

1. Общие вопросы моделирования

1.1 Предмет теории моделирования.

Моделирование - это замещение одного объекта (оригинала) другим (моделью) и фиксация и изучение свойств модели. Замещение производится с целью упрощения, удешевления, ускорения изучения свойств оригинала.

В общем случае объектом-оригиналом может быть естественная или искусственная, реальная или воображаемая система. Она имеет множество параметров S_0 и характеризуется определёнными свойствами. Количественной мерой свойств системы служит множество характеристик Y_0 , система проявляет свои свойства под влиянием внешних воздействий X .

Множество параметров S и их значений отражает её внутреннее содержание - структуру и принципы функционирования. Характеристики S - это в основном её внешние признаки, которые важны при взаимодействии с другими S .

Характеристики S находятся в функциональной зависимости от её параметров. Каждая характеристика системы $y_0 \in Y_0$ определяется в основном ограниченным числом параметров $\{S_{0k}\} \subset S_0$. Остальные параметры не влияют на значение данной характеристики S . Исследователя интересуют, как правило, только некоторые характеристики S : $\{y\} \subset Y_0$ при конкретных воздействиях на систему $\{x_{mn}\} \subset X$.

Модель — это тоже система со своими множествами параметров S_m и характеристик Y_m . Оригинал и модель сходны по одним параметрам и различны по другим. Замещение одного объекта другим равносильно, если интересующие исследователя характеристики оригинала и модели определяются однотипными подмножествами параметров и связаны одинаковыми зависимостями с этими параметрами:

$$y_{0k} = f(\{S_{0i}\}, \{x_{0n}\}, T); \quad (1.1)$$

$$y_{mn} = f(\{S_{mi}\}, \{x_{mn}\}, T_m) \quad (1.2)$$

где, y_{mn} - k -ая характеристика модели, $y_{mn} \in Y_m$

x_{mn} - внешнее воздействие на модель, $x_{mn} \in X$

T_m - модельное время.

При этом $S_{0i} = \Psi(S_{mi})$; $x_{0n} = \omega(x_{mn})$, $T = mT_m$ (где m - масштабный коэффициент) на всём интервале $[0-T_m]$ или в отдельные периоды времени. Тогда с некоторым приближением можно сделать вывод о том, что характеристики O_p , связаны с характеристиками M зависимостями $y_{0k} = \varphi(y_{mk})$. Множество характеристик модели $Y_{mk} = \{y_{mk}\}$ является отображением множества интересующих характеристик оригинала $y_{0k} = \{y_{0k}\}$, т.е. $\varphi: Y_{0k} \rightarrow y_{mk}$, т.е. $\varphi: Y_{0k} \rightarrow Y_{mk}$.

При исследовании сложных естественных S , у которых известны Y_{0k} , но мало изучен состав элементов и принципы их взаимодействия с помощью моделирования может решаться обратная задача. Строят предположительную модель, определяющую её характеристики Y_{mk} при эквивалентных внешних воздействиях $\{x_{mn}\}$ ($\omega: \{x_{0n}\} \rightarrow \{x_{mn}\}$) и, если оказывается, что имеет место отображение $\varphi: Y_{0k} \rightarrow Y_{mk}$ с некоторой известной функцией φ , то считается, что система-оригинал имеет такие же параметры.

Моделирование целесообразно, когда у модели отсутствуют те признаки оригинала, которые препятствуют его исследованию.

Теория моделирования — взаимосвязанная совокупность положений, определений, методов и средств создания моделей. Сами модели являются предметом теории моделирования.

Теория моделирования является основной составляющей общей теории систем - системологии, где в качестве главного принципа постулируются осуществимые модели: система представима конечным множеством моделей, каждая из которых отражает определённую грань её сущности.

1.2 Роль и место моделирования в исследовании систем.

Познание любой системы (S) сводится по существу к созданию её модели. Перед изготовлением каждого устройства или сооружения разрабатывается его модель - проект. Любое произведение искусства является моделью, фиксирующее действительность.

Достижения математики привели к распространению математических моделей различных объектов и процессов. Подмечено, что динамика функционирования разных по физической природе систем однотипными зависимостями, что позволяет моделировать их на ЭВМ.

На качественно новую ступень поднялась моделирование в результате разработки методологии имитационного моделирования на ЭВМ.

Сейчас трудно указать область человеческой деятельности, где бы применялось моделирование. Разработаны модели производства автомобилей, выращивания пшеницы, функционирования отдельных органов человека, жизнедеятельности Азовского моря, атомного взрыва, последствий атомной войны.

Специалисты считают, что моделирование становится основной функцией ВС. На практике широко используются АСУ технологическими процессами организационно-экономическими комплексами, процессами проектирования, банки данных и знаний. Но любая из этих систем нуждается в информации об управляемом объекте и модели управляемого объекта, в моделировании тех или иных управляющих решений.

Сами ВС как сложные и дорогостоящие технические системы могут являться объектами моделирования.

Обычно процесс разработки сложной системы осуществляется итерационно с использованием моделирования проектных решений. Если характеристики не удовлетворяют предъявленным требованиям, то по результатам анализа производят корректировку проекта, затем снова проводят моделирование.

При анализе действующих систем с помощью моделирования определяют границы работоспособности системы, выполняют имитацию экспериментальных условий, которые могут возникнуть в процессе функционирования системы. Искусственное создание таких условий на действительной системе затруднено и может привести к катастрофическим последствиям.

Применение моделирования может быть полезным при разработке стратегии развития ВС, её усовершенствования при создании сетей ЭВМ.

1.3 Классификация моделей.

Физические модели. В основу классификации положена степень абстрагирования модели от оригинала. Предварительно все модели можно подразделить на 2 группы — физические и абстрактные (математические).

Ф.М. обычно называют систему, эквивалентную или подобную оригиналу, но возможно имеющую другую физическую природу. Виды Ф.М.:

- натуральные;
- квазинатуральные;
- масштабные;
- аналоговые;

Натуральные модели — это реальные исследуемые системы (макеты, опытные образцы). Имеют полную адекватность (соответствия) с системой оригиналом, но дороги.

Квазинатуральные модели — совокупность натуральных и математических моделей. Этот вид используется тогда, когда модель части системы не может быть математической из-за сложности её описания (модель человека оператора) или когда часть системы должна быть исследована во взаимодействии с другими частями, но их ещё не существует или их включение очень дорого. (вычислительные полигоны, АСУ)

Масштабная модель — это система той же физической природы, что и оригинал, но отличается от него масштабами. Методологической основой масштабного моделирования является теория подобия. При проектировании ВС масштабные модели могут использоваться для анализа вариантов компоновочных решений.

Аналоговые модели называют системы, имеющие физическую природу, отличающуюся от оригинала, но сходные с оригиналом процессы функционирования. Для создания аналоговой модели требуется наличие математического описания изучаемой системы. В качестве аналоговых моделей используются механические, гидравлические, пневматические и электрические системы. Аналоговое моделирование используется при исследовании средств ВТ на уровне логических элементов и электрических цепей, а так же на системном уровне, когда функционирование системы описывается например, дифференциальными или алгебраическими уравнениями.

Математические модели. Математические модели представляют собой формализованное представление системы с помощью абстрактного языка, с помощью математических соотношений, отражающих процесс функционирования системы. Для составления математических моделей можно использовать любые математические средства — алгебраическое, дифференциальное, интегральное исчисления, теорию множеств, теорию алгоритмов и т.д. По существу вся математика создана для составления и исследования моделей объектов и процессов.

К средствам абстрактного описания систем относятся также языки химических формул, схем, чертежей, карт, диаграмм и т.п. Выбор вида модели определяется особенностями изучаемой системы и целями моделирования, т.к. исследование модели позволяет получить ответы на определённую группу вопросов. Для получения другой информации может потребоваться модель другого вида. Математическая модель

можно классифицировать на детерминированные и вероятностные, аналитические, численные и имитационные.

Аналитической моделью называется такое формализованное описание системы, которое позволяет получить решение уравнения (1.2) в явном виде, используя известный математический аппарат.

Численная модель характеризуется зависимостью (1.2) такого вида, который допускает только частные решения для конкретных начальных условий и количественных параметров моделей.

Имитационная модель — это совокупность описания системы и внешних воздействий, алгоритмов функционирования системы или правил изменения состояния системы под влиянием внешних и внутренних возмущений. Эти алгоритмы и правила не дают возможности использования имеющихся математических методов аналитического и численного решения, но позволяют имитировать процесс функционирования системы и производить вычисления интересующих характеристик. Имитационные модели могут быть созданы для гораздо более широкого класса объектов и процессов, чем аналитические и численные. Поскольку для реализации имитационных моделей служат ВС, средствами формализованного описания ИМ служат универсальные и специальные алгоритмические языки. ИМ в наибольшей степени подходят для исследования ВС на системном уровне.

2. Математические схемы моделирования систем.

2.1 Основные подходы к построению ММ систем.

Исходной информацией при построении ММ процессов функционирования систем служат данные о назначении и условиях работы исследуемой (проектируемой) системы S . Эта информация определяет основную цель моделирования, требования к ММ, уровень абстрагирования, выбор математической схемы моделирования.

Понятие **математическая схема** позволяет рассматривать математику не как метод расчёта, а как метод мышления, средства формулирования понятий, что является наиболее важным при переходе от словесного описания к формализованному представлению процесса её функционирования в виде некоторой ММ.

При пользовании мат. схемой в первую очередь исследователя системы должен интересовать вопрос об адекватности отображения в виде конкретных схем реальных процессов в исследуемой системе, а не возможность получения ответа (результата решения) на конкретный вопрос исследования.

Например, представление процесса функционирования ИВС коллективного пользования в виде сети схем массового обслуживания даёт возможность хорошо описать процессы, происходящие в системе, но при сложных законах входящих потоков и потоков обслуживания не даёт возможности получения результатов в явном виде.

Математическую схему можно определить как звено при переходе от содержательного к формализованному описанию процесса функционирования системы с учётом воздействия внешней среды. Т.е. имеет место цепочка: описательная модель — математическая схема — имитационная модель.

Каждая конкретная система S характеризуется набором свойств, под которыми понимаются величины, отображающие поведение моделируемого объекта (реальной системы) и учитываются условия её функционирования во взаимодействии с внешней средой (системой) E .

При построении ММ системы S необходимо решить вопрос о её полноте. Полнота моделирования регулируется, в основном, выбором границ "Система S — среда E ". Также должна быть решена задача упрощения ММ, которая помогает выделить основные свойства системы, отбросив второстепенные в плане цели моделирования.

ММ объекта моделирования, т.е. системы S можно представить в виде множества величин, описывающих процесс функционирования реальной системы и образующих в общем случае следующие подмножества:

- совокупность X - входных воздействий на S $x_i \in X, i=1 \dots n_x$;
- совокупность воздействий внешней среды $v_l \in V, l=1 \dots n_v$;
- совокупность внутренних (собственных) параметров системы $h_k \in H, k=1 \dots n_h$;
- совокупность выходных характеристик системы $y_j \in Y, j=1 \dots n_y$.

В перечисленных множествах можно выделить управляемые и неуправляемые величины. В общем случае X, V, H, Y не пересекаемые множества, содержат как детерминированные так и стохастические составляющие. Входные воздействия E и внутренние параметры S являются независимыми (экзогенными) переменными, $\bar{X}(t); \bar{V}(t); \bar{H}(t)$. Выходные характеристики - зависимые переменные (эндогенные) $\bar{Y}(t)$. Процесс функционирования S описывается оператором F_S :

$$\bar{Y}(t) = F_S(\bar{X}, \bar{V}, \bar{H}, t) \quad (1)$$

$\bar{Y}(t)$ - выходная траектория. F_S - закон функционирования S. F_S может быть функция, функционал, логические условия, алгоритм, таблица или словесное описание правил.

Алгоритм функционирования A_S — метод получения выходных характеристик $\bar{Y}(t)$ с учётом входных воздействий $\bar{X}(t); \bar{V}(t); \bar{H}(t)$. Очевидно один и тот же F_S может быть реализован различными способами, т.е. с помощью множества различных A_S .

Соотношение (1) является математическим описанием поведения объекта S моделирования во времени t, т.е. отражает его динамические свойства. (1) - это динамическая модель системы S. Для статических условий ММ есть отображения X, V, H в Y , т.е. $\bar{Y} = f(\bar{X}, \bar{V}, \bar{H})$ (2)

Соотношения (1), (2) могут быть заданы формулами, таблицами и т.д.

Также соотношения в ряде случаев могут быть получены через свойства системы в конкретные моменты времени, называемые состояниями.

Состояния системы S характеризуются векторами:

$$\bar{Z}^I(\bar{z}_1^I, \dots, \bar{z}_k^I) \text{ и } \bar{Z}^{II}(\bar{z}_1^{II}, \dots, \bar{z}_k^{II}), \text{ где } \bar{z}_1^I = z_1(t^I) \dots \bar{z}_k^I = z_k(t^I) \text{ в момент } t^I \in (t_0, T)$$

$$\bar{z}_1^{II} = z_1(t^{II}) \dots \bar{z}_k^{II} = z_k(t^{II}) \text{ в момент } t^{II} \in (t_0, T) \text{ и т.д. } k=1 \dots n_Z.$$

$Z_1(t), Z_2(t) \dots Z_k(t)$ - это координаты точки в k-мерном фазовом пространстве. Каждой реализации процесса будет соответствовать некоторая фазовая траектория.

Совокупность всех возможных значений состояний $\{\bar{Z}\}$ называется пространством состояний объекта моделирования Z , причём $z_k \in Z$.

Состояние системы S в интервале времени $t_0 < t \leq T$ полностью определяется начальными условиями $\bar{Z}^0 = (\bar{z}_1^0, \dots, \bar{z}_k^0)$, где $\bar{z}_1^0 = z_1(t_0) \dots$ входными $\bar{X}(t)$, внутренними параметрами $\bar{H}(t)$ и воздействиями внешней среды $\bar{V}(t)$, которые имели место за промежуток времени $t^* - t_0$ с помощью 2-х векторных уравнений:

$$\bar{Z}(t) = \Phi(\bar{Z}^0, \bar{X}, \bar{V}, \bar{h}, t); \quad (3)$$

$$\bar{Y}(t) = F(\bar{Z}, t). \quad (4)$$

$$\text{иначе: } Y(t) = F(\Phi(\bar{Z}^0, \bar{X}, \bar{V}, \bar{h}, t)). \quad (5)$$

Время в мод. S может рассматриваться на интервале моделирования (t_0, T) как непрер., так и дискретное, т.е. квантованное на отрезке длин. Δt .

Таким образом под ММ объекта понимаем конечное множество переменных $\{\bar{X}, \bar{Z}, \bar{h}\}$ вместе с математическими связями между ними и характеристиками \bar{Y} .

Моделирование называется детерминированным, если операторы F, Φ детерминированные, т.е. для конкретного входа выход детерминированный. Детерминированное моделирование - частный случай стохастического моделирования. В практике моделирование объектов в области системного анализа на первичных этапах исследования рациональнее использовать типовые математические схемы: диф. уравнения, конечные и вероятностные автоматы, СМО и т.д.

Не облад. такой степенью общности, как модели (3), (4), типовые математические схемы имеют преимущество простоты и наглядности, но при существенном сужении возможности применения.

В качестве детерминированных моделей, когда при исследовании случайный факт не учитывается, для представления систем, функционирующих в непрерывном времени, используются дифференциальные, интегральные и др. уравнения, а для представления систем, функционирующих в дискретном времени — конечные автоматы и конечно разностные схемы.

В начале стохастических моделей (при учёте случайного фактора) для представления систем с дискретным временем используются вероятностные автоматы, а для представления систем с непрерывным временем — системы массового обслуживания (СМО). Большое практическое значение при исследовании сложных индивидуальных управленческих систем, к которым относятся АСУ, имеют так называемые агрегативные модели.

Агрегативные модели (системы) позволяют описать широкий круг объектов исследования с отображением системного характера этих объектов. Именно при агрегативном описании сложный объект расчленяется на конечное число частей (подсистем), сохраняя при этом связи, обеспечивая взаимодействие частей.

2.2 Непрерывно детерминированные модели (Д - схемы).

Рассмотрим особенности непрерывно детерминированного подхода на примере, используя в качестве ММ дифференциальные уравнения.

Дифференциальными уравнениями называются такие уравнения, в которых неизвестными будут функции одной переменной или нескольких переменных, причём в уравнение входят не только их функции но их производные различных порядков.

Если неизвестные - функции многих переменных, то уравнения называются — уравнения в частных производных. Если неизвестные функции одной независимой переменной, то имеют место обыкновенные дифференциальные уравнения.

Математическое соотношение для детерминированных систем в общем виде:

$$\dot{\bar{y}} = \bar{f}(\bar{y}, t), \bar{y}(t_0) = \bar{y}_0 \quad (7).$$

Например, процесс малых колебаний маятника описан обыкновенным дифференциальным уравне-

нием $m_1 l_1^2 \frac{\partial^2 \Theta(t)}{\partial t^2} + m_2 g l \Theta(t) = 0$ где m_1, l_1 - масса, длина подвески маятника, Θ - угол отклонения маятника от положения равновесия. Из этого уравнения можно найти оценки интересующих характеристик, например период колебаний $T = 2\pi\sqrt{l/g}$

Диф. уравнения, Д - схемы являются математическим аппаратом теории систем автоматического регулирования, управления.

При проектировании и эксплуатации систем САУ необходимо выбрать такие параметры системы, которые бы обеспечивали требуемую точность управления.

Следует отметить, что часто используемые в САУ системы диф. уравнений определяются путём линеаризацией управления объекта (системы), более сложного вида, имеющего нелинейности:

$$F(y^n, y^{n-1}, \dots, y, x^m, x^{m-1}, \dots, x_n) = 0:$$

$$\frac{dF}{dy^n} \Delta y^n + \frac{dF}{dy^{n-1}} \Delta y^{n-1} \dots \frac{dF}{dy_0} \Delta y + \Delta y = \frac{dF}{dx_0^m} \Delta x^m + \dots \frac{dF}{dx_0} \Delta x + \Delta x$$

2.3 Дискретно – детерминированные модели (F-схемы)

ДДМ являются предметом рассмотрения теории автоматов (ТА). ТА - раздел теоретической кибернетики, изучающей устройства, перерабатывающие дискретную информацию и меняющего свои внутренние состояния лишь в допустимые моменты времени.

Конечный автомат имеет множество внутренних состояний и входных сигналов, являющихся конечными множествами. Автомат задаётся F- схемой: $F = \langle z, x, y, \varphi, \psi, z_0 \rangle$, (1)

где z, x, y - соответственно конечные множества входных, выходных сигналов (алфавитов) и конечное множество внутренних состояний (алфавита). $z_0 \in Z$ - начальное состояние; $\varphi(z, x)$ - функция переходов; $\psi(z, x)$ - функция выхода. Автомат функционирует в дискретном автоматном времени, моментами которого являются такты, т.е. примыкающие друг к другу равные интервалы времени, каждому из которых соответствуют постоянные значения входного, выходного сигнала и внутреннего состояния. Абстрактный автомат имеет один входной и один выходной каналы.

В момент t , будучи в состоянии $z(t)$, автомат способен воспринять сигнал $x(t)$ и выдать сигнал $y(t) = \psi[z(t), x(t)]$, переходя в состояние $z(t+1) = \varphi[z(t), x(t)]$, $z(t) \in Z$; $y(t) \in Y$; $x(t) \in X$. Абстрактный КА в начальном состоянии z_0 принимая сигналы $x(0), x(1), x(2) \dots$ выдаёт сигналы $y(0), y(1), y(2) \dots$ (выходное слово).

Существуют F- автомат 1-ого рода (Миля), функционирующий по схеме:

$$z(t+1) = \varphi[z(t), x(t)], t=0, 1, 2, \dots \quad (1)$$

$$y(t) = \psi[z(t), x(t)], t=0, 1, 2, \dots \quad (2)$$

F- автомат 2-ого рода:

$$z(t+1) = \varphi[z(t), x(t)], t=0, 1, 2, \dots \quad (3)$$

$$y(t) = \psi[z(t), x(t-1)], t=1, 2, 3, \dots \quad (4)$$

$$\text{Автомат 2-ого рода, для которого } y(t) = \psi[z(t)], t=0, 1, 2, \dots \quad (5)$$

т.е. функция выходов не зависит от входной переменной $x(t)$, называется автоматом Мура.

Т.о. уравнения 1-5 полностью задающие F- автомат, являются частным случаем уравнения

$$\dot{z}(t) = \Phi(\bar{z}_0, \bar{x}, \bar{v}, \bar{h}, t) \quad (6)$$

где \vec{z} - вектор состояния, \vec{x} - вектор независимых входных переменных, \vec{v} - вектор воздействий внешней среды, \vec{h} - вектор собственных внутренних параметров системы, \vec{z}_0 - вектор начального состояния, t - время; и уравнение $\vec{y}(t) = F(\vec{z}, t)$, (7)

когда система S - денормированная и на её вход поступает дискретный сигнал x .

По числу состояний конечные автоматы бывают с памятью и без памяти. Автоматы с памятью имеют более одного состояния, а автоматы без памяти (комбинационные или логические схемы) обладают лишь одним состоянием. При этом согласно (2), работа комбинационной схемы заключается в том, что она ставит в соответствие каждому входному сигналу $x(t)$ определённый выходной сигнал $y(t)$, т.е. реализует логическую функцию вида:

$$y(t) = \psi[x(t)], t=0, 1, 2, \dots$$

Эта функция называется булевой, если алфавиты X и Y , которым принадлежат значения сигналов x и y состоят из 2-х букв.

По характеру отсчёта времени (дискретному) F - автоматы делятся на синхронные и асинхронные. В синхронных автоматах моменты времени, в которые автомат "считывает" входные сигналы, определяются принудительно синхронизирующими сигналами. Реакция автомата на каждое значение входного сигнала заканчивается за один такт синхронизации. Асинхронный F - автомат считывает входной сигнал непрерывно и поэтому, реагируя на достаточно длинный входной сигнал постоянной величины x , он может, как это следует из 1-5, несколько раз изменить своё состояние, выдавая соответствующее число выходных сигналов, пока не перейдёт в устойчивое.

Для задания F - автомата необходимо описать все элементы множества $F = \langle z, x, y, \varphi, \psi, z_0 \rangle$, т.е. входной, внутренний и выходной алфавиты, а также функции переходов и выходов. Для задания работы F - автоматов наиболее часто используются табличный, графический и матричный способ.

В табличном способе задания используется таблицы переходов и выходов, строки которых соответствуют входным сигналам автомата, а столбцы - его состояниям. При этом обычно 1-ый столбец слева соответствует начальному состоянию z_0 . На пересечении i -ой строки и j -ого столбца таблицы переходов помещается соответствующее значение $\varphi(z_k, x_i)$ функции переходов, а в таблице выходов - $\psi(z_k, x_i)$ функции выходов. Для F - автомата Мура обе таблицы можно совместить, получив т.н. отмеченную таблицу переходов, в которой над каждым состоянием z_k автомата, обозначающим столбец таблицы, стоит соответствующий этому состоянию, согласно (5), выходной сигнал $\psi(z_i)$.

Описание работы F - автомата Мили таблицами переходов φ и выходов ψ иллюстрируется таблицей (1), а описание F - автомата Мура - таблицей переходов (2).

Таблица 1

x_j	z_k			
	z_0	z_1	...	z_k
Переходы				
x_1	$\varphi(z_0, x_1)$	$\varphi(z_1, x_1)$...	$\varphi(z_k, x_1)$
x_2	$\varphi(z_0, x_2)$	$\varphi(z_1, x_2)$...	$\varphi(z_k, x_2)$
.....				
x_l
Выходы				
x_1	$\psi(z_0, x_1)$	$\psi(z_1, x_1)$...	$\psi(z_k, x_1)$
.....				
x_l	$\psi(z_0, x_l)$	$\psi(z_1, x_l)$...	$\psi(z_k, x_l)$

Таблица 2

x_i	$\psi(z_k)$			
	$\psi(z_0)$	$\psi(z_1)$...	$\psi(z_k)$
	z_0	z_1	...	z_k
x_1	$\varphi(z_0, x_1)$	$\varphi(z_1, x_1)$...	$\varphi(z_k, x_1)$
x_2	$\varphi(z_0, x_2)$	$\varphi(z_1, x_2)$...	$\varphi(z_k, x_2)$
.....				
x_l	$\varphi(z_0, x_l)$	$\varphi(z_1, x_l)$...	$\varphi(z_k, x_l)$

Примеры табличного способа задания F - автомата Мили $F1$ с тремя состояниями, двумя входными и двумя выходными сигналами приведены в таблице 3, а для F - автомата Мура $F2$ - в таблице 4.

Таблица 3

x_j	z_0		
	z_0	z_1	z_2
Переходы			
x_1	z_2	z_0	z_0
x_2	z_0	z_2	z_1
Выходы			
x_1	y_1	y_1	y_2
x_2	y_1	y_2	y_1

Таблица 4

x_i	y				
	y_1	y_1	y_3	y_2	y_3
	z_0	z_1	z_2	z_3	z_4
x_1	z_1	z_4	z_4	z_2	z_2
x_2	z_3	z_1	z_1	z_0	z_0

При другом способе задания конечного автомата используется понятие направленного графа. Граф автомата представляет собой набор вершин, соответствующих различным состояниям автомата и соединяющих вершин дуг графа, соответствующих тем или иным переходам автомата. Если входной сигнал x_k вызывает переход из состояния z_i в состояние z_j , то на графе автомата дуга, соединяющая вершину z_i с вершиной z_j обозначается x_k . Для того, чтобы задать функцию переходов, дуги графа необходимо отметить соответствующими выходными сигналами. Для автоматов Мили эта разметка производится так: если входной сигнал x_k действует на состояние z_i , то согласно сказанному получается дуга, исходящая из z_i и помеченная x_k ; эту дугу дополнительно отмечают выходным сигналом $y=\psi(z_i, x_k)$. Для автомата Мура аналогичная разметка графа такова: если входной сигнал x_k , действуя на некоторое состояние автомата, вызывает переход в состояние z_j , то дугу, направленную в z_j и помеченную x_k , дополнительно отмечают выходным сигналом $y=\psi(z_j, x_k)$. На рис. 1 приведены заданные ранее таблицами F- автоматы Мили F1 и Мура F2 соответственно.

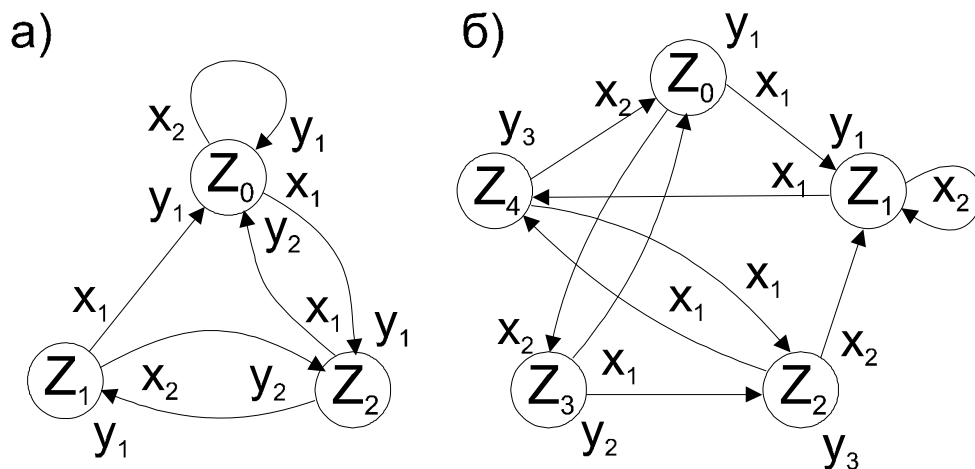


Рис. 1. Графы автоматов Мили (а) и Мура (б).

При решении задач моделирования часто более удобной формой является матричное задание конечного автомата. При этом матрица соединений автомата есть квадратная матрица $C=||c_{ij}||$, строки которой соответствуют исходным состояниям, а столбцы - состояниям перехода. Элемент $c_{ij}=x_k/y_s$ в случае автомата Мили соответствует входному сигналу x_k , вызывающему переход из состояния z_i в состояние z_j и выходному сигналу y_s , выдаваемому при этом переходе. Для автомата Мили F1, рассмотренного выше, матрица соединений имеет вид:

$$C_1 = \begin{pmatrix} x_2/y_1 & - & x_1/y_1 \\ x_1/y_1 & - & x_2/y_2 \\ x_1/y_2 & x_2/y_1 & - \end{pmatrix}$$

Если переход из состояния z_i в состояние z_j происходит под действием нескольких сигналов, элемент матрицы c_{ij} представляет собой множество пар "вход/выход" для этого перехода, соединённых знаком дизъюнкции.

Для F- автомата Мура элемент c_{ij} равен множеству входных сигналов на переходе $(z_i z_j)$, а выход описывается вектором выходов:

$$\bar{y} = \begin{bmatrix} \Psi(z_0) \\ \Psi(z_1) \\ \dots \\ \Psi(z_k) \end{bmatrix} \quad i\text{-ая компонента которого выходной сигнал, отмечающий состояние } z_i$$

Пример. Для рассмотренного ранее автомата Мура F2 запишем матрицу состояний и вектор выходов:

$$C_2 = \begin{pmatrix} - & x_1 & - & x_2 & - \\ - & x_2 & - & - & x_1 \\ - & x_2 & - & - & x_1 \\ x_2 & - & x_1 & - & - \\ x_2 & - & x_1 & - & - \end{pmatrix}; \quad \bar{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_2 \\ y_3 \end{bmatrix}$$

Для детерминированных автоматов переходы однозначны. Применительно к графическому способу задания F- автомата это означает, что в графе F- автомата из любой вершины не могут выходить 2 и более ребра, отмеченные одним и тем же входным сигналом. Аналогично этому в матрице соединений автомата С в каждой строке любой входной сигнал не должен встречаться более одного раза.

Рассмотрим вид таблицы переходов и графа асинхронного конечного автомата. Для F- автомата состояние z_k называется устойчивым, если для любого входа $x_i \in X$, для которого $\phi(z_k, x_i) = z_k$ имеет место $\psi(z_k, x_i) = y_k$. Т.о. F- автомат называется асинхронным, если каждое его состояние $z_k \in Z$ устойчиво.

На практике всегда автоматы являются асинхронными, а устойчивость их состояний обеспечивается тем или иным способом, например, введением сигналов синхронизации. На уровне абстрактной теории удобно часто оперировать с синхронными конечными автоматами.

Пример. Рассмотрим асинхронный F- автомат Мура, который описан в табл. 5 и приведён на рис. 2.

Таблица 5

x_i	y		
	y_1	y_2	y_3
	z_0	z_1	z_2
x_1	z_1	z_1	z_1
x_2	z_2	z_1	z_2
x_3	z_0	z_0	z_2

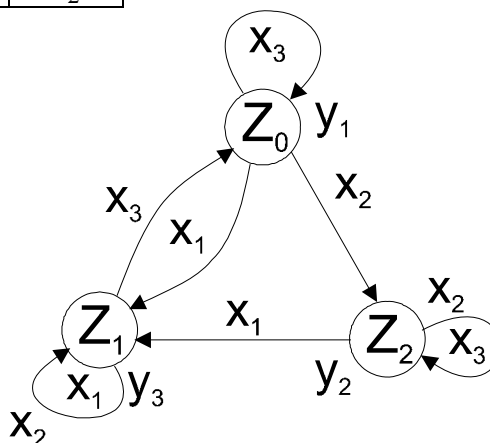


Рис. 2.Граф асинхронного автомата Мура.

Если в таблице переходов асинхронного автомата некоторое состояние z_k стоит на пересечении строки x_s и столбца $z_s (S \neq k)$, то это состояние z_k обязательно должно встретиться в этой же строке в столбце z_k .

С помощью F-схем описываются узлы и элементы ЭВМ, устройства контроля, регулирования и управления, системы временной и пространственной коммутации в технике обмена информацией. Широта применения F-схем не означает их универсальность. Этот подход непригоден для описания процессов при-

нятия решений, процессов в динамических системах с наличием переходных процессов и стохастических элементов.

3. Непрерывно-стохастические модели (Q - схемы).

К ним относятся системы массового обслуживания (англ. queuing system), которые называют Q-схемами.

3.1 Методы теории массового обслуживания.

Предмет ТМО — системы массового обслуживания (СМО) и сети массового обслуживания. Под СМО понимают динамическую систему, предназначенную для эффективного обслуживания случайного потока заявок при ограниченных ресурсах системы. Обобщённая структура СМО приведена на рисунке 3.1.

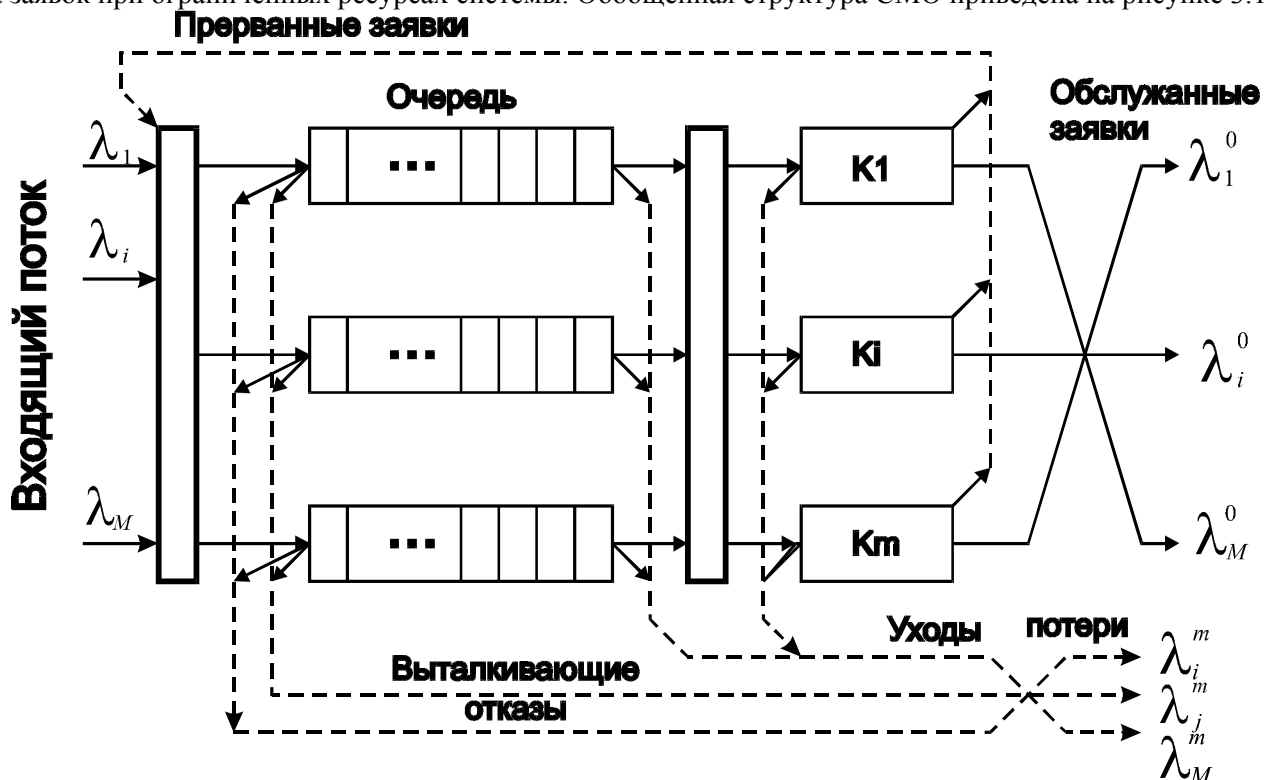


Рис. 3.1. Схема СМО.

Поступающие на вход СМО однородные заявки в зависимости от порождающей причины делятся на типы, интенсивность потока заявок типа i ($i=1 \dots M$) обозначено λ_i . Совокупность заявок всех типов - входящий поток СМО.

Обслуживание заявок выполняется m каналами. Различают универсальные и специализированные каналы обслуживания. Для универсального канала типа j считается известными функции распределения $F_{ji}(\tau)$ длительности обслуживания заявок произвольного типа. Для специализированных каналов функции распределения длительности обслуживания каналов заявок некоторых типов являются неопределёнными, назначение этих заявок на данный канал.

В качестве процесса обслуживания могут быть представлены различные по своей физической природе процессы функционирования экономических, производственных, технических и других систем, например, потоки поставок продукции некоторому предприятию, потоки деталей и комплектующих изделий на сборочном конвейере цеха, заявки на обработку информации ЭВМ от удалённых терминалов и т.д. При этом характерным для работы таких объектов является случайное поведение заявок (требований) на обслуживание и завершение обслуживания в случайные моменты времени.

Q - схемы можно исследовать аналитически и имитационными моделями. Последнее обеспечивает большую универсальность.

Рассмотрим понятие массового обслуживания.

В любом элементарном акте обслуживания можно выделить две основные составляющие: ожидание обслуживания заявкой и собственно обслуживание заявки. Это можно отобразить в виде некоторого i -ого прибора обслуживания Π_i , состоящего из накопителя заявок, в котором может находиться одновременно $l_i=0 \dots L_i^H$ заявок, где L_i^H - ёмкость i -ого накопителя, и канала обслуживания заявок, k_i .

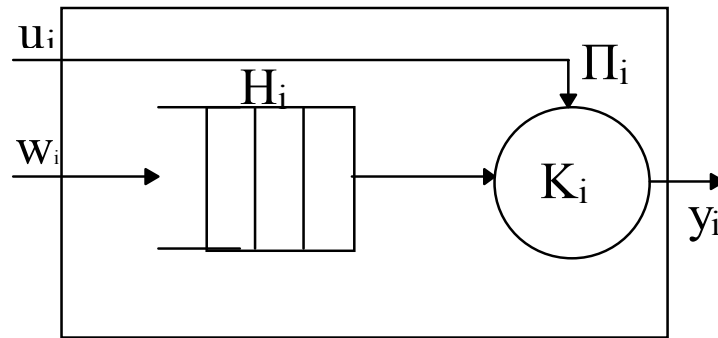


Рис. 3.2. Схема прибора СМО

На каждый элемент прибора обслуживания Π_i поступают потоки событий: в накопитель H_i поток заявок w_i , на канал k_i - поток обслуживания u_i .

Потоком событий (ПС) называется последовательность событий, происходящих одно за другим в какие-то случайные моменты времени. Различают потоки однородных и неоднородных событий. Однородный ПС характеризуется только моментами поступления этих событий (вызывающими моментами) и задается последовательностью $\{t_n\} = \{0 \leq t_1 \leq t_2 \leq \dots \leq t_n \leq \dots\}$, где t_n - момент поступления n -ого события - неотрицательное вещественное число. ОПС может быть также задан в виде последовательности промежутков времени между n -ым и $n-1$ -ым событиями $\{\tau_n\}$.

Неоднородным ПС называется последовательность $\{t_n, f_n\}$, где t_n - вызывающие моменты; f_n - набор признаков события. Например, может быть задана принадлежность к тому или иному источнику заявок, наличие приоритета, возможность обслуживания тем или иным типом канала и т.п.

Рассмотрим ОПС, для которого $\tau_i \in \{\tau_n\}$ - случайные величины, независимые между собой. Тогда ПС называется потоком с ограниченным последствием.

ПС называется ординарным, если вероятность того, что на малый интервал времени Δt , примыкающий к моменту времени t попадает больше одного события $P_{\geq 1}(t, \Delta t)$ пренебрежительно мала.

Если для любого интервала Δt событие $P_0(t, \Delta t) + P_1(t, \Delta t) + P_{\geq 1}(t, \Delta t) = 1$, $P_1(t, \Delta t)$ - вероятность попадания на интервал Δt ровно одного события. Как сумма вероятностей событий, образующих полную группу и несовместных, то для ординарного потока событий $P_0(t, \Delta t) + P_1(t, \Delta t) \approx 1$, $P_{\geq 1}(t, \Delta t) = \Theta(\Delta t)$, где $\Theta(\Delta t)$ - величина, порядок малости который выше, чем Δt , т.е. $\lim_{\Delta t \rightarrow 0} \Theta(\Delta t) = 0$ при $\Delta t \rightarrow 0$.

Стационарным ПС называется поток, для которого вероятность появления того или иного числа событий на интервале времени τ зависит от длины этого участка и не зависит от того, где на оси времени $0 - t$ взят этот участок. Для ОПС справедливо $0 * P_0(t, \Delta t) + 1 * P_1(t, \Delta t) = P_1(t, \Delta t)$ - среднее число событий на интервале Δt . Среднее число событий, наступающих на участке Δt в единицу времени составляет $P_1(t, \Delta t) / \Delta t$. Рассмотрим предел этого выражения при $\Delta t \rightarrow 0$

$$\lim_{\Delta t \rightarrow 0} P_1(t, \Delta t) / \Delta t = \lambda(t) * (1/\text{един.вр.}).$$

Если этот предел существует, то он называется интенсивностью (плотностью) ОПС. Для стандартного ПС $\lambda(t) = \lambda = \text{const}$.

Применительно к элементарному каналу обслуживания k_i можно считать что поток заявок $w_i \in W$, т.е. интервалы времени между моментами появления заявок на входе k_i образуют подмножество неуправляемых переменных, а поток обслуживания $u_i \in U$, т.е. интервалы времени между началом и окончанием обслуживания заявки образуют подмножество управляемых переменных.

Заявки, обслуженные каналом k_i и заявки, покинувшие прибор Π_i по различным причинам не обслуживаемыми, образуют выходной поток $y_i \in Y$.

Процесс функционирования прибора обслуживания Π_i можно представить как процесс изменения состояний его элементов во времени $Z_i(t)$. Переход в новое состояние для Π_i означает изменение кол-ва заявок, которые в нём находятся (в канале k_i и накопителе H_i). Т.о. вектор состояний для Π_i имеет вид :

$\vec{Z}_i = (z_i^H, z_i^K)$, где z_i^H - состояния накопителя, ($z_i^H = 0$ - накопитель пуст, $z_i^H = 1$ - в накопителе одна заявка..., $z_i^H = L_i^H$ - накопитель занят полностью; z_i^K - состояние канала k_i ($z_i^K = 0$ - канал свободен, $z_i^K = 1$ канал занят).

Q-схемы реальных объектов образуются композицией многих элементарных приборов обслуживания Π_i . Если k_i различных приборов обслуживания соединены параллельно, то имеет место многоканальное обслуживание (многоканальная Q-схема), а если приборы Π_i и их параллельные композиции соединены последовательно, то имеет место многофазное обслуживание (многофазная Q-схема).

Т.о. для задания Q-схемы необходимо оператор сопряжения R, отражающий взаимосвязь элементов структуры.

Связи в Q-схеме изображают в виде стрелок (линий потока, отражающих направление движения заявок). Различают разомкнутые и замкнутые Q-схемы. В разомкнутой выходной поток не может снова поступить на какой-либо элемент, т.е. обратная связь отсутствует.

Собственными (внутренними) параметрами Q-схемы будут являться кол-во фаз L_Φ , количество каналов в каждой фазе, L_{kj} , $j=1 \dots L_\Phi$, количество накопителей каждой фазы L_{kj} , $k=1 \dots L_\Phi$, ёмкость i-ого накопителя L_i^H . Следует отметить, что в теории массового обслуживания в зависимости от ёмкости накопителя применяют следующую терминологию:

- системы с потерями ($L_i^H=0$, накопитель отсутствует);
- системы с ожиданием ($L_i^H \rightarrow \infty$);
- системы с ограниченной ёмкостью накопителя N_i (смешанные).

Обозначим всю совокупность собственных параметров Q-схемы как подмножество H.

Для задания Q-схемы также необходимо описать алгоритмы её функционирования, которые определяют правила поведения заявок в различных неоднозначных ситуациях.

В зависимости от места возникновения таких ситуаций различают алгоритмы (дисциплины) ожидания заявок в накопителе N_i и обслуживания заявок каналом k_i . Неоднородность потока заявок учитывается с помощью введения класса приоритетов.

В зависимости от динамики приоритетов Q-схемы различают статические и динамические. Статические приоритеты назначаются заранее и не зависят от состояний Q-схемы, т.е. они являются фиксированными в пределах решения конкретной задачи моделирования. Динамические приоритеты возникают при моделировании. Исходя из правил выбора заявок из накопитель N_i на обслуживание каналом k_i можно выделить относительные и абсолютные приоритеты. Относительный приоритет означает, что заявка с более высоким приоритетом, поступившая в накопитель H, ожидает окончания обслуживания представляющей заявки каналом k_i и только после этого занимает канал. Абсолютный приоритет означает, что заявка с более высоким приоритетом, поступившая в накопитель, прерывает обслуживание каналом k_i заявки с более низким приоритетом и сама занимает канал (при этом вытесненная из k_i заявка может либо покинуть систему, либо может быть снова записана на какое-то место в N_i).

Необходимо также знать набор правил, по которым заявки покидают N_i и k_i : для N_i – либо правила переполнения, либо правила ухода, связанные с истечением времени ожидания заявки в N_i ; для k_i – правила выбора маршрутов или направлений ухода. Кроме того, для заявок необходимо задать правила, по которым они остаются в канале k_i , т.е. правила блокировок канала. При этом различают блокировки k_i по выходу и по входу. Такие блокировки отражают наличие управляющих связей в Q-схеме, регулирующих поток заявок в зависимости от состояний Q-схемы. Набор возможных алгоритмов поведения заявок в Q-схеме можно представить в виде некоторого оператора алгоритмов поведения заявок A.

Т.о. Q-схема, описывающая процесс функционирования СМО любой сложности однозначно задаётся в виде набора множеств: $Q = \langle W, U, H, Z, R, A \rangle$.

4. Имитационное моделирование систем.

4.1 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 получении 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 находиться в работоспособ-

ном 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: $y=f(\alpha, \beta, \dots, \omega)$ где $\alpha, \beta, \dots, \omega$ случайные величины с известными функциями распределения.

Для решения задач такого вида применяется следующий алгоритм:

- 1) по каждой из величин $\alpha, \beta, \dots, \omega$ производится случайное испытание, в результате каждого определяется некоторое конкретное значение случайной величины $\alpha_i, \beta_i, \dots, \omega_i$;
- 2) используя найденные величины, определяется одно частное значение 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.

4.2 Имитация функционирования системы.

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

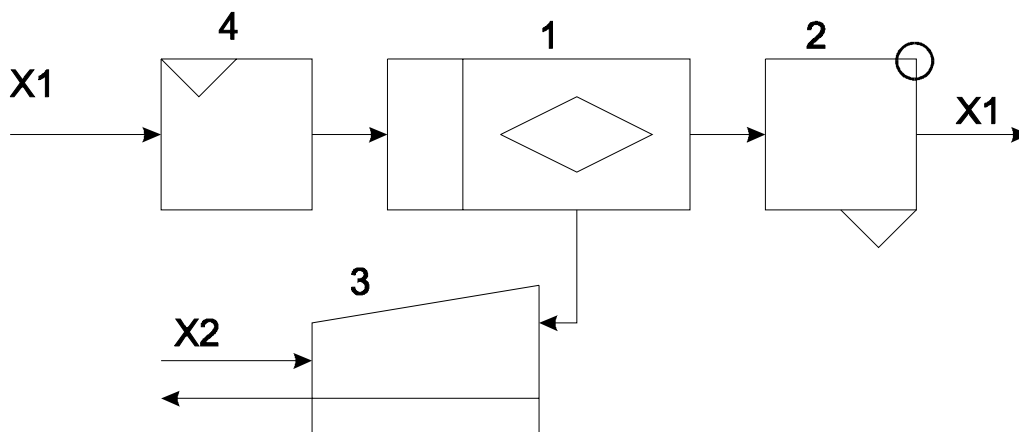


Рис. 4.1. Упрощённая схема моделируемой системы.

Через устройство 4 поступает поток заданий $X1$. 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 $X2$, которые обрабатываются процессором и ответы выводятся на экран дисплея. Процессор работает в 2-х программном режиме: в одном разделе обрабатываются задания $X1$, в другом, с более высоким относительным приоритетом запросы $X2$. Представим данную ВС в упрощённом варианте в виде стохастической сети из 4-х СМО. Потоки заданий и запросы будем называть потоками заявок. Считаем потоки $X1$ и $X2$ независимыми. Известны ф.р. периодов следования заявок τ_1 и τ_2 и длительность обслуживания T_{1K} , T_{2K} заявок в k -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 $X1$ по результатам случайного испытания в соответствии с ф.р. периода следования заявок.

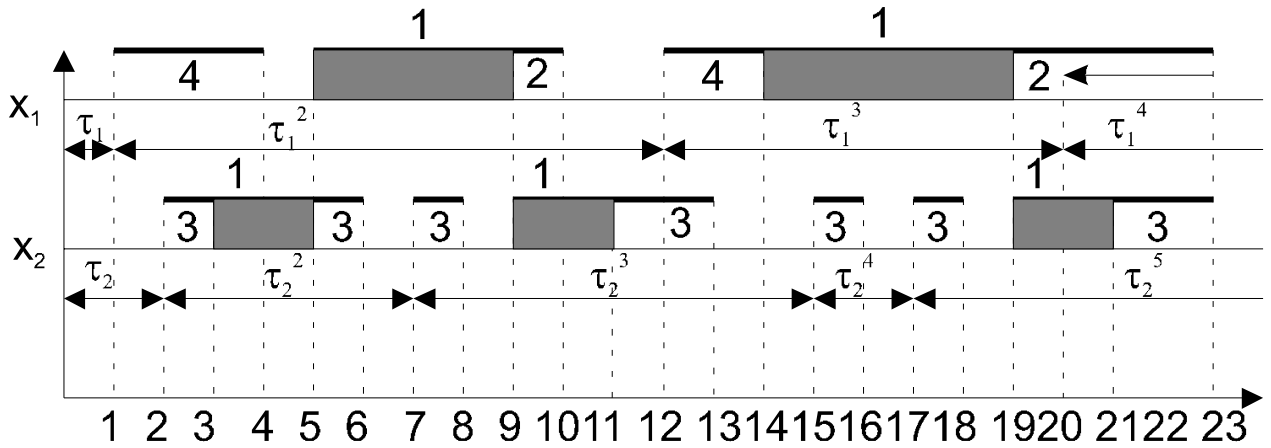


Рис. 4.2. Временная диаграмма функционирования ВС.

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 X1 определяется время обслуживания устройством ввода перфокарт T_{14}^1 методом случайного испытания и отмечается момент окончания обслуживания $t_4=t_1+T_{14}^1$. 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_{23}^1 и отслеживается время окончания обслуживания $t_3=t_2+T_{23}^1$. 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_{21}^1 и отмечается момент $t_5=t_3+T_{21}^1$ 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, становится 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_{11}^1 и T_{23}^1 и отмечаются моменты времени $t_9=t_5+T_{11}^1$ и $t_6=t_5+T_{23}^1$. 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 $u_{12}^1=t_6-t_2$. 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_{23}^2 и отмечается момент $t_8=t_7+T_{23}^2$, 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_{21}^2 и T_{12}^1 по результатам случайных испытаний и отмечаются моменты окончания обслуживания $t_{11}=t_9+T_{21}^2$ и $t_{10}=t_9+T_{12}^1$. 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 $u_{11}^1=t_{10}-t_1$.

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 надёжные 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;

5. Обобщённые алгоритмы имитационного моделирования.

5.1 Алгоритм моделирования по принципу особых состояний.

Оно использовалось в приведённом выше примере. В качестве событий выделены:

- поступление заявки в систему;
- освобождение элемента после обслуживания заявки;
- завершения моделирования;

- | | | |
|---------------------------------------|---|------------------------|
| * возникновение отказа устройств | } | другие типы
событий |
| * завершение восстановления устройств | | |

Процесс имитации развивался с использованием управляющих последовательностей, определяемых по функциям распределения вероятностей исходных данных путём проведения случайных испытаний. В качестве управляющих последовательностей использовались в примере последовательности значений периодов следования заявок по каждому i -ому потоку $\{\tau_i\}$ и длительности обслуживания заявок i -ого потока устройством $\{T_{ik}\}$. Моменты наступления будущих событий определялись по простым рекуррентным соотношениям. Эта особенность даёт возможность построить простой циклический алгоритм моделирования, который сводится к следующим действиям:

- 1) определяется событие с минимальным временем — наиболее раннее событие;
- 2) модельному времени присваивается значение времени наступления наиболее раннего события;
- 3) определяется тип события;
- 4) в зависимости от типа события предпринимаются действия, направленные на загрузку устройств и продвижение заявок в соответствии с алгоритмом их обработки, и вычисляются моменты наступления будущих событий; эти действия называют реакцией модели на события;
- 5) перечисленные действия повторяются до истечения времени моделирования.

В процессе моделирования производится измерение и статистическая обработка значений выходных характеристик. Обобщённая схема алгоритма моделирования по принципу особых состояний приведена на рисунке 5.1.

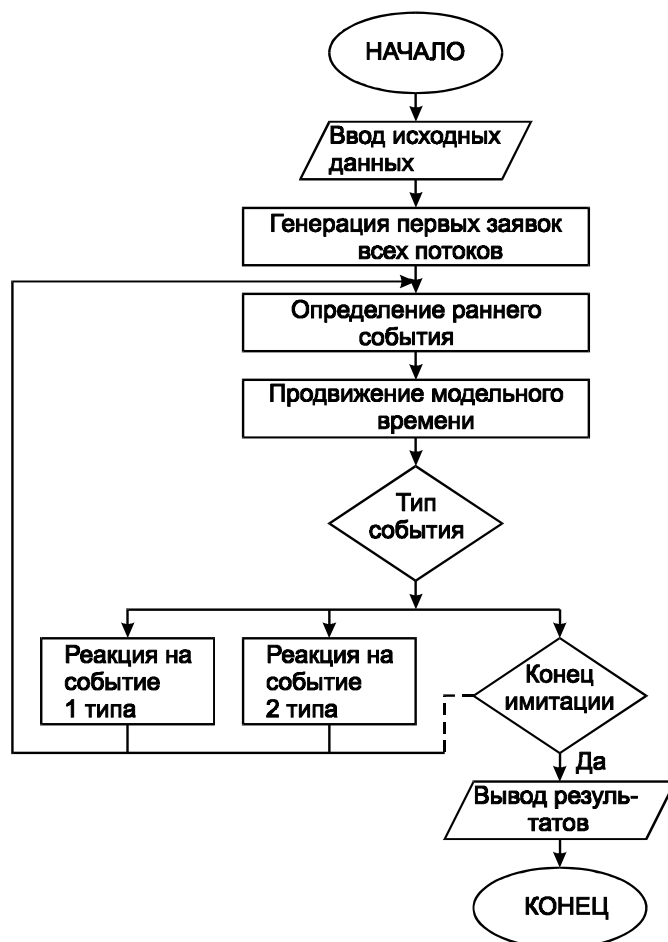


Рис. 5.1. Обобщённый алгоритм моделирования систем по принципу особых состояний

5.2 Алгоритм моделирования по принципу Δt .

Укрупнённая схема моделирующего алгоритма, который реализует принцип постоянного приращения модельного времени (принципа Δt), представлен на следующем рисунке:

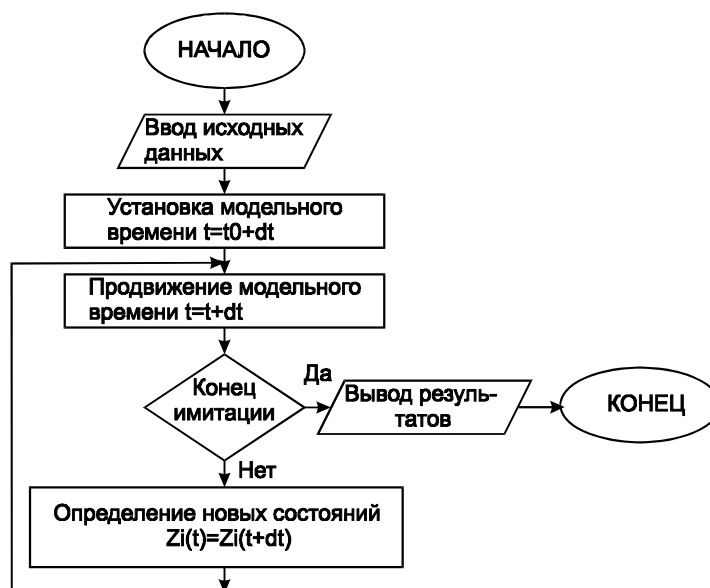


Рис. 5.2. Обобщённый алгоритм моделирования систем по принципу приращений " Δt "

В начале инициализируется программа, в частности вводятся значения $Z_i(t_0)$, $i=1,2,\dots,k$. Которые характеризуют состояние системы в k -мерном фазовом пространстве состояний в начальный момент времени t_0 . Модельное время устанавливается $t = t_0 = 0$. Основные операции по имитации системы осуществляется в

цикле. Функционирование системы отслеживается по последовательной схеме состояний $Z_i(t)$. Для этого модельному даётся некоторое приращение dt . Затем по вектору текущих состояний определяются новые состояния $Z_i(t + dt)$, которые становятся текущими. Для определения новых состояний по текущим в формализованном описании системы должны существовать необходимые математические зависимости. По ходу имитации измеряются, вычисляются, фиксируются необходимые выходные характеристики. При моделировании стохастических систем вместо новых состояний вычисляются распределения вероятностей для возможных состояний. Конкретные значения вектора текущих состояний определяются по результатам случайных испытаний. В результате проведения имитационного эксперимента получается одна из возможных реализаций случайного многомерного процесса в заданном интервале времени (t_0, T_k).

Моделирующий алгоритм, основанный на применении dt применим для более широкого круга систем, чем алгоритм, построенный по принципу особых состояний. Однако при его реализации возникают проблемы определения величины dt . Для моделирования ВС на системном уровне в основном используется принцип особых состояний.

6. Методы определения характеристик моделируемых систем.

6.1 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 и число обслуженным устройством заявок, время простоя устройства в результате отказов и количество отказов, возникших в процессе моделирования, длины очередей и занимаемые ёмкости памяти.

При статистическом моделировании большая часть характеристик — это случайные величины. По каждой такой характеристике y 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:

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

V_k - среднее время обслуживания одной заявки k -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.

6.2 Расчёт математического ожидания и дисперсии выходной характеристики.

В случае стационарного эргодического процесса функционирования системы вычисление $M(y)$ и $D(y)$ выходной характеристики y производится усреднением не по времени, а по множеству $N_{\text{знач.}}$ измеренных по одной реализации достаточной длительности. В целях экономии ОЗУ ЭВМ $M(y)$ и $D(y)$ вычисляются по рекуррентным формулам:

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

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

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

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

При большом количестве измерений эти оценки являются состоятельными и несмещёнными.

6.3 Расчёт среднего по времени значения выходной характеристики.

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

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

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.

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

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

6.4 Построение гистограммы для стационарной системы.

Γ - эмпирическая плотность распределения вероятностей. Задаются границы изменения интересующей характеристики. $y_i \rightarrow [y_n; y_b]$, числом интервалов N_g . Определяется ширина интервала $\Delta = (y_n - y_b) / N_g$.

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:

$$S_i = \sum_1^{N_g} G_i \cdot \Delta = \sum \frac{R_i}{N \cdot \Delta} \cdot \Delta = \sum \frac{R_i}{N} = 1, \text{ т.к. } N = \sum_1^{N_g} R_i$$

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

$$\chi^2 = \frac{\sum_1^{N_g} (R_i - N * P_i)^2}{N * P_i} \quad (6);$$

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

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

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

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

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

Функция распределения χ^2 табулирована. По вычисленному значению χ^2 и числу степеней свободы с помощью таблиц определяется вероятность $P(\chi^2 < Z)$. Если она превышает заданный уровень значимости C , то выдвинутая гипотеза принимается.

7. Моделирование случайных воздействий.

Важной задачей в практике имитационного моделирования систем на ЭВМ является расчёт случайных величин. В языках программирования существуют датчики равномерно распределённых псевдослучайных величин в интервале $\{0,1\}$. Остановимся на вопросах преобразования последовательности псевдослучайных величин $\{X_i\}$ в последовательности $\{Y_i\}$ с заданным законом распределения и моделировании различных случайных событий.

7.1 Рассмотрим особенности моделирования случайных событий.

Пусть имеются случайные числа x_i , т.е. возможные значения случайной величины ξ , равномерно распределённой в интервале $\{0,1\}$. Необходимо реализовать случайное событие A , наступающее с заданной

вероятностью P . Определим A как событие, состоящее в том, что выбранное значение x_i удовлетворяет неравенству:

$$x_i \leq P \quad (1)$$

Тогда вероятность события A будет: $P(A) = \int_0^P dx = P$. Противоположное событию A состоит в том,

что $x_i > P$. Тогда $P(\bar{A}) = 1 - P$. Процедура моделирования состоит в этом случае в выборе значений x_i и сравнение их с P . При этом, если условие (1) удовлетворяется, то исходом испытания будет событие A .

Таким же образом можно рассмотреть группу событий. Пусть A_1, A_2, \dots, A_n – полная группа событий, наступающая с вероятностями P_1, P_2, \dots, P_n соответственно. Определим A_m как событие, состоящее в том, что выбранное значение x_i случайной величины ξ удовлетворяет неравенству:

$$l_{m-1} < x_i \leq l_m, \text{ где } l_k = \sum_{i=1}^k P_i. \quad (2)$$

Тогда $P(A_m) = \int_{l_{m-1}}^{l_m} dx = P_m$. Процедура моделирования испытаний в этом случае состоит в последо-

вательности сравнений случайных чисел x_i со значениями l_k . Исходом испытания оказывается событие A_m , если выполняется условие (2). Эту процедуру называют определением исхода по жребию в соответствии с вероятностями P_1, P_2, \dots, P_n .

При моделировании систем часто необходимо осуществить такие испытания, при которых искомый результат является сложным событием, зависящим от 2-х и более простых.

Пусть например, независимые события A и B имеют вероятности наступления P_A и P_B . Возможными исходами совместных испытаний в этом случае будут события $AA, \bar{A}B, A\bar{B}, \bar{A}\bar{B}$ с вероятностями $P_AP_B, (1-P_A)P_B, P_A(1-P_B), (1-P_A)(1-P_B)$. Для моделирования совместных испытаний можно использовать последовательную проверку условия (1). Он требует двух чисел x_i .

Рассмотрим случай, когда события A и B являются зависимыми и наступают с вероятностями P_A и P_B . Обозначим через $P(B/A)$ условную вероятность события B при условии, что событие A произошло. Считаем, что $P(B/A)$ задана. Из последовательности случайных чисел $\{X_i\}$ извлекается определённое число x_m и проверяется справедливость неравенства $x_m < P_A$. Если это неравенство справедливо, то наступило событие A . Для испытания, связанного с событием B используется вероятность $P(B/A)$. Из совокупности чисел $\{X_i\}$ берётся очередное число x_{m+1} и проверяется условие $x_{m+1} \leq P(B/A)$. В зависимости от того выполняется или нет это неравенство, исходом испытания является AB или $\bar{A}\bar{B}$. Если неравенство $x_m < P_A$ не выполняется, то наступило событие \bar{A} . Поэтому для испытания, связанного с событием B необходимо определить вероятность:

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

Выберем из совокупности $\{X_i\}$ число x_{m+1} и проверим справедливость неравенства $x_{m+1} \leq P(B / \bar{A})$. В зависимости от того, выполняется оно или нет, получаем исходы испытания $\bar{A}B$. Алгоритм вычислений можно представить в виде схемы, которая изображена на рисунке 7.1.

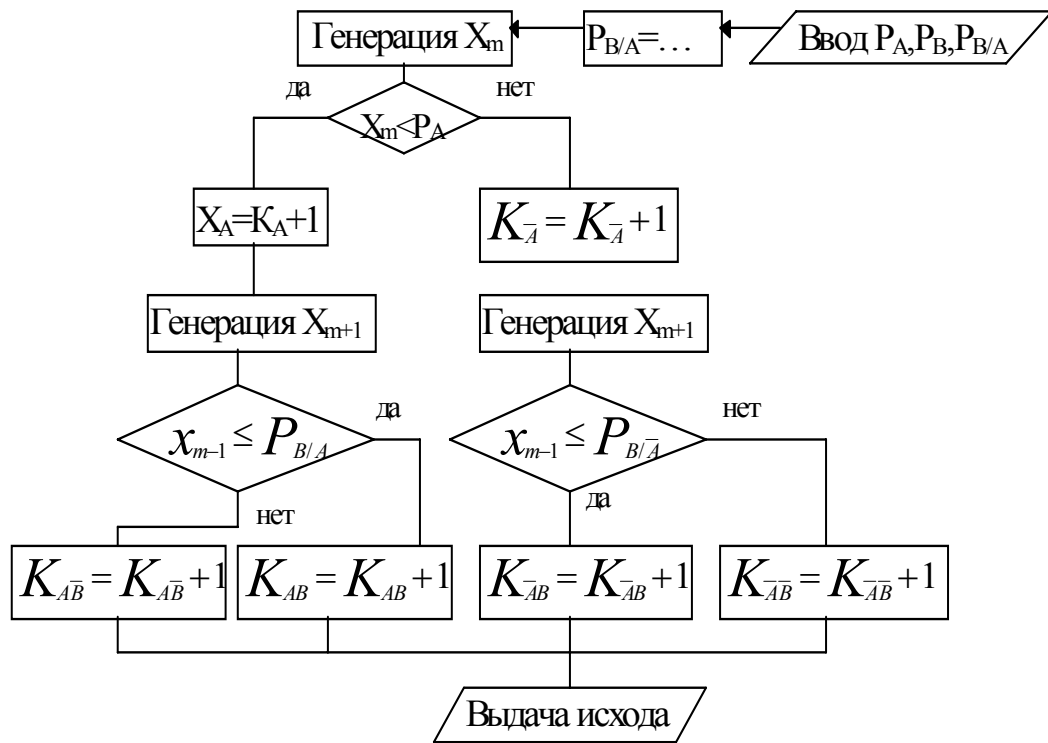


Рис.7.1. Схема моделирования группы случайных событий

7.2 Преобразование случайных величин.

Дискретная случайная величина η принимает значения $y_1 \leq y_2 \leq y_3 \dots y_l$ с вероятностями P_1, P_2, \dots, P_l составляющими дифференциальное распределение вероятностей:

$$\begin{array}{ccc} y & y_1 & y_2 & \dots & y_j & \dots \\ P(\eta=y) & P_1 & P_2 & \dots & P_j & \dots \end{array} \quad (3)$$

При этом интегральная функция распределения $F_\eta(y) = P(\eta \leq y) = \sum_{j=1}^m P_j$; $y_m \leq y_{m+1}$; $m=1, 2, \dots$

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

Для получения дискретных случайных величин можно использовать метод обратной функции. Если ξ - равномерно распределённая на интервале $(0, 1)$, случайная величина η получается с помощью преобразования

$$\eta = F_\eta^{-1}(\xi), \quad \text{где } F_\eta^{-1} - \text{функция, обратная } F_\eta. \quad (5)$$

Алгоритм вычисления по (4) и (5) сводится к выполнению следующих действий:

$$\begin{array}{l} \text{если } x_1 < P_1 \text{ то } \eta = y_1 \text{ иначе,} \\ \text{если } x_2 < P_1 + P_2 \text{ то } \eta = y_2 \text{ иначе,} \\ \dots \dots \dots \end{array} \quad (6)$$

$$\text{если } x_j < \sum_{j=1}^m P_j \text{ то } \eta = y_m \text{ иначе}$$

При счёте по (6) среднее число циклов сравнения равняется $\sum_{j=1}^{\infty} j P_j$

Пример 1. Необходимо методом обратной функции на основании базовой последовательности случайных чисел $\{x_i\}$, равномерно распределённых в интервале $(0, 1)$, получить последовательность чисел $\{y_i\}$, имеющих биномиальное распределение, задающее вероятность y удачных исходов в N реализациях некоторого эксперимента:

$$P(\tau_i = y) = P_N(y) = C_N^y P^y (1-P)^{N-y}, \quad \text{где } P=0.5 \text{ и } N=6; C_N^y = N! / (y!(N-y)!)$$

Математическое ожидание и дисперсия биномиального распределения соответственно будут $M[y] = np(1-p)$. Используя для P_j обозначения, принятые в (6), вычислим:

j	...	1	2	3	4	5	6	7
y _j	...	0	1	2	3	4	5	6

P_j	...	0.01562	0.09375	0.23438	0.3125	0.23438	0.09375	0.01562
$\sum_{j=1}^m P_j$...	0.01562	0.10937	0.34375	0.65625	0.89063	0.98438	1.0000

Например, получив из равномерного распределения число $X_i=0.89063$ и проведя сравнения по алгоритму (6), найдём, что $0.85393 < 0.89063$, т.е. $y_i=4$. При этом среднее число циклов сравнения $\bar{\mu}=1*0.01562+2*0.09375+3*0.23438+4*0.31250+5*0.23438+6*(0.09375+0.01562)\approx 3.98$.

7.3 Вычисление непрерывных случайных величин.

Непрерывная случайная величина η задана интегральной функцией распределения:

$$F_{\eta}(y) = P(\eta \leq y) = \int_{-\infty}^y f_{\eta}(y) dy, \text{ где } f_{\eta}(y) - \text{плотность вероятностей.}$$

Для получения непрерывных случайных величин с заданным законом распределения, как и для дискретных величин можно использовать метод обратной функции. Взаимно однозначная монотонная функция $\eta=F_{\eta}^{-1}(\xi)$, полученная решением относительно η уравнения $F_{\eta}(y)=\xi$ преобразует равномерно распределённую на интервале $(0,1)$ величину ξ в η с требуемой плотностью $f_{\eta}(y)$.

Действительно, если случайная величина η имеет плотность распределения $f_{\eta}(y)$, то распределение случайной величины $\xi = \int_0^{\eta} f_{\eta}(y) dy$ является равномерным.

Т.о. чтобы получить число, принадлежащее последовательность случайных чисел $\{y\}$, имеющих функцию плотности $f_{\eta}(y)$, необходимо разделить относительно y_i уравнение

$$\int_{-\infty}^{y_i} f_{\eta}(y) dy = x_j \quad (7)$$

Пример 2. Необходимо получить случайные числа y_i с показательным законом распределения.

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

В силу соотношения (7) получим $\lambda \int_0^{y_i} e^{-\lambda y} dy = x_i$, где x_i - случайное число, имеющее равномерное распределение в интервале $(0,1)$, тогда $1 - e^{-\lambda y_i} = x_j, y_j = -\ln(1 - x_j) / \lambda$.

Рассмотрим универсальный метод моделирования непрерывных случайных величин (**метод исключения**).

При моделировании случайной величины y с плотностью распределения вероятностей $f_{\eta}(y)$ в интервале $a \leq y \leq b$ независимые значения x_m и x_{m+1} преобразуются в значения

$$y_{1m} = a + (b-a) * x_m \quad (8)$$

$$z_{1m+1} = f_{1\eta}(y) * x_{m+1} \quad (9)$$

где $f_{1\eta}(y) = \max |f_{\eta}(y)|$. При этом y_{1m} и z_{1m+1} - значения случайных величин, равномерно распределённых на интервале (a,b) и $(0, f_{1\eta})$. Эти значения можно рассматривать как абсциссы и ординаты случайных точек, равномерно распределяющихся внутри прямоугольника со сторонами $b-a$ и $f_{1\eta}$, охватывающего кривую распределения $f_{\eta}(y)$ (см. рисунок 7.2.).

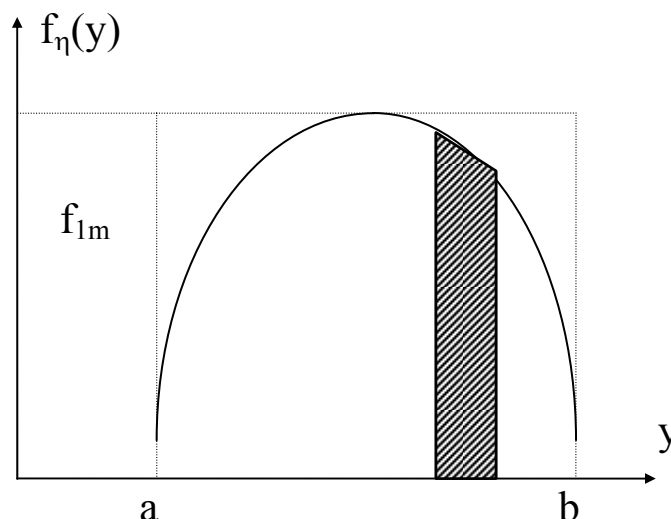


Рис. 7.2. Иллюстрация метода исключений

$$\text{Если } z_{1m+1} \leq f_{\eta}(y_{1m}), \quad (10)$$

тогда пара y_{1m}, z_{1m+1} определяет случайную точку под кривой f_{η} . Вероятность попадания случайной точки, удовлетворяющей условию (10) под кривую f_{η} равна единице, а вероятность попадания в заштрихованную элементарную площадку равна $f_{\eta}(y_{1m}) \cdot \Delta y_{1m}$. Это обозначает, что абсциссы y_{1m} случайных точек, попадающих под кривую f_{η} - значения случайной величины y с заданной плотностью вероятности $f_{\eta}(y)$. Моделируемый алгоритм состоит из функций : 1) получения x_m и x_{m+1} от датчика; 2) расчёта y_{1v} и z_{1m+1} согласно (8) и (9); 3) вычисления $f_{\eta}(y_{1m})$; 4) сравнения z_{1m+1} с $f_{\eta}(y_{1m})$. Если условия (10) выполняются, то $y=y_{1m}$; если нет, то значения y_{1m} и z_{1m+1} исключаются и процесс повторяется, начиная с пункта 1. При моделировании системы 2-х случайных величин (y_1, y_2) с плотностью вероятности $f(y_1, y_2)$, $a_1 \leq y_1 \leq b_1$; $a_2 \leq y_2 \leq b_2$, аналогично моделированию одной случайной величины, три значения: x_m, x_{m+1}, x_{m+2} , выданные датчиком E , преобразуются в значения:

$$y_{1m} = a_1 + (b_1 - a_1)x_m \quad (11)$$

$$y_{2m} = a_2 + (b_2 - a_2)x_{m+1} \quad (12)$$

$$z_{3m} = f_{\eta m} x_{m+2}, \text{ где } f_{\eta m} = \max[f(x_1, x_2)] \quad (13)$$

$$\text{Если } z_{3m} \leq f(y_{1m}, y_{2m}) \quad (14)$$

то $y_1 = y_{1m}$; $y_2 = y_{2m}$.

В этом случае случайные точки с координатами y_{1m}, y_{2m}, z_{3m} - равномерно распределены в пределах параллелепипеда со сторонами, равными $(b_1 - a_1)(b_2 - a_2)$, $f_{\eta m}$ и условие (14) означает попадание точки под поверхность f_{η} . Аналогично моделируется система n случайных величин (y_1, y_2, \dots, y_n) .

7.4 Моделирование нормально распределённой случайной величины Y .

Оно может быть осуществлено на основании центральной предельной теоремы, согласно которой закон распределения суммы независимых случайных величин стремится к нормальному с увеличением числа слагаемых. Для решения некоторых задач практически сумму $y = \sum_{i=1}^n y_i$ значений, выданных с генератором случайных чисел с характеристиками $f(x_i)=1$, $0 \leq x_i \leq 1$, $m_x=0.5$, $\sigma_{x_i} = 0.5 / \sqrt{3}$.

Можно считать значениями распределённой случайной величины $y = \sum_{i=1}^n x_i$ при $n \geq 8$. Так как все слагаемые x_i имеют одинаковые математические ожидания m_x и дисперсии D_{x_i} , то $m_y = n m_x$, $D_y = n D_{x_i}$. В таблице 1 приведены формулы для расчёта случайных величин для различных видов распределений на базе случайной величины с равномерным распределением.

Получение случайной величины с различными распределениями.

Таблица 1.

Нужное распределение	Плотность распределения	Способ получения случайной величины
Экспоненциальное	$f(y) = \lambda e^{-\lambda(y-\mu)}, \mu \leq y < \infty$	$y' = -\frac{1}{\lambda} \ln(1 - x_i) + \mu$
Гамма распределение (целочисленные значения η)	$f(y) = \frac{\lambda^{\eta}}{\Gamma(\eta)} e^{-\lambda y} y^{\eta-1}, 0 < y < \infty$	$y' = -\frac{1}{\lambda} \sum_{i=1}^{\eta} \ln(1 - x_i)$
Распределение χ^2	$f(y) = \frac{1}{2^{\gamma/2} \Gamma(\gamma/2)} y^{(\gamma/2)-1} e^{-y/2}, 0 \leq y < \infty$	$y' = \sum_{i=1}^{\gamma} R_N$; R_N - нормированная случ. величина с нормальным законом распределения
Логарифмические норм. Распределение	$f(y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left[\frac{\ln y - \mu}{\sigma}\right]^2\right), 0 \leq y < \infty$	$y' = e^{\sigma R_N + \mu}$
Вейбулла	$f(y) = \frac{\eta}{\sigma^{\eta}} y^{\eta-1} \exp\left[-\left(\frac{y}{\sigma}\right)^{\eta}\right], 0 \leq y < \infty$	$y' = -\sigma \left[\ln(1 - x_i)\right]^{1/\eta}$

8. Моделирование систем с использованием типовых математических схем

8.1 Блочные иерархические модели процессов функционирования систем

Рассмотрим машинную модель M_m системы S как совокупность блоков $\{m_i\}$, $i=1,2,\dots,n$. 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 $t_i^j \leq T$, 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_m в целом $t^{(k)} \leq T$ будем считать все моменты изменения блоков $\{m_i\}$, рис. 8.1. см. ниже.

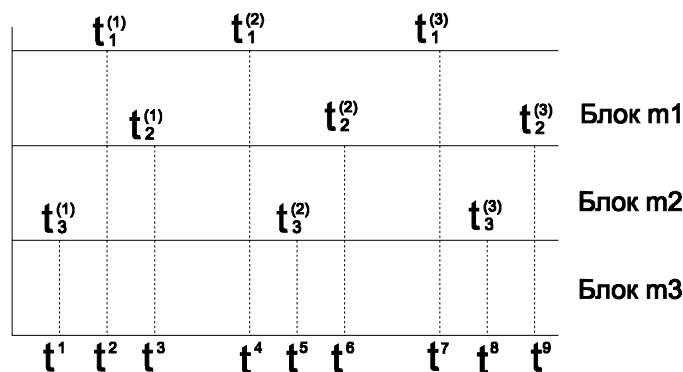


Рис. 8.1. Смена состояний модели для случаев 3-х блоков

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_m её блки представляются соответствующими программными модулями.

8.2 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.):

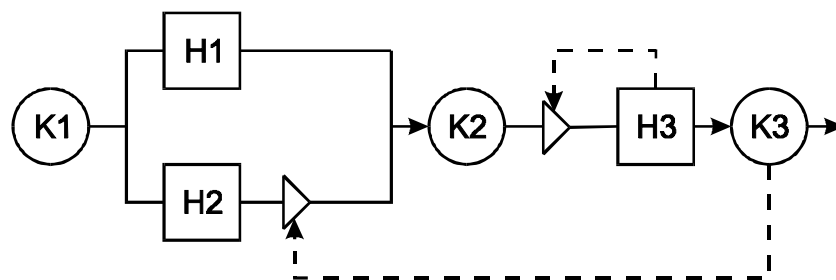


Рис. 8.2. Фрагмент Q-схемы.

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 заблокированным 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 в 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 K может привести в этот момент времени к последовательному изменению состояний нескольких элементов (H, K) , 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.):

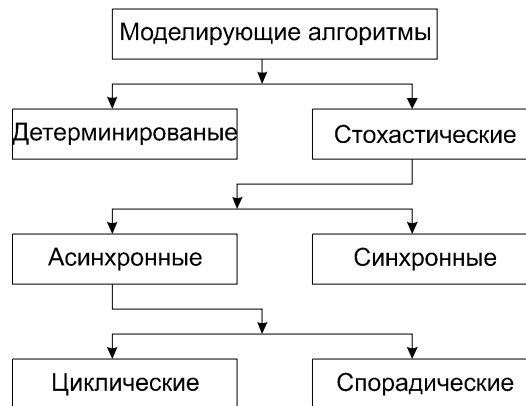


Рис. 8.3. Виды моделирующих алгоритмов Q -схемы.

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 (I , H или K) 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 I , H и K . 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)

8.3 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.):

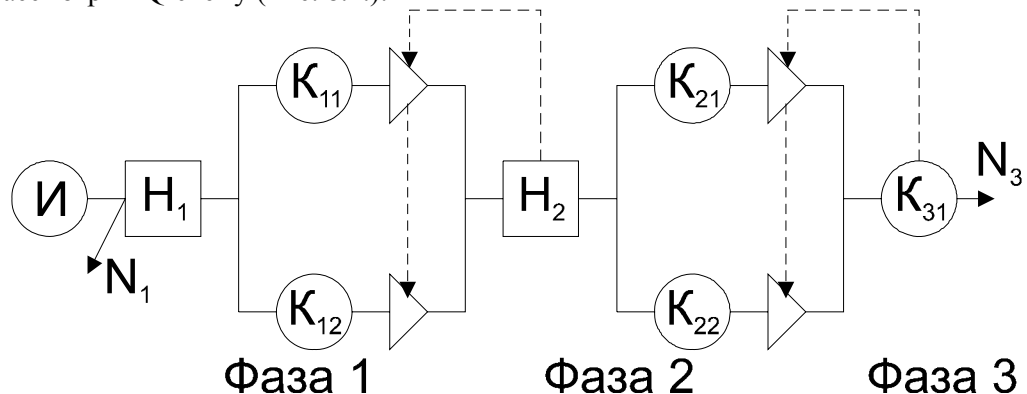


Рис. 8.4. Трёхфазная Q -схема.

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 $K_{k,j}$, $k=1,2,3,\dots$; $j=1,2,\dots$;

$z_i, z_{k,j}$ - состояния накопителей и каналов обслуживания;

t_n - текущее время моделирования;

L_i - ёмкость i -ого накопителя;

L_k^m - число каналов в k -ой фазе;

N_1, N_2 - число выходных заявок;

T - 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.

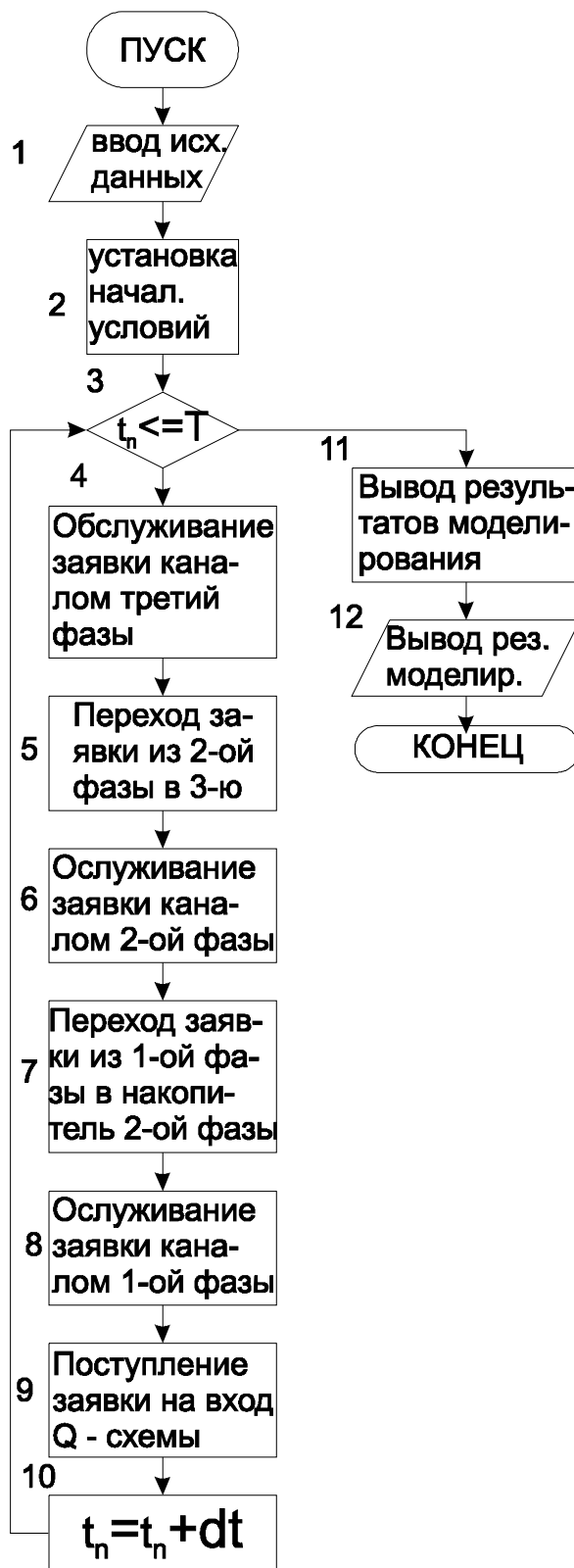
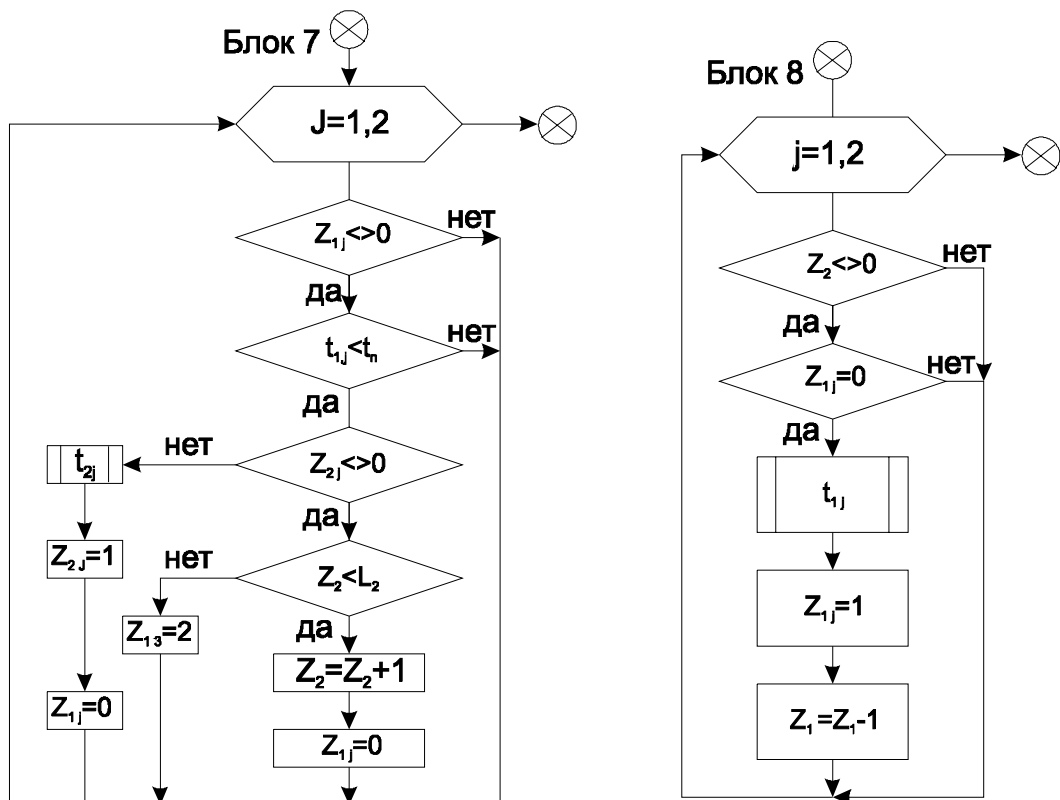
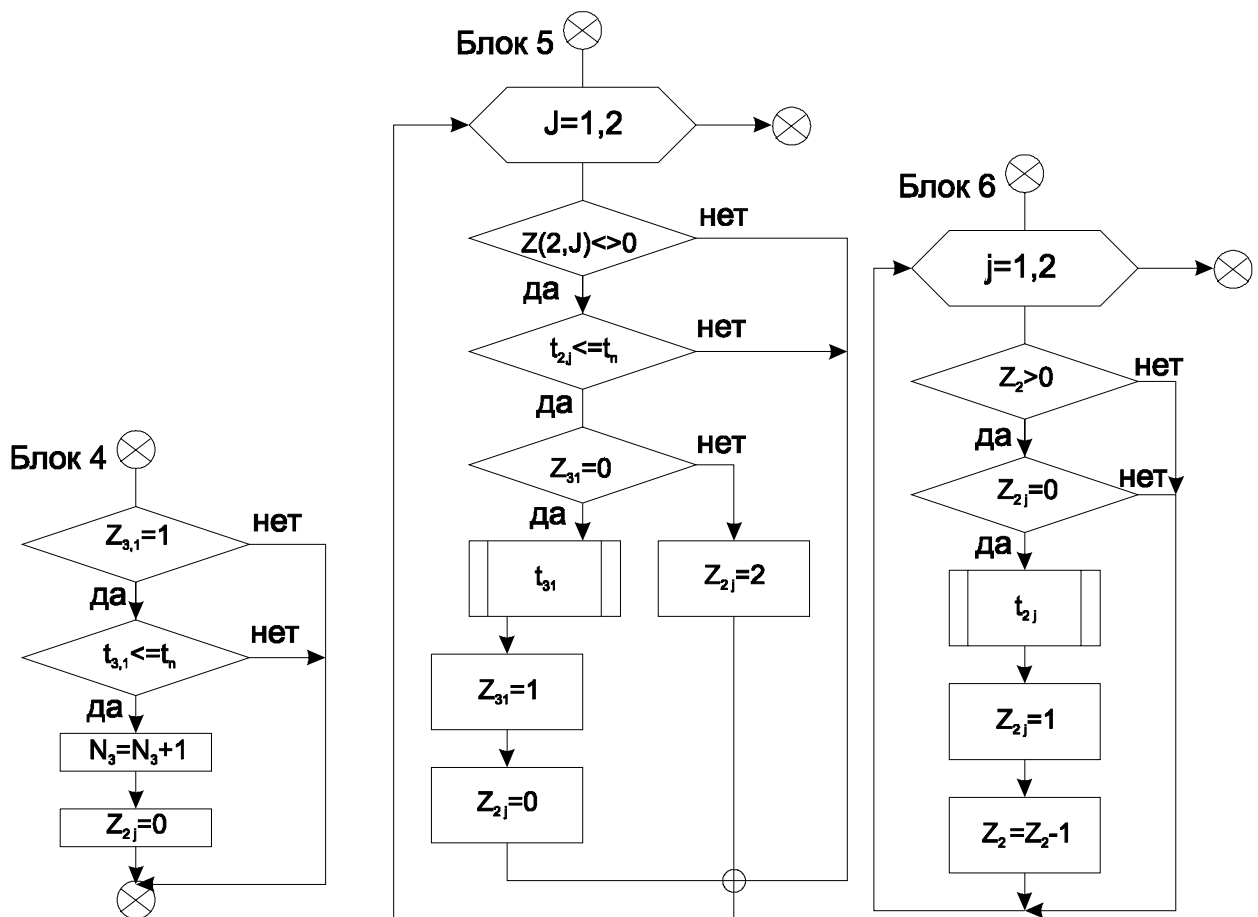
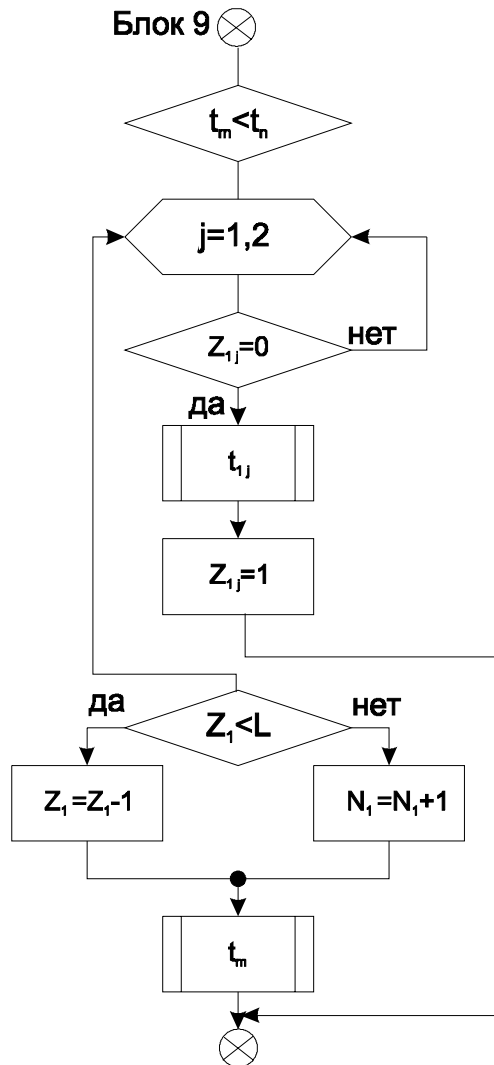


Рис. 8.5. Блок схема моделирования Q-схемы по принципу "Δt".

А далее более подробно рассмотрены алгоритмы блоков 4-9.





9. Программные и технические средства моделирования систем.

9.1 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 включают 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.)

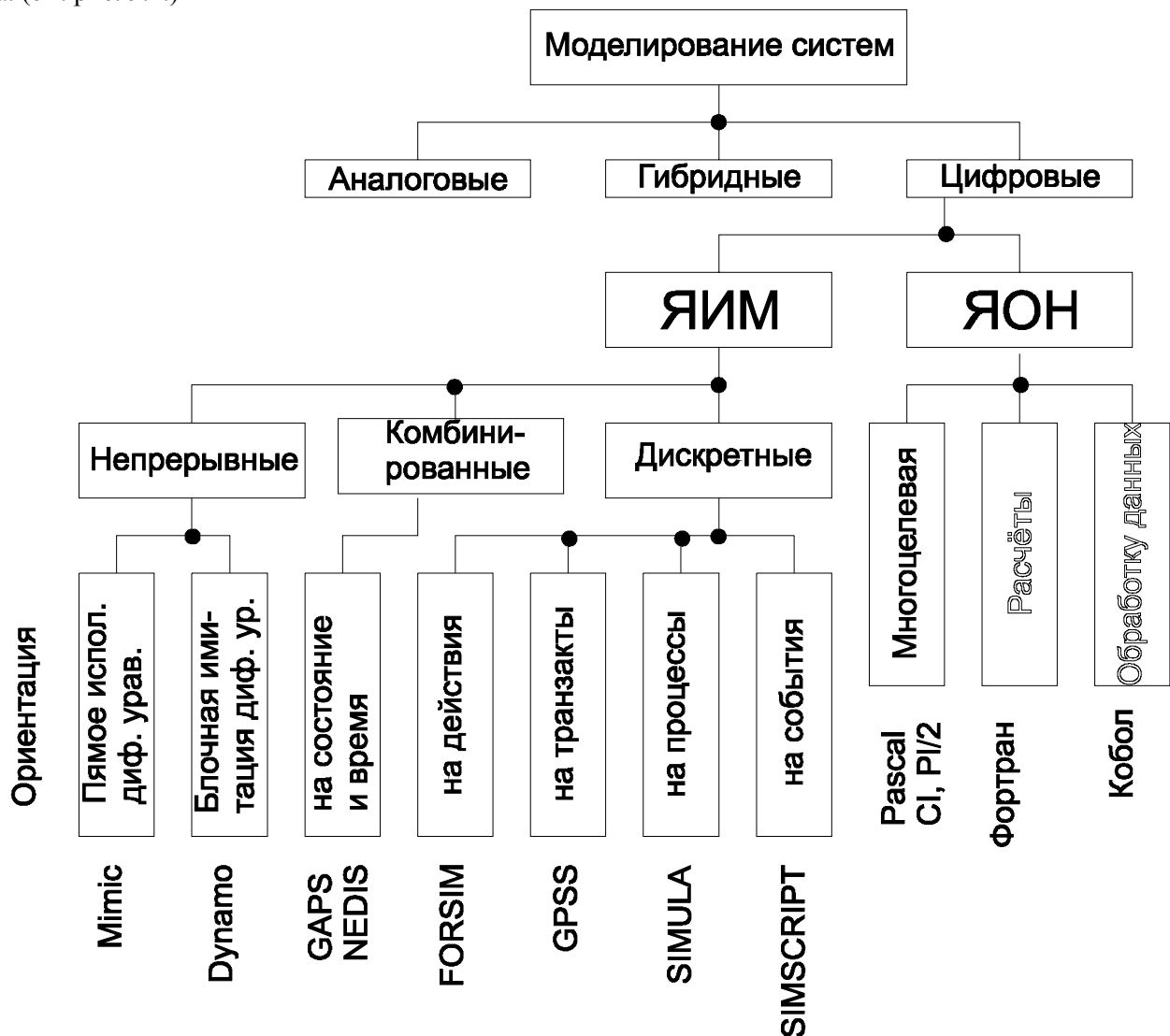


Рис. 9.1. Классификация языков моделирования.

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 собы-

тия двух видов: 1) события, от состояния Z_i ; 2) события, зависящие от времени t . При использовании языка GAPS на пользователя 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 M 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 отдадут 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 балл:

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.

10. Язык программирования GPSS

Этот язык с 1968 года входит в математическое обеспечение машин фирмы IBM, один из наиболее популярных языков ИМ.

Общие сведения.

GPSS составлен из объектов и операций (логических правил). Объекты делятся на семь классов:

- динамические (ДО);
- аппаратно-ориентированные (АО);
- статические (СО);
- операционные (ОО);
- вычислительные (ВО);
- запоминающие (ЗО);
- группирующие (ГО).

До — элементы потока обслуживания заявки или "транзакты". Они создаются и уничтожаются, с каждым транзактом может быть связано некоторое число "параметров"

АО — соответствуют элементам оборудования, которые управляются ДО.

К ним относятся:

- накопители;
- устройства;
- логические переключатели.

СО:

- очереди;
- таблицы.

ЗО:

- ячейки;
- матрицы ячеек.

ГО:

- группы;
- списки.

ВО:

- арифметические и булевы переменные;
- функции.

Каждой очереди соответствует перечень транзактов, задержанных в какой-либо точке системы и запись длительности этих задержек: $T_3 = \{\tau_{i3}\}$.

Таблицы могут использоваться для построения распределений выбранных величин.

ОО - блоки – формируют логику системы, давая транзактам указания, куда идти дальше.

Для того чтобы смоделировать систему, необходимо составить её описание в терминах GPSS, затем симулятор генерирует транзакты, продвигает через заданные блоки и выполняет действия соответствующие блокам. Продвижение создаёт блок GENERATE. Каждое продвижение транзакта является событием, которое должно произойти в определённый момент времени. Симулятор регистрирует время наступления каждого события, затем производит обработку событий в правильной хронологической последовательности.

Если транзакты заблокированы, то симулятор продвинет их тогда, когда изменятся блокирующие правила. Симулятор моделирует часы, их показания в любой момент времени называют абсолютным временем. Относительное время показывает текущее время в модели. При помощи специальной операции относительное время может устанавливаться в нуль и последующий счёт времени будет производиться от этой точки. Другой операцией в нуль могут устанавливаться оба значения времени. Все времена в модели изображаются целыми числами. Симулятор рассчитывает схему по принципу ближайшего события. Центральной задачей симулятора является просмотр и проверка всех возможных событий. Транзакты входят в цепи. Существует пять видов цепей:

- 1) Цепь текущих событий включает в себя те транзакты, планируемое время наступления которых равно или меньше текущего часового.
- 2) Цепь будущих событий включает в себя транзакты, время которых не дошло для обслуживания.
- 3) Цепь прерванных событий.
- 4) Цепь парных транзактов – в текущий момент времени имеют статус парности (ожидают прибытия синхронизирующих транзактов).
- 5) Цепь пользователя включает транзакты, которые пользователь удалили из цепи текущих транзактов.

Цепь текущих событий организуется в порядке убывания приоритетов транзактов и в порядке очередности поступления.

В зависимости от различных условий и требований пользователь система помещает транзакты в те или иные цепи.

Программа на GPSS создаётся в текстовом редакторе в определённом формате. Формат ввода содержит 3 различные поля: поле метки (позиции 5-9), поле операции (позиции 13-23) и поле переменных (позиции 26-80). Поле переменных содержит подполя, которые обозначены A, B, C, D, ..., H. Последующие отделяются от предыдущих запятыми. Пропущенное значение в поле переменных выделяется запятыми (кроме конца поля).

Каждый из объектов требует определённого числа ячеек ОЗУ, в которых во время моделирования хранятся атрибуты объекта (АТО). АТО, к которым может обращаться программист, называются стандартными числовыми атрибутами (СЧА). Все СЧА имеют одно- или 2-х буквенные мнемонические обозначения. Мнемонические обозначения указывают на тип СЧА, а целочисленное значение – на конкретный СЧА.

Номера блоков можно определять символическими обозначениями. При этом обозначение должно включать от 3-х до 5-ти знаков, отличных от пробела, первые три из которых должны быть буквами. Эти ограничения необходимы для того, чтобы избежать смешивания атрибутов системы и символов. Дополнительным ограничением является недопустимость таких специальных знаков, как "-", "+", " " и т.д.

Если в полях A, B, C блока представлены стандартные числовые атрибуты N_j или W_j , то необходимо, чтобы номер блока был представлен в качестве аргумента. Если этот номер блока определяется символически, то такое представление должно быть отличным от мнемонических обозначений, указанных СЧА (N или W). В префикса символического имени используется знак доллара \$. Пользователь может относительно-

ную адресацию. В символической записи CROSS±n символ CROSS указывает на нужный блок, а число n – на номер блока, отсчитываемого от номера блока CROSS. При косвенной адресации предполагается, что нужный аргумент представлен некоторым параметром. Последний обозначается *, за которой следует целое число. Например, S*10 соответствует текущему значению накопителя, номер которого задан параметром 10 (буква S - означает накопитель). Косвенная адресация неприменима только для СЧА C1, M1, RN_n.

10.1 Аппаратно - ориентированные блоки.

К группе АО - блоков относятся:

SEIZE - блок занятия прибора;
RELEASE - освобождение прибора;
PREEMT- захват устройства;
RETURN - возврат захваченного прибора старому транзакту;
ENTER - вход в устройство (накопитель);
LEAVE - выход из накопитель;
LOGIG - изменение логических переключателей.

Введение в моделирующую программу устройств и накопителей позволяет автоматически регистрировать статическую информацию.

Для управления **ключами** используется оператор LOGIG. Предусмотрено три режима изменения состояния ключа: сброс в "0", установка в "1", инвертированное изменение состояния ключа на противоположное.

10.2 Динамически - ориентированные блоки.

В процесс моделирования транзакты создаются, порождают другие транзакты, собираются и уничтожаются. Каждому сообщению соответствует набор параметров, количество которых может быть установлено до 100. По умолчанию количество параметров принимается равным 12. Сообщениям можно присваивать приоритет от 0 до 127. По умолчанию приоритет равен 0. С динамической категорией объектов связаны блоки, которые можно разделить на пять групп: задержки, создания и уничтожения, изменения маршрутов, синхронизация, изменения атрибутов сообщений.

Группы: задержки: ADVANCE ;
создания: GENERATE;
уничтожения: TERMINATE, SPLIT, ASSEMBLE;
изменения маршрутов: TRANSFER, LOOP, GATE, TEST;
синхронизации: MATCH, GATHER;
изменения атрибутов сообщений: ASSIGN, INDEX, MARK, PRIORITY.

Функции блоков:

ADVANCE - задержка транзактов;
GENERATE - генерации;
TERMINATE - уничтожения;
SPLIT - расщепления;
ASSEMBLE - соединения;
TRANSFER - передачи;
LOOP -организации цикла;
GATE - проверка состояния;
TEST - сравнения атрибутов;
MATCH - синхронизации;
GATHER - сбора;
ASSIGN - изменений значений параметров;
INDEX - увеличение индекса;
MARK - ;
PRIORITY - изменение приоритета;

10.3 Вычислительная категория

В вычислительной категории используются объекты 3-х видов: арифметические, логические, и функции. Арифметические объекты описываются блоком variable в режиме целых чисел и FVARIABLE в режиме с плавающей точкой. Название карты описывают арифметические действия над СЧА. Аргументы и результаты рассматриваются как целые числа. При вычислении используются операции: +, -, *, / (с отбрасы-

ванием остатка, d - деление по модулю (остаток считается положительным). Допускается использование не более 5-ти скобок.

FVARIABLE - описывает арифметические сочетания различных СЧА, вычисляемых в режиме с плавающей запятой. От аргументов и промежуточных результатов не берётся целая часть, а только от конечного результата.

Блок - BVARIABLE - логическое высказывание, состоящее из некоторой совокупности СЧА и логических атрибутов. При вычислении используется 3 типа операторов: логические, условные и булевы, например, 2 BVARIABLE M1 'LE' P6.

Функции описываются с помощью блока FUNCTION в виде совокупности диапазонов, например:

3 FUNCTION RN1,C5

0,0/.35,11/.42,1.7/.75,2.2/1.0,3.8

10.4 Статическая категория

К ней относятся блоки:

QUEUE - для занятия очереди;

DEPART - для освобождения из очереди;

TABULATE - для регистрации частоты попадания заданного СЧА;

TABLE - для вывода характеристик таблицы;

SAVEVALUE - для сохранения информации в специальных ячейках ОЗУ;

MSAVEVALUE - для сохранения информации в ячейках ОЗУ;

MATRIX - для описания матрицы;

INITIAL - для присвоения ячейкам и матрицам начальных значений.

10.5 Группирующая категория

К ней относятся блоки LINK и UNLINK, позволяющие переводить сообщения из списка текущих активных событий в списки пользователя неактивных событий и обратно. Использование списков позволяет моделировать различные дисциплины обслуживания событий.

10.6 Специальные типы блоков

Они используются для составления сложных моделей. К ним относятся блоки вывода статистики (PRINT, TRACE, UNTRACE), изменения модели (EXECUTE, CHANGE) блоки BUFFER и HELP, а так же блоки управления группами транзактов (JOIN, REMOVE, EXAMINE, SCAN, ALTER).

Блок BUFFER даёт возможность для немедленного прекращения обработки транзакта и возврата к началу списка текущих событий.

HELP - для составления пользователем независимых программ, которые могут взаимодействовать с GPSS.

JOIN - для включения транзакта или числового значения в группу.

REMOVE - обратная функция JOIN.

EXAMINE - предоставляет возможность пользователю выбора пути, по которому пойдёт транзакт в зависимости от того, принадлежит он группе или нет.

SCAN - для анализа получения значений атрибутов транзактов и изменения их пути.

ALTER - для изменения значений атрибутов транзактов из группы.

10.7 Примеры решения задач моделирования на GPSS

Пример 1. Моделирование непрерывной случайной величины распределённой по экспоненциальному закону с $\lambda = 0.1$.

```
10 SIMULATE
20 EXPON FUNCTION RN1, C24
30 0.0/.1,.104/.2,.222/.3,.365/.4,.509/.5,.69
   .6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12
   .9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5
   .98,3.9/.99,4.6/.995,5.3/.995,6.2/.999,7.01.9997,8.0
40 GENERATE 10,FN$EXPON
50 MARK1
60 TABULATE XTIME
70 TERMINATE 1
80 XTIME TABLE P1,0,2,100
```



```

90      START 200
100     END

```

Модель включает 4 блока, выполняющие следующие функции:

- 40 - генерирование транзакта;
- 50 - присвоение параметру 1 транзакта значения, равного текущему значению часового времени;
- 60 - уничтожение транзакта;
- 10 - признак, необходимый для прогона модели;
- 20 - описание функции (EXPON - метка, RN1 - генератор случайной функции, число пар координат-

24);

- 30 - задание значений пар координат функции;

80 - определение таблицы; XTIME - метка, табулируемой величиной является P1 значение параметров последовательных транзактов, верхний предел первого интервала равен 0, ширина интервала - 2, общее число интервалов - 100;

90 - признак ввода данных, необходимый для выполнения моделирования; прогон модели должен завершиться после прохождения через неё 200 транзактов;

- 100 - признак конца программы.

Пример 2. Составить модель композиции двух случайных величин X_1 и X_2 имеющих экспоненциальные распределения с параметрами λ_1 и λ_2 ($X = X_1 + X_2$), удовлетворяющих обобщённому закону Эрланга 1-ого порядка: $g(t) = \lambda_1 \lambda_2 (e^{-\lambda_1 t} - e^{-\lambda_2 t}) / (\lambda_1 - \lambda_2)$.

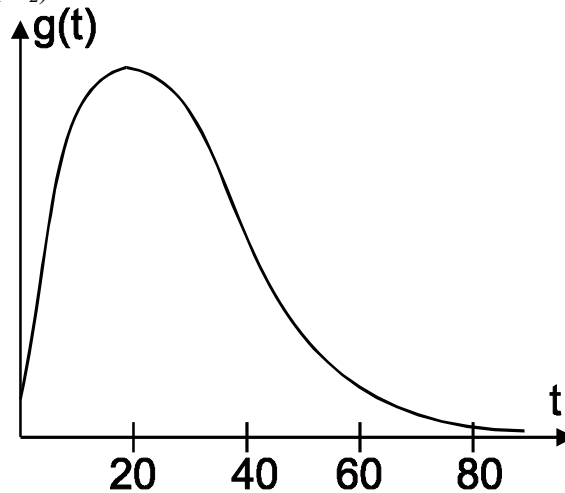


Рис. 10.1. К задаче моделирования композиции 2-х случайных величин.

```

10      SIMULAT
20      EXPON FUNCTION RN1, C24
30      0.0/.1,.104/.2,.222/.3,.365/.4,.509/.5,.69
        .6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12
        .9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5
        .98,3.9/.99,4.6/.995,5.3/.995,6.2/.999,7.01.9997,8.0

40      GENERATE 0,0,,1
50      ASSIGN 1,K500
60      INPUT ADVANCE 10,FN$EXPON
70      ADVANCE 20, FN$EXPON
80      TABULATE XTIME
90      LOOP 1,INPUT
100     TERMINATE 1
110     XTIME TABLE M1,0,5,100
120     START 1
130     END.

```

Функции блоков:

- 40 - генерирование 1-ого транзакта в момент времени $t=0$;
- 50 - присвоение параметру 1 значения, равного 500;
- 60 - моделирование экспоненциального распределённых временных интервалов с параметром λ_1 ;
- 70 - моделирование экспоненциального распределённых временных интервалов с параметром λ_2 ;

80 - формирование таблиц частот XTIME для суммарных интервалов;

90 - контроль числа прохождений транзактов через сегмент блоков, начинающийся с блока INPUT;

100 - уничтожение транзакта.

Пример 3. Моделирование однолинейной системы с пуассоновским входящим потоком с параметром $\lambda = 0.1$ 1/сек. И экспоненциальным временем обслуживания с параметром $\mu = 0.2$ 1/сек.

```
10      SIMULATE
20      LINE EQU      1
30      EXPON FUNCTION RN1,C24
40      0.0/.1,.104/.2,.222/.3,.365/.4,.509/.5,.69
        .6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12
        .9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5
        .98,3.9/.99,4.6/.995,5.3/.995,6.2/.999,7.01.9997,8.0
50      GENERATE 10, FN$EXPON
60      ASSIGN 1,LINE
70      QUEUE O1
80      SEIZE LINE
90      DEPART O1
100     ADVANCE 5,FN$EXPON
110     RELEASE LINE
120     TABULATE XTIME
130     TERMINATE 1
140     XTIME TABLE M1,0,10,100
150     START 500
160     END.
```

9 блоков: 50 - генерирование транзактов;

60 - назначение параметру 1 транзакта номера, соответствующего прибору LINE;

70 - вхождение транзакта в очередь на прибор;

80 - занятие прибора;

90 - выход из очереди;

100 - моделирование обслуживания;

110 - освобождения прибора;

120 - формирование таблицы частот XTIME для времени прохождения транзакта;

130 - уничтожение транзакта;

20 - назначение величины 1 переменной LINE.

Пример 4. Моделирование работы однолинейной системы, имеющей 3 Пуассоновских потока требований с относительными приоритетами и параметрами $\lambda_1=0.01$ 1/сек., $\lambda_2=0.04$ 1/сек., $\lambda_3=0.05$ 1/сек. Экспоненциальный закон обслуживания $\mu_1=0.2$ 1/сек.

```
10      SIMULATE
20      LINE EQU 1
30      EX FUNCTION
40      0.0/.1,.104/.2,.222/.3,.365/.4,.509/.5,.69
        .6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12
        .9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5
        .98,3.9/.99,4.6/.995,5.3/.995,6.2/.999,7.01.9997,8.0
50      GENERATE 100, FN$EX,,,3
60      TRANSFER ,INPUT
70      GENERATE 25, FN$EX,,,2
80      TRANSFER ,INPUT
90      GENERATE 20, FN$EX,,,1
100     INPUT ASSIGN 1,LINE
110     QUEUE LINE
120     SEIZE LINE
130     DEPART LINE
140     ADVANCE 5,FN$EX
150     RELEASE LINE
160     TERMINATE 1
170     START 1000
```

60, 80 - безусловная передача транзактов;
50, 70, 90 - генерирование транзактов с приоритетами 3, 2, 1.

11. Планирование машинных экспериментов с моделями систем.

11.1 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 (экзогенные) и выходные (эндогенные) переменные: x_1, x_2, \dots, x_k ; y_1, y_2, \dots, y_e . Входные переменные в ТПЭ называют факторами а выходные — реакциями. Каждый фактор x_i , $i=1, 2, \dots, k$ может принимать в эксперименте одно или несколько значений, называемых уровнями. 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 уровней 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 показано для случая двух факторов X_1 и X_2 на рисунке (см. ниже рис. 11.1.).

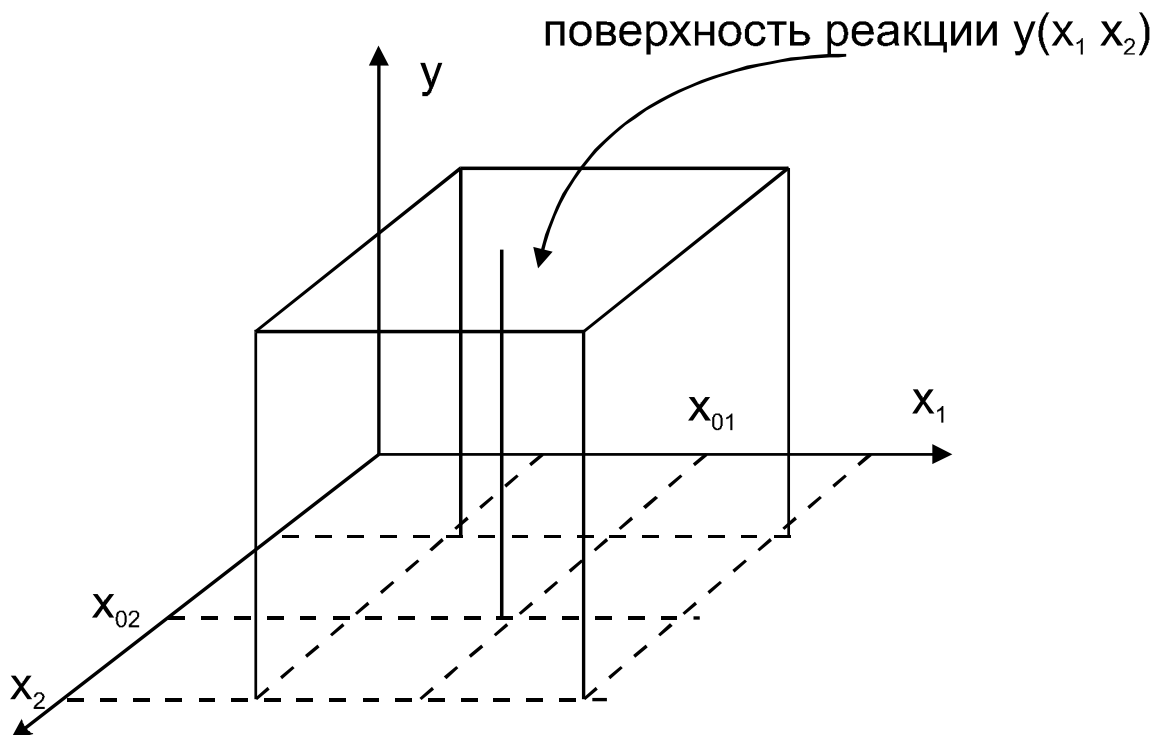


Рис. 11.1. Геометрическое представление поверхности реакции.

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

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 K количественных факторов x_i на некоторую η в отведённый для экспериментирования локальной области факторного пространства ограниченного $x_{i \min} \dots x_{i \max}$, $i = 1 \dots k$.

Функцию отклика обычно выбирают линейной или квадратичной.

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

где $\vec{f}(\vec{x}) =$ вектор с элементами $f_\alpha(\vec{x})$, $\alpha = \overrightarrow{0, d}$, входящих в исходный полином; \vec{B} - вектор коэффициентов. Для двух факторов имеем: $f_0=1$, $f_1=x_1$, $f_2=x_2$, $f_{12}=x_1 x_2$, $f_{11}=x_1^2$, $f_{22}=x_2^2$. $\vec{B} = (b_0, b_1, b_2, b_{12}, b_{11}, b_{22})$.

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

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

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

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

$$\vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_N \end{bmatrix}$$

где y_u - реакция соответствующей u -ой точки плана.

Плану эксперимента поставим в соответствие матрицу планирования:

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

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 , $i=1 \dots k$.

Эксперимент, в котором реализуются все возможные сочетания уровней факторов, называется полным факторным экспериментом (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_{iH} и верхнем x_{iB} , симметрично расположенных относительно основного уровня x_{i0} , $i=1 \dots k$. Геометрическая интерпретация показана ниже на рис. 11.2.:

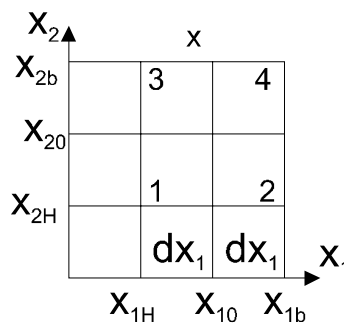


Рис. 11.2. ПЭФ типа 2^2 .

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

11.2 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 $k=10$ ПФЭ должен состоять из 1024 точек. Используют неполные планы, метод "поверхности реакции".

Следующей проблемой является многокомпонентность функции реакции. Здесь можно использовать последовательное однокомпонентное ПФЭ. 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 ПФЭ. В качестве результатов ПФЭ используется средние некоторых распределений, для оценки которых применяют выборочные средние, найденные путём многократных прогонов модели на ЭВМ. Сходимость выборочных средних с ростом объёма выборки называется стохастической. Эта сходимость, как правило, медленная. Если σ - стандартное отклонение среднего N наблюдений будет равно σ/\sqrt{N} , т.е. для уменьшения случайной выборки в k раз требуется увеличить объём 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 ПФЭ 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 даёт принятие числа уровней всех факторов одинаковыми (не больше 3). Функциональная модель ПФЭ определяет количество элементов структурной модели N_Φ , т.е. необходимое число различных информационных точек N_Φ . Причём $N_\Phi < N_c$, где $N_c = q_1 \cdot q_2 \cdot \dots \cdot q_k$ – число экспериментов ПФЭ.

11.3 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
\bar{x}_1	-1	-1	-1	-1	+1	+1	+1	+1
\bar{x}_2	-1	-1	+1	+1	-1	-1	+1	+1
\bar{x}_3	-1	+1	-1	-1	-1	+1	-1	+1

ПФЭ 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 $b_{12}, b_{13}, \dots, b_{123} \dots$ план эксперимента D расширяют до матрицы планирования X путём добавления соответствующей фиктивной переменной: единичного столбца \bar{x}_0 и столбцов произведений $\bar{x}_1\bar{x}_2, \bar{x}_1\bar{x}_3, \dots, \bar{x}_1\bar{x}_2\bar{x}_3$ как показано, например, для ПФЭ типа 2^3 в таблице (см. ниже):

Номер испытания	\bar{x}_0	План ПФЭ			$\bar{x}_1\bar{x}_2$	$\bar{x}_1\bar{x}_3$	$\bar{x}_2\bar{x}_3$	$\bar{x}_1\bar{x}_2\bar{x}_3$	Реакция y
		\bar{x}_0	\bar{x}_2	\bar{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 и 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:

$$y = b_0 + b_1 x_1 + b_2 x_2 + b_{12} x_1 x_2$$

N/n	x_0	План ПЭФ		$x_1 x_2 (x_0)$	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 $x_1 x_2$ (эффектов взаимодействия).

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

$$\tilde{b}_1 = \beta_1 + \beta_{23}; b_2 = \beta_2 + \beta_{13}; b_3 = \beta_3 + \beta_{12}.$$

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 $x_1 x_2$, 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 ($j=1 \dots n$) может повлечь статистическую незначимость 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.