле:

$$\boldsymbol{\rho_k = V_k * N_{ok} / T_m} \qquad (1)$$

V_k- среднее время обслуживания одной заявки к-ым устройством;

N_{ok} **-** количество обслуженных заявок устройством за время моделирования T_m.

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

Расчёт математического ожидания и дисперсии выходной характеристики.

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

$$m_n = m_{n-1} * (n-1)/n + y/n; \qquad (2)$$

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

$$d_n = d_{n-1} * (n-2)/(n-1) + 1/n * (y_n - m_{n-1})^2 \qquad (3)$$

здесь d_{n-1} - дисперсия, вычисленная на предыдущем шаге.

При большом количестве измерений эти оценки являются состоятельными и несмещёнными.

Расчёт среднего по времени значения выходной характеристики.

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

$$l_N = \sum_{i=1}^{N} l_i \tau_i \Big/ T_m \qquad (4)$$

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

Ёмкость накопитель: $$q_n = \sum_{i=1}^{N} q_i \tau_i \Big/ T_m \qquad (5)$$

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

Построение гистограммы для стационарной системы.

Г - эмпирическая плотность распределения вероятностей. Задаются границы изменения интересующей характеристики. $y_i \to [y_н; y_в]$, числом интервалов N_g. Определяется ширина интервала $\Delta = (y_н - y_в)/N_g$.

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

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

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

$$\chi^2 = \frac{\sum_{1}^{N_g} \left(R_i - N * P_i\right)^2}{N * P_i} \qquad (6);$$

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

$$P_i = \int_{y_i}^{y_{i+1}} \varphi(x)dx = F(y_{i+1}) - F(y_i) \qquad (7).$$

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

$M_k(z) = p(\chi^2 < z) = \dfrac{1}{2^{k/2} * \Gamma(\kappa/2)} \displaystyle\int_0^z t^{k/2 - 1} * e^{-t/2} dt$, где z - значение случайной величины χ^2 ,

k=N_g-(r +1) - число степеней свободы распределения χ^2 . r - количество параметров теоретического распределения, Г(к/2) - гамма функция.

Функция распределения χ^2 табулирована. По вычисленному значению χ^2 и числу степеней свободы с помощью таблиц определяется вероятность

P(χ^2<Z). Если она превышает заданный уровень значимости C, то выдвинутая гипотеза принимается.

Моделирование систем с использованием типовых математических схем

Блочные иерархические модели процессов функционирования систем

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

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

Моментами смены состояний модели $M_м$ в целом $t^{(k)} \le T$ будем считать все моменты изменения блоков $\{m_i\}$, рис. 8.1. см. ниже.

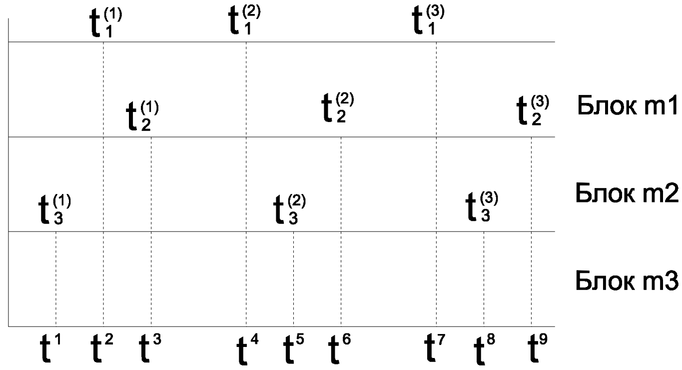


Рис. 8.1. Смена состояний модели для случаев 3-х блоков

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

Особенности реализации процессов с использованием Q-схем

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

Рассмотрим фрагмент Q-схемы (Рис. 8.2.):

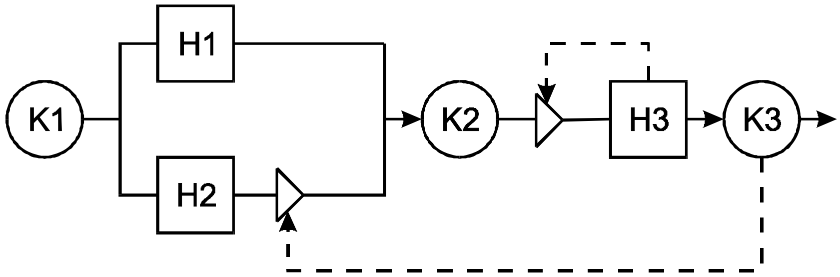


Рис. 8.2. Фрагмент Q-схемы.

Примерами управляющих связей являются различные блокировки обслуживающих каналов (по входу и по выходу): "клапаны" изображены в виде треугольников, а управляющие связи пунктирными линиями. Блокировка канала по входу означает, что этот канал отключается от входящего потока заявок, а блокировка канала по выходу указывает, что заявка обслуженная блокированным каналом, остаётся в этом канале до момента снятия блокировки. В этом случае, если перед накопителем нет "клапана", то при его переполнении будут иметь место потери заявок.

Моделирующий алгоритм должен отвечать следующим требованиям:

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

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

Все виды моделирующих алгоритмов Q-схемы можно классифицировать следующим образом (см. Рис. 8.3.):

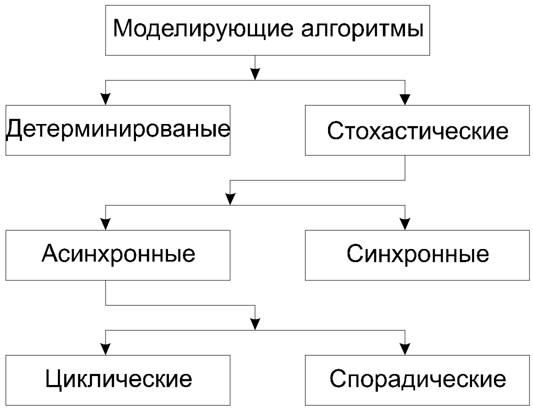


Рис. 8.3. Виды моделирующих алгоритмов Q-схемы.

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

При синхронном способе один из элементов Q-схемы (И, Н или К) выбирается в качестве ведущего и по нему "синхронизируется" весь процесс моделирования.

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

Построение и реализация моделирующих алгоритмов Q-схем

Прежде чем использовать какой либо язык для моделирования Q-схемы, необходимо глубже вникнуть в суть процесса построения и реализации М.А.

Пример. Рассмотрим Q-схему (Рис. 8.4.):

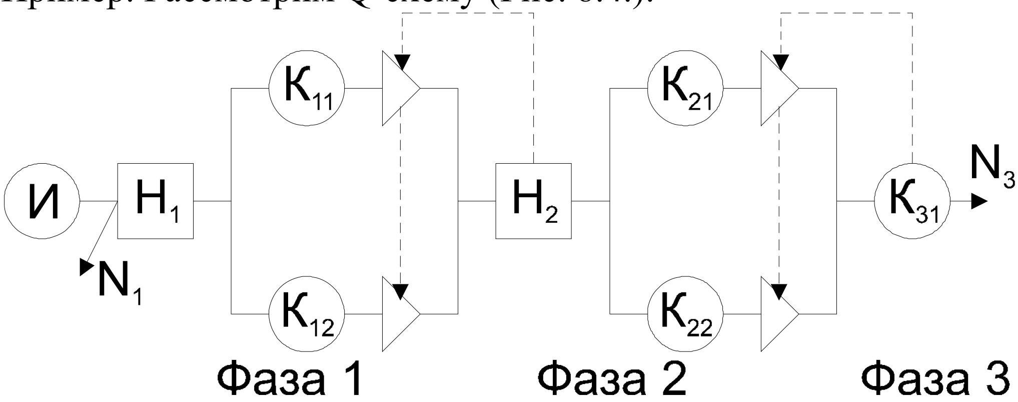


Рис. 8.4. Трехфазная Q-схема.

Примем обозначения:

P - вероятность потери заявки ($P=N_1/(N_1+N_3)$);

t_m - время появления очередной заявки из источника;
$t_{k,j}$ - время окончания обслуживания заявки каналом $K_{к,j}$, k=1,2,3…; j=1,2…;
z_i, $z_{k,j}$ - состояния накопителей и каналов обслуживания;
t_n - текущее время моделирования;
L_i - ёмкость i-ого накопителя;
L_k^m - число каналов в к-ой фазе;
N_1, N_2 - число выходных заявок;
Т - интервал моделирования;
При имитации Q-схемы на ЭВМ требуется организовать массив состояний:
$z_{k,j}$, $t_{k,j}$, j=1, L_k^m; z_i - число заявок в накопителе H_i; i=1,2; t_i - i-ая заявка из источника.
$z_{k,j}$ = {1- канал занят; 0 - канал свободен; 2 - заблокирован};
Укрупнённая схема детерминированного МА Q-схемы, построенного по "принципу Δt" представлена на рисунке 8.5.

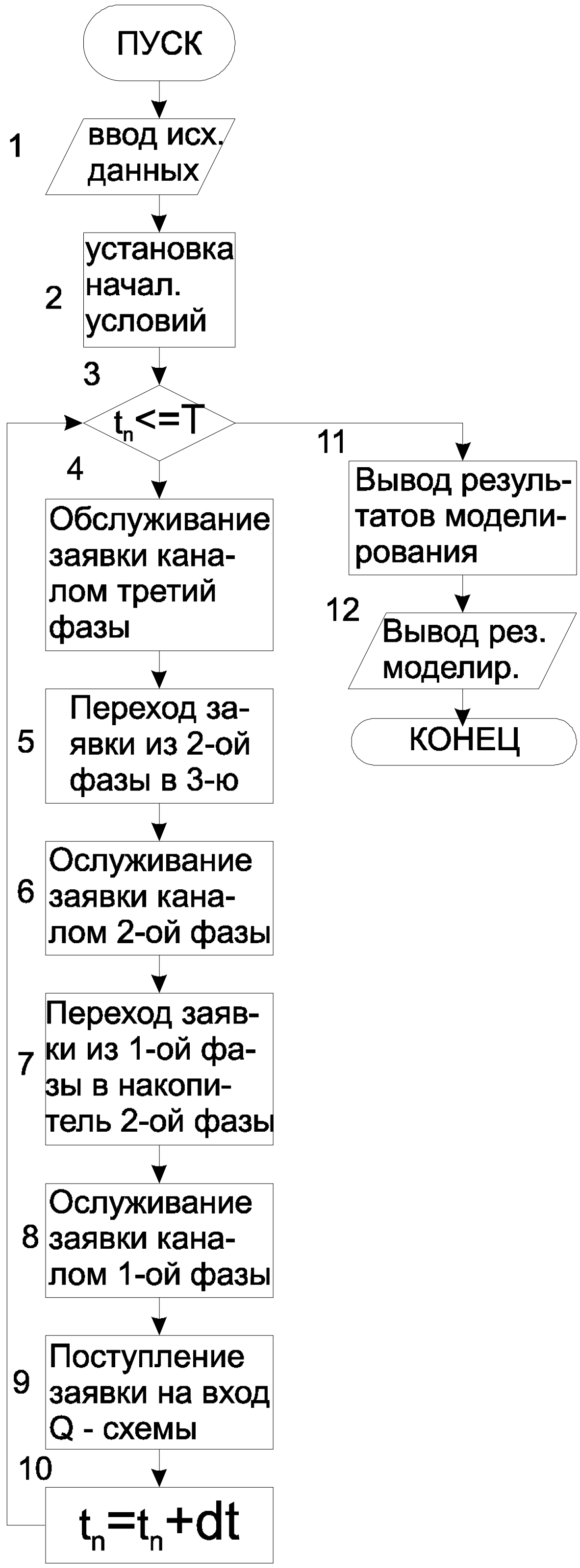
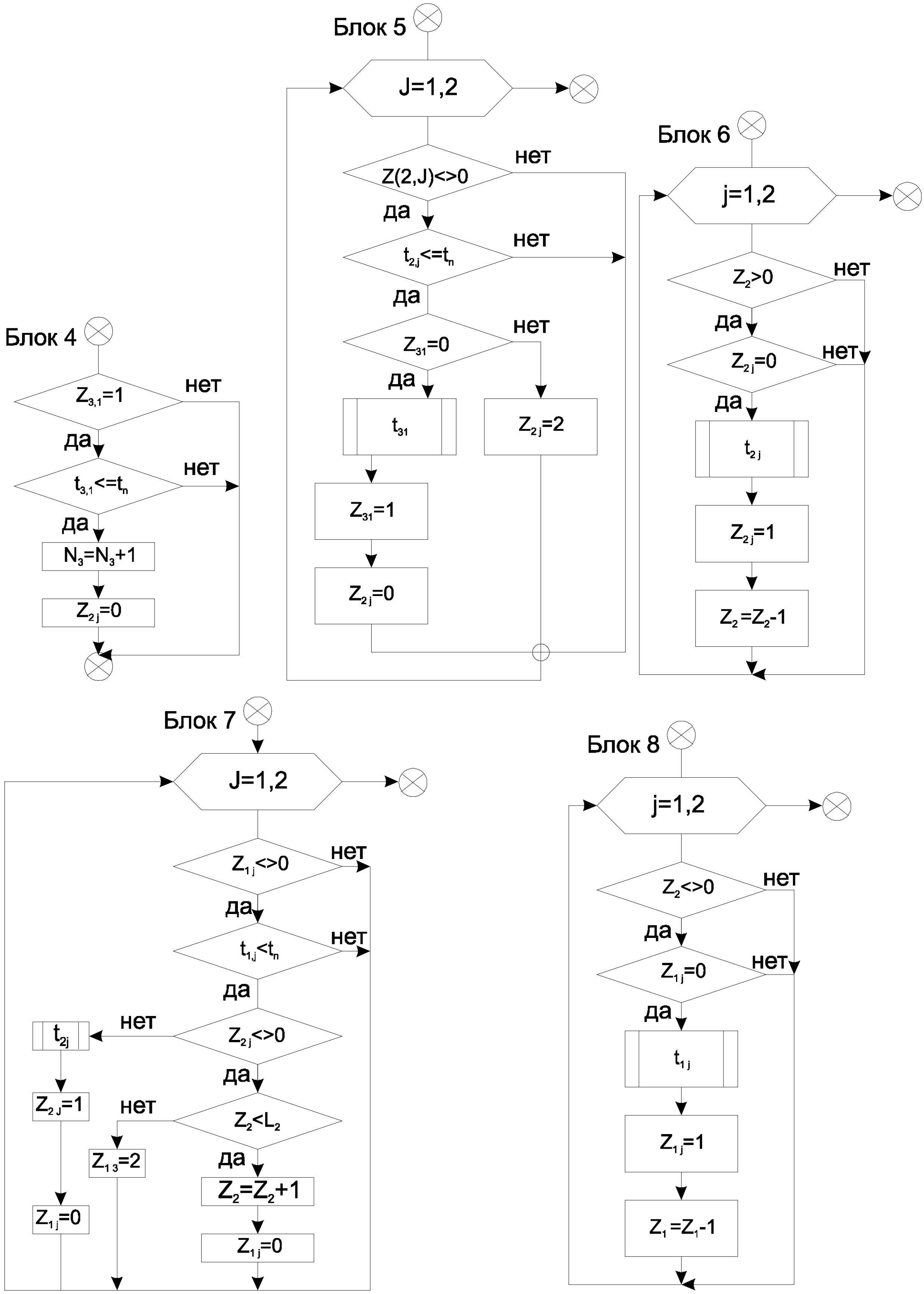
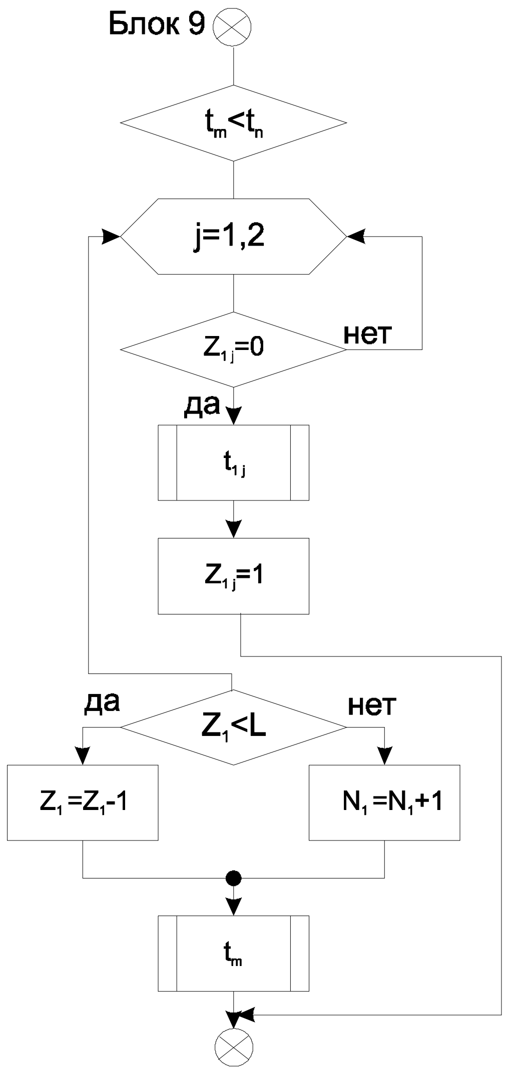


Рис. 8.5. Блок схема моделирования Q-схемы по принципу "Δt".

А далее более подробно рассмотрены алгоритмы блоков 4-9.


Блок 4
Z3,1=1
да
нет
t3,1<=tn
да
нет
N3=N3+1
Z2j=0
Блок 5
J=1,2
Z(2,J)<>0
да
нет
t2,j<=tn
да
нет
Z31=0
да
нет
t31
Z2j=2
Z31=1
Z2j=0
Блок 6
j=1,2
Z2>0
да
нет
Z2j=0
да
нет
t2j
Z2j=1
Z2=Z2-1
Блок 7
J=1,2
Z1j<>0
да
нет
t1,j<tn
да
нет
Z2j<>0
да
нет
t2j
Z2J=1
Z1j=0
Z2<L2
да
нет
Z13=2
Z2=Z2+1
Z1j=0
Блок 8
j=1,2
Z2<>0
да
нет
Z1j=0
да
нет
t1j
Z1j=1
Z1=Z1-1



Планирование машинных экспериментов с моделями систем.

Методы планирования эксперимента на модели.

Основная задача планирования машинных экспериментов заключается в получении необходимой информации об исследуемой системе при ограниченных ресурсах (затраты машинного времени, памяти и т.п.). К числу частных задач, решаемых при планировании машинных экспериментов, относятся задачи уменьшения затрат машинного времени на моделирование, уменьшения погрешности результатов моделирования, проверки адекватности модели и т.п.

Эффективность машинных экспериментов существенно зависит от выбора плана эксперимента, т.к. именно план определяет объём и порядок проведения вычислений на ЭВМ, приёмы накопления и статистической обработки результатов моделирования системы. Поэтому основная задача планирования машинных экспериментов с моделью формируется следующим образом: необходимо получить об объёме моделирования, заданном в виде моделирующего алгоритма (программы) при минимальных или ограниченных затратах машинных ресурсов на реализацию процесса моделирования.

Таким образом, при машинном моделировании необходимо не только рационально планировать и проектировать саму модель системы, но и процесс её использования, т.е. проведения с ней эксперимента.

При планировании машинных экспериментов возникает целый ряд проблем, взаимно связанных как с особенностью функционирования моделируемого объекта, так и с особенностью машинной реализации модели и обработки результатов эксперимента. В первую очередь к таким относятся проблемы построения плана машинного эксперимента, стохастической сходимости результатов, ограниченности машинных ресурсов, уменьшения дисперсии оценок, полученных на машинной модели и т.д.

Рассмотрим основные понятия теории планирования эксперимента. В планировании эксперимента различают входные (изогенные) и выходные (эндогенные) переменные: $x_1, x_2, \ldots, x_к$; $y_1, y_2 \ldots, y_e$. Входные переменные в ТПЭ называют <u>факторами</u> а выходные — <u>реакциями</u>. Каждый фактор x_i, $i=1,2,\ldots,k$ может принимать в эксперименте одно или несколько значений, называемых <u>уровнями</u>. Фиксированный набор уровней факторов определяет одно из возможных состояний рассматриваемой системы. Одновременно этот набор представляет собой условия проведения одного из возможных экспериментов.

Каждому фиксированному набору уровню факторов соответствует определённая точка в многомерном пространстве, называемая факторным пространством. Эксперименты не могут быть реализованы во всех точках факторного пространства, а лишь в принадлежащих допустимой области, как это например оказано для случая двух факторов X_1 и X_2 на рисунке (см. ниже рис. 1.).

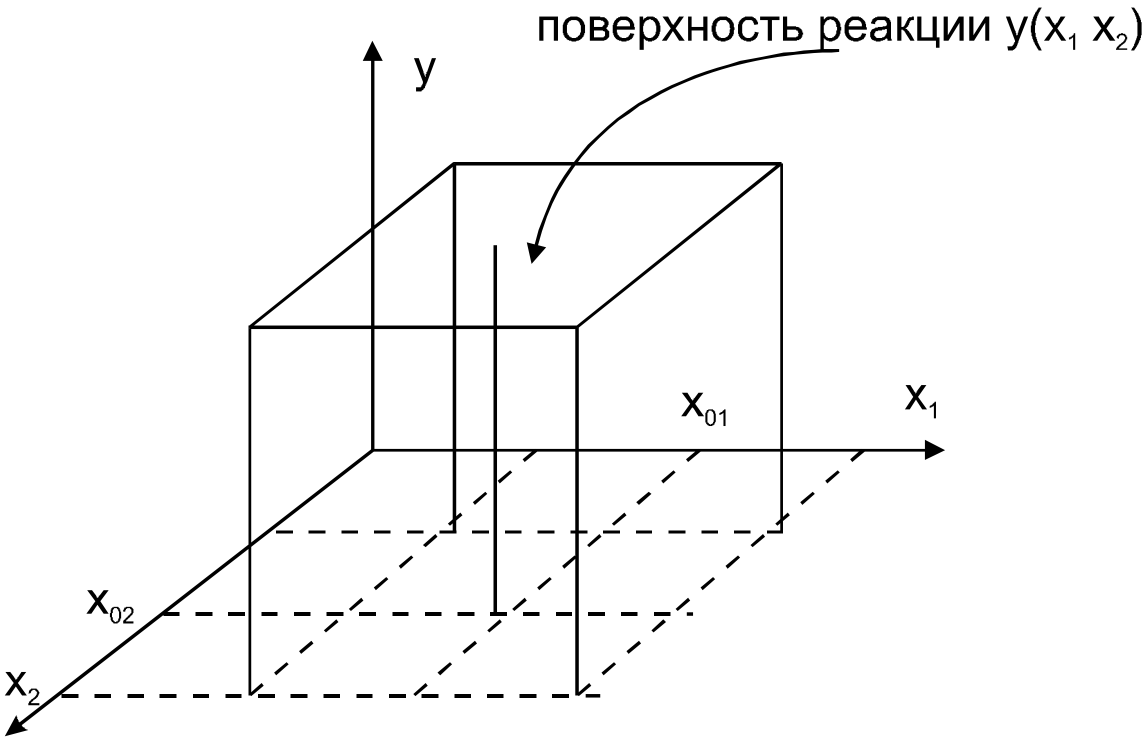


Рис. 1. Геометрическое представление поверхности реакции.

Реакцию (отклик) системы можно представить в виде зависимости: $y_l=\Psi_l(x_1, x_2,\ldots,x_k)$; e=1…m. Функцию Ψ_e, связанную с факторами, называют функцией отклика, а её геометрический образ – поверхностью отклика. Исследователь заранее не известен вид зависимостей Ψ_l, l=1…m, поэтому используют приближение соотношения: $\tilde{y}_l = \Psi_l(x_1, x_2, ..., x_k), l = \overline{1,m}.$

Зависимость и Ψ_l находятся по данным эксперимента. Последний необходимо поставить так, чтобы при минимальных затратах ресурсов (числе испытаний), варьируя выходные значения по специально сформулированным правилам, построить математическую модель системы и оценить её характеристики. Факторы при проведении эксперимента могут быть управляемыми и неуправляемыми, количественными или качественными, фиксированными и случайными. Фактор относится к изучаемым, если он включён в модель для изучения свойств системы. Количественными факторами являются интенсивности входящих потоков заявок, интенсивности потоков обслуживания, ёмкости накопителей, количество обслуживающих каналов и другие. Качественным факторам не соответствует числовая шкала (дисциплины постановки на очередь, обслуживание каналов и другие).

Фактор является управляемым, если его уровни целенаправленно выбираются экспериментатором.

При планировании эксперимента обычно изменяются несколько факторов.

Основными требованиями, предъявляемыми к факторам - независимость и совместимость. Совместимость означает, что все комбинации факторов осуществимы.

Для выбора конкретной модели планирования эксперимента необходимо сформулировать такие её особенности, как адекватность, содержательность, простота.

План эксперимента обычно используется для определения экстремальной характеристики объекта. Поэтому планирование эксперимента называется экстремальным. В планировании эксперимента наибольшее значение нашли модели в виде алгебраических полиномов.

Предполагаем, что изучается влияние К количественных факторов x_i на некоторую η в отведённый для экспериментирования локальной области факторного пространства ограниченного $x_{i\,min}$—$x_{i\,max}$, i=1…k.

Функцию отклика обычно выбирают линейной или квадратичной.

$$\eta = b_0 + \sum b_i x_i + \sum b_{ij} x_i x_j = f(\vec{x})\vec{B} \qquad (1)$$

где $\vec{f}(\vec{x}) =$ вектор с элементами $f_\alpha(\vec{x}), \alpha = \overrightarrow{o,d}$, входящих в исходный полином; $\vec{B}$ - вектор коэффициентов. Для двух факторов имеем: $f_0=1$, $f_1=x_1$, $f_2=x_2$, $f_{12}=x_1x_2$, $f_{11}=x_1^2$, $f_{22}=x_2^2$. $\vec{B} = (b_0,b_1,b_2,b_{12},b_{11},b_{22})$.

Так как полином (1) содержит d коэффициентов, то план эксперимента должен содержать N≥d различных экспериментальных точек:

$$D = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{k1} \\ x_{12} & x_{22} & \cdots & x_{k2} \\ \cdots & \cdots & \cdots & \cdots \\ x_{1N} & x_{2N} & \cdots & x_{kN} \end{bmatrix}$$

где x_{in} - значение, которое принимает i-ая переменная в u-ом испытании. i=1…k, u=1...N. Матрица D называется планом эксперимента.

Реализовав испытания в N очках области факторного пространства, определённом планом эксперимента, получим вектор наблюдений имеющий следующий вид:

$$\bar{y} = \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_N \end{bmatrix}$$

где y_u - реакция соответствующей u-ой точке плана.

Плану эксперимента поставим в соответствие матрицу планирования:

$$x = \begin{bmatrix} f_{01} & f_{11} & f_{21} & \cdots & f_{111} & f_{221} \\ f_{02} & f_{12} & f_{22} & \cdots & f_{112} & f_{222} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ f_{0N} & f_{1N} & f_{2N} & \cdots & f_{11N} & f_{22N} \end{bmatrix}$$

где f_{il}, f_{ijl} - координатные функции при соответствующих коэффициентах модели, в l - ом эксперименте.

Построению плана эксперимента предшествует проведение ряда неформализованных действий (принятия решения) направленных на выбор локальной области факторного пространства G.

Необходимо учитывать, что как только модель сформирована включение дополнительных факторов для уточнения модели невозможно. Вначале следует выбрать границы $x_{i\,min}$ и $x_{i\,max}$ области определения факторов исходя из свойств объекта. Например, температура при термобарических экспериментах не может быть ниже абсолютного нуля и выше температуры плавления материала из которого изготовлена термобарокамера.

После определения области G необходимо найти нулевые (основные) уровни факторов и интервалы варьирования Δx_i, i=1…k.

Эксперимент, в котором реализуются все возможные сочетания уровней факторов, называется <u>полным факторным экспериментом</u> (ПЭФ). Если выбранная модель включает только линейные члены полинома и их произведения, то для оценки коэффициентов модели используется ПЭ с варьированием всех k факторов на двух уровнях, т.е. q=2. Такие планы называются планы типа 2^k, где $n=2^k$- число всех возможных испытаний.

Начальным этапом ПЭ для получения коэффициентов линейной модели основан на варьировании факторов на двух уровнях: нижнем $x_{iн}$ и верхнем $x_{iв}$, симметрично расположенных относительно основного уровня x_{i0}, i=1…k.

Геометрическая интерпретация показана ниже на рис. 2.:

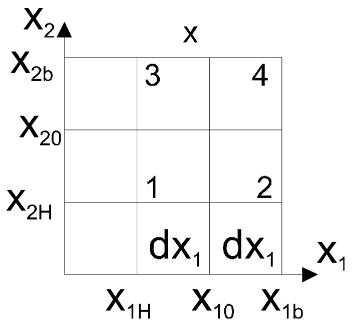


Рис. 2. ПЭФ типа 2^2.

Для упрощения записи условий каждого эксперимента факторы кодируют в виде безразмерных величин $\tilde{x}_i = (x_i - x_{i0}) / \Delta x_i, i = 1,2,...,k$.Средний уровень кодированного фактора является нулём 0, граничные значения соответственно +1 и -1.

Стратегическое планирование машинных экспериментов с моделями систем

Можно выделить стратегическое и тактическое ПЭ на моделях систем.

Стратегическое планирование – ставит своей целью получение необходимой информации о системе S с помощью модели M_M, реализованной на ЭВМ. Оно аналогично внешнему проектированию при создании системы S.

Тактическое планирование – определяет способы проведения каждой серии испытаний машинной модели M_M. Оно аналогично внутреннему проектированию системы S.

Рассмотрим элементы стратегического планирования ПЭ. Его целью может быть:

1. Получение функции реакции системы от независимых фактов: $y=f(b_0,b_1\dots,x_1,x_2,\dots x_k)$
2. Нахождение экстремума: $f(b_0,b_1\dots,x_1,x_2,\dots x_k)$.

Во 2-ом случае для определения наилучшей комбинации фактов могут быть использованы методы систематической или случайной выборки.

К систематическим относятся методы:

- одного фактора;
- предельного анализа;
- наискорейшего спуска;
- равномерной сетки.

Проблемой является большое количество факторов. Для k=10 ПЭФ должен состоять из 1024 точек. Используют неполные планы, метод "поверхности реакции".

Следующей проблемой является многокомпонентность функции реакции. Здесь можно использовать последовательное однокомпонентное ПЭ. Этот подход не всегда возможен из-за связанности компонентов. Используются интегральные оценки с применением весовых функций, функций полезности и т.д.

Другой проблемой является стохастическая сходимость результатов ПЭ. В качестве результатов ПЭ используется средние некоторых распределений, для оценки которых применяют выборочные средние, найденные путём многократны прогонов модели на ЭВМ. Сходимость выборочных средних с ростом объема выборки называется стохастической. Эта сходимость, как правило, медленная. Если σ - стандартное отклонение среднего N наблюдений будет равно $\sigma/\sqrt{N}$, т.е. для уменьшения случайной выборки в k раз требуется увеличить объем выборки в k^2 раз.

Планирование машинного эксперимента представляет собой итерационный процесс, когда выбранная модель плана эксперимента проверяется на реализуемость, а затем, если это необходимо, вносят соответствующие коррективы в модель.

Планирование эксперимента с моделью проводится в несколько этапов:

1) построение структурной модели;
2) построение функциональной модели.

Структурная модель ПЭ характеризуется числом факторов и числом уровней для каждого фактора. Из опыта известно, что 20% факторов определяют 80% свойств системы.

Ортогональное распределение плана упрощает определение коэффициентов аппроксимации. Упрощение дает принятие числа уровней всех факторов одинаковыми (не больше 3). Функциональная модель ПЭ определяет количество элементов структурной модели $N_ф$, т.е. необходимое число различных информационных точек $N_ф$. Причём $N_ф<N_c$, где $N_c=q_1,q_2\dots q_k$ – число экспериментов ПФЭ.

Тактическое планирование машинных экспериментов с моделями систем

Здесь решают проблемы:

– определения начальных условий и их влияния на достижения установившегося результата при моделировании;

– обеспечения точности и достоверности результатов моделирования;

– уменьшения дисперсии оценок характеристик процесса функционирования моделируемых систем;

– выбора правил автоматической остановки имитационного эксперимента с моделями.

Рассмотрим ПФЭ типа 2^3:

номер испытания	1	2	3	4	5	6	7	8
x_1	-1	-1	-1	-1	+1	+1	+1	+1
x_2	-1	-1	+1	+1	-1	-1	+1	+1
x_3	-1	+1	-1	-1	-1	+1	-1	+1

ПЭФ даёт возможность определить не только коэффициенты регрессии, соответствующие линейным эффектам, но и коэффициенты регрессии соответствующие всем эффектам взаимодействия. Эффект взаимодействия двух или более факторов появляется при одновременном варьировании этих факторов, когда действие каждого из них на выход зависит от уровня, на которых находятся другие факторы.

Для оценки свободного члена b_0 и определения эффектов взаимодействия b_{12}, b_{13}, …,b_{123}… план эксперимента D расширяют до матрицы планирования X путём добавления соответствующей фиктивной переменной: единичного столбца x_0 и столбцов произведений $x_1x_2, x_1x_3, ..., x_1x_2x_3$ как показано, например, для ПЭФ типа 2^3 в таблице (см. ниже):

№ Исп.	x_0	План ПЭФ			x_1x_2	x_1x_3	x_2x_3	$x_1x_2x_3$	Реакция y
		x_0	x_2	x_3					
1	+1	+1	+1	+1	+	+	+	+	y_1
2	+1	–1	+1	+1	–	–	+	–	y_2
3	+1	+1	–1	+1	–	+	–	–	y_3
4	+1	–1	–1	+1	+	–	–	+	y_4
5	+1	+1	+1	–1	+	–	–	–	y_5
6	+1	–1	+1	–1	–	+	–	+	y_6
7	+1	+1	–1	–1	–	–	+	+	y_7
8	+1	–1	–1	–1	+	+	+	–	y_8

Как видно из рассмотренных ПЭ типа 2^2 b 2^3 количество испытаний ПЭФ значительно превосходит число определяемых коэффициентов линейной модели плана эксперимента, что увеличивает расход ресурсов ЭВМ по времени. Возникает проблема сокращения количества экспериментов.

С этой целью рассмотрим построение планов так называемого дробного факторного эксперимента (ДФЭ). Пусть имеется ПЭФ типа 2^2. Используя матрицу планирования X, например приведённую в предыдущей таблице, можно вычислить коэффициенты и предусмотреть результаты в виде уравнения:

$$y = b_0 + b_1x_1 + b_2x_2 + b_{12}x_1x_2$$

N/n	x_0	План ПЭФ		$x_1x_2(x_0)$	Отклик y
		x_1	x_2		
1	1	+1	+1	+1	y_1
2	1	–1	+1	–1	y_2
3	1	+1	–1	–1	y_3
4	1	–1	–1	+1	y_4

Если в выбранных интервалах варьирования уровня процесс можно описать линейной моделью, то достаточно определить три коэффициента b_0, b_1,b_2. Т.о. остаётся одна степень свободы, которую можно использовать для построения плана эксперимента D для 3-х переменных, в которых уровни 3-

его фактора изменяются как в таблице рассмотренной немного раньше для столбца x_1x_2 (эффектов взаимодействия).

Мы получим так называемый дробный факторный эксперимент. В нём уже не будет раздельных оценок для коэффициентов регрессии, как а ПЭФ, они будут рассчитываться по формулам:

$$Б_1 = \beta_1 + \beta_{23}; b_2 = \beta_2 + \beta_{13}; b_3 = \beta_3 + \beta_{12}.$$

При постулировании линейной модели все парные взаимодействия не учитывают. Т.о. вместо γ испытанный в ПЭФ для 3-х факторов получим 4 испытания в ДФЭ. Правило проведения ДФЭ формулируется так: для сокращения числа испытаний новому фактору присваивается значение вектор-столбца матрицы, принадлежащего взаимодействию, которым можно пренебречь.

При проведении эксперимента из 4-х испытаний для оценки влияния 3-х факторов пользуются половинный ПЭФ типа 2^3, так называемой "полу репликой". Если приравнять x_3 и x_1x_2, что можно получить 2-ую "полу реплику".

Для обозначения дробных реплик, в которых λ линейных эффектов приравнены к эффектам взаимодействия, пользуются условным обозначением $2^{k-\lambda}$. Например, "полу реплика" от 2^6 записывается в виде 2^{6-1}, а "четверть реплика" 2^{6-2}.

Смешивание надо производить так, чтобы основные коэффициенты были смешаны с коэффициентами при взаимодействиях самого высокого порядка. Не рекомендуется использовать реплики для $n \geq 15$.

Следует иметь в виду, что малый шаг варьирования Δx; $(j=1\ldots n)$ может повлечь статистическую не значимость оценки коэффициента уравнения регрессии. В случае, если полученная мат. модель окажется неадекватной, проводятся эксперименты с меньшим шагом варьирования.

Если линейные модели, построенные с помощью ПЭФ и ДФЭ, неадекватны, то переходят к построению квадратичных моделей.

Оптимальный план для квадратичной модели целесообразно строить таким образом, чтобы он включал точки плана для линейной модели. Это позволяет сократить число опытов.

Список литературы