

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный университет
имени Франциска Скорины»

И. В. Максимей

**МАТЕМАТИЧЕСКИЕ ОСНОВЫ
ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ
СЛОЖНЫХ СИСТЕМ**

Гомель
ГГУ им. Ф. Скорины
2014

УДК 519.876.5

Автор: доктор технических наук, профессор **И. В. Максимей**

Максимей, И. В. Математические основы имитационного моделирования сложных систем / И. В. Максимей; под ред. О. М. Демиденко; М-во образования РБ, Гом. гос. ун-т им. Ф. Скорины. – Гомель : ГГУ им. Ф. Скорины, 2014. – 243 с. – ISBN 978-985-439-936-2

В монографии излагается информация, необходимая для успешного построения и использования имитационных моделей сложных систем: методики использования математических основ в прикладных исследованиях на ЭВМ, понятия модельного времени и организации квазипараллелизма в имитационных моделях, существующие способы описания имитационных моделей сложных систем, средства автоматизации имитационных экспериментов.

В издании содержатся базовые понятия и определения основных направлений прикладной математики: теории вероятностей, математической статистики, компьютерного моделирования, планирования экспериментов на ЭВМ и др.

Табл. 4. Ил. 47. Библиогр.: 60 назв.

Рекомендовано к изданию научно-техническим советом
учреждения образования
«Гомельский государственный университет имени Франциска Скорины»

Рецензенты:

доктор технических наук, профессор К. А. Бочков;

доктор физико-математических наук Ю. С. Харин

ISBN 978-985-439-936-2

© Максимей И. В., 2014

© УО «Гомельский государственный университет им. Ф. Скорины», 2014

Список условных обозначений и терминов

СВ – случайные величины;
МО – математическое ожидание;
ОКЗ – область критических значений СВ;
 σ^2 – дисперсия СВ;
МНК – метод наименьших квадратов;
 S' – выборочное средне-квадратичное отклонение СВ;
АНМ – аналитические модели;
КВ – квантили распределения СВ;
СС – сложные системы;
ФР – функция распределения;
ВТПП – вероятностный технологический процесс производства;
mod – мода распределения СВ;
ГСЧ – генераторы случайных чисел;
med – медиана распределения СВ;
ММ – математические модели;
ЭВМ – электронная вычислительная машина;
ИМ – имитационная модель;
НЭ – натуральный эксперимент;
ИЭ – имитационный эксперимент;
ПЭ – план эксперимента; планирование эксперимента;
ПФЭ – полный факторный эксперимент;
ОЭ – однофакторный эксперимент;
ДФЭ – дробный факторный эксперимент;
ЦКП – центральный композиционный план;
ЛПР – лицо принимающее решение;
ФД – функциональные действия;
УПМ – управляющая программа моделирования;
АК – активность;
ПП – подпрограмма;
СМО – система массового обслуживания;
ИСТ – источник транзактов;
КЛА – кусочно-линейный агрегат;
TR – транзакт;
IBD – информационная база данных;
AGR – агрегат;
PROC – процесс;
СМ – система моделирования;
МК – моделирующий комплекс;
УС – устройство;
ВД – временная диаграмма;
МТХО – микротехнологическая операция;
ВСГР – вероятностный сетевой график.

ВВЕДЕНИЕ

Объектом изучения в данной монографии являются сложные системы (СС). Существует много определений. Исходя из опыта исследований технических систем, определим систему в качестве сложной, в тех случаях, когда имеет место сочетание следующих факторов: в системе имеется большое число компонентов; динамика взаимодействия компонентов описывается с помощью сложных алгоритмов; характер взаимодействия компонентов случаен; результат поведения компонентов трудно предсказуем. Существует еще много случаев, когда объект исследования можно отнести к классу СС. По мнению автора, систему можно считать сложной в тех случаях, когда не выполняются предпосылки для применения аналитических методов для ее исследования. Реальные СС можно исследовать в некоторых случаях с помощью аналитических моделей (АНМ), но почти всегда имитационные модели (ИМ) позволяют изучать динамику взаимодействия компонентов СС во времени на высоком уровне детализации процессов, происходящих в ней. В имитационных моделях сложных систем поведение ее компонентов описывается набором алгоритмов, которые реализуют ситуации, возникающие в реальной СС. При этом моделирующие алгоритмы позволяют по исходным данным, содержащим информацию о начальном состоянии СС, и фактическим значениям параметров СС отобразить во времени реальные процессы в системе и получить сведения о возможном поведении СС для конкретной ситуации в системе и ее окружения.

Как известно, предсказательные возможности ИМ значительно меньше, чем АНМ. На основании опыта имитационного моделирования автор рекомендует использовать ИМ при решении задач исследования СС в следующих случаях.

1. Если не существует законченной постановки задачи исследования и идет процесс познания СС. Здесь ИМ служит средством изучения явлений в СС.

2. Если аналитические методы решения подобных задач имеются, но математические процедуры столь сложны и трудоемки, что ИМ дает более простой и экономичный способ решения задачи исследования СС.

3. Когда кроме оценки влияния параметров на поведение СС желательно осуществлять наблюдение за поведением компонентов СС в течение определенного периода времени.

4. Если имитационное моделирование оказывается единственным способом исследования СС из-за невозможности наблюдения явлений в реальной СС.

5. Когда необходимо контролировать протекание процессов в СС путем замедления или ускорения явлений в ходе имитации их на ЭВМ.

6. При подготовке специалистов и освоении новой техники, когда на ИМ обеспечивается возможность приобретения новых навыков в эксплуатации этой техники.

7. Когда изучаются новые ситуации в СС, о которых мало что известно или ничего неизвестно. В этом случае имитация служит средством для предварительной проверки новых стратегий и правил принятия решений перед проведением экспериментов на реальной СС.

8. Если особое значение имеет последовательность событий в практикуемой СС и модель ИМ используется для предсказания узких мест в функционировании СС и других трудностей, появляющихся при введении в нее новых компонентов.

Однако использование ИМ наряду с характерными для них достоинствами имеет ряд существенных недостатков. Разработка хорошей ИМ сложной системы часто обходится дороже создания АИМ и требует больших затрат времени реализации ИМ. Иногда исследователь считает, что ИМ точно отображает реальное положение дел в СС, а в действительности это не так. ИМ в принципе может быть неточна и невозможно даже измерить степень этой неточности.

Кроме того, имитация представляет собой весьма ресурсоемкий способ исследования. Поэтому окончательные результаты анализа удастся получить с помощью последовательности исследований разных сторон функционирования групп компонентов СС. Из достоинств имитации функционирования СС можно выделить: возможность описания поведения компонентов СС на высоком уровне детализации; отсутствие ограничений на вид зависимостей между параметрами ИМ и состоянием внешней среды СС; возможность комплексных исследований динамики взаимодействия компонентов СС во времени и в пространстве параметров СС. Указанные достоинства обеспечивают имитационному методу исследований СС широкую перспективу его применения.

Необходимо отметить, что зачастую исследователи обращаются к имитационному методу, не представляя при этом тех трудностей, с которыми они встретятся, и совершают ряд ошибок методологического характера. Затем наступает разочарование от негативного опыта, иногда преждевременное. По опыту преподавания ряда спецкурсов студентам математических специальностей автором был установлен необходимый минимум знаний по прикладной математике, который будет использоваться будущим молодым специалистом для успешного применения имитационного метода исследования СС. В этот минимум входят: методики использования математических основы в прикладных

исследованиях на ЭВМ, понятие об модельном времени и организации квазипараллелизма в имитационных моделях, знание о существующих способах описания ИМ сложных систем и средствах автоматизации имитационных экспериментов (ИЭ). Перечисленными обстоятельствами определится актуальность данного учебного пособия.

Исходя из вышеизложенного в состав и структуру первой части учебного пособия включены следующие разделы:

- основные элементы теории вероятностей как основа для понимания особенностей функционирования реальных объектов исследования;
- основные элементы математической статистики, аппарата обработки данных ИЭ и известные методы статической проверки гипотез;
- методика использования процедуры Монте-Карло как основы моделирования стохастических сложных систем;
- элементы теории планирования экспериментов на ЭВМ для обоснованного выбора стратегий исследований при имеющемся ресурсе проведения экспериментов с моделями СС;
- основы оптимизации вариантов компьютерного моделирования динамики функционирования СС;
- элементы принятия решений при вероятностном моделировании СС;
- понятие об особенностях построения и использования ИМ, включая знакомство с известными способами организации квазипараллелизма на ЭВМ;
- знакомство с возможностями существующих языков описания структуры СС средствами реализации этих языков моделирования, а также способами автоматизации имитационных экспериментов.

Перечисленный состав необходимых знаний для успешного построения и использования ИМ сложных систем излагается в последующих главах данной монографии. Таким образом, монография представляет собой попытку помочь работникам, специализирующимся в области вероятностного моделирования сложных процессов и систем, освоить необходимый минимум знаний и навыков по организации модельных экспериментов на ЭВМ. Материалы, изложенные в данной монографии апробированы автором и его учениками на математическом факультете УО «Гомельский государственный университет им. Ф. Скорины» при подготовке специалистов инженерно-технических специальностей. Поэтому она рекомендуется для подготовки студентов вузов специальностей: «Прикладная математика», «Информатика», «Автоматизированные системы управления», «Программное обеспечение информационных систем». Монография будет также полезна магистрантам, аспирантам, инженерам и научным работникам, работающим в области проектирования и эксплуатации сложных систем.

1. ЭЛЕМЕНТЫ ТЕОРИИ ВЕРОЯТНОСТЕЙ

1.1. Определение вероятности случайных событий

Случайными называются явления, исход или протекание которых при одинаковом комплексе условий заранее не предсказуем. Предметом теории вероятностей является изучение вероятностных закономерностей массовых однородных случайных явлений. Понятие «массовые» означает то, что закономерности случайных явлений проявляются, как правило, при их многократном воспроизведении. Свойство однородности указывает на то, что условия воспроизведения изучаемого явления должны быть фиксированными, иначе мы не обнаружим никаких устойчивых закономерностей исследуемого явления. Каждый эксперимент заканчивается исходом (результатом). Если этот результат однозначно не предсказуем, то данный эксперимент называется вероятностным (в дальнейшем будет обозначаться символом «Б»). Любой мысленно возможный неразложимый результат вероятностного эксперимента E называется элементарным исходом (элементарным событием) и будем обозначать его « ω ». Множество всех мыслимых взаимоисключающих результатов вероятностного эксперимента E образует пространство элементарных событий, обозначаемых Ω . Случайным называется событие, о котором нельзя заведомо точно сказать, произойдет оно или нет. Событие $A = \emptyset$, которому не благоприятен ни один из исходов пространства элементарных событий Ω и которое в результате вероятностного эксперимента E никогда не произойдет, называется невозможным и обозначается \emptyset . Событием, противоположным событию A , называется событие \bar{A} , которое происходит тогда, когда A не происходит. Суммой (объединением) событий A и B называется третье событие $A \vee B$, которое происходит, если происходит хотя бы одно из событий A или B , или оба вместе. Поэтому сумма событий $(A \vee B)$ есть событие, которому благоприятны элементарные исходы, которые принадлежат хотя бы одному из событий A или B , или обоим вместе. Произведением (пересечением) события A называется третье событие $A \wedge B$, которое происходит только тогда, когда одновременно происходят оба события A и B . Несовместными называют события, произведения которых являются пустым множеством, т. е. невозможным событием.

Пусть вероятностный эксперимент E воспроизведен при одинаковых условиях n раз. При этом некоторое событие A произошло m

раз ($m \leq n$). Число m называется частотой появления случайного события A , а отношение $p^x = m/n$ называется относительной частотой (или частью) случайного события A . Вероятностью случайного события A называется числовая функция $P(A)$, определенная на пространстве элементарных событий Ω , характеризующая меру объективной возможности наступления события A и удовлетворяющая при этом следующим аксиомам А. Н. Колмогорова:

– $P(A) \geq 0$, т. е. вероятность наступления произвольного случайного события – неотрицательная функция;

– $P(\Omega) = 1$, т. е. вероятность наступления достоверного события равна 1;

– $P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$, $A_i \cap A_j = \emptyset$, $i \neq j$, т. е. вероятность наступления

суммы счетного множества попарно несовместимых событий A_i , $i = 1, \dots, \infty$ равна сумме этих событий.

Таким образом, случайные события характеризуются вероятностями их наступления. Основной задачей теории вероятностей является отыскание вероятностей событий без проведения экспериментов с помощью непосредственного анализа случайных событий и условий проведения эксперимента или через известные вероятности других событий, связанных с первым некоторыми соотношениями. Существует несколько способов вычисления вероятностей. Классический способ предполагает, что вероятность случайного события A равна отношению числа элементарных исходов N_A , благоприятных событию A , к общему количеству элементарных исходов N пространства элементарных событий $P(A) = N_A/N$. Недостатками способа являются требования того, чтобы все элементарные исходы вероятностного эксперимента E должны быть равновероятными, а множество элементарных исходов пространства Ω должно быть конечным (чтобы отношение $|A| / |\Omega|$ не являлось неопределенностью вида ∞/∞). Как видим, одним из ограничений классического способа вычисления вероятностей является то, что пространство элементарных исходов Ω вероятностного эксперимента E должно быть конечным. Если пространство Ω является бесконечным, то иногда для вычисления вероятностей случайных событий применяют геометрический способ, идея которого состоит в следующем. Пусть E есть множество элементарных исходов ω , которые можно трактовать как точки в евклидовом пространстве. События трактуются как точки в евклидовом пространстве, а события эксперимента E означают некоторые ограниченные области этого пространства. Вероятность случайного события A определяется выражением:

$$P(A) = \text{MES}(A) / \text{MES}(\Omega), \quad (1.1)$$

где $\text{MES}(A)$ – геометрическая мера (длина, площадь, объем, гиперобъем) области, соответствующей событию A ;

$\text{MES}(\Omega)$ – геометрическая мера области, соответствующая пространству элементарных исходов Ω .

Статистический способ определения вероятности заключается в оценке вероятности случайного события по результатам многократного воспроизведения эксперимента E , например, по относительной частоте появления случайного события. Существует метод экспертных оценок, заключающийся в опросе мнения экспертов о значении случайного события. Вероятности можно определить через известные вероятности других событий. При этом необходимо учитывать следующие важные свойства вероятностей:

- вероятность невозможного события равна 0;
- когда в пространстве Ω , содержащем возможные исходы ($\Omega = (\omega_1, \omega_2, \dots, \omega_i)$) заданы вероятности элементарных исходов $P(\omega_1), P(\omega_2), \dots, P(\omega_i), \dots$, то вероятность произвольного события $A = \{\omega_j, \omega_k, \omega_l\}$ равна сумме вероятностей элементарных событий, благоприятных событию A , т. е. $P(A) = \sum_{\omega_i \in A} P(\omega_i)$:

- если $A \subseteq B$, то $P(B/A) = P(B) - P(A)$;

- если $A \subseteq B$, то $P(A) \leq P(B)$; (1.2)

- $P(\bar{A}) = 1 - P(A)$, т. е. сумма противоположных событий равна 1;

- $0 \leq P(A) \leq 1$, т. е. вероятность произвольного события принадлежит отрезку $[0; 1]$;

- вероятность произведения двух несовместных событий равна 0;

- сумма вероятностей гипотез о событии H_1, H_2, \dots, H_n равна 1.

Теорема сложения вероятностей. Пусть A и B – произвольные случайные события, принадлежащие Ω , тогда вероятность суммы этих событий равна сумме вероятностей этих событий без вероятности их совместного появления:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B), \quad (1.3)$$

Для трех произвольных событий вероятность его определяется по формуле:

$$P(A \cup B \cup C) = P(A) + P(B) + P(C) - P(A \cap B) - P(A \cap C) - P(B \cap C) + P(A \cap B \cap C). \quad (1.4)$$

Теорема умножения двух вероятностей. Вероятность произведения двух произвольных событий равна произведению вероятности одного из события на условную вероятность другого события при условии, что первое уже произошло:

$$P(A \cap B) = P(A) \cdot P(B/A) = P(B) \cdot P(A/B). \quad (1.5)$$

Теорема умножения нескольких вероятностей. Вероятность свершения одновременно трех событий определяется по формуле

$$P(A \cap B \cap C) = P(A) \cdot P(B/A) \cdot P(C/A \cap B). \quad (1.6)$$

Формула полной вероятности является следствием теоремы сложения и умножения вероятностей. Например, пусть требуется определить вероятность некоторого случайного события A , которое может произойти или не произойти вместе с одной из гипотез H_1, H_2, \dots, H_n . Тогда вероятность указанного события можно вычислить по формуле:

$$P(A) = \sum_{i=1}^n (P(H_i) \cdot P(A/H_i)), \quad (1.7)$$

т. е. как сумму произведений вероятностей каждой гипотезы на условную вероятность события A при условии наступления этой гипотезы.

Формула Байеса является следствием теорем сложения и умножения вероятностей и формулы полной вероятности. Применяется формула Байеса для переопределения вероятностей гипотез, сопутствующих некоторому событию A , о котором стало известно, что оно произошло. Например, пусть некоторое событие A может произойти или же не произойти вместе с одной из гипотез H_1, H_2, \dots, H_n . Причем известны априорные вероятности этих гипотез $P(H_i); i = \overline{1, n}$. Известно, что событие A произошло. Требуется найти апостериорные (т. е. послеопытные) вероятности гипотез $H_i; i = \overline{1, n}$. Нужно пересчитать вероятности гипотез, сопутствующих (предшествующих) случайному событию при наличии дополнительной информации о нем. В таком случае формула Байеса имеет вид:

$$P(H_i/A) = \frac{P(H_i) \cdot P(A/H_i)}{P(A)} = \frac{P(H_i) \cdot P(A/H_i)}{\sum_{i=1}^n (P(H_i) \cdot P(A/H_i))}, \quad (1.8)$$

где $P(H_i | A)$ – апостериорная (послеопытная) вероятность гипотезы H_i , при условии, что событие A произошло;

$P(H_i)$ – априорная (доопытная) вероятность гипотезы H_i , $i = \overline{1, n}$;

$P(A/H_i)$ – условная вероятность события A при условии справедливости гипотезы H_i ;

$p(A) > 0$ – безусловная вероятность случайного события A , определяемая по формуле полной вероятности (1.7).

Последовательность независимых испытаний. Схема Бернулли.

Повторные испытания называются *независимыми*, если вероятности их исходов не зависят от исходов предыдущих испытаний СС. Испытаниями Бернулли называют повторные независимые испытания, в каждом из которых возможны два исхода (успех или неудача), вероятности которых не изменяются от испытания к испытанию. Обозначим вероятность успеха через p , а вероятность неуспеха q . Очевидно, что $p + q = 1$. Схемой Бернулли называется проведение заранее определенного числа n испытаний Бернулли. В схеме Бернулли исследователей интересуют, как правило, не исходы каждого эксперимента в отдельности, а то, сколько успехов произошло во всей серии испытаний. Исход этой серии экспериментов можно представить в виде комбинации исходов эксперимента серии:

$$\omega = (\omega_1, \omega_2, \dots, \omega_n), \text{ где } \omega_i = \begin{cases} p - \text{произошел успех} \\ q - \text{неуспех} \end{cases}$$

Обозначим через k количество успехов в n испытаниях Бернулли. Вероятность того, что в испытаниях произойдет k успехов, будет $P_n(k)$ определяться по формуле:

$$P_n(k) = C_n^k \cdot p^k \cdot q^{n-k}, \quad (1.9)$$

где p и q – вероятности соответственно успеха и неуспеха в одном испытании.

Наиболее вероятным числом успехов в схеме Бернулли называется число k_0 , для которого справедливо двойное неравенство:

$$P_n(k_0 - 1) \leq P_n(k_0); P_n(k_0 + 1) \leq P_n(k_0).$$

Обычно k_0 определяется неравенством:

$$n \cdot p - q \leq k_0 \leq n \cdot p + p. \quad (1.10)$$

Если число экспериментов велико ($n \rightarrow \infty$) и вероятность успеха в каждом из них мала ($p \rightarrow 0$; $p < 0,2$) таким образом, что произведение $np = \lambda = \text{const}$, тогда вероятность появления равно k успехов в n испытаниях Бернулли будет равна:

$$\lim_{\substack{n \rightarrow \infty \\ np = \lambda}} P_n(k) = \frac{\lambda^{k-1}}{k!} e^{-\lambda}. \quad (1.11)$$

Когда в схеме Бернулли при $0 < p < 1$ и при $n \rightarrow \infty$ используется предельная теорема Муавра-Лапласа:

$$\lim_{\substack{n \rightarrow \infty \\ kp < 1}} P_n(k) \frac{\sqrt{npq}}{\Phi(x_k)} = 1, \quad (1.12)$$

где $\Phi(x_k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x_k} e^{-\frac{\alpha^2}{2}} d\alpha$, называется функцией Лапласа.

При больших n вероятность появления равно k успехов в испытаниях Бернулли определяется приближенным выражением:

$$P_n(k) \approx \frac{\Phi(x_k)}{npq}. \quad (1.13)$$

Если в схеме Бернулли $0 < p < 1$ и при $n \rightarrow \infty$, тогда для любых k_1 и k_2 таких, что $0 \leq k_1 \leq k_2 \leq n$, справедлива теорема Муавра-Лапласа:

$$\lim_{\substack{n \rightarrow \infty \\ 0 < p < 1}} (P(k_1 \leq k \leq k_2) - \Phi(x_2) + \Phi(x_1)) = 0, \quad (1.14)$$

где $\Phi(x_1) = \frac{1}{\sqrt{2\pi}} \int_0^{x_1} e^{-\frac{\alpha^2}{2}} d\alpha$; (1.15)

$$x_i = \frac{k_i - np}{\sqrt{npq}}; \quad i = 1, 2.$$

При больших n вероятность того, что число успехов m в серии n испытаний Бернулли будет принадлежать отрезку $[k_1, k_2]$, определим приближенным выражением:

$$P_n(k_1, k_2) = P(k_1 \leq m \leq k_2) = \Phi(x_2) - \Phi(x_1). \quad (1.16)$$

Теорема Муавра-Лапласа позволяет получать приближенные значения искомой вероятности, однако оно тем точнее, чем ближе вероятность успеха P к 0,5 и чем больше проводится испытаний Бернулли.

1.2. Типы случайных величин

В зависимости от количества возможных значений случайные величины разделяются на два класса: дискретные и непрерывные. Дискретной называется случайная величина ξ , которая в результате эксперимента E может принимать только определенные значения. Множество значений дискретной величины, определяемое пространством Ω , может быть конечным, либо счетным. Непрерывной называется случайная величина ξ , которая в результате эксперимента может принимать все значения из некоторого интервала на всей числовой оси. Множество значений непрерывной случайной величины, определяемое пространством Ω , несчетно.

Принятие дискретной случайной величиной того или иного значения x_1, x_2, \dots, x_n является случайным событием, которое может произойти с вероятностью $P_i, i = \overline{1, n}$. Поскольку указанные события несовместны (случайная величина не может принять сразу несколько значений) и в результате эксперимента ξ обязательно примет некоторое значение x_k , то сумма вероятностей возможных значений величины ξ равна 1. Эта суммарная единичная вероятность некоторым образом распределена между отдельными значениями величины ξ . Случайная величина будет полностью определена с вероятностной точки зрения, если мы зададим указанное распределение ξ , т. е. укажем, с какой вероятностью наступает каждое из случайных событий $\{\xi = x_i\}$. Таким образом устанавливается закон распределения дискретной случайной величины ξ . Поэтому *законом распределения* дискретной случайной величины ξ называется правило (или зависимость), которое каждому возможному значению x величины ξ ставит в соответствие вероятность появления данного значения. Закон распределения случайной величины ξ может быть задан одним из следующих способов:

– таблично, с помощью ряда распределений, задаваемого парами значений $(x_i, p(\xi = x_i))$; $i = \overline{1, n}$;

– столбиковой диаграммой или многоугольником распределений, в котором по оси y откладывается $P(\xi = x_i) = P_i$, а по оси координат задается x_i .

Функцией распределения $F(x)$ случайной величины ξ в точке x называется вероятность того, что величина ξ принимает значение x , т. е. функция определяет вероятность события $\{\xi < x\}$:

$$F(x) = P(\xi < x).$$

Функция распределения произвольной случайной величины ξ обладает следующими свойствами:

- $F(x) \geq 0$, т. е. это неотрицательная функция;
- $F(-\infty) = 0$; $F(\infty) = 1$; она стремится от 0 до 1; (1.17)
- Если $x_1 < x_2$, то $F(x_1) \leq F(x_2)$, т. е. это неубывающая функция;
- Если $x_1 < x_2$, то $P(x_1 \leq \xi \leq x_2) = F(x_2) - F(x_1)$;
- $F(x - 0) = F(x)$ т. е. функция непрерывна слева.

У дискретных случайных величин функция распределения (ф.р.) есть разрывная ступенчатая функция. У непрерывных величин ф.р. непрерывна и дифференцируема на всей числовой оси. Производная от ф.р. непрерывной случайной величины называется плотностью распределения:

$$f(x) = \lim_{\Delta x \rightarrow 0} \frac{P(x \leq \xi < x + \Delta x)}{\Delta x} = F'(x); \quad (1.18)$$

Вероятностный смысл $f(x)$ заключается в том, что она указывает, насколько вероятно значения непрерывной случайной величины ξ попадут в окрестность точки x . Она обладает следующими свойствами:

$$f(x) \geq 0; F'(x) = f(x); P(x_1 \leq \xi \leq x_2) = \int_{x_1}^{x_2} f(x) dx; \quad (1.19)$$
$$\int_{-\infty}^{\infty} f(x) dx = 1; \int_{-\infty}^{x_1} f(x) dx = F(x_1).$$

Чаще всего стараются определить лишь числовые характеристики закона распределения случайной величины. Такими характеристиками бывают:

– характеристики положения: математическое ожидание (МО), мода (mod), медиана (med), квантили (КВ);

– характеристики рассеяния: (среднеквадратического отклонения (σ_n); дисперсии $D_n = \sigma_n^2$, коэффициенты вариации (V_n), коэффициенты асимметрии, распределения (γ_3) и эксцесса (γ_4).

Математическое ожидание ξ определяется по формуле:

$$MO = M[\xi] = \begin{cases} \sum_{i=1}^n x_i P(\xi = x_i) - \text{для дискретной СВ;} \\ \int_{-\infty}^{\infty} x f(x) dx - \text{для непрерывной СВ.} \end{cases} \quad (1.20)$$

Оно обладает следующими свойствами:

$$\begin{aligned} M[\alpha] &= \alpha = \text{Const}; \\ M[\alpha\xi] &= \alpha M[\xi]; \\ M[\xi + \eta] &= M[\xi] + M[\eta]; \\ M[\xi - \eta] &= M[\xi] - M[\eta]. \end{aligned} \quad (1.21)$$

Если ξ и η независимые СВ, то $M[\xi \cdot \eta] = M[\xi] \cdot M[\eta]$.

Мода – наиболее вероятное значение СВ, обозначаемое $\text{mod} [\eta]$;

Медиана – значение СВ ξ , обозначаемое $\text{med} [\xi]$, для которого $P(\xi < \text{med} [\xi]) = P(\xi > \text{med} [\xi]) = 0,5$, т. е. величина ξ равновероятно примет значение, меньше или больше медианы. Поэтому медиану называем средневероятным значением СВ ξ : $F(\text{med} [\xi]) = P(\xi < \text{med} [\xi])$. Геометрический смысл медианы заключается в том, что прямая $x = \text{med} [\xi]$ делит площадь криволинейной трапеции, образованной графиком функции плотности непрерывной случайной величины ξ и осью абсцисс пополам.

Квантилью распределения СВ ξ , уровня $(1 - \alpha)$ называется значение x величины ξ , для которого определяется равенство и $F(x) = 1 - \alpha$.

Дисперсия СВ α определяется по формуле $D[\xi] = M[\xi^2] - M^2[\xi]$.

$$D[\xi] = \begin{cases} \sum_{i=1}^{\infty} x_i^2 P(\xi = x_i) - M^2[\xi] - \text{для дискретной СВ;} \\ \int_{-\infty}^{\infty} x^2 f(x) dx - M^2[\xi] - \text{для непрерывной СВ.} \end{cases} \quad (1.22)$$

Она обладает следующими свойствами:

$$\begin{aligned}
 & D[\xi] \geq 0; D[\alpha] = 0 \text{ при } \alpha = \text{const}; D[\alpha \cdot \xi] = \alpha \cdot D[\xi]; \\
 & D[\xi + \eta] = D[\xi] + D[\eta] \pm 2M_{\xi\eta}; \text{ где } M_{\xi\eta} = M[(\xi - M[\xi]) \cdot (\eta - M[\eta])]; \\
 & - \text{корреляционный момент случайных величин } \xi \text{ и } \eta; \\
 & - D[\xi \pm \eta] = D[\xi] + D[\eta] \text{ для } \xi \text{ и } \eta \text{ независимых СВ}; \\
 & - D[\alpha \pm \xi] = D[\xi]; \text{ где } \alpha = \text{const}, \text{ а } \xi \rightarrow \text{СВ}; \\
 & - D[\xi \cdot \eta] = D[\xi] \cdot D[\eta] + D[\eta] \cdot (M[\xi])^2 + D[\xi] \cdot (M[\eta])^2, \\
 & \text{где } \xi \text{ и } \eta - \text{независимые СВ}.
 \end{aligned} \tag{1.23}$$

Средне-квадратичным отклонением случайной величины ξ называется число $\sigma[\xi]$, равное положительному значению квадратного корня из дисперсии:

$$\sigma[\xi] = +\sqrt{D[\xi]}; \tag{1.24}$$

Таким образом, $\sigma[\xi]$ имеет размерность СВ, и оно характеризует степень разброса СВ в окрестности $M[\xi]$.

Коэффициент вариации СВ ξ не есть число $V[\xi]$, определяемое выражением $V[\xi] = \sigma[\xi]/M[\xi]$, он показывает, насколько близко МО отображает ряд возможных значений СВ ξ .

Моментом СВ ξ k -го порядка называется число, определяемое с помощью $M[(\xi - a)^k]$,

где a – константа (целое число). Если $a = 0$, то моменты называются начальными моментами ($v_k = M[\xi^k]$), когда $a = M[\xi]$, то имеем *центральный* момент k -го порядка $\mu_k = M[(\xi - M[\xi])^k]$. Очевидно, что $v_0 = M[\xi^0] = 1$; $v_1 = M[\xi]$; $v_2 = M[\xi^2]$; $\mu_0 = M[(\xi - M[\xi])^0] = 1$; $\mu_1 = M[\xi - M[\xi]] = 0$; $\mu_2 = D[\xi]$.

Начальные и центральные моменты связаны между собой следующими соотношениями:

$$\begin{aligned}
 \mu_0 &= 1; \mu_1 = 0; \mu_2 = v_2 - v_1^2, \\
 \mu_3 &= v_3 - 3v_2v_1 + 2v_1^3; \mu_4 = v_4 - 4v_3 \cdot v_1 + 6v_1v_1^2 - 3v_1^4.
 \end{aligned} \tag{1.25}$$

Рассмотрим несколько важных особенностей центральных моментов старшего порядка.

Коэффициентом асимметрии (скошенности) распределения СВ ξ называется число, вычисляемое по формуле:

$$\beta_1[\xi] = \frac{\mu_3}{\sigma^3} = \frac{M[(\xi - M[\xi])^3]}{\sigma^3}. \tag{1.26}$$

Если распределение вероятностей СВ ξ скошено влево, то $\beta_1[\xi] > 0$, а если вправо, то $\beta_1[\xi] < 0$. (рисунок 1.1). Если распределение симметрично относительно $M[\xi]$, то $\beta_1[\xi] = 0$.

Коэффициентом эксцесса СВ ξ называется число, вычисляемое по формуле:

$$\beta_2[\xi] = \frac{\mu_4}{\sigma^4} - 3. \quad (1.27)$$

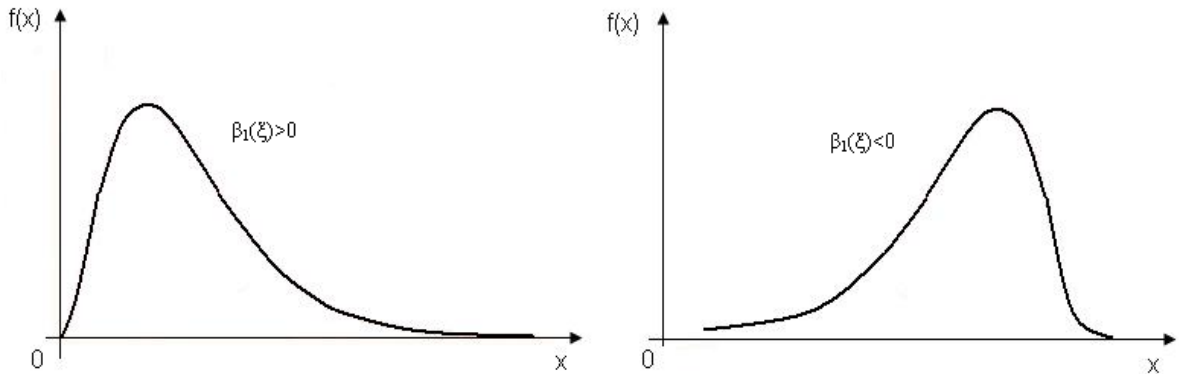


Иллюстрация значений коэффициента асимметрии СВ ξ

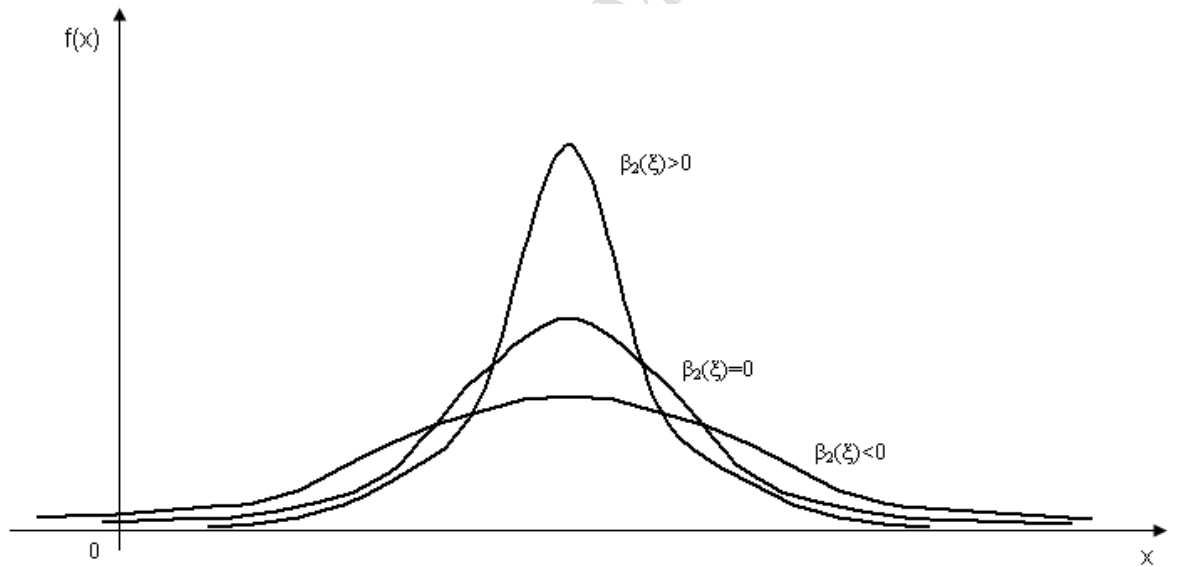


Иллюстрация значений коэффициента эксцесса СВ ξ

Рисунок 1.1 – Различные виды функций плотности $f(x)$ распределения случайных величин

У СВ ξ , имеющей нормальный закон распределения, коэффициент эксцесса равен нулю ($\beta_2[\xi] = 0$). У СВ с более островерженным распределением $\beta_2[\xi] > 0$, а у более пологих распределений, чем нормальное распределение, $\beta_2[\xi] < 0$ (рисунок 1.1).

1.3. Дискретные случайные величины

Рассмотрим вид функций плотности и формулы для расчета значений стандартных распределений дискретных случайных величин, которые наиболее часто используются в имитационных или вероятностных моделях СС. Такими распределениями СВ ξ являются: биномиальное распределение, распределение Пуассона, геометрическое распределение.

Биномиальное распределение. Говорят, что случайная величина x имеет биномиальный закон распределения (обозначается $x: B_i(n, p)$), если она дискретна и означает число успехов k в схеме n испытаний Бернулли. Отметим, что x , имеющая распределение $B_i(n, p)$, принимает только целые значения на отрезке $[0; n]$ с вероятностями, определяемыми функцией Бернулли:

$$P(x = k) = P_n(k) = C_n^k \cdot p^k \cdot q^{n-k}; k = \overline{0, n}. \quad (1.28)$$

Это распределение характеризуется двумя параметрами: числом проводимых экспериментов n и вероятностью успеха p в каждом испытании.

Основные числовые характеристики определяются выражениями:

$$M[x] = n \cdot p; D[x] = np[1 - p]. \quad (1.29)$$

Эквивалентом моды дискретного распределения $\text{mod}[x_n]$ является ближайшее целое в величине $((n + 1) \cdot p - 0,5)$.

Распределение Пуассона. Было определено ранее для случая биномиального распределения, когда $n \rightarrow \infty$ и $p \rightarrow 0$, но $p \cdot n = \text{Const}$.

Поэтому формула 1.11 для распределения Пуассона приобретает вид:

$$P(x = k) = \frac{\lambda^k}{k!} e^{-\lambda}; k = \overline{0, \infty}. \quad (1.30)$$

Эквивалентом моды также является понятие $\text{mod}[x]$ – ближайшее целое к величине $(\lambda = 0,5)$.

Основными характеристиками распределения Пуассона являются математическое ожидание $M[x] = \lambda$ и дисперсия $D[x] = \lambda$. Равенство $M[x] = D[x] = \lambda$ (параметров распределения дискретной СВ) является первым признаком того, что значение дискретной СВ x может определяться распределением Пуассона.

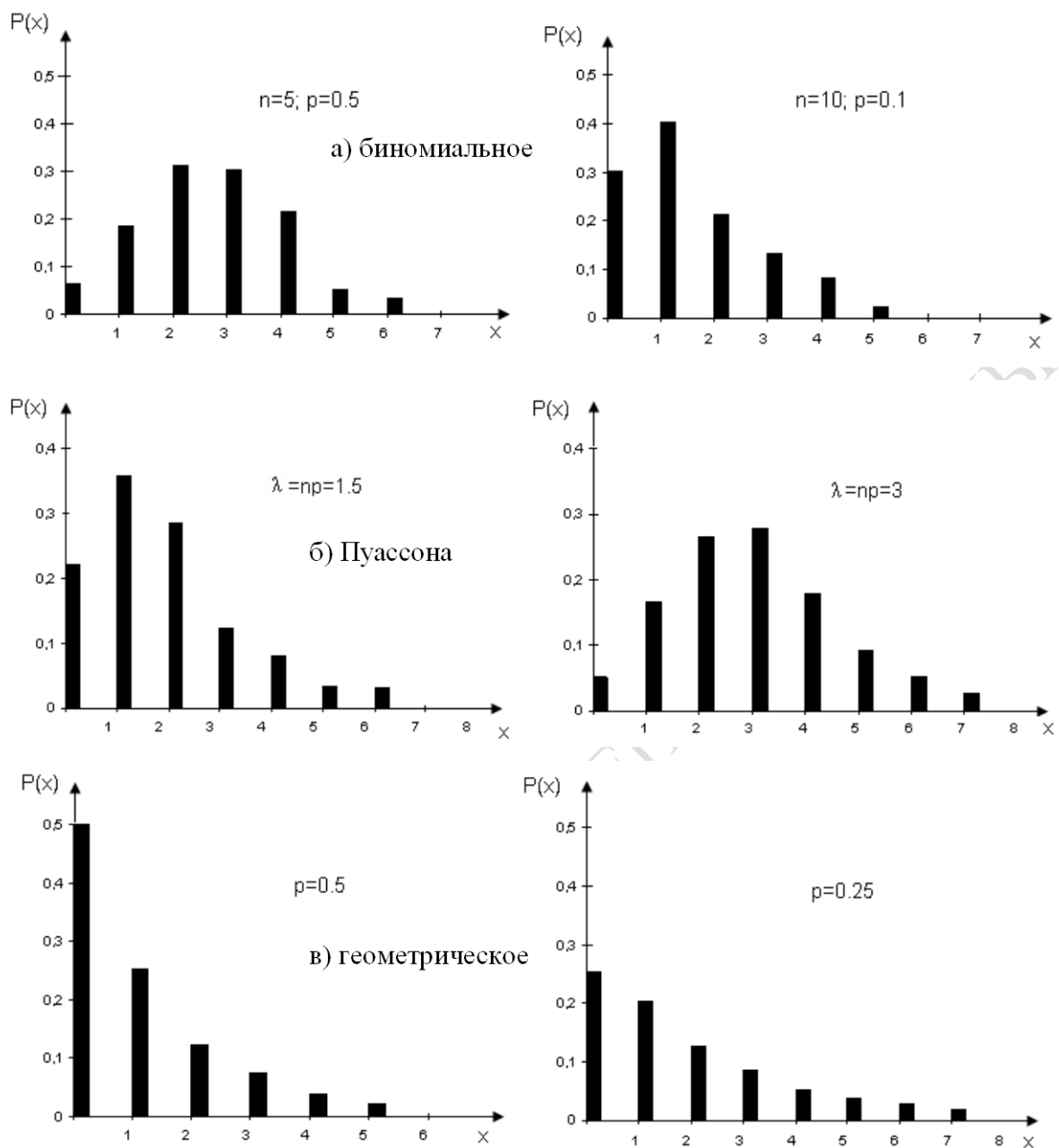


Рисунок 1.2 – Вид функций плотности стандартных дискретных распределений при различных значениях параметров n и p

Геометрическое распределение. Говорят, что случайная величина x имеет геометрический закон распределения и обозначается $(x \rightarrow G(p))$, если она дискретная, и определяет число независимых испытаний Бернулли, предшествующих первому появлению успеха. Множество значений случайной величины, имеющей геометрический закон распределения, это множество неотрицательных целых чисел $0, 1, 2, \dots$. Геометрическое распределение характеризуется только вероятностью успеха p в испытаниях Бернулли. Случайная величина x имеет геометрический закон распределения с параметром p . Рассмотрим

следующие события: $A_i = \{\text{в } i\text{-м испытании произошел успех}\}$; $P(A_i) = p = A_i$ $\{\text{в } i\text{-м испытании произошла неудача}\}$; $P(\bar{A}_i) = 1 - p = q$. Очевидно, что событие $\{x = 0\} \equiv \{\text{число неудач, предшествующих успеху, равно } 0\} = A_1$ – следовательно, $P(x = 0) = P(A_1) = P$. Событие $\{x = 1\} \equiv \{\text{число неудач, предшествующих успеху, равно } 1\} = \{\text{в первом эксперименте произошла неудача, а во втором – успех}\} = \bar{A}_1 \cap A_2$; следовательно, $P(x = 1) = P(\bar{A}_1 \cap A_2) = P(\bar{A}_1) p = q * p$ в силу независимости событий A_i . По аналогии получаем $P(x = k) = q^k * p$. Таким образом, вероятности значений случайной величины x , имеющей геометрическое распределение, определяются выражением:

$$P(x = k) = (1 - p)^k \cdot p; k = \overline{0, \infty}. \quad (1.31)$$

На рисунке 1.2 приведены примеры функций плотности дискретных распределений Бернулли, Пуассона и геометрического.

1.4. Непрерывные случайные величины

В состав стандартных распределений непрерывной случайной величины обычно включают: равномерное, показательное, Эрланга, Вейбулла, нормальное, Стьюдента, хи-квадрат, Фишера.

Равномерное распределение. Говорят, что непрерывная случайная величина x имеет равномерный (прямоугольный) закон распределения – $R(a, b)$ если она может принимать значения только на отрезке $[a, b]$, причем равно-возможно. На рисунке 1.3 а) приведена функция плотности и функция распределения равномерного закона распределения. Аналитически функция распределения плотности и функция распределения имеют следующий вид:

$$f(x) = \begin{cases} \frac{1}{b-a}, & \text{если } x \in [a; b]; \\ 0, & \text{при } x \notin [a, b]; \end{cases} \quad (1.32)$$

$$F(x) = \begin{cases} 0, & \text{если } x < a; \\ \frac{x-a}{b-a}, & \text{если } a \leq x \leq b; \\ 1, & \text{если } x > b. \end{cases} \quad (1.33)$$

Числовые характеристики равномерно распределенной случайной величины x определяются следующими выражениями:

$$M[x] = \text{med}[x] = (a + b) / 2; D[x] = (b - a)^2 / 12; \quad (1.34)$$

$\text{mod}[x]$ – отсутствует; $\beta_1[x]=0$;

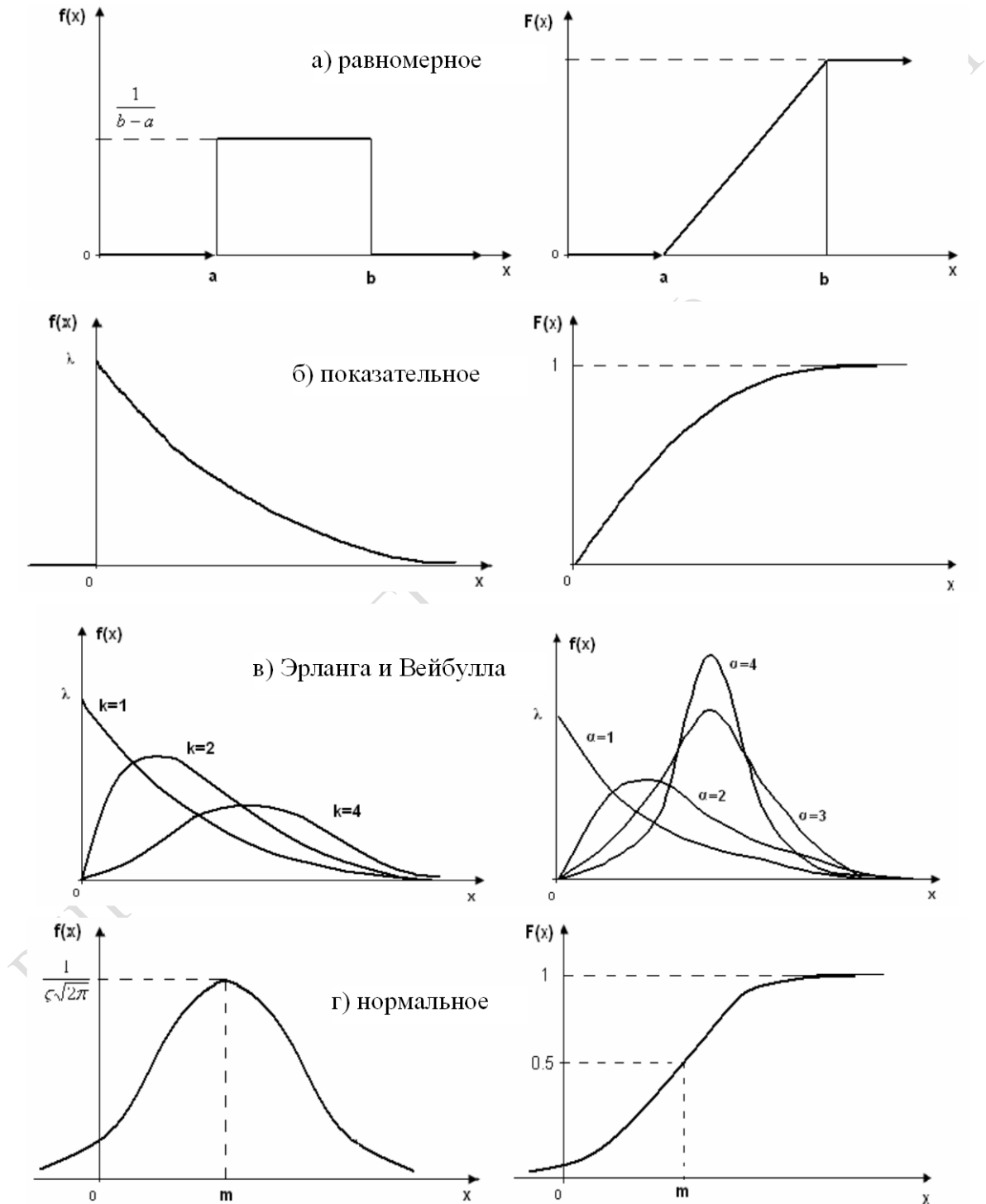


Рисунок 1.3 – Вид дифференциальных и интегральных функций стандартных непрерывных распределений

Показательное (экспоненциальное) распределение. Говорят, что случайная величина x имеет показательное (экспоненциальное) распределение $E(\lambda)$, если она непрерывна, принимает только положительные значения и имеет функцию распределения и функцию плотности распределения:

$$f(x) = \begin{cases} 0, & \text{если } x \leq 0; \\ \lambda \cdot e^{-\lambda x}, & \text{если } x > 0; \end{cases} \quad (1.35)$$

$$F(x) = \begin{cases} 0, & \text{если } x \leq 0; \\ 1 - e^{-\lambda x}, & \text{если } x > 0. \end{cases} \quad (1.36)$$

Из графика функции плотности показательного распределения $f(x)$ (рисунок 1.3, б) видно, что случайная величина x , имеющая показательное распределение, наиболее вероятно принимает малые положительные значения.

Основные числовые характеристики случайной величины x , имеющей экспоненциальное распределение, определяются следующими выражениями:

$$M[x] = \sigma(x) = 1/\lambda; \quad \text{mod}[x] = 0; \quad \text{med}[x] = (\ln 2) / \lambda; \quad D[x] = 1/\lambda^2. \quad (1.37)$$

Распределение Эрланга. Пусть случайные величины x_1, x_2, \dots, x_k независимы и имеют показательный закон распределения с одинаковым параметром $\lambda_i = \lambda = \text{Const}$, обозначаемый $E[\lambda]$, $i = 1, 2, k$. Тогда случайная величина $x = x_1 + x_2 + \dots + x_k$ имеет распределение Эрланга k -го порядка с параметром λ , т. е. $ER(\lambda, K)$. Очевидно, что x непрерывна и принимает лишь положительные значения. Функция плотности распределения x определяется выражением:

$$f(x) = \begin{cases} 0, & \text{если } x \leq 0; \\ \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!}, & \text{если } x > 0; \lambda > 0; k = 1, 2, 3, \dots \end{cases} \quad (1.38)$$

График функции плотности $f(x)$ распределения Эрланга приведен на рисунке 1.3, в. Очевидно, что при $k = 1$ распределение Эрланга совпадает с показательным распределением. Числовые характеристики случайной величины x , имеющей закон распределения Эрланга, определяются следующими выражениями:

$$M[x] = k/\lambda; \text{mod}[x] = (k - 1)/\lambda; D[x] = k/\lambda^2. \quad (1.39)$$

Распределение Вейбулла. Говорят, что случайная величина x имеет распределение Вейбулла с параметрами $\alpha > 0; \lambda > 0$, обозначаемые как $W(\alpha, \lambda)$, если x непрерывна, принимает только положительные значения и имеет функцию плотности распределения:

$$f(x) = \begin{cases} 0; & \text{если } x \leq 0; \\ \alpha\lambda \cdot x^{\alpha-1} e^{-\lambda x^\alpha}; & \text{если } x > 0; \lambda > 0; \alpha > 0. \end{cases} \quad (1.40)$$

Очевидно, что при $\alpha = 1$ распределение Вейбулла $W(\alpha, \lambda)$ совпадает с показательным распределением. Вид функции плотности распределения $W(\alpha, \lambda)$ приведен на рисунке 1.3, в. Числовые характеристики $W(\alpha, \lambda)$ определяются следующими выражениями:

$$M[x] = \lambda^{-\frac{1}{\alpha}} \Gamma\left(1 + \frac{1}{\alpha}\right); \text{mod}[x] = \begin{cases} 0, & \text{если } \alpha \leq 1; \\ \left(\frac{1 - 1/\alpha}{\lambda}\right)^{1/\alpha}, & \text{если } \alpha > 0; \end{cases} \quad (1.41)$$

$$D[x] = \lambda^{-\frac{2}{\alpha}} \left\{ \frac{2}{\alpha} \Gamma\left(\frac{2}{\alpha}\right) - \alpha^{-2} \cdot \left[\Gamma\left(\frac{1}{\alpha}\right) \right]^2 \right\}, \quad (1.42)$$

где $\Gamma(x)$ – гамма-функция (интеграл Эйлера второго рода).

Распределение $W(\alpha, \lambda)$ часто используется в теории надежности.

Нормальное распределение. Говорят, что случайная величина x имеет нормальный (Гауссовский) закон распределения, обозначаемый как $N(m, \sigma)$, если она имеет функцию плотности и функцию распределения вида:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-m)^2}{2\sigma^2}}; \quad (1.43)$$

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(t-m)^2}{2\sigma^2}} dt, \quad (1.44)$$

где параметр m – численно равен математическому ожиданию величины x , а параметр $\sigma > 0$ – среднеквадратичное отклонение (т. е. $m = M[x]; \sigma = \sigma[x]$). Нормальное распределение симметрично

(рисунок 1.3, г) и имеет эксцесс, равный 0. Числовые характеристики случайной величины x , имеющей нормальный закон распределения $N(m, \sigma)$, определяются выражениями:

$$M[x] = \text{med}[x] = \text{mod}[x] = m; D[x] = \sigma^2; \beta_1[x] = \beta_2[x] = 0. \quad (1.45)$$

Случайная величина, которая имеет нормальный закон распределения с параметрами $m = 0$ и $\sigma = 1$, называется стандартной нормальной случайной величиной. Функция распределения стандартной нормальной случайной величины имеет вид:

$$N(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{2}} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^0 e^{-\frac{t^2}{2}} dt + \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{t^2}{2}} dt = \frac{1}{2} + \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{t^2}{2}} dt = \frac{1}{2} + \Phi(x), \quad (1.46)$$

где $\frac{1}{\sqrt{2\pi}} \int_{-\infty}^0 e^{-\frac{t^2}{2}} dt = \frac{1}{2}$ (интеграл Пуассона);

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{\alpha^2}{2}} d\alpha - \text{функция Лапласа (рисунок 1.3, а) [10]}. \quad (1.47)$$

Более известна следующая формула для вычисления функции распределения нормальной величины:

$$F(x) = \frac{1}{2} + \Phi\left(\frac{x-m}{\sigma}\right). \quad (1.48)$$

Для определения вероятности попадания случайной величины, подчиняющейся нормальному закону в интервале $[a, b]$, используется выражение:

$$P(\alpha < x < \beta) = F(\beta) - F(\alpha) = \Phi\left(\frac{\beta-m}{\sigma}\right) - \Phi\left(\frac{\alpha-m}{\sigma}\right). \quad (1.49)$$

Хи-квадрат распределение. Пусть случайные величины x_1, x_2, \dots, x_n независимы и имеют стандартное нормальное распределение $N(0,1)$. Тогда случайная величина χ^2 , определяемая суммой величин x_i^2 $i = \overline{1, n}$, $\chi^2 = x_1^2 + x_2^2 + \dots + x_n^2$, имеет распределение χ^2 (Хи-квадрат распределения) с ν степенями свободы и обозначается как $\chi^2(\nu)$. Случайная величина χ^2 непрерывна, определяется единственным параметром ν , имеет функцию плотности распределения вида:

$$f(\chi) = \begin{cases} 0, & \text{при } \chi \leq 0; \\ \frac{1}{2^{\frac{\nu}{2}} \Gamma(\frac{\nu}{2})} \chi^{\frac{\nu}{2}-1} \cdot e^{-\frac{\chi}{2}}, & \text{при } \chi > 0; \end{cases} \quad (1.50)$$

$$F(\chi) = \begin{cases} 0, & \text{при } \chi \leq 0 \\ \frac{1}{2^{\frac{\nu}{2}} \Gamma(\frac{\nu}{2})} \int_0^{\chi} t^{\frac{\nu}{2}-1} \cdot e^{-\frac{t}{2}} dt, & \text{при } \chi > 0, \end{cases} \quad (1.51)$$

где $\Gamma(\frac{\nu}{2})$ – гамма-функция (интеграл Эйлера второго рода).

На рисунке 1.4, а приведена функция плотности хи-квадрат распределения.

Распределения Стьюдента. Пусть случайные величины x_1, x_2, \dots, x_n независимы и имеют стандартные нормальные распределения $N(0,1)$; $i = \overline{0, n}$. Тогда случайная величина, определяемая выражением:

$$t = \frac{x_0}{\sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}} = \frac{x_0}{\sqrt{\frac{1}{n} x^2}} \quad (1.52)$$

имеет t -распределение Стьюдента с $\nu = n > 0$ степенями свободы и обозначается $t(\nu)$. При этом случайная величина t непрерывна, определяется единственным параметром ν и имеет функцию плотности распределения:

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\pi\nu} \Gamma(\frac{\nu}{2})} \left[1 + \frac{t^2}{\nu} \right]^{-\frac{\nu+1}{2}}; \quad -\infty < t < +\infty \quad (1.53)$$

Вид функции плотности распределения Стьюдента при различных значениях степеней свободы ν приведен на рисунке 1.4, б. Так например, при $\nu = 1$ распределение Стьюдента становится распределением Коши, а при неограниченном увеличении числа степеней свободы распределение Стьюдента сходится к нормальному распределению (на рисунке 1.4, б при $\nu = 300$). Очевидно, что $M[x] = \text{mod}[x] = \text{med}[x]$;

$\beta_1[x]; \beta_2[x] = 0$, а дисперсия (при $\nu > 2$) определяется выражением $D[t] = \nu/(\nu - 2)$, где $\nu > 2$.

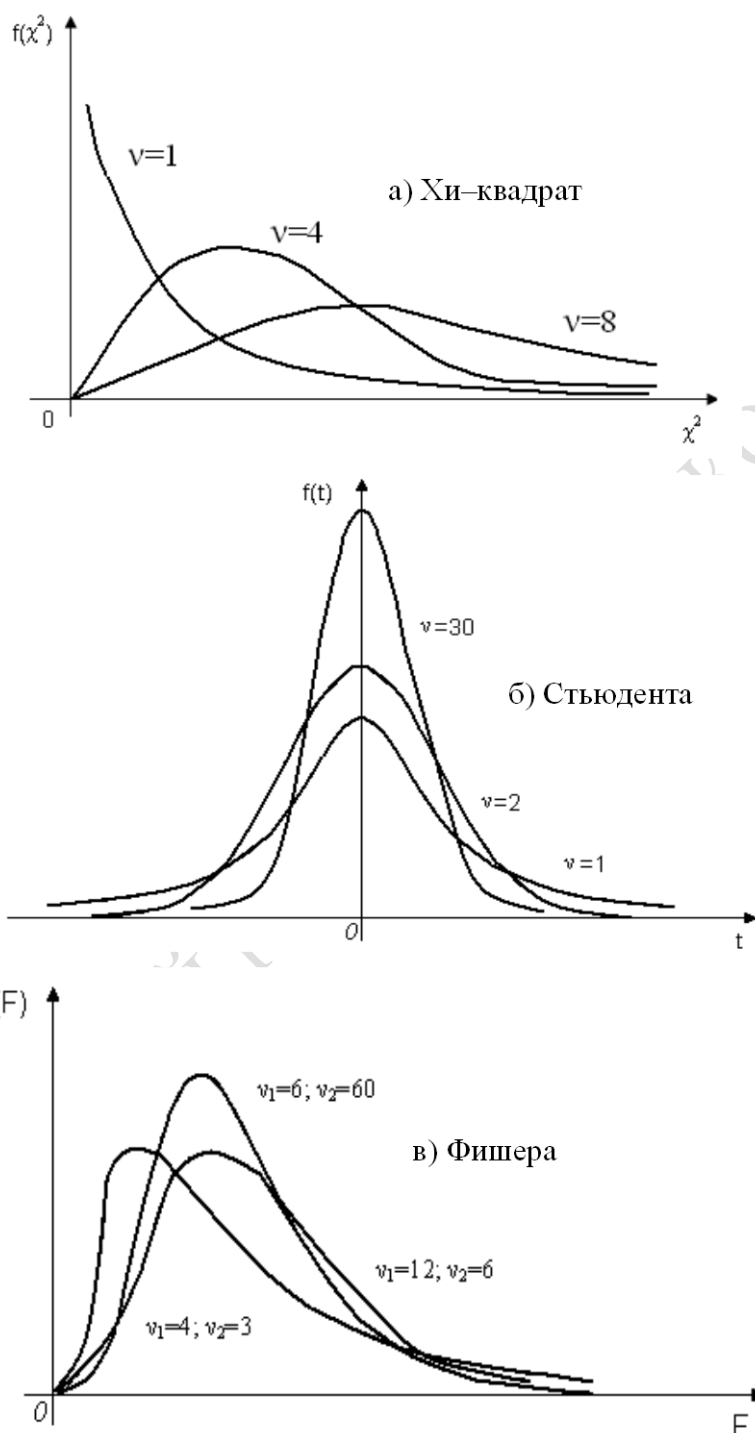


Рисунок 1.4 – Вид функций плотности стандартных непрерывных распределений

Распределение Фишера. Если случайные величины x_1, x_2, \dots, x_n и $\eta_1, \eta_2, \dots, \eta_n$ независимы и имеют каждая стандартное нормальное

распределение, т. е. $x_i, \eta_j \rightarrow N_i(0,1); i = \overline{1, m}, j = \overline{1, n}$, тогда случайная величина F , определяемая выражением

$$F = \frac{\frac{1}{m} \sum_{i=1}^m x_i^2}{\frac{1}{n} \sum_{j=1}^n \eta_j^2} = \frac{n}{m} \frac{X_m^2}{X_n^2}, \quad (1.54)$$

имеет распределение Фишера с $\nu_1 = m; \nu_2 = n$ степенями свободы и функцию плотности распределения:

$$f(x) = \begin{cases} 0, & \text{при } x \leq 0; \\ \frac{\Gamma\left(\frac{\nu_1 + \nu_2}{2}\right) \nu_1^{\frac{\nu_1}{2}} \cdot \nu_2^{\frac{\nu_2}{2}} x^{\frac{\nu_1}{2} - 1} (\nu_2 + \nu_1 x)^{-\frac{(\nu_1 + \nu_2)}{2}}}{\Gamma\left(\frac{\nu_1}{2}\right) \Gamma\left(\frac{\nu_2}{2}\right)}, & \text{при } x > 0, \end{cases} \quad (1.55)$$

графики которой (для разных значений параметров $\nu_1 \nu_2$) представлены на рисунке 1.4, в.

Мы, рассмотрели только типы специальных стандартных распределений, которые часто используются для проверки значимости при статистической обработке результатов натурального и имитационного эксперимента.

1.5 Многомерные случайные величины

1.5.1 Распределение многомерной случайной величины

Многомерной (n -мерной) случайной величиной называется функция $X(\omega)$, определенная на множестве элементарных событий Ω , которая каждому элементарному исходу ω ставит в соответствие значения n действительных чисел. По существу n -мерная случайная величина является совокупностью n одномерных величин, называемых её компонентами. Все компоненты многомерной *дискретной* случайной величины есть одномерные дискретные случайные величины. Компоненты многомерной непрерывной случайной величины являются одномерными *непрерывными* случайными величинами.

Встречаются также и *смешанные* случайные величины, которые содержат одновременно дискретные и непрерывные компоненты. Основной характеристикой многомерной случайной величины является закон распределения, который (как и для одномерных величин) может быть задан таблично, графически или аналитически.

1.5.2 Функция распределения двумерной случайной величины (ξ, η)

Совместной функцией распределения случайных величин ξ и η называется функция $F_{\xi\eta}(x, y)$, равная вероятности того, что компонент ξ примет значение меньшее, чем x , а компонент η – значение меньшее, чем y :

$$F_{\xi\eta}(x, y) = P(\xi < x, \eta < y) = P(\{\xi < x\} \wedge \{\eta < y\}),$$

где x и y – верхние значения соответственно величин ξ и η .

Таким образом, функция распределения двумерной случайной величины $F_{\xi\eta}(x, y)$ в точке (x, y) определяет вероятность, с которой двумерная случайная величина, примет значение в нижнем левом квадранте относительно точки (x, y) . Она обладает следующими свойствами:

- 1) $F_{\xi\eta}(x, y) \geq 0$ – это неотрицательная функция;
- 2) если $x_1 < x_2$, то $F_{\xi\eta}(x_1, y_1) \leq F_{\xi\eta}(x_2, y_1)$,

т. е. это неубывающая функция каждого аргумента, при условии, что другой аргумент фиксирован;

3) $F_{\xi\eta}(x, \infty) = F_{\xi}(x)$; $F_{\xi\eta}(\infty, y) = F_{\eta}(y)$, т. е. если один из аргументов равен ∞ , то в пределе она становится равной функции распределения одномерной случайной величины, соответствующей другому аргументу;

- 4) $F_{\xi\eta}(-\infty, y) = 0$; $F_{\xi\eta}(x, -\infty) = 0$;

- 5) $F_{\xi\eta}(\infty, \infty) = 1$;

6) по каждому из аргументов $F_{\xi\eta}(x, y)$ непрерывна слева.

Двумерная случайная величина называется непрерывной, если её функция распределения $F_{\xi\eta}(x, y)$ – непрерывна, дифференцируема по каждому из аргументов, у которой существует вторая смешанная производная $\partial^2 F_{\xi\eta}(x, y) / (\partial x, \partial y)$.

Функция плотности распределения $f_{\xi\eta}(x, y)$ непрерывной двумерной случайной величины представляет собой предел отношения вероятности попадания случайной величины (ξ, η) элементарный

прямоугольник, примыкающий к точке (x, y) , к площади этого прямоугольника, когда оба его размера стремятся к нулю:

$$f_{\xi\eta}(x, y) = \partial^2 F_{\xi\eta}(x, y) / \partial x \partial y. \quad (1.58)$$

Функция плотности распределения непрерывной двумерной случайной величины (ξ, η) обладает следующими свойствами:

1) $f_{\xi\eta}(x, y) \geq 0$ – это неотрицательная функция;

$$2) P(\{x_1 \leq \xi < x_2\} \wedge \{y_1 \leq \eta < y_2\}) = \int_{x_1}^{x_2} \int_{y_1}^{y_2} f_{\xi\eta}(x, y) dx dy; \quad (1.59)$$

$$3) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{\xi\eta}(x, y) dx dy = 1;$$

$$4) \int_{-\infty}^x \int_{-\infty}^y f_{\xi\eta}(x, y) dx dy = F_{\xi\eta}(x, y);$$

5) $P(\{\xi = x\} \wedge \{\eta = y\}) = 0$.

Т. е. вероятность того, что непрерывная двумерная случайная величина (ξ, η) попадет в точку (x, y) , равна нулю (рисунок 1.5).

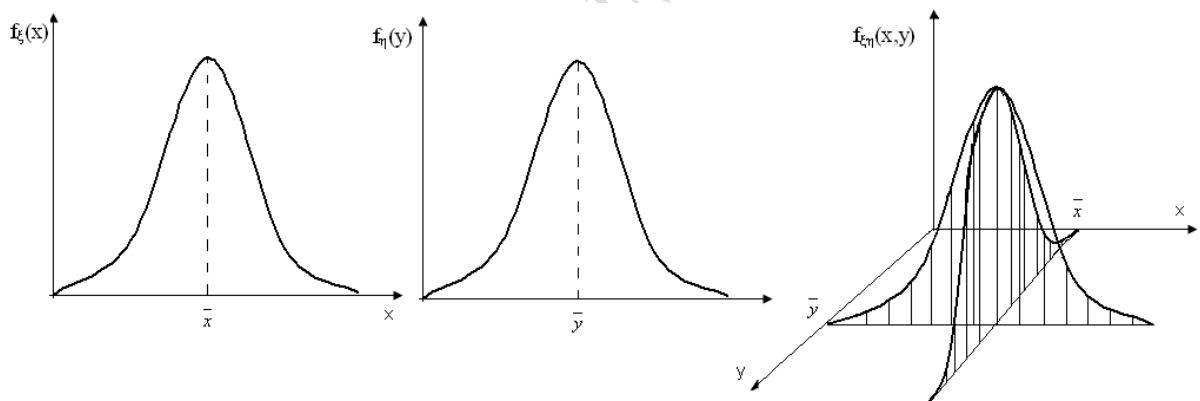


Рисунок 1.5 – Пример функций плотности распределений $f_{\xi}(x)$ и $f_{\eta}(y)$ и плотности соответствующей двумерной случайной величины $f_{\xi\eta}(x, y)$

1.5.3 Независимые случайные величины

Независимыми называются случайные величины ξ и η , если независимыми являются все случайные события, связанные с этими случайными величинами. Это означает, что закон распределения каждой случайной величины не зависит (не изменяется) от того, какие значения приняла другая случайная величина. Зная совместный закон

распределения многомерной случайной величины $F_{\xi\eta}(x, y)$, можно найти законы распределения её компонентов. Пусть известно $F_{\xi\eta}(x, y) = P(\{\xi = x\} \wedge \{\eta = y\})$ и величины ξ и η независимы, то независимыми являются события $\{\xi = x\}$ и $\{\eta = y\}$. В этом случае

$$F_{\xi\eta}(x, y) = F_{\xi}(x) * F_{\eta}(y). \quad (1.60)$$

Тождество независимости двух случайных величин называется теоремой умножения функций распределения независимых случайных величин. Если ξ и η – непрерывные независимые случайные величины, то дифференцируя левую и правую части выражения (1.60), получим:

$$\frac{\partial^2 F_{\xi\eta}(x, y)}{\partial x \partial y} = \frac{\partial F_{\xi}(x)}{\partial x} * \frac{\partial F_{\eta}(y)}{\partial y}. \quad (1.61)$$

Учитывая определения функции плотности распределения непрерывной двумерной случайной величины и функции плотности распределения непрерывной одномерной величины, получаем равенство, называемое теоремой умножения функций плотности распределения независимых величин

$$f_{\xi\eta}(x, y) = f_{\xi}(x) * f_{\eta}(y). \quad (1.62)$$

Если компоненты двумерной случайной величины (ξ, η) зависимы, то для нахождения совместного закона распределения недостаточно знать законы распределения компонентов: требуется знать так называемый условный закон распределения одной из них.

Условный закон распределения величины ξ . Условным называется закон её распределения, вычисленный в предположении, что другая случайная величина η приняла определенное значение. Функция распределения $F_{\xi\eta}(x, y)$ систем зависимых случайных величин может быть записана в виде, называемом теоремой умножения функций распределения величин:

$$F_{\xi\eta}(x, y) = P(\{\xi < x\} \wedge \{\eta < y\}) = F_{\xi}(x) * F_{\eta}(y | \xi < x); \quad (1.63)$$

где $F_{\eta}(y | \xi < x) = P(\eta < y | \xi < x)$ – условная функция распределения величины η при условии наступления события $\{\xi < x\}$. На практике чаще используют форму условного закона распределения $F_{\eta}(y | \xi = x)$

или условную функции плотного распределения $f_{\eta}(y | \xi < x)$, когда величина ξ принимает фиксированные значения x .

$$f_{\xi\eta}(x, y) = f_{\xi}(x) * f_{\eta}(y | \xi = x). \quad (1.64)$$

Т. е. функция плотности распределения непрерывной двумерной величины равна произведению функции плотности распределения одной из них на условную плотность распределения другой при заданном значении первой. Это означает, что кривая условной плотности распределения $f_{\eta}(y | \xi = x_1)$ пропорциональна сечению поверхности функции плотности распределения $f_{\xi\eta}(x, y)$ двумерной величины плоскостью, параллельной координатной плоскости $f_o y$; отсекающей на оси Ox отрезок x_1 .

Для описания многомерных случайных величин используются числовые характеристики её составляющих, а также характеристики, определяющие зависимость между компонентами многомерной величины. Одной из таких характеристик является корреляционный момент $\mu_{\xi\eta}$ (ковариация):

$$\mu_{\xi\eta} = M[(\xi - M[\xi]) (\eta - M[\eta])]. \quad (1.65)$$

Корреляционный момент имеет размерность, равную произведению размерностей величины ξ и η . Часто пользуются безразмерной характеристикой, называемой коэффициентом корреляции, который определяется по формуле:

$$r_{\xi\eta} = \mu_{\xi\eta} / (\sigma[\xi] * \sigma[\eta]); \quad (1.66)$$

Коэффициент корреляции может принимать любые значения из отрезка $[-1, 1]$. Он характеризует степень линейной зависимости между величинами ξ и η . Нулевые значения этой характеристики указывают на отсутствие линейной зависимости между величинами ξ и η .

Более подробно с основными определениями теории вероятности и с примерами случайных величин, а также многомерных СВ можно ознакомиться в работах [57] и [58].

2. ОСНОВНЫЕ ПОНЯТИЯ МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ

2.1. Выборочный метод при исследовании вероятностных систем

Математическая статистика представляет собой науку, которая изучает методы сбора, сохранения, анализа и интерпретации данных, относящихся к случайным явлениям. Она обеспечивает анализ результатов натуральных и имитационных экспериментов, обеспечивая сокращения больших объемов статистического материала. Множество всех исследуемых объектов (ими возможных результатов наблюдений) называется генеральной совокупностью, которая бывает конечной или бесконечной. На практике генеральную совокупность получить или исследовать невозможно, из-за того, что она может быть бесконечной, либо требуется значительное время и финансовые затраты на ее получение, либо ее получение может привести к разрушению исследуемых объектов. Поэтому ограничиваются исследованиям не всей генеральной совокупности, а ее части, называемой *выборкой*. Обычно это ограниченная совокупность значений случайной величины, которые наблюдались в результате n -кратного воспроизведения вероятностного эксперимента. Число n называется *объемом выборки*, а полученное в i -м испытании значение x_i является элементом выборки ($i = \overline{1, n}$). Впоследствии по результатам анализа выборки делают выводы о свойствах всей генеральной совокупности исследуемых объектов. Подобный метод изучения случайных величин или явлений называется *выборочным методом*. Однако не по каждой выборке можно делать обоснованные выводы о свойствах изучаемого явления.

Для этого выборка должна удовлетворять следующим требованиям:

- достаточный *объем* (чем больше элементов содержит выборка, тем достовернее результаты исследования);
- репрезентативность (представительность), т. е. хорошее отражение свойств всей совокупности:
 - *независимость испытаний* друг от друга, что наиболее часто используется для задач статистического анализа данных;
 - проверка предположения о виде распределения случайной величины;

– оценка числовых характеристик случайной величины и параметров функции распределения;

– проверка согласованности гипотезы о виде распределения исследуемой случайной величины с опытными данными.

Первичной формой представления статистического материала является *статистический ряд*, в котором элементы выборки x_1, x_2, \dots, x_n случайной величины X указываются в порядке их появления. Изучение статистического ряда (при большом числе испытаний n) практически не дает представления о характере распределения величины X . Поэтому x_i упорядочивают.

Неубывающая последовательность $\{x_i\}$ называется *вариационным рядом*. Дальнейшая статистическая обработка данных эксперимента заключается в группировке x_i . Если $\{x_i\}$ дискретна, то для всех ее наблюдаемых значений подсчитываются частоты m_i и относительные частоты $v_i = m_i/n$ появления. Данные сводятся в таблицу, называемую сгруппированным статистическим рядом дискретной случайной величины. При исследовании непрерывной случайной величины X наблюдаемые значения, как правило, редко повторяются, поэтому группировку элементов выборки осуществляют по интервалам. Для этого диапазон ее изменения, определяемый минимальным (x_{\min}) и максимальным (x_{\max}) значениями элементов выборки, разрабатывается на k интервалах либо одинаковой, либо различной длины. В каждом k -м интервале подсчитываются частоты и частости попадания выборочных значений случайной величины x . Тем самым строится *интервальный статистический ряд*. Количество равных интервалов можно определить по формуле Стерджесса [57]:

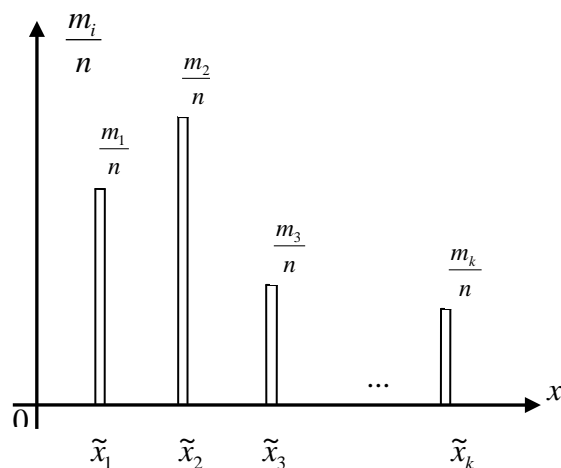
$$k = 1 + 3,322 \log_{10} n. \quad (2.1)$$

При этом количество интервалов k должно находиться в диапазоне $5 < k < 25$. Если длины всех интервалов интервального статистического ряда непрерывной случайной величины предполагаются одинаковыми, их длина $h_i = h$ и границы C_i ($i = 1, \overline{k-1}$) определяются следующими формулами:

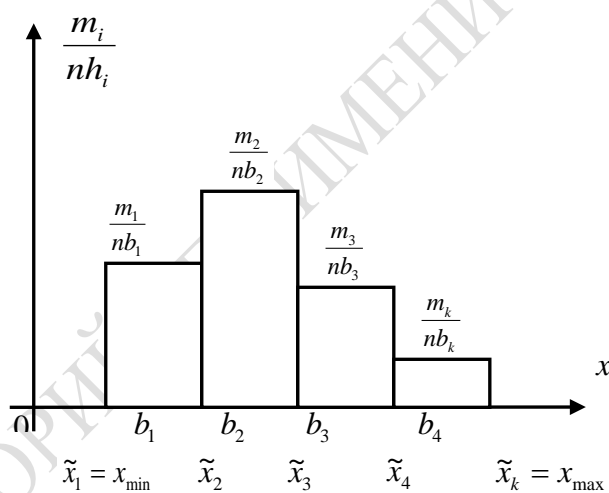
$$h_i = h = (x_{\max} - x_{\min})/k; C_i = x_{\min} + ih; i = 1, \overline{k-1} \quad (2.2)$$

Графическим изображением сгруппированного статистического ряда дискретной случайной величины является столбчатая диаграмма

(рисунок 2.1, а), а для интегрального статистического ряда непрерывной случайной величины используется гистограмма (рисунок 2.1, б).



а) столбиковая диаграмма



б) гистограмма

Рисунок 2.1 – Графическое изображение статистического ряда

Величина столбцов диаграммы (рисунок 2.1, а) определяется относительной частотой (m_i / n) появления в выборке соответствующего значения дискретной X , а высота прямоугольников гистограммы (рисунок 2.1, б) равна $m_i / (nh_i)$. Таким образом, относительная частота $v_i = m_i / n$ попадания непрерывной случайной величины в каждый из интервалов равна площади соответствующего прямоугольника.

Эмпирической функцией распределения случайной величины x называется функция $F^*(x)$ действительного аргумента x , которая для каждого x_i определяет относительную частоту события $(x_i < x)$, т. е. равна отношению количества экспериментов, в котором величина x_i приняла значение меньше x , к общему числу проведенных экспериментов

$$F^*(x) = \frac{n_{x_i < x}}{n}, \quad (2.3)$$

где $n_{x_i < x}$ – число элементов выборки объема n меньше x .

Эмпирическая функция распределения $F^*(x)$ является опытным аналогом теоретической функции распределения $F(x)$ случайной величины x (рисунок 2.2). Для нее выполняются все свойства функции распределения, рассмотренные в 1.1. Вместе с тем, эмпирическая функция распределения $F^*(x)$ является ступенчатой функцией. Точки разрыва $F^*(x)$ совпадают со значениями элементов выборки x_i , исследуемой величины X .

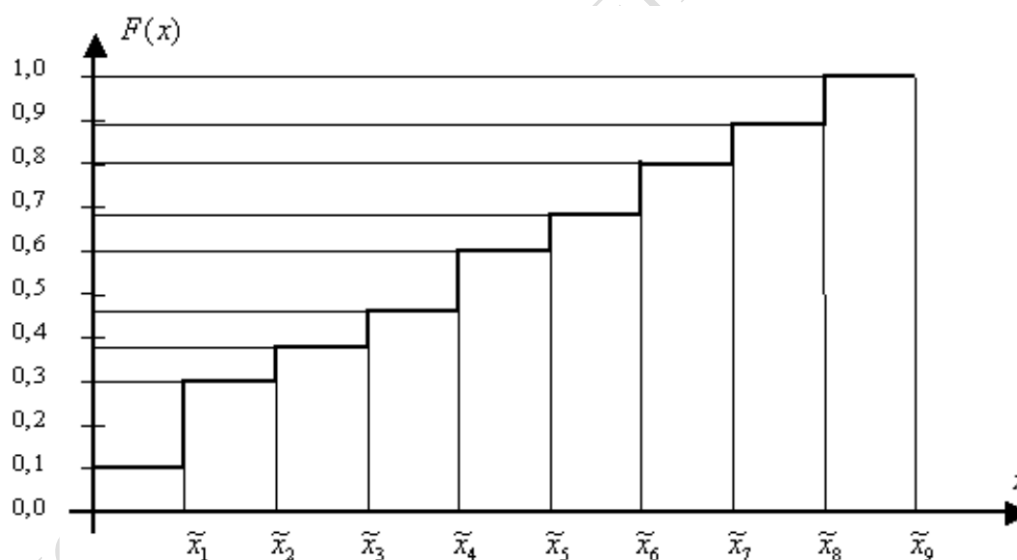


Рисунок 2.2 – Эмпирическая функция распределения случайной величины

2.2. Элементы статистического оценивания результатов исследований

Случайные величины наряду с законом распределения характеризуются числовыми характеристиками, поэтому вместе с определением

статистического закона распределения встает задача оценки по выборке числовых характеристик. Кроме того, часто на практике найденный закон распределения пытаются аппроксимировать одним из типовых законов. В этом случае необходима оценка параметров предполагаемого распределения по выборке. *Задача оценивания* неизвестного параметра θ распределения случайной величины x состоит в построении функции V от элементов выборки x_1, x_2, \dots, x_n :

$$\theta \approx V(x_1, x_2, \dots, x_n), \quad (2.4)$$

которую называют выборочной функцией, а найденное значение этой функции $\hat{\theta} = V(x_1, x_2, \dots, x_n)$ называют оценкой неизвестного параметра θ . Поскольку выборка имеет ограниченный объем n , а элементы ее случайны, то сами оценки – это также случайные величины. Различают два вида оценок: точечные, которые характеризуют оцениваемый параметр θ одним числом $\hat{\theta} = V(x_1, \dots, x_n)$; интервальные, определяемые двумя числами $\hat{\theta}_1$ и $\hat{\theta}_2$, являющимися границами интервала $(\hat{\theta}_1, \hat{\theta}_2)$, накрывающего значение оцениваемого параметра с заданной вероятностью P .

Точечная оценка числовых характеристик θ называется *состоятельной*, если при увеличении числа испытаний она сходится по вероятности к значению оцениваемого параметра. Она бывает *несмещенной*, если ее математическое ожидание равно значению оцениваемого параметра ($M[\hat{\theta}] = \theta$). Точечная несмещенная оценка параметра называется *эффективной*, если она обладает минимальной дисперсией.

Приведем формулы для вычисления оценок числовых характеристик случайных величин с указанием свойств полученных оценок. В качестве оценки математического ожидания случайной величины x используется среднее арифметическое \bar{x} элементов выборки $\{x_i\}$, $i = 1, n$, определяемое по известной формуле:

$$\hat{M}[x] = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad (2.5)$$

Она состоятельная и несмещенная. Если величина x имеет нормальный закон распределения, то эта оценка является эффективной. Для определения несмещенной и состоятельной оценки дисперсии случайной величины используется статистика:

$$\hat{D}[x] = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2; \quad (2.6)$$

$\hat{\sigma}[x] = \sqrt{\hat{D}[x]}$ – оценка среднеквадратичного отклонения.

Оценки коэффициента вариации $V[x]$, коэффициента асимметрии $A[x]$ и коэффициента эксцесса $E[x]$ случайной x определяются выражениями [58]:

$$\hat{V}[x] = \frac{\hat{\sigma}[x]}{\bar{x}}. \quad (2.7)$$

Здесь \bar{x} и $\hat{\sigma}$ – оценки соответственно математического ожидания и среднеквадратичного отклонения случайной величины x .

$$\hat{A}[x] = \frac{n \sum_{i=1}^n (x_i - \bar{x})^3}{(n-1)(n-2)\hat{\sigma}^3[x]}; \quad (2.8)$$

$$\hat{E}[x] = \frac{n(n+1) \sum_{i=1}^n (x_i - \bar{x})^4 - 3(n-1) \left(\sum_{i=1}^n (x_i - \bar{x})^2 \right)^2}{(n-1)(n-2)(n-3)\hat{\sigma}^4[x]}. \quad (2.9)$$

Для оценки моды дискретной случайной величины используется значение сгруппированного статистического ряда $\text{mod} \hat{[x]}$, которому соответствует наибольшее значение частоты m_i . Оценка моды непрерывной случайной величины определяется по интервальному статистическому ряду серединой интервала, которому соответствует наибольшее значение $\frac{m_i}{n \cdot h_i}$, где (m_i – частота попадания значений непрерывной случайной величины в n -й интервал длиной h_i). Оценкой медианы непрерывной случайной величины $\text{med} \hat{[x]}$ является член номера $(n+1)/2$ (средний) вариационного ряда.

Интервальные оценки числовых характеристик. Отметим, что в силу случайности точечных оценок $\hat{\theta}$ неизвестно, насколько они отличаются от реального значения оцениваемого параметра θ . Поэтому

в ответственных исследованиях используются интервальные оценки. *Доверительной вероятностью* оценки параметра θ называется вероятность $P_{\text{дов}}$ выполнения неравенства $\hat{\theta}_1 < \theta < \hat{\theta}_2$. Интервал $(\hat{\theta}_1, \hat{\theta}_2)$, накрывающий неизвестный параметр θ с заданной вероятностью $P_{\text{дов}}$, называется доверительным интервалом. Длины этого интервала определяются $P_{\text{дов}}$ и объемом выборки n . При увеличении объема выборки n точечная оценка $\hat{\theta}$ вероятнее приблизится к значению оцениваемого параметра θ .

Рассмотрим общий алгоритм построения доверительного интервала, состоящий в выполнении следующей последовательности действий исследователя:

1. По выборке $\{x_i\}$, объема n ($i = \overline{1, n}$) определяется точечная оценка $\hat{\theta}$ числовой характеристики θ .

2. Составляется вспомогательная случайная величина (выборочная статистика) η , связанная с параметром θ и имеющая известную функцию плотности распределения $f(y)$.

3. Для заданной доверительной вероятности $P_{\text{дов}} = 1 - \alpha$ (α – уровень значимости) находят интервал $(y_1; y_2)$, для которого справедливо равенство: $P(y_1 < \eta < y_2) = P_{\text{дов}}$.

4. Проводят обратное преобразование величины η к значению параметра θ и находят интервал $(\hat{\theta}_1, \hat{\theta}_2)$, для которого $P(\hat{\theta}_1 < \theta < \hat{\theta}_2) = P_{\text{дов}}$. Полученный интервал является доверительным интервалом для оцениваемого параметра θ .

Рассмотрим примеры построения доверительных интервалов для числовых характеристик случайной величины, имеющей нормальный закон распределения.

Доверительный интервал математического ожидания случайной величины, нормально распределенной, с известным среднеквадратичным отклонением определяется следующей последовательностью действий:

1. Точечную оценку неизвестного математического ожидания находим по выборке $\{x_i\}$ ($i = \overline{1, n}$) объема n , используя (2.5).

2. Для построения доверительного интервала для параметра $M[x]$ используем вспомогательную случайную величину, полученную с помощью преобразования

$$U = \frac{\sqrt{n}}{\sigma(x)}(\bar{x} - M[x]). \quad (2.10)$$

Как и \bar{x} величина U , полученная из \bar{x} линейным преобразованием, также имеет нормальный закон распределения с математическим ожиданием $M[U] = 0$ и дисперсией $D[U] = 1$. Это означает, что U имеет стандартное нормальное распределение $N(0, 1)$.

3. Находим интервал $\left(U_{1-\frac{\alpha}{2}}; U_{\frac{\alpha}{2}} \right)$, в который с заданной вероятностью $P_{\text{дов}} = 1 - \alpha$ попадают значения вспомогательной случайной величины U , исходя из условий (рисунок 2.3): $P\left(U < U_{1-\frac{\alpha}{2}} \right) = \frac{\alpha}{2}$;

$$P\left(U > U_{\alpha/2} \right) = \frac{\alpha}{2}.$$

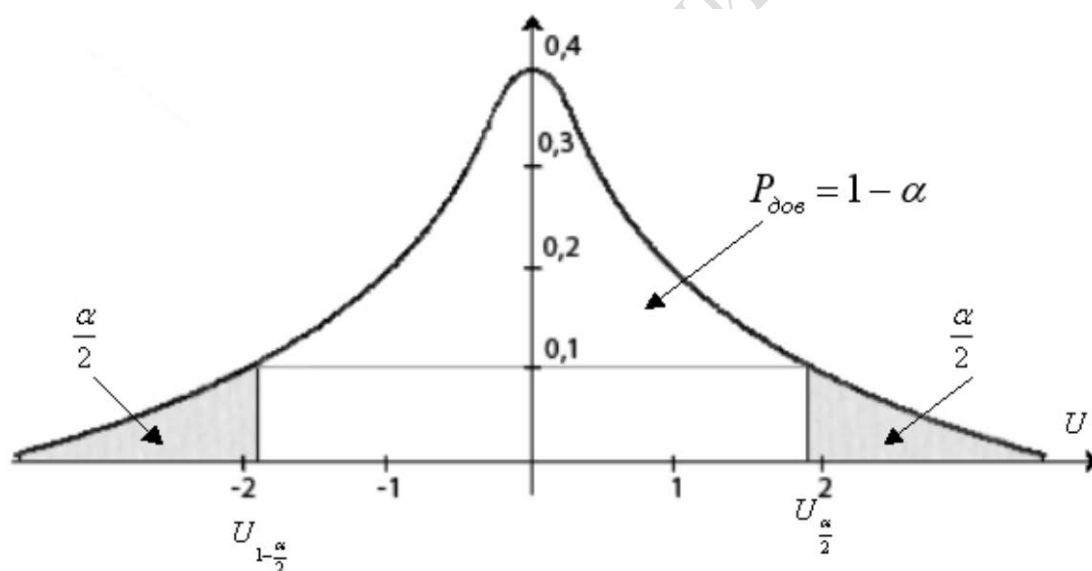


Рисунок 2.3 – Функция плотности распределения вспомогательной случайной величины U

Находим значения квантилей стандартной случайной величины U уровней $1 - \alpha/2$ и $\alpha/2$ из значений функций Лапласа:

$$\Phi\left(U_{\alpha/2} \right) = \frac{1-\alpha}{2}; \quad \Phi\left(U_{1-\frac{\alpha}{2}} \right) = \frac{1-\alpha}{2}.$$

4. Определяем интервал $\left(U_{1-\frac{\alpha}{2}}, U_{\frac{\alpha}{2}} \right)$, в который с вероятностью

$P_{\text{ДОВ}}$ попадают значения вспомогательной случайной величины U :

$$P_{\text{ДОВ}} = P \left(\underbrace{\bar{x} - \frac{\sigma[x]}{\sqrt{n}} U_{\frac{1-\alpha}{2}}}_{\theta_1} < M[x] < \underbrace{\bar{x} + \frac{\sigma[x]}{\sqrt{n}} U_{\frac{\alpha}{2}}}_{\theta_1} \right). \quad (2.11)$$

Таким образом, с точностью $\varepsilon = \frac{\sigma[x]}{\sqrt{n}} U_{\frac{\alpha}{2}}$ доверительный интервал определяется по формуле:

$$\bar{x} - \frac{\sigma(x)}{\sqrt{n}} U_{\frac{1-\alpha}{2}} < M[x] < \bar{x} + \frac{\sigma[x]}{\sqrt{n}} U_{\frac{\alpha}{2}}. \quad (2.12)$$

Доверительный интервал математического ожидания случайной величины, нормально распределенной с неизвестным среднеквадратическим отклонением по выборке малого объема ($n \leq 30$) определим с помощью следующей последовательности действий:

1. Для определения параметров $\hat{M}[x]$ и $\hat{\sigma}[x]$ используем выборку $\{x_i\}$ $i = \overline{1, n}$ и формулы (2.5 и 2.6).

2. Для построения доверительного интервала $M[x]$ с неизвестным $\sigma[x]$ используем t -статистику, имеющий распределении Стьюдента с $\nu = n - 1$ степенями свободы:

$$t = \frac{\sqrt{n}}{\hat{\sigma}[x]} (\bar{x} - M[x]). \quad (2.13)$$

3. Для заданной доверительной вероятности $P_{\text{ДОВ}} = 1 - \alpha$ находим интервал $\left(t_{1-\frac{\alpha}{2}, n-1}, t_{\frac{\alpha}{2}, n-1} \right)$, в который попадают значения вспомогательной случайной t из условий (рисунок 2.4)

$$P \left(t < t_{1-\frac{\alpha}{2}, n-1} \right) = \alpha/2 ; P \left(t > t_{\frac{\alpha}{2}, n-1} \right) = \frac{\alpha}{2}.$$

Находим значения квантилей t -распределения Стьюдента уровней

$1 - \alpha/2$ и $\alpha/2$. На основании симметрии распределения Стьюдента эти квантили равны $\left(t_{1-\frac{\alpha}{2}, n-1} = -t_{\frac{\alpha}{2}, n-1} \right)$.

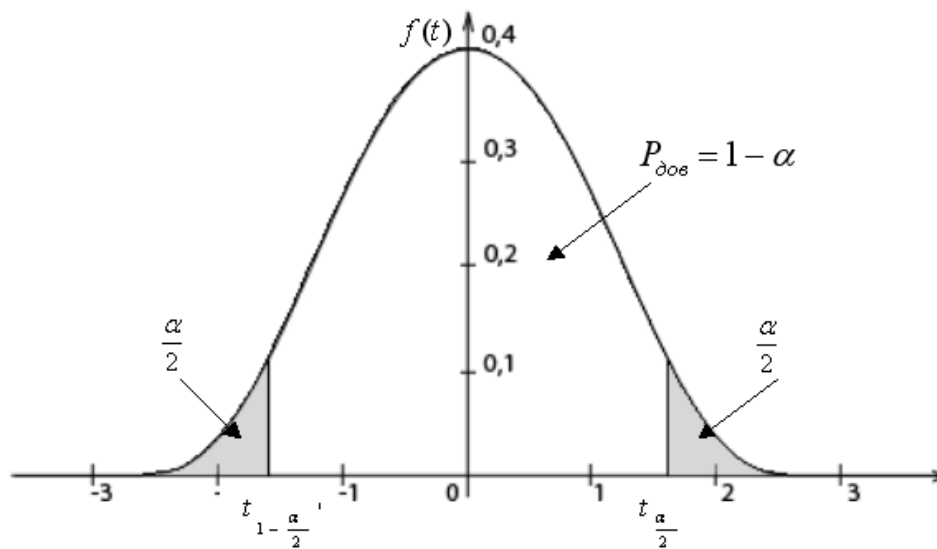


Рисунок 2.4 – Функция плотности распределения Стьюдента

4. Находим интервал $\left(t_{1-\frac{\alpha}{2}, n-1}; t_{\frac{\alpha}{2}, n-1} \right)$, который с доверительной вероятностью $P_{\text{довер}}$ покрывает значение t :

$$P_{\text{довер}} = P \left(\underbrace{\bar{x} - \frac{S[x]}{\sqrt{n}} t_{1-\frac{\alpha}{2}, n-1}}_{\theta_1} < \underbrace{M[x]}_{\theta} < \underbrace{\bar{x} + \frac{S[x]}{\sqrt{n}} t_{\frac{\alpha}{2}, n-1}}_{\theta_2} \right), \quad (2.14)$$

где $S[x] = \hat{\sigma}[x]$; точность вычисления $\varepsilon = \frac{S[x]}{\sqrt{n}} t_{\alpha/2, n-1}$.

Таким образом, доверительный интервал математического ожидания определяется по формуле:

$$\bar{x} - \frac{S[x]}{\sqrt{n}} \cdot t_{\left(\frac{\alpha}{2}, n-1\right)} < M[x] < \bar{x} + \frac{S[x]}{\sqrt{n}} t_{\frac{\alpha}{2}, n-1}. \quad (2.15)$$

Доверительный интервал среднеквадратичного отклонения случайной величины, имеющей *нормальное* распределение, с *неизвестными* параметрами $M[x]$ и $\sigma[x]$ при выборке большого размера ($n > 30$), определяется с помощью следующих действий.

1. Для определения точечных оценок $M[x]$ и $\sigma[x]$ используем формулы (2.5) и (2.6). Поэтому с помощью этих формул определяют выборочные оценки этих параметров $\hat{M}[x] = \bar{x}$ и $\hat{\sigma}[x] = S[x]$.

2. Для построения доверительного интервала параметра распределения $\sigma[x]$ нормально распределенной величины x используем вспомогательную величину (выборочную статистику):

$$\chi^2 = \frac{(n-1)S[x]^2}{\sigma^2[x]}, \quad (2.16)$$

которая имеет хи-квадрат распределение с $\nu = (n - 1)$ степенями свободы.

3. Для заданной доверительной вероятности $P_{\text{дов}} = 1 - \alpha$ находим доверительный интервал $(\chi_{1-\alpha/2, n-1}^2; \chi_{\alpha/2, n-1}^2)$, в который попадают значения вспомогательной случайной величины χ^2 из условий (рисунок 2.5):

$$P(\chi^2 < \chi_{1-\alpha/2, n-1}^2) = \int_0^{\chi_{1-\alpha/2, n-1}^2} f(\chi^2) d\chi^2 = \frac{\alpha}{2},$$

$$P(\chi^2 > \chi_{\alpha/2, n-1}^2) = \int_{\chi_{\alpha/2, n-1}^2}^{\infty} f(\chi^2) d\chi^2 = \frac{\alpha}{2}. \quad (2.17)$$

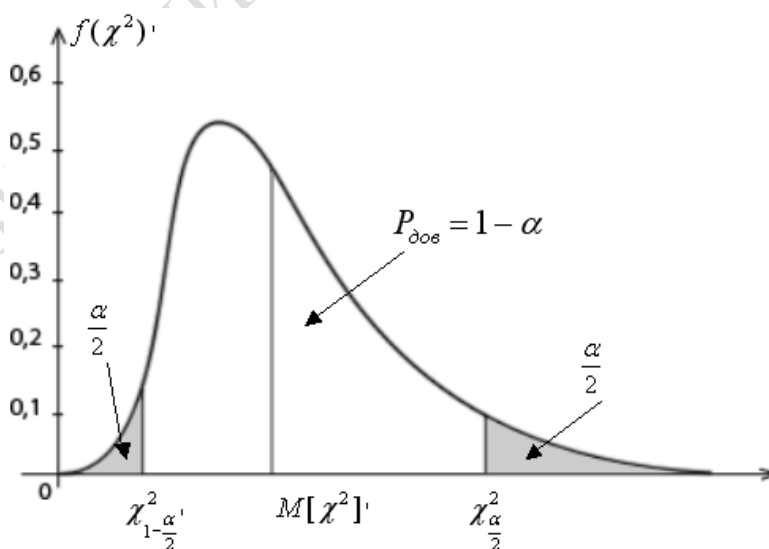


Рисунок 2.5 – Функция плотности χ^2 распределения

Квантили распределения χ^2 уровней $\left(1 - \frac{\alpha}{2}\right)$ и $(\alpha/2)$ для различного числа степеней свободы $(n - 1)$ определяются по соответствующим таблицам или с помощью программ определения квантилей распределения хи-квадрат.

4. Определим интервал $\left(\chi^2_{1-\frac{\alpha}{2}, n-1}; \chi^2_{\alpha/2, n-1}\right)$, в который с доверительной вероятностью попадают значения величины χ^2 :

$$P_{\text{ДОВ}} = \left(\underbrace{S[\chi] \sqrt{\frac{n-1}{\chi^2_{\frac{\alpha}{2}, n-1}}}}_{\hat{\theta}_1} < \underbrace{\sigma[\chi]}_{\theta} < S[\chi] \sqrt{\frac{n-1}{\chi^2_{1-\frac{\alpha}{2}, n-1}}} \right). \quad (2.18)$$

Поэтому

$$S[\chi] - S[\chi] \sqrt{\frac{n-1}{\chi^2_{\frac{\alpha}{2}, n-1}}} < \sigma[\chi] < S[\chi] + S[\chi] \sqrt{\frac{n-1}{\chi^2_{1-\frac{\alpha}{2}, n-1}}}. \quad (2.19)$$

2.3. Статистическая проверка гипотез

Определение статистического закона распределения, точечных и интервальных оценок числовых характеристик случайных величин является предварительной стадией статистического анализа. Конечная цель исследования заключается, например, в сравнительной оценке различных процессов по разным критериям. Из-за ограниченного объема и случайности выборки при проверке статистических гипотез могут быть совершены ошибки. Отклонение верной нулевой гипотезы H_0 называется *ошибкой 1-го рода*. Принятие ложной нулевой гипотезы H_0 называется *ошибкой 2-го рода*. *Уровнем значимости α* статистического критерия называется вероятность совершения ошибки 1-го рода. *Мощностью статистического критерия* называется вероятность $(1 - \beta)$ несовершения ошибки 2-го рода, т. е. вероятность отклонения неверной гипотезы. Пусть для проверки гипотезы H_0 относительного значения параметра распределения θ величины x используется стати-

стический критерий K . Предположим, что в случае истинности или верности гипотезы H_0 , функция плотности распределения критерия K равна $f(K/H_0)$. Если же верна гипотеза H_a , то статистический критерий K имеет функцию плотности распределения $f(K/H_a)$. Разделим область возможных значений критерия K на две непересекающиеся области: область допустимых значений (ОДЗ); область критических значений (ОКЗ). В случае справедливости H_0 (а не альтернативной H_a) гипотезы вероятность попадания критерия \hat{K} в ОКЗ равна заранее фиксированному значению α , вероятность его попадания в область ОДЗ равна соответственно $(1 - \alpha)$. На рисунке 2.6 приведены функции плотности распределения статистического критерия при условии справедливости нулевой или альтернативной гипотезы. Если известно, что $M[K/H_a] > M[K/H_0]$ (рисунок 2.6), то критическое значение статистического критерия $K_{\text{крит}}$ определяется из условия:

$$P(K < K_{\text{крит}}) = \int_{-\infty}^{K_{\text{крит}}} f(K/H_0) dk = \alpha. \quad (2.20)$$

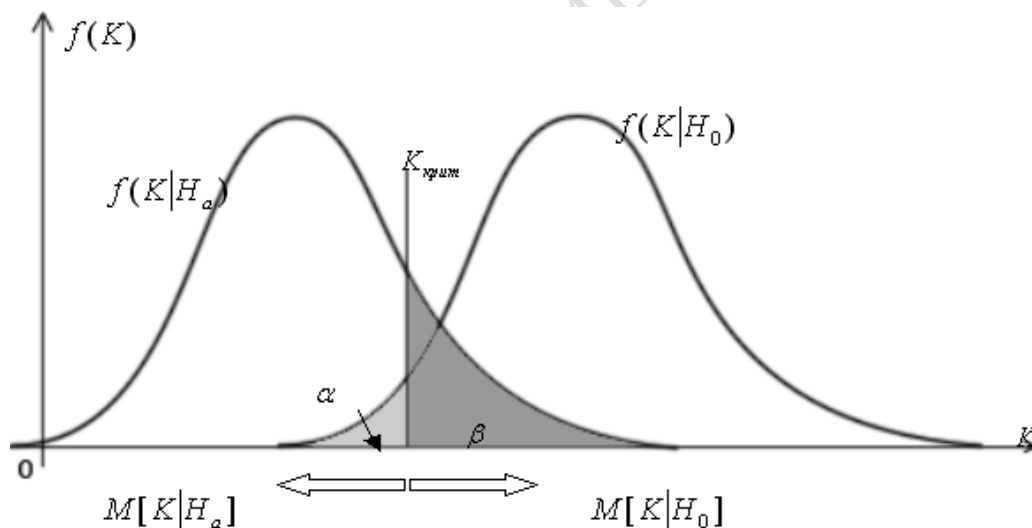


Рисунок 2.6 – Функция плотности распределения статистического критерия при условии справедливости нулевой или альтернативной гипотезы

Если в случае верности нулевой и альтернативной гипотез функция плотности распределения статистического критерия K имеет вид, представленный на рисунке 2.7, то критические значения $K_{\text{крит}1}$ и $K_{\text{крит}2}$ определяются из условия:

$$P(K < K_{\text{крит1}}) = P\left(K < K_{1-\frac{\alpha}{2}}\right) = \int_{-\infty}^{k_{1-\alpha/2}} f(k/H_0) dk = \alpha/2; \quad (2.21)$$

$$P(K > K_{\text{крит1}}) = P(K > K_{\alpha/2}) = \int_{K_{\alpha/2}}^{\infty} f(k/H_0) dk = \alpha/2. \quad (2.22)$$

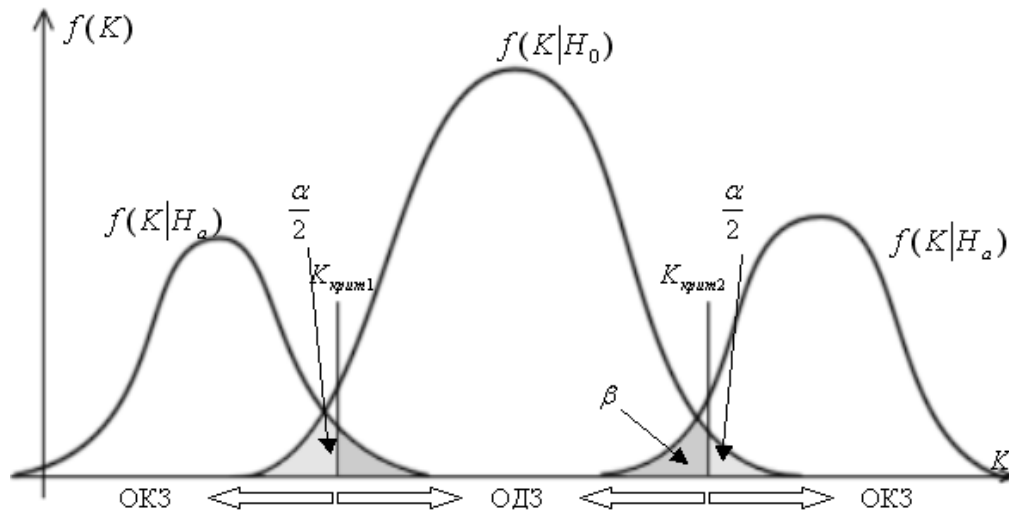


Рисунок 2.7 – Пример двусторонней области критических значений

Если гипотеза H_0 верна, то вероятность того, что значение статистического критерия K не попадет в ОДЗ, равна α . Зададим вероятность α очень малой, чтобы попадание K в ОКЗ можно было считать практически невозможным событием. Тогда, если нулевая гипотеза H_0 справедлива, то при ее проверке выборочное значение критерия K должно обязательно попасть в ОДЗ. Если этого не происходит (т. е. наблюдаемое значение критерия K принадлежат ОКЗ), то с вероятностью ошибки α , близкой к нулю, можно утверждать, что проверяемая гипотеза H_0 является ложной. В случае справедливости нулевой гипотезы H_0 функция плотности распределения статистического критерия имеет вид $f(k/H_0)$, а если справедлива противоположная гипотеза H_a , то функция плотности критерия K имеет вид $f(k/H_a)$ (рисунок 2.6). Для альтернативной гипотезы и уровня значимости α формулой (2.20) определяются области ОКЗ и ОДЗ статистического критерия K . Выбор значения α зависит от потерь, вызываемых ошибками 1-го и 2-го рода. Так, если совершение ошибки 1-го рода приведет к большим потерям, чем от ошибки 2-го рода, то следует уменьшить α (рисунок 2.6). При этом вероятность β совершения ошибки 2-го рода увеличится. Вопрос о значении уровней значимости α зависит от целей исследования, а также выбора нулевой и альтернативной гипотезы. Обычно в инженерных исследованиях задаются уровнем значимости $\alpha = 0,05$, а в медицинских исследованиях он задается $\alpha = 0,01$.

Использование статистических критериев значимости. При статистической проверке гипотез, когда одна из гипотез (H_0 или H_a) является сложной (содержит множество предположений), условное распределение статистического критерия $f(K/H_0)$ или $f(K/H_a)$ точно не известно (рисунок 2.8). Поэтому нельзя точно контролировать ошибку 1-го или 2-го рода. Обычно при статистической проверке гипотез H_0 выбирают простой, а альтернативную H_a считают сложной. В этом случае ограничиваются учетом информации лишь о вероятности совершения ошибки 1-го рода (рисунок 2.8). Распределение статистического критерия K при условии вероятности сложной альтернативной гипотезы $f(K/H_a)$ неизвестно, и вероятность свершения ошибки 2-го рода β не определена и поэтому часто не учитывается.

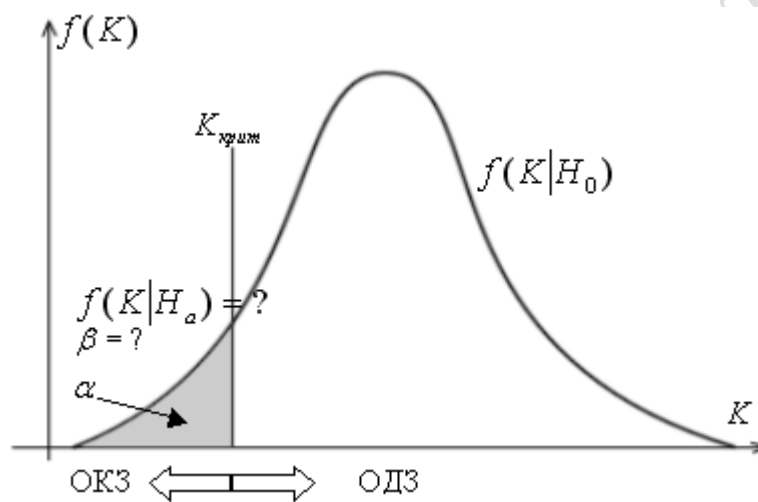


Рисунок 2.8 – Использование статистического критерия значимости при проверке простой гипотезы H_0

В дальнейшем нами рассматривается только один вид статистических критериев – статистические критерии значимости, при применении которых вероятность α зафиксирована заранее, а вероятность свершения ошибки 2-го рода β не контролируется. То есть, применяя статистические критерии значимости с заранее фиксированным риском α , принимается лишь одно из двух решений:

- нулевая гипотеза H_0 не согласуется с результатом эксперимента, поэтому H_0 отклоняется;
- нулевая гипотеза согласуется с результатами эксперимента, поэтому нет оснований для отклонения H_0 .

Отметим, что статистические критерии значимости не позволяют принимать решений о том, что гипотеза H_0 является верной, и она принимается (хотя не известно, с какой вероятностью принятая гипотеза H_0 является ложной).

Алгоритм статистической «проверки» гипотез с помощью критериев значимости реализуется следующей последовательностью шагов:

1. Формируется нулевая (проверяемая) H_0 и альтернативная H_a гипотезы. При этом H_0 должна быть простой.

2. Подбирается статистический критерий значимости K (вспомогательная случайная величина), который в случае справедливости гипотезы H_0 имеет заранее известное распределение.

3. Множество возможных значений статистического критерия значимости K разбивается (с учетом альтернативной гипотезы) на два непересекающихся множества: ОДЗ и ОКЗ. Область ОКЗ выбирается из условия $P(K \notin \text{ОКЗ}) = \alpha$.

4. По результатам n испытаний определяется выборочное значение \hat{K} статического критерия значимости K .

5. Если \hat{K} принадлежит ОДЗ, то формулируется вывод: нулевая гипотеза H_0 согласуется с опытными данными, следовательно, оснований для отклонения гипотезы H_0 нет. В противном случае (если \hat{K} принадлежит ОКЗ) считают, что нулевая гипотеза H_0 не согласуется с опытными данными, следовательно, гипотеза H_0 отклоняется.

Рассмотрим наиболее употребительные примеры применения этих критериев.

1⁰. Проверка гипотезы H_0 о **значении** $M[x] = A_0$ случайной величины, имеющей *нормальное* распределение с *известным* $\sigma[x]$. Для проверки гипотезы H_0 используем статистический критерий значимости:

$$U = \frac{\sqrt{n}}{\sigma[x]}(\bar{x} - A_0).$$
 Если гипотеза H_0 верна, критерий значимости U

наиболее вероятно имеет значение, близкое к нулю. Если верна альтернативная гипотеза H_a ($M[x] \neq A_0$), то критерий U будет значительно отличаться от нуля. На множестве возможных значений статистического критерия $\{U\}$ определим ОКЗ, которая является характерной для критерия U в случае справедливости H_a . Размер ОКЗ определим из условия $P(U \in \text{ОКЗ}/H_0) = \alpha$. Учитывая симметричность $f(U)$ относительно нуля, критическое значение статистического критерия U определяется соотношением $U_{\text{крит}} = U_{\alpha/2}$, где $U_{\alpha/2}$ – квантиль стандартного нормального распределения $N(0,1)$ уровня $\alpha/2$. Далее определя-

ется выборочное значение $\hat{U} = \frac{\sqrt{n}}{S}(\bar{x} - A_0)$. Если выполняется нера-

венство $U_{\alpha/2} < \hat{U} < U_{\alpha/2}$, то гипотеза $H_0: M[x] = A_0$ считается согласующейся с результатами экспериментов. В противном случае H_0

не считается согласующейся с результатами экспериментов и она отклоняется в пользу альтернативной гипотезы – т. е. с вероятностью $(1 - \alpha)$ можно считать, что справедлива гипотеза $H_a : (M[x] \neq A_0)$. При отклонении гипотезы H_0 (когда $\bar{U} > U_{\frac{\alpha}{2}}$) с вероятностью α возможна

ошибка 1-го рода. Если $\hat{U} < U_\alpha$, то гипотеза $H_0: M[x] = A_0$ считается согласующейся с результатами экспериментов. В противном случае гипотеза H_0 считается не согласующейся с результатами экспериментов, она отклоняется в пользу альтернативной гипотезы. То есть, с вероятностью $(1 - \alpha)$ можно утверждать, что справедлива гипотеза $H_a : M[x] > A_0$. При отклонении нулевой гипотезы H_0 (когда $\hat{U} > U_0$) с вероятностью α возможно совершение ошибки 1-го рода.

2⁰. Проверка H_0 о значении $M[x] = A_0$ случайной величины, имеющей нормальное распределение с неизвестным среднеквадратичным отклонением. Для проверки гипотезы H_0 используем

t -критерий значимости: $t = \frac{\sqrt{n}}{S}(\bar{x} - A_0)$, имеющий распределение

Стьюдента с $\nu = n - 1$ степенями свободы. Если H_0 верна, критерий t наиболее вероятно примет значение близкое к нулю. Если же верна H_a , то t будет значительно отличаться от нуля. На множестве возможных значений t определим ОКЗ, которые являются характерными для критерия t в случае справедливости альтернативной гипотезы. Размер ОКЗ определим из условия $P(t \in \text{ОКЗ}/H_0) = \alpha$. Учитывая симметричность $f(t)$ относительно нуля, критические значения t -критерия определим как квантиль распределения Стьюдента уровня $\alpha/2$ ($t_{\text{крит}} = t_{\alpha/2, n-1}$) с $\nu = n - 1$ степенями свободы. Определяем выборочное значение ста-

тистического критерия значимости $\hat{t} = \frac{\sqrt{n}}{S[x]}(\bar{x} - A_0)$. Если выполняется

неравенство $-t_{\frac{\alpha}{2}, n-1} < \hat{t} < t_{\frac{\alpha}{2}, n-1}$, то гипотеза $H_0: M[x] = A_0$ согласуется

с результатами экспериментов. В противном случае H_0 не согласуется с результатами экспериментов и отклоняется в пользу альтернативной гипотезы, т. е. справедлива гипотеза $H_a : M[x] \neq A_0$. При отклонении H_0 (когда $|\hat{t}| > t_{\frac{\alpha}{2}, n-1}$) возможно свершение ошибки 1-го рода, однако

вероятность такой ошибки α мала.

3⁰. Проверка гипотезы H_0 о равенстве математических ожиданий двух независимых величин, имеющих нормальное распределение с известными среднеквадратичными отклонениями при больших

объемах выборок n_1 и n_2 . Величины x и y имеют распределения: $N(M[x], \sigma[x])$ и $N(M[y], \sigma[y])$, причем $\sigma[x]$ и $\sigma[y]$ известны. Нужно проверить гипотезу $H_0 : M[x] = M[y]$ против альтернативной гипотезы $H_a : M[x] \neq M[y]$. Другими словами, следует установить, значимо ли различаются \bar{x} и \bar{y} . Поскольку $x \sim N(M[x], \sigma[x])$ и $y \sim N(M[y], \sigma[y])$, то

$$\bar{x} \sim N\left(M[x], \frac{\sigma[x]}{\sqrt{n_1}}\right) \text{ и } \bar{y} \sim N\left(M[y], \frac{\sigma[y]}{\sqrt{n_2}}\right),$$

где n_1 и n_2 объемы выборок случайных величин x и y соответственно.

Составляется переменная $(\bar{x} - \bar{y})$, которая имеет нормальное распределение как линейная комбинация \bar{x} и \bar{y} , имеющих нормальное распределение. Если гипотеза $H_0 : M[x] = M[y]$ верна, то $M[\bar{x} - \bar{y}] = 0$. Когда x и y независимы, то независимы \bar{x} и \bar{y} . Поэтому дисперсия этой разности определяется по формуле:

$$D[\bar{x} - \bar{y}] = D[\bar{x}] + D[\bar{y}] = \left(\frac{\sigma[x]}{\sqrt{n_1}}\right)^2 + \left(\frac{\sigma[y]}{\sqrt{n_2}}\right)^2.$$

Используем статистический критерий значимости:

$$U = \frac{\bar{x} - \bar{y}}{\sqrt{D[\bar{x} - \bar{y}]}} = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{\sigma^2[x]}{n_1} + \frac{\sigma^2[y]}{n_2}}},$$

который подчиняется стандартному нормальному закону распределения. Если гипотеза H_0 верна, то $M[U] = 0$ и $D[U] = 1$. Таким образом, если $M[x] = M[y]$ и n большое, то можно использовать нормально распределенный статистический критерий значимости $U \sim N(0,1)$. Если гипотеза H_0 верна, критерий U наиболее вероятно примет значение, близкое к нулю. Если же верна альтернативная гипотеза $H_a : M[x] \neq M[y]$, то величина U примет очень большие значения. На множестве возможных значений статистического критерия U определим ОКЗ, которые являются характерными для критерия U в случае справедливости альтернативной гипотезы. Величину ОКЗ определим из условия $P(t \in \text{ОКЗ}/H_0) = \alpha$. Учитывая симметричность $f(U)$

относительно нуля, критические значения статистического критерия U определяются соотношением: $U_{\text{крит}} = U_{\alpha/2}$, где $U_{\frac{\alpha}{2}}$ – квантиль стандартного нормального распределения уровня $\alpha/2$. Выборочное значение статистического критерия значимости U определяется по формуле:

$$\hat{U} = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{\sigma^2[x]}{n_1} + \frac{\sigma^2[y]}{n_2}}}; \quad (2.23)$$

Если $U_{\frac{\alpha}{2}} < \hat{U} < U_{\frac{\alpha}{2}}$, то гипотеза $H_0 : M[x] = M[y]$ согласуется с результатами экспериментов. В противном случае H_0 считается не согласующейся с экспериментами, и она отклоняется в пользу гипотез H_a . То есть с вероятностью $(1 - \alpha)$ можно утверждать, что справедлива гипотеза $H_a : M[x] \neq M[y]$. При отклонении гипотезы H_0 , если $|\hat{U}| > U_{\alpha/2}$ возможно совершение ошибки 1-го рода с вероятностью α .

4⁰. Проверка гипотезы H_0 о равенстве математических ожиданий двух независимых величин, нормально распределенных с неизвестными, но одинаковыми среднеквадратичными отклонениями. Известно, что $x \sim N(M[x], \sigma[x])$; $y \sim N(M[y], \sigma[y])$; $\sigma[x] = \sigma[y]$, но истинное значение x неизвестно. Нужно проверить гипотезу $H_0 : M[x] = M[y]$ против альтернативной гипотезы $H_a : M[x] \neq M[y]$. То есть, следует установить, значимо или незначимо различаются \bar{x} и \bar{y} двух выборок. Для проверки H_0 используем t -критерий значимости:

$$t = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{n_1 S^2[x] + n_2 \cdot S^2[y]}{(n_1 + n_2 - 2)} \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}, \quad (2.24)$$

который (в случае справедливости H_0 имеет распределение Стьюдента с $v = (n_1 + n_2 - 2)$ степенями свободы, где n_1 и n_2 – объемы выборок величин x и y ; $S^2[x]$ и $S^2[y]$ – несмещенные оценки среднеквадратичных отклонений. Если H_0 верна, t -критерий наиболее вероятно примет значение, близкое к нулю. Если же верна $H_a : M[x] \neq M[y]$, то t -критерий примет очень большие или очень маленькие значения. На множестве возможных значений t -критерия определим ОКЗ, которые характерны для t -критерия в случае справедливости H_a . Величину

ОКЗ определим из условия $P(t \in \text{ОКЗ}/H_0) = \alpha$. Учитывая симметричность плотности распределения $f(t)$ относительно нуля, $t_{\text{крит}} = t_{\frac{\alpha}{2}, n_1+n_2-2}$,

где $t_{\frac{\alpha}{2}, n_1+n_2-2}$ – квантиль распределения Стьюдента уровня $\alpha/2$ с

$\nu = n_1 + n_2 - 2$ степенями свободы. Выборочное значение t -критерия определяется по формуле:

$$\hat{t} = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{n_1 S^2[x] + n_2 S^2[y]}{n_1 + n_2 - 2} \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}. \quad (2.25)$$

Итак, если $-t_{\frac{\alpha}{2}, \nu} < \hat{t} < t_{\frac{\alpha}{2}, \nu}$, то гипотеза $H_0: M[x] = M[y]$ согласуется с результатом экспериментов. В противном случае H_0 не согласуется с результатами экспериментов, она отклоняется в пользу гипотезы H_a . То есть с вероятностью $(1 - \alpha)$ можно утверждать, что справедлива гипотеза $H_a: M[x] \neq M[y]$. При отклонении H_0 (если $|\hat{t}| > t_{\alpha/2, \nu}$) возможна ошибка 1-го рода с вероятностью α .

2.4. Проверка непараметрических гипотез

Все множество задач статистической проверки непараметрических гипотез условно разделяется на две группы:

1) проверка гипотезы относительно вида функции распределения, которая помогает ответить на вопрос, имеет ли исследуемая случайная величина x некоторый закон распределения, например, $H_0: F_{\xi}(x) = 1 - \exp(-\lambda x), x > 0$;

2) проверка гипотезы о принадлежности двух выборок одной генеральной совокупности, т. е. гипотезы $H_0: F_{\xi}(x) = F_{\eta}(x)$ (о равенстве функций распределения двух случайных величин).

Статистическая проверка непараметрических гипотез выполняется с помощью непараметрических критериев значимости (вспомогательных случайных величин, которые при условии верности гипотезы H_0 имеют заранее известное распределение). Если выборочное значение статистического критерия значимости K будет принадлежать области маловероятностных критических значений (ОКЗ), то гипотеза H_0 отклоняется как не согласующаяся с экстремальными данными. Если K принадлежит к области ОДЗ, то H_0 согласуется с данными эксперимента.

Для проверки гипотез 1-го типа (о виде функции распределения) используются критерии «согласия» (критерий χ^2 -Пирсона, λ -Колмогорова, ω^2 -Мизеса, λ -Пирсона), а для проверки гипотез 2-го типа (о равенстве функций распределения) используют критерии «однородности» (критерии Смирнова и Вилкоксона). Как правило, альтернативной гипотезой является сложная: $H_a: F_\xi(x) \neq F^*(x)$ или $H_a: F_\xi(x) \neq F_\eta(x)$. Поэтому распределение статистического критерия в случае верности H_a однозначно неизвестно, и при исследовании гипотезы не контролируется вероятность β совершения ошибки 2-го рода. Поэтому выводы о принятии гипотезы H_0 вовсе не делают. Рассмотрим подробнее применение двух критериев согласия χ^2 -Пирсона и λ -Колмогорова.

Критерий согласия Пирсона χ^2 позволяет проверить согласование гипотетического (предполагаемого) и эмпирического (опытного) законов распределения. Последовательность анализа следующая:

1. Выполняется первичный анализ данных состояний, заключающийся в:

- построении статистического ряда распределения случайной величины x (сгруппированного, если x дискретна, или интервального, если x непрерывна);

- оценивании числовых характеристик величины x ;

- на основании предметной постановки задачи, визуального анализа статистического ряда распределения и анализа числовых характеристик выдвигается гипотеза о виде закона распределения величины x , т. е. $H_0: F_\xi(x) = F^*(x)$; $H_a: F_\xi(x) \neq F^*(x)$; где $F^*(x)$ – гипотетическая функция распределения;

- определение оценок параметров гипотетической функции распределения $F^*(x)$.

2. Для проверки гипотезы о виде функции распределения $F^*(x)$ величины x используется статистика вида:

$$\chi^2 = \sum_{i=1}^k \frac{(m_i - np_i)^2}{np_i}, \quad (2.26)$$

который (в случае варианта H_0) независимо от вида $F^*(x)$ имеет распределение хи-квадрат с $\nu = (k - r - 1)$ степенями свободы. Здесь k – количество интервалов статического рода, r – количество параметров $F^*(x)$; m_i – количество значений x , попавших в i -й интервал группирования; n – объем выборки $\{x\}$, p_i – вероятность попадания значений x в i -й интервал при условии $F_\xi(x) = F^*(x)$.

3. Попадание значений x в i -й интервал статистического рода распределения есть испытание Бернулли. Следовательно, m_i имеет биномиальный закон распределения с параметрами n и p_i . Считаем, что $M[m_i] = np_i$. Поэтому выражение (2.25) вычисляет отклонение эмпирической частоты попаданий (m_i) значений x_i в i -й интервал от гипотетической частоты np_i при условии верности $H_0 : F_{\xi}(x) = F^*(x)$. Если H_0 верна, то данные отклонения незначительны и критерий значимости χ^2 принимает малые значения. Если же H_0 не верна, то отклонения эмпирических и гипотетических частот будут значительны, а $\chi^2 \rightarrow \infty$. Таким образом, ОКЗ критерия χ^2 (рисунок 2.9) является правосторонним. Для заданного уровня значимости α определяется ОКЗ из условия $P(\chi^2 \in \text{ОКЗ}) = P(\chi^2 > \chi_{\text{крит}}^2) = \alpha$. Значение $\chi_{\text{крит}}^2 = \chi_{\alpha, v}^2$ – квантиль распределения χ^2 с $v = k - r - 1$ степенями свободы уровня α .

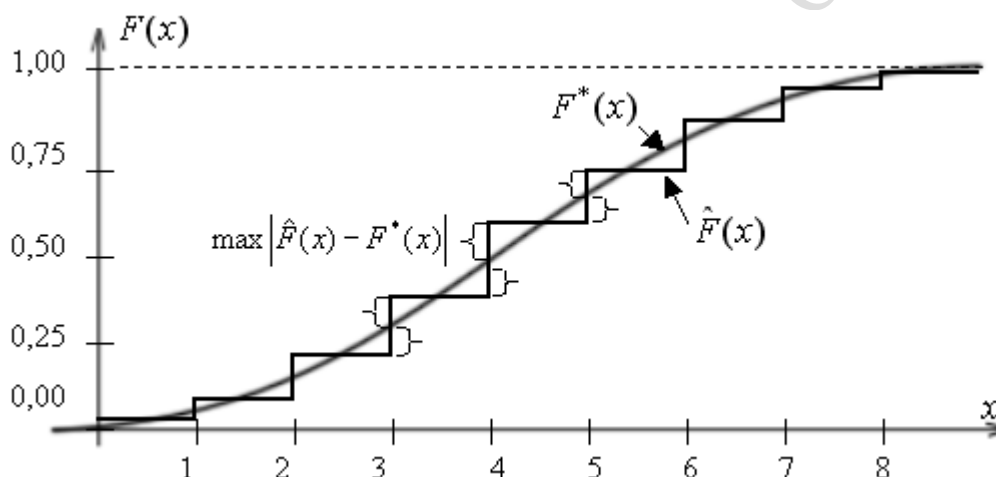


Рисунок 2.9 – Определение максимального отклонения гипотетической и эмпирической функции распределения величины x

4. Рассчитывается выборочное значение статистического критерия значимости по формуле (2.26).

5. Если гипотеза $H_0 = F_{\xi}(x) = F^*(x)$ верна, то $\hat{\chi}^2 \leq \chi_v^2$ и значит $P(\hat{\chi}^2 \in \text{ОДЗ}) = P(\chi^2 < \chi_{\text{крит}}^2) = 1 - \alpha$. Если же $\chi^2 \in \text{ОКЗ}$, т. е. происходит практически невозможное событие, то с большой вероятностью $(1 - \alpha)$ можно утверждать, что H_0 верна. Итак, если $\chi^2 \leq \chi_{\text{крит}}^2$, то H_0 считается согласующейся с результатами эксперимента, а если $\hat{\chi}^2 > \chi_{\text{крит}}^2$, то H_0 не согласуется с опытными данными, и она отклоняется с вероятностью $(1 - \alpha)$.

Критерий согласия λ Колмогорова применяется для проверки гипотез $H_0: F_\xi(x) = F^*(x)$ о виде функции распределения только непрерывных случайных величин, когда параметры гипотетической функции распределения $F^*(x)$ точно известны. В отличие от критерия χ^2 -Пирсона, где сравниваются гипотетическое и эмпирическое частоты (np_i и m_i) попадания значений величины в интервалы статистического ряда распределения, критерий согласия λ Колмогорова сравнивает гипотетические и эмпирические функции распределения $F^*(x)$ и $\hat{F}(x)$. Для проверки гипотезы H_0 используется статистический критерий значимости λ , равный максимальному отклонению эмпирической и гипотетической функции распределения:

$$\lambda = \max_x \left| \hat{F}(x) - F^*(x) \right| \cdot \sqrt{n}, \quad (2.27)$$

где $\hat{F}(x)$ и $F^*(x)$ – значения соответственно эмпирической и гипотетической функций распределения соответственно;

n – объем выборки.

Например, если $\hat{F}(x)$ и $F^*(x)$ имеют вид, представлений на рисунке 2.9 (при этом предполагается нормальность распределения x), то выборочное значение критерия λ Колмогорова определяется следующим образом:

$$\hat{\lambda} = \max \left\{ \max_{1 \leq i \leq n} \left| \frac{i-1}{n} F^*(x_i) \right|, \max_{1 \leq i \leq n} \left| \frac{i}{n} - F^*(x_i) \right| \right\} \cdot \sqrt{n}; \quad (2.28)$$

где i – номер элемента вариационного ряда;

n – объем выборки;

$F^*(x_i)$ – значение гипотетической функции распределения в точке x_i ;

x_i – значение i -го элемента вариационного ряда.

А. Н. Колмогоров доказал, что в случае справедливости гипотезы $H_a: F_\xi(x) = F^*(x)$ при $n \rightarrow \infty$, статистический критерий λ имеет функцию

распределения Колмогорова $F_\lambda(x) = \sum_{k=-\infty}^{\infty} (-1)^k \exp(-2k^2 x^2)$, которая не

зависит от $F^*(x)$. При идеальном согласовании гипотезы с выборочными данными λ равно нулю. В противном случае λ принимает большие положительные значения. Если наблюдается $\hat{\lambda} < \lambda_{\text{крит}}$, определяемо

для уровня значимости α и объема n , то гипотеза H_0 согласуется с результатами опытов и оснований для ее отклонения нет. В противном случае (когда $\hat{\lambda} > \lambda_{\text{крит}}$) гипотеза H_0 отклоняется, и с вероятностью ошибки α можно утверждать, что верна альтернативная гипотеза $H_a: F_{\xi}(x) \neq F^*(x)$. Несмотря на вышеупомянутые ограничения, критерий λ Колмогорова, по сравнению с критерием χ^2 -Пирсона, обладает рядом преимуществ. Во-первых, критерий λ -Колмогорова построен на индивидуальных (а не сгруппированных) значениях случайных величин (x), поэтому результаты анализа не зависят от способа групповых данных. При этом из исследования исключается субъективный фактор. Во-вторых, критерий λ -Колмогорова можно применять для проверки гипотезы о виде распределения случайных величин x на основе малого объема выборки (от 20 до 30 элементов).

2.5. Элементы статистического анализа данных

При исследовании многомерных случайных величин основной задачей является анализ зависимости между ее компонентами (одномерными случайными величинами). Эти исследования нужны для предсказания значений одной величины по значению другой. Существует два класса зависимостей между величинами: функциональная и статистическая. Зависимость между случайными величинами ξ и η , которая каждому значению x величины ξ ставит в однозначное соответствие единственное значение y величины η , называется *функциональной*. Однако часто на практике одному значению величины ξ может соответствовать не одно, а множество значений величины η , характеризующихся для каждого $\xi = x$ условным распределением с плотностью вероятностей $f_{\eta}(y/\xi = x)$. Такая зависимость называется *статистической*. Информация о совместной функции плотности распределения $f_{\xi\eta}(x, y)$ двумерной величины (ξ, η) или функции плотности распределения $f_{\eta}(y/\xi = x)$ величины η , при условии, что ξ приняла некоторое значение $x(\xi = x)$, полностью определяет статистическую зависимость между величинами. Однако на практике такая информация часто бывает излишней, а для описания зависимости между величинами используют регрессионную зависимость. *Регрессионной* называют зависимость между значениями одной случайной величины ξ и условным математическим ожиданием другой величины $M[\eta/\xi = x]$ при условии, что ξ приняла значение x .

Предметом регрессионного анализа является нахождение вида уравнения зависимости между случайными величинами, а предметом *корреляционного анализа* является оценка тесноты связи между случайными величинами и качества описания зависимости выбранным уравнением регрессии.

Пусть дана выборка значений двумерной случайной величины $(\xi, \eta) = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, где n – объем двумерной выборки. Первым шагом в построении уравнения регрессии между величинами является графическое отображение значений двумерной случайной величины в виде точек $(x_1, y_1), \dots, (x_n, y_n)$ на плоскости XOY . Такое изображение называется *диаграммой рассеяния* (или *корреляционным полем*). Визуальный анализ диаграммы рассеяния, а также предметная постановка задачи позволяют сделать предположение о виде уравнения регрессии. Например, по характеру расположения точек можно сделать предположение о линейном, гиперболическом, параболическом и других видах уравнения регрессии.

Если предполагается, что зависимость между величинами ξ и η линейна, то модель регрессионной зависимости задается уравнением линейной регрессии:

$$M[\eta/\xi = x] = \beta_0 + \beta_1 x, \quad (2.29)$$

т. е. для каждого $\xi = x$ определяется условное распределение величины η со средним значением $(\beta_0 + \beta_1 x)$. Таким образом для каждого i -го наблюдения справедлива следующая зависимость:

$$y_i = \beta_0 + \beta_1 x_i + e_i, \quad (2.30)$$

где y_i – i -е выборочное значение величины η ;

β_0 и β_1 – параметры линейной регрессии, которые нужно определить;

x_i – i -е выборочное значение величины ξ ; $i = \overline{1, n}$;

e_i – ошибка, вызванная отклонением i -го наблюдения величины η от условного среднего $M[\eta/\xi = x]$, обусловленная неучтенными факторами;

n – объем выборки.

Если зависимость между величинами ξ и η предполагается известной, то регрессионная модель между ними имеет один из следующих видов:

$$\begin{aligned} M[\eta/\xi = x] &= \beta_0 + \beta_1 \exp(\beta_2 x) - \text{для экспоненциальной зависимости;} \\ M[\eta/\xi = x] &= \beta_0 + \beta_1 x + \beta_2 x^2 - \text{для параболической зависимости;} \end{aligned} \quad (2.31)$$

$M[\eta/\xi = x] = \beta_0 + \beta_1/x$ – для гиперболической зависимости.

Вторым шагом в построении уравнения регрессии является определение параметров уравнения β_0, β_1, \dots . По выборке невозможно достоверно определить значения параметров регрессии, поэтому на практике ограничиваются их оценками $\hat{\beta}_0, \hat{\beta}_1, \dots$, найденными *методом наименьших квадратов* (МНК), суть которого заключается в следующем. Рассмотрим функцию $S(\beta_0, \beta_1, \dots)$, равную сумме квадратов отклонений e_i выборочных значений y_i величины η от значения $\tilde{y}(x_i)$, предсказанного уравнением регрессии в точке $\xi = x$ (рисунок 2.10):

$$S(\beta_0, \beta_1, \dots) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}(x_i))^2 ; \quad (2.32)$$

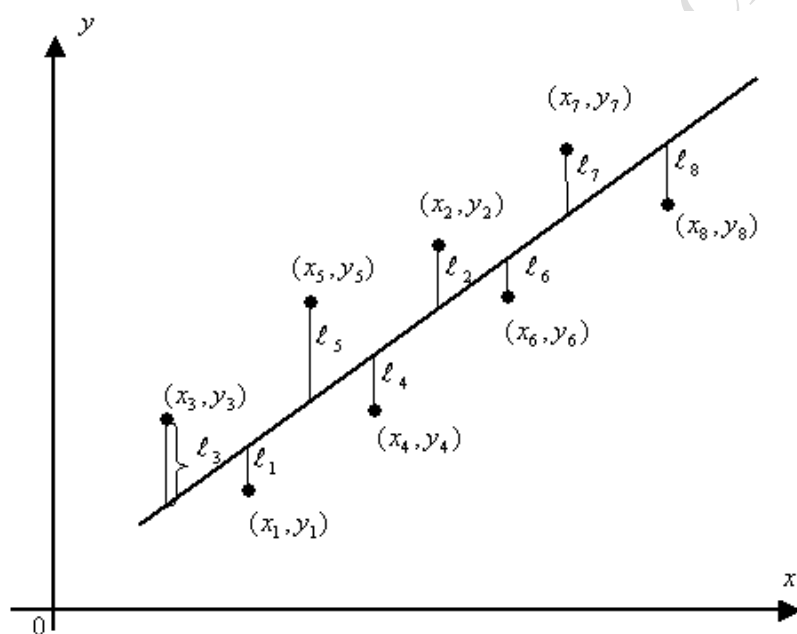


Рисунок 2.10 – Отклонения значений величины η от значений, представленных уравнением регрессии

Фактически отклонения e_i равны ошибке, которая обусловлена упрощением вида зависимости η от ξ (неучетом всех влияющих факторов), а также возможной ошибкой в выборе формы регрессии (например, она может описываться другим уравнением). Идея МНК заключается в определении таких значений параметров регрессии

β_0, β_1, \dots , которые минимизировали бы функцию $S(\beta_0, \beta_1, \dots) = \sum_{i=1}^n e_i^2 \rightarrow \min$.

Например, оценки $\hat{\beta}_0$ и $\hat{\beta}_1$ параметров линейной регрессии β_0 и β_1

получаются максимизацией соответственно по β_0 и β_1 суммы квадратов отклонений e_i выборочных значений величины η от предполагаемой прямой (рисунок 2.10):

$$S(\beta_0, \beta_1) = \sum_{i=1}^n (y_i - \hat{y}(x_i))^2 = \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2 \rightarrow \min. \quad (2.33)$$

Известно, что для нахождения минимума $S(\beta_0, \beta_1)$ необходимо приравнять к нулю частные производные (по неизвестным параметрам β_0, β_1, \dots) и решить систему уравнений, называемых нормальными. Система уравнений имеет единственное решение, если определитель матрицы ее коэффициентов не равен нулю. Полученные значения $\hat{\beta}_0, \hat{\beta}_1, \dots$, являются решением системы и называются оценками параметров регрессии.

Для линейной регрессии (2.30) указанная система имеет вид:

$$\begin{cases} \frac{\partial S(\beta_0, \beta_1)}{\partial \beta_0} = 0 \\ \frac{\partial S(\beta_0, \beta_1)}{\partial \beta_1} = 0 \end{cases} \Rightarrow \begin{cases} \frac{\partial}{\partial \beta_0} \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2 = 0 \\ \frac{\partial}{\partial \beta_1} \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2 = 0 \end{cases} \Rightarrow \begin{cases} -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) = 0, \\ -2 \sum_{i=1}^n x_i (y_i - \beta_0 - \beta_1 x_i) = 0. \end{cases}$$

После преобразований система этих уравнений принимает вид:

$$\begin{cases} \sum_{i=1}^n y_i - \beta_0 n - \beta_1 \sum_{i=1}^n x_i = 0 \\ \sum_{i=1}^n x_i y_i - \beta_0 \sum_{i=1}^n x_i - \beta_1 \sum_{i=1}^n x_i^2 = 0 \end{cases}. \quad (2.34)$$

Оценки β_0 и β_1 , минимизирующие ошибку, возникающую при аппроксимации точек на диаграмме рассеяния (рисунок 2.10) прямой, являющихся решением системы (2.32), вычисляются по формулам:

$$\beta_0 = \bar{y} - \hat{\beta}_1 \bar{x}; \beta_1 = \frac{\sum_{i=1}^n ((x_i - \bar{x})(y_i - \bar{y}))}{\sum_{i=1}^n (x_i - \bar{x})^2}, \quad (2.35)$$

$$\text{где } \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i;$$

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i.$$

В результате оценка уравнения линейной регрессии (эмпирическое уравнение регрессии, прямая МИК) будет иметь вид:

$$\bar{y}(x) = \hat{\beta}_0 + \hat{\beta}_1 x. \quad (2.36)$$

Коэффициенты корреляции и детерминации. Помимо предположения о форме регрессионной зависимости между случайными величинами и нахождения его параметров исследователю требуется оценить, насколько удачно выбранное уравнение регрессии объясняет (предсказывает) существующую зависимость между величинами. Основной числовой характеристикой, определяющей меру линейной регрессионной зависимости между двумя случайными величинами ξ и η , ее близость к линейной функциональной зависимости, является коэффициент корреляции $r_{\xi\eta}$. Оценка коэффициента корреляции по заданной двумерной выборке определяется выражением:

$$\hat{r}_{\xi\eta} = \frac{\sum_{i=1}^n ((x_i - \bar{x})(y_i - \bar{y}))}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \cdot \sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (2.37)$$

Для характеристики качества описания зависимости между случайными величинами произвольным (в том числе нелинейным) уравнением регрессии используется *коэффициент детерминации*. Очевидно, чем теснее результаты наблюдения примыкают к линии регрессии, тем лучше уравнение описывает существующую зависимость. И поэтому уравнение регрессии может быть применено для практических расчетов с большей надежностью. Коэффициент детерминации, обозначаемый $R_{\xi\eta}^2$, принимает значение на отрезке $[0, 1]$ и определяется выражением:

$$R_{\xi\eta}^2 = \frac{\sum_{i=1}^n (\bar{y}(x_i) - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad (2.38)$$

где n – объем выборки;

$\bar{y}(x_i)$ – среднее значение η , вычисленное по уравнению регрессии $\xi = x_i$ (условное среднее);

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \text{ – безусловное среднее } \eta;$$

y_i – i -е выборочное значение величины η .

Обычно используют другую формулу для определения коэффициента детерминации:

$$R_{\xi\eta}^2 = \frac{D[\bar{y}(x)]}{D[\eta]}, \quad (2.39)$$

где $D[\bar{y}(x)]$ – дисперсия значений, предсказываемых уравнением регрессии относительно среднего значения величины η ;

$D[\eta]$ – дисперсия случайной величины η .

Учитывая, что i -е выборочное значение y_i величины η помимо уравнения регрессии $\bar{y}(x_i)$ определяется неучтенными факторами e_i (рисунок 2.10), получим, что $y_i = \bar{y}(x_i) + e_i$. Воспользовавшись свойством дисперсии суммы линейно независимых величин $D[\eta] = D[\bar{y}(x)] + D[e]$, получим:

$$R_{\xi\eta}^2 = 1 - \frac{\sum_{i=1}^n e_i^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad (2.40)$$

где $D[e]$ – дисперсия отклонений e_i ;

e_i – отклонения наблюдаемых значений y_i величины η от значений, полученных по уравнению регрессии $\bar{y}(x_i)$.

В соответствии с (2.40) коэффициент детерминации характеризует долю факторов, влияющих на зависимую случайную величину η , неучтенных в выбранном уравнении регрессии, которые обуславливают отклонения e_i .

Равенство коэффициента детерминации $R_{\xi\eta}^2 = 0$ указывает на то, что полученное уравнение регрессии (модель зависимости) никак не объясняет действительную зависимость между случайными величинами (например, если неудачно предложен вид зависимости или если

между величинами зависимость отсутствует). Равенство коэффициента детерминации $R_{\xi\eta}^2 = 1$ указывает на то, что зависимость между величинами является функциональной и она полностью определяется полученным уравнением регрессии, т. е. выбранное уравнение регрессии полностью (однозначно) определяет зависимость между случайными величинами. На практике, если значение $R_{\xi\eta}^2 > 0,7$, то считают, что выбранное уравнение регрессии хорошо описывает зависимость, существующую между случайными величинами. Если же $R_{\xi\eta}^2 < 0,3$, то уравнение регрессии незначительно описывает зависимость между случайными величинами (если таковая существует).

Отметим, что коэффициент детерминации $R_{\xi\eta}^2$ не является мерой какой-либо зависимости априорно (что справедливо для коэффициента корреляции $r_{\xi\eta}$, который всегда характеризует степень линейной зависимости). Он только показывает, насколько удачно выбранное уравнение регрессии описывает действительную зависимость между величинами ξ и η . Например, при проверке линейной регрессии между ξ и η по результатам испытаний, представленным на рисунке 2.11, $R_{\text{лин}}^2$ определяет, насколько удачно зависимость между величинами описывается прямой линией. В данном случае характеристики коэффициентов детерминации и корреляции будут совпадать, а их значения будут определяться равенством $R_{\text{лин}}^2 = (\hat{r}_{\xi\eta})^2$. В случае, если зависимость между величинами (рисунок 2.11) предполагается параболической, то коэффициент детерминации $R_{\text{пар}}^2$ будет характеризовать качество описания зависимости между ξ и η уравнением параболической регрессии. Очевидно, что $R_{\text{пар}}^2 > R_{\text{лин}}^2$, поскольку уравнение параболы описывает связь между ξ и η (рисунок 2.12).

Значимость коэффициентов корреляции и детерминации. Оценки $r_{\xi\eta}$ и $R_{\xi\eta}^2$ являются случайными величинами, т. к. для разных выборок одной и той же пары величины ξ и η оценки эти могут принимать различные значения. Для малых выборок эти различия будут особенно существенными при значениях этих коэффициентов, близких к нулю, они могут приводить к большим относительным погрешностям оценок и, следовательно, к неверным статистическим выводам. Поэтому в ответственных исследованиях приходится проверять значимость отличия этих коэффициентов от нуля. Проверка значимости позволяет сделать вывод о существенности описания значимости уравнением регрессии, либо о том, что данное уравнение практически

никак не определяет существующую зависимость между ξ и η , а нулевое значение $\hat{r}_{\xi\eta}^2$ и $\hat{R}_{\xi\eta}^2$ обусловлено лишь случайностью выборок.

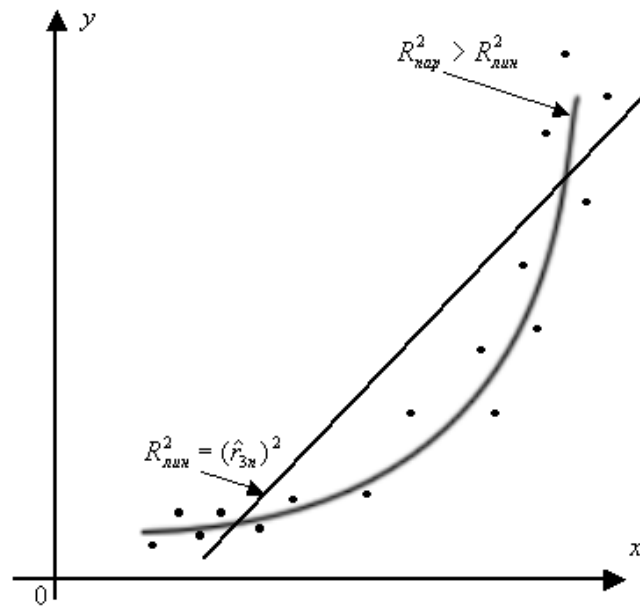


Рисунок 2.11 – Отклонения значений зависимой величины η от значений, предсказываемых уравнениями линейной и параболической регрессии

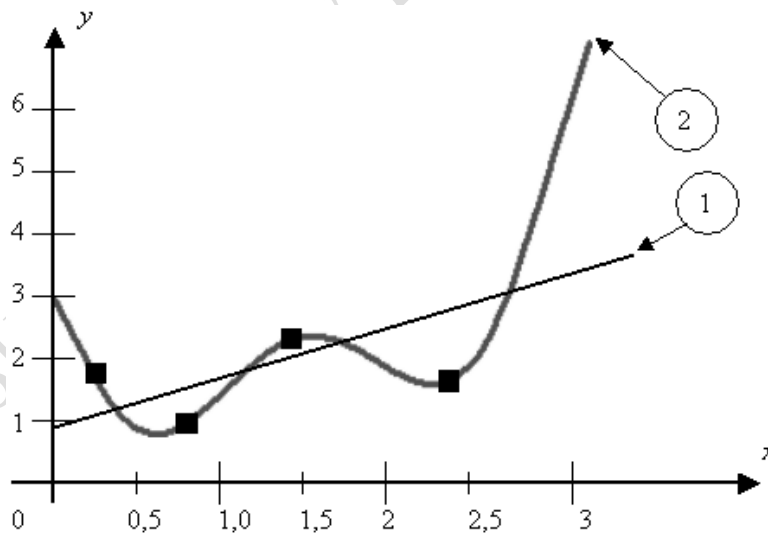


Рисунок 2.12 – Пример использования линейной и полиномиальной регрессионных моделей:

- 1) $y(x) = 0,901 + 0,636x$, ($R^2 = 0,73$);
- 2) $\bar{y}(x) = 2,78 - 4,76x - 4,13x^2 + 20,3x^3 - 17,94x^4 + 6,12x^5 - 0,723x^6$, ($R^2 = 1$)

Чтобы сделать статистический вывод о значимости оценки коэффициента корреляции (при проверке линейности регрессии), выдвигается

нулевая гипотеза об отсутствии линейной зависимости между исследуемыми величинами, т. е. $H_0: r = 0$. Если H_0 отклоняется, то считается, что уравнение регрессии η по ξ действительно имеет линейный вид, т. е. $H_a: r \neq 0$. Для проверки H_0 используется t -критерий значимости [58]:

$$\hat{t} = \hat{r} \sqrt{\frac{n-2}{1-\hat{r}^2}}, \quad (2.41)$$

который при условии справедливости H_0 имеет распределение Стьюдента с $v = n - 2$ степенями свободы. Вычисленное по (2.41) значение \hat{t} сравнивается с критическими значениями распределения Стьюдента $t_{\alpha/2, v}$ при $v = n - 2$ степенях свободы. Если $|\hat{t}| \leq t_{\alpha/2, v}$ для заданного уровня значимости α , то нулевая гипотеза о линейной значимости двух случайных величин ξ и η согласуется с результатами экспериментов. Если же $|\hat{t}| > t_{\alpha/2, v}$, то для данного уровня значимости α гипотеза $H_0: r = 0$ о линейной независимости двух случайных величин отклоняется, и с вероятностью ошибки α можно утверждать, что верна альтернативная гипотеза ($H_a: r \neq 0$), т. е. между ξ и η существует линейная зависимость.

Для проверки значимости оценки коэффициента детерминации выдвигается гипотеза о том, что предложенное уравнение регрессии никак не отражает реальную зависимость между случайными величинами ($H_0: R^2 = 0$). Альтернативная гипотеза заключается в том, что выбранное уравнение регрессии $\bar{y}(x)$ в достаточной степени объясняет действительную зависимость между случайными величинами ($H_a: R^2 > 0$). Для проверки значимости коэффициента детерминации используется F -статистика значимости:

$$F = R^2 \frac{n-m}{(m-1)(1-\hat{R}^2)}, \quad (2.42)$$

который (при справедливости H_0) имеет распределение Фишера с $v_1 = m - 1$ и $v_2 = n - m$ степенями свободы. Здесь m – число неизвестных параметров предполагаемого уравнения регрессии ($\beta_0, \beta_1, \dots, \beta_{m-1}$), определяемых с помощью МНК. Значение оценки F , вычисленной по формуле (2.42), сравнивается с критическим значением распределения

Фишера F_{α, v_1, v_2} . Если $\hat{F} \leq F_{\alpha, v_1, v_2}$, то $H_0: R^2 = 0$ согласуется с результатами испытаний. Соответственно нет оснований считать, что выбранное уравнение регрессии описывает зависимость между величинами ξ и η . Когда $\hat{F} > F_{\alpha, v_1, v_2}$, то $H_0: R^2 = 0$ отклоняется и принимается гипотеза $H_a: R^2 > 0$ и вероятностью ошибки α можно утверждать, что выбранное уравнение регрессии может использоваться в исследованиях.

Пошаговый регрессионный анализ. Говорят, что ξ и η коррелированы, если $r_{\xi\eta} \neq 0$. Когда при увеличении одной из величин существует тенденция роста другой величины, то ξ и η *коррелированы положительно*. В противном случае ($r_{\xi\eta} < 0$) величины ξ и η *коррелированы отрицательно*. Коррелированность указывает на наличие между ξ и η линейной зависимости, при $r_{\xi\eta} = 0$ говорят, что величины ξ и η не коррелированы. Но это не означает независимость этих величин. Главной целью регрессионного анализа является подбор наилучшего уравнения регрессии, минимизирующего сумму квадратов отклонений по методу наименьших квадратов (МНК) e_i наблюдаемых значений величины η от линии регрессии. Обычно исследователь желает либо усложнить вид уравнения регрессии, либо включить в модель регрессии как можно большее число факторов. Очевидно, что наилучшим образом эта зависимость описывается полиномами. Например, результаты n совместных измерений двух случайных величин могут быть интерполированы полиномами $(n - 1)$ степени вида:

$$\bar{y}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_{n-1} x^{n-1}. \quad (2.43)$$

В этом случае коэффициент детерминации, характеризующий эту регрессионную модель, будет равен 1. На рисунках 2.11 и 2.12 приведены примеры использования линейной и полиномиальной регрессионных моделей. При этом подобранные уравнения регрессии (см. линию 2) недостаточно точно «предсказывают» среднее значение «зависимой» величины η при известном значении «влияющей» величины $\xi = x$ по сравнению с линейной регрессией (см. линию 1 на рисунке 2.12). Таким образом, сложные регрессионные модели зачастую обладают меньшими предсказательными возможностями по сравнению с простыми моделями. Это обстоятельство объясняется еще и тем, что сложные модели характеризуются большими числом параметров $\beta_1, \beta_2, \dots, \beta_{n-1}$, выборочные оценки которых вносят большую неопределенность в модель регрессии и, таким образом, ухудшают ее

предсказательные возможности. Поэтому при подборе уравнения регрессии и сравнении нескольких альтернативных моделей регрессии целесообразно сравнивать по приведенным значениям коэффициентов детерминации:

$$R_n^2 = 1 - (1 - R^2) \frac{n-1}{n-m}, \quad (2.44)$$

где R^2 – коэффициент детерминации;

n – объем выборки;

m – количество параметров уравнения регрессии.

Таким образом, при подборе оптимального уравнения регрессии приходится искать компромисс между двумя противоречивыми тенденциями:

– с одной стороны, требуется чтобы подобранная модель наилучшим образом описывала реальную зависимость, учитывая множество и характер включающих факторов;

– с другой стороны, модель должна быть достаточно проста.

Поэтому эффективным способом оптимизации модели регрессии является *пошаговый регрессионный анализ*, который реализуется следующей итерационной процедурой усложнения (или упрощения) регрессионной модели.

1. Выявляются набор величин $\xi_1, \xi_2, \dots, \xi_n$ и характер зависимости точки η от каждой из влияющих переменных путем визуального анализа диаграмм рассеяния и физического смысла задачи. Характер зависимости определяется функциями $\varphi_j(\xi_1), \varphi_2(\xi_1), \dots, \varphi_1(\xi_1), \varphi_1(\xi_2), \dots, \varphi_k(\xi_n)$, например: $\beta_0; \beta_1 x_1; \beta_{2/x_1}; \beta_3 x_1^2, \beta_4 \ln(x_1); \beta_5 x_2, \beta_6 / x_2 \dots$

2. Методом наименьших квадратов определяются параметры β_1 этих частных уравнений регрессии и частные коэффициенты детерминации каждой из «влияющих» переменных x_i с «зависимой» величиной η , при неизменных значениях остальных «влияющих» переменной $x_i (j \neq i)$.

3. Из множества функций $\varphi_1(\xi_1), \varphi_2(\xi_1), \dots, \varphi_j(\xi_1), \varphi_1(\xi_2), \dots, \varphi_k(\xi_n)$ выбирается та функция, для которой коэффициент детерминации имеет наибольшее значение, и строится модель, включающая лишь указанную функцию $\varphi_j(\xi_1)$.

4. Вычисляются выборочные коэффициенты детерминации всех еще не включенных в модель функций $\varphi_d(\xi_t) (d = \overline{1, k}; t = \overline{1, n})$ с остатком e_i по формуле:

$$\bar{R}^2 = 1 - \frac{\sum_{i=1}^n \hat{e}_i^2}{\sum_{i=1}^n e_i^2}, \quad (2.45)$$

где $\varphi_d(\xi_t)$ – дополнительные функции в уравнении регрессии;

\tilde{e}_i – остатки уравнения регрессии после включения в него дополнительной функции;

e_i – остатки уравнения регрессии, полученные на предыдущем шаге.

5. Из дополнительных функций $\varphi_d(\xi_t)$ выбирается та, которой соответствует наибольший коэффициент детерминации \bar{R}^2 .

6. Проверяется значимость оценки коэффициента детерминации \bar{R}^2 . Если оценка коэффициента детерминации \bar{R}^2 оказывается статистически значимой, то строится новое уравнение регрессии, включающее дополнительную функцию $\varphi_d(\xi_t)$.

7. Проверяются гипотезы о возможности пренебречь зависимостью от каждой из ранее включенных в модель регрессии функций $\varphi_j(\xi_t)$.

8. Пункты 4–7 повторяются до тех пор, пока включение новых функций $\varphi_d(\xi_t)$ будет значимо улучшать модели регрессии, а исключение ранее включенных функций будет ухудшать модель регрессии *незначимо*. С целью предотвращения закливания описанной итерационной процедуры количество итераций ограничивается исследователем.

При исследовании зависимости между величинами возможны два подхода к проведению экспериментов. В первом случае исследователь сам задает (или варьирует) значения «независимых» (влияющих) величин, измеряя значения «зависимой» величины. Такой подход называется *активным экспериментом*. Во втором случае значения «независимых» величин не задаются, а только контролируются (замеряются) исследователем. Заменяемые величины также измеряются. Указанный подход называется *пассивным экспериментом*.

При анализе зависимости ξ и η на основе активных экспериментов коэффициенты корреляции (или детерминации) нельзя рассматривать как строгую меру взаимосвязи явлений, поскольку большую роль играет выбор самих значений x_i . В этом случае коэффициент корреляции (или детерминации) характеризует лишь меру близости эмпирических точек к линии регрессии. Поэтому при активном

эксперименте в качестве меры близости точек y_{ij} к условным выборочным средним \bar{y}_i используется *корреляционное отношение*:

$$\rho = \sum_{rj} (\bar{y}_i - \bar{y})^2 / \left(\sum_{i,j} (y_{ij} - \bar{y}) \right)^2, \quad (2.46)$$

где y_{ij} – значение величины η при i -м значении ξ ;

\bar{y}_i – условное среднее значение η при i -м значении ξ ;

\bar{y} – среднее значение η во всех экспериментах.

Нулевое значение ρ указывает на отсутствие какой-либо зависимости между изучаемыми величинами ξ и η . Для функционально зависимых $\rho = 1$.

Элементы дисперсионного анализа данных экспериментов. При прогнозировании случайных явлений исследователь стремится обычно учитывать большое количество влияющих факторов. Однако учесть все факторы зачастую невозможно. Поэтому приходится ограничиваться лишь факторами, которые значимо влияют на явление. Ответы на вопросы о значимости влияния на случайные явления тех или факторов дают методики *дисперсионного анализа*. Мы ограничимся однофакторным анализом. Предметом *однофакторного дисперсионного анализа* является проверка того, зависит ли математическое ожидание случайной величины от некоторого фактора A . *Многофакторный* дисперсионный анализ позволяет проверить, оказывают ли влияние на математическое ожидание случайной величины ξ определенные факторы (A, B, C, \dots, K). При однофакторном дисперсионном анализе пытаются разложить общую дисперсию случайной величины ξ $D[\xi]$ на два независимых слагаемых: факторную дисперсию ($D_{\text{фак}}$) и остаточную дисперсию ($D_{\text{ост}}$): $D[\xi] = D_{\text{фак}} + D_{\text{ост}}$. Чем сильнее влияет фактор A на значение ξ , тем больше $D_{\text{фак}}$ по отношению к $D_{\text{ост}}$. В результате сравнения факторной и остаточной дисперсий по критерию Фишера $F = D_{\text{фак}} / D_{\text{ост}}$ приходят к выводу о значимости влияния фактора A . С целью изучения влияния фактора A на случайную величину ξ измерения разбиваются в зависимости от вариации фактора A на k групп по n_i измерений в каждой группе. Результаты измерений обозначим x_{ij} , где i – номер фактора A ($i = \overline{1, k}$); j – номер результата измерения на данном уровне ($j = \overline{1, n}$).

На основе имеющейся статистической информации требуется проверить нулевую гипотезу $H_0 : M[\xi/A_1] = M[\xi/A_2] = \dots = M[\xi/A_k]$

о равенстве условных математических ожиданий величины ξ при различных значениях уровня фактора A против альтернативной гипотезы H_a о том, что не все условные математические ожидания совпадают (а, следовательно, их различия вызваны различием уровней фактора A). Проверка нулевой гипотезы H_0 основывается на вычислении критерия значимости:

$$F = \frac{\frac{1}{k-1} \sum_{i=1}^k ((x_i - \bar{x})^2 \cdot n_i)}{\frac{1}{n-k} \sum_{i=1}^k \sum_{j=1}^n (x_{ij} - x_i)^2}, \quad (2.47)$$

который при условии справедливости гипотезы H_0 имеет распределение Фишера с $\nu_1 = (k - 1)$ и $\nu_2 = (n - k)$ степенями свободы. Если значение критерия значимости F , вычисленное по выражению (2.47), ожидается больше критического F_{α, ν_1, ν_2} , определяемого квантилью распределения Фишера с ν_1 и ν_2 степенями свободы уровня α , то с вероятностью ошибки, меньшей α , можно утверждать, что гипотеза H_0 не верна (не все условные математические ожидания равны между собой). Следовательно, с вероятностью ошибки, меньшей α , можно утверждать, что значения исследуемого фактора A влияют на значения случайной величины ξ . Дисперсионный анализ основан на следующих допущениях:

- результаты измерений x_{ij} внутри i -й группы имеют нормальный закон распределения;
- внутригрупповые дисперсии результатов измерений x_{ij} при различных уровнях фактора A одинаковы, т. е. $D_1 = D_2 = \dots = D_k$.

Приведенные условия выполняются в случае, когда количество неучтенных факторов велико, а их совместное действие на величину ξ компенсируется и не зависит от значений уровней фактора A .

Пусть моделирование сложной системы проводилось при различных значениях фактора A (характеризующего внешнее воздействие, управление, или параметры системы). В единичном имитационном эксперименте x может зависеть от фактора A . Проведено $g \geq 2$ серий (групп) имитационных экспериментов (ИЭ): результатом i -й серии является случайная выборка $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ объема n_i ($i = \overline{1, g}$). Результат j -го ИЭ в i -й группе представляют в виде:

$$x_{ij} = \mu_i + \xi_{ij} \left(j = \overline{1, n_i}; i = \overline{1, g} \right), \quad (2.48)$$

где μ_i – неизвестное среднее значение наблюдений в i -й группе;
 ξ_{ij} – случайная Гауссовская ошибка наблюдения с распределением $N(0, \sigma^2)$, причем $\{\xi_{ij}\}$ независимы в совокупности. Модель данных (2.48) допускает эквивалентное представление $\left(n = \sum_{i=1}^g n_i \right)$:

$$x_{ij} = \mu + \alpha_i + \xi_{ij}, \quad (2.49)$$

где $\mu = \frac{1}{n} \sum_{i=1}^g n_i \mu_i = \sum_{i=1}^g (n_i / n) \mu_i$ – взвешенное среднее для всех g групп;

n_i/n – доля v -й группы в общей совокупности экспериментов
 $\alpha_j = \mu_j - \mu$ с независимым i -м эффектом взаимодействия.

Заметим, что по построению справедливо тождество:

$$\sum_{i=1}^g (n_i / n) \alpha_i = 0.$$

На практике обычно актуальны следующие две задачи статистического анализа результатов моделирования x_1, x_2, \dots, x_g :

– проверить гипотезу однородности $H_0: \mu_1 = \mu_2 = \dots = \mu_g$, имеющих дисперсии ($\alpha_1 = \alpha_2 \dots = \alpha_g = 0$), означающую отсутствие эффектов воздействия фактора A на x ;

– построить статистические оценки для $\mu_j \{ \alpha_i \}, \sigma^2$;

– построить доверительные интервалы для $\{ \mu_i \}$ и $\{ \Delta_{ik} = \mu_i - \mu_k \}$.

Эти задачи применительно к модели данных (2.46) относятся к проблематике однофакторного дисперсионного анализа, который мы рассмотрели выше и методика проведения которого реализуется следующей последовательностью расчетов:

1. Определяются основные статистики ИЭ:

$$\bar{x}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij} = \hat{\mu}_i \text{ – выборочное среднее } i\text{-й группы};$$

$$\bar{x} = \frac{1}{n} \sum_{i=1}^g \sum_{j=1}^{n_i} x_{ij} = \hat{\mu} \text{ – общее выборочное среднее};$$

$$\hat{\alpha}_i = x - \bar{x}_i \text{ – оценка } i\text{-го главного эффекта}; \quad (2.50)$$

$$S_{\text{выгр}}^2 = \frac{1}{(m - g)} \sum_{i=1}^g \sum_{j=1}^{n_i} (x_{ij} - \bar{x})^2 = \hat{\sigma}^2 \text{ – внутригрупповая дисперсия};$$

$$S_{\text{мгр}}^2 = \frac{1}{(g - 1)} \sum_{i=1}^g n_i (\bar{x}_i - \bar{x})^2 \text{ – межгрупповая дисперсия};$$

$$F = \frac{n-g}{g-1} \cdot \left(\frac{S_{\text{мгр}}^2}{S_{\text{выгр}}^2} \right) - F\text{-статистика.}$$

2. Для проверки гипотез H_0 , H_a с уровнем значимости α используется F критерий:

$$\text{Принимается гипотеза } \begin{cases} H_0, & \text{если } 1 - F(\alpha, v_1, v_2) > \alpha \\ H_a, & \text{в противном случае} \end{cases}, \quad (2.51)$$

где F_{α, v_1, v_2} – квантиль распределения Фишера.

Границы доверительного интервала для μ_i с доверительной вероятностью $(1 - \alpha)$ определяются выражением:

$$\bar{x}_i \pm t_{n-g}^{-1} (1 - \alpha/2) \cdot (S_{\text{выгр}}^2 / n_i)^{1/2}. \quad (2.52)$$

Границы доверительного интервала для различий выборок $\Delta_{ik} = \mu_i - \mu_k$ при сравнении двух средних μ_i и μ_k равны:

$$(\bar{x}_i - \bar{x}_k) \pm t_{n-g}^{-1} \left(1 - \frac{\alpha}{2} \right) \left((S_{\text{выгр}}^2) / (n_i^{-1} + n_k^{-1}) \right)^{1/2}; \quad (i \neq k = \overline{1, g}), \quad (2.53)$$

где $t(n - g)$ – квантиль распределения Стьюдента с $(n - g)$ степенями свободы для уровня значимости $\alpha/2$.

Более подробно с методикой многофакторного дисперсионного анализа можно познакомиться в работах [10; 57; 58].

3. МЕТОД СТАТИСТИЧЕСКИХ ИСПЫТАНИЙ КАК ОСНОВА МОДЕЛИРОВАНИЯ ВЕРОЯТНОСТНЫХ СИСТЕМ

3.1. Особенности использования вероятностных моделей сложных систем

Когда параметры элементов СС имеют вероятностную природу и когда многие ограничения формализации СС не выполняются, оказывается, что аналитические модели (АНМ) трудно применять для исследования динамики функционирования СС. В этих случаях результаты исследования на АНМ зачастую противоречат друг другу. Объясняется это особенностями СС, имеющими место именно из-за вероятностного характера поведения компонентов в реальной СС. Выделим три основные особенности моделирования вероятностных СС: во-первых, о поведении СС в лучшем случае известен только алгоритм, в котором разветвление расчетов зависит от жребия и результатов которых заранее непредсказуем; во-вторых, имеет место зависимость результата моделирования от исходных данных; в-третьих, необходимо накапливать статистику результатов моделирования и затем уже проводить статистический анализ накопленной статистики; в-четвертых, возникает необходимость в использовании механизма получения в модели значений этих случайных чисел. Необходимо помнить, что любое моделирование случайного процесса в СС позволяет дать только ответ на вопросы, что же произойдет при конкретных значениях этих случайных величин. И чтобы узнать больше, необходимы повторные эксперименты, в ходе которых накапливается статистика поведения элементов системы. Эти особенности моделирования вероятностных СС приводят к тому, что исследователю проще использовать универсальную процедуру метода статистических испытаний, которую мы назовем «процедурой Монте-Карло» и суть которой сводится к следующему.

Вместо того чтобы описывать СС, с помощью АНМ проводится N «розыгрышей» случайного явления в модели объекта имитации (заранее заданным числом N раз). При этом вектор параметров модели X не меняется (фиксирован) и на модели определяются значения компонентов вектора откликов Y . Так же, как в реальности, конкретная j -я реализация случайного процесса (из-за случайного характера алгоритма поведения реального явления) в модели всякий раз будет

складываться по-разному. Таким образом, проведя N экспериментов с моделью СС при одних и тех же значениях вектора параметров модели X из-за случайного характера алгоритма СС получают выборку значений откликов $\{Y_l\}$; $l = \overline{1, N}$. Усредняя выборку значений $\{Y_l\}$, находят вектор математических ожиданий значений компонент этого вектора $\bar{Y} = (\bar{Y}_{1x}, \bar{Y}_{2x}, \dots, \bar{Y}_l)$ и дисперсию S_y^2 компонент вектора отклика при фиксированных значениях вектора параметров $\{X\}$.

Число экспериментов N определяется из необходимой точности оценки Y . Задавшись точностью вычислений ϵ , по таблицам нормального распределения для доверительной вероятности $\beta = 1 - \alpha$ находят требуемое число экспериментов N , обеспечивающее оценку среднего \bar{Y} с вероятностью ошибки α . При оценивании среднего значения компонент вектора \bar{Y} наиболее часто встречаются следующие случаи.

а) Y_i имеют нормальное распределение, но объем выборки мал ($N \leq 30$). Тогда составляется t -статистика [2]

$$t = \sqrt{N-1}(\bar{y} - \mu) / S_y; S_y = \sqrt{S_y^2}, \quad (3.1)$$

где S_y^2 и μ – оценка дисперсии и истинное математическое ожидание компонент вектора откликов Y ;

\bar{Y} – среднее значение компонент вектора откликов.

Как известно, t -статистика имеет распределение Стьюдента [1]. Поэтому по соответствующим таблицам при $N - 1$ степенях свободы и заданном уровне значимости α оценивают доверительные интервалы для математических ожиданий вектора откликов:

$$\bar{Y} \pm d = \bar{Y} \pm t_{kp} S_y / \sqrt{N-1}, \quad (3.2)$$

где t_{kp} – значение t -статистики, вычисленное по функции распределения Стьюдента при заданном уровне значимости α числе степеней свободы $\nu = N - 1$, откуда требуемый объем выборки для оценки отклика с точностью d будет равен:

$$N \geq \left(\frac{t_{kp} S_y}{d} \right)^2 + 1; \quad (3.3)$$

б) Y_i имеют нормальное распределение; истинное математическое ожидание μ известно, объем выборки велик ($N \geq 30$). В этом случае используется двусторонняя нормальная статистика Z при заданном уровне значимости $\alpha/2$, из стандартного нормального распределения. Поэтому доверительный интервал можно определить в виде соотношения

$$\bar{Y} \pm d = \bar{Y} \pm Z_{\alpha/2} \frac{S_x}{\sqrt{N}}, \quad (3.4)$$

где $Z_{\alpha/2}$ – вычисляется из функции нормального распределения. Требуемый объем выборки определяется по формуле:

$$N \geq \left(\frac{S_y - Z_{\alpha/2}}{d} \right)^2, \quad (3.5)$$

где $Z_{\alpha/2}$ – квантиль стандартного нормального распределения при уровне доверия $\left(1 - \frac{\alpha}{2}\right)$;

d – допустимая величина ошибки для оценки средних значений откликов;

в) нормальность распределений компонентов вектора Y предположить нельзя, но известны истинные значения математического ожидания μ и дисперсии σ^2 при большом объеме выборки ($N \geq 30$). В этом случае лучше всего использовать неравенство Чебышева

$$P \left\{ \left| \bar{y} - \mu \right| \geq K \frac{\sigma}{N} \right\} \leq \frac{1}{K^2}. \quad (3.6)$$

Из этого неравенства доверительный интервал для оценки среднего значения можно записать в виде

$$\bar{Y} \pm d = \bar{Y} \pm \frac{K \sigma}{\sqrt{N}} = \bar{Y} \pm \frac{\sigma}{\sqrt{N} (1 - \alpha)}, \quad (3.7)$$

а требуемый объем выборки отклика

$$N \geq \frac{\sigma}{d^2(1-\alpha)}. \quad (3.8)$$

Обычно задаются $d \approx \sigma / 4$, $\alpha = 0,05$. Тогда $N \geq 320$.

При оценивании дисперсии σ^2 задача отыскания ее оценки S_Y^2 с достоверностью $1 - \alpha$ записывается в виде

$$P\{(1-d)\sigma^2 \leq S_Y^2 \leq (1+d)\sigma^2\} = 1 - \alpha \quad (3.9)$$

Эту задачу можно решать только при больших объемах выборки N . В этих случаях удобнее всего использовать χ^2 статистику:

$$\chi^2 = \frac{(N-1)S^2}{\sigma^2} \text{ с } (N-1) \text{ степенями свободы.} \quad (3.10)$$

Поскольку объем выборки N большой, то эту статистику можно аппроксимировать нормальным распределением. Поэтому формула для расчета требуемого объема выборки отклика имеет вид

$$N \geq 1 + 2 \left(\frac{Z_{\alpha/2}}{d} \right)^2, \quad (3.11)$$

где $Z_{\alpha/2}$ – квантиль стандартного нормального распределения при уровне значимости $\frac{\alpha}{2}$;

d – доверительный интервал оценки (чаще всего $d = \frac{\sigma_3}{4}$; σ_3^2 – эмпирическая оценка дисперсии отклика).

Основной операцией, из совокупности которых складывается процесс моделирования СС, согласно процедуре Монте-Карло является некая l -я реализация случайного процесса $F(X, \gamma)$. Она представляет собой как бы один экземпляр реализации алгоритмов развития СС в дереве вариантов его функционирования. Как правило, она складывается из последовательности, состоящей из двух типов составляющих. Первый тип составляющих представляет собой вычислительные процедуры F , а вторая составляющая означает розыгрыш значений γ_i с помощью специального разработанного алгоритма,

носящего название «бросание жребия». Каждый раз, когда в ход моделируемого явления вмешивается случайность, ее влияние в модели учитывается не расчетом, а «жребием». Далее расчетами F уже учитывается влияние исхода «жребия» на дальнейшее течение процесса. Учитываются при этом и условия розыгрыша следующего «жребия». Идею, положенную в основе процедуры Монте-Карло, рассмотрим на следующем примере. Пусть необходимо вычислить некоторую скалярную величину C , заданную некоторым математическим выражением, сложным для его аналитического исследования. На первом шаге определяют случайную величину ξ такую, что выполняются условия: $M[\xi] = C$; $D[\xi] \leq \infty$. На втором шаге моделируют N независимых реализаций этой случайной величины $(\xi_1, \xi_2, \dots, \xi_N)$. На третьем шаге по случайной выборке $\{\xi_l\}$ ($l = \overline{1, N}$) строят выбо-

рочную оценку величины $\left(C = \frac{1}{N} \sum_{l=1}^N \xi_l \right)$.

Классическим примером применения процедуры Монте-Карло является задача вычисления площади области сложной формы A . Процедура Монте-Карло при решении данной задачи реализуется следующим образом.

1. Область A погружается в область G с известной площадью $g = \text{mes}(G) < \infty$.

2. Формируется случайная величина Бернулли ξ , математическое ожидание которой равно C :

$$\xi = gF_A(\eta) = \begin{cases} g & \text{для } \eta \in A, \\ 0 & \text{для } \eta \notin A, \end{cases} \quad (3.12)$$

где $F_A(\eta)$ – индикаторная функция области A ;

η – случайный двумерный вектор равномерно распределенный в области G .

Обозначим $p = P(\eta \in A)$ – вероятность события $\{\eta \in A\}$. Тогда

$$\begin{aligned} M(\xi) &= g \cdot p + 0(1 - p) = gp = C; \\ D(\xi) &= g^2 p(1 - p) = C(g - C) < \infty. \end{aligned} \quad (3.13)$$

Моделируется N независимых реализаций $(\xi_1, \xi_2, \dots, \xi_N)$. Строится случайная выборка $\xi_1, \xi_2, \dots, \xi_N$ с помощью функции распределения (3.12) и вычисляется оценка величины C :

$$\hat{C} = \frac{1}{N} \sum_{l=1}^N \xi_l = g \cdot \frac{1}{n} \sum_{l=1}^N F_A(\eta_l) = g \cdot \frac{h'}{N}; \quad (3.14)$$

где h' – число реализаций из $\{\xi_l\}$, $l = \overline{1, N}$, попавших в область A ;

h'/N – относительная частота попаданий внутрь области, т. е. эффективная оценка вероятности p .

Для вычисления одной и той же величины C возможны разные варианты построения случайной величины ξ , удовлетворяющие условию $M(\xi) = C$; $D(\xi) < \infty$. При этом предпочтение отдается варианту, наиболее полно удовлетворяющему следующим условиям:

- дисперсия $D(\xi)$ минимальна на множестве $\{\xi\}$;
- моделирование ξ осуществляется одним из известных способов реализации жребия первого типа.

Рассмотрим еще один пример моделирования вероятностного технологического процесса производства (ВТПП), блок-схема реализации которого приведена на рисунке 3.1. ВТПП представляет собой совместное выполнение множества алгоритмов $\{АЛ_j\}$, $j = \overline{1, 11}$. Каждый из этих алгоритмов реализуется в течение постоянного времени τ_j . При выполнении алгоритмов в трех местах ВТПП имеет место вероятностный выбор выполнения последующих алгоритмов технологического процесса. На рисунке 3.1 выбор выполнения $АЛ_j$ по одному из направлений осуществляется вероятностным образом и на рисунке 3.1 представлен операторами проверки PR_k ($k = \overline{1, 3}$). Сумма вероятностей в каждой из этих операторов равна 1. Отметим, что основой функции алгоритма $АЛ_{11}$ является синхронизация выполнения ветвей ВТПП. Допустим, имел место выбор в этих операторах проверки направления выполнения, помеченный «1». Это приведет к тому, что алгоритмы $АЛ_4$, $АЛ_5$ и $АЛ_8$ не будут выполняться. Кроме того, время выполнения ВТПП равно:

$$T_{a1} = \max [(\tau_1 + \tau_7); (\tau_2 + \tau_4 + \tau_{11}); (\tau_3 + \tau_9 + \tau_{11}); (\tau_3 + \tau_6 + \tau_{10} + \tau_{11})]. \quad (3.15)$$

Если же выбор направления выполнения ВТПП совпал с пометкой «0» у операторов PR_k , то имеет место ситуация, когда алгоритмы $АЛ_6$, $АЛ_7$, $АЛ_9$ не будут выполняться. Время выполнения ВТПП при этом будет иметь другое значение:

$$T_{A2} = \max [(\tau_1 + \tau_4 + \tau_8 + \tau_{11}); (\tau_2 + \tau_8 + \tau_{11}); (\tau_3 + \tau_8 + \tau_{11}); (\tau_3 + \tau_6 + \tau_{10} + \tau_{11})]. \quad (3.16)$$

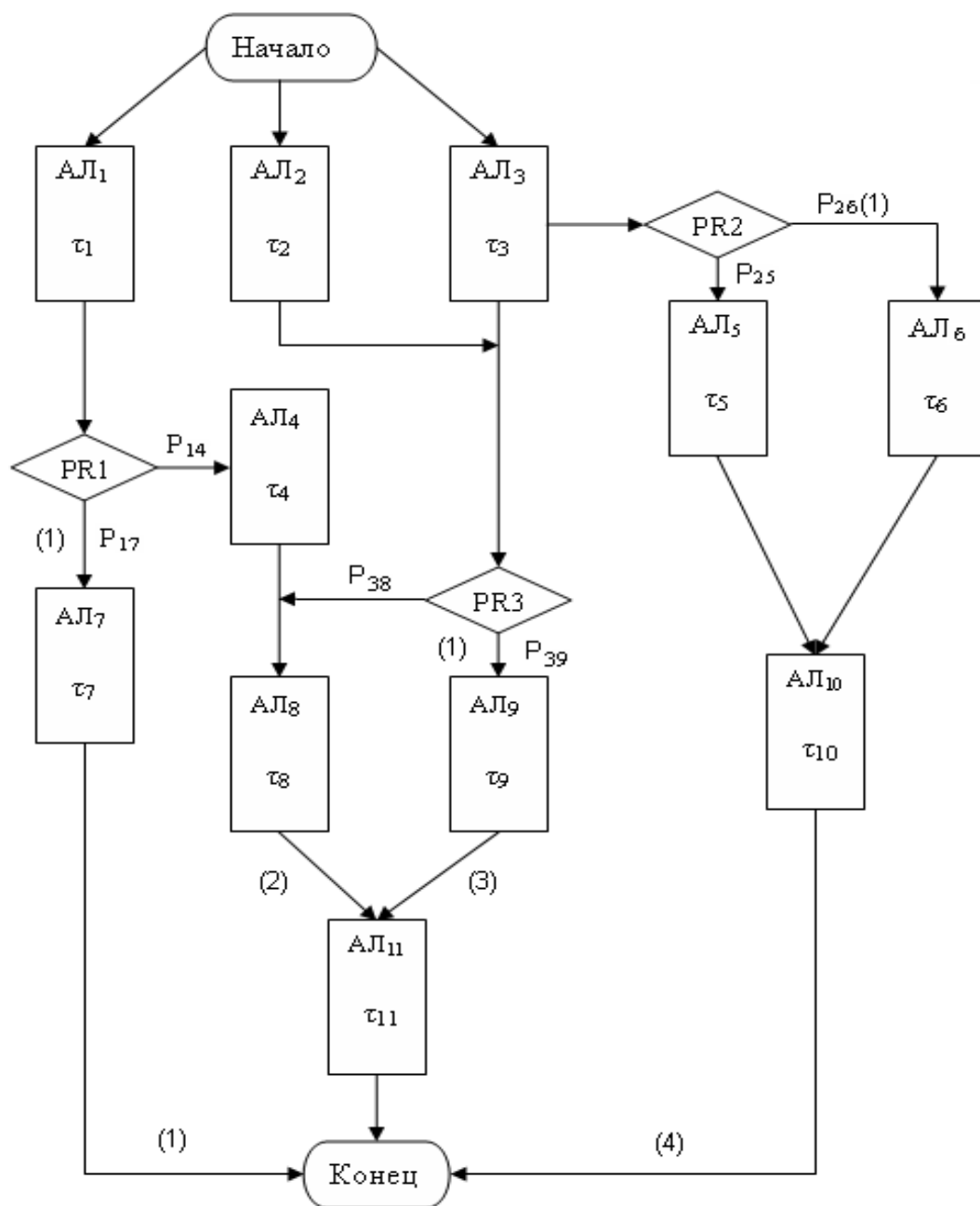


Рисунок 3.1 – Блок-схема выполнения ВТПП

Таким образом, в реальном ВТПП могут быть не только рассмотренные случаи выполнения множества алгоритмов, но и различные времена выполнения ВТПП. В общем случае, если не делать никаких предположений, то результаты выполнения множества алгоритмов и время их реализации будут вероятностными. Задача эта усложняется, если считать, что времена выполнения алгоритмов τ_j не являются детерминированными величинами. Например, если τ_j определяются по функциям распределения $F_j(\tau)$, то всякий раз могут быть различными. Этот пример показывает, насколько возрастает сложность решения

вероятностных задач и появляется неопределенность результата моделирования самого ВТПП и времени его выполнения. Очевидно, что эту задачу нужно решить с помощью процедуры Монте-Карло.

Для этого построим имитационную модель ВТПП. В подмоделях алгоритмов ИМ $АЛ_j$ будем осуществлять в l -й реализации модели розыгрыш по жребию 3-го типа значений τ_{jl} по соответствующим функциям распределения.

$$\tau_{jl} = F_j(\tau). \quad (3.17)$$

Сами операции проверки направления реализации ветвей ВТПП в операторах PR_k реализуются путем розыгрыша по жребию 2-го типа. При этом фиксируется статистика выполнения не только каждого алгоритма $АЛ_j$, но и всего ВТПП. В итоге каждой l -й реализации в ИМ фиксируются множество статистик выполнения $АЛ_j$ (τ_{jl}); $j = \overline{1,11}$ и общее время выполнения ВТПП (T_{al}). После N реализаций ИМ ВТПП получаем выборки значений времени выполнения алгоритмов $\{\tau_{jl}\}$; $j = \overline{1,11}$, $l = \overline{1, N}$ и общего времени выполнения ВТПП $\{T_{al}\}$; $l = \overline{1, N}$.

Далее согласно процедуре Монте-Карло по этим выборкам определяются оценки математического ожидания и дисперсии времен выполнения алгоритмов и всего времени реализации алгоритма:

$$\bar{\tau}_j; S^2 \tau_j; j = \overline{1,11}; \bar{T}_a; S_{T_a}^2. \quad (3.18)$$

Зная оценки математического ожидания откликов модели и их выборочные дисперсии, по известным формулам вычисляется ошибка расчета откликов имитационной модели ВТПП.

Приведенных примеров достаточно, для того чтобы установить известный факт, что вероятностные СС в общем случае легче решить с помощью процедуры Монте-Карло. При этом процедура Монте-Карло лежит в основе моделирования всех вероятностных ситуаций, и требуется в ИМ розыгрыш жребиев различного типа для моделирования в модели тех вероятностных ситуаций, которые имели место в реальной СС.

3.2. Виды и алгоритмы реализации единичных жребиев на ЭВМ

Условимся называть «единичным жребием» любой опыт со случайным исходом, в котором разыгрывается:

1) произошло или не произошло событие A , имеющее вероятность появления P ;

2) какое из возможных событий A_1, A_2, \dots, A_m произошло, если заданы вероятности их появления P_1, P_2, \dots, P_m ;

3) какое значение x приняла случайная величина X ;

4) какую совокупность значений (x_1, x_2, \dots, x_k) приняла система случайных чисел (X_1, X_2, \dots, X_k) .

Существует стандартный механизм, с помощью которого можно осуществить любой из перечисленных видов жребия. Он состоит в формировании случайного числа ξ , значения которого равновероятны на интервале $[0, 1]$ и последующем формульном преобразовании этого случайного числа ξ . Рассмотрим алгоритмы моделирования всех четырех видов жребия с помощью генератора случайных чисел ξ , равномерно распределенных на интервале $[0, 1]$.

Жребий 1⁰. Моделирование события A . Моделируется очередное значение случайного числа ξ путем обращения к генератору случайных чисел (ГСЧ). Если $p > \xi$, то $A :=$ «истина» (событие произошло), иначе $A :=$ «ложь» (событие не произошло).

Жребий 2⁰. При заданном векторе вероятностей появления (P_1, P_2, \dots, P_m) моделирование одного из возможных событий (A_1, A_2, \dots, A_m) . Делят интервал $[0, 1]$ на m участков, длины которых соответствуют вероятностям появления событий $P_i, i = \overline{1, m}$. Разыгрывается очередное значение величины ξ путем обращения к ГСЧ. Если ξ накрывает i -й участок, то считают, что событие $A_i :=$ «истина», а остальные $A_j :=$ «ложь», $i \neq j, j = \overline{1, m-1}$. Считается также, что все события A_i независимы.

Жребий 3⁰. Какое значение x приняла случайная величина X ?

Возможны два случая: X – дискретная и X – непрерывная величина. Если X – дискретная величина и известно распределение вероятностей в виде следующей последовательности пар значения и вероятности его появления:

$$\{(x_1, P_1), (x_2, P_2), \dots, (x_k, P_k)\},$$

где $\sum_{i=1}^k P_i = 1$, то формируют эмпирическую функцию дискретного

распределения $\left\{ (x_1, P_1), (x_2, P_1 + P_2), \dots, \left(x_k, \sum_{i=1}^k P_i \right) \right\}$. Разыгрывается

очередное значение ξ и проверяется выполнение неравенства $\sum_{i=1}^l P_i < \xi \leq \sum_{i=1}^{l+1} P_i$. Если это неравенство выполняется для l -го дискретного интервала, то считают, что $x := x_l$.

Если же X – непрерывная величина и известна функция плотности вероятностей ее распределения $f(x)$, формируют функцию распределения вероятностей появления величин, стремящуюся к 1 $\left(F(x) = \int_{-\infty}^{+\infty} f(x) dx \right)$. Разыгрывается очередное значение величины ξ путем обращения к ГСЧ. Далее вычисляется обратная функция Ψ от $F(x)$ при заданном значении $x' = \xi$; тогда $x := \Psi(\xi)$ и будет искомым значением случайной величины X . Предполагается при этом, что кривая $F(x)$ аналитически задана. Поэтому используется процедура получения обратной функции Ψ в точке $x' = \xi$. Если же $F(x)$ и $\Psi(x)$ трудно представимы аналитически, то их заменяют (аппроксимируют) линейно-кусочной (полигоном частот) или ступенчатой линией (гистограммой). А затем уже поступают аналогично предыдущему случаю для нахождения значений обратной функции в точке $x' = \xi$.

Жребий 4⁰. Какую совокупность значений (x_1, x_2, \dots, x_k) приняла система случайных чисел (X_1, X_2, \dots, X_k) ? Возможны два случая: X_i независимы друг от друга; X_i представляют систему зависимых величин. В первом случае розыгрыш (x_1, x_2, \dots, x_k) сводится в k раз повторенному моделированию величин X по жребию 3^0 . Гораздо хуже обстоит дело, когда x_1, x_2, \dots, x_k зависимы. В этом случае, разыграв одну из переменных, например $x_1 = \tilde{x}_1$, далее согласно условному распределению двух величин и с учетом того, что $x_1 = \tilde{x}_1$, разыгрываются значения $x_2 = \tilde{x}_2$. Третью величину, например x_3 , разыгрывают по условным распределениям уже трех переменных, а именно: $x_3 = \tilde{x}_3$ при условии, что $x_1 = \tilde{x}_1, x_2 = \tilde{x}_2$ и т. д. Это существенно усложняет алгоритм получения случайных величин и увеличивает размеры памяти, требуемой для моделирования под функцией условных распределений. Эти обстоятельства затрудняют имитацию зависимых переменных. Исследователи считают их независимыми, оценивая при этом ошибку от такой замены.

3.3. Моделирование на ЭВМ псевдослучайных величин

В основе реализации четырех видов единичных жребиев лежит последовательное генерирование равномерно распределенных величин ξ_i , $i = 0, 1, 2, \dots$. Во всех ГСЧ используется процедура формирования равномерно распределенных величин ξ_i на интервале $[0, 1]$. Затем уже производится функциональное преобразование ξ_i в x с заданным законом распределения $F(x)$.

Исторически возникло два подхода к моделированию величин ξ : с помощью специальных физических генераторов-приставок к ЭВМ и с помощью подобранного алгоритма, реализованного на ЭВМ в виде специальной процедуры, реализуемой схемами приставок к ЭВМ. Первый подход позволяет получить действительно случайные величины ξ на любом интервале их значений. Никогда предыдущее значение ξ не повторится вновь. Моделирование ξ основано на известных физических явлениях (гамма-излучение, косметические лучи, белый шум в электрических цепях и т. д.) с заданными законами распределений. Основными достоинствами таких генераторов являются: действительно случайный характер их появления, неповторимость и возможность получать одно число за одну специальную команду обращения к генератору-приставке к ЭВМ. При этом они обладают и существенными недостатками: нестабильностью работы, зависимостью от температуры, влажности, погоды, давления. Кроме того, неповторимость ξ затрудняет организацию многовариантных исследований процессов, требуя для этого специальных запоминаний последовательностей $\{\xi_j\}$. Эти недостатки и определили появление ГСЧ второго типа – программные ГСЧ. Это специальные процедуры, реализованные на ЭВМ, позволяющие моделировать последовательность $\{\xi_j\}$. При этом ξ_j случайны на некотором интервале значений j , называемом периодом аperiodичности ($N_{АП}$). Далее, с некоторого $N_{АП}$, все ξ_j начинают повторяться. Поэтому числа ξ_j и назвали псевдослучайными. Достоинства ГСЧ второго типа состоят в том, что они стабильно работают и не боятся перепадов температур, влажности и давления; достигается полная воспроизводимость всей последовательности чисел, начиная с некоторого начального значения ξ_0 . Главным недостатком является их большая ресурсоемкость. На получение одного

числа ξ_j ; необходимо выполнение целой программы (процедуры) генератора. Однако, несмотря на указанные недостатки, эти процедуры практически вытеснили из употребления физические датчики случайных чисел.

Базовый программный ГСЧ формирует последовательность псевдослучайных величин, равномерно распределенных на интервале $[0, 1]$: $\xi_1, \xi_2, \dots, \xi_n$. Практически в силу дискретности представления чисел в ЭВМ можно генерировать $M = 2^k$ различных чисел (здесь k – разрядность кода ЭВМ, а M – максимальное число, представленное в ЭВМ). Вероятность появления очередного соседнего числа составляет $P(\xi_i = x_i) = \frac{1}{M}$. Обычно моделирование в ГСЧ второго типа ведется по итеративной формуле, т. е. очередное значение x_i^* является функцией всех предыдущих значений:

$$x_i^* = \varphi(x_{i-1}^*, \dots, x_0^*) \bmod M. \quad (3.19)$$

Нормировка по модулю M необходима для перевода значений x_i^* в интервал $[0, 1]$. Реализация x_i^* получается внутри алгоритма путем подобранного преобразования φ из ранее имитированных чисел $x_0^*, x_1^*, \dots, x_{i-1}^*$. Существует множество процедур, реализующих в ГСЧ различные алгоритмы преобразований φ . Наиболее распространен конгруэнтный метод генерации псевдослучайных чисел, сводящийся к формальному преобразованию:

$$x_i^* = (Ax_{i-1}^* + C) \bmod M, \quad x_i = x_i^*/M \in [0, 1], \quad (3.20)$$

где A и C специально подбираются для конкретного генератора псевдослучайных чисел. В результате генератор формирует новое значение x_i , но при этом требуется задавать предыдущее значение x_{i-1}^* из последовательности псевдослучайных чисел:

Затем нормировкой максимальной величиной $\mu = 2^k$ (k – разрядность ЭВМ) получают:

Работу этого алгоритма представим для случая $A = 2$ и $C = 0$, т. е. $x_i^* = 2x_{i-1}^* \bmod M$. Графически моделирование по этому алгоритму представлено на рисунках 3.2 и 3.3. В первом случае начальное значение x_0 подобрано очень неудачно, что приводит к очень короткому

периоду аperiodичности ГСЧ ($N_{АП} = 2$). Этот ГСЧ позволил получить всего 2 различных значения, и далее началось повторение этой пары чисел. Здесь диагональ играет роль подстановки предыдущего значения x в качестве x_{i-1} для расчета x_i . Во втором случае x_0 подобрано гораздо лучше, что позволило существенно увеличить период аperiodичности. И, как видно из рисунка 3.3, все шесть x_i действительно отличаются друг от друга и распределены относительно равномерно по всему интервалу значений (от 0 до 1). Именно переходной режим в ГСЧ и позволяет получать псевдослучайные последовательности чисел ξ_i .

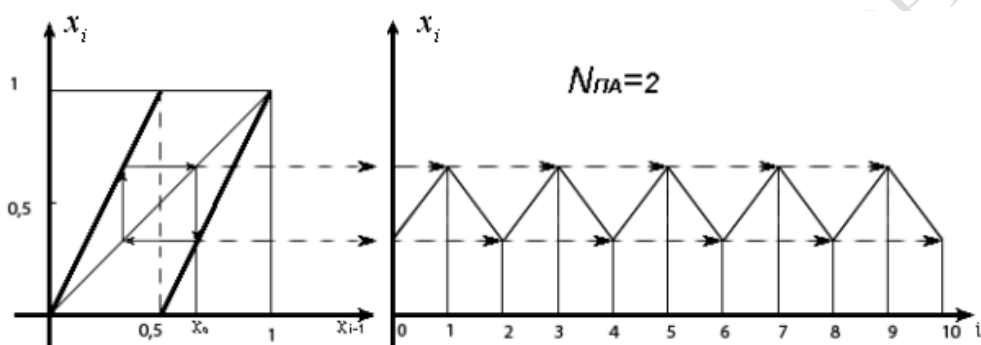


Рисунок 3.2 – Диаграмма моделирования псевдослучайных последовательностей по конгруэнтному методу, когда $x_i^* = 2x_{i-1}^* \bmod M$ и $N_{АП} = 2$

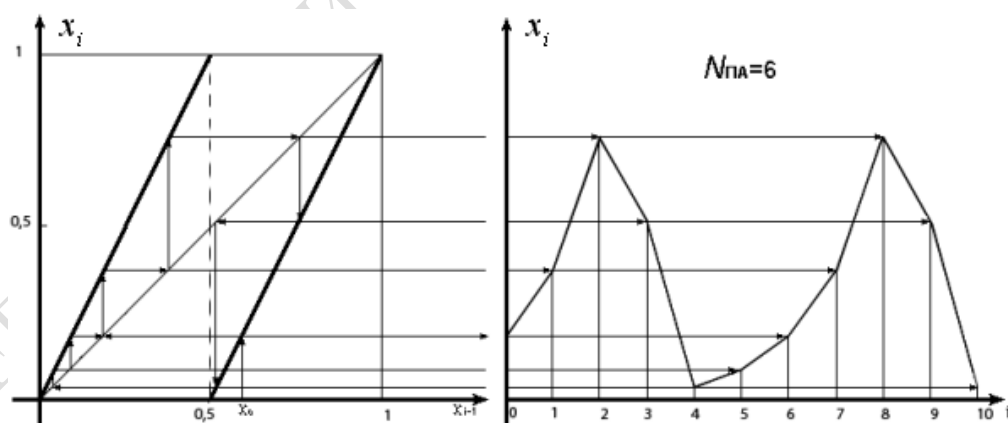


Рисунок 3.3 – Диаграмма моделирования псевдослучайных последовательностей по конгруэнтному методу, когда $x_i^* = 2x_{i-1}^* \bmod M$ и $N_{АП} = 6$

Кроме рассмотренного, существует еще один способ, называемый «способом серединных квадратов». Суть этого способа заключается

в следующем. Берут произвольное m -разрядное число x_0 , возводят его в квадрат, и результат уже имеет $2m$ разрядов. Из него выбирают средние m цифр, получая таким способом новое число x_1 . Далее процедура повторяется, в качестве начального значения используют ранее полученное x_1 . Его возводят в квадрат. Из него выбирают средние m цифр и получают новое число x_2 .

На рисунке 3.2 приведена диаграмма моделирования псевдослучайных последовательностей по конгруэнтному методу, когда $x_i^* = 2x_{i-1}^* (C = 0)$, начальное значение x_0 . На рисунке 3.3 приведена диаграмма моделирования псевдослучайных последовательностей по конгруэнтному методу ($x_i^* = 2x_{i-1}^*$), с другим начальным значением x_0^* .

Третий способ представляет собой совмещение двух конгруэнтных способов. Имеет место двойное обращение к ГСЧ, работающему по конгруэнтному способу, а затем происходит формульное преобразование:

$$x_{i+1}^* = (x_i - x_{i-1}) \bmod M \in \{1, \dots, M - 1\} \quad (3.21)$$

Эта простейшая процедура существенно увеличивает период периодичности ГСЧ. Наиболее эффективна процедура возмущения основного алгоритма ГСЧ. Суть этой процедуры состоит в следующем. Одновременно используются две процедуры ГСЧ. Основная процедура моделирует последовательность случайных чисел по алгоритму:

$$x_{i1}^* = \varphi_1(x_{i1-1}) \bmod M. \quad (3.22)$$

Для этой процедуры через каждые k чисел меняется начальное значение x_0 , моделируемое по алгоритму второй процедуры:

$$x_{i2}^* = \varphi_2(x_{i2-1}) \bmod M. \quad (3.23)$$

Комбинацией обращений к двум процедурам ГСЧ удастся увеличить период аperiodичности на несколько порядков.

Более подробно с алгоритмами формирования базовых генераторов псевдослучайных величин $\{\xi\}$, равномерно распределенных на интервале $[0, 1]$, можно познакомиться в работах [37; 57].

3.4. Примеры формирования и использования псевдослучайных величин

Моделирование псевдослучайных величин, имеющих распределение вероятностей, отличающееся от равномерного, осуществляется путем функционального преобразования величин ξ . Рассмотрим наиболее употребительные алгоритмы функционального преобразования ξ к искомой случайной функции.

Нормальное распределение можно моделировать множеством способов. Рассмотрим два алгоритма. **Первый алгоритм** моделирования нормально распределенной величины η с параметрами $M = 0$ и $\sigma^2 = 1$ основан на центральной предельной теореме теории вероятностей. Согласно этой теореме при сложении достаточно большого числа независимых случайных величин $\{\xi_i\}$, сравнимых между собой по своим дисперсиям, получаем новую случайную величину η , распределенную по нормированному нормальному закону. Чем больше складывается значений ξ_i , тем ближе распределение $\sum_i \xi_i$ оказывается

к нормальному. Поскольку ξ_i имеют одну и ту же дисперсию и считаются независимыми друг от друга, то для практики достаточно от 6 до 12 сложений ξ_i с последующим вычислением по формуле:

$$\eta = \sqrt{\frac{12}{n}} \left[\sum_{i=1}^n \xi_i - \frac{n}{2} \right], \quad (3.24)$$

при $n = 12$ формула (3.24) упрощается:

$$\eta = \sum_{i=1}^{12} \xi_i - 6,$$

где n – число обращений к ГСЧ.

Второй алгоритм является более экономичным, поскольку позволяет по двум обращениям к базовому ГСЧ (ξ_1 и ξ_2) получить пару псевдослучайных чисел (η_1 и η_2), распределенных по нормальному закону. Расчеты ведутся по формулам, предложенным Г. А. Михайловым [14].

$$\eta_1 = \sqrt{-2 \ln(\xi_1)} \cdot \cos(2\pi \cdot \xi_2); \quad \eta_2 = \sqrt{-2 \ln(\xi_1)} \cdot \sin(2\pi \xi_2). \quad (3.25)$$

Экспоненциальное распределение широко используется при моделировании сложных технических систем. Наиболее распространены следующие два способа имитации псевдослучайных величин, которые имеют одну и ту же функцию плотности распределения $f(x) = \lambda e^{-\lambda x}$, где λ – параметр экспоненциального распределения.

Первый способ основан на методе обратной функции

$$F(x) = 1 - e^{-\lambda x}. \quad (3.26)$$

Обратной к ней будет функция $e^{-\lambda x}$. Тогда экспоненциально распределенная величина с параметром λ формируется путем следующего функционального преобразования величины ξ :

$$\varepsilon = -\frac{1}{\lambda} \ln(1 - \xi). \quad (3.27)$$

Второй способ основан на методе композиции. Опытным путем находят величины τ и θ такими, чтобы характеристики величины ε были как можно ближе к экспоненте. С помощью генератора равномерно распределенных величин моделируется θ штук значений $(\xi_1, \xi_2, \dots, \xi_\theta)$. Затем осуществляется пересчет по формуле

$$\varepsilon = \tau - \max\{\xi_1, \xi_2, \dots, \xi_\theta\}. \quad (3.28)$$

Для практики достаточно задавать $\tau = 4$ и $\theta = 4$, что позволяет получать экспоненциально распределенную величину ε с параметром $\lambda = 1$.

Дискретные распределения также моделируются из базовых значений ξ_1 . При схеме независимых испытаний можно моделировать псевдослучайные числа, распределенные по **биномиальному закону**. В этих случаях используются n -разрядные коды чисел, полученных на ЭВМ. При каждом i -м обращении к генератору случайных чисел подсчитывается число единиц в коде псевдослучайного числа ξ_i , равное Y_i . Оказалось, что величина Y_i имеет биномиальное распределение ($Y_i = C_n^k$, k – разрядность кода ЭВМ).

Если необходимо моделировать дискретные случайные величины, имеющие **геометрическое распределение вида:**

$$P(v = y) = P(1 - P)^{y-1}, \quad \text{где } y = 1, 2, \dots, \quad (3.29)$$

то находят номер испытания (y), которое принесет первый успех для появления события A с вероятностью P . Поэтому последовательно обращаются к ГСЧ и моделируют очередное ξ_i . Проверяют выполнимость жребия первого типа. Номер обращения (y) к ГСЧ, обеспечивающего $A_i :=$ «истина», и дает очередное значение величины y , распределенной по геометрическому распределению с параметром P .

В технике широко распространено распределение **Пуассона**

$$P(k) = \frac{\lambda^k}{k!} e^{-\lambda}, \quad (3.30)$$

где λ – параметр распределения Пуассона для дискретной переменной k .

Для моделирования величин k последовательно обращаются к ГСИ, получают каждый раз очередное значение ξ_i и проверяют выполнимость неравенства

$$\prod_{i=1}^k \xi_i \geq e^{-\lambda} > \prod_{i=1}^{k+1} \xi_i. \quad (3.31)$$

Как только это неравенство (3.31) выполнится, то сумма ξ_i будет иметь распределение Пуассона с параметром λ . Для более полного ознакомления с алгоритмами моделирования псевдослучайных величин, имеющих различные стандартные распределения, можно рекомендовать книги [37; 57].

3.5. Методы исключения и суперпозиции при моделировании случайных величин

В тех случаях, когда плотность распределения $f_0(x)$ моделируемой случайной величины x имеет сложный аналитический вид и метод обратной функции неприменим, используется метод исключения. Обозначим функцию распределения $F_0 = f(x, y); 0 \leq y \leq f_0(x), x \in R$ – область, ограниченная кривой $y = f_0(x)$ и осью абсцисс (рисунок 3.4, а). Определяется мажорирующая функция $y = g(x)$; где $g(x) \geq f_0(x) \geq 0$ и область $G = f(x, y); 0 \leq y \leq g(x); x \in R > F_0$ (рисунок 3.4, а). При этом мажорирующая функция $g(x)$ должна иметь более простой вид, чем $f_0(x)$.

Как видим, G охватывает (мажорирует) область F_0 и имеет более простой вид, который позволяет легко моделировать случайный

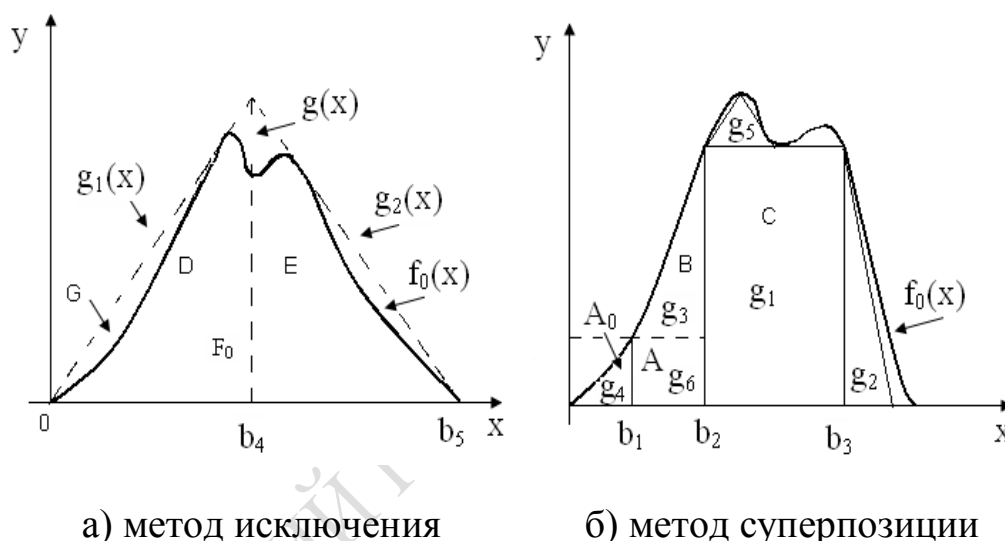
вектор (ξ', η') , равномерно распределенный в области G . Моделирующий алгоритм включает следующие этапы:

1) подбор мажорирующей функции $g(x)$;

2) моделирование случайного вектора (ξ', η') , равномерно распределенное в области G ;

3) если (x, y) – реализация вектора (ξ', η') и $y \leq f_0(x)$ (т. е. точка с координатами (x, y) попадает в область F_0 , то x принимается в качестве реализации случайной ξ . Если же $y > f_0(x)$, что означает то, что точка (x, y) лежит за пределами F_0 , то реализация (x, y) исключается и этап 2 повторяется снова.

Математическое обоснование метода можно найти в работе [57].



а) метод исключения

б) метод суперпозиции

Рисунок 3.4 – Вид функций плотности распределений, моделируемых методами исключения и суперпозиции

В тех случаях, когда ни метод обратной функции, ни метод исключения в виду сложности функции плотности $f_0(x)$ трудно осуществить, моделирование случайной величины x с плотностью распределения $f_0(x)$ можно свести к моделированию случайного вектора (ξ, η) , равномерно распределенного в области F_0 (образованной кривой $y = f_0(x)$ и осью абсцисс), поскольку компонента ξ этого вектора имеет плотность распределения $f_0(x)$. Из-за сложного вида $f_0(x)$ моделирование равномерного распределения в области F_0 трудно осуществимо. Поэтому согласно методу суперпозиции область F_0 разбивается на фиксированное число простых областей, моделирование равномерного распределения в которых осуществляется просто. При этом вероятности попадания в эти области равны их площадям.

Используя эти вероятности, вначале с помощью жребия 2-го типа, поочередно «разыгрывают» номер области, в которой осуществляется моделирование с помощью типового ГСЧ.

Метод суперпозиции основан на формуле полной вероятности для плотностей. Пусть ξ^x и v – случайные величины, заданные на одном и том же вероятностном пространстве: $F_v(z)$ – функция распределения; $P_{\xi^x/v}(x/z)$ – условная плотность распределения ξ^x при условии $v = z$.

Тогда безусловная плотность распределения ξ^x равна:

$$f_0(x) = \int_{-\infty}^{\infty} P_{\xi^x/v}(x/z) dF_v(z). \quad (3.32)$$

Если v – дискретная величина со множеством значений (C_1, \dots, C_N) и вероятностями $P(v = c_i) = P_i$ ($i = \overline{1, N}; N \leq \infty$); $P_{\xi^x/v}(x/c_i) = f_i(x)$, то функция безусловной плотности распределения принимает вид:

$$f_0(x) = \sum_{i=1}^N P_i \cdot f_i(x). \quad (3.33)$$

Моделирующий алгоритм включает следующие этапы:

- 1) определение вспомогательной случайной величины v , удовлетворяющей условиям (3.32) и (3.33);
- 2) моделирование v , и считают, что Z есть реализация v ;
- 3) моделирование реализации случайной величины с плотностью $f_i(x)$ (т. е. получение x – реализации ξ^x).

Графически эти операции можно проиллюстрировать примером на рисунке 3.4, б). Из рисунка 3.4, б) видно, что область $F_0 = \{(x, y)\}; 0 \leq y \leq f_0(x); b \leq x \leq c$ разбивается на 6 непересекающихся частей $\{g_i\}, i = \overline{1, 6}$ с площадями $\{P_i\}$. Основным принцип разбиения (3.33) заключается в том, что части g_i , имеющие наибольшую площадь и (наибольшую вероятность P_i), должны соответствовать просто и быстро моделируемым плотностям $f_i(x)$. На рисунке 3.4, б) число областей равно $n = 6$, причем плотности $f_i(x) i = \overline{1, 5}$. Эти распределения либо треугольного, либо ступенчатого вида. Они легко моделируются.

Остаточную плотность можно моделировать методом исключения:

$$f_0(x) = (f_{0/x}) - \sum_{i=1}^5 P_i \cdot f_i(x) / P_6; P_6 = \left(1 - \sum_{i=1}^5 P_i\right). \quad (3.34)$$

При моделировании случайных величин с функцией распределения сложной формы (как показано на рисунке 3.4, б) нужны алгоритмы формирования псевдослучайных величин, у которых функция плотности имеет вид прямоугольного треугольника (распределения g_1 и g_2), обычного треугольника (распределение g_5), прямоугольника (распределение g_1) и прямоугольной трапеции (распределение g_3). На рисунке 3.4, а) функция $g(x) = g_1(x) + g_2(x)$. Это означает, что нужно моделировать нормированное распределение (на интервале от 0 до 1), имеющее форму прямоугольника. А далее осуществлялось обычное преобразование от нормированной величины к распределению, заданному на интервале $[b_1, b_2]$ или на интервале $[b_2, b_3]$.

Если распределение имеет форму прямоугольного треугольника (рисунок 3.4, б)) на интервале $[0, b_1]$ и площадь которого равна $A_0 = P_1$, то используя конгруэнтный метод, расчетная формула будет иметь вид:

$$\xi_i = \frac{1}{2} A_0 \xi_{i-1}, \quad (3.35)$$

где $A_0 = 1/2 \cdot b_1$; вероятность попадания ξ в область $[0, b_1]$.

Когда распределение имеет форму прямоугольника, представленного на рисунке 3.4, б) на интервале $[b_2, b_3]$, и площадь которого равна $A_0 = P_1$, то согласно конгруэнтному способу расчетная формула имеет вид:

$$\xi_i = 2 \cdot C \cdot \xi_{i-1}. \quad (3.36)$$

Для случая, когда распределение имеет форму прямоугольной трапеции и площадь ее равна $(A + B)$, составные части которой представляют собой вероятность P_2 и P_3 попадания в интервал $[b_1, b_2]$, то согласно суперпозиции двух конгруэнтных методов расчетная формула примет вид:

$$\xi_i = (A + 2B) \xi_{i-1}. \quad (3.37)$$

Наконец, при моделировании треугольного распределения, имеющего площадь, равную $(D + E)$, где D и E – вероятности попадания ξ соответственно в области $[0, b_4]$ и $[b_4, b_5]$ (рисунок 3.4), расчетная формула примет вид:

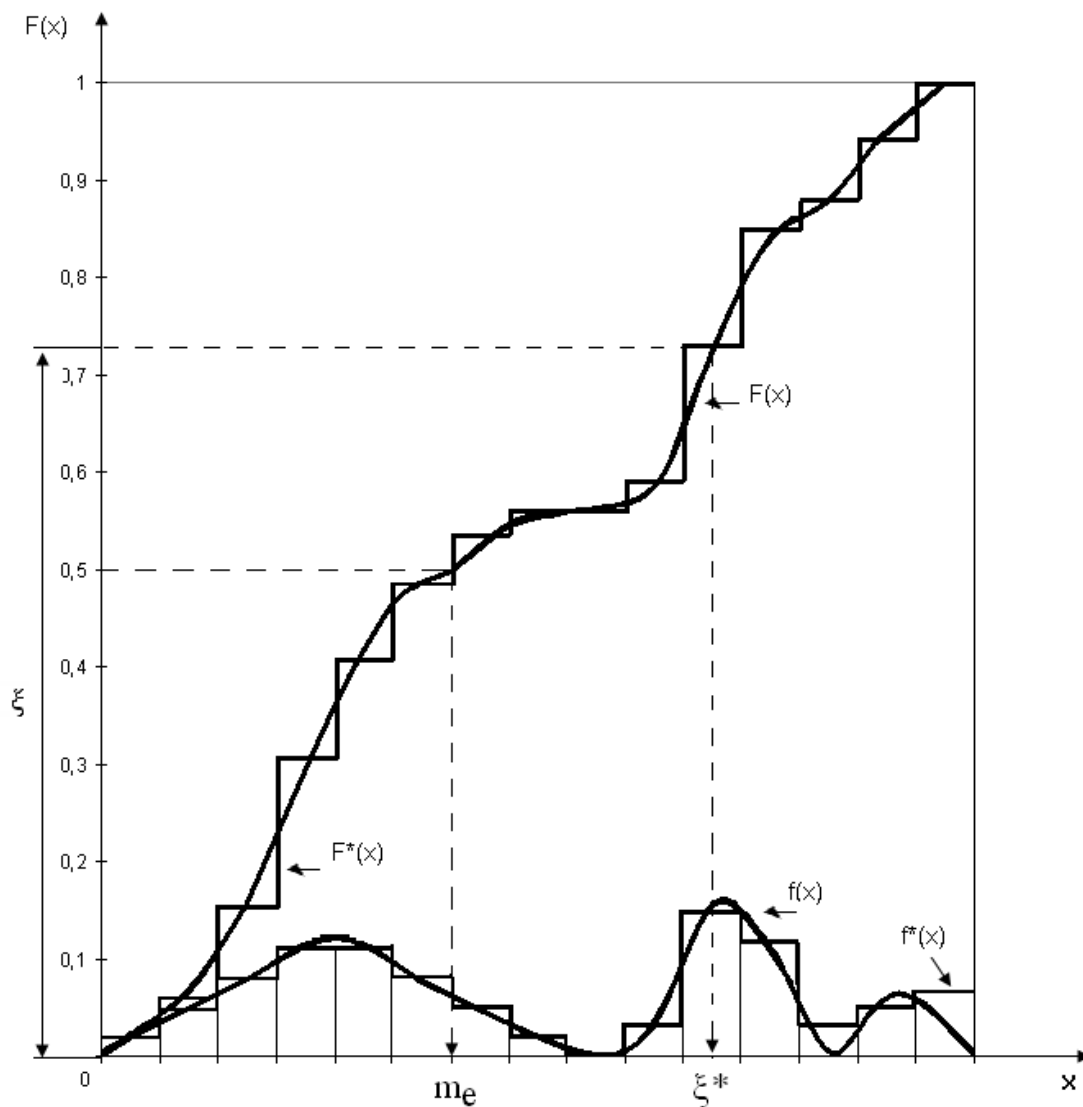
$$\xi_{i+1} = \frac{1}{2}D \cdot \xi_{i-1} + \frac{1}{2}E\xi_i. \quad (3.38)$$

Как видим, для формирования случайной величины, имеющей сложную форму функции плотности распределения, нужно несколько обращений к ГСЧ и следует осуществить формульное преобразование. Во всех случаях разыгрывается жребий вероятности выбора необходимой части сложной функции плотности распределения, а затем уже проводится формальное преобразование значений ξ_i , полученных от ГСЧ. В тех случаях, когда это необходимо, имеет место обычный переход от нормированной случайной величины ξ_i к необходимому значению случайной величины, исходя из границ области нахождения этой случайной величины.

3.6. Моделирование случайных величин, имеющих мультимодальную функцию плотности распределения

Все приведенные выше способы моделирования случайной величины x , имеющей функцию плотности, аналитически трудно представимую, являются весьма ресурсоемкими. В случае мультимодальной плотности распределения у случайной величины эти методы весьма громоздки, и поэтому на практике гораздо проще использовать следующую технологию имитации случайной величины, которая оказывается при многократном имитационном эксперименте менее ресурсоемкой, чем рассмотренные ранее способы ее имитации.

Для изложения технологии имитации рассмотрим пример, когда случайная величина имеет мультимодальную функцию плотности распределения, представленную на рисунке 3.5. В начале аппроксимируют функцию плотности $f(x)$ эмпирической ступенчатой функцией плотности $f^*(x)$. Затем формируют эмпирическую ступенчатую функцию плотности $F^*(x)$ по известному алгоритму накопления $F_k^*(x) = \sum_k f_k^*(x)$, значения которой стремятся к единице.



Рисунке 3.5 – Графическая интерпретация моделирования случайных величин мультимодального распределения $F(x)$

На рисунке 3.5 приведен также вид функции распределения $F(x)$. Используем рассмотренный ранее алгоритм формирования случайной величины по жребию третьего типа (случай с эмпирической функцией распределения). Допустим, в l -й реализации ГСЧ сформировал значение величины ξ (рисунок 3.5). Тогда искомое значение случайной величины x в l -й реализации будет равно ξ_l^* . Как видим из рисунка 3.5, значение ξ_l определяется по функции распределения $F^*(x)$ на каждом интервале группирования. Как только выполняется неравенство $F_k^*(x) < \xi_b \leq F_{k+1}^*(x)$, определяется искомая величина ξ_l^* , которая равна середине интервала $(k + 1)$ группирования x .

3.7. Пример статистического моделирования процесса обстрела ракетами площадной цели сложной формы

В качестве примера использования процедуры Монте-Карло и генераторов псевдослучайных чисел рассмотрим моделирование поражения ракетами площадной цели со сложной структурой. В распоряжении нападающей стороны имеется n ракет. Зона разрушения одной ракеты представляет собой круг радиуса r . В результате n выстрелов ракеты будет поражена какая-то часть (S_n) общей площади S_0 . Чтобы увеличить долю поражения цели $U = S_n/S_0$ и избежать ненужных перекрытий зон поражения, на площадной цели устанавливаются n точек прицеливания (O_1, O_2, \dots, O_n).

Известны характеристики баллистического рассеяния ракет (по осям координат соответственно σ_x^2 и σ_y^2). Предполагается, что систематические ошибки при попадании ракет на цель отсутствуют. Рассеяние ракет подчиняется двумерному нормальному закону распределения. Координаты точек попадания $(x_j, y_j), j = \overline{1, n}$ независимы друг от друга и от координат других точек попадания.

Требуется: при заданном расположении точек прицеливания O_1, O_2, \dots, O_n вычислить следующие характеристики обстрела: $m_u = M[U]$ среднюю долю пораженной площади цели; D_u – дисперсию доли поражения площадной цели; $P(U > \tilde{U})$ – вероятность того, что будет поражено не менее заданной доли площади \tilde{U} ; r_0 – математическое ожидание числа промахов ракет.

Если не делать никаких упрощений о форме цели и зонах поражения, то аналитически эту задачу решить трудно. Проще ее решить с помощью процедуры Монте-Карло. Каждая реализация представляет собой обстрел цели n ракетами, при котором все точки попадания (x_j, y_j) будут разыграны по жребию. Моделирование каждой l -й реализации будет состоять из единичных жребиев плюс расчет площади пораженной цели S_n и производных от нее характеристик поражения цели. На рисунке 3.6 представлен пример попадания ракет на площадную цель для l -й реализации по методу Монте-Карло.

Координаты попадания ракеты j распределены по двумерному нормальному закону с параметрами $(m_{xj}, m_{yj}, \sigma_x, \sigma_y)$, где m_{xj} и m_{yj} – координаты точек прицеливания $O_j, j = \overline{1, n}$. По условию независимости

друг от друга x_j и y_j полагаем, что коэффициент корреляции $Z_{xy} = 0$.

Тогда координаты точек попадания определяются по формулам:

$$x_j = \sigma_x \sqrt{2} \left(\sum_{k=1}^6 \xi_k - 3 \right) + m_{xj}; \quad (3.39)$$

$$y_j = \sigma_y \sqrt{2} \left(\sum_{k=7}^{12} \xi_k - 3 \right) + m_{yj}, \quad (3.40)$$

где ξ_k – 12 экземпляров случайных равномерно распределенных величин.

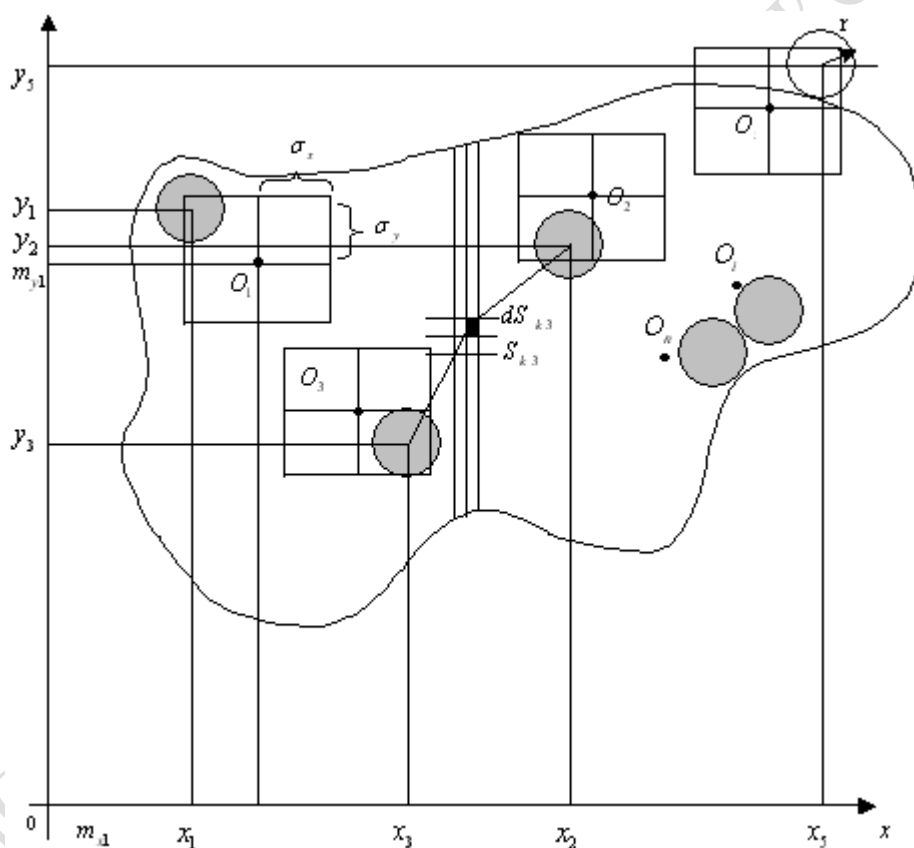


Рисунок 3.6 – Пример попадания n ракет по площадной цели для i -й реализации по методу Монте-Карло

Для моделирования из ξ_k нормально распределенных величин используем способ сумм. Допустим, что этап обстрела смоделирован и мы получили n точек попадания ракет, которые при заданном распределении точек прицеливания O_j попали так, как показано на рисунке 3.6. Вокруг каждой точки попадания опишем круг радиуса r

и подсчитаем площадь той части цели, которая покрыта хотя бы одним из кругов. Всю цель разделим на ряд элементарных площадок dS_j . Для каждой площадки j определяется расстояние ρ_{ij} от точки попадания j -й ракеты до центра площадки dS_j . Если хотя бы для одной из точек попадания j это расстояние оказалось равным или меньше r ($\rho_{ij} \leq r$), то считается dS_j пораженной, т. е. некоторая переменная-индикатор поражения площадки x_{ij} , которая в начале проверки площадок была равна 0, становится равной 1 ($x_{ij} = 1$, если $\rho_{ij} \leq r$). После чего производится суммирование пораженных площадок по всей цели $\left(S_n = \sum_{i,j} dS_j \cdot x_{ij} \right)$. Далее для l -й реализации процесса обстрела площади n ракетами ($l = \overline{1, N}$) долю $S_{цк}$ пораженной площади U_l находят как отношение числа площадок dS_{njr} к общему числу элементарных площадок в цели:

$$U_l = S_{цк} / S_{ц}. \quad (3.41)$$

Одновременно для l -й реализации вычисляют количество ракет, расстояние которых от точек попадания больше, чем границы площадки $S_{ц}$ (т. е. находят число промахов $r_{ок}$). Математическое ожидание и дисперсия доли пораженной площади определяются из соотношений

$$M_U = \frac{1}{N} \sum_{l=1}^N U_k; \quad D_U = \frac{1}{N} \sum_{l=1}^N U_k^2 - M_U^2. \quad (3.42)$$

Вероятность того, что доля пораженной площади будет не меньше, чем \tilde{U} , определяют следующим образом. С каждой l -й реализацией формируется число

$$Z_l = \begin{cases} 1 & \text{при } U_l \geq \tilde{U}, \\ 0 & \text{при } U_l < \tilde{U}. \end{cases} \quad (3.43)$$

Тогда $P(U \geq \tilde{U}) = \frac{1}{N} \sum_{l=1}^N Z_k$. Наконец, определяют среднее число промахов ракеты по цели:

$$\bar{r}_0 = \frac{1}{N} \sum_{k=1}^N r_{0k}. \quad (3.44)$$

Само же число реализаций N по методу Монте-Карло рассчитывается исходя из заданной доверительной вероятности β . Находят такое число элементов выборок, чтобы была обеспечена требуемая доверительная вероятность β . Расчетная формула имеет вид

$$N \geq S^2 / (1 - \beta) \Delta^2, \quad (3.45)$$

где $\Delta = kS_x$; $S_x = S / \sqrt{N}$; $k = 1, 2, 3, \dots$; S – выборочное средне-квадратическое отклонение;

Δ – допустимая (требуемая) точность оценки дисперсии вычислений.

4. ПЛАНИРОВАНИЕ ВЕРОЯТНОСТНЫХ ЭКСПЕРИМЕНТОВ НА ЭВМ

4.1. Кибернетический подход к исследованию на вероятностных моделях сложных систем

Обычно математические модели (ММ) представляют собой формализованную запись сложной системы и процессов, происходящих в объектах исследования. Они служат как для исследования свойств СС, так и для предсказания их поведения в различных ситуациях. Если исходить из соотношений в этих ММ, то обратим внимание на то, что эти соотношения, параметры объекта моделирования и исходная информация, поступающая на вход объекта исследования, могут быть детерминированными, вероятностными и смешанными. При построении ММ исследователю должны быть заранее известны все составляющие ММ, и вся сложность состоит в определении самих этих соотношений и в получении явного вида функциональных зависимостей между компонентами вектора откликов системы $\{Y_{jt}\}$, векторов состояний системы $\{S_m\}$ в каждый момент времени t , при изменении компонент вектора параметров объекта $\{P_{ct}\}$, компонент вектора параметров $\{x_{it}\}$ и вектора начальных условий функционирования объекта $\{U_{nt}\}$. В таких случаях СС представляется в виде «черного ящика», на вход которого поступают значения компонент этих векторов, а на его выходе определяются значения компонент векторов откликов $\{Y_{jt}\}$ и векторов состояний СС $\{S_{mt}\}$. На начальный момент исследования внутренняя структура и взаимосвязи между компонентами СС еще не установлены. Поэтому стоит задача нахождения функциональных зависимостей для детерминированных ММ в явном виде (и таким образом превращения «черного ящика» в «белый ящик»):

$$\begin{aligned} \vec{Y} &= \Phi_1(\vec{X}, \vec{U}, \vec{P}); \\ \vec{S}_m &= \Phi_2(\vec{S}_{m-1}, \vec{X}, \vec{U}, \vec{P}), \end{aligned} \quad (4.1)$$

где Φ_1, Φ_2 – обычно сложные функционалы;

m – номер нового состояния СС;

$m - 1$ – номер предыдущего состояния объекта исследования.

В вероятностных ММ (и тем более в имитационных моделях) с помощью соотношений (4.1) можно определить лишь распределения

вероятностей для компонент вектора состояний $F_{1m}(S)$ и вектора откликов $F_{ij}(Y)$, если известны распределения вероятностей для начальных условий $F_{3r}(U)$, параметров $F_{4k}(\Pi)$ и входной информации $F_{5i}(x)$. Из-за вероятностного характера компонентов векторов $\{S_m\}$, $\{Y_j\}$, $\{U_n\}$, $\{X_i\}$ в явном виде трудно найти зависимости вида (4.1). Поэтому в ММ измеряют множество статистик моделирования и в явном виде находят зависимости:

$$ST_{ml} = \varphi_{3l}(\vec{X}, \vec{U}, \vec{\pi}). \quad (4.2)$$

Затем, используя процедуру Монте-Карло, определяют математическое ожидание и выборочную дисперсию статистик моделирования:

$$\overline{ST}_m = \frac{1}{N} \sum_{l=1}^N ST_{ml}; D_m = \frac{1}{N} \sum_{l=1}^N (ST_{ml} - \overline{ST}_m)^2. \quad (4.3)$$

По этим интегральным статистикам непосредственно определяются значения компонент векторов откликов и состояний:

$$Y_j = \varphi_{jm}(ST); S_m = \varphi_m(\overline{ST}). \quad (4.4)$$

Существует принципиальное различие при получении откликов с помощью детерминированных моделей вида (4.1) и вероятностных моделей вида (4.4). Когда в детерминированных ММ отклики не зависят от времени, то они полностью определяются значениями компонент этих векторов. В вероятностных же ММ отклики зависят от момента времени, и поэтому они предварительно усредняются согласно процедуре Монте-Карло с помощью зависимостей (4.3). Это означает, что переход от «черного ящика» к «белому ящику» гораздо сложнее для вероятностных моделей, чем для детерминированных. Поэтому в имитационных моделях важную роль играют «наблюдатели» за изменениями во времени выполнения процессов внутри самих имитационных моделей СС. Отсюда в ряде случаев появляется необходимость в определении интегральных состояний во время имитации S_m . В этом и состоит основная идея кибернетического подхода к исследованию вероятностных СС.

Если исходить из способа дальнейшего использования ММ для изучения СС, то ММ можно разделить на три класса:

- аналитические, полностью определяемые выражением (4.1);

– численные модели, когда в общем виде, вместо явных зависимостей из-за их сложной структуры, стараются получить числовые результаты решения уравнений вида (4.1) при конкретных начальных расчетных данных компонент векторов \vec{X} , \vec{U} и $\vec{\pi}$;

– имитационные модели, когда СС сложна, имеет вероятностную структуру и требуется получение результатов вида (4.4), или же необходим высокий уровень детализации представления процессов СС, возникающих во времени t_0 в самой СС.

Для аналитических ММ процессы функционирования СС записываются в виде некоторых функциональных соотношений (алгебраических и интегро-дифференциальных). При этом всегда стремятся в общем виде получить явные зависимости вида (4.1). Поэтому предсказательные возможности аналитических ММ очень велики. Если математические зависимости вида (4.1) сложны, то зачастую используется ЭВМ, позволяющая быстро вычислить значения компонент векторов \vec{S} и \vec{Y} для любых заданных значений \vec{X} , \vec{U} , $\vec{\pi}$ или их распределений (для вероятностных ММ).

Для численных ММ используются численные методы решения зависимостей Φ_1 и Φ_2 в (4.1). Причем в явном виде эти зависимости удается получить на практике весьма редко. Поэтому исследователи вначале стремятся получить аналитическое решение задачи. При этом они вынуждены идти на упрощение реальной ситуации, чтобы иметь возможность изучать некоторые общие свойства СС. В отдельных случаях приближенное решение задачи о поведении СС исследователей удовлетворяет. Обычно для них достаточно результатов, полученных с помощью качественных методов на численной модели. Для этой цели создан мощный математический аппарат (интегральные и разностные уравнения, теория вероятностных процессов, численные методы, методы оптимизации и т. д.)

В имитационных моделях моделируемый алгоритм поведения СС приближенно воспроизводит сам процесс-оригинал в смысле его функционирования во времени. При этом имитируются элементарные явления, составляющие процесс в СС, с сохранением их логической структуры и порядка его протекания во времени. Таким образом, на ЭВМ реализуется специальный алгоритм, который воспроизводит формализованный процесс поведения СС. Этот алгоритм по исходным данным $(\vec{U}, \vec{\pi}, \vec{X})$ позволяет получить статистику об изменении во времени t сначала $S\vec{T}_t$, а затем вычислить значения компонент векторов \vec{S} и \vec{Y} . В этом алгоритме можно выделить три функциональные

части: моделирование элементарных процессов; учет их взаимодействия и объединение в единый процесс; обеспечение согласованной работы отдельных компонентов ИМ СС на ЭВМ. Влияние случайных факторов на течение процессов в ИМ СС имитируется с помощью ГСЧ с заданными вероятностными характеристиками. В ходе имитации постоянно фиксируются: статистика (ST_l) , состояния системы (\bar{S}) и характеристика вектора откликов (\bar{Y}) . Эта статистика должным образом обрабатывается в ходе имитации, либо накапливается и, по окончании заданного интервала моделирования (T_m), обрабатывается статистическими методами. Как видим, идея имитации привлекательна по своей простоте, но дорога по реализации. Поэтому ИМ применяются только в тех случаях, когда другие способы моделирования СС неэффективны.

4.2. Концептуальная модель исследования вероятностной сложной системы

Поскольку моделирование вероятностей СС реализуется с помощью имитационных или вероятностных моделей, построенных на основе многократного использования процедуры Монте-Карло, то особую роль приобретает планирование имитационных экспериментов (ИЭ), позволяющее уменьшить ресурсоемкость исследований на ЭВМ. Перед построением планов натурных экспериментов (НЭ) с реальной СС или же перед построением планов ИЭ необходимо составить концептуальную модель исследования (КМИ). Блок-схема КМИ вероятностной СС, построенная на основе использования кибернетического подхода к исследованию вероятностной СС, приведена на рисунке 4.1.

Согласно этому подходу вероятностная СС представлена «черным ящиком», на входы которого во время l -й реализации процедуры Монте-Карло ($l = \overline{1, N}$) действуют векторы: $\{X_{hil}\}$, $\{U_{hil}\}$, $\{П_{hkl}\}$, $\{G_{hq}\}$, $\{\xi_r\}$.

Здесь $\{X_{hil}\}$ – значения вектора входной информации l -й реализации, формируемые по функциям распределения $F_{5i}(x)$ в h -м варианте исследования СС;

$\{U_{hil}\}$ – вектор начальных условий моделирования h -го варианта СС l -й реализации модели, формируемый по функции распределения $F_{3i}(U)$;

$\{П_{hkl}\}$ – вектор параметров h -го варианта модели СС в l -й реализации модели, формируемый по функции распределения $F_{4k}(П)$;

$\{G_{hq}\}$ – множество постоянных характеристик модели СС, определяющих, обычно, влияние внешней среды, в которой функционирует исследуемая СС;

$\{\xi_r\}$ – вектор неконтролируемых случайных воздействий внешней среды, о которых известна информация только об их существовании и их влияние оценивается суммарной ошибкой моделирования СС.

В ходе моделирования l -й реализации ИМ фиксируется вектор статистик имитации $\{ST_{hml}\}$ h -го варианта модели СС. Для каждой компоненты вектора по выборке статистик определяются по формулам (4.3) математическое ожидание и выборочная дисперсия значений статистик $(\overline{ST_{hm}}, D_{ST_{hm}})$.

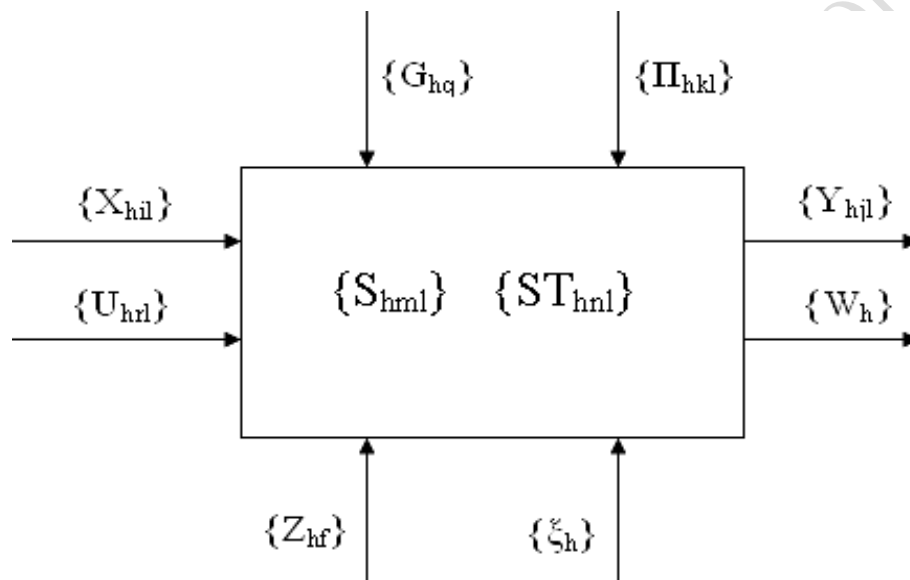


Рисунок 4.1 – Схема концептуальной модели исследования вероятностной СС

В общем случае значений компонент вектора, наряду с фиксацией статистик моделирования в l -й реализации h -го варианта модели, фиксируется множество состояний модели СС $\{S_{hmlt}\}$ в моменты времени t .

После усреднения, согласно процедуре Монте-Карло, определяется множество наиболее вероятных состояний системы в моменты времени t $\{\tilde{S}_{hmlt}\}$. Используя \tilde{S}_{hmlt} , формируются временные диаграммы (ВДГ) перехода модели системы в различные состояния. В совокупности ВДГ позволяют исследователю следить за динамикой функционирования СС.

По значениям ST_{hml} , полученным с помощью формул (4.4), определяются значения компонент вектора откликов h -го варианта l -й реализации модели СС $\{Y_{hjl}\}$. Согласно процедуре Монте-Карло,

по завершении N реализаций ИМ СС, по выборкам $\{Y_{hjl}\}$, $l = \overline{1, N}$ определяются оценки математического ожидания и дисперсии компонент вектора откликов по формулам:

$$\bar{Y}_{hj} = \frac{1}{N} \sum_{l=1}^N Y_{hjl}; D_{ynj} = \frac{1}{N} \sum_{l=1}^N (Y_{hjl} - \bar{Y}_{hj})^2; \quad (4.5)$$

Для сравнения друг с другом h -х вариантов модели СС обычно стараются «свернуть» вектор $\{\bar{Y}_{hj}\}$ к скалярной величине W_h по формуле:

$$W_h = Z(\bar{Y}_{hj}). \quad (4.6)$$

Способы «свертки» вектора \vec{Y} в скалярную величину и выбор оптимального варианта h_0 модели СС рассматриваются в главе 6. В итоге обработки статистики имитации и вычисления откликов моделирования и обобщенного показателя вариантов СС формируются интегральные характеристики модели и временные диаграммы изменения статистик и состояний модели системы. Имея значения характеристик КМИ, приведенные на рисунке 4.1, исследователь может предварительно оценить изменения откликов во времени в системе и сравнить варианты друг с другом по обобщенному показателю $W(Z)$ для каждой компоненты вектора откликов \vec{Y} .

4.3. Основные определения планирования экспериментов на ЭВМ с моделями вероятностных систем

С точки зрения теории планирования экспериментов (ПЭ) **входными** наблюдениями и **управляемыми** переменными модели являются компоненты множеств $\{X_{hi}\}$, $\{U_{hr}\}$, $\{P_{hk}\}$, поскольку исследователь может варьировать их значениями по своему усмотрению. На систему (или на ее модель) может воздействовать множество **неуправляемых** переменных $\{Z_{hl}\}$, поскольку их значения не зависят от исследователя, но в модели СС они должны быть представлены в тех случаях, когда их влияние на поведение системы и ее отклики являются значимыми. Множества $\{G_{hr}\}$ и $\{U_{hr}\}$ можно также отнести к **неуправляемым**

переменным, поскольку исследователь должен задать их значения такими, какими они были при исследовании реальной СС. Множество $\{\xi_n\}$ также представляет собой неуправляемые параметры, но их влияние исследователь должен оценить, определив при этом общую ошибку имитации. Выходными характеристиками ИМ являются вектор откликов $\{Y_{hj}\}$ и обобщенный показатель W_h . Назовем $\{X_{hi}\}$ и $\{П_{hk}\}$ факторами. Конкретное значение каждой компоненты этих векторов назовем уровнем фактора.

Первичными факторами назовем те входные переменные и параметры модели, в изучении влияния которых исследователь заинтересован. Остальные факторы назовем **вторичными**, поскольку их влиянием нельзя пренебречь. Однако качественная оценка этого влияния не входит в цели исследования СС. Наоборот, исследователь зачастую пытается ослабить это влияние путем специально подобранного плана эксперимента (ПЭ), чтобы выделить эффекты первичных факторов. Неуправляемые переменные вместе с $\{\xi_n\}$ вносят случайные помехи.

Моделирование СС может преследовать следующие цели:

- определение вида зависимости (4.4) по данным натурального или модельного эксперимента;
- нахождение такого сочетания параметров и входных переменных, которые обеспечивают максимум выражения (4.6);
- изучение динамики изменения во времени состояний СС в виде множества $\{S_{it}\}$ и поиска «узких мест» в структуре модели СС.

В первом случае речь идет об определении регрессионной зависимости либо по данным натурального эксперимента (НЭ), либо по данным имитационного эксперимента (ИЭ). Зависимости (4.4) уточняются с помощью системы регрессионных моделей:

$$Y_j = \varphi_{jm} [(\vec{X}, \vec{П}, \vec{G}, \vec{U}); \theta], \quad (4.7)$$

где $\theta = (\theta_0, \theta_1, \theta_2, \dots, \theta_d)$ – вектор параметров регрессионной модели, подлежащих определению.

Как правило, вид таких моделей ищут среди компонентов вектора Y :

$$- Y_0 = v_0 \text{ (полином нулевой степени);} \quad (4.8)$$

$$- Y_1 = v_0 + v_1 \cdot x_1 + v_2 \cdot x_2 \text{ (полином первой степени);} \quad (4.9)$$

$$- Y_2 = v_0 + v_1 x_1 + v_2 x_2 + v_{12} x_1 x_2 + v_{11} x_1^2 + v_{22} x_2^2 \text{ (полином второй степени);} \quad (4.10)$$

Если вид полинома выбран, то необходимо составить план эксперимента и затем осуществить либо НЭ, либо МЭ. Отметим, что любой

эксперимент представляет собой систему операций с реальной системой или ее моделью, воздействий и наблюдений, которые в совокупности направлены на получение информации о СС в ходе испытаний. Опыт является отдельной элементарной частью эксперимента. Он воспроизводит исследуемое явление в СС при определенных условиях проведения эксперимента (составе параметров модели) и при возможности регистрации результатов НЭ или МЭ, которые осуществляются согласно плану эксперимента. В общем случае **план эксперимента (ПЭ)** представляет собой совокупность данных, определяющих число, условия и порядок реализации опытов. Поэтому планирование эксперимента устанавливает совокупность действий, направленных на разработку тактики эксперимента от начальных (выбор ПЭ) до заключительных этапов изучения СС (обработка экспериментальных данных и создание математической модели СС). ПЭ позволяет получить максимум информации при минимуме опытов на СС. При ПЭ модель СС должна удовлетворять двум основным требованиям:

- имеется воспроизводимость исследований результатов эксперимента на объекте, т. е. разброс значений отклика при повторении опытов не превышает точности эксперимента;

- обеспечена управляемость объекта исследования либо через непосредственные воздействия на него, либо с помощью модели объекта, использующей управление.

Исходя из второго требования, эксперимент бывает пассивным и активным. При пассивном эксперименте существуют только неуправляемые факторы. В этом случае осуществляется только регистрация статистик ST_i , состояний S_i , откликов Y_j без каких-либо возмущений функционирования СС. В таких случаях решаются только задачи идентификации явлений в СС. Активные эксперименты являются основным объектом планирования экспериментов. Они проводятся при условии, что существуют управляемые параметры (\vec{X} и \vec{P}) и исследователь изменяет компоненты этих векторов целенаправленно в соответствии с ПЭ. Для активного эксперимента область действия определяется областью возможных значений компонент векторов \vec{X} и \vec{P} . Если область планирования задается интервалами возможного изменения факторов $X_{i\min} \leq X_i \leq X_{i\max}$; $i = \overline{1, n}$, то она представляет собой, при соответствующем масштабировании, гиперкуб.

Точка плана означает упорядоченную совокупность значений факторов, соответствующих условиям проведения опыта. Точке плана с номером g соответствует вектор-строка:

$$X_g^T = \|X_{1g}, X_{2g}, \dots, X_{ng}\|. \quad (4.11)$$

Общая совокупность таких векторов X_g образует ПЭ, а совокупность различных векторов определяет **спектр плана**. Вводится понятие об уровнях фактора. Основной уровень фактора X_i^0 , $i = \overline{1, n}$ указывает такие условия эксперимента, которые представляют основной интерес в данный момент времени. Обычно вектор-строка $(X^0)^T = \|X_1^0, X_2^0, \dots, X_n^0\|$ задает в факторном пространстве точку, являющуюся центром области планирования. Если координаты $X_j^0 = (X_{j\min} + X_{j\max}) / 2$, а интервал (шаг) формирования фактора X_i равен $\Delta X_i = (X_{i\min} - X_{i\max}) / 2$, то

$$X_{i\max} = X_i^0 + \Delta X_i; \quad X_{i\min} = X_i^0 - \Delta X_i. \quad (4.12)$$

В практике ПЭ осуществляется операция нормализации фактора, которая сводится к изменению начала отсчета координатных осей и их масштаба в соответствии с соотношением:

$$X_i = (X_i - X_i^0) / \Delta X_i; \quad i = \overline{1, N}; \quad N - \text{число факторов}. \quad (4.13)$$

Для переменной X_i в стандартизованном масштабе начало координат совмещено с центром эксперимента, а в качестве единицы измерения используется шаг варьирования фактора. Вклад фактора в величину отклика Y_j при переходе от минимального (нижнего) к максимальному (верхнему) уровню называется эффектом фактора (основным или главным эффектом фактора). Эффект фактора может зависеть от уровня, на котором находится другой фактор. При этом имеет место эффект взаимодействия факторов.

Для организации ПЭ вначале составляется матрица плана, которая представляет собой стандартную форму записи условий проведения эксперимента в виде прямоугольной таблицы, строки которой отвечают номерам опытов, а столбцами являются типы факторов. Размер матрицы плана равен $(L \times n)$, где L – число опытов, в n – число факторов. Она может иметь совпадающие строки. Каждый (ij) -й элемент матрицы плана равен уровню j -го фактора в i -м опыте. **Матрица спектра плана** составляется из всех строк матрицы плана и отличается наличием значений хотя бы одного фактора. Поэтому размер матрицы спектра равен $N \times n$. Все строки этой матрицы различны. Матрица

дублирования представляет собой квадратную диагональную матрицу, у которой диагональные элементы равны числам параллельных опытов в соответствующих точках спектра плана и имеет вид:

$$R = \begin{pmatrix} r_1 & 0 & \dots & 0 \\ 0 & r_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & r_N \end{pmatrix} \quad (4.14)$$

Итак, ПЭ может быть задан либо матрицей плана, либо матрицей спектра плана в совокупности с матрицей дублирования. Во втором случае речь идет о выборе такого сочетания уровней факторов $\{X_i\}$, при котором обеспечивается максимум обобщенного отклика W . Здесь возможны два случая. Либо вид функции (4.6) известен, и тогда осуществляется поиск экстремума одним из способов векторной оптимизации [24]; либо зависимость (4.6) неизвестна, и исследователь ограничен конечным числом комбинаций уровней факторов. В такой ситуации исследователь имеет только возможность простого перебора вариантов сочетания факторов и задача сводится к выбору той комбинации факторов, которая обеспечивает максимум обобщенного показателя качества вариантов организации СС ($\max W$), вычисляемого из выражения (4.6). В такой ситуации говорить об оптимизации структуры СС нельзя, и поэтому мы будем говорить о выборе рационального варианта сочетания факторов (лучшего в смысле критерия (4.1) из рассмотренных вариантов).

В третьем случае речь идет о способе «свертки» вектора состояний СС $\{S_{mt}\}$, который позволяет при необходимости в любой момент времени t воспроизвести диаграмму состояний системы и на основе анализа этой диаграммы определить «узкие места» функционирования СС. Частым способом такой «свертки» $\{S_{mt}\}$ является построение матрицы $\|P_{mf}\|$ вероятностей переходов СС из состояния m в состояние f , предполагая при этом возможность представления смен состояний СС как полумарковский процесс. Длительности τ_{mf} нахождения СС в состоянии m при переходах системы в состояние f описывается функцией распределения $F_{mf}(\tau)$. Поэтому, кроме матриц вероятностей $\|P_{mf}\|$, для описания переходов СС, как полумарковского процесса, необходимо задание матрицы, элементами которой будут функции распределения $\|F_{mf}(\tau)\|$, а также задание начальных условий для таких

переходов (U) и способов завершения полумарковского процесса (либо это будет вероятность перехода в конечное состояние P_{mf} , либо это будет общее число состояний (v_k) исследуемой СС). Ниже мы рассмотрим более подробно все три случая анализа результатов эксперимента с моделью СС.

4.4. Планы построения регрессионной модели поведения сложной системы

В основе построения таких моделей лежит регрессионный анализ, позволяющий строить модели вида (4.9) и (4.10). Для использования регрессионного анализа нет необходимости различать факторы управляемые (X) и неуправляемые (Z). Поэтому полагаем, что Z входит в X . Кроме того, поскольку нельзя объять необъятное, ограничимся лишь наиболее распространенными линейными, по параметрам, регрессионными моделями. При необходимости изучения регрессий более высокого порядка можно рекомендовать работы [4; 56; 57]. Для линейных регрессий используются планы первого порядка. Такие планы используются для экспериментального получения регрессионных моделей вида:

$$\varphi(x) = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n, \quad (4.15)$$

где $b_i = \beta_i / \Delta x_i$;

$$b_0 = \beta_0 + \sum_{i=1}^n \beta_i x_i^0, \quad -1 \leq x_i \leq +1, \quad i = \overline{1, n}.$$

Известно несколько разновидностей планов первого порядка: однофакторный эксперимент (ОЭ); полный факторный эксперимент (ПФЭ); дробный факторный эксперимент (ДФЭ); насыщенный план (симплекс-план). Рассмотрим суть каждого из этих типов планов.

Однофакторный эксперимент. Он предусматривает поочередное варьирование каждого из факторов, в то время как все остальные факторы стабилизированы на некотором уровне (чаще всего на нулевом). Можно считать, что в ходе варьирования фактор X_i принимает значения $X_{iB} = X_i^0 + \Delta X_i$ и $X_{iH} = X_i^0 - \Delta X_i$, а все другие факторы фиксированы на своем базовом уровне X_j^0 ($j \neq i$). Для нормированных величин $X_{iB} = +1$; $X_{iH} = -1$. Матрица спектра ОЭ должна иметь вид:

$$X = \begin{pmatrix} -1 & 0 & \dots & 0 \\ +1 & 0 & \dots & 0 \\ 0 & -1 & \dots & 0 \\ 0 & +1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & -1 \\ 0 & 0 & \dots & +1 \end{pmatrix}. \quad (4.16)$$

Геометрический смысл спектра данного плана для двумерного случая означает, что точки располагаются на концах гиперкуба со сторонами, равными 2. Обычно ОЭ редко используется в силу его низкой точности и отсутствия возможности оценить эффект взаимодействия факторов.

Полный факторный эксперимент 2^n . Спектр ПФЭ содержит в себе все возможные комбинации n факторов на всех уровнях их изменения. Если рассматривать вариант, когда каждый из n факторов меняется на двух уровнях (что достаточно для построения линейной регрессии), то общее число элементов спектра плана равно 2^n . Такой план обозначают в виде ПФЭ 2^n . Например, ПФЭ 2^2 имеет следующий спектр плана:

$$X = \begin{pmatrix} -1 & -1 \\ +1 & -1 \\ -1 & +1 \\ +1 & +1 \end{pmatrix}. \quad (4.17)$$

Отметим, что геометрически точки этого плана размещаются в вершинах квадрата с ребрами, равными 2.

С помощью ПФЭ 2^n возможно получение регрессионных моделей вида:

$$\begin{aligned} \varphi(x) = & b_0 + \sum_{i=1}^n b_i x_i + \sum_{i_1=1}^n \cdot \sum_{i_2=2}^n b_{i_1} \cdot b_{i_2} x_{i_1} x_{i_2} + \\ & + \sum_{i_1=1}^n \sum_{i_2=1}^n \sum_{i_3=1}^n b_{i_1 i_2 i_3} X_{i_1} X_{i_2} X_{i_3} + \dots + b_{12\dots n} X_{i_1} X_{i_2} \dots X_{i_n}; i_1 < i_2. \end{aligned} \quad (4.18)$$

Как правило, такие модели содержат:

- один коэффициент b_0 (свободный член уравнения регрессии);
- n коэффициентов b_i (линейных членов уравнения регрессии);
- C_n^2 коэффициентов $b_{i_1 i_2}$ (при парных взаимодействиях факторов);
- C_n^3 коэффициентов $b_{i_1 i_2 i_3}$ (при тройных взаимодействиях факторов);

и так далее, включая один коэффициент $b_{12\dots n}$, при взаимодействиях факторов максимального n -го порядка.

Общее число коэффициентов регрессии в этой модели равно $2^n = 1 + n + C_n^2 + C_n^3 + \dots + C_n^n$; т. е. 2^n является насыщенным. План называется **насыщенным**, если общее число наблюдений равно числу неизвестных параметров регрессионной модели. Желательно, чтобы любой реальный план был близок к насыщенному. Это важно на этапе предварительного исследования, когда требуется получить хотя бы представление об объекте исследования, но с минимальными затратами ресурса опытов. Различают четыре типа матриц планирования:

1) *симметричность* относительно центра эксперимента, означающая, что алгебраическая сумма вектор-столбца каждого фактора равна нулю:

$$\sum_{i=1}^L X_{ik} = 0; \quad i = \overline{1, k}, \quad (4.19)$$

где k – номера компонент вектора;
 L – общее число факторов;

2) *условие нормировки*, означающее, что сумма квадратов элементов каждого вектор-столбца равна числу u опытов:

$$\sum_{i=1}^L X_{ij}^2 = L; \quad (4.20)$$

3) *условие ортогональности*, показывающее, что сумма почленных произведений любых двух вектор-столбцов матрицы плана равна нулю:

$$\sum_{i=1}^L X_{ij} X_{ui} = 0; \quad j \neq u; \quad u = \overline{1, n}. \quad (4.21)$$

4) *ротатабельность*, предполагающая, что точка в матрице планирования подбирается таким образом, чтобы точность предсказания

значений отклика была одинакова на равных расстояниях от центра эксперимента и не зависела от направления движения в факторном пространстве.

Свойство ортогональности дает возможность снизить вычислительные затраты расчета коэффициентов регрессии. Ротатабельность обеспечивает равномерность распределения дисперсии предсказанных значений отклика в изучаемой области факторного пространства. В этом смысле ПФЭ 2^n оптимален при построении линейной модели. Недостатком ПФЭ 2^n является быстрый рост количества опытов при увеличении n , например при $n = 10$ число опытов равно 1 024. Как правило, исследователь ограничивается на первых порах линейной моделью с учетом дополнительных парных взаимодействий. Количество опытов в ПФЭ значительно превосходит число определяемых коэффициентов линейной модели. Поэтому ПФЭ обладает большой избыточностью и целесообразно сократить число опытов за счет использования той информации, которая не очень существенна при построении линейных моделей.

Дробный факторный эксперимент 2^{n-p} . Дробный факторный эксперимент позволяет исследовать линейные регрессионные модели, а также модели, содержащие дополнительно некоторые взаимодействия факторов при меньшем числе наблюдений по сравнению с ПФЭ. Подобная экономия числа наблюдений достигается за счет использования априорной информации о свойствах СС. Эта информация должна быть сформулирована в виде списка переменных-факторов и их взаимодействий. Например, X_0, X_1, X_2, X_3 . Здесь X_0 – фиктивная переменная, соответствующая коэффициенту b_0 в линейной регрессионной модели. Под X_1, X_2 и X_3 можно понимать либо одну переменную, либо произведение переменных. В последнем случае имеет место переобозначение переменных. Поэтому наибольшее распространение получили регулярные планы ДФЭ типа 2^{n-p} , где n – число факторов, p – степень дробности ПФЭ. Соответствующие типы ДФЭ называют репликами с указанием их степени дробности. Например, ДФЭ 2^{n-1} означает, что исследователем получена 1/2 реплика ПФЭ 2^n . Аналогично ДФЭ 2^{n-2} означает 1/4 реплики ПФЭ 2^n . *Регулярность* плана ДФЭ 2^{n-p} состоит в том, что матрица спектра плана является некоторой неслучайной частью матрицы спектра типа ПФЭ 2^n .

Основная идея ДФЭ состоит в том, чтобы использовать такой экономичный план, который обеспечивал бы оценку влияния только переменных, находящихся в списке существенных, и не учитывал возможное воздействие прочих переменных. Если список существенных переменных содержит $(d + 1)$ элемент, то для того, чтобы оценить

по отдельности коэффициенты регрессии при каждой из них, необходимо выполнение условия $N = 2^{n-1} \geq d + 1$. Это условие при заданном числе факторов n и известном числе существенных переменных $d + 1$ позволяет найти минимально допустимую степень дробности p . Тем самым определяется общее число строк в матрице спектра плана ДФЭ. ДФЭ строится следующим образом: для $(n - p)$ факторов, условно называемых основными, строится матрица ПФЭ, а для p факторов, называемых *дополнительными*, уровни выравнивания в опытах выбираются на основании априорной информации. Генеральное соотношение – это формальное равенство, показывающее, знаки каких основных переменных, стоящих в правой части равенства (например, $X_3 = X_1X_2$), необходимо перемножить для получения знака дополнительного фактора (уровня варьирования), чтобы матрица ДФЭ оказалась ортогональной, нормированной и симметричной. Можно, например, построить и другую реплику ПФЭ $X_3 = -X_1X_2$. Аналогичным образом для ПФЭ 2^3 можно выбрать две 1/2 реплики, не имеющих общих строк. Поэтому подобная разновидность планов получила широкое практическое применение.

Насыщенным планом первого порядка называется план, который позволяет оценить коэффициенты линейной модели относительно n факторов, а его спектр содержит $(n + 1)$ точку. Совокупность $n + 1$ точек насыщенного линейного плана задает в n -мерном факторном пространстве вершины геометрической фигуры, называемой n -мерным симплексом, и поэтому подобные планы называют симплекс-планами. Например, одномерный симплекс – это отрезок прямой, двумерный симплекс-треугольник, n -мерный – выпуклый многоугольник с $(n + 1)$ вершинами. Симплекс-планы используются, как правило, на стадии предварительного исследования СС, когда желательно применять наиболее экономичные насыщенные планы, а также при построении процедур поиска экстремальных значений целевой функции.

При практической реализации планов первого порядка исследователь выполняет следующие действия:

- 1) выбирает центр плана X^0 , т. е. значения базовых уровней x_i^0 , $i = \overline{1, n}$;
- 2) определяет шаги варьирования Δx_i по каждому фактору (обычно $\Delta x_i = 0,3 \div 0,5$ допустимого диапазона изменения фактора x_i);
- 3) устанавливает общее число опытов, т. е. формируется план эксперимента в целом, что включает также построение спектра плана и выбор элементов матрицы дублирования R (4.14).

В планировании первого порядка применяется, как правило, равномерное дублирование опытов, т. е. в матрице R все диагональные

элементы равны между собой: $r_1 = r_2 = \dots = r_N = m$. Эксперименты реализуются отдельными сериями. Каждая серия включает проведение N опытов, соответствующих всем строкам матрицы спектра плана. Количество серий равно m . Внутри каждой серии порядок реализации опытов должен быть случайным. Для этой цели существует процедура *рандомизации*, т. е. обеспечивается случайный порядок реализации строк матрицы плана. Таким способом устраняются (в математических моделях) смещения, вызванные действием неконтролируемых систематических переменных.

Конечным результатом проведения эксперимента является таблица 4.1 значений отклика Y . Данные, полученные с помощью планов первого порядка, сосредоточенные в таблице 4.1, обрабатываются согласно семишаговому алгоритму.

Таблица 4.1 – Представление результатов эксперимента

	Серия опытов				
	Y_1	Y_2	Y_3	Y_4	Y_m
1					Y_{1m}
2					Y_{2m}
3					Y_{3m}
4					Y_{4m}
–	–	–	–	–	–
N	Y_{N1}	Y_{N2}	Y_{N3}	Y_{N4}	Y_{Nm}

Шаг 1. Проверка воспроизводимости эксперимента. Находят средние значения и выборочную дисперсию:

$$\bar{Y} = \sum_{l=1}^m Y_{gl/m}; S_g^2 = \sum_{l=1}^m (Y_{gl} - \bar{Y})^2 / (m-1); \quad (4.22)$$

Проверяется гипотеза $H_0 : \delta_1^2 = \delta_2^2 = \dots = \delta_N^2$, где δ_g^2 , $g = \overline{1, n}$ значения дисперсии шума, соответствующие спектрам плана.

Эта проверка проводится с помощью критерия Кохрена, статистика которого имеет вид:

$$G = S_{\max}^2 / \sum_{g=1}^N S_g^2, \quad (4.23)$$

где S_{\max}^2 – наибольшая из оценок S_g^2 ; $g = \overline{1, N}$.

Шаг 2. Вычисление оценок коэффициентов регрессии по методу наименьших квадратов.

Шаг 3. Определение дисперсии оценок коэффициентов регрессии.

Шаг 4. Проверка значимости коэффициентов регрессии.

Шаг 5. Расчет предсказанных (по уравнению регрессии) значений отклика в точках спектра (4.22) и анализ точности предсказания.

Шаг 6. Проверка адекватности уравнения регрессии (по критерию Фишера).

Шаг 7. Проверка работоспособности регрессионной модели.

Перед проведением эксперимента устанавливается, какие из факторов, которые влияют на выходную величину отклика СС, нужно включать в состав (4.22). При этом определяют минимальный набор факторов. Если число предполагаемых факторов невелико (порядка $6 \div 8$), то для предварительного изучения СС можно применить ПФЭ или ДФЭ. Далее определяют оценки коэффициентов модели и проверяют их статистическую значимость. По абсолютному значению факторов осуществляется их ранжирование согласно степени влияния на величину отклика.

Планы второго порядка используются для изучения регрессионной модели в виде полинома второй степени. Они используются тогда, когда план первого порядка не позволил получить адекватную регрессионную модель или априори известно, что регрессия обладает существенно нелинейными свойствами. Квадратичная модель содержит $k = 1 + 2n + C_n^2 = ((n+1)(n+2))/2$ членов, что в $(n+2)/2$ раз больше, чем для линейной модели. Соответственно возрастает и минимальное необходимое количество точек в спектре плана. Для исследования квадратичной зависимости каждый из факторов должен изменяться, по крайней мере, на трех уровнях. Это вызывает необходимость постановки большого числа опытов. Полный факторный эксперимент содержит 3^n точек. Например, для четырехфакторного эксперимента число точек плана в пять раз больше, чем в плане первого порядка.

На практике часто используется *центральный композиционный план* (ЦКП), позволяющий сократить число точек до $L = n_c + 2n + n_0$, где n_c – число точек ПФЭ 2^n , или дробной реплики; n_0 – число опытов в центре плана. План называется композиционным, если в его спектр, в качестве составной части, входят точки спектра плана, который был реализован при построении более простой модели. Это дает возможность применить точки спектра плана (одного из предшествующих этапов исследования) в качестве точек части спектра плана следующего этапа. Она выполняется тогда, когда модель, полученная

на последующем этапе (например, полином степени k), не устраивает исследователя и требует усовершенствования (например, перехода к полиному степени $k + 1$). ЦКП состоит из трех частей. Первая часть – ядро плана – это ПФЭ 2^n илиДФЭ 2^{n-p} . Вторая часть ЦКП – «звездные» точки, расположенные на координатных осях на расстоянии $\pm d$ от центра эксперимента. Общее число таких точек равно 2^n . Третья часть ЦКП – постановка опытов в центре плана (число таких опытов $n_0 - 1$). Например, если рассматривать трехфакторную модель, то идея ЦКП состоит в том, что ставят серию из 8 опытов, располагая опыты в вершине куба, т.е. реализуется линейный план ПФЭ 2^3 . Затем к этим точкам добавляют еще шесть «звездных» точек, находящихся на дополнительных линиях, которые выходят из центра куба. В центре ставят еще одну точку. В результате общее количество равно 15 (Тогда как для ПФЭ 3^3 оно равно 27). Как видим, удается сэкономить почти половину опытов. В общем случае ЦКП не ортогонален. Ортогональность достигается специальным образом выбранным «звездным» планом λ и преобразованием квадратичных значений факторов. Недостатком ЦКП является неодинаковая точность математической модели в факторном пространстве. Оптимальным планированием второго порядка считается *ротатабельное планирование*. Такие планы годятся для любого числа факторов. Их можно построить, комбинируя вместе вершины правильных геометрических фигур и центральные точки. Использование ротатабельных планов гарантирует, что в любой точке пространства факторов стандартное отклонение отклика зависит только от расстояния этой точки до центра и не зависит от направления. Построение ротатабельных планов требует высокой квалификации исследователя. Отметим, что композиционное планирование представляет собой один из возможных способов построения ротатабельных планов, в этом случае выбирают такое «звездное» плечо, при котором $n_c = 2^n$ и $\lambda = 2^{n/4}$. Такое планирование называют центральным ротатабельным равномер-планированием. При этом величина, обратная дисперсии отклика, остается постоянной в области факторного пространства, ограниченной радиусом, равным единице.

4.5. Другие планы эксперимента на ЭВМ моделей сложных систем

Решение большого числа задач управления объектами связано с оптимизацией. Поиск экстремума функции отклика происходит путем

исследования поверхности отклика. Эти исследования осуществляются посредством измерения поверхности отклика в различных точках факторного пространства. Возникает вопрос: какой должна быть стратегия планирования эксперимента, чтобы число опытов, необходимых для нахождения экстремума обобщенного отклика W или близких к нему значений, было как можно меньше. Решение этой задачи обычно выполняется в два этапа:

- осуществляются поисковые движения к области экстремума;
- уточняются расположения экстремальной точки либо с помощью дополнительных поисковых опытов, либо с помощью математической модели SS в области экстремума откликов, которая определяется с помощью организованных опытов.

Сами по себе поисковые методы определения экстремума, при активном эксперименте, бывают двух типов: классические и факторные. Среди классических методов поиска оптимальной области наиболее известным является метод Гаусса-Зейделя, при котором все факторы, кроме одного, поочередно фиксируются, т. е. эксперимент проводится по схеме однофакторного эксперимента. Ставится серия опытов при различных значениях незафиксированного фактора. Двигаясь параллельно одной из осей факторного пространства, исследователь находит наилучшее значение для рассматриваемого разреза поверхности отклика. Затем в этой наилучшей точке ставится следующая серия опытов при различных значениях следующего фактора (остальные факторы при этом являются зафиксированными). Таким образом, осуществляется движение параллельно следующей оси факторного пространства. Последовательное прохождение всех осей факторного пространства составляет первый цикл исследования. Эту процедуру повторяют до получения оптимума или до попадания в некоторую точку, любое движение из которой ухудшает значение отклика. Данный метод требует большого количества опытов.

Применение градиентных методов позволяет быстрее отыскать экстремум. Широко распространен *метод крутого восхождения* [56]. Основная идея этого метода заключается в том, что на начальном этапе на основании ПФЭ и ДФЭ получают простейшие линейные модели в качестве приближенного описания некоторой части функции отклика (может быть далекой от экстремума). Коэффициенты модели, полученной в результате ПФЭ или ДФЭ, пропорциональны проекциям вектор-градиента и позволяют оценить само направление градиента, т. е. направление крутого склона поверхности отклика $W(z)$. Затем вдоль этого направления совершается постепенное пошаговое движение к области экстремума. При достижении области экстремума про-

водится ее исследование. В этом случае строятся планы более высокого порядка, так как поверхность отклика $W(z)$ вблизи экстремума обычно плохо аппроксимируется гиперплоскостью. Поэтому используются центральные композиционные и ротатабельные планы второго порядка.

На практике исследователь не имеет информации о виде функции отклика $W(X, \pi, z)$. В таких случаях за центр плана принимается множество значений входной величины $\{X_0\}$, имевшее место при натурном эксперименте на СС при фиксированных значениях вектора параметров π и существующих неуправляемых воздействий $\{z\}$ на СС. Кроме того, число возможных комбинаций уровней фактора $\{x_i\}$ ограничено, и поэтому речь может идти только о выборе наилучшей комбинации уровней факторов x_i , т. е. определяется рациональный состав комбинаций $\{x_i\}$. Здесь может помочь метод «секущих плоскостей», суть которого заключается в следующем. Во-первых, вместо трех уровней (x_i^-, x_i^0, x_i^+) рассматриваются еще два дополнительных уровня, означающих, что между центром фактора и крайними значениями уровней выбираются по два уровня $(x_i^-, x_{1i}, x_i^0, x_{2i}, x_i^+)$. Поэтому, если число всевозможных комбинаций уровней факторов резко возрастает, то из них целенаправленно выбираются те комбинации, которые практически достижимы в данном эксперименте.

Во-вторых, фиксируют все факторы, кроме одного, который варьируется уже на 5 уровнях. Например, пусть изменяется x_i на пяти уровнях, а для всех остальных факторов фиксируются значения центральной точки плана $(x_{i+1}^0; i+1 \neq i)$.

В-третьих, проводятся эксперименты для каждого из уровней фактора.

В-четвертых, из всех 5 уровней фактора x_i выбирают то значение (например, x_{ri}), которое обеспечивает максимум $W(z)$. Для фактора x_i фиксируют величину его уровня значением x_{ri} . Далее, переходят к варьированию уже фактора x_j (i -й фактор зафиксирован на уровне x_{ri}). Для всех остальных x_j уровень остается фиксированным в центре области пространства параметров. Далее, осуществив 5 вариаций фактора x_j , находят тот уровень (например, пусть это будет x_{1j}), который обеспечивает увеличение $W(z)$ и фиксируют это значение фактора x_j . Теперь уже два фактора зафиксированы соответственно на уровнях x_{zi} и x_{1j} , а остальные факторы находятся на уровнях $\{x_k^0\}$, ($k \neq i; k \neq j$). Эту процедуру повторяют до тех пор, пока не будут модифицированы уровни всех факторов. Если позволяет ресурс исследований, то новая

центральная точка уже будет находиться на модифицированных уровнях факторов и весь процесс модификации повторяется до тех пор, пока у исследователя не закончится ресурс опытов.

Во многих практических задачах влияние некоторых величин на отклик объекта исследования оценить нельзя. Однако исследователя может интересовать вопрос, насколько существенно влияние того или иного фактора (или их комбинации) на обобщенный отклик $W(z)$ модели СС. Кроме того, исследователь часто заинтересован в том, чтобы ослабить влияние вторичных, наблюдаемых и управляемых факторов. Сделать это можно путем специального планирования и привлечения для обработки (в случае качественных факторов) методов дисперсионного анализа. План типа «классический латинский квадрат» позволяет исследовать влияние трех факторов, варьируемых на n уровнях. Размерность латинского квадрата определяется числом уровней варьирования факторов. Например, пусть $n = 4$, тогда размерность латинского квадрата равна 4×4 . В таблице 4.2 приведен план типа латинского квадрата. Уровни двух факторов представлены в виде строк (фактор 1) и столбцов (фактор 2), а фактор 3, уровни которого обозначены буквами латинского алфавита (А, В, С, Д), заносятся в ячейки квадрата 4×4 . В первую строку (первый уровень фактора 1) в упорядоченной последовательности заносятся буквенные обозначения уровней фактора 3. Порядок расположения уровней фактора 3 в последующих строках определяется путем циклического сдвига элементов предыдущей строки вправо, при этом (буквы) расположены таким образом, что в каждой строке и в каждом столбце любая буква встречается только один раз. Отметим, что этот эксперимент состоит из 16 опытов, тогда как соответствующий ПФЭ 4^3 потребовал бы 64 опытов. Более подробно с методиками построения комбинированных планов в задачах многофакторного эксперимента можно познакомиться в работе [40].

Таблица 4.2 – Пример плана типа «Латинский квадрат»

Фактор 1	Фактор 2			
	1	2	3	4
1	А	В	С	Д
2	Д	А	В	С
3	С	Д	А	В
4	В	С	Д	А

4.6. Планирование имитационного эксперимента

При ограниченном ресурсе времени моделирования необходимо так спланировать эксперименты, чтобы выборка статистик, необходимых для определения откликов, была достаточной для вычисления зависимости откликов $W(X)$ от вектора параметров моделирования X . Поскольку разбросы выборочных значений случайны, то обусловленная ими неточность результатов ИЭ определяется размером выборки значений откликов. На практике размер выборки обычно является функцией ресурсов, отпущенных на ИЭ. При этом предполагается независимость и нормальность распределений откликов Y_j , представляющих собой сумму большого числа небольших эффектов. Переменная отклика является результатом действия всех этих эффектов. Поэтому необходимо спланировать количество опытов, которые осуществляются внутри модели по правилу «автоматической остановки», базирующихся на методе доверительных интервалов. Этот метод предполагает задание точности представления d_n математического ожидания (MY_n) и b_n дисперсии (D_n) у n -й компоненты вектора отклика Y_n и уровня значимости λ , гарантирующих попадание MY_n и D_n внутрь интервалов $(\bar{Y}_n \pm d_n)$, $(S_n^2 \pm b_n)$ с вероятностью $(1 - \lambda)$. Здесь \bar{Y}_n и S_n^2 представляют собой среднее значение и выборочную дисперсию, вычисленных по выборке объема N и являющимися оценками MY_n и D_n .

В ходе испытания и исследования свойств ИМ исследователь определяет векторы точностей (d_1, \dots, d_2) или (b_1, \dots, b_2) представления компонент вектора отклика Y . Выполнение правила «автоматической остановки» представляет собой итерационную процедуру, суть которой состоит в следующем. До начала проведения серии ИЭ известны уровень значимости λ и векторы точностей представления компонент вектора отклика b_n или d_n . Из априорных сведений (или опыта) устанавливается количество начальных экспериментов N_1 , необходимых для получения выборок значений компонент вектора откликов модели $\{Y_{nk}\}$, $n = \overline{1, L}$, $k = \overline{1, N_1}$. Если исследователь предполагает, что в ходе исследования свойств ИМ число опытов не будет большим, то в качестве начального значения он полагает $N_1 = 5$. Далее алгоритм выбора числа экспериментов состоит из следующих шагов.

Шаг 1. По выборке $\{Y_{nk}\}$ находит среднее значение $\{\bar{Y}_n\}$ и выборочную дисперсию $\{S_n^2\}$.

Шаг 2. Для очередного номера n компоненты вектора откликов модели определяют достаточную точность оценки \bar{Y}_n и S_n^2 при выполнении N_1 экспериментов. Здесь возможны следующие случаи.

Если выборка малого объема ($N_1 \leq 30$), то для вычисления доверительного интервала используют t -статистику, имеющую распределение Стьюдента:

$$d_{1n} = t_{kp} \sqrt{S_n^2 / (N_1 - 1)}, \quad (4.24)$$

где t_{kp} – критическое значение;

t – статистики при $(N_1 - 1)$ степенях свободы и заданном уровне значимости α .

Если размер выборки большой ($N_1 > 30$) и можно предположить нормальность Y_n , то для вычисления доверительного интервала MY_n используют двухстороннюю статистику нормированного распределения

$$d_{1n} = Z_{\alpha/2} \sqrt{S_n^2 / N_1}, \quad (4.25)$$

где $Z_{\alpha/2}$ – значение нормированного нормального распределения на уровне значимости $\alpha/2$.

Если нормальность Y_n предположить нельзя, но N_1 большое, то применяют неравенство Чебышева:

$$P\left(\left|\bar{Y}_n - MY_n\right| \geq h \cdot S_n \sqrt{N_1} \leq \frac{1}{h^2}\right), \quad (4.26)$$

где h – некоторая наперед заданная константа, означающая число среднеквадратичных отклонений, удовлетворяющих исследователя.

В этом случае, с достаточной для практики точностью, доверительной интервал отклика можно вычислить по формуле:

$$d_{1n} = \sqrt{S_n^2 / N_1 (1 - \alpha)} \quad (4.27)$$

при оценивании дисперсии откликов формула отыскания оценки D_n с достоверностью $(1 - \alpha)$ имеет вид:

$$P\{(1-b_n)S_n^2 \leq D_n \leq (1+b_n)S_n^2\} \leq 1-\alpha. \quad (4.28)$$

Поскольку N_1 достаточно велико, то эту статистику аппроксимируют нормальным распределением и получают оценку дисперсии:

$$b_{1n} = Z_{\alpha/2} \sqrt{2(N_1 - 1)}, \quad (4.29)$$

где $Z_{\alpha/2}$ – значение нормированного нормального распределения при заданном уровне значимости $\alpha/2$.

Шаг 3. Сравнивают достигнутую точность d_{1n} или b_{1n} оценок MY_n и D_n в N_1 опытах с заданными значениями d_n или b_n . Если выполняется неравенство $d_{1n} \leq d_n$ или $b_{1n} \leq b_n$, то это означает, что по n -й компоненте вектора откликов Y достигнута требуемая точность оценки в N_1 экспериментах, и переходят к шагу 4.

Шаг 4. Выполняют еще один модельный эксперимент, увеличив N_1 на единицу ($N_2 = N_1 + 1$), и переходят к шагу 1.

Шаг 5. Проверяют, все ли компоненты вектора откликов проверены на удовлетворение точности оценки MY_n и D_n . Если точность по всем (d_1, \dots, d_n) достигнута, то это означает завершение ИЭ. В противном случае меняют компоненты вектора откликов модели и переходят к шагу 2.

Достижимость заданной точности d_n или b_n оценки некоторой статистики (или группы статистик) может быть одним из условий окончания ИЭ. Необходимо при этом считать, что для достижения требуемой точности математического ожидания $\{MY_n\}$ требуется N_2 , равное максимальному значению для оценки всех компонентов.

Для достижения нужной точности $\{D_n\}$ требуется N_2 , равное максимальному значению для оценки всех компонентов. Кроме того, поскольку имитация в модели СС осуществляется с помощью процедуры Монте-Карло, то требуется повторение опыта N_2 раз с одними и теми же значениями параметров модели (но с разными начальными величинами базового генератора $\{\xi_{0k}\}$; k – номер места, использующего базовый генератор случайных чисел).

Итак, объем выборки $\{Y_{nl}\}$ в одной реализации для достижения требуемой точности вектора (4.23), равен $\max(N_1, N_2)$. Потом этот эксперимент в следующей реализации ИМ должен быть повторен N_2 раз согласно процедуре Монте-Карло.

В любом ИЭ важно определить интервалы изменения компонент вектора параметров модели X . До начала имитации необходимо найти

предельные значения статистик моделирования. Существуют три фактора, влияющие на выбор интервала изменения параметров модели: необходимость получения одинаковой относительной точности статистик на разных участках области изменения параметров ИМ; характер функции отклика; назначение ИЭ, определяющее способ планирования эксперимента. Если анализ погрешностей показывает, что на каком-либо участке моделируемого процесса статистик имеются сомнения, то в ходе следующей имитации на этом участке необходимо спланировать поиск более точной статистики. Обычно стремятся к тому, чтобы точность функции отклика модели на всех ее участках была одинакова. Если известно, что в ИЭ имеются особенности, которые можно обнаружить при получении данных в регулярной последовательности, используют классический последовательный план. В этом случае все параметры ИМ полагают постоянными в центральной области плана, а один из них меняют во всей области его значений. Выбор интервалов между соседними значениями параметра осуществляется с учетом баланса точности. Аналогичным образом планируют изменения второго параметра, третьего параметра и т. д. Затем составляют план изменения параметров при переходе от одного имитационного опыта к другому по методике, которая ранее была рассмотрена в 4.3–4.5.

Ранее были рассмотрены случаи, когда в качестве рабочей точки серии ИЭ выбирались уровни факторов, $\{x_i^0\}$, имевшие место во время мониторинга реальной СС.

Рассмотрим способы задания $\{x_i^0\}$ на основе априорной информации о моделируемой СС. Наиболее распространен способ задания основного уровня имитации на основе анализа функции отклика $Y = \varphi(X, G)$. Если у исследователя имеются сведения о координатах только одной точки на функции отклика и нет информации о границах изменения факторов, то ему ничего не остается, как рассматривать эту точку в качестве основного уровня плана ИЭ. Когда границы изменения компонент вектора факторов известны исследователю и он знает, что наилучшие значения X находятся внутри области изменения фактора, то в качестве основного уровня он может выбрать любую из точек факторного пространства. Хуже обстоит дело, когда известная точка лежит на границе факторного пространства. В этом случае исследователю приходится выбирать уровень со сдвигом внутрь области. Когда исследователю известно, что имеется много наилучших значений функции отклика (мультимодальная зависимость),

то в качестве основного уровня он может выбирать любую случайную точку внутри факторного пространства. Наконец, когда исследователю (до постановки ИЭ) известна подобласть в факторном пространстве, где исследуемый процесс протекает оптимально и эксперименты ведутся только для уточнения места этого оптимума, тогда в качестве основного уровня он выбирает центр этой подобласти.

Назначение ИЭ существенно влияет на выбор интервалов изменения параметров модели. Если предметом имитации является поиск узких мест в функционировании СС или выбор гипотез о механизме явлений, то применяется многоуровневое факторное планирование. В этом случае на выбор уровней факторов влияет относительная точность фиксации статистик имитации ST_{ml} на разных участках изменения параметров ИМ. Когда предметом моделирования является поиск оптимальных условий функционирования СС, либо выбор существенных для управления СС параметров, либо нахождение вида функции, аппроксимирующей поведение откликов СС, то задача сводится к выбору для каждого фактора его уровней изменения. Обычно на интервал варьирования факторов накладываются ограничения сверху и снизу. Интервал варьирования не может быть меньше той погрешности, с которой исследователь фиксирует уровни, чтобы они были различимы. Но интервал варьирования не может быть настолько большим, чтобы эти уровни оказались за пределами изменения факторов. Для выбора интервала варьирования исследователь обычно использует следующую априорную информацию: сведения о точности фиксации факторов; данные о кривизне функции откликов (линейная или нелинейная); результаты оценки чувствительности откликов $\{Y_n\}$ к изменениям параметров модели $\{x_i\}$. Поэтому часто проводятся серии «затравочных» ИЭ для знакомства с исследуемой СС и для получения недостающей априорной информации о влиянии выбранных интервалов варьирования факторов на вид искомой функции откликов.

При составлении плана анализа результатов ИЭ необходимо предусмотреть несколько проверок точности и приемлемости результатов имитации. В ряде случаев для определения источников погрешностей можно использовать *уравнения баланса точности*. Рассмотрим суть технологии составления этого уравнения на следующем примере. Пусть в ходе ИЭ на модели измеряются переменные A, B, C, D , которые связаны между собой уравнением сохранения значений между парами переменных $A \cdot B = C \cdot D$. Установлено, что из-за неточности описания реальной СС одна из этих переменных (не известно, какая из них) является источником систематической погрешности. Для коррекции алгоритма ИМ нужно выявить не только виновника

погрешности, но и место ее возникновения в алгоритме модели. Пусть в ходе имитации не меняется значение переменной A и остальные переменные изменяются следующим образом: $C \rightarrow m \cdot C$, $D \rightarrow n \cdot D$, $B \rightarrow mnB$. Уравнение баланса при этих условиях имеет вид: $A \cdot mnB = mC \cdot nD$. Далее используется следующее правило: если одна из переменных в этом уравнении баланса, например, может быть представлена в виде суммы $(A + \varphi(A))$, где $\varphi(A)$ – систематическая погрешность, то эту переменную можно обнаружить, рассматривая поочередно случаи с фиксированным значением каждой переменной. Переменная, относительная погрешность которой при фиксированном ее значении не меняется, и считается причиной систематической погрешности. Исключением является случай, когда переменная имеет пропорциональное приращение ошибки $A + k \cdot A$ (k – постоянная величина). В этом случае погрешность предлагаемым способом обнаружить невозможно.

Часто для поиска погрешности ИЭ используется предварительный эксперимент, по результатам которого исследователь строит график зависимости $Y = f(X)$. Система координат и сама функция f выбираются такими, чтобы график был линейным или хотя бы не имел большой кривизны вблизи начала координат для последующей экстраполяции графика. Обычно используются линейные или полупологарифмические шкалы, позволяющие строить прямую зависимость отклика Y от вектора параметров X . Например, пусть исследователю известно, что один из графиков линейной зависимости отклика имеет систематическую погрешность. Необходимо определить, какой из графиков верен, и оценить при этом эту погрешность. Пусть для простоты каждый набор результатов ИЭ характеризуется одинаковым показателем степени, хотя сами графики не совпадают. Однако известно, что в области малых значений зависимость $Y = f(X)$ должна проходить через начало координат. Тогда для идентификации и оценки систематической погрешности ИЭ строят оба графика в области малых значений. Тот график, который не удовлетворяет условиям прохождения через начало координат, имеет систематическую погрешность.

Отметим, что среднее значение отклик \bar{Y}_n при $X = 0$ и является оценкой этой погрешности.

5. ОСНОВЫ ОПТИМИЗАЦИИ ВАРИАНТОВ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

5.1. Методы векторной оптимизации сложных систем

При решении большинства задач проектного моделирования СС возникает необходимость их оптимизации по совокупности откликов вероятностных и имитационных моделей, оценивая при этом по различным критериям эффективность вариантов их функционирования. Такая оптимизация получила название векторной или многокритериальной. Ее особенностью является наличие не одного решения, а целого множества решений, каждое из которых может быть выбрано в качестве оптимального. В общем виде задачи векторной оптимизации СС могут быть записаны следующим образом:

$$\begin{aligned} Y_i = f_i(x) &\rightarrow \max; \quad i = \overline{1, k}; \\ Y_j = f_j(x) &\rightarrow \min; \quad j = \overline{k+1, m}, \quad x \in G. \end{aligned} \quad (5.1)$$

Понятие *оптимального* решения заменяется понятием *эффективного* решения. При этом чаще всего используются следующие методы многокритериальной оптимизации.

5.1.1. Выделение главного критерия

Выделяется из (5.1) главный критерий. Например, пусть $Y_1 = f_1(x) \rightarrow \max$ и есть главный показатель оптимальности СС. Преобразуем задачи (5.1) следующим образом:

$$\begin{aligned} Y_1 = f_1(x) &\rightarrow \max; \quad \text{при ограничениях:} \\ Y_i = f_i(x) &\geq \hat{Y}_i^*; \quad i = \overline{2, k}; \\ Y_j = f_j(x) &\leq \hat{Y}_j^*; \quad j = \overline{k+1, m} \quad x \in G. \end{aligned} \quad (5.2)$$

Как видим, для остальных критериев качества вариантов СС стремятся, чтобы они были либо больше, либо меньше некоторого граничного значения (соответственно \hat{Y}_i^* и \hat{Y}_j^*). Часть контроля каче-

ства вариантов СС осуществляется по главному критерию, но при условии, что по всем остальным критериям качества отклика СС находится в допустимых пределах.

5.1.2. Метод последовательных уступок

Все отклики вариантов СС можно расположить в порядке уменьшения их важности. Например, пусть важность откликов расположена следующим образом: Y_1, Y_2, \dots, Y_n , и пусть для всех Y_i необходимо добиться их максимизации из возможных вариантов. Алгоритм метода сводится к следующему. Вначале находится решение, обращающее в максимум главный критерий $\hat{Y}_1 = f_1^*(x) \rightarrow \max$. Затем из априорных соображений назначается некоторая «уступка» Δf_1 , требующая выполнения неравенства:

$$f_1(x_1) \geq f_1^*(x) - \Delta f_1, \quad (5.3)$$

где $f_1^*(x) = \max f_1(x)$, находим такое решение x_i , при котором $f_2(x) \rightarrow \max$.

Далее снова назначается «уступка» по критерию $Y_2(x) = f_2(x)$ (например, пусть это будет Δf_2), с помощью которой можно максимизировать $f_2^*(x) = \max f_2(x)$. Этот процесс продолжается до тех пор, пока по всем откликам не будет достигнута некоторая близость к максимальным значениям с учетом сделанных уступок:

$$\Delta f_i; \quad i = \overline{1, n}. \quad (5.4)$$

5.1.3. Методы приведения откликов к одному типу и масштабу

В ряде случаев исследователю удобно, чтобы все отклики были однотипны (требовали либо максимизации, либо минимизации). На практике очень часты случаи, когда какой-либо отклик, например $Y_2(x)$, требует минимизации ($Y_2(x) \rightarrow \min$), а все остальные отклики требуют максимизации ($Y_i(x) \rightarrow \max; i \neq 2; i = 1, 3, \dots, n$). В таких случаях для откликов, требующих минимизации, находят обратную величину:

$$Y_2^*(x) = 1/Y_2(x). \quad (5.5)$$

Однако этой операции может оказаться недостаточно, поскольку каждый отклик имеет различный смысл и его значение определяется с различной размерностью (килограммы, метры, секунды, миллиметры, метры). Поэтому отклики трудно сравнивать друг с другом, даже если они одного типа. В таких случаях используется операция «нормировка». Находят для каждого отклика максимальное значение:

$$Y_{mi} = \max_i Y_i \quad \text{или} \quad Y_{mj}^* = \max_j Y_j^*; \quad i, j = \overline{1, m}. \quad (5.6)$$

Проводится нормировка каждого отклика:

$$\begin{aligned} Y_i^{**} &= Y_i / Y_{mi} && \text{— для откликов требующих максимизации;} \\ Y_j^{**} &= Y_j^* / Y_{mj}^* && \text{— для откликов, полученных с помощью} \end{aligned} \quad (5.7)$$

операции приведения к одному типу.

В итоге все компоненты вектора параметров $\{Y_i^{**}\}; i = \overline{1, m}$ будут иметь одинаковый тип (в данном случае требовать максимизации) и изменяться в одном и том же диапазоне их значений $[0, 1]$ (нормированных максимальным значением). Поэтому их можно сравнивать.

5.1.4. Метод составного критерия

Когда все отклики имеют один и тот же масштаб изменения и они все одного типа, исследователь может формировать составной отклик $Y_0(x)$ согласно следующей процедуре. Во-первых, исследователь сам определяет важность каждого критерия с помощью весовых коэффициентов важности (δ_i). Предполагаются, что $\delta_i \geq 0, i = \overline{1, n}$, а сумма коэффициентов важности равна единице:

$$\delta_i \geq 0; \quad \sum_{i=1}^n \delta_i = 1. \quad (5.8)$$

Определяется составной (обобщенный) отклик варианта СС:

$$Y_0(x) = \sum_{i=1}^n \delta_i Y_i(x). \quad (5.9)$$

Если $Y_i(x) \rightarrow \max$, то и $Y_0(x) \rightarrow \max$.

Несмотря на удобную форму записи, обобщенные критерии качества вариантов СС имеют существенные недостатки, связанные с субъективностью выбора весовых коэффициентов важности откликов (δ_i), а также с тем фактом, что недостатки эффективности вариантов СС по одним критериям могут компенсироваться за счет преимуществ по другим критериям.

5.1.5. Нормативные методы

Нормативные методы являются обобщением рассмотренных выше методов и состоят в предварительном получении нормативов $nr f_i$, $i = \overline{1, n}$ на основе приближенного решения многоцелевой задачи и приближения к этим нормативам по некоторой заранее заданной метрике $\rho(f(x), nr f) \rightarrow \min$, где весовые коэффициенты определяются различными способами.

Например, в качестве метрики выбираются:

$$\rho_1^2(f(x), nr f) = \sum_{i=1}^n [f_i(x) - nr f_i]^2; \quad (5.10)$$

$$\rho_2(f(x), nr f) = \sum_{i=1}^n |f_i(x) - nr f_i|; \quad (5.11)$$

$$\rho_3(f(x), nr f) = \max_i |f_i(x) - nr f_i|; \quad (5.12)$$

Выбор метрики исследователь осуществляет из априорной информации об исследуемой СС.

5.1.6. Методы логического объединения критериев

Используются только в тех случаях, когда отклики $Y_i = f_i(x)$, $i = \overline{1, n}$, могут принимать только два значения 0 или 1 (т. е. являются булевыми переменными):

$$Y_i(x) = f_i(x) = \begin{cases} 1, & \text{если } i\text{-я цель исследования достигнута,} \\ 0, & \text{если } i\text{-я цель исследования не достигнута.} \end{cases} \quad (5.13)$$

В таких случаях используются два вида обобщенных откликов моделирования:

– в виде конъюнкции откликов $Y_i(x)$, если общая цель состоит в достижении всех целей одновременно:

$$Y_{01}(x) = \prod_{i=1}^n f_i(x) = \prod_{i=1}^n Y_i(x); \quad (5.14)$$

– в виде дизъюнкции откликов $Y_i(x)$, если общая цель состоит в достижении хотя бы одной из целей:

$$Y_{02}(x) = 1 - \prod_{i=1}^n f_i(x) = 1 - \prod_{i=1}^n Y_i(x); \quad (5.15)$$

Во всех методах параметр X выступает в качестве обобщенной характеристики. На практике X может, в свою очередь, быть многомерной величиной. К ней также могут быть применены способы отображения ее в виде скалярной величины, путем приведения к одному типу и нормировки значений ее компонентов, приводящей к тому, что у X все компоненты изменяется на одном и том же интервале значений $[0, 1]$, и в этом случае используют составной параметр X_0 . Способы такой модификации параметра X к обобщенной величине можно найти в работах [36, 56]. Оставив в стороне методику приведения многомерного вектора параметров $\{X_j\} \dots, j = \overline{1, m}$, к обобщенному скаляру, вначале рассмотрим методики оптимизации целевой функции моделирования от одной переменной X .

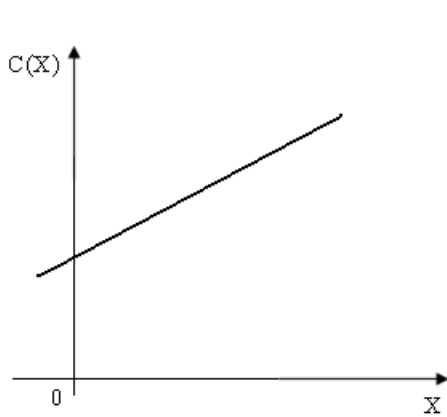
5.2. Оптимизация нелинейной функции отклика от одной переменной

Рассмотрим задачу максимизации различных видов целевых функций. В случае максимизации линейной целевой функции (ЦФ) одной переменной имеем:

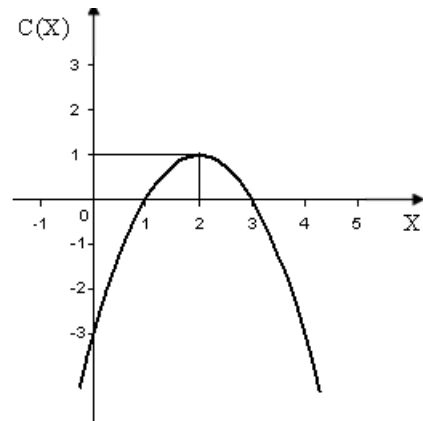
$$C(x) \equiv C_0 + C_1x,$$

график её приведен на рисунке 5.1, а). Если на x не налагается никаких ограничений, то при любом $C_1 \neq 0$ значение ЦФ можно сделать сколь угодно большим. Если же $a_1 \leq x \leq a_2$, то $x = a_1$ оптимально при $c_1 \leq 0$, а при $c_1 \geq 0$ оптимальным будет $x = a_2$. Следовательно, оптимальным

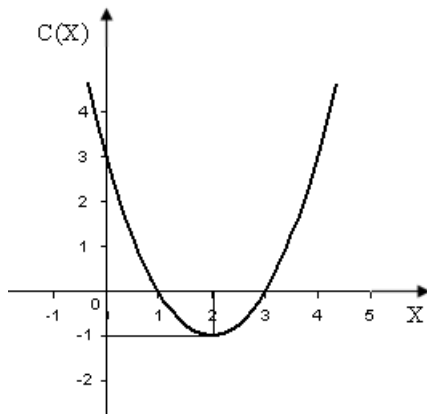
является решение, соответствующее экстремальной точке. Однако, если функция $C(x)$ не линейна, то возможны другие ситуации, изображенные на рисунке 5.1.



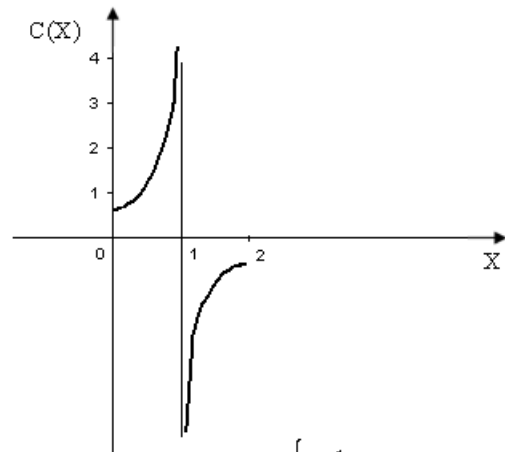
Пример 1: $C(X) = C_0 + C_1 X$



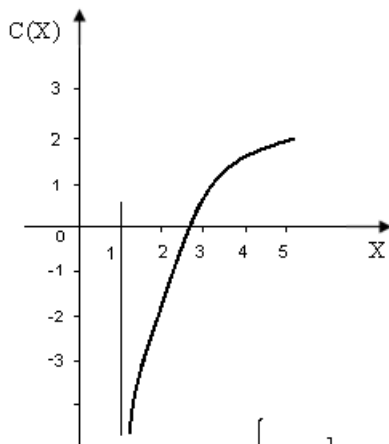
Пример 2: $C(X) = -X^2 + 4X - 3$;
 $\max C(X) = C(2) = 1$



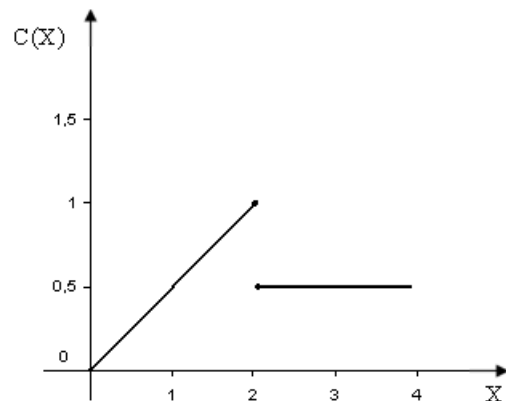
Пример 3: $C(X) = X^2 - 4X + 3$



Пример 4: $C(X) = \begin{cases} \frac{1}{1-X}; & 0 < X < 1 \\ 0; & X = 1 \end{cases}$



Пример 5: $C(X) = \begin{cases} 1 - \frac{1}{X-1}; & X > 1 \\ 0; & X \leq 1 \end{cases}$



Пример 6: $C(X) = \begin{cases} X; & 0 \leq X < 1 \\ \frac{1}{2}; & X \geq 1 \end{cases}$

Рисунок 5.1 – Примеры нелинейных функций отклика

Как видно из рисунка 5.1, функция $C(x)$ может неограниченно возрастать, даже если значение x ограничено. В других случаях, когда функция $C(x)$, ограниченная сверху, она не обязательно достигает максимума, причем вне зависимости от того, ограничены ли значения x . Предположим, что:

1) значения x лежат в пределах конечного сегмента $I \equiv [a_1, a_2]$, т. е. $a_1 \leq x \leq a_2$; где $-\infty < a_1 < a_2 < +\infty$;

2) функция $C(x)$ непрерывна и ограничена сверху для всех x на сегменте.

Из этих допущений следует, что $C(x)$ достигает максимума при значении x , находящемся на сегменте I . Рассмотрим случай, когда функция $C(x)$ является монотонной на сегменте I и всегда достигается максимум x на I в одной из экстремальных точек сегмента. Если $C(x)$ является выпуклой функцией x на I , т. е. для любых x_1 и x_2 на I , где $x_1 < x_2$ и для любого $p (0 \leq p \leq 1)$ функция $C(x)$ удовлетворяет условию:

$$pC(x_1) + (1 - p)C(x_2) \geq C[px_1 + (1 - p)x_2] - \text{выпуклая.} \quad (5.16)$$

Рассмотрим функцию $C(x)$, график которой изображен на рисунке 5.2. Максимальное значение $C(x)$ достигается при $x = f$. Предположим, что проверка значения $C(x)$ начинается со значения $x = a$. Если первое проверяемое значение x меньше a , тогда каждое проверяемое значение x все больше и больше будет приближаться к a , но не будет превышать a , поскольку $C(a)$ есть *локальный максимум*.

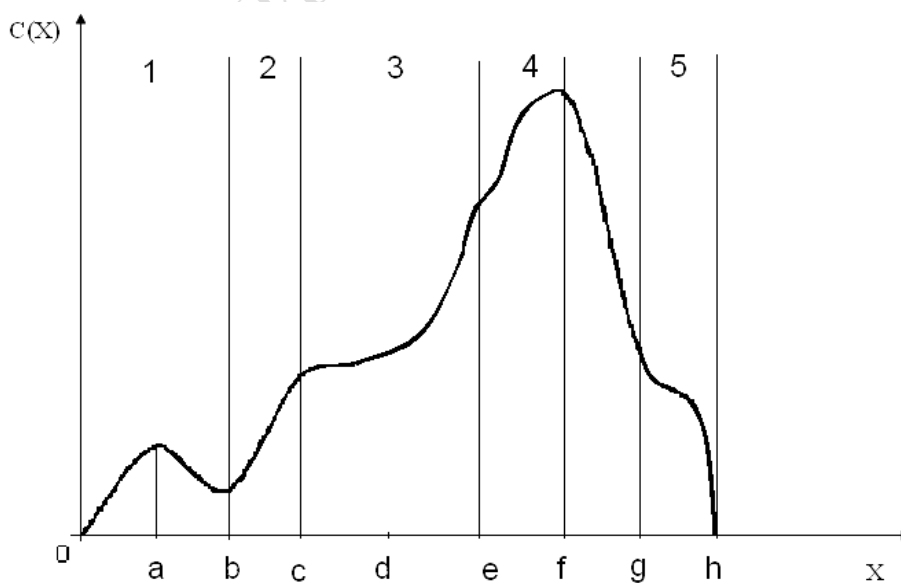


Рисунок 5.2 – Пример функции отклика, обладающей многими локальными оптимумами

Далее предположим, что проверяемое значение x находится на сегменте $[d, e]$. Тогда, по-видимому, последовательный поиск закончится сразу же, поскольку как при небольшом увеличении, так и при небольшом уменьшении значения x функция $C(x)$ не меняет своей величины и ее улучшение не представляется возможным. Как видно из рисунке 5.2, рассмотрение функции $C(x)$ на сегменте $[0, g]$ является более общей ситуацией, чем это действительно необходимо для достижения задачи максимизации. График $C(x)$ можно разделить на 5 различных участков, где функция является одномеральной. При этом на каждом j участке ($j = \overline{1,5}$) модами являются соответственно: $\text{mod}_1 = C(a)$; $\text{mod}_2 = C(c)$; $\text{mod}_3 = C(d)$; $\text{mod}_4 = C(f)$; $\text{mod}_5 = C(d)$.

Экстремумом функции $C(x)$ будет мода mod_4 . На практике исследователи стараются поделить область изменения функции на ряд унимодальных участков, а затем из значений мод выбирают ту, которая является максимальной на всем участке определения x . В нашем примере экстремум $C(x)$ достигается при $x = f$ и равен он $\text{ekst}C = C[f]$. Обычно исследователи для поиска экстремума унимодальной функции от одной переменной пользуются методами *дихотомии* и *золотого сечения*. Рассмотрим алгоритмы реализации этих методов.

Метод дихотомии. Пусть целевая функция $f(x)$ имеет на отрезке $[a, b]$ минимум в точке x^* (рисунок 5.3). Вычисление абсциссы x^* точки минимума $f(x)$ фактически сводится к вычислению отрезка $[a_n, b_n]$, длина которого $|b_n - a_n| < \varepsilon$, ε – заданная точность вычисления и $x \in [a_n, b_n]$.

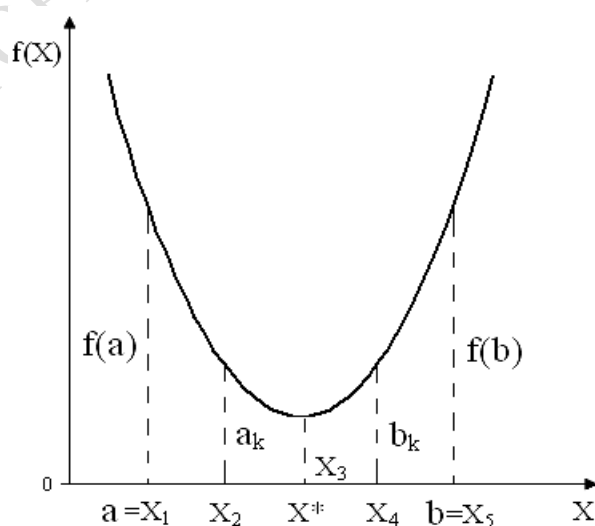


Рисунок 5.3 – Пример функции отклика с одним минимумом

Сущность метода дихотомии заключается в следующем:

1) полагаем $x_1 = a$ и $x_5 = b$. Делим отрезок $[a, b]$ точками x_2, x_3 и x_4 на четыре равные части;

2) вычисляем значения функции $f(x)$: $f(x_1), f(x_2), f(x_3), f(x_4), f(x_5)$;

3) среди чисел $f(x_i)$ находим минимальные. Пусть, например, $f(x_i) = \min \{ f(x_i), i = \overline{1,5} \}$. В силу унимодальности $f(x)$ на отрезке $[a, b]$ точка x^* находится на отрезке $[x_2, x_4]$, длина которого по построению вдвое меньше длины отрезка $[a, b]$.

Если все-таки $|x_{k+1} - x_k| > \varepsilon$, то полагаем $a = x_{k-1}, b = x_{k+1}$ и вновь выполняем все действия, начиная с 1). Если на n -й итерации мы получим, что локализованный отрезок $|x_{k+1} - x_k| \leq \varepsilon$, то в качестве значения x^* можно, например, принять $x^* = \frac{x_{k-1} + x_{k+1}}{2}$.

В качестве примера найдем методом дихотомии минимум функции $f(x) = \sin x + \sqrt{x}$ на отрезке $[3, 6]$. Точность ε принять равной 1.

График функции $f(x)$ имеет вид, представленный на рисунке 5.4, а.

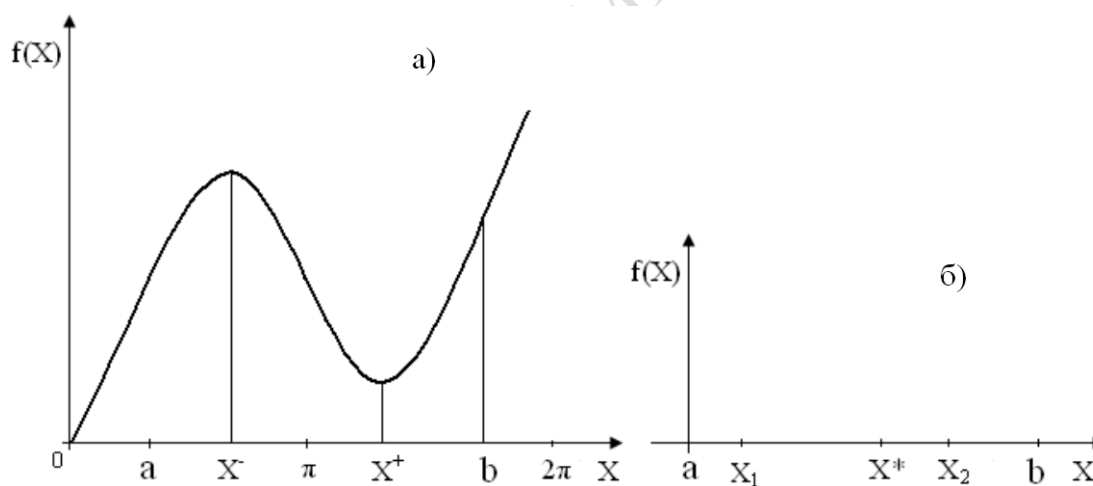


Рисунок 5.4 – Примеры: а) функции отклика с двумя экстремумами; б) выбора точек деления отрезка по методу золотого сечения

В соответствии с вышеизложенным алгоритмом выполняем указанные действия.

Первая итерация:

1) $x_1 = a = 3; x_5 = b = 6; x_2 = 3,75; x_3 = 4,5; x_4 = 5,25;$

2) вычисляем значения $f(x_i), i = \overline{1,5}$:

$f(x_1) = 1,87; f(x_2) = 1,36; f(x_3) = 1,14; f(x_4) = 1,43; f(x_5) = 2,17;$

3) выбираем $\min \{f(x_i), i = \overline{1,5}\} = f(x_3) = 1,14$. Следовательно, $x^* \in [x_2, x_4] = [3,75; 5,25]$. Длина этого отрезка равна $5,25 - 3,75 = 1,5$, что больше ε , и мы вынуждены повторить все действия, полагая теперь, что новый отрезок $[a, b]$, на котором находится x^* , равен $[3,75; 5,25]$.

Вторая итерация:

1) $x_1 = a = 3,75, \quad x_5 = b = 5,25, \quad x_2 = 4,12, \quad x_3 = 4,5, \quad x_4 = 4,88;$

2) $f(x_1) = 1,36, \quad f(x_5) = 1,43, \quad f(x_2) = 1,20, \quad f(x_3) = 1,14, \quad f(x_4) = 1,22;$

3) $\min \{f(x_i), i = \overline{1,5}\} = f(x_3) = 1,14$. Выбираем отрезок $[x_2, x_4] = [4,12; 4,88]$.

Длина этого отрезка равна $0,76$, т. е. $|x_4 - x_2| < \varepsilon$. Следовательно, можно сказать, что $x^* = \frac{4,88 + 4,12}{2} = 4,5$, и $f(x^*) \approx 1,1438$.

Безусловно, $\varepsilon = 1$ обеспечивает нам очень грубое решение задачи. На практике, как правило, задают ε малым, например, равным $0,000001$. Но наша цель – продемонстрировать практическое применение метода дихотомии для нахождения минимума целевой функции, достигнута.

Метод золотого сечения. Прежде всего напомним, что если d – длина отрезка, то золотое сечение отрезка определяется пропорцией $\frac{d}{x} = \frac{x}{d-x}$, т. е. x находится из квадратного уравнения $x^2 + dx - d^2 = 0$ и, следовательно, $x_1 \approx 0,38d$ и $x_2 \approx 0,62d$ (рисунок 5.4, б).

Пусть по-прежнему целевая функция $f(x)$ имеет на отрезке $[a, b]$ единственный минимум в точке x^* . Абсциссу точки x^* мы и стремимся найти с заданной точностью $\varepsilon = 1$. Метод золотого сечения предписывает выполнение следующих действий:

1) вычисляем $x_1 = a + 0,38(b - a)$ и $x_2 = a - 0,62(b - a)$;

2) вычисляем $f(x_1)$ и $f(x_2)$. При этом могут иметь место случаи:

а) $x_2 > x_1$ и $f(x_1) > f(x_2)$. В этом случае $x^* \in [x_1, b]$;

б) $x_2 > x_1$ и $f(x_1) < f(x_2)$. В этом случае $x^* \in [a, x_2]$;

в) $x_2 > x_1$ и $f(x_1) = f(x_2)$. В этом случае $x^* \in [x_1, x_2]$;

3) выбираем отрезок, которому принадлежит точка x^* . Назовем его $[a_1, b_1]$. Длина этого отрезка меньше длины исходного отрезка

$[a, b]$, но если только $|b_1 - a_1| \leq \varepsilon$, мы можем считать, что с точностью ε , $x^* = \frac{b_1 + a_1}{2}$.

В противном случае мы повторяем действия 1–3 до тех пор, пока на n -м шаге не будет локализован отрезок $[a_n, b_n]$, для которого будет выполнено требование $|b_n - a_n| \leq \varepsilon$.

Проиллюстрируем применение метода золотого сечения для нахождения минимума целевой функции из примера.

Найти методом золотого сечения минимум функции $f(x) = \sin x + \sqrt{x}$ на отрезке $[3; 6]$. Точность ε принять равной 1.

Первая итерация:

1) вычислим абсциссы точек золотого сечения $x_1 = 3 + 0,38(6 - 3) = 4,15$ и $x_2 = 6 - 0,38(6 - 3) = 4,86$;

2) вычислим $f(x_1) = 1,19$, $f(x_2) = 1,21$. Очевидно, мы имеем случай б), т. е. при $x_2 > x_1$ $f(x_1) < f(x_2)$. Следовательно $x^* \in [3; 4,86]$.

3) длина отрезка, содержащего x^* , равна 1,86, что больше ε , поэтому следует выполнить еще одну итерацию, принимая отрезок $[a, b_1] = [3; 4,86]$.

Вторая итерация:

1) $x_1 = 3 + 0,38 \cdot 1,86 = 3,7$, $x_2 = 4,86 - 0,38 \cdot 1,86 = 4,15$;

2) $f(x_1) = 1,39$, $f(x_2) = 1,19$. Поскольку $f(x_1) > f(x_2)$, то $x^* \in [3,7; 4,86]$. Длина этого отрезка равна $4,86 - 3,7 > \varepsilon$, поэтому выполним третью итерацию.

Третья итерация:

1) $x_1 = 3,7 + 0,38 \cdot 1,16 = 4,15$, $x_2 = 4,86 - 0,38 \cdot 1,16 = 4,42$;

2) $f(x_1) = 1,19$, $f(x_2) = 1,44$, $f(x_1) < f(x_2)$, следовательно, $x^* \in [4,15; 4,86]$. Так как длина последнего отрезка равна $0,71 < \varepsilon$, мы можем считать, что с точностью ε $x^* = \frac{4,15 + 4,86}{2} = 4,505$, а значит $f(x^*) = 1,1439$.

5.3. Максимизация функции отклика от многих переменных без ограничений

Вычислительный алгоритм максимизации функции отклика.

В тех случаях, когда вычислительный алгоритм максимизации функции отклика $C(x_i)$ зависит от значений компонентов вектора параметров (x_1, x_2, \dots, x_n) , исследователю приходится постулировать, что

функция $C(x_i)$ $i = \overline{1, m}$ достигает конечного максимума при конечных значениях $(\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n)$. Сокращенно обозначим: $\{x_i\}$ через X , $\{\hat{x}_i\}$ через \hat{X} , $C(x_i)$ через $C(X)$. Кроме того, исследователь должен предположить следующие допущения:

1. При всех значениях $\{x_i\}$ функция $C(X)$ однозначна и конечна;
2. При всех значениях X каждая частная производная $\partial C(X) / \partial X_j$ однозначна, конечна и непрерывна, так, что функция $C(X)$ также непрерывна;
3. Функция $C(X)$ обладает конечным максимумом \hat{C} ;
4. При возможных значениях $C(X)$, например C , существует такое конечное число M_c , что $\{x_j\} \leq M_c$, если $C(X) \geq C$.

Из теории дифференциального исчисления известно, что необходимым условием существования максимума является выполнение первых трех допущений. Это означает, что функция $C(X)$ имеет максимум в точке \hat{x} только в том случае, если $\partial C(\hat{X}) / \partial X_j = 0$ для всех $j = 1, 2, \dots, n$. Многие вычислительные методы максимизации $C(X)$ имеют следующую стандартную форму:

Шаг 1. Выбирается в качестве исходной произвольная пробная точка X^0 .

Шаг 2. Расчеты прекращаются, если в пробной точке X^k частные производные равны нулю ($\partial C(X) / \partial X_j = 0$; $j = 1, 2, \dots, n$), иначе определяется значение U_j^k для $j=1, \dots, n$ и переходят к шагу 3 (где k – номер пробной точки).

Шаг 3. Определяется новая пробная точка

$$X_j^{k+1} = X_j^{k+1} + X_j^k \quad (5.17)$$

и переходят к шагу 1.

Возвращаются к выполнению шага 2, заменив X^k на X^{k+1} . При этом определяется направление d_j^n для $j = 1, 2, \dots, n$ на основе информации о поведении $C(X)$ вблизи проверяемой точки X^n , т. е. на локальных свойствах изменения $C(X)$. Выбор длины шага t^n основан на информации о поведении $C(X)$ за пределами окрестности точки X^n при движении в выбранном направлении $d_1^k, d_2^k, \dots, d_n^k$. Наконец, направление и длина шага объединяются в величинах:

$$X_j^k = t^k d_j^h; \quad j = 1, 2, \dots, n. \quad (5.18)$$

Рассмотрим следующий пример: нужно максимизировать функцию

$$C(X_1, X_2) \equiv -(X_1 - 3)^2 - 4(X_2 - 2)^2. \quad (5.19)$$

Исследуя $C(X)$, можно установить, что единственным оптимальным решением является $\hat{X}_1 = 3$; $\hat{X}_2 = 2$, при которых $C(\hat{X}) = 0$. Поскольку $C(X)$ является квадратичной функцией, оптимальное значение X можно вычислить непосредственно из необходимых условий $\frac{\partial C}{\partial X_i} = 0$; $i = \overline{1, 2}$. На рисунке 5.5 приведен график линий уровня $C(X_1, X_2)$. Как видно из рисунка 5.5, линии уровня имеют вид эллипсов. Начало алгоритма осуществляется при $X_1^0 = X_2^0 = 0$ и при $C(X^0) = -25$. Уравнения этих эллипсов:

$$\frac{\partial C}{\partial X_1} = -2(X_1 - 3); \quad \frac{\partial C}{\partial X_2} = -8(X_2 - 2). \quad (5.20)$$

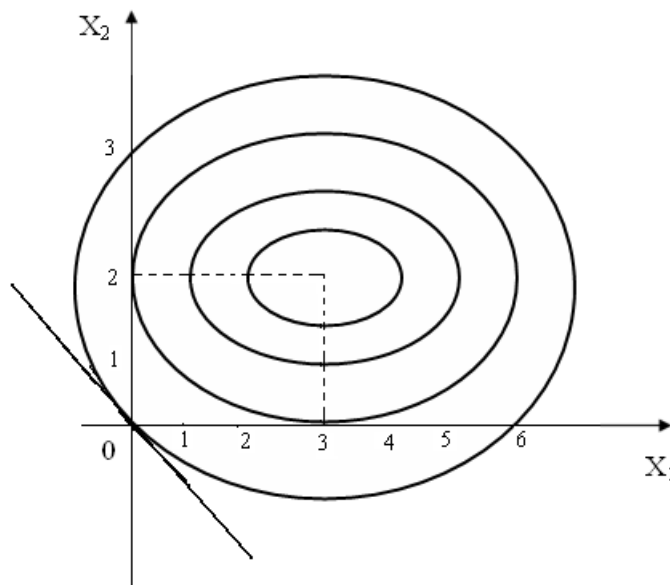


Рисунок 5.5 – Касательная к линиям уровня

Обе эти производные в точке X^0 строго положительны. Поэтому направление дальнейшего изменения на шаге 2 значений X_1 и X_2 приведено на рисунке 5.6. В процессе подготовки к выбору направления d_1 и d_2 проводится касательная к линиям уровня $C(X)$ в точке X^0 . В итоге имеет место система уравнений:

$$\frac{\partial C}{\partial X_1}(X_1 - X_1^0) + \frac{\partial C}{\partial X_2}(X_2 - X_2^0) = 0. \quad (5.21)$$

При $X_1^0 = X_2^0 = 0$ система уравнений (5.21) имеет решение:

$$6X_1 + 16X_2 = 0; \text{ или } X_2 = -\frac{3}{8}X_1. \quad (5.22)$$

Сама же касательная показана на рисунке 5.6. Направление d_1d_2 изображено в виде вектора, выходящего из исходной точки. Из рисунка 5.7 видно, что значение $C(X)$ на каждой итерации улучшается. При возрастании номера шага k пробные точки лежат на спирали, все больше и больше приближаясь к окружности с центром в начале координат и с единичным радиусом.

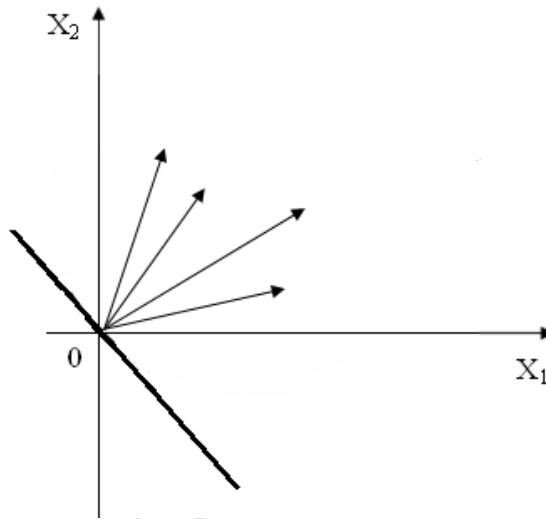


Рисунок 5.6 – Направления движения

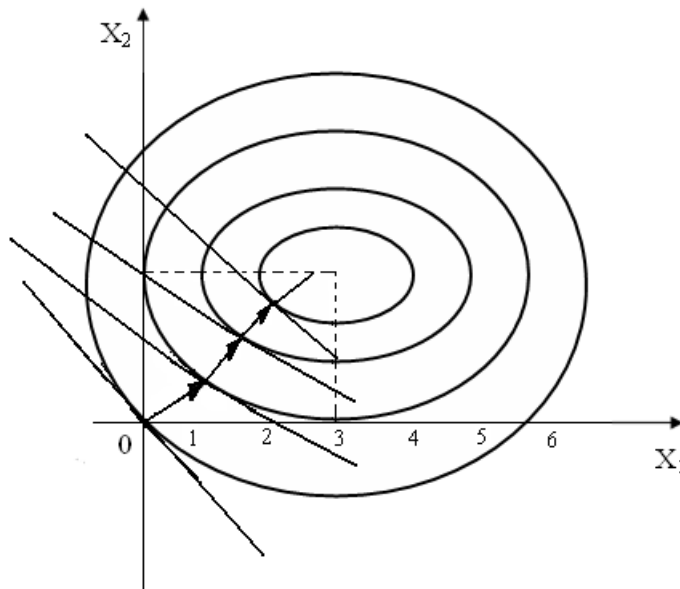


Рисунок 5.7 – Оптимальная длина шага

Метод скорейшего подъема. Позволяет учитывать информацию о виде целевой функции при движении от произвольной точки \hat{X} к значению \hat{X}_h , обеспечивая его движение за минимальное число итераций. Приняв в качестве $d_j^k = \frac{\partial C}{\partial X_j}$ в пробной точке X^k величину приращения, равную $t^k \geq 0$, можно максимизировать целевую функцию путем добавления приращений на каждом k -м шаге оптимизации:

$$C(X_1^k + t^k \cdot d_1^k, \dots, X_n^k + t^k \cdot d_n^k). \quad (5.23)$$

В методе применена идея использования в качестве направления набор частных производных, обычно именуемый градиентом функции $C(X)$ в точке X^k , обозначаемый символом $\nabla C(X)$:

$$\nabla C(x) = \text{grad } C(X) = \left(\frac{\partial C(X)}{\partial X_1}, \frac{\partial C(X)}{\partial X_2}, \dots, \frac{\partial C(X)}{\partial X_n} \right). \quad (5.24)$$

Вектор $[\text{grad } C(X)]$ называется антиградиентом функции $C(X)$. Он указывает направление наискорейшего убывания функции $C(X)$.

Для простоты рассмотрим функцию двух переменных $C(X_1, X_2)$. На рисунке 5.8 изображены две линии уровня:

$$C(X_1, X_2) = S; C(X_1, X_2) = S + \Delta S. \quad (5.25)$$

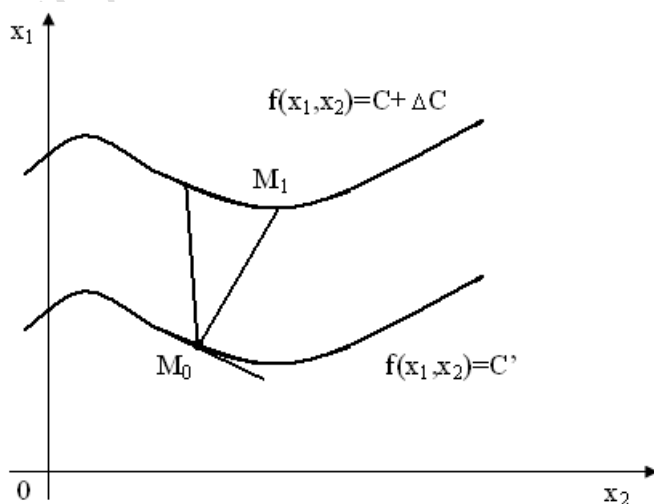


Рисунок 5.8 – Линии уровня функции двух переменных

Как известно, линии уровня функции $C(X)$ – это множество точек, в которых данная функция $C(X_1, X_2)$ принимает одно и то же фиксированное значение. Рассмотрим какую-либо точку $M_0(X_1^0, X_2^0)$ на линии $C(X_1, X_2) = S$ и точку $M_1(X_1^1, X_2^1)$ на линии $C(X_1, X_2) = S + \Delta S$. Приращение функции ΔC при переходе из точки M_0 в точку M_1 равно $\Delta C = \Delta S$, и это приращение не зависит от положения точки M_1 на линии $C(X_1, X_2) = S + \Delta S$. Отношение приращения функции к длине отрезка M_0M_1 назовем средней скоростью изменения функции $C(X_1, X_2)$ на отрезке M_0M_1 . Очевидно, что средняя скорость ее изменения $\Delta C / |M_0M_1|$ зависит от длины отрезка M_0M_1 или от положения точки M_1 на линии $C(X_1, X_2) = S + \Delta S$. При этом скорость $\Delta C / |M_0M_1|$ достигает наибольшего значения там, где длина отрезка M_0M_1 будет наименьшей. Известно, что длина отрезка наименьшая в том случае, когда направление отрезка M_0M_1 перпендикулярно касательной линии в точке $C(X_1, X_2) = S$ в точке M_0 при $\Delta C \rightarrow 0$. Теперь, если рассматривать предел отношения $\lim \frac{\Delta C}{|M_0M_1|}$, то это означает определение про-

изводной. Итак, частные производные $\frac{\partial C(M_0)}{\partial X_1}$ и $\frac{\partial C(M_0)}{\partial X_2}$ определяют максимальные скорости изменения $C(X_1, X_2) = C$ в точке M_0 по координатным осям.

Как видим, идея градиентного метода реализуется следующей последовательностью шагов при поиске экстремума функции $C(X)$.

Шаг 1. Выбирается начальная точка M_0 , принадлежащая области определения функции $C(X)$.

Шаг 2. Находится направление градиента функции $C(X)$ в точке M_0 :

$$C(M_0) = \left(\frac{\partial C(M_0)}{\partial X_1}, \frac{\partial C(M_0)}{\partial X_2}, \dots, \frac{\partial C(M_0)}{\partial X_n} \right). \quad (5.26)$$

Шаг 3. Задаем приращение из точки M_0 на величину, равную α , в точку M_i . В результате получаем расчетные формулы:

$$X_i(M_1) = X_j(M_0) + \alpha \frac{\partial C(M_0)}{\partial X_j}; \quad j = \overline{1, n}. \quad (5.27)$$

Когда ищется минимум функции, необходимо перемещаться в направлении антиградиента и в этом случае $\alpha < 0$. Если же ищется максимум функции $C(X)$, то $\alpha > 0$. Признаком достижения минимума (максимума) функции служит обращение в нуль градиента $\text{grad}C = 0$. Но на практике, как правило, ограничиваются приближенным решением. Поэтому критерием достижения требуемого результата является выполнение условия

$$\sum_{j=1}^n \left(\frac{\partial C(X)}{\partial X_j} \right)^2 \leq \varepsilon, \quad (5.28)$$

где $\varepsilon > 0$ – наперед заданное число.

Существуют и другие критерии. Например, после каждой итерации можно сравнивать достигнутое значение целевой функции с ее значением в предыдущей точке. Если выполняется неравенство $|C(M_n) - C(M_{n-1})| \leq \varepsilon$, то поиск экстремума функции $C(X)$ прекращается. На рисунках 5.5–5.7 приведен пример поиска экстремума методом скорейшего подъема.

В качестве примера рассмотрим поиск экстремума целевой функции закупки сырья двух видов фабрикой. Пусть фабрика закупает сырье в количестве соответственно X_1 и X_2 в тоннах. Расходы на хранение сырья на складе определяются по функциям:

$C(X_1, X_2) = X_1^2 - 7X_1 + X_2^2 - 4X_2 - X_1X_2 + 35$ (тыс. руб.). Необходимо определить объемы сырья (X_1^* и X_2^*), расходы на хранение которых будут минимальны. Примем точность вычислений, равную $\varepsilon = 1$. В качестве начальной M_0 выберем точку с координатами $X_1 = 1$; $X_2 = 1$, причем шаг установим, равным $\alpha = 0,5$. Расходы на хранение сырья в количестве $X_1 = 1$; $X_2 = 1$ равны $C(M_0) = 25$. Вычислим значения компонент градиента. Поскольку

$$\frac{\partial C(M_0)}{\partial X_1} = -2X_1 - X_2 - 2; \quad \frac{\partial C(M_0)}{\partial X_2} = 2X_2 - X_1 - 4, \quad \text{то} \quad \frac{\partial C'(M_0)}{\partial X_1} = -6;$$

$$\frac{\partial C'(M_0)}{\partial X_2} = -3;$$

Переходим к точке $M_1 (X_1^1, X_2^1)$. Уравнение параметрических прямых имеет вид:

$$X_1^1 = 1 - 0,5(-0,6) = 4; \quad X_2^1 = 1 - 0,5(-0,3) = 2,5.$$

Откуда $\frac{\partial C(M_1)}{\partial X_1} = -1,5; \quad \frac{\partial f(M_1)}{\partial X_2} = -3; \quad C(M_1) = 9,25.$

Разность $|C(M_1) - C(M_0)| = 15,75 > \varepsilon$, поэтому выполняем вторую итерацию, вычисляем значения целевой функции в точке $M_1(X_1^2, X_2^2)$:

$$X_1^2 = 4 - 0,5(-1,5) = 4,75; \quad X_2^2 = 2,5 - 0,5(-0,3) = 4;$$

$$\frac{\partial C(M_2)}{\partial X_1} = -1,5; \quad \frac{\partial C(M_2)}{\partial X_2} = -0,75; \quad C(M_2) = 5,31;$$

Поскольку $|C(M_2) - C(M_1)| = 3,94 > \varepsilon$, мы должны выполнить третью итерацию.

$$X_1^3 = 4,75 - 0,5(-1,5) = 5,5; \quad X_2^3 = 4 - 0,5(-0,75) = 4,4.$$

$$\frac{\partial C(M_3)}{\partial X_1} = -0,38; \quad \frac{\partial C(M_2)}{\partial X_2} = -4,74; \quad C(M_3) = 4,31;$$

$$|C(M_3) - C(M_2)| = 1 = \varepsilon.$$

Учитывая, что требуемая точность результата достигнута, процесс вычисления издержек прекращается. Таким образом, минимальные издержки на хранение при объемах сырья $X_1 = 5,5$ и $X_2 = 4,4$ составляют $C(M_3) = 4\,631$ тыс. руб.

5.4. Нахождение экстремума функции отклика от многих переменных при наличии ограничений

Задача поиска экстремума функции отклика от многих переменных при наличии ограничений усложняется. Например, нужно найти минимум функции $C(X)$ при ограничениях:

$$h_j(X) = b_j; \quad j = 1, \overline{m}. \quad (5.29)$$

При этом предполагается выполнение условий применения методов оптимизации: функции $C(X)$ и $h_j(X)$ обладают непрерывными

частными производными по аргументам X_1, X_2, \dots, X_n . В таких случаях используется метод *неопределенных множителей Лагранжа*. Суть метода состоит в следующем. Составляется вспомогательная функция:

$$\begin{aligned} L(X_1, X_2, \dots, X_n, \lambda_1, \lambda_2, \dots, \lambda_m) = \\ = L(X_1, X_2, \dots, X_n) + \sum_{j=1}^m \lambda_j (b_j - h_j(X_1, X_2, \dots, X_n)), \end{aligned} \quad (5.30)$$

которая называется функцией Лагранжа, а множители $\lambda_j, j = \overline{1, m}$ называются неопределенными множителями Лагранжа. Обозначим $\Lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$ и запишем функцию Лагранжа в более компактном виде:

$$L(X, \Lambda) = C(X) + \sum_{j=1}^m \lambda_j \cdot (b_j - h_j(X)). \quad (5.31)$$

Эту функцию Лагранжа нужно минимизировать. Для решения задач (5.29–5.30) с помощью (5.31) имеем систему уравнений:

$$\frac{\partial L(X, \Lambda)}{\partial X_i} = \frac{\partial C(X)}{\partial X_i} + \sum_{j=1}^m \lambda_j \frac{\partial h_j(X)}{\partial X_i} = 0; \quad i = \overline{1, n}, \quad (5.32)$$

$$\frac{dL(X, \Lambda)}{d\lambda_j} = b_j - h_j(X) = 0; \quad j = \overline{1, m} \quad (5.33)$$

Как видим, идея метода неопределенных множителей Лагранжа заключается в том, что исходная задача (5.29), (5.31) отыскания условного экстремума целевой функции $C(X)$ сводится к задаче безусловного экстремума функции Лагранжа $L(X, \Lambda)$, которая решается классическим методом – путем приравнивания к нулю частных производных по $X_j; j = \overline{1, n}$ и $\lambda_j; j = \overline{1, m}$.

Отметим, что при применении метода неопределенных множителей Лагранжа также существуют следующие проблемы. Во-первых, для отыскания точек экстремума функции $L(X, \Lambda)$ требуется проводить анализ производных

$$\frac{\partial^2 L(X, \Lambda)}{\partial X_i^2} \quad \text{и} \quad \frac{\partial^2 L(X, \Lambda)}{\partial \lambda_j^2}.$$

Во-вторых, условия (5.32) и (5.33) не являются достаточными для существования решений X^* задачами (5.29) и (5.31), то есть не обязательно, чтобы любое решение системы (5.32) и (5.33) было решением задачи (5.29), (5.31). Однако простота составления функции Лагранжа и возможность обобщения задачи оптимизации на случай ограничений-неравенств в (5.33) обусловили широкое применение метода неопределенных множителей Лагранжа в современных исследованиях СС.

Итак, если нужно найти решение задачи (5.29), (5.31) методом неопределенных множителей Лагранжа, мы должны выполнить следующие действия:

1) составить функцию Лагранжа (5.31);

2) вычислить частные производные $\frac{\partial^2 L(X, \Lambda)}{\partial X_i^2}$, $i = \overline{1, n}$, и $\frac{\partial^2 L(X, \Lambda)}{\partial \lambda_j^2}$

$j = \overline{1, m}$ и приравнять их к нулю, получив таким образом систему (5.32), (5.33);

3) решить систему уравнений (5.32), (5.33) любым известным способом, получив тем самым стационарные точки (X, Λ) ;

4) исследовать функцию (5.29) на наличие экстремума в полученных стационарных точках.

Рассмотрим следующий *пример*. Найти прямоугольник наименьшего периметра, ограничивающий заданную площадь.

Математическая модель задачи имеет следующий вид:

найти $\min f(x_1, x_2) = 2x_1 + 2x_2$ при ограничениях $x_1 x_2 = s$; $x_1 \geq 0$; $x_2 \geq 0$.

Запишем функцию Лагранжа $L(x_1, x_2, \lambda) = 2x_1 + 2x_2 + \lambda(s - x_1 x_2)$ и составим систему уравнений

$$\begin{cases} \frac{\partial L}{\partial x_1} = 2 - \lambda x_2 = 0, \\ \frac{\partial L}{\partial x_2} = 2 - \lambda x_1 = 0, \\ \frac{\partial L}{\partial \lambda} = s - x_1 x_2; \end{cases} \Rightarrow \begin{cases} x_1 = \frac{2}{\lambda}; \\ x_2 = \frac{2}{\lambda}; \\ \lambda^2 s = 4. \end{cases}$$

Решение этой системы (рисунок 5.9):

$$x_1^* = \sqrt{s}; \quad x_2^* = \sqrt{s}; \quad \lambda = \frac{2}{\sqrt{s}}.$$

На рисунке 5.9 пунктирной линией изображены уровни функции цели $f(x_1, x_2) = 2x_1 + 2x_2$. На линии $x_1, x_2 = s$ лежат точки, удовлетворяющие ограничениям задачи, и точка $M(x_1^* = \sqrt{s}; x_2^* = \sqrt{s})$ является решением задачи.

Более подробно с методикой оптимизации при нелинейной целевой функции откликов математических моделей можно ознакомиться в работе [8].

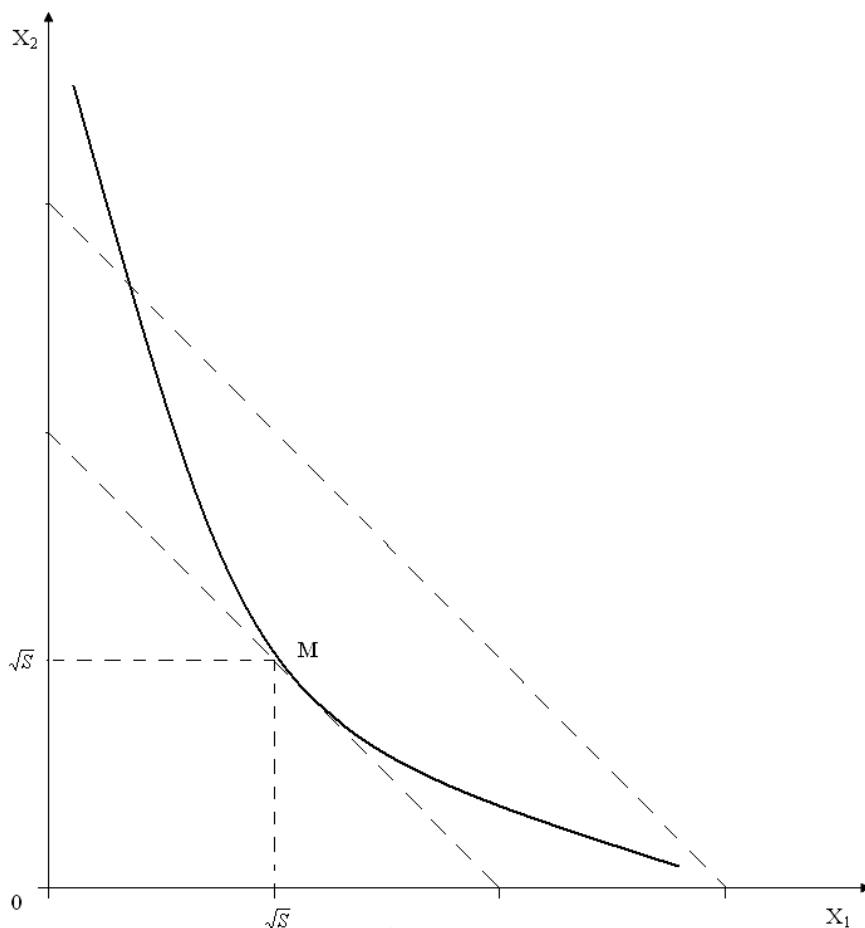


Рисунок 5.9 – Пример нахождения решения системы уравнений

6. ЭЛЕМЕНТЫ ПРИНЯТИЯ РЕШЕНИЙ ПРИ ВЕРОЯТНОСТНОМ МОДЕЛИРОВАНИИ СЛОЖНЫХ СИСТЕМ

6.1. Виды стратегий исследователя при анализе сложных систем

Управление СС с помощью вероятностных или имитационных моделей, а также проектное моделирование СС предполагают, как правило, достижение определенной цели. Для этого исследователям приходится принимать определенные решения. Однако только в простейших случаях удается указать целевую функцию, значения которой измеряют качество принятого решения. В сложных ситуациях качество решений не может быть единственной функцией. Поэтому механизм рационального выбора в таких случаях требует дополнительной информации, позволяющей сравнивать возможные альтернативы решений. Часто возникает необходимость принимать решения, для которых не удастся полностью учесть предопределяющие их условия, а также представляющие их влияние (эффект неопределенности) на функционирование СС. В таких ситуациях ответственность за принимаемые решения очень велика. Поэтому исследователи стремятся к оптимальному использованию имеющейся информации относительно поставленной задачи, чтобы, взвесив все возможные варианты решения, постараться найти среди них наилучший. При этом, в зависимости от имеющейся в их распоряжении осведомительной информации и от наличия ресурсов для организации исследований, возможны разные комбинации вариантов анализа СС (стратегий изучения и оценка последствий от принятия решений). На выбор стратегии исследователя СС влияют следующие основные факторы:

- наличие информации об исследуемом процессе в СС;
- величина риска потерь от ошибочных решений;
- размер ресурсов для организации исследований на модели;
- количество и иерархия целей, преследуемых исследователем при выборе решения.

Если информации о СС недостаточно, то имеет место принятие решений в условиях неопределенности. Когда же при этом необходимо учитывать возможные потери от ошибочных решений, то перед исследователем стоит дополнительная задача – принятие решений в условиях риска. Если количество целей большое и зачастую они

противоречат друг другу, то исследователю приходится иметь дело с многоцелевым выбором решений. Как правило, наиболее трудными являются задачи векторной оптимизации, решение которых предполагает нахождение оптимального соотношения параметров управления СС. Наконец, размеры ресурсов, которыми располагает исследователь при использовании моделей, играют существенную роль при выборе решения. При решении большинства практических задач моделирования СС исследователи ощущают недостаток ресурсов для постановки исследований, анализа результатов и принятия обоснованного решения [36].

6.2. Принятие решений в условиях неопределенности и риска

Принятие решений представляет собой выбор одного из некоторого множества вариантов $E_i \in E$. Условимся считать, что каждый вариант E_i вырабатывает некоторую количественную оценку e_i . В качестве такой характеристики может выступать обобщенный отклик (скаляр как результат «свертки» спектра одним из ранее рассмотренных способов). Например, можно считать, что e_i характеризует такие величины, как полезность, надежность, выигрыш, прибыль. Таким образом, выбор оптимального варианта СС производится с помощью критерия

$$E_0 = \left\{ E_{2i} / E_{i0} \in E \cap e_{i0} = \max_i e_i \right\}. \quad (6.1)$$

Случай, когда каждому варианту решения соответствует единственное внешнее состояние (случай детерминированных решений), является простейшим. Однако при решении большинства практических задач каждому допустимому варианту решения E_i могут соответствовать (вследствие различных внешних условий) разные состояния F_j и количественные оценки решений e_{ij} . Подобную ситуацию проиллюстрируем на следующем примере. Пусть требуется изготовить изделие, долговечность которого зависит от вида материала, из которого оно состоит, и внешних условий, связанных с той или иной степенью нагрузки при эксплуатации изделия. Нагрузки считаются известными. Требуется определить вид материала, из которого целесообразно изготовить изделия. Варианты решений в данном примере таковы:

E_1 – выбор материала из соображений максимальной долговечности;

E_m – выбор материала из соображений минимальной долговечности;

E_i – промежуточные решения ($i = \overline{2, m-1}$).

Внешние условия, учет которых необходим, следующие:

F_1 – условия, обеспечивающие максимальную долговечность;

F_L – условия, обеспечивающие минимальную долговечность;

F_j – промежуточные условия ($j = \overline{2, n-1}$).

По результатам решения e_{ij} будем понимать оценку, соответствующую варианту решения E_i и условию F_j , характеризующую экономический эффект (прибыль), полезность или надежность изделия. Ситуацию можно охарактеризовать матрицей решений $\|e_{ij}\|$:

	F_1	F_2	\dots	F_n
E_1	e_{11}	e_{12}	\dots	e_{1n}
E_2	e_{21}	e_{22}	\dots	e_{2n}
\cdot	\dots	\dots	\dots	\dots
\cdot				
\cdot				
E_m	e_{m1}	e_{m2}	\dots	e_{mn}

По данной матрице необходимо выбрать тот вариант решения, которому соответствует наилучший результат. Но так как неизвестно, какое из внешних условий может наступить, необходимо принимать во внимание все оценки e_{ij} . Таким образом, первоначальную задачу максимизации согласно критерию (6.1) нужно заменить другой задачей.

Процедуру выбора в случае нескольких внешних состояний можно реализовать следующим образом. Матрица решений $\|e_{ij}\|$ дополняется некоторым столбцом, т. е. каждому варианту E_i приписывается некоторый результат e_{ir} . Проблема состоит в том, какой смысл вложить в результат e_{ir} . Оценочные функции можно вводить различным образом. Если, например, последствия каждого из альтернативных решений характеризовать комбинацией из его наибольшего и наименьшего результатов, то e_{ir} вычисляется следующим образом:

$$e_{ir} = \min_j e_{ij} + \max_j e_{ij}. \quad (6.2)$$

Наилучший в этом смысле результат имеет вид:

$$\max_i e_{ir} = \max_i \left\{ \min_j e_{ij} + \max_j e_{ij} \right\}. \quad (6.3)$$

Определяя таким образом желаемый результат, лицо, принимающее решение (ЛПР), исходит из компромисса между оптимистическим и пессимистическим подходами. Зависимость (6.1) называют *критерием компромисса*. Целесообразность применения той или иной оценочной функции определяется комплексом условий исследования СС. В качестве других оценочных функций используются следующие варианты критериев.

Оптимистический критерий. Наилучший результат используется *азартным* исследователем, который делает ставку на то, что при любом варианте его решения внешняя среда находится в максимально благоприятном состоянии:

$$\max_i e_{ir} = \max_i \left\{ \max_j e_{ij} \right\}. \quad (6.4)$$

Позиция нейтралитета. Исследователь исходит из того, что все встречающиеся отклонения результата решения зависят от «среднего» случая состояния внешней среды. В этом случае в качестве *критерия нейтралитета* используется выражение:

$$\max_i e_{ir} = \max_i \left\{ \frac{t}{n} \sum_{j=1}^n e_{t_0} \right\}. \quad (6.5)$$

Позиция пессимиста. Исследователь ориентируется на то, что выпадет наименее благоприятное состояние внешней среды. Он приписывает каждому из альтернативных вариантов наихудшее значение и затем выбирает самый выгодный вариант, дающий максимум из минимумов вариантов. В качестве *критерия пессимиста* используется выражение:

$$\max_i e_{ir} = \max_i \left\{ \max_k -e_k \right\}. \quad (6.6)$$

Позиция относительного пессимиста. Для каждого варианта решения ЛПР оценивает потери по сравнению с определенным

по каждому варианту наилучшим результатом, а затем из совокупности наихудших результатов выбирает наилучший вариант согласно данной оценочной функции. В качестве критерия относительного пессимиста используется выражение:

$$\min_i e_{ir} = \min_i \max_j \left\{ \max_i e_{ij} - e_{ir} \right\}. \quad (6.7)$$

Влияние исходной позиции ЛПР на эффективность результатов интерпретируем с помощью графического изображения на плоскости, соответствующего двум внешним состояниям при m вариантах решения. Введем прямоугольную систему координат. По оси абсцисс отложим значение результатов решений e_{i1} , соответствующее внешнему состоянию F_1 , а по оси ординат отложим значение e_{i2} , соответствующее состоянию внешней среды F_2 . Таким образом, каждому варианту E_i соответствует на плоскости XOY точка (e_{i1}, e_{i2}) ; $i = \overline{1, m}$. Все m точек (e_{i1}, e_{i2}) лежат внутри прямоугольника, стороны которого параллельны координатным осям, а противоположными вершинами являются: *утопическая* точка (УТ) с координатами $(\max e_{i1}, \max e_{i2})$, *антиутопическая* точка (АУТ) с координатами $(\min e_{i1}, \min e_{i2})$ $i = \overline{1, m}$. Данный прямоугольник называется *полем полезности решений*. Чтобы сравнить варианты решений с точки зрения их качества, назовем вариант E_i не худшим, чем вариант E_j , если для соответствующих точек (e_{i1}, e_{i2}) и (e_{j1}, e_{j2}) выполняются неравенства $e_{i1} \geq e_{j1}$ и $e_{i2} \geq e_{j2}$. При этом решение E_i считается лучшим, если хотя бы одно из двух неравенств является строгим. Очевидно, что при таком определении не все варианты решения допускают сравнение друг с другом, так как в общем случае существуют такие варианты решений E_i и E_j , что, например, $e_{i1} < e_{j1}$ и $e_{i2} > e_{j2}$. Это означает, что в поле полезности решений установлены отношения частного порядка.

Выберем в поле полезности некоторую точку РТ. С помощью прямых, параллельных координатным осям, разобьем плоскость XOY на четыре части и обозначим их соответственно I, II, III, IV. В случае произвольной размерности эти части будем называть *конусами*. Все точки конуса I в смысле введенного выше порядка являются лучшими, чем точка РТ, и соответственно, все точки конуса III являются заведомо худшими, чем точка РТ. Потому конус I называется *конусом предпочтительности*, а конус III – *антиконусом*. Оценка же точек конусов II и IV является неопределенной, поэтому эти конусы назовем *конусами неопределенности*. Для точек конусов неопределенности

оценки вариантов осуществляются только с помощью выбранного критерия принятия решений. На рисунке 6.1 приведен пример графического изображения поля полезности в ординатах (e_1 и e_2).

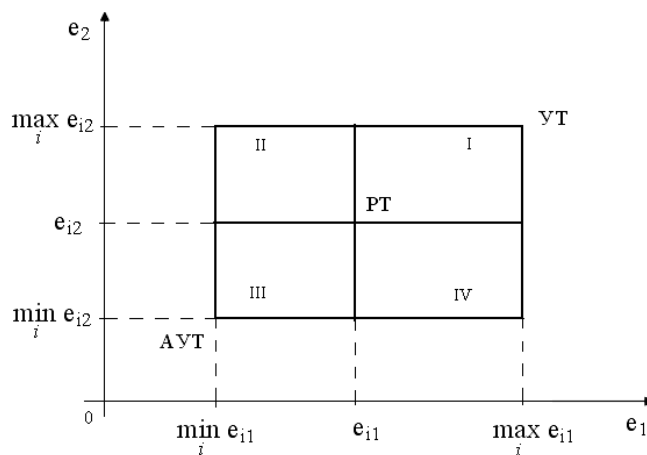


Рисунок 6.1 – Поле полезности решений

6.3. Критерии принятия решений в условиях неопределенности и риска

Минимаксный критерий (ММ-критерий). Этот критерий использует оценочную функцию, соответствующую позиции крайнего пессимиста:

$$Z_{MM} = \max_i e_{ir}, \quad e_{u2} = \max_{ij} e_{ij} \div \min_{ij} e_{ij}. \quad (6.8)$$

Выбранные таким образом варианты полностью исключают риск. Однако это достоинство критерия стоит исследователю потерь. Поэтому применение ММ-критерия оправдано, если в СС ситуация характеризуется следующими обстоятельствами:

- о возможности появления состояний внешней среды F_j ничего не известно;
- решение принимается одно или малое число раз;
- когда необходимо исключить какой бы то ни было риск.

Критерий Севиджа (S-критерий). Оценочная функция критерия Севиджа имеет вид:

$$Z_s = \min_i e_{ir} = \min_i \max_j \left\{ \max_k e_{kj} - e_{ij} \right\}. \quad (6.9)$$

Величины $a_{ij} = \max_i e_{ij} - e_{ij}$ трактуются как дополнительный выигрыш, если вместо варианта E_i при состоянии внешней среды F_j выбрать другой, оптимальный для данного состояния внешней среды. Условия для применения критерия Севиджа такие же, как и для MM -критерия.

Критерий Байеса–Лапласа (BL -критерий). Применяется в тех случаях, когда априори известна вероятность q_j появления внешнего состояния F_j . Оценочная функция имеет вид:

$$Z_{BL} = \max_i e_{ir} = \sum_{j=1}^n e_{ij} \cdot q_j; Z_{BL} = \max_i e_{ir} = \sum_{j=1}^n e_{ij} \cdot q_j. \quad (6.10)$$

Применение критерия рекомендуется, если ситуация в СС характеризуется следующим образом:

- вероятности q_j появления состояний F_i известны и не зависят от времени;
- решение принимается (теоретически) бесконечно много раз;
- при малом числе принятия решений допускается некоторый риск.

Критерий Ходжа–Лемана (HL -критерий). Критерий представляет собой комбинацию BL -критерия и MM -критерия. С помощью параметра v определяется степень доверия ЛПР к имеющимся распределениям вероятностей состояния внешней среды. Если это доверие ЛПР велико, то используется BL -критерий, в противном случае предпочтение отдается MM -критерию. Оценочная функция имеет вид:

$$Z_{HL} = \max_i e_{ir}; e_{ij} = v \sum_{j=1}^n e_{ij} q_j + (1-v) \min_i e_{ij}; 0 \leq v \leq 1. \quad (6.11)$$

Критерий Ходжа–Лемана предъявляет к ситуации принятия решения следующие требования:

- вероятности q_j появления событий F_j неизвестны, но возможны некоторые предположения о распределении этих вероятностей;
- допускается большое число принятия решений;
- при малом числе принятия решений допускается некоторый риск.

Критерий Гурвица (HW -критерий). Стараясь занять уравновешенную позицию, ЛПР может использовать критерий Гурвица. Его

оценочная функция представляет собой средневзвешенные между точками зрения предельного оптимиста и крайнего пессимиста:

$$Z_{HW} = \max_i e_{ir}; e_{ir} = C \cdot \min_i e_{ij} + (1 - C) \cdot \max_j e_{ij}; 0 \leq C \leq 1, 0. \quad (6.12)$$

Здесь весовой множитель обычно принимается равным $C = 0,5$. Для применения HW -критерия должны выполняться следующие требования:

- о вероятности появления q_j состояний внешней среды F_j ничего не известно;
- решение принимается малое число раз;
- допускается некоторый риск.

Критерий Гермейера (G-критерий). Критерий Гермейера ориентирован на оценку величины потерь, т. е. при его применении предполагается, что обобщенный отклик модели СС e_{ij} является отрицательной величиной. Оценочная функция имеет вид:

$$Z_G = \max_i e_{ir}; e_{ir} = \max_i \min_j e_{ij} \cdot q_j. \quad (6.13)$$

При решении ряда производственных и экономических задач обычно ЛПР имеет дело с ценами и затратами. Поэтому выполняется условие $e_{ij} < 0$. Если среди e_{ij} имеются положительные величины, то путем преобразования $(e_{ij} - \alpha)$ при подходящем выборе величины $\alpha > 0$ матрица решений преобразуется к виду с отрицательными компонентами. Основная трудность состоит в определении величины $\alpha > 0$. Условия применимости G -критерия следующие:

- вероятности q_j появления состояний F_j известны;
- допускается некоторый риск;
- решение может реализоваться как малое, так и большое число раз.

Однако, если вероятности q_j известны с малой достоверностью β и число принятий решений мало, то при использовании G -критерия появляется неоправданно большой риск.

Составной BL (ММ)-критерий. В тех случаях, когда эмпирически было определено множество вероятностей состояний F_j внешней среды $q = (q_1, \dots, q_n)$, но не известно, какое из событий происходит, используется составной критерий BL (ММ)-критерий. Фиксируется опорное значение ММ-критерия: $Z_{MM} = \max_i \min_j e_{ij} = e_{i_0 j_0}$. Посредством некоторого заданного, или выбираемого уровня допустимого

риска определяется некоторое множество согласия, являющееся подмножеством I_1 индексов (i, \dots, m) . Возможные потери по сравнению со значением MM -критерия равны $Z_i = e_{i_0j_0} - \min_j e_{ij}$. С другой стороны, в результате такого снижения требований к риску ошибки открываются возможности для увеличения выигрыша по сравнению с тем, который обеспечивается MM -критерием. Поэтому рассматривается некоторое выигрышное подмножество индексов $I_2 = E_{ij} \leq \max_j e_{ij} - \max_j e_{i_0j}$. Тогда во множестве пересечений $I_1 \cap I_2$ соберутся только такие варианты решений, для которых, с одной стороны, в определенных состояниях могут иметь потери по сравнению с состоянием, задаваемым MM -критерием, но зато в других состояниях имеется по меньшей мере такой же прирост выигрыша. В итоге оптимальными в смысле составного критерия BL (MM)-критерия будут решения из множества:

$$E_0 = \max_{i \in I_1 \cap I_2} \left\{ \sum_{j=1}^n e_{ij} q_j \cap \sum_{j=1}^n q_j = 1 \right\}. \quad (6.14)$$

Применение BL (MM)-критерия целесообразно, если:

- вероятность появления состояний F_j неизвестна, однако имеется некоторая априорная информация в пользу какого-либо определенного распределения;
- необходимо считаться с появлением различных состояний как по отдельности, так и в комплексе;
- допускается ограниченный риск;
- принятое решение реализуется один только раз или многократно.

6.4. Методы поиска решений в экспертных системах

Классификация методов решения задач в экспертных системах (ЭС). Методы решения задач, основанные на сведениях их к поиску решений, зависят от особенностей проблемной области, в которой решается задача, и от требований, предъявляемых пользователем к решению.

Особенности проблемной области, с точки зрения методов решения, можно характеризовать следующими параметрами:

- размер пространства, в котором решается задача;
- изменяемость области;
- полнота модели, описывающей область;

– определенность данных о решаемой задаче.

Параметр «размер» характеризует размер пространства, в котором предстоит искать решение. Параметр «изменяемость» характеризует степень изменяемости области во времени и пространстве. Различают статические и динамические области. Параметр «полнота модели» характеризует адекватность модели, используемой для описания данной проблемной области. Если модель неполна, то для описания данной области используют несколько моделей, дополняющих друг друга. Параметр «определенность данных» характеризует степень точности – ошибочности и полноты-неполноты данных. Ошибочность означает, что данные о проблемной области неточны. Неполнота означает недостаточность входных данных для однозначного решения задачи. Требования пользователя к результату задачи определяются количеством решений; свойствами результата и способами его получения. Параметр «свойство результата» включает время решения, объем памяти, указание об обязательности использования каких-либо знаний и т. п.

Задачи, сложность которых определяется перечисленными выше параметрами, могут быть решены различными методами. Существующие методы решения задач, используемые в ЭС, можно классифицировать следующим образом:

- методы поиска в одном пространстве – это методы, предназначенные для использования в следующих условиях: малые области, статические области, полнота модели, точные и полные данные;
- методы поиска в иерархических пространствах – методы, предназначенные для работы в больших статических областях;
- методы поиска при неточных и неполных данных;
- методы поиска в динамической проблемной области – это методы, предназначенные для работы с областями, изменяемыми во времени и (или) в пространстве;
- методы поиска, использующие несколько моделей, предназначенные для работы с областями, для адекватного описания которых одной модели недостаточно.

Поиск решений в одном пространстве состояний. Задача поиска в пространстве состояний может быть сформулирована следующим образом.

Пусть задана тройка показателей (S_0, F, S_t) , где S_0 – множество начальных состояний (условия задачи), F – множество операторов задачи, отображающих одни состояния в другие, S_t – множество конечных состояний (решения задачи). Осуществить поиск решения задачи – значит определить такую последовательность операторов, которая

преобразует начальные состояния в конечные. Процесс решения можно представить в виде графа $G = (X, Y)$, $X = \{x_0, x_1, \dots\}$ – множество вершин графа, каждая из которых отождествляется с одним из состояний, Y – множество, содержащее пары вершин (x_i, x_j) , где x_i, x_j принадлежат X . Итак, граф G задает пространство состояний, т. е. пространство, в котором осуществляется поиск решения. Построение пространства осуществляется следующим образом. Берется некая вершина x_0 из области начальных состояний X_0 , к ней применяются все возможные операторы, порождающие все дочерние вершины. Этот процесс называется процессом раскрытия вершин. Если получена целевая вершина, то она не раскрывается. Процесс построения пространства состояний заканчивается, когда все нераскрытые вершины являются целевыми или терминальными. Терминальными называются вершины, к которым нельзя применить никаких операторов.

На практике требуется обеспечить полноту поиска, т. е. организовать поиск так, чтобы все целевые вершины были найдены. Надежным способом обеспечения полноты является полный перебор.

В зависимости от порядка, в котором перебираются вершины, различают поиск в глубину и поиск в ширину.

При поиске в глубину сначала раскрывается та вершина, которая была построена самой последней. При практической реализации поиск в глубину в некотором направлении завершается в следующих случаях: при достижении целевой вершины; при построении в ходе поиска вершины, глубина которой превышает некоторую граничную глубину.

При поиске в ширину вершины раскрываются в том же порядке, в котором они порождаются.

Если в пространстве состояний ввести операторы, переводящие состояние S_i в предыдущее состояние S_{i-1} , то поиск можно осуществлять не только в направлении от начального к целевому, но и в обратном направлении. Поиск первого типа называют поиском от данных (прямой поиск), а поиск второго типа – поиском от цели (обратный поиск).

Более того, можно организовать поиск в обоих направлениях одновременно (двунаправленный или бинарный поиск) [52].

На рисунках 6.2–6.3 представлены графы, демонстрирующие поиск в глубину и поиск в ширину соответственно. Вершины пронумерованы в том порядке, в котором они раскрываются.

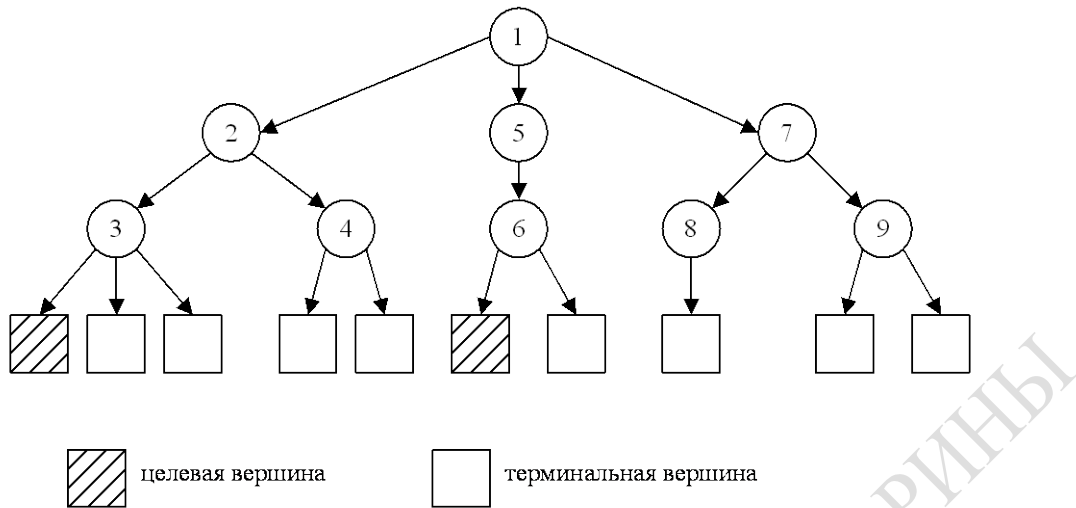


Рисунок 6.2 – Граф, демонстрирующий поиск в глубину

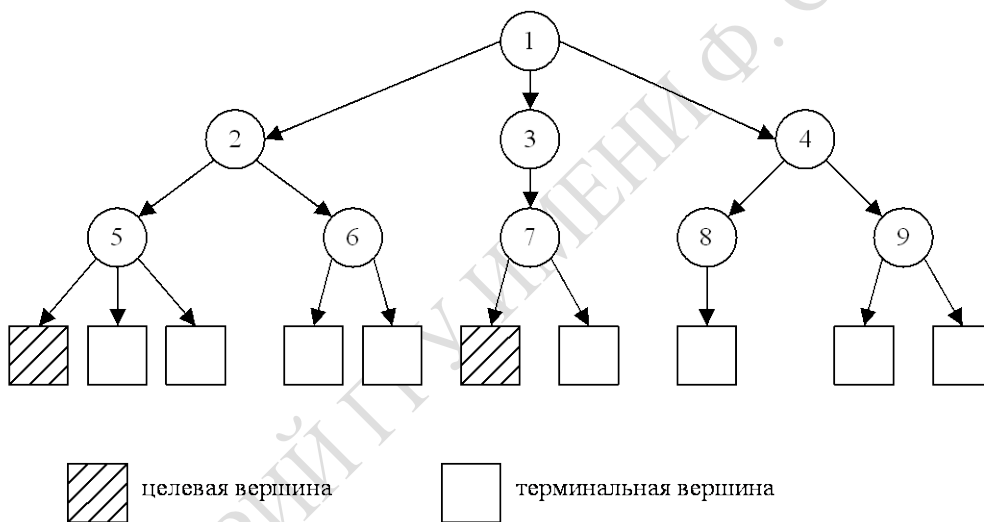


Рисунок 6.3 – Граф, демонстрирующий поиск в ширину

Оба метода анализируют одинаковое число вершин, однако поиск в ширину требует больше памяти, так как он запоминает все пути поиска (а не один, как при поиске в глубину).

Поиск методом редукции. Решение задачи этим методом сводится к решению совокупности образующих ее подзадач. Процесс решения задачи разбиением ее на подзадачи можно представить в виде специального направленного графа G , называемого И/ИЛИ графом. Каждой вершине этого графа ставится в соответствие описание некоторой задачи (подзадачи). В графе выделяют два типа вершин: конъюнктивные вершины и дизъюнктивные вершины. Конъюнктивные вершины (И) вместе со своими дочерними вершинами интерпретируются так: решение задачи сводится к решению всех ее подзадач, соответствующих дочерним вершинам конъюнктивной вершины.

Дизъюнктивные вершины (ИЛИ) вместе со своими дочерними вершинами интерпретируются так: решение задачи сводится к решению любой из ее подзадач, соответствующих дочерним вершинам дизъюнктивной вершины.

Пример графического представления разбиения задачи на подзадачи представлен на рисунке 6.4. Здесь S_0 – исходная задача, для решения которой требуется решить подзадачу S_3 или подзадачи S_1 и S_2 . Решение S_1 сводится к решению подзадачи S_4 либо подзадачи S_5 . Решение задач S_2, S_5, S_7 предполагается известным, решение задач S_4 и S_6 неизвестно. В приведенном примере задача S_0 может быть решена либо путем решения задачи S_3 , либо путем решения задач S_1 и S_2 .

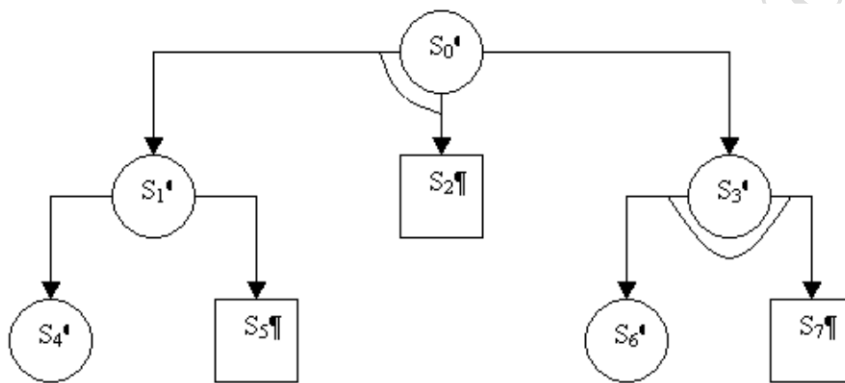


Рисунок 6.4 – Графическое представление процесса разбиения задачи на подзадачи

Графическое представление соответствующего И/ИЛИ графа представлено на рисунке 6.5. При переходе вводится дополнительная вершина R_1 , так как любая вершина должна быть И-вершиной или ИЛИ-вершиной.

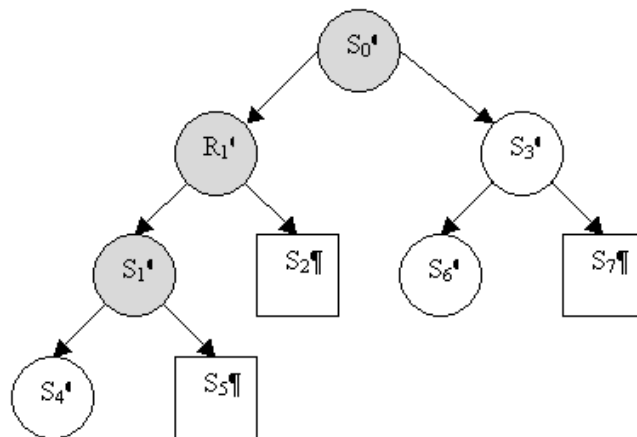


Рисунок 6.5 – Графическое представление операций И/ИЛИ

Решение задачи при поиске методом редукции сводится к нахождению в И/ИЛИ графе решающего графа. Решающий граф определяется как подграф из разрешимых вершин, который показывает, что начальная вершина разрешима.

Рекурсивное определение разрешимой вершины:

- конечные (целевые) вершины разрешимы, так как их решение известно по исходному предположению;
- вершина ИЛИ разрешима тогда и только тогда, когда разрешима, по крайней мере, одна из ее дочерних вершин;
- вершина И разрешима тогда и только тогда, когда разрешима каждая из ее дочерних величин.

На рисунке 6.5 двойными линиями выделен решающий граф задачи S_0 , а конечные вершины обозначены прямоугольниками.

Для графа И/ИЛИ так же, как и для поиска в пространстве состояний, можно определить поиск в глубину и поиск в ширину как в прямом, так и в обратном направлении.

Заметим, что метод редукции является обобщением подхода с использованием пространства состояний. Метод поиска в глубину в И/ИЛИ графе используется в ЭС MYCIN, EMYCIN [52].

Эвристический поиск. Методы поиска в глубину и ширину называют слепым поиском, поскольку в этих методах порядок раскрытия вершин предопределен и никак не зависит от расположения цели. Стремление сократить время поиска привело к созданию эвристических методов поиска. Эти методы используют некоторую информацию о проблемной области для рассмотрения не всего пространства поиска, а таких путей в нем, которые с наибольшей вероятностью приводят к цели. Например, один из способов сокращения перебора состоит в выборе «информационного» оператора, который не строит так много вершин, не относящихся к делу.

Другой способ состоит в использовании эвристической информации для определения на каждом шаге дальнейшего направления перебора. Для этого вводится мера «перспективности» вершины в виде некоторой оценочной функции. Как правило, оценочные функции пытаются количественно оценить расстояние от текущей вершины до конечной. Из двух вершин при одинаковой глубине перспективней та, от которой меньше расстояние до цели. Недостатком эвристических методов является потеря полноты решения.

Метод «генерация-проверка». Процесс поиска может быть сформулирован в терминах «генерация-проверка». Как правило, пространство поиска явно не задано. Поэтому для осуществления процесса поиска необходимо генерировать очередное возможное решение

и проверить, не является ли оно результирующим. При этом необходимо, чтобы генератор удовлетворял требованиям полноты и избыточности, т. е. генерировал все возможные решения и каждое решение только один раз.

При генерации текущего возможного решения возникает проблема распределения знаний между генератором и устройством проверки. Увеличение знаний генератора проблемной области приводит к сокращению пространства, в котором осуществляется поиск. Однако при этом повышаются затраты на генерацию каждого очередного состояния (подзадачи). Если генератор прост, то генерируется много решений и, следовательно, много времени уходит на проверку.

Можно выделить важную форму метода «генерация-проверка», называемую иерархической «генерацией-проверкой». В этом случае на верхнем уровне генератор вырабатывает не полное, а частично определенное решение. Устройство проверки по виду частичного решения определяет, ведет ли оно к успеху. Если проверка не отвергает решение, то на следующем уровне генератор продолжает вырабатывать из данного частичного решения все полные решения. Устройство проверки определяет, являются ли они целевыми.

Поиск решений в иерархии пространств. Перечисленные методы поиска в одном пространстве не позволяют решать сложные задачи, поскольку с увеличением размера пространства время поиска решений экспоненциально растет. При большом размере пространства поиска можно попробовать разбить общее пространство на подпространства и осуществлять поиск сначала в них. В данном случае говорят об иерархии подпространств. Развитием этой идеи является дополнение конкретных пространств абстрактными пространствами, то есть пространствами, которые имеют описание только наиболее важных сущностей.

Рассмотрим *пример* использования абстрактных пространств. Пусть требуется переехать из центра города A в центр города B . Если осуществлять поиск требуемого пути на детальной карте, содержащей все улицы во всех городах, встретившихся по дороге, то задача может стать практически неразрешимой. При определении пути из города A в город B целесообразно спланировать маршрут по крупномасштабной карте (т. е. осуществить поиск в абстрактном пространстве), а затем по детальной карте спланировать выезд из города A и въезд в город B .

Поиск в факторизованном пространстве. Пространство называется факторизованным, если оно разбивается на пересекающиеся подпространства частичными решениями. Поиск в факторизованном пространстве осуществляется на основе метода иерархической «генерации-проверки». Генератор вырабатывает текущее частичное

решение, а затем проверяется, может ли это решение привести к успеху. Если текущее частичное решение отвергается, то из рассмотрения без генерации и проверки устраняются все полные решения этого класса. Если текущее частичное решение не отвергается, то генератор вырабатывает на его основе все полные решения, а устройство проверки определяет, являются ли эти решения целевыми.

Поиск в фиксированном множестве пространств. Применение рассмотренного выше метода ограничено тем, что для ряда областей не удастся по частичному решению сделать заключение об его непригодности. Например, при решении задач планирования и конструирования по фрагменту плана нельзя сказать, что этот фрагмент не может являться частью полного решения.

В рассматриваемом методе пространство поиска разбивается на фиксированную последовательность подзадач (пространств), с помощью которых можно решить любую входную задачу.

Подобный метод поиска использован в системе ЭС R_1 [52]. На основании заказа покупателя на требуемую ему конфигурацию системы VAX система R_1 выдает диаграммы, изображающие пространственные взаимосвязи компонент VAX, и указывает, какие компоненты необходимы для функционирования заказанной конфигурации. Система R_1 разбивает общую задачу на шесть подзадач. Порядок, в котором вызываются эти подзадачи, зависит от заказанной конфигурации. Действия, выполняемые каждой подзадачей, зависят от комбинации заказанных компонент и способа их взаимосвязи. Каждой подзадаче соответствует свой набор правил, т. е. каждая задача решается в своем подпространстве.

Поиск в изменяющемся множестве иерархических пространств. Существуют задачи, которые нельзя свести к фиксированному набору подзадач. Например, задачи планирования перемещений в пространстве. Для решения подобных задач может быть использован метод нисходящего уточнения.

Для того чтобы упростить процесс решения некоторой задачи в сложном пространстве, целесообразно получить обобщенное пространство и получить решение в этом пространстве. Указанный прием можно повторять многократно. При этом процесс решения задачи выглядит как «нисходящее» движение в иерархии пространств от наиболее абстрактного к конкретному, в котором получается окончательное решение. Существенной характеристикой такого процесса является поиск решения задачи в абстрактном пространстве, преобразование этого решения в решение более низкого уровня и т. д.

Формирование более абстрактного пространства осуществляется путем игнорирования части описаний менее абстрактного пространства. Игнорирование описаний осуществляется на основе ранжирования описаний по степени важности. Часто ранжирование осуществляется на основе ранжирования описаний по степени важности, а также на основе учета степени неизменности фактов. Наиболее абстрактными считаются те описания, которые не могут изменяться.

Программа ABSTRIPS [15] является наиболее известным примером использования метода «нисходящего уточнения». Она составляет план перемещения роботом объектов между компонентами. Отметим, что в этом методе абстрактные пространства создаются индивидуально, в соответствии с решаемой задачей.

Метод, основанный на принципе наименьших свершений. Основной недостаток предыдущего метода состоит в том, что он не имеет обратной связи. При решении ряда задач детализация решения, полученного на абстрактном уровне, оказывается невозможна, так как при построении абстрактного решения были опущены детали, препятствующие его уточнению, т. е. требуется пересмотр абстрактного решения.

В подобных ситуациях применяется принцип наименьших свершений. В соответствии с ним решение не строится сразу до конца на верхних уровнях абстракции. Частичное решение детализируется постепенно, по мере появления информации, подтверждающей возможность решения и вынуждающей принять решение. При этом необходимо, чтобы система была в состоянии:

- определять, когда накопилось достаточно информации для принятия решения;
- приостанавливать работу над некоторой подзадачей, когда для решения нет достаточной информации;
- переходить с одной подзадачи на другую, возобновляя выполнение приостановленной подзадачи при появлении недостающей информации;
- объединять информацию, полученную различными подзадачами.

Принцип наименьших свершений использует экспертную систему ЭС MOLGEN [16], предназначенную для планирования экспериментов по молекулярной генетике. Система чередует использование принципа наименьших свершений и использование эвристических стратегий. Таким образом, принцип наименьших свершений координирует процесс поиска решения с наличием необходимой информации и в соответствии с доступной информацией перемещает фокус активности по решению задачи от одной подзадачи к другой.

Использование метапространства в иерархии пространств.

При решении любой задачи многократно возникает вопрос: «Что делать на следующем шаге?» В простейших случаях решение о том, что делать на следующем шаге, предопределено методом поиска решения. В последнее время разрабатываются системы, которые планируют процесс своих рассуждений. В такой постановке задачи речь идет о решении задачи на метауровне. Решатель в метапространстве содержит явное описание процесса организации поиска, т. е. описание состояний, операторов, условий применимости операторов, описание доступных методов поиска и способов их взаимодействия.

Получить решение в метапространстве – это значит определить, какой метод будет применен на следующем шаге, т. е. составить метаплан решения задачи. Отметим, что метаплан, в отличие от абстрактного плана, выражается не в терминах операторов проблемной области, а в терминах методов, известных решателю. Примером ЭС с метапланированием является система MOLGEN [46].

Таким образом, методы поиска в иерархии пространств используют пространства трех видов: абстрактные, конкретные и метапространства. В одной системе могут использоваться пространства всех трех типов. А поиск решения в пределах одного пространства осуществляется любым из рассмотренных способов.

Поиск решений в альтернативных пространствах. Ранее рассмотренные методы исходят из предпосылки о том, что знания о проблемной области точны и полны. Однако эксперту часто приходится работать в условиях неполноты и неточности знаний. Когда эксперт решает задачу, он использует методы, отличающиеся от формальных математических рассуждений. В математических рассуждениях каждое заключение должно строго следовать из предыдущей информации. В противоположность этому эксперт делает правдоподобные предположения, основанные на частичной информации, истинность которых не доказана.

Таким образом, для того, чтобы система могла делать умозаключения, основанные на здравом смысле, при работе с неполными знаниями, она должна быть способна делать предположения, а при получении новой информации – отказываться от сделанных предположений и выводов, полученных на их основе. Мнение системы о том, какие факты имеют место, изменяются в ходе рассуждения. Таким образом, если рассматривать проблемную область как статическую, неполнота знаний влечет за собой рассмотрение этой области при различных предположениях. А это приводит к представлению области

в виде альтернативных возможных пространств, соответствующих различным предположениям и мнениям.

Система мнений – множество мнений, свойственных некоторому индивиду (системе). Основываясь на некоторой системе мнений, можно образовать пространство поиска, предназначенное для решения каких-либо задач. В ходе рассуждений человек (система) может менять свои мнения, образуя различные системы мнений. Совокупность мнений, которых система придерживается в текущий момент, называется активной системой мнений. Каждой из систем мнений соответствует свое пространство поиска, а все вместе они образуют альтернативные пространства.

Для того чтобы изменить мнение, система должна быть способна рассуждать о зависимостях, существующих в активной системе мнений. Новые мнения могут быть следствием новой информации, полученной извне или вновь выведенной. Зависимости в системе мнений должны содержать сведения о мнениях, правилах, выводах и обоснованиях мнений. Простейшим видом обоснования может являться информация о том, на каких мнениях основывается данное мнение.

В рассмотренных ранее методах поиск в некотором направлении прерывался при достижении целевого или терминального состояния. Если достигалось целевое состояние, то либо работа завершалась, либо продолжался поиск следующего решения. При достижении терминального состояния осуществлялся возврат в некоторое предыдущее состояние пространства и продолжается поиск в новом направлении. Обычно при реализации поиска использовался механизм, который восстанавливает состояние непосредственно предшествующее текущему и затем выбирает очередную альтернативу. Более подробно с методиками поиска решений в экспертных системах при диалоге с ЭВМ можно ознакомиться в работе [35].

7. СПОСОБЫ ОРГАНИЗАЦИИ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ

7.1. Понятие о модельном времени

Для того чтобы понять механизм имитации функционирования СС на ЭВМ, рассмотрим два примера.

Пример 1. Объектом имитации является движение i -й космической ракеты. Можно выделить ряд стадий ее движения, в ходе которых выполняется последовательность функциональных действий (ФД). Пусть каждая стадия движения ракеты характеризуется своим номером j . При $j = 1$ осуществляется запуск ракеты i на стартовой площадке. На этой стадии выполняется функциональное действие ФД_{i1} . Коррекция движения и сброс первой ступени i -й ракеты на стадии 2 определяют ФД_{i2} . Вторая коррекция движения и сброс второй ступени i -й ракеты означают выполнение ФД_{i3} . Движение множества ракет представляет собой СС, и каждая i -я ракета является компонентой K_i .

Пример 2. Объектом имитации является последовательная обработка i -й детали на нескольких станках. Каждый j -й станок обеспечивает выполнение своего набора операций обработки детали i , реализуя таким образом функциональное действие ФД_{ij} . Обработку множества деталей можно рассматривать как СС, в которой каждая i -я деталь является компонентой K_i .

В обоих случаях функционирование компоненты K_i СС представляет собой последовательность ФД_{ij} . Будем говорить, что в результате выполнения ФД_{ij} в СС происходит событие S_{ij} . Каждое из событий в реальной СС связано, как правило, с соответствующей компонентой K_i . При этом любое ФД_{ij} выполняется на некотором временном интервале τ_{ij} . Для каждой K_i введем понятие локального времени t_i . В СС все t_i изменяются одновременно, однако характер этих изменений различен и определяется последовательностью временных интервалов $\{\tau_{ij}\}$.

При построении ИМ СС ФД_{ij} аппроксимируются некоторыми упрощенными функциональными действиями $\text{ФД}'_{ij}$. Степень этого упрощения определяет уровень детализации ИМ. Отличия $\text{ФД}'_{ij}$ от ФД_{ij} порождают ошибки имитации реальной СС. В ИМ ФД_{ij} представляется парой $(\text{ФД}'_{ij}, \tau_{ij})$, которая выполняется следующим образом:

вначале реализуется $\Phi D'_{ij}$ при неизменном значении t_i , а затем уже отображается изменение t_i на величину τ_{ij} , инициируя таким образом появление события C_{ij} .

На рисунке 7.1 представлен пример развития действий K_i в системе координат $(\Phi D'_{ij}, t_i)$. Конечно, подобная система координат носит условный характер, поскольку по оси ординат нельзя отложить «значение» функционального действия $\Phi D'_{ij}$. Подобное представление используется только для отображения того факта, что с изменением «локального времени» t_i некоторая K_i выполняет несколько различных ΦD_{ij} . Так, согласно рисунку 7.1 K_i последовательно выполняет ΦD_{i1} , ΦD_{i2} , ΦD_{i3} , а в СС соответственно происходят события C_{i1} , C_{i2} , C_{i3} . Причем с появлением каждого нового C_{ij} происходят изменение ΦD_{ij} и увеличение его временной координаты t_i соответственно на величину τ_{i1} , τ_{i2} , τ_{i3} . Условно на рисунке 7.1 появление событий в K_i при выполнении ΦD_{ij} показано штрихпунктирной линией. В ИМ появление событий реализуется ступенчатой линией $(0, a, C_{i1}, b, C_{i2}, d, C_{i3})$. Это означает, что вначале выполняется $\Phi D'_{i1}$ при неизменном t_i , а затем уже отображается изменение t_i на величину τ_{i1} , инициируя появление события C_{i1} . Затем реализуется $\Phi D'_{i2}$ при неизменном времени t_i на величину τ_{i2} , инициируя в K_i свершения события C_{i2} . Далее в два приема (реализация $\Phi D'_{i3}$ и последующее изменение t_i на величину τ_{i3}) аналогичным образом инициируется событие C_{i3} . Отметим при этом, что порядок появления события может быть и обратным: сначала изменяется t_i , а затем уже выполняется $\Phi D'_{ij}$.

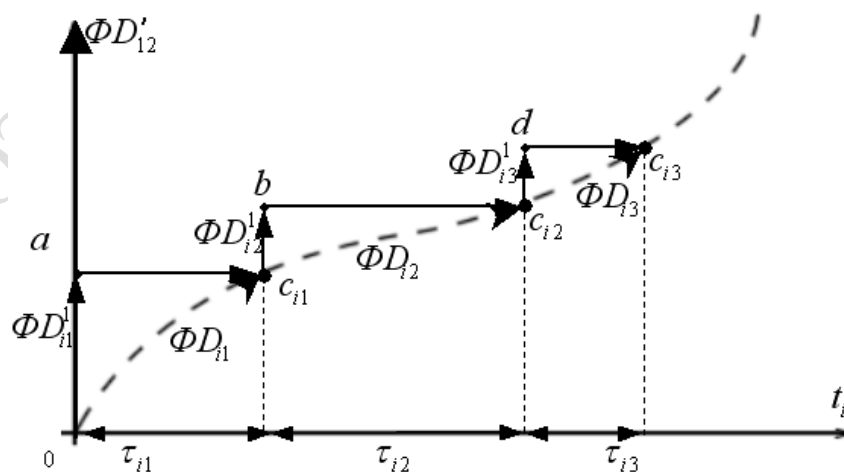


Рисунок 7.1 – Пример аппроксимации функциональных действий i -й компоненты системы (K_i) в имитационной модели СС

В ИМ каждое $\Phi D'_{ij}$ описывается в общем случае некоторым алгоритмом AL_{ij} . В ходе имитации происходит реализация $\Phi D'_{ij}$ по соответствующим алгоритмам AL_{ij} и последующее изменение t_i на величину τ_{ij} . Таким образом, любая ИМ описывается набором некоторых «молекул», каждая из которых содержит в себе описание алгоритма выполнения AL_{ij} соответствующего $\Phi D'_{ij}$ и оператора Mt_{ij} , осуществляющего изменение временной координаты t_i на величину τ_{ij} . Пару (AL_{ij}, τ_{ij}) обычно называют ij -й активностью ИМ и обозначают AK_{ij} . Иногда активности называют «работами» [60]. Любая ij -я активность представляет собой запись поведения компоненты K_i имитационной модели СС. Реализация этой активности в ИМ приводит к появлению в модели СС события S_{ij} .

Если бы на ЭВМ имитировалось поведение только одной компоненты системы, то выполнение активностей в ИМ можно было бы осуществить строго последовательно, и дело свелось бы к пересчету временной координаты t_i после очередного выполнения алгоритма AL_{ij} . В действительности СС состоит из нескольких компонент. Все эти компоненты K_i функционируют одновременно. Это должна отражать ИМ. В большинстве современных ЭВМ в каждый момент времени может реализоваться алгоритм только одной из компонент модели.

Чтобы обеспечить имитацию параллельных событий реальной системы, вводят некоторую глобальную переменную t_0 , которую называют модельным (системным) временем. С помощью этой переменной организуется синхронизация всех событий S_{ij} в модели и выполнение алгоритмов AL_{ij} компонент K_i модели системы.

При реализации ИМ используются обычно три представления времени: t_R – реальное время системы, работа которой имитируется на данной ИМ; t_0 – модельное время, по которому организуется синхронизация событий в системе; t_s – машинное время имитации, отражающее затраты ресурса времени ЭВМ на организацию имитации.

С помощью модельного времени t_0 реализуется квазипараллельная работа компонент ИМ. Приставка «квази» в данном случае отражает последовательный характер обслуживания событий в ИМ, одновременно возникающих в разных компонентах реальной системы.

Корректировка временных координат t_i нескольких K_i ИМ осуществляется с помощью модельного времени t_0 следующим образом. Если значения t_i при выполнении AL_{ij} нескольких K_i совпадают (это означает, что в реальной системе происходит одновременно несколько событий S_{ij}), то последовательно обслуживаются AL_{ij} , совпадающие по времени выполнения, т. е. имеющие одинаковые значения t_{ij} .

Здесь и далее под t_{ij} будем понимать конкретное значение t_i , при котором происходит событие C_{ij} . При этом модельное время t_0 не меняется до окончания выполнения всех совпавших по времени реализации алгоритмов $АЛ_{ij}$. Таким способом последовательно выполняются соответствующие $ФД'_{ij}$ при неизменном значении t_0 . После каждой реализации $АЛ_{ij}$, обеспечивающей выполнение в ИМ $ФД'_{ij}$, выполняется оператор корректировки временной координаты $Мt_{ij}$. Чаще всего эта корректировка сводится к вычислению нового значения t_{ij} по формуле:

$$t_{ij} = t_0 + \tau_{ij}. \quad (7.1)$$

Это значение временной координаты t_i запоминается и используется в дальнейшем для определения момента новой активизации в ИМ компоненты K_i . Под активизацией компоненты модели K_i будем понимать начало выполнения следующей ее активности (выполнение алгоритма $АЛ_{ij}$ и оператора корректировки временной координаты t_i).

Когда имитация одновременно появившихся событий C_{ij} завершена, выполнены соответствующие алгоритмы активностей $АЛ_{ij}$ и проведены корректировки временных координат t_i , меняется значение глобальной переменной модели t_0 . Существуют два способа изменения t_0 : с помощью фиксированных и переменных интервалов изменения модельного времени. Часто их называют соответственно способами фиксированного шага и шагов до следующего события.

Для того чтобы легче было представить оба способа организации изменения модельного времени, рассмотрим следующий пример (рисунок 7.2). Пусть в системе функционируют три компоненты K_i ($i = \overline{1,3}$). При функционировании K_1 последовательно происходят четыре события ($C_{11}, C_{12}, C_{13}, C_{14}$) соответственно четырем моментам изменения t_1 ($t_{11}, t_{12}, t_{13}, t_{14}$). Между этими моментами K_1 выполняет четыре различных функциональных действия ($ФД_{11}, ФД_{12}, ФД_{13}, ФД_{14}$). Каждое из указанных функциональных действий выполняется в течение соответствующих интервалов времени ($\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}$). Аппроксимация $ФД_{ij}$ осуществляется последовательностью $\{ФД'_{ij}\}$, $j = \overline{1,4}$. Аналогичным образом K_2 последовательно выполняет три функциональных действия ($ФД'_{2j}$, $j = \overline{1,3}$) соответственно в течение трех интервалов времени (τ_{2j}), которые в ИМ аппроксимируются соответственно $ФД'_{2j}$. Наконец, K_3 также последовательно выполняет три функциональных действия ($ФД'_{3j}$, $j = \overline{1,3}$) соответственно в течение

трех интервалов времени (τ_{3j}), аппроксимируемых в ИМ соответственно тремя $\Phi D'_{3j}$. Отметим, что каждое $\Phi D'_{ij}$ описывается соответствующим алгоритмом $АЛ_{ij}$ и реализуется соответствующей активностью, представляющей собой выполнение этого алгоритма при неизменном значении t_i и модификацию временной координаты t_i на величину τ_{ij} . Алгоритм функционирования K_i в координатах $(\Phi D'_i, t_i)$ представлен на рисунке 7.2. В ходе выполнения активностей ($АЛ_{ij}, \tau_{ij}$) в модели K_i происходят последовательно события C_{ij} .

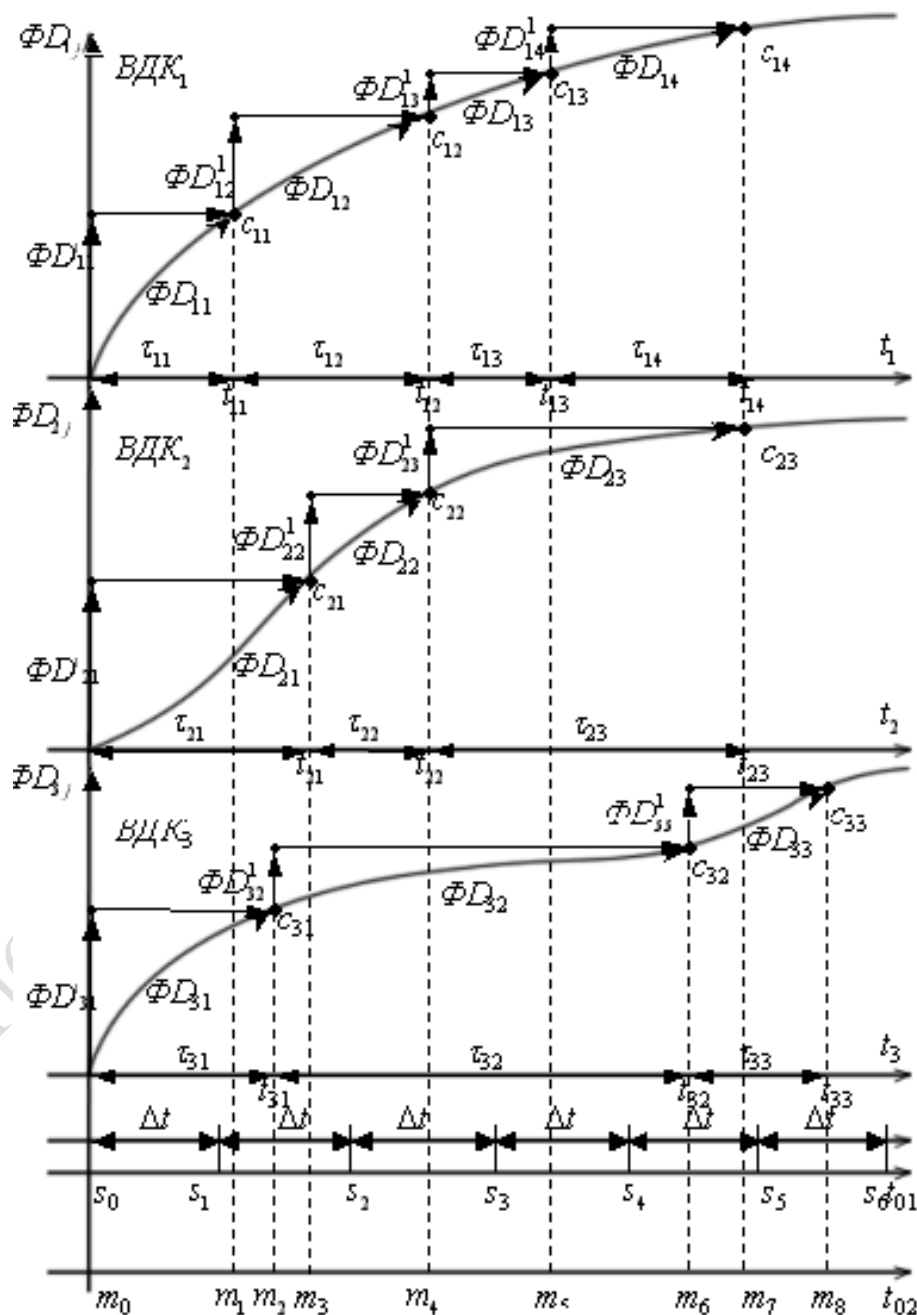


Рисунок 7.2 – Временная диаграмма моделирования событий в СС, состоящей из трех компонент

Рассмотрим имитацию событий C_{ij} в системе при каждом из способов изменения t_0 . При имитации по способу фиксированного шага модельное время, обозначенное на рис. 7.2. как t'_0 , меняется каждый раз на величину шага Δt . Тогда в моменты времени $0, \Delta t_1, 2\Delta t_1, 3\Delta t_1$ и т. д. в ИМ происходят соответственно события S_0, S_1, S_2 и т. д., означающие переход на обслуживание управляющей программой моделирования тех событий C_{ij} , которые попадают внутрь очередного интервала Δt . Как видно из рисунка 7.2, в действительности события C_{11}, C_{31}, C_{21} происходят на интервале (S_0, S_1) . Считается, что они происходят одновременно в момент $t' = \Delta t$. Управляющая программа моделирования (УПМ) последовательно обслуживает эти события, что сводится к инициализации и выполнению соответствующих активностей $(АЛ_{11}, \tau_{11}; АЛ_{31}, \tau_{31}; АЛ_{21}, \tau_{21})$. Затем в момент $t'_0 = \Delta 2t$ УПМ инициирует событие S_2 . Считается, что события C_{12} и C_{22} также происходят одновременно, поэтому УПМ последовательно инициирует выполнение активностей $(АЛ_{12}, \tau_{12})$ и $(АЛ_{22}, \tau_{22})$. Время t'_0 получает приращение на заранее выбранную величину Δt , а моменты реагирования УПМ на появления событий S_l не связаны с моментами t_{ij} появления событий в реальной системе. Таким образом, точность моделирования событий C_{ij} определяется шагом Δt изменения модельного времени t'_0 и, как правило, все события обслуживаются в точке, соответствующей верхней границе интервала изменения t'_0 .

При имитации по способу шагов до следующего события время t''_0 меняется в моменты m_i , которые соответствуют моментам t_{ij} появления событий в реальной системе. Обработку событий C_{ij} , одновременно появляющихся в реальной системе, УПМ осуществляет последовательно при неизменном модельном времени t''_0 . Например, события C_{12} и C_{22} в реальной системе происходят одновременно. В модели вначале реализуется, например, алгоритм $АЛ_{12}$ и модифицируется координата t_1 на величину τ_{12} , а затем выполняется алгоритм $АЛ_{22}$ и модифицируется координата t_2 на величину τ_{22} . Таким образом, t''_0 каждый раз сдвигается на величину, равную минимальному значению приращения модельного времени τ_{ij} , которое получают координаты t_i .

Независимо от способа изменения t_0 механизм регламентации изменения модельного времени обычно предусматривает выполнение следующих действий:

- выбор событий в модели, которые необходимо обслужить при одном и том же модельном времени t_0 ;

- обслуживание событий (инициализация активностей), которые имеют одинаковое время инициализации;
- по окончании обслуживания всех одновременных (в пределах шага) событий определение очередного значения модельного времени;
- корректировка временной координаты модели t_0 ;
- проверка условий окончания моделирования либо по времени завершения имитации, либо по выполнению других событий в системе.

Все эти действия выполняет УПМ. Каждое событие C_{ij} в модели системы, являющееся результатом выполнения активности AK_{ij} , обслуживается УПМ и требует для реализации некоторого ресурса времени работы ЭВМ. Поэтому чем чаще обслуживаются события УПМ, тем больше увеличивается машинное время t_3 . С этой точки зрения способ фиксированного шага изменения t_0 является более экономичным, поскольку имеет место групповое обслуживание всех событий C_{ij} , которые попали внутрь очередного шага изменения Δt модельного времени.

На практике предпочтение способу фиксированного шага отдается в двух случаях. Во-первых, когда события C_{ij} распределены равномерно на всем интервале моделирования и исследователь может подобрать интервал изменения временной координаты Δt , обеспечивающий минимальную погрешность имитации. Во-вторых, когда событий очень много и они появляются группами. Во всех остальных случаях способ задания шага до следующего события более предпочтителен. В тех случаях, когда события C_{ij} распределены неравномерно и появляются они через значительные временные интервалы τ_{ij} , способ задания шага до следующего события экономичнее и точнее способа фиксированного изменения t_0 . С учетом достоинств и недостатков каждого из способов изменения t_0 наметилась специализация их применения при моделировании СС.

На практике наибольшее распространение получил способ шага до следующего события. Поэтому в дальнейшем будем рассматривать все методологические вопросы, связанные только со вторым способом изменения модельного времени.

7.2. Структурная схема имитационных моделей сложных систем

Как уже установлено, в ИМ происходит аппроксимация каждого ФД_{ij} парой действий: упрощенным функциональным действием ФД'_{ij} и модификацией временной координаты i -й компоненты модели t_i . Эта пара действий представляет собой «молекулу» ИМ и называется

активностью AK_{ij} . Любая AK_{ij} состоит из описания алгоритма AL_{ij} и оператора модификации временной координаты компоненты модели Mt_{ij} . Для синхронизации выполнения компонент ИМ на ЭВМ используется глобальная переменная, называемая модельным временем t_0 . Кроме модификации модельного времени t_0 и выполнения операторов Mt_{ij} в функции УПМ входят: запуск на выполнение алгоритмов AL_{ij} , организация взаимодействия AL_{ij} друг с другом в ходе имитации и проверка условий окончания имитации. Таким образом, любая ИМ представляет собой совокупность набора «молекул», отражающих поведение объекта имитации, и УПМ, организующей взаимодействия этих «молекул» друг с другом.

Для задания входных условий, начальных значений параметров и запуска ИМ в состав модели включается подпрограмма (ПП), организующая начало имитации. ИМ создается для изучения на основе статистики, характеризующей поведение компонент K_i , реальной системы в различных режимах ее функционирования. Для сбора статистики моделирования в ИМ вводится соответствующая программа, которая присутствует в ИМ либо в явном виде, либо рассредоточена по всем компонентам ИМ. Наконец, по достижении условий окончания имитации УПМ передает управления на подпрограмму окончания имитации, которая зачастую и обеспечивает расчет характеристик поведения модели системы и выдает исследователю результаты моделирования. Все эти подпрограммы взаимодействуют только с УПМ.

Таким образом, описание ИМ существенно разрастается в объеме по сравнению с собственно описанием объекта имитации. Можно говорить о двух частях ИМ. Первая часть ИМ является переменной и создается исследователем системы с помощью имеющихся средств автоматизации моделирования. Вторая часть ИМ представляет собой реализацию этих средств автоматизации моделирования в виде: процедур, обеспечивающих выполнение операторов синхронизации модели (например, Mt_{ij}), подпрограмм запуска и завершения имитации, УПМ и подпрограмм сбора статистики моделирования.

На рисунке 7.3 представлена схема перевода объекта моделирования в его имитационную модель. Из рисунка видно, что любое FD_{ij} в ИМ описывается соответствующей активностью AK_{ij} . Каждая AK_{ij} представляет собой попарное сочетание описания алгоритма AL_{ij} и оператора модификации временной координаты компоненты модели Mt_{ij} . УПМ передает управление на начало программы, реализующей алгоритмы AL_{ij} . Возврат на УПМ имеет место при выполнении операторов Mt_{ij} . УПМ и служебные подпрограммы обычно являются универсальными и не изменяются при переходе от одной модели к

другой. Однако принципы их построения и способ управления выполнением активностей зависят от класса объектов моделирования. Вопросы организации квазипараллельного выполнения активностей интересны для специалистов по созданию средств автоматизации моделирования. Составление же текста описания ИМ и использование системных средств при разработке $АЛ_{ij}$ являются областью интересов исследователей функционирования СС.

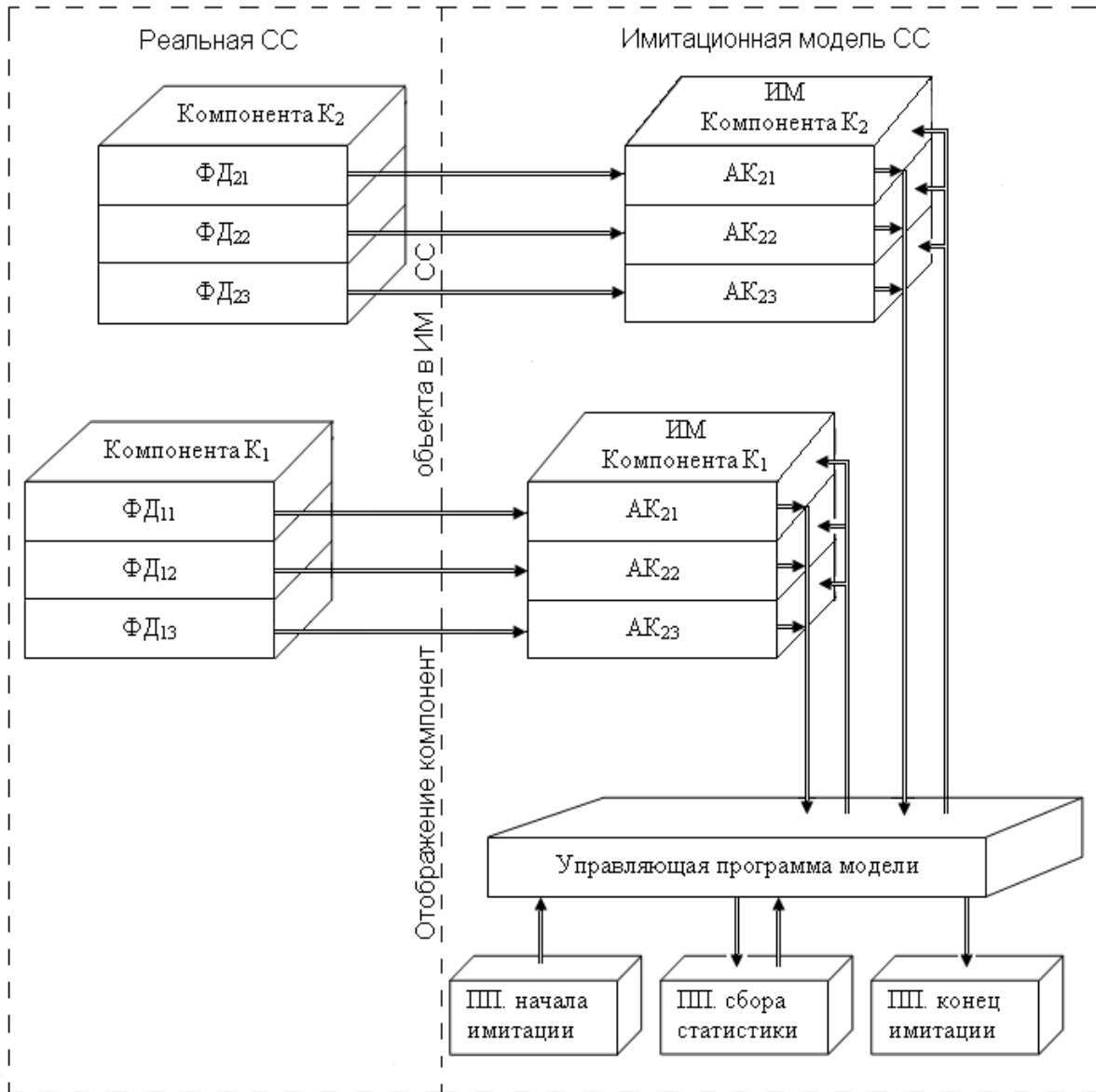


Рисунок 7.3 – Различие между объектом моделирования и его имитационной моделью

В моделях сложной системы состав активностей в разных компонентах и характер взаимодействия друг с другом могут быть различными. В одних системах все $ФД_{ij}$ различны, и для реализации каждого алгоритма $АЛ_{ij}$ активности в модели требуется выполнение своих условий. В других системах некоторые $ФД_{ij}$ в разных компонентах

аналогичны, совпадают по времени и приводят к одному и тому же событию C_{ij} . Для реализации в ИМ каждой такой группы ΦD_{ij} одним алгоритмом AL_{ij} требуется выполнение своих определенных условий.

Для некоторых систем существует сильное взаимодействие между ΦD_{ij} , которые можно аппроксимировать явно задаваемыми математическими зависимостями при создании ИМ. В зависимости от состава алгоритмов AL_{ij} , наличия связей между компонентами, целей и задач моделирования выбирается тот или иной способ представления K_i и реализации активностей в имитационных моделях. Чаще всего составление описания ИМ называют формализацией объекта моделирования. Наибольшее распространение получили пять способов описания ИМ: непосредственно активностями, аппаратом событий, транзактами, агрегатами, процессами. Каждому способу формализации объекта моделирования соответствует свой способ организации квазипараллелизма обслуживания УПМ активностей, из которых составлена ИМ. Поэтому различают соответственно следующие способы организации квазипараллелизма в ИМ: просмотр активностей, составление расписания событий, управление обслуживанием транзактов, управление агрегатами, синхронизация процессов.

Поэтому различаем следующие способы организации квазипараллелизма в ИМ: путем просмотра активностей; составления расписания событий; имитации с помощью потоков транзактов; представления СС набором агрегатов, путем просмотра процессов, взаимодействия транзактов с процессами; организации взаимодействия агрегатов с процессами.

Одну и ту же СС принципиально можно представить любым из перечисленных способов формализации. Однако ИМ, построенные на их основе, будут отличаться размерами и качеством ресурсов, затрачиваемых на их создание, испытание и использование.

Отличительные особенности и принципы организации квазипараллелизма в ИМ каждым из перечисленных способов рассмотрим на примере, в котором в качестве объекта моделирования выбрана система, состоящая из трех компонент K_i (рисунок 7.4). В соответствии с временной диаграммой перехода K_i в состояния C_{ij} в системе координат $\Phi D'_{ij}$ время его выполнения (рисунок 7.2) установлено следующее. Компонента K_1 выполняет четыре функциональных действия (ΦD_{11} , ΦD_{12} , ΦD_{13} , ΦD_{14}) во времени t_1 , которые в ИМ описываются соответственно четырьмя активностями. Аналогичным образом компоненты K_2 и K_3 выполняют по три функциональных действия (ΦD_{2j} , ΦD_{3j} , $j = \overline{1,3}$) во временах соответственно t_2 и t_3 . Опуская для простоты

некоторые детали, рассмотрим основные функции УПМ и принципы ее взаимодействия с компонентами модели системы K_i при каждом из способов организации квазипараллелизма.

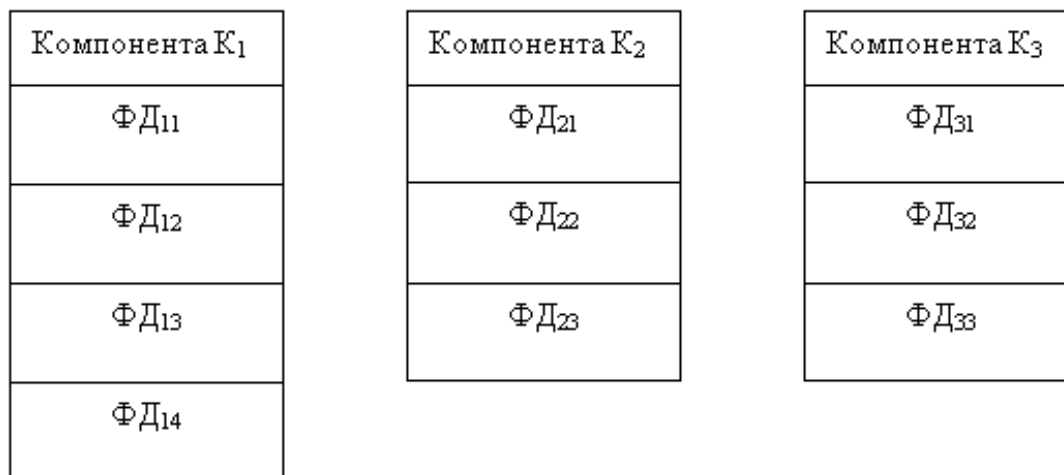


Рисунок 7.4 – Структура обмена, используемого для иллюстрации способов имитации

7.3. Организация квазипараллелизма в имитационных моделях путем просмотра активностей

Данный способ организации квазипараллелизма используется при моделировании реальных систем, характеризующихся следующими особенностями. Все $\Phi Д_{ij}$ компонент K_i реальной системы различны, причем для выполнения каждой из них требуется выполнение своих условий. Эти условия конкретны, известны заранее исследователю и могут быть представлены алгоритмически. В результате выполнения $\Phi Д_{ij}$ в системе происходят различные события C_{ij} . Связи между $\Phi Д_{ij}$ отсутствуют, все $\Phi Д_{ij}$ функционируют независимо друг от друга. В таких случаях исследователь описывает ИМ в виде двух частей: множества активностей $\{AK_{ij}\}$ и набора процедур проверки выполнимости условий инициализации активностей. Под инициализацией AK_{ij} понимают передачу управления от УПМ на выполнение AL_{ij} данной активности. В процедурах проверки выполнимости условий инициализации AK_{ij} реализуется зависимость выполнения $\Phi Д_{ij}$ от конкретной ситуации, имеющей место в реальной системе на каждый момент t_{ij} . Выполнение на ЭВМ алгоритма AL_{ij} данной активности назовем обслуживанием AK_{ij} . Завершается обслуживание AK_{ij} выполнением оператора модификации временной координаты компоненты Mt_{ij} . По этому оператору

управление возвращается УПМ. Таким образом, имитация на ЭВМ представляет собой чередование выполнения алгоритмов активностей, операторов модификации временной координаты t_i и алгоритма УПМ. Вся ИМ представляет собой набор активностей AK_{ij} , каждая из которых после выполнения на ЭВМ возвращает управление УПМ. Схема взаимодействия активностей с УПМ представлена на рисунке 7.5.

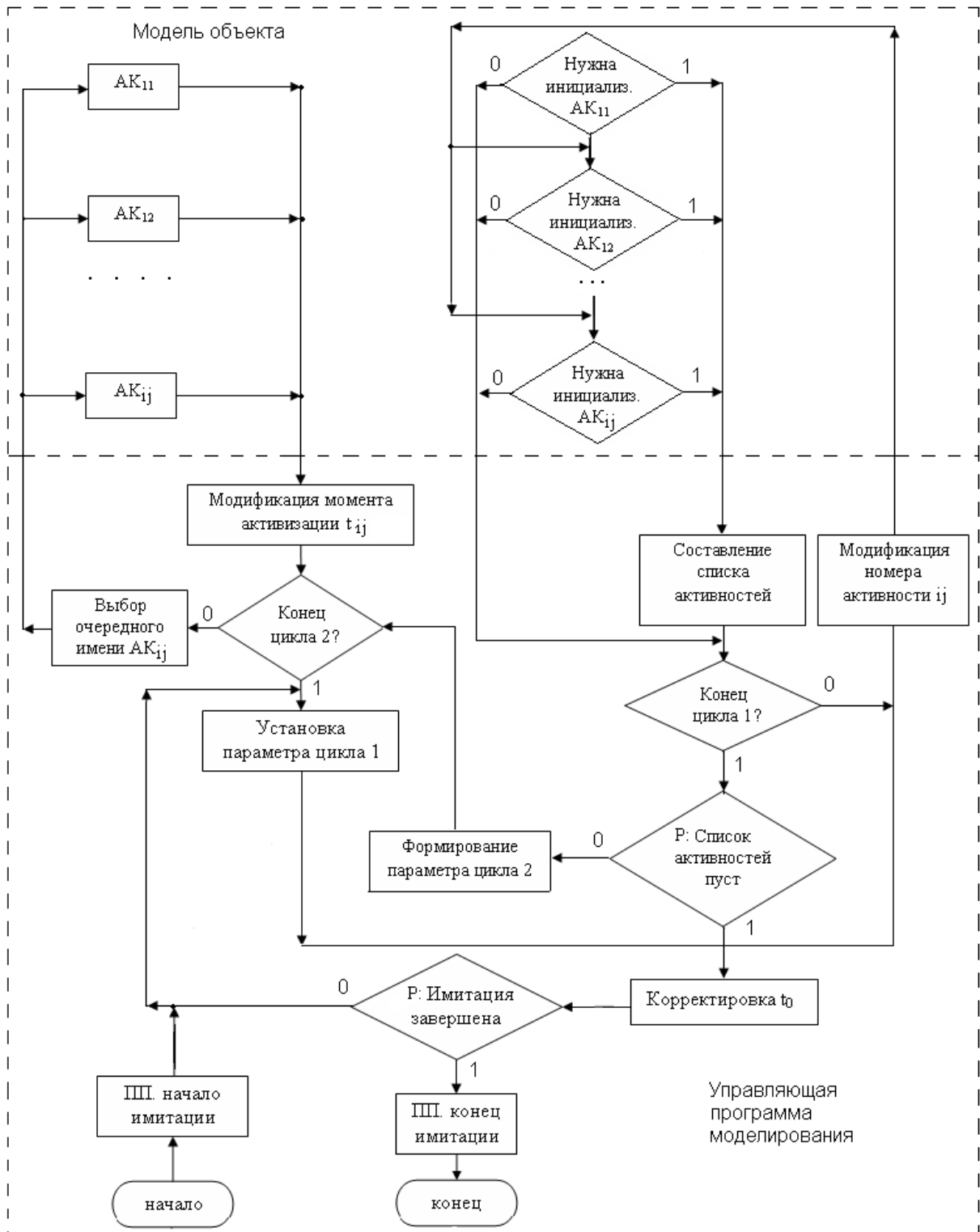


Рисунок 7.5 – Схема ИМ объекта, состоящей из активностей

Перед началом имитации соответствующая подпрограмма устанавливает значения начальных состояний компонент модели K_i и параметров, определяющих условия инициализации активностей AK_{ij} . Обычно проверка выполнимости условия инициализации AK_{ij} состоит либо в определении значений параметров модели, либо в вычислении моментов t_{ij} , в которые должно начаться выполнение соответствующего FD_{ij} , либо в проверке значений переменных модели. С помощью алгоритма реализуется своя последовательность проверок выполнимости условия инициализации AK_{ij} . Для простоты предполагаем, что каждая AK_{ij} имеет только одну проверку условия ее инициализации (рисунок 7.5). Отметим, что пользователь описывает не только алгоритмы активностей AL_{ij} и операторы модификации временных координат Mt_{ij} , но и алгоритмы проверки условий инициализации AK_{ij} .

Алгоритм инициализации УПМ AK_{ij} состоит в следующем. По первому циклу (параметр цикла равен максимальному числу AK_{ij} в ИМ) управление последовательно передается на проверку выполнимости условий активизации AK_{ij} . Имена тех активностей, для которых эти условия выполнены, заносятся в список иницилируемых AK_{ij} . По завершении этого цикла начинается второй цикл (параметр цикла равен числу иницилируемых AK_{ij}), в котором осуществляется последовательная передача управления на выполнение AL_{ij} (обслуживание AK_{ij}). Каждая AK_{ij} завершается оператором Mt_{ij} , который вычисляет новый момент t_{ij} инициализации AK_{ij} и возвращает управление на продолжение просмотра второго цикла. Может оказаться, что при выполнении некоторых AK_{ij} , инициализированных на втором цикле, вновь устанавливаются соответствующие значения параметров и переменных, указывающие на то, что выполнены условия инициализации некоторых AK_{ij} . Поэтому управление передается на начало первого цикла для повторного просмотра выполнимости условий инициализации AK_{ij} . Только в случае, когда после первого цикла список иницилируемых AK_{ij} пуст, УПМ корректирует t_0 минимальным значением момента инициализации активности t_{ij} . Все t_{ij} заносятся в упорядоченный по их значениям список моментов инициализации AK_{ij} . Когда завершается обслуживание некоторой AK_{ij} , в этот список заносятся значение t_{ij} и имя соответствующей AK_{ij} . После корректировки t_0 проверяется выполнимость условия завершения имитации. Если это условие не выполнено, то имитация продолжается и управление передается на начало первого цикла. Когда условие завершения имитации выполнено, имитация прекращается и управление передается соответствующей подпрограмме, которая формирует результаты моделирования.

Поскольку выполнение алгоритмов одних активностей $АЛ_{ij}$ может привести к инициализации других $АК_{ij}$, то в УПМ возможны повторные циклы проверки выполнимости условий инициализации $АК_{ij}$. При этом события $С_{ij}$ в ИМ не регламентированы, а лишь указываются условия, при которых они могут произойти. Отметим, что инициализация $АК_{ij}$ и сдвиг ее временной координаты оператором $Мt_{ij}$ разрешаются только тогда, когда выполняются все условия ее инициализации. Если хотя бы одно из этих условий остается невыполненным, то соответствующая $АК_{ij}$ не попадает в список инициализируемых активностей.

Иногда удельный вес безуспешных проверок очень велик УПМ. Данный способ организации квазипараллелизма $АК_{ij}$ выгоден только при наличии достаточно простых алгоритмов проверки выполнимости условий инициализации активностей.

Более подробно организация имитации множеством активностей приведена в работе [38] при моделировании производственного участка работ.

7.4. Организация квазипараллелизма в имитационных моделях способом составления расписания событий

Данный способ организации квазипараллелизма используется при моделировании реальных систем, характеризующихся следующим. Различные компоненты K_i выполняют одни и те же функциональные действия $ФД_{ij}$. Начало выполнения этих $ФД_{ij}$ определяется одними и теми же условиями, которые также заранее известны исследователю и могут быть представлены алгоритмически. В результате выполнения одних и тех же $ФД_{ij}$ в системе происходят одинаковые события $С_{ij}$ независимо друг от друга. Связи между различными $ФД_{ij}$ отсутствуют, каждое $ФД_{ij}$ выполняется независимо друг от друга.

Пусть в нашем примере (рисунок 7.4) у компонент системы совпадают следующие $ФД_{ij}$, приводя к одним и тем же событиям:

$ФД_{11}, ФД_{12}, ФД_{13}$ – к событию $С_{11}$;

$ФД_{21}, ФД_{22}, ФД_{23}$ – к событию $С_{12}$;

$ФД_{31}, ФД_{32}, ФД_{33}$ – к событию $С_{13}$;

$ФД_{14}$ – к событию $С_{14}$.

В таких случаях исследователь также описывает ИМ в виде двух частей: множества активностей $\{АК_{ij}\}$ и набора процедур проверки появления событий и инициализации соответствующих активностей. При этом каждая $АК_{ij}$ имитирует выполнение группы совпавших

функциональных действий ΦD_{ij} у различных компонент K_i системы. В нашем случае AK_{11} инициализируется при появлении события C_{11} и имитирует выполнение ΦD_{11} , ΦD_{12} , ΦD_{13} . Аналогичным образом иницируются: AK_{12} при появлении события C_{12} , AK_{13} при появлении события C_{13} и AK_{14} при появлении события C_{14} . Инициализация AK_{ij} имеет тот же смысл, что и для предыдущего способа организации квазипараллелизма, и выполняется УПМ.

В процедурах проверки появления событий C_{ij} реализуется зависимость выполнения ΦD_{ij} от конкретной ситуации, имеющей место в реальной системе. Выполнение алгоритма AL_{ij} также называется обслуживанием AK_{ij} , которое завершается оператором модификации временной координаты Mt_{ij} активности, обслуживающей группу одинаковых ΦD_{ij} , у разных компонент реальной системы. По этому оператору управление возвращается на УПМ. Часто такие групповые активности называют процедурами обслуживания событий C_{ij} . Таким образом, ИМ состоит из двух типов процедур: проверки выполнимости событий C_{ij} . Таким образом, ИМ состоит из двух типов процедур: проверки выполнимости событий C_{ij} в модели системы и обслуживания события C_{ij} .

Выполнение этих процедур синхронизируется в модельном времени списковым механизмом планирования УПМ. Каждый элемент этого списка определяет момент t_{ij} свершения события C_{ij} , а также имя или номер той процедуры обслуживания событий, которая должна выполняться после завершения этого события.

Схема взаимодействия процедур, составляющих описание ИМ с УПМ представлена на рисунке 7.6. Перед началом имитации соответствующая подпрограмма устанавливает начальные состояния компонент модели системы и задает исходные значения параметрам, определяющим условия инициализации групповых активностей AK_{ij} . Принципы проверки выполнимости условий появления в ИМ событий C_{ij} те же, что и при имитации активностями. Здесь также с помощью соответствующих алгоритмов пользователь составляет описание процедур проверки условий появления событий C_{ij} .

Алгоритм инициализации УПМ процедур обслуживания событий состоит в следующем. По первому циклу (параметр цикла равен максимальному числу различных событий C_{ij} в модели системы) управление последовательно передается на проверку выполнимости условий появления событий. Имена тех событий, для которых эти условия выполнены, заносятся в список инициализируемых событий. По завершении этого цикла проверяется, имеется ли в этом списке хотя бы один элемент. Если список событий не пустой, то начинается второй цикл (параметр цикла равен числу свершившихся событий C_{ij}),

в котором осуществляется последовательная передача управления на выполнение процедур обслуживания событий. Каждая такая процедура реализует алгоритм соответствующей групповой активности и завершается оператором модификации временной координаты Mt_{ij} , который модифицирует значение момента свершения данного события в будущем и возвращает управление на продолжение второго цикла.

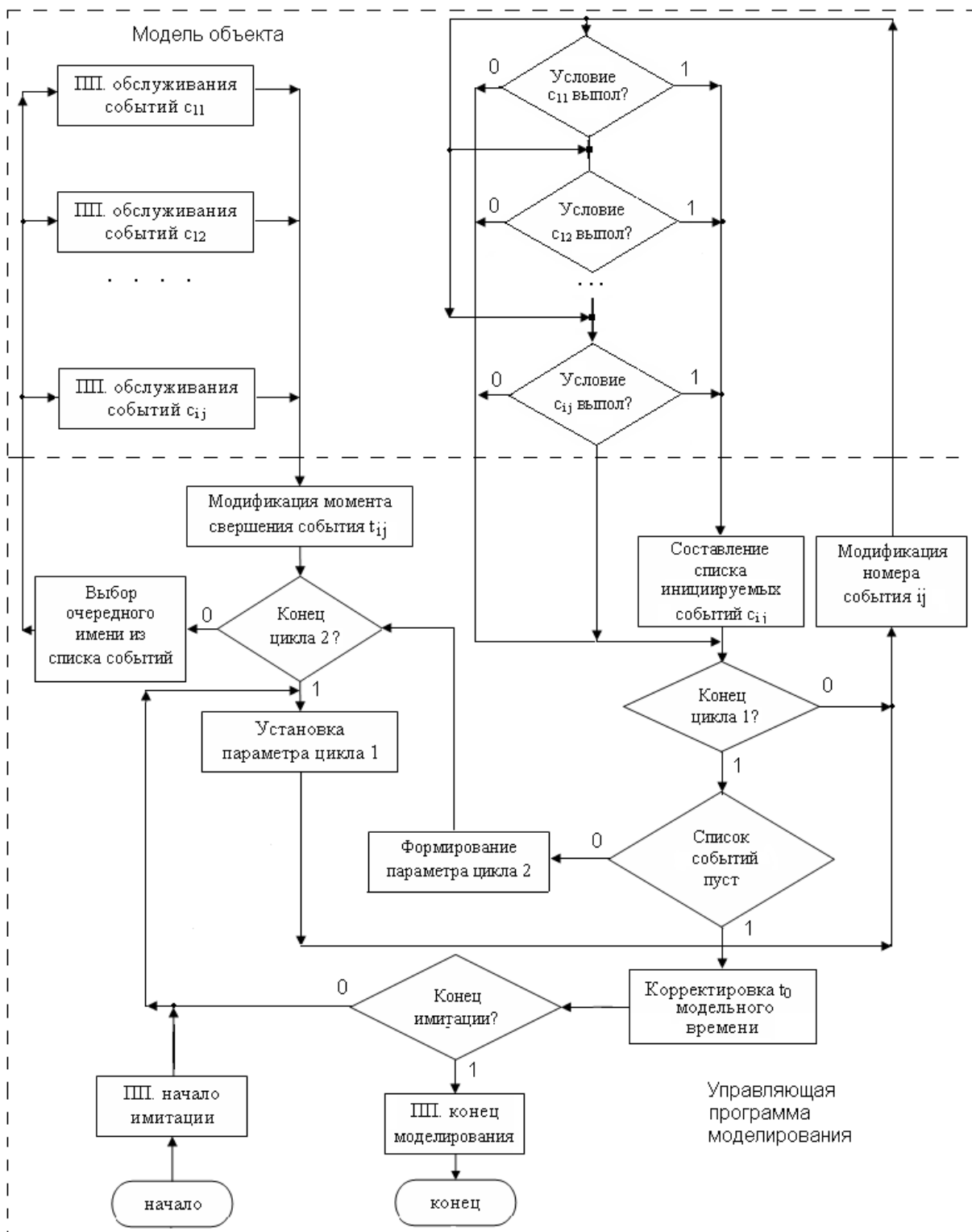


Рисунок 7.6 – Схема имитации способом обслуживания событий

Для того чтобы учесть появление новых событий во время обслуживания списка свершившихся событий, по окончании цикла 2 устанавливается начальное значение параметра цикла 1 и повторяется просмотр условий свершения событий. Как только список свершившихся событий окажется пустым, УПМ корректирует модельное время. Корректировка осуществляется подпрограммой-календарем, которая ищет в списке запланированных событий минимальное значение момента инициализации t_{ij} , которое становится новым значением модельного времени t_0 . После корректировки t_0 проверяется выполнимость условия завершения имитации. При невыполнении этого условия имитация продолжается, и управление передается на установку параметра цикла 1. Когда условие завершения имитации выполнено, она прекращается, и управление передается соответствующей подпрограмме, которая вычисляет результаты моделирования и выдает их исследователю.

Из приведенного примера видно, что объединение нескольких активностей в группы существенно сокращает размеры циклов 1 и 2, что значительно уменьшает накладные расходы на организацию имитации УПМ. Поэтому для некоторых случаев (условия применения способа оговорены в начале данного параграфа) данный способ имитации более экономичен, чем способ имитации активностями. Однако у рассматриваемого способа имеется существенный недостаток, который заключается в следующем. Из-за того, что происходит объединение активностей различных компонент в составе процедур обслуживания событий, описание ИМ может потерять сходство со структурой реальной системы. Зачастую в рамках одной процедуры обслуживания событий ИМ могут обслуживаться даже логически не связанные друг с другом активности, но приводящие к одним и тем же событиям. Это обстоятельство может затруднить анализ результатов моделирования и модификацию ИМ.

В работе [38] можно подробнее ознакомиться с алгоритмом имитации событийным способом при моделировании производственного участка работ.

7.5. Транзактный способ организации квазипараллелизма

В ряде случаев функциональные действия $\Phi Д_{ij}$ представляют собой набор простейших операций, и можно аппроксимировать

активностями, алгоритмы выполнения которых лишь корректируют значения временных координат t_i компонент системы. Важно и то, что существует зависимость выполнения ΦD_{ij} друг от друга, которую удобно представить в виде схемы. Взаимодействие такого вида активностей аналогично работе систем массового обслуживания (СМО) [60]. Для имитации поведения реальной системы в таких случаях используется транзактный способ организации квазипараллелизма в ИМ. Однотипные активности пользователем объединяются и называются приборами массового обслуживания [60]. Инициаторами появления событий S_{ij} в ИМ становятся заявки (транзакты) на обслуживание этими приборами массового обслуживания. Связь между обслуживающими приборами устанавливается с помощью системы очередей, выбранных дисциплин поступления и способов извлечения из них транзактов.

Пусть в нашем примере функциональные действия ΦD_{12} , ΦD_{22} , ΦD_{23} совпадают. Кроме того, в любой данный момент выполняется только одно из них, и исследователя интересует влияние этих ΦD_{ij} на поведение всей системы. Взаимосвязь между ΦD_{ij} исследуемой системы можно представить в виде схемы. Программа-модель создается в два этапа. Вначале ИМ представляется в виде схемы, отображающей рождение транзактов, их пространственное перемещение по схеме и уничтожение уже обслуженных транзактов. Для описания ИМ создается достаточно широкий, но с фиксированным набором стандартных блоков обслуживания транзактов. С их помощью представляются действия по образованию и исключению транзактов из модели, управлению движением транзактов, занятию и освобождению различных типов ресурсов системы, имитации задержек в продвижении транзактов. В общей сложности используется несколько десятков графически различимых блоков, позволяющих в наглядной форме представлять алгоритмы функционирования сложных систем с дискретными событиями. На втором этапе осуществляется символическое кодирование схемы, при котором каждому блоку ставится в соответствие определенный оператор соответствующего языка моделирования, образуя текст программы-модели. Программа-модель в этом случае реализуется в режиме интерпретации. На рисунке 7.7 изображена схема взаимодействия УПМ с моделью объекта при транзактном способе имитации реальной системы, представленной на рисунке 7.4.

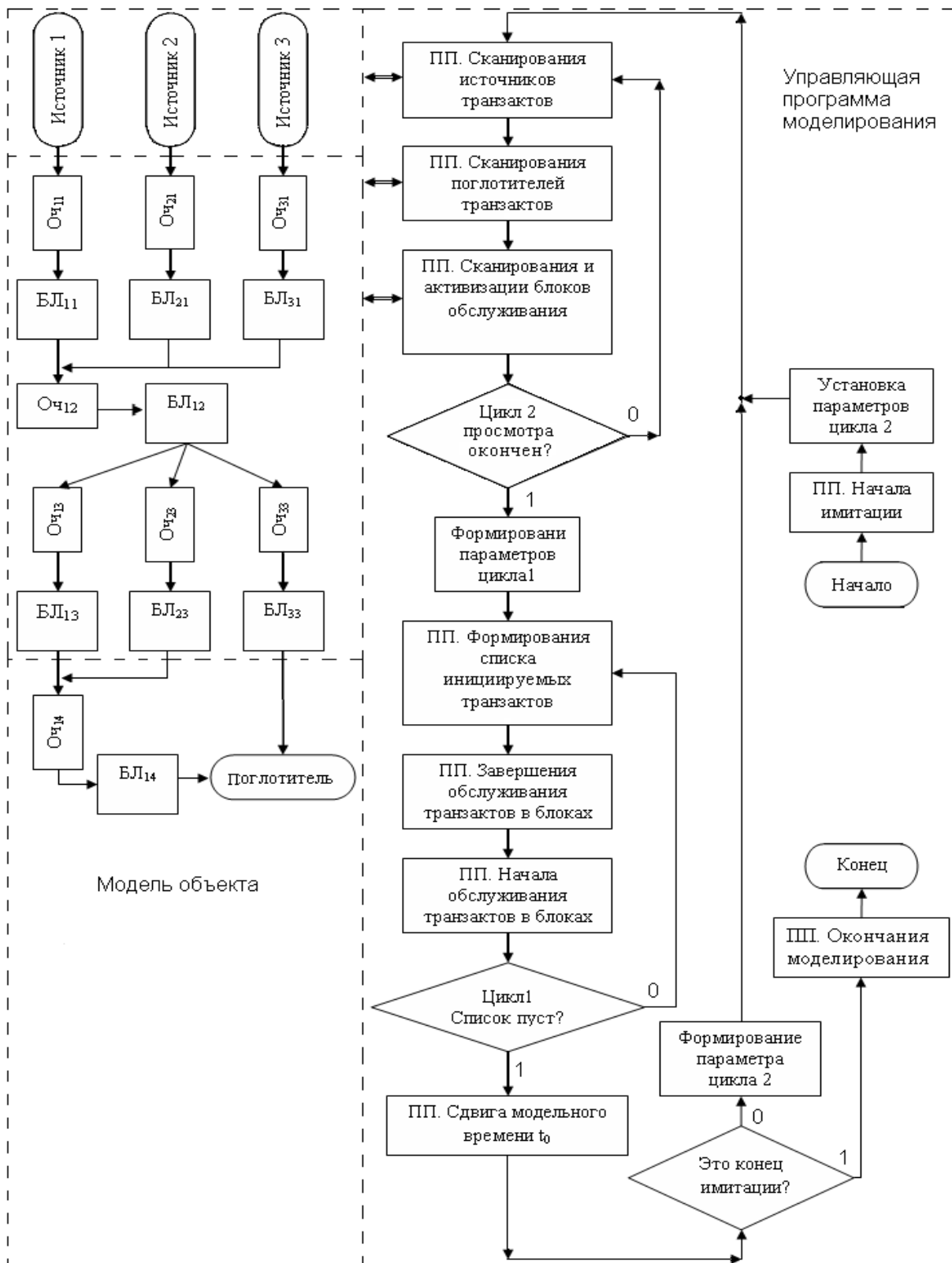


Рисунок 7.7 – Схема организации транзактного способа имитации СС

В источниках (ИСТ_{*j*}) транзакты создаются, а в блоке ПОГЛОТИТЕЛЬ все добравшиеся до него транзакты уничтожаются. УПМ просматривает списки источников транзактов (сканирует) и создает

новые транзакты с заданной интенсивностью их поступления в систему. Таким образом, все поступающие в систему транзакты имитируют собой внешнее окружение реальной системы. Событием в ИМ, построенной по транзактному способу имитации, является момент инициализации любого транзакта. В результате транзакт в такой ИМ выступает в роли активности. УПМ назначает транзактам новые моменты инициализации t_{ij} и до наступления этих моментов закрепляет за транзактами соответствующие блоки обслуживания (БЛ $_{ij}$). Алгоритм работы УПМ состоит в следующем. ПП сканирования источников просматривает условия создания новых транзактов каждым из источников и передает управление ПП создания транзактов при совпадении этих условий. ПП создания транзактов формирует новые транзакты и помещает их в очереди к блокам, в нашем примере к БЛ $_{11}$, БЛ $_{21}$ и БЛ $_{31}$. После создания всех транзактов, для которых выполнены условия их создания, УПМ приступает к просмотру поглотителей. ПП сканирования поглотителей просматривает очереди транзактов, завершивших свое пребывание в модели системы, и передает управление ПП уничтожения транзактов. Последняя при каждом уничтожении транзакта формирует статистику пребывания транзакта в модели системы. Когда просмотрены все поглотители ИМ, управление передается ПП формирования списка инициализируемых транзактов.

Под инициализацией транзакта понимают завершение пребывания транзакта в каком-либо блоке модели или поступление транзакта в соответствующие очереди к блокам модели системы. ПП формирования списка инициализируемых транзактов просматривает списки транзактов (очереди), поступивших на входы к блокам БЛ $_{ij}$, и выбирает из них те, у которых время инициализации совпадает с t_0 , образуя список активизируемых транзактов. По окончании просмотра всех блоков УПМ проверяет условие «Список инициализируемых транзактов пуст». Если в этом списке имеется хотя бы один транзакт, управление передается на ПП завершения обслуживания транзактов. Эта подпрограмма выбирает адрес блока, на котором завершено обслуживание транзакта, определяет имя очереди (ОЧ $_{ij}$) к блоку БЛ $_{ij}$, завершившему обслуживание, выбирает из очереди ОЧ $_{ij}$ новый транзакт и закрепляет за ним освободившийся транзакт. При этом формируется момент будущей инициализации транзакта $t_{ij} = t_0 + \tau_{ij}$ (здесь τ_{ij} означает длительность выполнения ФД $_{ij}$, которое имитируется обслуживанием транзактов блоком БЛ $_{ij}$, а t_0 – модельное время). Затем управление передается на ПП начала обслуживания транзактов. Эта подпрограмма определяет адрес блока и имя очереди, на которую должен поступить только что обслуженный транзакт, и организует его

перемещение по схеме модели. Если блок, на который поступает транзакт, свободен, то он закрепляется за данным транзактом на время, указанное в описании этого БЛ_{ij}. При этом выполняются те же действия по формированию момента t_{ij} инициализации транзакта, что и в ПП завершения обслуживания транзактов.

Как только список активизируемых транзактов оказывается пустым, ПП сдвига модельного времени выбирает из списка моментов инициализации транзактов $\{t_{ij}\}$ минимальное значение, которое становится новым модельным временем t_0 . Просматриваются в этом списке все те транзакты, у которых моменты инициализации совпадают, и формируется начальный список инициализируемых транзактов. Затем проверяется выполнение условий окончания имитации. Если условия не выполнены, то управление передается на ПП сканирования источников. При выполнении условий создания транзактов дополняется список инициализируемых транзактов, поскольку считается, что в моменты создания транзактов они также инициализируются, и алгоритм имитации УПМ повторяется. При выполнении условий окончания моделирования процесс имитации завершается – и управление передается ПП окончания моделирования, которая выдает пользователю результаты моделирования. Таким образом, УПМ циклически сканирует различные списки транзактов с целью инициализации и организации обслуживания транзактов соответствующими блоками, сдвигает модельное время и проверяет выполнимость условий окончания имитации.

Как видно из алгоритма функционирования УПМ, за внешнюю простоту описания ИМ приходится платить достаточно большими накладными расходами на организацию квазипараллелизма. Эти дополнительные затраты машинного времени t_s значительно возрастают при увеличении самой модели, поскольку возрастает количество списков транзактов и блоков, которые необходимо несколько раз просматривать при организации квазипараллельной работы транзактов с помощью УПМ.

Подробнее с алгоритмом имитации СС транзактным способом можно познакомиться в работе [38] при моделировании производственного участка работ.

7.6. Агрегатный способ организации квазипараллелизма

В некоторых реальных системах имеет место тесное взаимодействие между функциональными действиями компонент системы.

Компоненты системы обмениваются между собой сигналами. Каждый выходной сигнал от одной компоненты K_i является входным сигналом для другой компоненты системы. Если при этом сами функциональные действия ΦD_{ij} аппроксимируются явно задаваемыми математическими зависимостями, позволяющими определять момент появления выходных сигналов компонент K_i при наличии входных сигналов, поступающих от других компонент, то создаются подходящие условия для построения ИМ по модульному принципу. В этом случае каждый из модулей ИМ строится по унифицированной структуре и называется агрегатом.

Агрегат является математической схемой, с помощью которой возможно описание достаточно большого круга реальных процессов. [5]. В любой момент t_{ij} агрегат может находиться в одном из возможных состояний. Состояния агрегата $z(t)$ называют фазовыми траекториями. Переход агрегата из состояния в состояние описывается с помощью некоторого оператора перехода H , который позволяет по предыдущему определить очередное состояние агрегата. Агрегат имеет входы, куда поступают входные сигналы $X_l(t)$ от других агрегатов, и выходы, на которых формируются выходные сигналы $Y_r(t)$. Здесь t означает номера соответственно входных и выходных сигналов. Кроме того, у агрегата имеются дополнительные входы, на которые поступают управляющие сигналы $g(t)$. Выходные сигналы $Y_r(t)$ формируются из выходных $X_l(t)$ и управляющих $g(t)$ сигналов оператором выхода G в результате его взаимодействия с оператором H . Значения операторов H и G задаются исследователем при аппроксимации агрегатами выполнения ΦD_{ij} реальной системы.

Заметим, что квазипараллельная работа агрегатных систем может быть реализована различными способами (активностями, планирование событий, взаимодействием транзактов, процессами). Поэтому выделение агрегатного способа организации квазипараллелизма в ИМ является условным и обуславливается наличием ряда специфических функций, которые должна выполнять УПМ для реализации такой универсальной математической модели. Среди этих функций необходимо выделить проверку условий перехода агрегата в особые состояния и формирование выходных сигналов, изменение координат состояний агрегата, передачу сигналов от одного агрегата к другому. Моделирование поведения агрегата представляет собой последовательную цепь переходов из одного особого состояния в другое. Когда законы поступления управляющих и выходных сигналов заданы, то в функции УПМ входит проверка условий перехода агрегата в особые состояния и возможность выработки выходных сигналов. Если входные

сигналы не могут быть определены до момента их имитации, УПМ предусматривает процедуры проверки факта поступления внешних сигналов за некоторый интервал модельного времени Δt_0 . УПМ анализирует возможность моделирования выходных сигналов у агрегата.

Частным случаем агрегата является кусочно-линейный агрегат (КЛА), у которого отсутствуют управляющие сигналы. Обычно КЛА представляется в виде многополюсника, у которого имеется N входных и m выходных полюсов. На входные полюса поступают элементарные сигналы X_l ($l = \overline{1, N}$). Аналогичную структуру имеют выходные сигналы Y_r ($r = \overline{1, m}$). Связи между выходами одних агрегатов и входами других называются каналами. Существует следующее ограничение: по каждому каналу может передаваться только один сигнал, который является входным для одного агрегата и выходным для другого. Подробности формализации реальных объектов с помощью агрегативного представления рассмотрены в работе [38]. Ниже рассматриваются особенности организации квазипараллельного обслуживания агрегатов УПМ для случая, когда каждое функциональное действие компонент реальной системы ΦD_{ij} аппроксимируется в ИМ отдельным КЛА. Конечно, привязка отдельных ΦD_{ij} к различным агрегатам не всегда целесообразна и рассматривается нами только для простоты изложения специфики организации УПМ квазипараллельной работы агрегатов.

На рисунке 7.8 представлена схема взаимодействия УПМ с моделью объекта, состоящей из КЛА. Каждый КЛА аппроксимирует соответствующее ΦD_{ij} реальной системы, представленной на рисунке 7.4. Агрегативная система является замкнутой, и возникновение выходных сигналов в агрегате A_0 возможно лишь вследствие возникновения внутренних событий. Поэтому A_0 представляет собой модель поведения внешней среды, которая воздействует на исследуемую систему. Каждому ΦD_{ij} в ИМ соответствует свой агрегат A_{ij} . Процесс эволюции состояний в агрегативной ИМ и появление C_{ij} обеспечивает УПМ. Допустим, на выходе A_{32} появляется выходной сигнал Y_{32} . Производится пассивизация агрегата A_{32} , назначение момента будущей активизации агрегата A_{32} и формирование списка выходных сигналов. Под пассивизацией агрегата будем понимать занесение его в список агрегатов, ожидающих выполнения либо временных, либо логических условий, согласно которым УПМ возвращается к обслуживанию агрегата. Момент возврата УПМ к обслуживанию агрегата назовем активизацией агрегата. Далее проверяется наличие у агрегата A_{32} выходных сигналов.

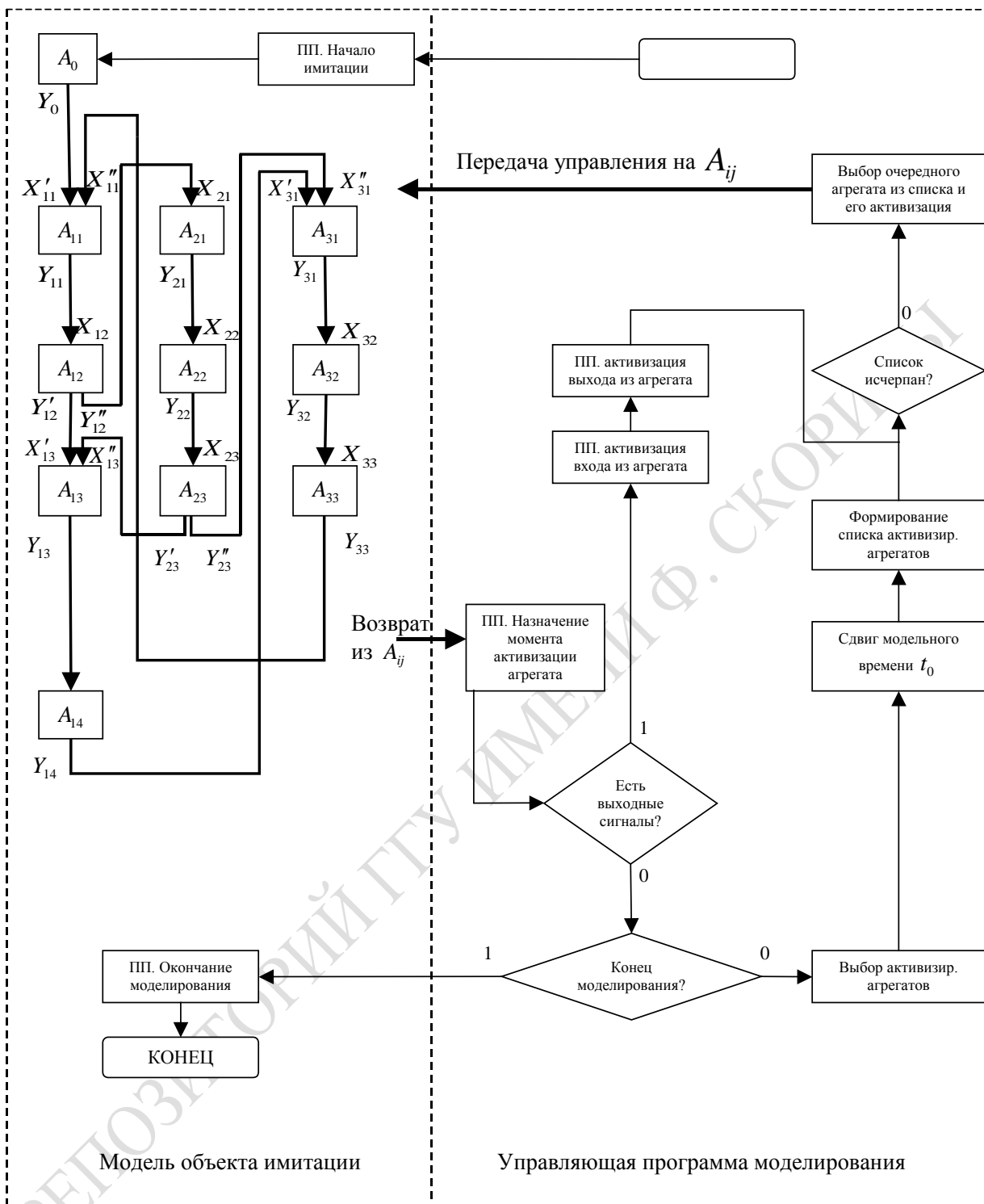


Рисунок 7.8 – Схема организации агрегатного способа имитации СС

Для того чтобы УПМ могла организовать взаимосвязь между агрегатами с помощью сигналов, пользователь при формировании реальной системы составляет матрицу коммутации, указывающую коммутацию агрегатов соответствующими каналами. Более подробно структура матрицы коммутации и правила ее составления приведены в работе [38]. При наличии выходного сигнала Y_{32} по матрице коммутации

агрегатов УПМ определяет имя входного сигнала (в данном случае это будет X_{33}) и организует активизацию агрегата A_{33} .

Когда все выходные сигналы Y_{ij} агрегатов A_{ij} обслужены, проверяется условие окончания моделирования, и при его невыполнении управление передается на подпрограмму активизации агрегатов по времени имитации. Это означает выбор минимального значения в упорядоченном списке моментов времени изменения состояний агрегатов t_{ij} .

Для простоты предполагаем, что все A_{ij} обладают только одной координатой изменения состояний и оператор переходов соответствует простейшему алгоритму вычисления момента $t_{ij} = t_0 + \tau_{ij}$ появления следующего события C_{ij} . При выборе минимального значения t_{ij} определяется точка входа (адрес) или имя того агрегата, который в данный момент должен быть активизирован. Это имя агрегата становится начальным значением списка активизируемых по времени агрегатов. Одновременно с этим происходит сдвиг модельного времени ($t_0 = \min t_{ij}$).

Затем УПМ формирует список активизируемых агрегатов, т. е. тех агрегатов, у которых момент активизации совпадает с новым значением t_0 . По окончании формирования списка активизируемых агрегатов УПМ приступает к последовательной активизации алгоритма тех A_{ij} , которые попали в этот список. Активизация любого A_{ij} состоит в передаче управления на A_{ij} и реализации последними операторов H и G , в результате чего агрегаты переходят в новые состояния $z(t)$ и формируются выходные сигналы. Далее процесс обслуживания УПМ агрегатов аналогичен ранее рассмотренному.

Итак, обобщая рассмотренный пример, согласно [48] можно выделить пять шагов эволюции агрегативной системы, состоящей из k агрегатов и УПМ:

- начальная установка состояний агрегатов;
- определение момента времени, в который произойдет следующее событие в агрегатах, и номера агрегата, в котором произойдет это событие;
- определение нового состояния агрегата, формирование множества выходных сигналов в результате свершения внутренних событий и фиксация новых состояний всех агрегатов;
- проверка множества выходных сигналов и, если необходимо, по соответствующей матрице коммутаций, определение номеров каналов, по которым должен быть передан выходной сигнал, при этом определяется также номер входного полюса, на который поступает входной сигнал;
- определение нового состояния агрегата в результате прихода входного сигнала и формирование выходного сигнала.

Описание поведения каждого агрегата реализуется с помощью соответствующего языка моделирования [5, 6, 48]. При этом исследователь указывает способ выдачи и характеристики операторов перехода H и операторов выдачи выходных сигналов G , составляет матрицы коммутации полюсов агрегатов. Далее система моделирования, реализующая соответствующий язык моделирования, обеспечивает перевод описаний в программные модули, а УПМ уже стандартным образом организует собственно имитацию событий в агрегативной системе, сбор статистики и управление ходом имитации. По выполнению условий окончания моделирования управление передается подпрограмме окончания моделирования, которая и выдает исследователю результаты моделирования в принятой для данной системы моделирования стандартной форме.

Как видно из приведенного примера, агрегатный способ организации квазипараллелизма в ИМ является достаточно удобным с точки зрения описания сложной системы. Однако необходимость коммутации и обслуживания сигналов требует дополнительных расходов ресурсов машинного времени t_3 . За удобство описания и математическую строгость модели приходится расплачиваться дополнительным расходом t_3 , и это обстоятельство является одним из факторов, сдерживающих использование агрегативного способа организации квазипараллелизма при моделировании сложных систем.

7.7. Процессный способ организации квазипараллелизма

При моделировании сложных систем исследователи зачастую встречаются со следующими ограничениями. Все функциональные действия $ФД_{ij}$ компонент реальной системы различны. Условия появления событий $С_{ij}$, приводящие к выполнению $ФД_{ij}$, также индивидуальны. У каждой компоненты K_i существует определенная последовательность выполнения $ФД_{ij}$. В любой момент времени в данной компоненте может выполняться только одно $ФД_{ij}$. Перечисленные ограничения определяют выбор исследователем процессного способа организации квазипараллелизма в ИМ. При процессном подходе краткость описания активностей объединяется с эффективностью событийного представления имитации.

Процессным способом можно организовать имитацию любых сложных систем. Однако процессный способ имитации особенно эффективен в тех случаях, когда требуется высокий уровень детализации

выполнения $\Phi_{D_{ij}}$ при аппроксимации с помощью AL_{ij} и сама ИМ используется для поиска узких мест в системе. При такой постановке задачи на моделирование очень важно соблюдение сходства структуры модели и объекта исследования, что обеспечивается процессным способом имитации. В нашем примере (рисунок 7.4) реальную систему удобнее всего имитировать процессным способом, когда в один и тот же момент времени не может реализоваться более одного $\Phi_{D_{ij}}$ данной i -й компоненты реальной системы. Тогда исследователю удобнее рассматривать функционирование компонент модели K_i как единое целое. Всю имитационную модель можно представить в виде набора описаний процессов, каждое из которых описывает один класс процессов, например компоненты K_i для нашего случая. При этом могут иметь место информационные и управляющие связи не только между компонентами K_i , но даже и между отдельными алгоритмами их функционирования.

На рисунке 7.9 представлена схема взаимодействия УПМ с моделью объекта, состоящей из описаний процессов. Алгоритм функционирования ИМ представляется последовательным взаимодействием процессов и УПМ. Причем в процессы объединяются связанные между собой активности, которые определяют функционирование одной и той же компоненты модели K_i . Таким образом, имеет место полное соответствие компонент реальной системы и ее ИМ. Каждой компоненте объекта моделирования соответствует свой процесс. Переход от выполнения одной активности к другой активности того же процесса считают изменением его состояния и называют активизацией процесса. Обычно под состоянием процесса понимают номер j -й активности, которая входит в состав i -го процесса и на которую УПМ передает управление при свершении события C_{ij} в K_i . Следовательно, изменения состояний реальной системы соответствуют изменениям соответствующих состояний процессов и появлению событий C_{ij} .

Вся ИМ представляет собой набор процессов, реализованных на соответствующем языке моделирования. Процессы связаны с УПМ с помощью некоторых операторов этого языка, по которым имеет место обращение к УПМ при завершении активности данного процесса, означающее переход процесса в другое состояние. Отметим при этом, что проверка выполнимости условий активизации процесса и появление событий C_{ij} осуществляются самим процессом. Будучи активизированным, выполнение процесса (некоторой его активности) может начаться немедленно или задержаться до появления определенных условий или до изменения состояний других процессов. Процессы могут переходить в новые состояния как по своей инициативе, так и в результате действий, выполняемых активностями других процессов.

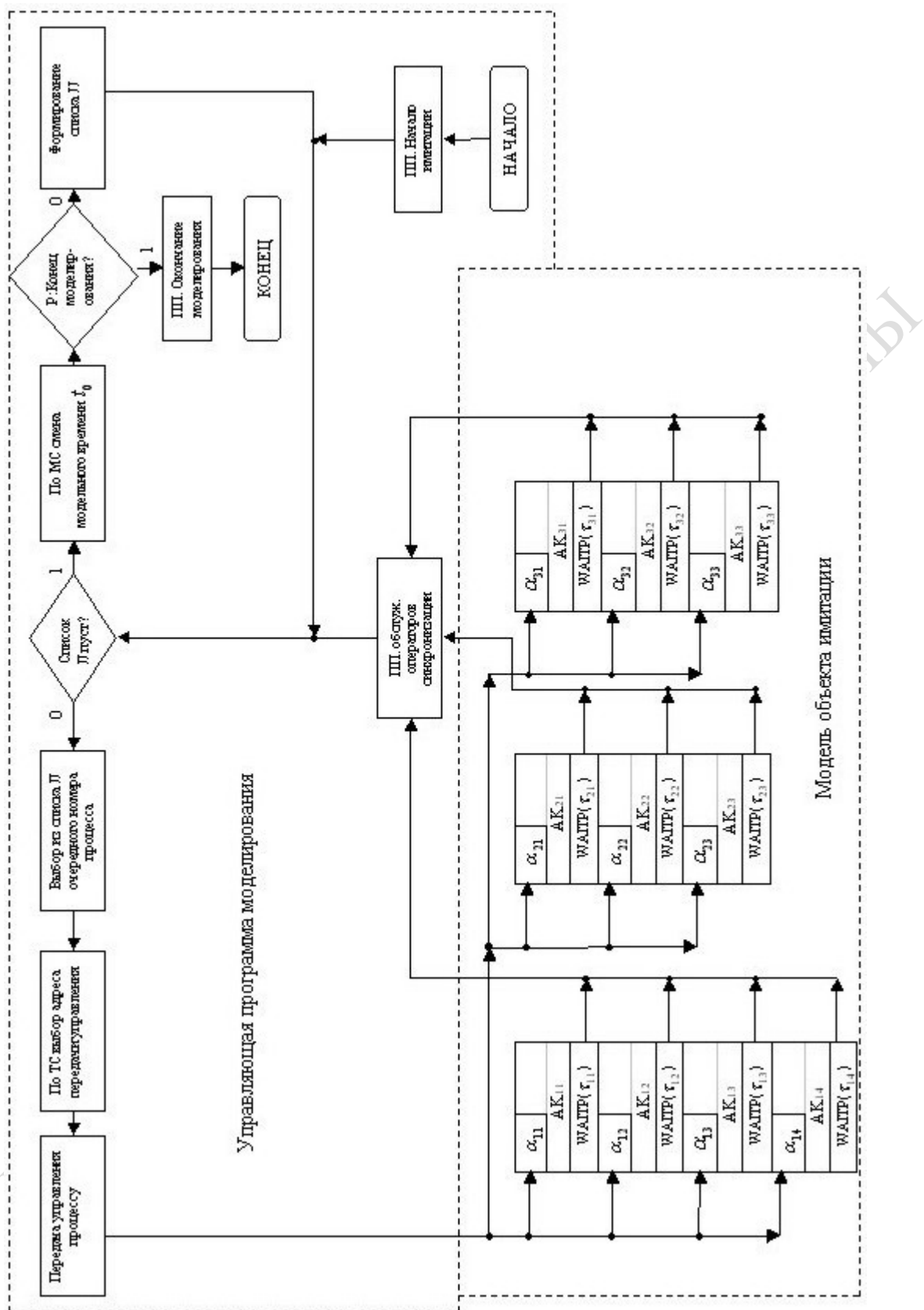


Рисунок 7.9 – Схема организации процессного способа имитации СС

В качестве примера реализации квазипараллелизма на основе процессного способа выберем алгоритм функционирования УПМ, реализованный в моделирующем комплексе (МК) PLSIM [29]. Для простоты изложения опустим некоторые детали функционирования этого алгоритма и будем считать, что взаимодействия между активностями, из которых состоят процессы, нет и все активности обращаются к УПМ с помощью операторов синхронизации WAIT (τ_{ij}). Каждый такой оператор означает, что данному i -му процессу по окончании выполнения алгоритма АЛ $_{ij}$ активности АК $_{ij}$ назначается момент следующей активизации t_{ij} по окончании ожидания процесса в модельном времени длительностью τ_{ij} . Началу выполнения АЛ $_{ij}$ процессов соответствуют адреса α_{ij} в подпрограммах, реализующих выполнение активностей, объединенных в процессы. УПМ работает с массивом состояний процессов (МС) и таблицей состояний процессов (ТС). В МС каждый элемент представляет собой пару значений (i, t_{ij}), где i – номер процесса, t_{ij} – момент j -й активизации процесса в будущем и появления события S_{ij} . Для выбора процессов, которые необходимо активизировать в момент t_0 , УПМ использует ТС. Строками этой таблицы являются списки параметров процессов.

Так, i -му процессу соответствует строка, в которой указаны следующие параметры: α_{ij} – адрес передачи управления на выполнение j -й активности в i -м процессе; $t_{\text{нди}}$ – время, которое осталось i -му процессу находиться в состоянии ожидания, когда он останавливается другим процессом; π_i – приоритет i -го процесса, согласно которому осуществляется последовательное обслуживание УПМ двух одновременно активизируемых процессов.

Все элементы МС упорядочены по возрастанию значения t_{ij} . Если какой-либо i -й процесс остановлен (хотя бы даже для обслуживания его УПМ в моменты активизации), то время последующей его активизации в МС бесконечно велико ($t_{ij} = \infty$) и соответствующий ему элемент находится в конце этого массива.

Определено понятие «конфликтная ситуация в системе дискретных событий»: несколько событий S_{ij} в различных процессах происходят одновременно и требуют немедленного обслуживания со стороны УПМ. Такая ситуация имеет место, например, при наличии в МС нескольких элементов с одним и тем же значением t_{ij} . Учет конфликтных ситуаций производится с помощью списка одновременно активизируемых процессов (список JJ). В списке JJ находятся номера всех процессов, которые необходимо активизировать в одно и то же модельное время t_0 . Запись номеров процессов i в список JJ и выбор их из этого списка УПМ осуществляет согласно приоритетам

процессов π_i . Обслуживание УПМ очередного элемента списка JJ состоит в передаче управления на выполнение алгоритма $АЛ_{ij}$ i -го процесса. Выполнение этого алгоритма продолжается до появления в нем очередного оператора синхронизации процесса WAIT (τ_{ij}). С помощью операторов ожидания типа WAIT (τ_{ij}) имитируется изменение временных координат активностей, входящих в состав i -го процесса.

В нашем примере (рисунок 7.4) реальный объект состоит из трех компонент, и поэтому его ИМ состоит также из трех процессов ($i = 1, 2, 3$). Пусть задано время окончания моделирования $T_{ок}$, а условием окончания моделирования является достижение модельным временем t_0 значения $T_{ок}$. УПМ обслуживает процессы согласно алгоритму, представленному на рисунке 7.9. Пусть в начальный момент моделирования $t_0 = m_0$ (рисунок 7.9) список активизации процесса JJ пуст и процессам назначены соответственно моменты активизации t_{11}, t_{21}, t_{31} в МС.

Шаг 1. Поскольку список JJ пуст, то осуществляется процедура смены t_0 , представляющая собой выбор минимального значения из МС ($t_0 = \min \{t_{ij}\}$).

Шаг 2. Формируется список JJ следующим образом. Все процессы, у которых $t_{ij} \leq t_0$, выбираются из МС и в список JJ заносятся согласно приоритетам процессов π_i . Приоритеты процессов определяются по таблице ТС.

Шаг 3. Выбирается первый элемент списка JJ, который необходимо в момент $t_{ij} = t_0$ активизировать. По ТС определяется адрес α_{ij} , и управление от УПМ передается на выполнение алгоритма соответствующей активности. По окончании выполнения алгоритма активности появляется оператор ожидания i -го процесса до следующей активизации WAIT (τ_{ij}). В ходе реализации алгоритма активности $АК_{ij}$ происходит вычисление или задание значения τ_{ij} для операторов синхронизации процессов. Появления в алгоритме i -го процесса операторов синхронизации WAIT (τ_{ij}) возвращают управление УПМ, которая выполняет в дальнейшем следующие действия:

- формирование нового элемента МС и занесение его в этот массив согласно значению $t_{ij} = t_0 + \tau_{ij}$;

- модификацию текущего состояния процесса и изменение при этом адреса последующей передачи управления α_{ij} для очередной активизации процесса i в ТС.

Шаг 4. Проверяется, исчерпан ли список JJ. Если в списке JJ есть еще элементы, то УПМ переходит к шагу 2. Иначе выполняется шаг 5.

Шаг 5. Проверяется момент окончания моделирования ($t_0 \geq T_{\text{ок}}$). Если это неравенство не выполняется, то это означает необходимость продолжения имитации, начиная с шага 1. В противном случае управление передается программе окончания моделирования.

Продолжим рассмотрение приведенного примера, используя временную диаграмму перехода компонент реальной системы (процессов модели системы) из состояния в состояние (рисунок 7.2) и схему взаимодействия УПМ с процессами (рисунок 7.9). Итак, первоначально в момент $t_0 = m_0$ в списке JJ находятся все три процесса, которые расположены соответственно их приоритетам, например $\pi_1 > \pi_2 > \pi_3$. Допустим, что время окончания моделирования $T_{\text{ок}}$ равно моменту m_8 на оси изменения модельного времени t_0 (рисунок 7.2).

Управляющая программа моделирования вначале передает управление по адресу α_{11} . Выполнение алгоритма активности АК₁₁ завершается появлением оператора WAIT (τ_{11}), и управление передается на ПП обслуживания операторов синхронизации. Эта подпрограмма выполняет следующие действия:

- формирует новый адрес продолжения процесса 1 в ТС (α_{12}); назначает момент будущей активизации процесса 1 ($t_{11} = m_0 + \tau_{11}$);
- формирует новую строку МС (1, t_{11}) и заносит ее в этот массив.

Далее по алгоритму УПМ (поскольку JJ не пуст) происходит выбор процесса $i = 2$, и управление передается по адресу α_{21} для выполнения алгоритма активности 21 до появления оператора WAIT (τ_{21}) и возврата управления на ПП обслуживания операторов синхронизации. Формируется новый момент активизации процесса 2 ($t_{21} = m_0 + \tau_{22}$). В ТС модифицируется адрес продолжения выполнения процесса 2 (α_{22}) и формируется новая строка МС (2, t_{21}), которая заносится в соответствии со значением t_{21} (после строки для процесса 1) в этот массив. Аналогично обслуживается процесс $i = 3$ и управление передается по адресу α_{31} для выполнения алгоритма активности 31 до появления оператора WAIT (τ_{31}) и возврата на ПП обслуживания операторов синхронизации. Формируется новый момент активизации процесса 3 ($t_{31} = m_0 + \tau_{31}$). В ТС модифицируется адрес продолжения выполнения процесса (α_{32}) и формируется новая строка МС (3, t_{31}), которая в соответствии со значением t_{31} заносится в этот массив между строками для первого и второго процессов (рисунок 7.2).

Когда список JJ исчерпан, происходит смена модельного времени. Новым значением модельного времени становится минимальное значение t_0 , где

$$t_0 = \min(t_{11}, t_{31}, t_{21}) = t_{11} = m_1. \quad (7.2)$$

При этом $i = 1$ заносится в список JJ, и поскольку $t_{11} < m_8$, то управление передается на формирование списка JJ. Так как в момент $t_0 = m_1$ активизируется только один процесс, то в списке JJ будет находиться всего один процесс. Согласно алгоритму УПМ управление передается по адресу α_{12} на выполнение алгоритма активности 12 до появления оператора WAIT (τ_{12}). Процессу 1 назначается новый момент активизации $t_{12} = t_0 + \tau_{12} = m_0 + \tau_{12}$, и в ТС модифицируется адрес последующей активизации процесса α_{13} . В соответствии со значением t_{12} строка $(1, t_{12})$ заносится в МС после строк для процессов 2 и 3. Далее аналогичным образом продолжают обслуживать УПМ процессов и смена модельного времени согласно временной диаграмме (рисунок 7.2) до момента, когда модельное время $t_0 = m_8$. В этот момент по окончании обслуживания процессов 1 и 2 выполняется условие $t_0 \geq T_{ок}$, и процесс моделирования завершается, а управление передается на программу окончания моделирования.

При процессном способе имитации близость модели к отображаемой системе облегчает обозрение принятых предположений и упрощений. Это необходимо в задачах моделирования проектируемых систем, когда в ходе проектирования или исследования реальной системы в модель приходится вносить частые изменения. Процессный способ имитации обладает хорошими изобразительными возможностями при осуществлении многоуровневого модульного подхода к моделированию систем. Эти преимущества возрастают по мере роста размеров модели.

Более подробно описание процессного способа имитации приведено в работе [38].

7.8. Транзактно-процессный способ имитации сложных систем

При моделировании сложных систем исследователям приходится учитывать особенности системы. Например, все функциональные действия различны, но они принадлежат к разным группам ΦD_{ij} , которые можно объединить по принадлежности к компонентам K_i . У каждой компоненты K_i только одно из ΦD_{ij} может выполняться в конкретный момент времени. Причем имеет место последовательный характер выполнения ΦD_{ij} друг за другом. Поэтому можно

считать факт выполнения ΦD_{ij} состоянием S_{ij} компоненты K_i . Кроме того, алгоритмы выполнения ΦD_{ij} состоят в выполнении некоторого набора запросов на ресурсы моделируемого объекта. Алгоритмы предоставления алгоритмом ΦD_{ij} ресурсов системы на каждый k -й запрос. Это означает, что алгоритмы ΦD_{ij} могут обслуживать группу этих запросов в соответствии с приоритетом этих запросов π_k и дисциплиной выбора их из группы. В таких случаях запросы ресурсов Z_k удобно представлять в виде динамических элементов ИМ, которые назовем транзактами (TR_k). Сами по себе Z_k имеют сложную структуру, в которой указаны: принадлежность динамических элементов k -го типа к определенной группе, обслуживаемой в соответствии с их приоритетом π_k ; величина запроса ресурсов V_{Rr} , представляющая собой случайную величину, номер ΦD_{ij} , алгоритм выполнения которого имитирует выделение ресурса, указатель места накопления статистики обслуживания динамических элементов модели. Перечисленные особенности статистических и динамических элементов ИМ определяют выбор исследователем транзактно-процессного способа организации квазипараллелизма в ИМ. При этом все ΦD_{ij} , относящиеся к компоненте K_i , объединяются в один статистический элемент модели, называемый в дальнейшем процессом. Номера j ΦD_{ij} относятся к одному и тому же процессу номера i . Сами Z_k , являясь динамическими элементами, представлены транзактами TR_k сложной структуры. TR_k , в отличие от традиционного понимания, обладает: идентификатором k , отображающим принадлежность к определенной группе запросов Z_k ; приоритетом π_k , величиной запроса ресурса r -го типа V_{RrK} , номер состояния процесса j , определяющий номер ΦD_{ij} , которое обеспечивает выделение ресурса r -го типа; номер процесса j_1 , который обслуживает TR_i после процесса j (номер процесса-преемника обслуживания TR_k). [26]

Как видим, транзактно-процессным способом можно организовать имитацию сложных систем, которые требуют на высоком уровне детализации обслуживания TR_k сложной структуры отдельными ΦD_{ij} . При этом имеет место сходство структуры модели и объекта исследователя, удобство рассмотрения K_i как единого целого, состоящего из нескольких ΦD_{ij} и объединенного в единое целое, называемое процессом. Процессы обслуживают динамические элементы (TR_k), и для синхронизации различных процессов на входе процессов используются очереди TR_k .

Всю ИМ системы можно представить состоящей из четырех частей: группа моделей динамических элементов транзактов $\{ДЭМ_k\}$; группа моделей статистических элементов-процессов $\{СТЭМ_i\}$,

база данных динамических элементов (БДДЭМ); база данных статических элементов (БДСТЭМ). При этом существуют управляющие и информационные связи между ДЭМ_k и СТЭМ_i. Синхронизация взаимодействия процессов друг с другом и транзактами сложной структуры осуществляется через систему очередей {ОЧ_i}. На входе ИМ имеется множество генераторов, создающих новые транзакты с заданными интенсивностями поступления Z_k в ИМ объекта исследования. В ОЧ_i TR_k могут поступать от СТЭМ₁ и обслуживаться СТЭМ_{i+1}.

Группа {ДЭМ_k} описывается алгоритмами генераторов транзактов, начинающимися по адресу β_k и заканчивающимися операторами ожидания WAIT Д_k. Множество массивов переменных динамических элементов {МДЭМ_k}, расположенное в базе данных динамических элементов (БДДЭМ) по адресам S_{1k}, содержит в себе следующую информацию: F(V_{Rrk}), j, j₁ (здесь F(V_{Rrk}) – функция распределения размеров запросов ресурсов r-го типа Z_k динамическим элементом). Кроме информационной части транзакт сложной структуры может выполнять алгоритмическую обработку информации, находящейся в массивах переменных МДЭМ_k. Это обстоятельство означает, что динамические элементы ДЭМ_k мало отличаются от статических элементов СТЭМ_i. По модели от одной очереди ОЧ_j к другой очереди ОЧ_i движется триада транзакта: идентификат (k), приоритет π_k, адрес «тела» транзакта (S_{1k}). При традиционной структуре модель динамического элемента (TR_k) не имеет «тела» транзакта, и поэтому S_{1k} = 0. Статические элементы ИМ СТЭМ_i могут содержать несколько моделей ФД_{ij} (активностей АК_{ij}). Алгоритм каждой АК_{ij}, начинающийся адресом α_{ij} и завершившийся оператором синхронизации WAITC_j, обозначим как набор алгоритмических операций, записанных на базовом языке программирования, через СТЭМ_i. Поэтому каждой активности АК_{ij} соответствует своя группа алгоритмических операций СТЭМ_i, которые используют содержимое массивов переменных МСТЭМ_i и формируют набор статистик имитации в виде множества массивов статистик выполнения СТЭМ_i (STST_i). На рисунке 7.10 приведена блок-схема организации транзактно-процессного способа имитации.

Как видно из рисунка 7.10, модель объекта имитации состоит из двух групп: элементов модели {ДЭМ_k} и {СТЭМ_i}, каждая из которых использует собственную базу данных моделирования системы: ({МДЭМ_k}, {СТДЕ_k}, {МСТД_i}, {STST_i}). Вторую часть ИМ объекта (рисунок 7.10) представляет стандартная управляющая программа моделирования (УМП).

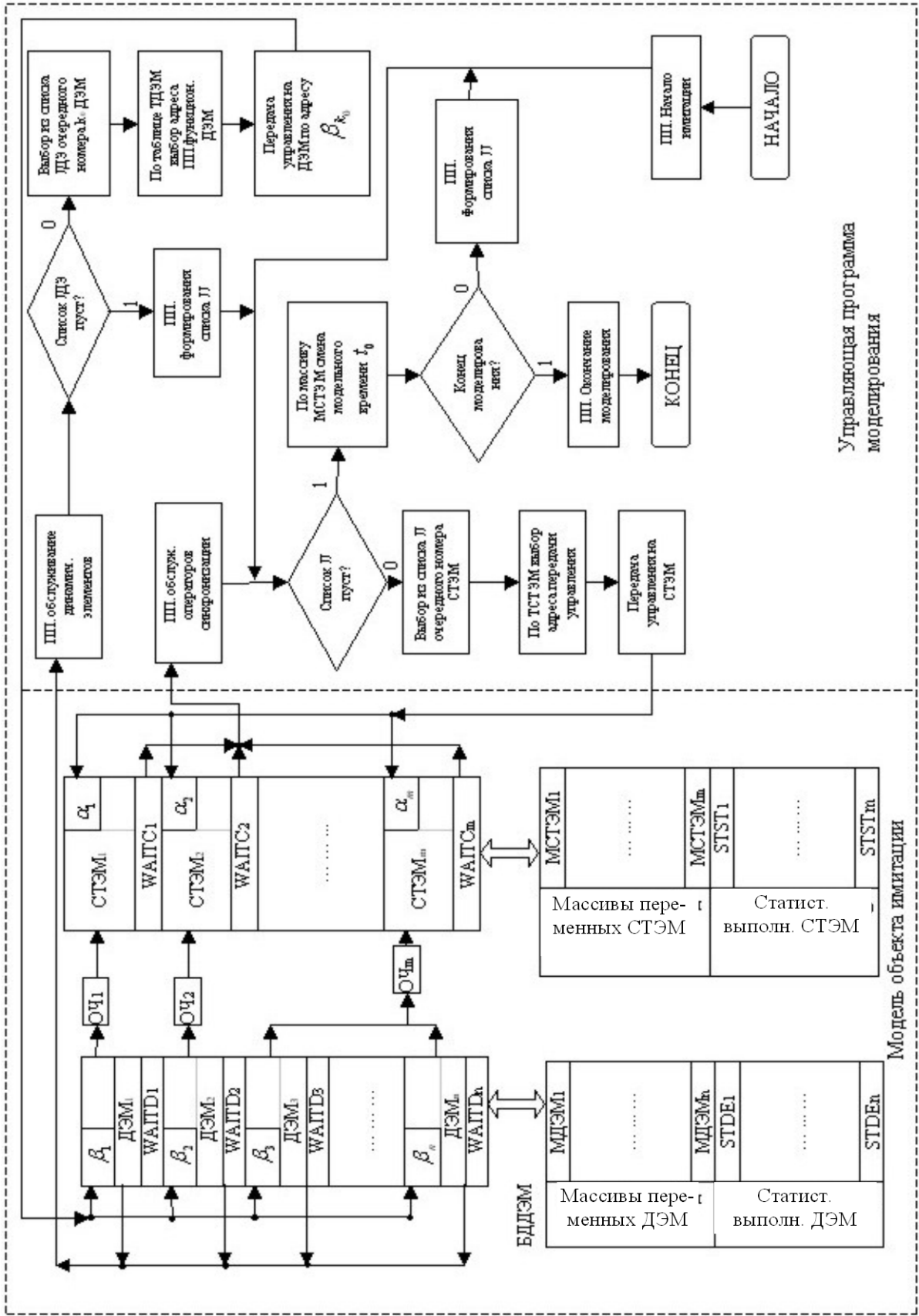


Рисунок 7.10 – Схема организации транзактно-процессного способа имитации СС

Алгоритм функционирования ИМ ДСС представляется последовательным взаимодействием ДЭМ, СТЭМ и УПМ. Процессы объединяются в связанные между собой активными элементами ДЭМ_к и СТЭМ_і, которые определяют функционирование одной и той же компоненты модели К_і и обслуживание множества динамических элементов (транзактов сложной структуры). Переход к выполнению активностей ДЭМ и СТЭМ считается состоянием соответственно динамических и статистических элементов модели и означает активизацию соответственно транзакта TR_к или процесса *ij*. Под состоянием транзакта или процесса понимают номер (*k*) или процесса (*ij*), на которые УПМ передает управление при свершении событий С_{ij} и К_і. Таким образом, изменение состояний реальной ДСС соответствует состояниям транзактов и процессов либо появлению соответствующих событий С_к и С_{ij}.

Процессы и транзакты связаны с УПМ с помощью соответствующих операторов WAITD_к и WAITC_і, по которым имеет место обращение к УПМ. Проверка выполнимости условий активизации осуществляется самими транзактами и процессами. Все переходы от программ ДЭМ_к на УПМ осуществляются с помощью программы обслуживания динамических элементов. Момент активизации ДЭМ_к определяется по формуле:

$$t_{\text{АДЭМ}_k} := t_0 + \tau_k, \quad (7.3)$$

где t_0 – модельное время;

τ_k – время выполнения алгоритма *k*-го динамического элемента.

Все моменты активизации ДЭМ_к упорядочены в списке моментов активизации СМАДЭ по возрастанию $t_{\text{АДЭМ}}$. Аналогичным образом все переходы от активностей процессов (СТЭМ_і) на УПМ осуществляются с помощью программ обслуживания операторов синхронизации СТЭМ. Момент активизации СТЭМ_і определяется по формуле:

$$t_{\text{АСТЭМ}_i} := t_0 + \tau_{ij}, \quad (7.4)$$

где τ_{ij} – время выполнения алгоритма *ij*-й активности.

Все моменты активизации СТЭМ_і упорядочены в списке моментов активизации МСТЭМ по возрастанию $t_{\text{АСТЭМ}_i}$.

Схема алгоритма выбора УПМ номера активизируемого динамического и статистического элементов представлена на рисунке 7.10.

Выбор очередного активизируемого ДЭМ_к осуществляется следующим образом. После установки в массиве СМАДЭ момента $t_{\text{АДЭМ}_k}$ проверяется условие: «список IDЭ пустой?». Таким способом

УПМ пытается завершить запуск тех ДЭМ_k, которые ожидают своей активизации. Если список IDЭ не пустой, то выполняется следующая последовательность действий: выбор из списка IDЭ очередного номера (k_0) ДЭМ. По таблице состояний динамических элементов (ТДЭМ) определяется адрес β_{k_0} активности ДЭМ_{k₀} и затем управление передается по адресу β_{k_0} на выполнение активности ДЭМ_{k₀}.

По окончании выполнения активности ДЭМ_{k₀} с помощью оператора синхронизации WAITLD_{k₀} управление возвращается на ПП обслуживания динамических элементов для назначения нового момента активизации активности ДЭМ_{k₀}. Если же предыдущая активизация ДЭМ_k завершена, то это означает, что список IDЭ пустой. В этом случае формируется новый список IDЭ только тех ДЭМ_k, у которых

$$t_{\text{АДЭМ}_k} < \min t_{\text{АСТЭМ}_i}. \quad (7.5)$$

Поэтому управление в УПМ передается на оператор проверки: пустой ли список процессов (список JJ)?

Если список JJ не пустой, то выполняется следующая последовательность действий: выбирается из списка JJ очередной номер процесса j ; по номеру j из таблицы статических элементов (ТСТЭМ) определяется адрес активности (α_j), которая должна быть активизирована именно в момент модельного времени t_0 ; передается управление по адресу α_j на выполнение активности СТЭМ_j. По окончании алгоритма выполнения активности СТЭМ_j с помощью оператора синхронизации WAITC_j управление возвращается на ПП обслуживания операторов синхронизации СТЭМ, которая назначает новый момент активизации данной активности по формуле (7.4). Снова проверяется условие: «Список JJ пустой?» Если все активности СТЭМ_k активизированы в момент модельного времени t_0 (список JJ пустой), то по массиву МСТЭМ осуществляется смена модельного времени t_0 . Проверяется условие окончания имитации. В случае необходимости продолжения имитации управление передается на ПП формирования списка JJ. В списке JJ формируются номера тех активностей СТЭМ_j, которые необходимо активизировать в новый момент модельного времени t_0 . По окончании формирования списка JJ управление передается на проверку содержимого списка JJ. Если же выполняются условия окончания моделирования, то управление передается на программу окончания моделирования. В самом начале имитации с помощью ПП «начало имитации» устанавливаются начальные состояния таблиц УПМ и управление передается на проверку: «Пуст ли список JJ?» Далее алгоритм УПМ выполняется аналогичным образом.

7.9. Агрегатно-процессный способ имитации сложных систем

В ряде случаев СС может состоять из компонентов трех типов: одна группа компонент $\{K_{1i}\}$ имеет типовой состав и алгоритмы выполнения нескольких ФД_{ij}; компоненты K_{2j} другой группы имеют уникальный состав ФД_{ij}; третья группа компонент $\{K_{3j}\}$ также имеет типовой состав, в их функции входит наблюдение за завершением функционирования $\{K_{1i}\}$ и управлением выполнением следующих $\{K_{1i}\}$. Примером такого совместного выполнения компонент сложной системы являются технологические процессы производства (ТПП). Выполнение ТПП представляется последовательностью технологических операций $\{ТХО_i\}$. Каждая ТХО_i, в свою очередь, может состоять из последовательности взаимосвязанных друг с другом микротехнологических операций $\{МТХО_{ij}\}$. При этом одна группа МТХО_{ij} не может начать выполняться до тех пор, пока не будет выполнена предшествующая группа МТХО_{ij} [31].

В таких ситуациях параллельно-последовательный характер выполнения $\{МТХО_{ij}\}$ удобно представлять в виде агрегатов. Причем, в отличие от классического представления кусочно-линейных агрегатов (КЛА), описанного в работах [6, 17], эти агрегаты могут иметь сложную структуру и состоять из нескольких взаимосвязанных активностей (АКА_{ij}). Каждая активность АКА_{ij} представляет собой модель ФД_{ij}. Внутри МТХО_{ij} группа ФД_{ij} может быть взаимосвязана, и их поведение определяется алгоритмом МТХО_{ij}. Каждую МТХО_{ij} в таких случаях можно представить агрегатами первого типа АGR_{1ij}, которые являются моделями МТХО_{ij} и могут состоять из одной или нескольких активностей АКА_{kij}. Начало выполнения одной группы $\{МТХО_{ij}\}$ зависит от завершения предшествующей группы $\{МТХО_{ij}\}$. Агрегаты АGR_{1ij} взаимодействуют друг с другом информационно, с помощью множества сигналов. После завершения алгоритма выполнения АGR_{1ij} на его выходе формируется SIGN_i сигнал, который поступает на один из агрегатов второго типа АGR_{2j}. В функции АGR_{ij} входит контроль выполнения предыдущих МТХО_{ij} и запуска на выполнение алгоритмов последующих МТХО_{jk}. С приходом последнего сигнала от АGR_{1ij} на входы АGR_{2j} срабатывает спусковая функция, которая формирует с выходов АGR_{2j} выходные сигналы на входы агрегатов АGR_{1ij}.

Число входных и выходных сигналов у АGR_{ij} может быть различным. В таких случаях компоненты 3-го типа K_{3j} реальной СС в ИМ представляются в виде агрегатов АGR_{2j}. Наконец, компоненты 2-го типа K_{2ij} можно представить в виде множества взаимосвязанных активностей

и объединить их в единый процесс ($PROC_{ij}$) номера m . Связи между $PROC_{mij}$ в ИМ организуются с помощью команд запуска ($PUSKC_m$) и остановки ($STOPC_m$). Поэтому каждая активность AKP_{km} процессов может оканчиваться либо командами $PUSKC_m$ и $STOPC_m$, либо оператором синхронизации процессов $WAITP$ ($\tau_{ij}; m$). Аналогичным образом, AGR_{1ij} состоят из набора активностей $AKTA_{kij}$ (где k -номер активности в агрегате номера ij), которые могут оканчиваться: либо операторами формирования сигналов агрегата ($SIGNA_{ij}$), либо командами $PUSKA_{ij}$ и $STOPA_{ij}$, либо операторами синхронизации агрегатов $WAITA$ ($\tau_{ij}; ij$). Агрегаты AGR_{2j} также состоят из набора активностей AKT_{kj} , которые могут оканчиваться либо операторами формирования сигналов агрегата $SIGNB_j$, либо командами $PUSKB_j$ и $STOPB_j$. Сами по себе эти сигналы могут иметь сложную структуру и быть «подкрашенными» (по аналогии с ДЭМ_к при транзактно-процессным способом имитации). В «теле» сигнала может находиться самая разнообразная информация, по которой последующие агрегаты AGR_{1ij} информируются об операционной обстановке в ИМ объекта имитации (например, имели ли место отказы оборудования при выполнении предыдущего агрегата). В отличие от транзактно-процессного способа имитации, сигналы по ИМ передаются только от агрегата-источника до агрегата – адресата их движения и далее по модели не распространяются. Поэтому сигналов между агрегатами нет.

На рисунке 7.11 приведена схема организации агрегатно-процессного способа имитации СС. Как видно из рисунка 7.11, модель объекта имитации состоит из трех групп элементов модели: агрегата универсальной структуры (AGR_1), универсального агрегата (AGR_2) и множества процессов $\{PROC_{ij}\}$. Каждый элемент этих агрегатов и процессов состоит из своего набора активностей. Таблицы информационного взаимодействия агрегатов первого типа и рабочие массивы агрегатов составляют информационную базу данных агрегатов первого типа (IBD_1). Агрегаты второго типа используют свои таблицы информационного взаимодействия, в которых рассматриваются приходы и отправления сигналов от агрегатов и для агрегатов первого типа. В совокупности эти таблицы и рабочие массивы составляют информационную базу данных второго типа (IBD_2). Процессы также используют таблицы и массивы их для информационной связи друг с другом, образуя в совокупности информационную базу данных третьего типа (IBD_3). Поскольку алгоритмы выполнения всех агрегатов первого типа одинаковы, то выполнение агрегатов номера (ij) первого типа представляется одной реентерабельной программой. Каждый номер AGR_{1ij} имеет в IBD_1 свою область, что позволяет одной и той же реентерабельной программе обслуживать параллельным образом в модельном времени все множество агрегатов первого типа.

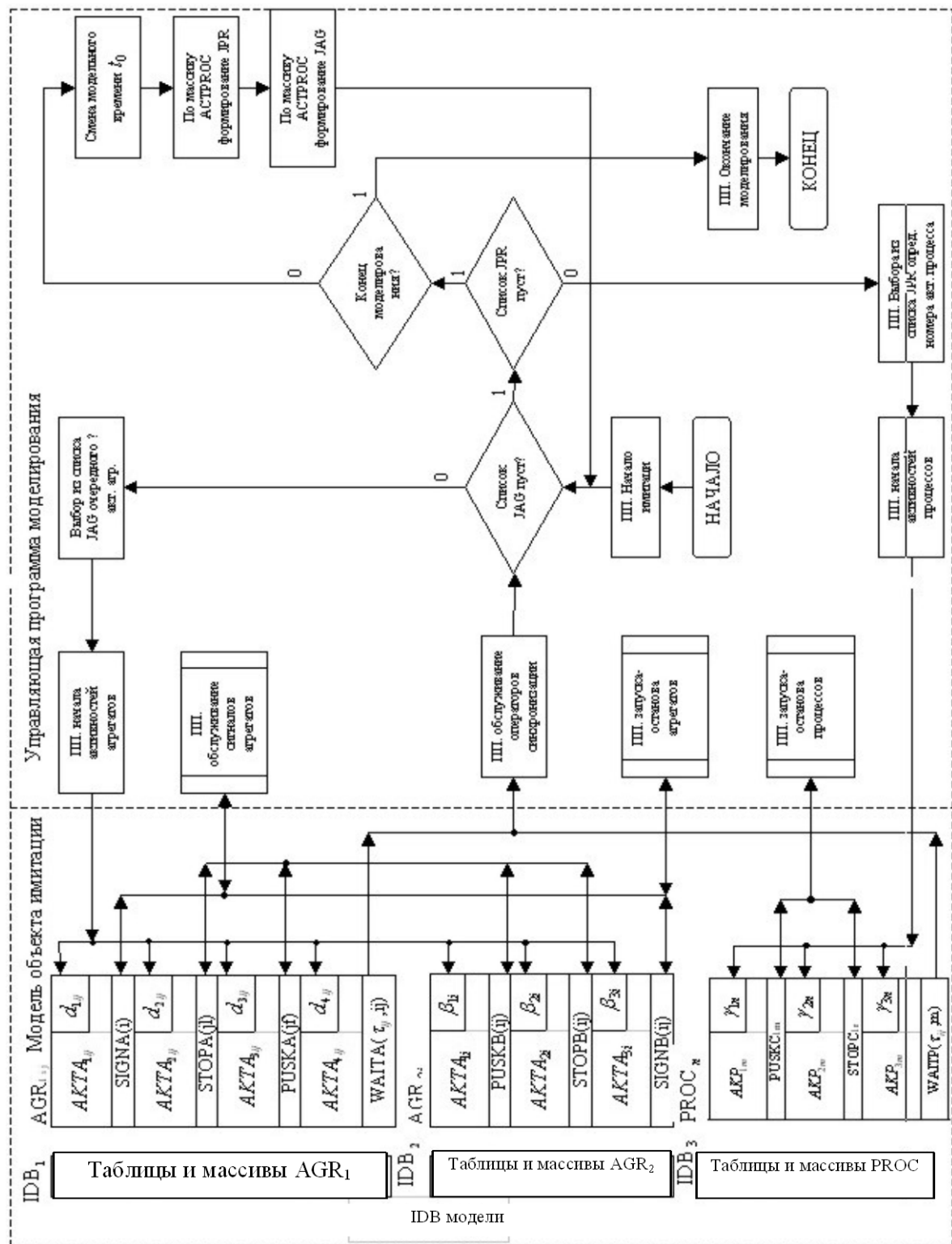


Рисунок 7.11 – Схема организации агрегатно-процессного способа имитации СС

Аналогичным образом, из-за типового характера алгоритмов второго типа, выполнение агрегатов AGK_{2j} также представляется одной реентерабельной программой. Здесь также каждый номер AGR_{2j} имеет в IBD_2 свою область, позволяя программе модели агрегата 2-го типа обслуживать параллельно в модельном времени все множество агрегатов 2-го типа.

Для организации квазипараллелизма выполнения агрегатов и процессов используется управляющая программа моделирования (УПМ). От ИМ объекта имитации на УПМ возможны следующие группы выходов:

- на выполнение сигналов агрегатов (по оператором $SIGNA_{ij}$ и $SINGNB_j$) инициируется ПП обслуживания сигналов;
- на выполнение операторов управления агрегатами по операторам $PUSKA_{ij}$ и $STOPA_{ij}$, инициирующим ПП обслуживания операторов пуска-останова агрегатов;
- на выполнение операторов управления процессами по операторам $PUSKC_m$ и $STOPC_m$, инициирующим ПП обслуживания операторов пуска-останова процессов;
- на выполнение операторов синхронизации агрегатов $WAITA(\tau_{ij}, ij)$ и процессов $WAITP(\tau_{ij}, m)$, инициирующим ПП обслуживания операторов синхронизации агрегатов и процессов.

Отметим, что три подпрограммы (ПП обслуживания сигналов агрегатов, ПП обслуживания операторов пуска-останова агрегатов, ПП обслуживания операторов пуска-останова процессов) представляют собой обычные процедуры, обеспечивающие обратный возврат от УПМ на следующую активность агрегата или процесса. Это обстоятельство определяет динамику взаимодействия УПМ и модели объекта имитации мгновенно в модельном времени. Поэтому выбор для активизации очередной активности агрегата первого типа или процесса осуществляется УПМ после возврата управления от модели объекта на ПП обслуживания операторов синхронизации.

Эта подпрограмма либо выполняет оператор $WAITA(\tau_{ij}, ij)$, назначая момент активизации активности агрегата, либо выполняет оператор $WAITP(\tau_{ij}, m)$, назначая момент активизации процедуры номера m . В этих операторах указывается (ij) номер агрегата первого типа (AGR_1), или же номер (m) процесса $PROC_m$. Назначение момента активизации агрегата ij или процесса m осуществляются по формулам:

$$t_{AKA_{ij}} := t_0 + \tau_{ij} \text{ и } t_{AKP_m} := t_0 + \tau'_{ij}. \quad (7.6)$$

Значения $t_{АКА_{ij}}$ и $t_{АКР_m}$ заносятся соответственно в массивы активизации $\{АСТАGR\}$ и $\{АСТPROC\}$. В этих массивах все агрегаты и процессы упорядочены по величине возрастания моментов их активизации ($t_{АКА_{ij}}$ и $t_{АКР_m}$). После выполнения операторов синхронизации УПМ проверяет условие: «Список (JAG) агрегатов, активизируемых в момент t_0 (JAG), пустой?» Если имеются номера агрегатов в этом списке, то выбирается из списка JAG очередной по приоритету номер первого типа (ij) и затем по таблице агрегатов первого типа определяется адрес начала активности α_{kij} и передается управление на выполнение этой активности АКТА $_{kij}$. Если в списке JAG нет номеров агрегатов первого типа, выбирается из списка JAG очередной по приоритету номер агрегата второго типа (j), затем по таблице агрегатов второго типа определяется адрес начала активности β_{kj} , и управление передается на выполнение активности АКТ $_{kj}$.

Если список JAG пустой, то проверяется список JPR номеров активизируемых процессов. Когда в списке JPR имеются номера процессов (m), которые нужно активизировать в момент времени t_0 , управление передается на программу выбора из списка JPR очередного активизируемого процесса (m), далее по этому номеру m ПП начала активностей процессов по таблице процессов определяет адрес начала активности γ_{km} и управление передается на выполнение активности PROC $_m$.

В тех случаях, когда список JPR пуст, проверяется условие: условие окончания моделирования выполнено. Если моделирование нужно продолжить, осуществляется смена модельного времени по формуле

$$t'_0 = \min \{t_{JPR}, t_{JAC}\}, \quad (7.7)$$

где t_{JAC} и t_{JAC} соответственно минимальные значения времен активизации соответственно процессов и агрегатов.

Далее с помощью ПП формирования списка процессов с помощью массива АСТPROC составляет новый список JPR, в который входит множество номеров (m_1) процессов, у которых

$$t_{АКР_m} = t'_0. \quad (7.8)$$

Затем с помощью ПП формирования списка агрегатов с помощью массива АСТАGR составляется новый список JAG, в который

входит множество всех номеров $\{ij\}$, у которых $t_{AKAGRij} = t'_0$. Далее управление в УПМ передается на оператор проверки условия: «Список JAG пустой».

В случае окончания моделирования управление передается на подпрограмму окончания моделирования и имитация завершается. В самом начале имитации подпрограмма «начало имитации» устанавливает начальные значения в таблицах и массивах имитационной модели и передается управление на проверку условия: «Пустой ли список JAG?»

РЕПОЗИТОРИЙ ГГУ ИМЕНИ Ф. СКОРИНЫ

8. СРЕДСТВА АВТОМАТИЗАЦИИ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ

8.1. Возможности языков и систем моделирования сложных систем

Преимущества языков и систем моделирования. Для того чтобы реализовать на ЭВМ модель сложной системы, нужен аппарат моделирования, который в большинстве существующих универсальных алгоритмических средств программирования отсутствует или недостаточно развит. В нем должны быть предусмотрены:

- способы организации данных, обеспечивающие простое и эффективное моделирование;
- удобные средства формализации и воспроизведения динамических свойств моделируемой системы;
- возможности имитации стохастических систем, т. е. процедуры генерирования и анализа случайных величин и временных рядов.

Такой аппарат должен иметь специализированные средства автоматизации моделирования, в состав которых обычно входят: язык описания объекта моделирования, средства обработки языковых конструкций (компилятор, транслятор или интерпретатор), управляющая программа моделирования (УПМ), осуществляющая имитацию во времени, и набор стандартных программных средств, реализующих дополнительные возможности по организации модельных экспериментов на ЭВМ.

Применение универсальных языков программирования в имитационном моделировании позволяет исследователю достигнуть гибкости при разработке, отладке и испытании модели. Однако при этом затрачиваются большие усилия на программирование модели. Поэтому целесообразнее применить специализированные средства моделирования, которые обладают следующими преимуществами перед универсальными: меньшими затратами времени на программирование; более эффективными методами выявления ошибок имитации; краткостью, точностью выражения понятий, характеризующих имитационные процессы; возможностью для некоторой предметной области исследований заранее построить стандартные компоненты, которые могут применяться пользователями при построении любых имитационных моделей; автоматическим формированием типов данных,

соответствующих принятому способу имитации и необходимых в процессе имитационного моделирования; удобством накопления и представления выходных данных; возможностью распределения памяти ЭВМ в процессе имитационного эксперимента.

Специализированные языки имитационного моделирования делятся на две самостоятельные группы, соответствующие двум видам имитации: для непрерывных и дискретных процессов. В последние годы наметилась тенденция моделирования непрерывно-дискретных систем. Для обеспечения средствами автоматизации указанного класса систем разработаны языки моделирования непрерывно-дискретных процессов.

Языки непрерывного имитационного моделирования. Группа языков непрерывного имитационного моделирования делится, в свою очередь, на языки аналогового моделирования и языки, которые применяются для решения систем дифференциальных уравнений, описывающих детерминированные замкнутые непрерывные системы. В языках аналоговых и гибридных ЭВМ, например, сумматор заменяется кодом операции суммирования, интегратор – кодом операции интегрирования и т. д. Взаимодействия между этими дополнениями функциональных компонент описываются с помощью блочно-ориентированного языка. Точно так же коммутационная панель аналоговой ЭВМ связывает компоненты аналоговой вычислительной программы, эмулируются и структура, и элементы аналоговой или гибридной ЭВМ.

Для моделирования явлений, представляемых переменными, непрерывными по диапазону своих значений, но дискретными по времени свершения событий, удобен язык ДИНАМО [7]. На языке ДИНАМО непрерывные процессы описываются соответствующими операторами, позволяющими представить зависимости в системе в виде дифференциальных уравнений первого порядка. Вводятся два типа переменных: состояний и скорости. Переменные состояний используются для описания состояний объекта моделирования и условий перехода моделируемой системы в эти состояния. Переменные скорости позволяют исследователю описывать порядок того, как меняются эти состояния за некоторый отрезок времени. Язык ДИНАМО нашел применение в тех случаях, когда исследователя интересуют информационные системы с обратной связью и он апробирует на модели методы решения задач в области управления.

В ряде случаев исследователь изучает причинно-следственные связи в сложной системе. Ему нужно использовать такие средства, как интегрирование, параметрический анализ или графическая оптимизация.

Для этой цели он строит имитатор непрерывной системы на цифровой ЭВМ. Модель представляется состоящей из взаимосвязанных функциональных элементов и эмулируется на цифровой ЭВМ. Наиболее удобным для этих целей является язык 360/Sistem CSMP [22]. С помощью этого языка специалисты, которые ранее применяли аналоговую технику, легко могут перейти к цифровой ЭВМ. Описание модели на этом языке представляет собой совокупность фраз, задающих правила коммутации общепринятых элементов аналогового моделирующего устройства: интеграторов, сумматоров, делителей, инверторов и т. д.

Поведение этих элементов реализуется соответствующими стандартными функциями. Для моделирования типовых элементов аналогового устройства у исследователя имеется возможность самому строить набор специальных функций применительно к его конкретной задаче. Допускаются также вставки в текст модели на языке ФОРТРАН для вычисления начальных условий, анализа результатов, выбора нового варианта значений параметров модели.

Языки дискретного имитационного моделирования. В зависимости от подхода к описанию рассматриваемых объектов имитации и метода организации квазипараллелизма языка моделирования дискретных систем языки можно разделить на следующие типы: ориентированные на просмотр активностей; позволяющие составить расписание событий в имитационной модели (ИМ); управляющие имитацией процессов, организующие взаимодействие транзактов; ориентированные на взаимодействие агрегатов, из которых скомпонована ИМ СС; транзактно-процессного способа имитации СС; агрегатно-процессного способа имитации технологических процессов производства. Рассмотрим основное назначение языков для каждого типа имитации дискретных систем. При этом рассматриваются только те языки моделирования, которые используются в странах СНГ и об их использовании имеются ссылки в русскоязычной литературе. Для знакомства с языками вообще разработанными рекомендуется работа [22].

Языки, ориентированные на просмотр активностей. В этом случае алгоритм функционирования системы представляется последовательным выполнением активностей. В УПМ предусмотрен механизм планирования, который обрабатывает упорядоченный во времени список моментов завершения активностей. Когда начинается выполнение некоторой активности, в этот список заносятся соответствующие время завершения и точка возврата в управляющую программу. После выполнения моделирующей программой всех операций, которые соответствуют некоторому моменту времени моделирования, возбуждается

механизм временного планирования. Он вычеркивает из списка элемент, соответствующий первой по времени завершения активности, устанавливает значение переменной, отображающей время моделирования, равным времени завершения соответствующей активности и передает управление в указанную точку управляющей программы. Завершение активностей может привести к инициализации новых активностей.

Некоторые языки, например SVPL [1], позволяют не регламентировать обслуживание активностей, а лишь указывать условия, при которых они могут произойти. УПМ просматривает набор всех условий перед сдвигом модельного времени на очередной интервал, что позволяет определить, могут ли произойти какие-либо события. Программа-модель состоит из двух частей: проверки условий инициализации активностей и обслуживания активностей.

Языки, составляющие расписание событий. В языках, ориентированных на реализацию событийного способа имитации, таких как GASP [22], СИМСКРИПТ [22], SMPL [22], ДИСМ [39], МОДИС-ВЕС [11], программа модели организована в виде совокупности так называемых процедур обслуживания событий. Выполнение этих процедур синхронизируется во времени списковых механизмов планирования. Каждый элемент списка определяет время события и имя процедуры события. Процедура события определяет, существуют ли условия запуска данной активности, в случае их наличия выполняет необходимые для запуска действия и планирует время его завершения. После выполнения всех действий, соответствующих текущему времени моделирования, активизируется механизм планирования, который выбирает новую процедуру события.

Программа, написанная на языке, реализующем событийный способ имитации, зачастую теряет всякое сходство со структурой реальной системы. Это связано прежде всего с тем, что одним и тем же событиям могут соответствовать активности, аппроксимирующие поведение различных компонентов системы. И при интерпретации результатов моделирования исследователю иногда трудно изучать взаимодействие компонент системы по множеству таких объединенных событий в системе. Это весьма существенный недостаток событийного способа имитации, особенно в тех случаях, когда имитационная модель используется как средство проектирования математического обеспечения систем управления реальными системами.

Языки, управления имитацией процессным способом. Процессный способ имитации получил наибольшее распространение

из-за того, что в нем сочетаются достоинства имитации активностями и событиями.

В языках, ориентированных на процессы, таких как SOL [22], ASPOL [1], SIMULA [22], ДИС [42], СЛЕНГ-Б [9], АЛСИМ [28], СИМУЛА-ИПИМ [2], СИМУЛА-67 [3], МПЛ/1 [47], СКИФ-Ф [19], МК PLSIM [29], моделирующая программа организуется в виде набора описаний процессов, каждое из которых соответствует одному классу процессов. Описание процесса объединяет активности всех процессов данного класса и имеет форму процедуры, которая выполняется одновременно для всех представителей этого класса, существующих в системе в текущее время. Фактически она является рентабельной программой, выполняемой в квазипараллельном режиме, когда управление передается в различные точки программы по мере того, как происходят инициализации, выполнение и задержка выполнения различных процессов. Синхронизация выполнения процессов реализуется с помощью механизма планирования, рассмотренного в главе 7. Программа имитационной модели, написанная на языке, реализующем процессный способ имитации, имеет ту же структуру, что и объект моделирования. Каждой компоненте системы соответствует свое описание процесса, что облегчает исследователю интерпретацию результатов моделирования, особенно в тех случаях, когда модель используется как средство изучения узких мест в системе. Перечисленные языки моделирования использовались ранее [22, 36].

К настоящему времени разработано и широко используется большое количество языков моделирования, представляющих собой либо расширение известных языков имитации, либо вновь разработанных. Перечислим состав языков опубликованных, с возможностями которых можно познакомиться в литературе, приводимой в данной книге. Прежде всего это языки моделирования, реализованные в системах моделирования **MISS** [52], **ДАСИМ** [52], **Arena** [52].

Инструментальная система моделирования **MISS** [52] реализует процессный способ имитации. По своей сути она весьма похожа на **СИМУЛА-67**. Предмет моделирования в **MISS** представляет собой набор объектов, организованных в многоуровневую иерархию, а функционирование объектов трактуется в **MISS** как совокупность параллельно текущих процессов. Между процессами существуют связи, реализуемые с помощью сигналов. Причем одни процессы могут изменять течение других процессов. Выполнение процессов в среде **СИМУЛА-67** представляется чередованием стадий. Моменты переключения между стадиями планируются по ходу имитации. Множество возможных переключений для всех стадий процессов фиксируется

заранее. Выбор конкретного переключения всякий раз определяется набором сигналов, полученных процессом на момент переключения. Существенным недостатком использования системы моделирования является отсутствие «покраски» сигналов (наличие информации о характеристиках сигнала и адресата его посылки), которое не позволяет моделировать иерархические структуры в динамике их развития. Скорее всего, исследователь должен заранее предусмотреть возможные динамические связи и состояния, чтобы адекватно реагировать в ИМ на момент их возникновения в ходе имитации системы.

Система моделирования **ДАСИМ** [52] используется для проектного моделирования локальных и распределенных сетей ЭВМ с коммутацией пакетов; специализированных мультипроцессорных ВС; выбора протоколов множественного доступа к каналам связи; исследования сетей СМО большой размерности; анализа состояний автоматических линий. Пользователь системы избавлен от необходимости программирования на всех этапах ИЭ. Система ориентирована на непрофессионала в области программирования и обработки данных. Для этого используется непроектный язык описания объектов с помощью устройств-имитаторов (автоматов), связей между ними с помощью сигналов управления этими автоматами. Сами автоматы специализированы, заранее запрограммированы и связи с УПМ стандартизованы. Все это позволяет описывать объекты на этом непроектном языке. Поэтому **СМ ДАСИМ** за счет узкой специализации ориентирована на определенный класс пользователей-проектировщиков перечисленных систем. В состав комплекса входят: интегрированная среда разработки и отладки моделей с набором скромных технологических возможностей; моделирующий комплекс закрытого типа, ориентированный на ограниченный состав автоматов; база данных результатов моделирования; справочная система ИЭ. Как видим, в случае иерархии технологических процессов комплекс использовать нельзя. Кроме того, сами технологические возможности системы в организации ИЭ также ограничены.

СМ Arena [52] позволяет создавать динамические компьютерные модели, с помощью которых можно представить производственные процессы. В основе ее реализации лежит процессный способ имитации. Моделированию подвергают только те системы, структура которых не меняется. Для самих процессов используется ограниченный набор переменных (размеры ресурсов, затраты времени на функциональные действия процесса, денежные затраты, служебные переменные организации ИЭ). В СМ имеются специальные средства оптимизации структурно-функциональных моделей. Средства организации

ИЭ позволяют: формировать отчеты и статистику функционирования основных объектов модели; использовать систему **Visual Basic** для изменения или определения параметров процессов с целью выявления наилучшего результата. Вспомогательное средство **Process Analyser** в составе СМ позволяет найти наилучший результат за счет изменения значений параметров модели, структуры модели, занятости ресурсов и их полезности. Имеется возможность путем сравнения альтернативных сценариев просмотреть различные решения исследования области допустимых решений и выбора лучшего из них. Другим дополнительным средством оптимизации является процесс **OptQuest**. С его помощью пользователь может менять пределы значений переменных и, запуская модель несколько раз, найти наилучшее решение. Отметим, что с помощью этого процесса реализуется однокритериальная оптимизация. Ограниченный состав процессов в данной СМ является ее первым недостатком. Вторым недостатком системы является большой процент времени программирования и подготовительной работы для организации ИЭ. Сама структура системы должна быть фиксирована на время ИЭ, ибо в противном случае оптимизация является проблематичной. Кроме того, эта система моделирования не может моделировать технологические процессы с меняющейся структурой.

Языки, организующие взаимодействия транзактов. В языках моделирования, реализующих транзактный способ имитации, в качестве динамических объектов при описании программы модели используются транзакты (заявки на обслуживание). Каждому транзакту может принадлежать определенный набор параметров, меняемых соответствующими блоками. Типичными представителями таких языков являются GPSS [22, 59], МОДЕЛЬ-6 [43], ИМСС [44], CSS [61], АСИМ [33]. Программа-модель на таком языке представляется в виде схемы, отображающей пространственное перемещение транзактов в системе. Затем осуществляется символическое кодирование схемы, при котором каждому блоку ставится в соответствие строка символов, отражающих текст программы. Управляющая программа моделирования создает транзакты, передвигает их по определенным блокам и производит действия, связанные с обслуживанием транзакта каждым блоком. По мере прохода транзактов через блоки происходит сдвиг модельного времени. Синхронизация обслуживания транзактов была рассмотрена в главе 7.

В последнее время появились новые системы моделирования СС, расширяющие технологические возможности языка моделирования GPSS [82]. Широкое распространение получили модификации языка GPSS в следующих системах моделирования: Any Logic [55], SLX [52],

GPSS/PC [59], ISS 2000 [52], Prof. Animation [52], Object GPS [59]. Появилась также современная версия МКАСИМ [14]. Основные расширения технологических возможностей этих систем рассматриваются в п. 8.3.

Языки, ориентированные на взаимодействия агрегатов. В таких языках моделирования в качестве динамических объектов выступают агрегаты. В функции управляющей программы моделирования входит синхронизация взаимодействия агрегатов. Она распознает, какое событие произошло в агрегате, управляет выдачей агрегатом выходных сигналов, синхронизирует работу в агрегатной системе, организует передачу выходных сигналов агрегата на входные полюсы других агрегатов. Сами же агрегаты имитируют поведение компонент сложной системы. Описания агрегатов также являются реентерабельными программами, выполняемыми в квазипараллельном режиме, когда управление передается различным программам, реализующим различные алгоритмы функционирования агрегата по мере того, как происходят инициализация и задержка выполнения алгоритмов агрегатов. Агрегатный метод имитации используется в системах моделирования АИС [6, 17,18] и САПАС [48].

Языки, ориентированные на комбинацию различных способов организации квазипараллелизма. Появились два типа комбинации процессного способа имитации:

- транзактно-процессного способа имитации, рассмотренного в главе 7, реализованного в системе моделирования МІСІС [27];
- агрегатно-процессного способа имитации, рассмотренного в главе 7, реализованного в системе моделирования САИМ [32].

Для увеличения технологических возможностей этих систем моделирования СС было разработано семейство программно-технологических комплексов имитации (ПТКИ), проблемно-ориентированных [25, 13, 36] на исследование технологических процессов производства и вычислительных процессов в локальных вычислительных сетях и системах ЭВМ. Описательные и технологические возможности этих систем моделирования изложены в п. 8.3.

8.2. Классификация языков имитационного моделирования сложных систем

К классификации языков моделирования можно подходить по-разному. Иногда к языкам предъявляют требования, определяемые

предметной областью их использования. В работе [22] содержится достаточно подробная классификация языков моделирования по описательным возможностям. В нашем случае мы прежде всего выделяем соответствие языков моделирования рассмотренным ранее способам формализации реальных объектов. Поскольку универсальные языки моделирования определяют изобразительные возможности представления алгоритмов, то естественна классификация их по способам реализации систем моделирования. При этом классификация на такой основе фактически представляет собой сравнение алгоритмических возможностей языков моделирования.

В таблице 8.1 приведена классификация систем моделирования по способу имитации процессов, протекающих в реальных системах. Отметим, что из всего множества языков моделирования рассматриваются только наиболее распространенные средства имитации, реализованные как на отечественных, так и на современных зарубежных типах ЭВМ, используемых в нашей стране. Часть языков имитации сохранили только свое историческое значение. Появились новые языки моделирования с более расширенными технологическими возможностями. Однако, по нашему мнению, описательные возможности языков моделирования только добавляются в новых системах моделирования, а известные описательные возможности языков имитации незначительно изменяются. По способу реализации различаем системы моделирования, использующие универсальную базовую систему программирования, и специально построенные системы, ориентированные на моделирование определенного класса сложных систем. В первом случае система моделирования представляет собой расширение возможностей универсальной системы программирования.

Таблица 8.1 – Классификация систем моделирования по способам имитации

Виды имитации	Способ организации имитации	Наименование системы моделирования
1	2	3
Непрерывные процессы	Аппроксимация системами уравнений	ДИНАМО [7]; 360/system CSMP [22]
Непрерывно-дискретные процессы	Событиями	НЕДИС [22]; НЕДИС-ЕС [41]

Окончание таблицы 8.1

1	2	3
Дискретные процессы	Процессами	DISLIN [23]
	Активностями	SMPL [1]
	Событиями	СИМСКРИПТ [22]; ДИСМ [33]; МОДИС-ВЕС [11]; GASP [22]; CMPL [22]
	Процессами	ДИС [42]; СЛЕНГ-Б [9]; АЛСИМ [28]; СКИФ [19]; MISS [52]; СИМУЛА-ИМП [2]; СИМУЛА-67 [3]; МПЛ/1 [47]; ДАСИМ [52]; MKPLSIM [29]; SOL [22]; ASPOL [1]; SIMULA [22]; Arena [52]
	Транзактами	МОДЕЛЬ-6 [43]; ИМСС [44]; GPSS [59]; GPSS/PCC [52]; GPSS/M [52]; Anylogic [55]; МК АСИМ [14]; CLX [52]; GPSSV [22]; АСИМ [52]; CSS [61]; ProfAnimation [52]; Objet GPSS [59]; ISS 2000 [52]
	Агрегатами	АИС [6; 17; 18]; САПАС [48]
	Процессами и транзактами	МИСИС [27]; ПТКИ ЛВС [13]; ПТКИ СРП [26]; ПТКИ УПС [25]
	Процессами и агрегатами	САИМ [34]; ПТКИ ТПП [26]

Обычно в качестве базового языка программирования используется АЛГОЛ, ФОРТРАН или PL/1. В таблице 8.2 приведена классификация систем моделирования по использованию базового программирования и типам ЭВМ, на которых они реализованы. Так, алголоподобную базу реализации имеют языки в таких системах моделирования, как ДИС, СЛЕНГ-Б, СИМУЛА-ИМП, СИМУЛА-67, АЛСИМ, ASPOL, SIMULA, SOL. На базе ФОРТРАНА реализованы языки в системах моделирования DISLIN, ДИНАМО, ИМСС, СКИФ-Ф, СИМСКРИПТ, GASP. Язык PL/1 в качестве базового используется в системах моделирования ДИСМ, СКИФ-П, SMPL, CMPL, ПЛ/1, МК, PLSIM.

В ряде случаев для моделирования разработаны системы с собственными языками моделирования. К таким системам можно отнести: НЕДИС, МОДИС-ВЕС, CSS, GPSS, GPSSV, АИС, АСИМ. Практически у всех перечисленных ранее систем моделирования название языка моделирования совпадает с названием системы моделирования.

Таблица 8.2 – Классификация систем моделирования по использованию базового языка программирования и типа ЭВМ

Тип ЭВМ	Системы моделирования						Собственные средства языка
	Реализация на базовом языке программирования						
	АЛГОЛ	ФОРТРАН	PL11	DELPHI	СИ	BASIC	
М-220	ДИС [42] СЛЕНГ-Б [9]	–	–	–	–	–	–
БЭСМ-6	ДИСМ [60] СИМУЛА-ИПМ [3] СИМУЛА-67 [2] АЛСИМ [28]	ДИНАМО [7]	–	–	–	–	МОДЕЛЬ-6 [43] НЕДИС [41]
ВС-ЭВМ	СКИФ-Б [19] СИМУЛА-67 [2] АЛСИМ [28]	СКИФ-Ф [19] ИМ СС [44]	СКИФ-П [19] ДИСМ [39] МПЛ/1 [46] МК PLSIM [33]	–	–	–	DISLIN [23] НЕДИС-ЕС [41] МОДИС-ВЕС [11] АСИМ [49] GPSS [60] АИС [5] САПАС [18]
Зарубежные типы ЭВМ	ASPOL [1] SIMULA [22] SOL [22]	GASP [22] СИМ-СКРИПТ [22]	SMPL [1] CMPL [22]	–	–	–	SPSSV [60] 360/Sistem GSNP [22] CSS ДИНАМО [7]
ПЭВМ	–	–	–	САИМ [29] ПТКИ ТПП [27]	МІСІС [25] ПТКИ ЛВС [12] ПТКИ УПС [26] ПТКИ СРП [27]	Arena [49]	GPSS [13] AnyLogic [55] GPSS/PC [13] ISS2000 [49] ProfAnimation [55] МК АСИМ [22] СМ ДАСИМ [49] ObjeetGPSS [13] MISS [49] SLX [49]

В современных версиях систем моделирования в качестве базового языка моделирования используются языки программирования **СИ**, **DELPHI** и **Visual Basic**. Так, например, на языке СИ реализованы языки моделирования в системе моделирования **MICIS** [27]; на языке **DELPHI** – язык имитации агрегатами в системе моделирования **САИМ** [32]; на языке **Visual Basic** – основные алгоритмы компонентов модели в системе моделирования **Arena**.

На ранних этапах разработки систем моделирования считалось более эффективным использовать базовый язык, расширив его до уровня языка моделирования. Однако с развитием моделирования в ряде предметных областей исследования оказалось, что эффективнее разрабатывать языки моделирования со своими системами компиляции. Этот способ более перспективен для пользователя из-за отсутствия ограничений, налагаемых на описание модели с помощью универсальных языков программирования, и благодаря более высокому уровню технологии их использования в определенной предметной области исследований.

8.3. Инструментальные и технологические возможности средств автоматизации моделирования сложных систем

При выборе средств автоматизации моделирования исследователя обычно интересуют две группы их свойств: изобразительные возможности представления алгоритма и удобство работы при использовании программного продукта. Автор на основе собственного опыта и контактов со специалистами, применяющими различные системы автоматизации моделирования, пришел к следующим выводам. Во-первых, из изобразительных возможностей языка моделирования исследователя интересуют три аспекта: состав операторов моделирования, наличие доступа к модельным характеристикам, возможности универсального базового языка программирования для реализации алгоритмов. Во-вторых, удобство работы с системой моделирования (СМ) определяется технологией ее использования, которая поддерживается данной СМ.

Обычно исследователь пытается получить ответы на вопросы в следующей последовательности: реализована ли СМ на ЭВМ, имеющейся у пользователя; имеет ли она средства автоматизации представления и интерпретации результатов моделирования; существует

ли возможность управления имитационным экспериментом; каков состав средств планирования экспериментов и обработки результатов моделирования; возможно ли управление процессом сбора статистики в ходе моделирования; имеются ли средства отладки и диагностики; удобен ли ввод-вывод информации в СМ; каков способ сборки компонент ИМ сложной системы. Конечно, этот список может быть дополнен. Однако анализ описаний существующих средств автоматизации моделирования убеждает, что вопросам технологии имитационного моделирования уделяется недостаточно внимания. Поэтому читателю предлагается систематизированная в виде таблиц информация об инструментальных и технологических возможностях наиболее распространенных СМ.

Будем считать, что инструментальные возможности средств автоматизации моделирования обусловлены:

- 1) описательными возможностями языков, реализуемых системами моделирования, в части отображения способа имитации;
- 2) возможностью (или отсутствием таковой) доступа пользователя к модельным характеристикам;
- 3) возможностью рекурсии в описании модели;
- 4) возможностью использования универсальных средств программирования, особенно при реализации арифметических операций.

Под описательными возможностями системы моделирования будем понимать богатство типов элементов модели. Каждый тип элементов модели, позволяющий исследователю формализовать явления в реальной системе, назовем фундаментальным элементом. От состава и числа типов фундаментальных элементов, используемых для построения моделей сложных систем, зависит удобство использования и легкость освоения исследователем системы моделирования. Читателю, интересующемуся описательными возможностями различных языков моделирования, можно рекомендовать работу [22].

Технологические возможности средств автоматизации моделирования определяются:

- 1) способом компилирования программ модели;
- 2) способом организации ввода-вывода информации в программе модели;
- 3) возможностью выдачи сообщения пользователю в ходе имитации;
- 4) наличием средств организации отладки и диагностики программы имитационной модели;
- 5) наличием средств организации сбора статистики и возможностью управления ими;

- 6) составом средств обработки результатов моделирования;
- 7) наличием специальных средств испытания имитационных моделей;
- 8) способностью управления имитационным экспериментом;
- 9) наличием средств автоматизации представления и интерпретации результатов имитации;
- 10) типом ЭВМ, на которой реализована система моделирования, и характеристиками организации вычислительного процесса на этой ЭВМ.

Ниже будут рассмотрены инструментальные и технологические возможности наиболее распространенных в нашей стране систем моделирования. Выбор для сравнения указанных систем моделирования определяется двумя обстоятельствами: наличием доступной технической документации или первоисточников, по которым можно составить представление о возможностях системы моделирования, и степенью знакомства автора с системой моделирования.

Отметим, что сравнивать между собой по вышеперечисленным пунктам можно только системы, реализующие один и тот же способ имитации. В противном случае достоинства способа формализации, проявляющиеся при построении данной конкретной модели, можно отнести к достоинствам языка моделирования. Зачастую оказывается, что при изменении модели, или даже условий проведения моделирования, удобнее было бы использовать другой способ имитации и, как следствие, другой язык моделирования. Кроме того, каждая из приведенных систем имеет свою область применения, в пределах которой она обладает преимуществами либо инструментального, либо технологического характера перед другими системами. Поэтому предлагаемое сравнение можно рассматривать как первое приближение в тех случаях, когда у исследователя нет опыта применения какой-либо из систем моделирования или же когда этот опыт негативный и исследователь стоит перед выбором нового средства моделирования для решения своих задач. В тех случаях, когда исследователь уже привык к определенной системе моделирования, данное сравнение поможет уяснить ему истинные технологические возможности применяемой им системы моделирования и, может быть, даст рекомендации на переход к более технологичной системе моделирования. У каждой из приведенных систем моделирования имеются свои сильные стороны, которые делают ее конкурентоспособной по сравнению с другими системами моделирования при построении некоторых моделей. Отметим некоторые из них, которые, по мнению автора, являются наиболее интересными.

Весьма перспективным способом компилирования программы модели обладает СМ АИС. Перевод базового языка на ФОРТРАН позволяет распространить СМ АИС на все типы ЭВМ, где используется ФОРТРАН.

Интерпретация текста модели, реализованная в СМ GPSS, является наиболее экономичным способом реализации описания модели по сравнению с другими СМ.

Применение препроцессора, реализованного в СМ МПЛ-1, позволяет легко наращивать возможности системы и расширять предметные области ее использования.

Процедурный способ расширения системы моделирования, реализованный в МК PLSIM, более экономичен по сравнению с СМ МПЛ/1 за счет исключения этапа препроцессорирования, но более труден при переходе из одной предметной области в другую.

Наличие специальных модулей ввода заданий и вывода информации в СМ НЕДИС делает ее удобной и привычной для широкого круга специалистов, имеющих опыт программирования на ЕС ЭВМ.

В ряде случаев исследователь может предпочесть набор специальных операторов для задания состава модели, ввода и вывода информации, реализованных в МК PLSIM.

Больше всего преимуществ при организации ввода-вывода информации исследователь получает при использовании диалоговой системы АЛИСА, реализованной в СМ АИС, именно из-за возможности оперативного изменения информации в интерактивном режиме взаимодействия с системой.

Весьма существенно для исследователя наличие специального набора операторов выдачи сообщений о ходе имитации, реализованного в СМ: DISLIN, МК PLSIM, МПЛ/1. В ряде случаев пользователь может применить систему трассировки, реализованную в СМ НЕДИС, либо генератор сообщений, реализованный в СМ СИМСКРИПТ.

Решающим фактором, определяющим технологичность системы моделирования, может оказаться наличие системы отладки имитационных моделей. Учитывая это обстоятельство, в СМ МОДИС-ВЕС введены специальный банк описания моделей и система выдачи сообщений об ошибках в моделях, возникающих в ходе имитации дискретных процессов. В МК PLSIM реализована специальная система отладки и диагностики для локализации ошибок. Использование диалоговых возможностей системы АЛИСА существенно облегчает этап отладки имитационных моделей, реализованных с помощью СМ АИС в виде агрегатных систем.

По организации сбора статистики наибольшие возможности пользователю представляют СМ: СКИФ, МПЛ/1, МК PLSIM. Важной для исследователя при этом является возможность управления сбором статистики, планирования его возобновления, исключения или выделения переходного периода имитации при сборе статистики моделирования.

Наличие специальных средств обработки результатов моделирования в составе СМ существенно повышает ее технологические возможности. С этой точки зрения, по мнению автора, больше всего возможностей предоставляет МК PLSIM. Весьма перспективным является включение либо специальной системы обработки данных АСОД (как это делается в СМ НЕДИС), либо обработка результатов моделирования с помощью набора задач в составе внешнего программного обеспечения (как это реализуется в СМ АИС).

В ранних СМ вопросам автоматизации испытаний имитационных моделей уделялось мало внимания. Только две из рассмотренных СМ имеют средства испытания имитационных моделей. В МК PLSIM для этой цели в распоряжение исследователя предоставляется набор стандартных процедур организации испытания моделей. СМ АИС предлагает воспользоваться специальными программами, имеющимися в составе ее внешнего программного обеспечения.

Вопросам организации и планирования модельных экспериментов на ЭВМ в ранних СМ также уделялось мало внимания. В основном дело ограничивалось введением либо специальных карт инициализации подмоделей с их отдельным компилированием и последующим автоматическим объединением в одну модель, либо созданием средств организации многопрогонного режима имитации. Использование диалоговой системы АЛИСА существенно облегчало организацию имитационных экспериментов в рамках СМ АИС. В МК PLSIM и СМ DISLIN для планирования экспериментов реализована система операторов, запрограммированных в виде PL-процедур. Интересными являются идеи применения в СМ ДИС параметризуемой конструкции «описание эксперимента» и набора операторов коммутации входов модели или указание ее состава в СМ АЛСИМ.

Средствами автоматизации представления результатов имитации в большинстве СМ являются операторы сбора статистики. В МК PLSIM для этой цели пользователю предоставляется набор операторов документирования результатов моделирования. СМ НЕДИС позволяет организовать графическое представление результатов моделирования.

Отметим, что основная доля ранних отечественных СМ была реализована на ЕС ЭВМ. Правда, такие СМ, как НЕДИС, АЛСИМ,

СИМУЛА-67, имеют также реализации на широко распространенной на то время отечественной ЭВМ БЭСМ-6. Однако технологические возможности реализации указанных СМ на ЕС ЭВМ и БЭСМ-6 различны и определяются возможностями организации вычислительного процесса на этих ЭВМ.

С развитием элементной базы и организацией обработки на ЭВМ появились новые возможности в технологии программирования и отображения результатов вычислений. Они обусловлены развитием новых систем моделирования, обладающих существенно большими технологическими возможностями и реализованных на ПЭВМ.

Например, наибольшее распространение в странах СНГ получила система моделирования **GPSS**. Мы рассмотрим наиболее широко используемые модификации системы моделирования **GPSS**.

Одной из немногих российских разработок в области имитационного моделирования, получивших признание за рубежом, является система **AnyLogic** [23]. Эта СМ предлагает пользователю графическую среду для создания моделей на основе простых виртуальных средств с дополнительным использованием всех возможностей объектно-ориентированного языка программирования **Java**. Любой объект в **AnyLogic** представляется как класс **Java**, и пользователь может добавлять в модель свои классы, использовать и разрабатывать свои библиотеки классов. По графической модели генерируется своя Java-программа. Во время выполнения модель представляет иерархию экземпляров активных объектов. Собранный моделью может работать локально. В ней будет отражена необходимая иерархия с помощью браузера. Таким образом, по сравнению с обычной СМ, функционирующей под контролем УПМ, данная система моделирования обладает тем преимуществом, что позволяет отобразить всю иерархию имитации процессов в соответствии с их визуальной спецификацией сложных инженерных систем. Кроме того, дополнительным технологическим средством данной СМ является возможность использования структурных диаграмм и карт состояний, что позволяет отражать в моделях структурную и поведенческую иерархию сложных систем. Пользователю предлагается возможность с помощью анимации наглядно представить динамику поведения системы во время имитации. Эти средства также помогают пользователю создавать свой виртуальный мир (в виде совокупности графических объектов и мнемосхем), который управляется динамическими параметрами модели по законам, определенным пользователем, с помощью уравнений и логики поведения моделируемых объектов. Моделирование основывается на механизме обмена сообщениями между активными объектами через

порты, а на внутриобъектном уровне используются таймеры и события с обычной семантикой. Логику обработки сообщений можно задать графически с помощью карт состояний. Поэтому СМ **AnyLogic** имеет аналоги блоков и транзактов языка моделирования GPSS. Дополнительной возможностью является отображение ресурсов и визуализация динамики развития процессов. Кроме того, в СМ присутствует автоматическая обработка случайных параметров. СМ допускает описание сложных систем с помощью дифференциальных уравнений первого порядка. Поддерживается 35 стандартных вероятностных распределений, имеются средства сбора и анализа статистики моделирования, а также возможность презентации модели в любых формах. Система позволяет проводить серии экспериментов по методике Монте-Карло, осуществлять анализ чувствительности и оптимизацию вариантов организации СС на имитационных моделях. В **AnyLogic** встроен оптимизатор, с помощью которого можно комбинировать различные эвристики, генетические алгоритмы, нейронные сети с математической оптимизацией. Возможно задание дискретных и непрерывных параметров модели, соответствующих целевым функциям в условиях неопределенности и при наличии ограничений. Как видим, СМ **AnyLogic** обладает очень развитыми возможностями технологии постановки имитационных экспериментов. Однако по своей сути и транзакты, и блоки их обслуживающие ничем не отличаются от СМ **GPSS**. Во всех сообщениях о системе транзакты перемещаются от одного блока к другому блоку при их неизменной структуре. Самый важный недостаток классического способа имитации системы транзактами остался в способе его представления: транзакт представляет собой пару (идентификатор (i) и приоритет (π_i)). Никакой дополнительной информации в себе транзакт не содержит. Поэтому траектория их движения по модели остается неизменной и должна быть заранее предусмотрена пользователем. Для перераспределения потоков транзактов необходимо устанавливать специальные блоки, которые меняют приоритеты транзактов и таким способом влияют на динамику обслуживания транзактов блоками. Это существенный недостаток СМ **AnyLogic**, показывающий, что разработчики увлеклись технологией имитации, но упустили главное (возможность динамического изменения алгоритмов их обслуживания). Описательные возможности СМ находятся на уровне **GPSS**.

Недостатки систем моделирования **GPSS** и **AnyLogic** позволили внедриться на рынке программных продуктов имитации СМ **SLX**, которая основывается на возможностях новой модификации **GPSS/M** [52]. При этом решались три проблемы: введение послойного построения

модели, создание механизма организации связей и управление между слоями и создание внешнего интерфейса для обеспечения доступа к слоям модели. Здесь разработчики пошли дальше, чем в **GPSS/M**, по пути повышения технологии создания проблемно-ориентированных пакетов прикладных программ имитации и затем специализации этих пакетов под конкретные предметные области. СМ **SLX** позволяет расширить возможности имитации с помощью команд и директив, являющихся, по сути дела, макросами. Базовым элементом СМ **SLX** является подпрограмма «ПАК», которая определяет время задержки или освобождения сервера. С помощью «ПАКов» реализуется квази-параллелизм в имитационной модели. Функциональная сторона обслуживания транзактов скрыта от пользователя. Поэтому роль транзактов выполняет класс объектов, а собственно, ПАКи являются носителями данных. В результате каждый объект является членом какого-либо класса и имеет копию в данных объекта. Параллелизм в СМ **SLX** реализуется двумя путями: осуществлением связей между объектами (внешний параллелизм); многократным использованием самого объекта (внутренний параллелизм). В итоге сама СМ **SLX** представляет собой платформу, на основе которой могут быть построены другие средства имитации. Поэтому СМ **SLX** по сути является объектно-ориентированной настройкой языка C^{++} над **GPSS**. Следовательно, СМ **SLX**, несмотря на все ее дополнительные возможности, сохраняет все недостатки описания с помощью **GPSS** отображения самих объектов. При этом необходимо иметь в виду, что язык **SLX** является набором макросов, содержащим небольшое число основных определений, позволяющих читать файлы в ходе выполнения предписаний макроса. Как видим, разработчики СМ **SLX** снова расширили технологические возможности базового языка моделирования **GPSS/M**. Однако основной недостаток такого подхода к использованию имитации состоит в использовании классического способа имитации с помощью транзактов традиционной структуры и назначения. При таком способе имитации, с нашей точки зрения, разработчики не предусматривали в ИМ возможности динамического изменения параметров обслуживания транзактов.

Другим развитием **GPSS** на ПЭВМ является СМ **GPSS/PC** [59]. В ней сочетаются фазы создания и эксплуатации ИМ. По мере ввода программы ИМ она строит соответствующие структуры данных, и новые операторы автоматически подсоединяются к составу ИМ. Поэтому возникает необходимость выполнения заново всех фаз сеанса при корректировке ИМ. Модель строится путем объединения объектов имитации в единую блок-схему. Основой ИМ являются

программные модули, описывающие функционирование объектов языка **GPSS**, а управляющая программа моделирования (УПМ) называется «симулятором». Язык **GPSS/PC** представляет собой два подмножества: операторов и системных команд. Три типа операторов (описания блоков, описания данных, управления имитацией) позволяют описывать объекты как компоненты системы массового обслуживания (СМО). Диалоговые возможности языка реализуются с помощью специальных команд. Для задания условий имитации (таких, как продолжительность моделирования, число прогонов имитации, порядок и условия сбора статистики) используются управляющие операторы, которые имеются и в языке моделирования **GPSS** [59]. Средства интерактивной графики позволяют пользователю наблюдать за ходом процесса имитации и оперативно модифицировать ИМ или менять условия моделирования во время имитации. Взаимодействие пользователя с УПМ осуществляется в режиме активного диалога. Для анализа результатов имитации пользователю предоставляются «виртуальные окна» и «микроокна». В «виртуальных окнах» на экране дисплея отображается информация о состояниях объектов. Пользователь может активно вмешиваться в процесс отображения в «виртуальном окне», задавая различные команды языка **GPSS/PC** и используя специальные поля команд, имеющиеся в этих окнах. «Микроокна» – это небольшие графические окна, в которых отображаются текущие значения системных переменных ИМ. В ходе имитации содержимое этих окон меняется динамически при изменении значений переменных в этих окнах. Например, окно «данных» ИМ позволяет обеспечить ввод и корректировку исходных текстов ИМ, а также отображение информации об использовании данных при моделировании. Окно «устройств» обеспечивает графическое отображение информации о состоянии объектов **GPSS/PC** типа «устройство» в процессе имитации. Окно «памяти» позволяет графически отобразить информацию о состояниях объектов типа «память» в процессе имитации. Окно «таблиц» обеспечивает визуализацию значений числовых характеристик и гистограмм распределения частот для значений моделируемого объекта. После завершения имитации автоматически создается системный файл статистики имитации, который затем обрабатывается для получения вторичной статистики моделирования. Выходная информация имитации состоит из следующих сегментов: основная информация о результатах имитации, информация о статистике имитации во времени моделирования, об очередях и коэффициентах использования. Основным недостатком **GPSS/PC** состоит в том, что в качестве динамического элемента используется транзакт классической структуры

(идентификатор и приоритет), а в качестве статических элементов модели применяются несколько типов блоков, представляющих в своей основе приборы СМО. Как видим, несмотря на все дополнительные технологические возможности, СМ **GPSS/PC** трудно использовать для имитации иерархических объектов, динамически меняющих свою структуру.

Следующим шагом в имитационном моделировании явилось использование интерактивного режима. Примером такого подхода к организации ИЭ явилась СМ **ISS2000** [52], которая в качестве базового языка моделирования использует все тот же **GPSS**. СМ реализована путем создания лингвистического процессора (ЛП), который с помощью интерактивного взаимодействия с пользователем на некотором языке L строит подмножества таких высказываний D из множества G , что только высказывания D всегда истинны. По своей сути язык L представляет собой систему меню для пользователя, и поэтому множество G соответствует дереву альтернативных вариантов построения подмодели, а подмножество высказываний D определяет параметрические настроенные модели конкретной реализации ИМ. Основными элементами СМ являются: язык интерактивного проектирования моделей (система меню), система параметрической настройки моделей; подсистема работы со стандартными функциями распределений; лингвистический процессор; подсистема организации имитационного эксперимента и статистической обработки результатов экспериментов. В процессе автоматизированного проектирования ИМ реализуется заданная стохастическая сеть СМО и определяются характеристики и параметры для каждой вершины сети и задаются параметры для ИЭ. В результате совокупность реализаций параметрической настройки для всех подмоделей системы определяет необходимую информацию для реализации конкретной модели в среде **GPSS**. Она отображает совокупность подмоделей в конкретную **GPSS**-модель. Проблемы у исследователя возникают при отображении в графическом виде маршрутов движения транзактов, когда маршрут определяется типом транзакта (например, разный тип заготовки обрабатывается в соответствии со своей технологической картой). Поэтому введено понятие виртуальной сети, которое упрощает при этом работу с разными типами транзактов, имеющими собственные маршруты движения по узлам сети СМО. В итоге виртуальная стохастическая сеть представляет собой СМО с переменной структурой связей устройств обслуживания. Такая сеть дает возможность представления в сети маршрутов движения транзактов разного типа и может применяться для моделирования сборочного производства или

транспортных систем. Для реализации такой сети в СМ **ISS2000** вводится элемент «маршрутизатор» для каждого узла стохастической сети СМО. Маршрутизатор реализуется путем использования средств косвенной адресации языка **GPSS** и реализации функции, которые задают маршруты движения транзактов по узлам сети и времена обработки транзактов. Здесь разработчики сделали шаг в правильном направлении. Но основной недостаток, определяемый использованием **GPSS** в качестве среды моделирования, все же остался. С помощью маршрутизатора удается менять траекторию движения транзакта, используя только идентификатор транзакта. Но на практике возникают ситуации, когда траектория движения транзакта по сети СМО меняется в зависимости от операционной обстановки. В частности, трудно моделировать в среде классического **GPSS** иерархические вероятностные технологические процессы.

Пакет прикладных программ **ProfAnimation** [52] позволяет использовать широкий спектр языков моделирования при любом программном обеспечении этих языков. В основе пакета лежит мощный командный язык, который используется для широкого класса приложений. Существует несколько версий пакета (для разного уровня пользователей, начиная от профессионалов и заканчивая студентами). Все реально использованные версии основаны на базовом языке моделирования **GPSS**. Этим обстоятельством определяются и недостатки пакета как средства моделирования иерархических вероятностных технологических процессов. Здесь сделан очередной шаг в технологии постановки ИЭ, но слабость формализации на основе традиционного языка моделирования СМО **GPSS** существенно ограничивает область его использования.

Система моделирования **Object GPSS** [59] основана на идее погружения языка **GPSS** в один из хороших языков программирования, обеспечивающих эффективную работу с ОС Windows и представления модели СМО в виде набора процедур на языке **Object Pascal**. Для этого был разработан набор модулей (Unit), которые реализуют процедуры и функции, необходимые для построения ИМ. Процедуры естественным образом разделяются на собственные процедуры и блоки. С помощью блоков (процедур) и формируется ИМ. Они соответствуют блокам языка **GPSS** и реализуют движение транзактов по модели. Остальные процедуры используются при построении других частей ИМ. СМ **Object GPSS** позволяет просто и естественно писать тексты моделей на **Object Pascal**, и ее возможности, безусловно, шире самого **GPSS**. Однако использование в качестве основы ИМ **GPSS** представления объектов имитации существенно сокращает возможности

описания сложных систем. Тем более ее трудно использовать для описания иерархических производственных систем.

Разработчики **МК АСИМ** [22], оставив без изменений богатые возможности описания СС на основе аппарата теории массового обслуживания (ТМО), реализовали современную версию СМ на персональных ЭВМ. В итоге исследователь получает в свое распоряжение возможности описания СС аппаратом ТМО при достаточных для практики технологических возможностях постановки ИЭ.

При разработке моделирующего комплекса **МК АСИМ** [22] основное внимание было уделено простоте перехода от временных диаграмм (ВД) компонент-объекта имитации к ИМ системы за счет расширения по сравнению с **GPSS** возможностей описания самих приборов, обслуживающих транзакты классического типа (TR_i). Любую компоненту системы представляют прибором СМО, который называется в **МК АСИМ** устройством ($УС_j$). Устройство отличается от прибора СМО следующим: на входе и выходе устройства возможно любое число очередей, в которые попадают транзакты в соответствии с их приоритетами π_i , выбор которых определяется дисциплиной обслуживания транзактов ($DISC_{ij}$). Любое $УС_j$ может обслуживать любое число TR_i по индивидуальному для каждого из них закону распределения времени обслуживания $F_{ij}(t)$ TR_i на $УС_j$. Каждое $УС_j$ дополнительно имеет 3 управляющих входа: «включить», «выключить» и «прервать» обслуживание TR_i на устройстве $УС_j$. На эти управляющие входы поступают соответственно управляющие сигналы. Если $УС_j$ выключено, на входе его образуются очереди, несмотря на то, что устройство может быть свободным от обслуживания. И это состояние продолжается до прихода сигнала на вход «включить». На входе и выходе $УС_j$ возможно рождение из TR_i сигналов всех типов. На выходе $УС_j$ по окончании обслуживания каждому TR_i указывается траектория его поступления в очередь к следующему $УС_j$. Возможны либо вероятностный выбор направления TR_i по вектору вероятностей $\{P_{ij}\}$, либо детерминированный выбор маршрута движения транзакта по его индексу (i). На входе и выходе $УС_j$ допускается: размножение TR_i на любое число копий; формирование из TR_i любого числа управляющих сигналов $УС_j$ всех этих типов. В результате изменения числа $УС_j$, мест их генерации при постоянном составе $УС_j$ можно изменять логику моделируемой системы. В **МК АСИМ** модель вначале задается графически. Логика функционирования ИМ системы состоит в следующем: каждый элемент ИМ представляет собой графическое отображение $УС_j$, которые обслуживают поток TR_i . Это отображение легко записывается набором декларативных утверждений, отражающих

все аспекты функционирования $УС_j$. К числу описателей ИМ относятся: состав $УС_j$, связи между ними, логика функционирования и взаимодействия $УС_j$. В итоге формируется описание объекта имитации на входном языке **МК АСИМ**, которое затем переводится автоматически в программу ИМ. При таком подходе задача исследователя состоит в построении графической модели и последующем описании элементов этой модели набором декларативных предложений, не представляющих собой пакет программы для ЭВМ. Как видим, каждый из элементов ИМ характеризуется набором терминов СМО, в которых указываются: ресурс памяти, входящие потоки, очереди и дисциплины обслуживания TR_i , выходные потоки, маршруты TR_i , логика управления состояний $УС_j$. Каждое $УС_j$ на языке **МК АСИМ** описывается комбинацией шести предложений: устройство, входящий поток, очередь, механизм обслуживания, время отказа, выходящий поток. ИМ представляется в виде сети СМО, состояний $УС_j$ и связей между ними (поток TR_i). Исследователь проводит декомпозицию сети на отдельные узлы, а затем каждый узел отождествляется шестью языковыми конструкциями (предложениями **МК АСИМ**). Общий текст описаний $\{УС_j\}$ завершается предложением «ВРЕМЯ МОДЕЛИРОВАНИЯ», в котором задаются: интервал времени имитации, конструкция и состав стандартной статистики. По каждому устройству и транзакту собирается стандартная статистика. Технологическая оболочка **МК АСИМ** весьма бедная, однако предметное представление сети СМО в виде множества из шести предложений очень привлекательно для специалистов СМО, поскольку не требует от них этапа программирования сети СМО из-за того, что эту операцию **МК АСИМ** выполняет автоматически. В качестве недостатка укажем, что предложенного для описания $УС_j$ явно недостаточно при моделировании иерархических систем. Кроме того, очень трудно представить результаты имитации иерархических систем. Правда, в **МК АСИМ** имеется возможность использовать для этой цели: управляющие сигналы; синхронизирующие транзакты с помощью операции деления транзактов на входе $УС_j$; блоки управления. Генерация новых транзактов осуществляется посредством включения в алгоритм ИМ управляемых транзактов. Как видим, эти дополнительные возможности лишь частично устраняют указанный недостаток **МК АСИМ**. Кроме того, у **МК АСИМ** очень ограничены технологические возможности, что сужает его применение несмотря на кажущуюся простоту языка **МК АСИМ**. Более подробно возможности **МК АСИМ** и его современной версии излагаются соответственно в работах [12] и [50].

Несмотря на широкое распространение систем моделирования на базе языка **GPSS**, необходимо констатировать их слабые описательные возможности. Увлечение технологическими возможностями организации имитационных экспериментов (ИЭ) привело к тому, что в ряде случаев эти системы моделирования трудно использовать. Необходимо расширить структуру самого транзакта, сделать ее «информационно подкрашенной» и при этом модифицировать алгоритмы приборов СМО (расширив их до процессного способа имитации), которые понимают и реагируют на информацию, содержащуюся в самих транзактах. Возникшая необходимость расширения описательных возможностей СМ путем «информационной подкраски» транзактов определила актуальность разработки новых средств. Например, в СМ **МІСІС** [22] эта задача решалась путем введения «информационной подкраски», которая состояла в том, что кроме пары характеристик транзакта (идентификатор и приоритет) был введен адрес нахождения в базе данных модели «тела транзакта». Далее в «тела транзактов» исследователь может поместить для каждой предметной области исследований следующие характеристики:

- множество характеристик запросов внешней среды ИМ, которые влияют на траекторию движения транзакта по ИМ объекта исследователя (как это было сделано в **ПТКИ УПС** [25]);
- технологическую карту движения модели изделия сложной структуры, в которой зашифрован порядок разборки изделия на составные части, характеристик ремонта компонент изделия, порядок сборки отремонтированных компонент в единое изделие (как это было сделано в **ПТКИ СРП** [26]);
- характеристики очередных запросов на ресурсы локальных вычислительных сетей (ЛВС), которые влияют на организацию вычислительного процесса в ЛВС (как это было сделано в **ПТКИ ЛВС** [13]).

8.4. Возможности программно-технологических комплексов имитации

В основе упомянутых программно-технологических моделирующих комплексов имитации (**ПТКИ**) используется система моделирования **МІСІС** [22]. Перечислим основные описательные и технологические характеристики СМ **МІСІС**. В распоряжение пользователя СМ **МІСІС** предоставляется язык разработки элементов ИМ. Он позволяет описывать программы обслуживания устройств (процессов ИМ),

классов транзактов (TR_i) сложной структуры и очередей к устройствам, обслуживающим TR_i . Пользователь может описывать алгоритмы своих процессов на языке СИ, а также использовать обширную библиотеку функций языка моделирования. С помощью подсистемы *технология* из программы на языке СИ, представляющей собой смесь последовательностей операторов языка СИ, управляющей программы организации квазипараллелизма и функций языка моделирования, пользователь получает объектный модуль ИМ. Далее, скомпоновав программу с ядром СМ МІСІС, исследователь уже в среде технологической оболочки системы моделирования окончательно формирует СИ-программу, которая затем превращается в программу ИМ объекта исследования. Поскольку СМ МІСІС реализует транзактно-процессный способ имитации, то ИМ сложной системы в среде СМ МІСІС представляет собой множество процессов, имитирующих работу у активных компонентов СС, обслуживающих в ИМ множество транзактов сложной структуры $\{TR_i\}$. В «теле» каждого TR_i исследователь может расположить: программу обработки информации PR_i на языке СИ, находящейся в базе данных имитационной модели (БДИМ); поле накопления статистики (ST_i) обслуживания в ИМ транзакта процессами и нахождения его в очередях к процессам; множество параметров модели данных $\{ПД_i\}$, отображаемых «информационно подкрашенным» TR_i . Сами активные компоненты ИМ СС являются процессами (PR_j), состоящими из набора активностей $АКТ_{jk}$ (где k – номер активности в процессе PR_j). Под активностями ($АКТ_{ik}$) понимаем реализованный в виде операторов и функций языка программирования СИ некоторый алгоритм, аппроксимирующий функциональные действия k -го номера в составе процесса номера j . В СМ МІСІС [22] цепочка активностей составляется для каждой пары: j – тип обслуживающего устройства (процесса), i – тип обслуживающего транзакта. Способ обращения к данным ИМ является стандартным для языка СИ.

Таким образом, вместо приборов, используемых для описания процесса обслуживания «информационно подкрашенных» транзактов, используются типовые процессы, обладающие типовыми алгоритмами обслуживания. Поэтому за счет расширения описательных возможностей СМ МІСІС и разработки типовых алгоритмов процессов были разработаны новые средства автоматизации ИЭ в виде семейства ПТКИ, ориентированных на каждый тип СС. Поскольку многие технологические этапы ИЭ для всех типов СС совпадают, но различаются именно алгоритмами обслуживания процессов транзактов, то расширение возможностей СМ МІСІС осуществляется за счет

объединения: проблемной, ориентированной на классе СС, части СМ МІСІС со стандартной частью в составе универсальной управляющей программы моделирования, реализующей транзактно-процессный способ имитации; подсистем, реализующих транзактно-процессный способ имитации; подсистем, реализующих стандартные функции организации ИЭ. Такой подход позволил создать три версии СМ в виде трех **ПТКИ** для исследования и модификации технологических процессов: производства услуг почтовой связи (**ПТКИ УПС** [25]); ремонта изделий сложной структуры (**ПТКИ СРП** [26]); имитации обработки информации в локальных вычислительных сетях (**ПТКИ ЛВС**) [13]. В качестве «клумбы», на которой расположены все три версии ПТКИ, используется технологическая оболочка **СМ МІСІС**.

Другой подход к организации ИЭ с моделями технологических процессов производства реализован в **ПТКИ** технологических процессов производства (**ТПП**). Основой комплекса является система автоматизации имитационного моделирования (**САИМ** [34]), реализующая агрегатно-процессный способ имитации. Перечислим основные описательные и технологические возможности **СМ САИМ**.

Объектами имитации являются вероятностные технологические производства (**ВТПП**), представляющие собой выполнение последовательностей микротехнологических операций ($MTXO_{ij}$). При параллельно-последовательном характере их выполнения в вероятностном технологическом процессе производства используется аппарат сетевого планирования на основе процедуры Монте-Карло [37]. При представлении объектов графовой структуры аппаратом формализации вероятностных сетевых графиков (**ВСГР**) узлами являются агрегаты-события ($ASOB_i$), а дугами служат агрегаты-имитаторы микротехнологических операций ($AMTXO_{ij}$). Алгоритмическая часть агрегатов реализована в среде системы программирования **Delphi** [54]. Поэтому **САИМ** представляет собой расширение объектно-ориентированного языка программирования **Object Pascal** за счет добавления типовых операторов организации квазипараллелизма между компонентами модели на основе агрегатно-процессного способа имитации СС.

В состав **ПТКИ ТПП** кроме **САИМ** входят подсистемы: реализации моделирования вероятностных сетевых графиков (**ВСГР**) на основе процедуры Монте-Карло (**PS.MONTEK**); обработки результатов имитации **ВСГР** (**PS.OBRABOT**); визуализации результатов ИЭ (**PS.VIZUAL**), библиотека агрегатов-имитаторов $AMTXO_{ij}$ и $ASOB_i$ (**LIB.AGREG**), библиотека версий имитационных моделей **ВСГР**, адекватных конкретным структурам ИМ **ТПП** (**LIB.MODEL**), информационная база данных комплекса (**IBD SGR**).

Как видим, технология исследования с помощью **ПТКИ ТПП** ориентирована на технологическое имитационное моделирование производственных систем в тех случаях, когда динамику функционирования ВТПП можно имитировать только на основе аппарата сетевого планирования при высоком уровне автоматизации и простоте описания ВТПП с помощью ИМ ВСГР. Отметим гибкость изменения структуры ИМ ТПП с помощью **ПТКИ ТПП** [26]. Возможность организации оперативного исследования динамики использования ресурсов ВТПП и текущего анализа надежности оборудования позволяет использовать предметно-ориентированный комплекс **ПТКИ ТПП** при проектном моделировании ТПП.

ЗАКЛЮЧЕНИЕ

В монографии «Математические основы имитационного моделирования сложных систем» приведены сведения из соответствующих разделов прикладной математики, которые читатели могут подчеркнуть из курсов по теории вероятности, математической статистики и математического моделирования систем. Сведения эти предлагаются в справочном виде и поэтому автор в каждой главе излагал набор знаний, который в совокупности составляет математические основы имитационного моделирования сложных систем. Безусловно без знания математических основ имитации СС невозможно разобраться с методами постановки имитационных экспериментов на ЭВМ.

Итак, цель монографии «Математические основы имитационного моделирования сложных систем» состояла в том, чтобы дать читателям необходимый минимум знаний для изучения технологии имитации сложных систем. Автор на основе личного опыта, накопленного в течении 30 лет использования методик имитации СС, постарался изложить знания, необходимые студентам, аспирантам и научным работникам, занимающихся имитацией на ЭВМ функционирования сложных систем.

Для более углубленного изучения базовых вопросов имитационного моделирования автор предлагает ряд ссылок на литературные источники.

СПИСОК ЛИТЕРАТУРЫ

1. Автоматизация проектирования вычислительных систем. Языки, моделирование и базы данных / под ред. М. Брейера. – М.: Мир, 1979. – 463 с.
2. Андрианов, А. И. Система моделирования на базе языка СИМУЛА-67 для БЭСМ-6 и ЕС ЭВМ / А. Н. Андрианов [и др.] // Моделирование дискретных управляющих и вычислительных систем: Тез. докл. III Всесоюз. семинара. – Свердловск, 1981. – С. 44–45.
3. Андрианов, А. Н. Средства раздельной компиляции СИМУЛА-программ на БЭСМ-6-й ЕС ЭВМ / А. Н. Андрианов, С. П. Бычков, А. И. Хорошилов. – М.: ИПМ АН СССР, 1982. – 40 с.
4. Асатурян, В. И. Теория планирования эксперимента / В. И. Асатурян. – М.: «Радио и связь», 1983. – 213 с.
5. Бусленко, В. Н. Автоматизация имитационного моделирования сложных систем / В. Н. Бусленко. – М.: Наука, 1977. – 240 с.
6. Бусленко, Н. П. Лекции по теории сложных систем / Н. П. Бусленко, В. В. Калашников, И. Н. Коваленко. – М.: Сов. Радио, 1973. – 439 с.
7. Бычков, С. П. Входной язык компилятора с языка ДИНАМО для БЭСМ-6 / С. П. Бычков, О. В. Перевицкий, В. А. Фисун // Математическое обеспечение моделирования сложных систем: тез. докл. Всесоюз. конф. – Киев, 1977. – Ч. 1. – С. 119.
8. Вагнер, Г. Основы исследования операций: в 3 т. Т. 2. / Г. Вагнер. – М.: «Мир», 1973. – 481 с.
9. Глушков, В. М. СЛЕНГ – система программирования для моделирования дискретных систем / В. М. Глушаков [и др.]. – Киев: Ин-т кибернетики АН УССР, 1969. – 320 с.
10. Гмурман, В. Е. Теория вероятностей и математическая статистика: уч. пособие для ВУЗов. – М.: Выш. шк, 1998. – 479 с.
11. Гуревич, Е. И. Система функционального моделирования цифровых систем МОДИС-ВЕС / Е. И. Гуревич, Ю. А. Татарникова // Моделирование дискретных управляющих и вычислительных систем: тез. докл. IV Всесоюз. семинара. – Свердловск, 1984. – С. 31–34.
12. Демиденко, О. М. Проектное моделирование вычислительного процесса в локальных вычислительных сетях / О. М. Демиденко, И. В. Максимей. – Мн.: Беларус. навука, 2001. – 252 с.
13. Демиденко, О. М. Технология мониторинга и адаптации вычислительного процесса под рабочую нагрузку на локальную вычислительную сеть / О. М. Демиденко. – Мн.: Беларус. навука. 2002. – 143 с.

14. Ермаков, С. М. Курс статистического моделирования / С. М. Ермаков, Г. А. Михайлов. – М.: Наука, 1976. – 319 с.
15. Искусственный интеллект: в 2 кн. Кн. 1.: Системы общения и экспертные системы / под ред. Д. А. Поспелова. – М.: Радио и связь, 1990. – 464 с.
16. Искусственный интеллект: в 2 кн. Кн. 2: Модели и методы / под ред. Д. А. Поспелова. – М.: Радио и связь, 1990. – 304 с.
17. Калашников, В. В. Вопросы разработки имитационных систем / В. В. Калашников [и др.] // Электронная техника. Сер. «Экономика и системы управления». – 1983. – Вып. 1. – С. 71–87.
18. Калашников, В. В. КОМПАС – программный язык описания агрегативных систем / В. В. Калашников, Б. В. Немчинов, С. Т. Петров // Моделирование дискретных управляющих и вычислительных систем: тез. док. IV Всесоюз. семинара. – Свердловск, 1984. – С. 44–46.
19. Калиниченко, Л. А. СКИФ – независимый от языков программирования пакет программ, ориентированный на имитационное моделирование систем с дискретными событиями / Л. А. Калиниченко, В. М. Щербин // Управляющие системы и машины. – 1976. – № 4. – С. 37–42.
20. Кельтон, В. Имитационное моделирование СС / В. Кельтон, А. Лоу. – 3-е изд. – СПб.: Питер; Киев: Изд. группа ВНИ, 2004. – 847 с.
21. Кендал, М. Дж. Многомерный статистический анализ и временные ряды / М. Дж. Кендал, А. Стьюарт. – М.: Наука, 1976. Т. 1. – 736 с.
22. Киндлер, Е. Языки моделирования / Е. Киндлер. – М.: Энергоатомиздат, 1985. – 288 с.
23. Кононенко, И. А. Базовый язык моделирования DISLIN / И. А. Кононенко, О. И. Ситников // Математическое обеспечение автоматического проектирования (моделирование дискретных систем): Труды ИММ УНЦ АН СССР. – Свердловск, 1981. – С. 11–38.
24. Коршунов, Ю. М. Математические основы кибернетики: учеб. пособие для вузов / Ю. М. Коршунов. – М.: Энергоиздат, 1987. – 496 с.
25. Левчук, В. Д. Базовая схема формализации системы моделирования MICIS 4 / В. Д. Левчук // Проблемы программирования. – Киев. – 2005, № 1. – С. 85–96.
26. Левчук, В. Д. Инструментарий имитационного моделирования дискретных вероятностных технологических процессов производства / В. Д. Левчук, И. В. Максимей, Г. А. Терещенко // Проблемы программирования. – 2003. – № 4. – С. 92–99.
27. Левчук, В. Д. Программно-технологические комплексы имитации сложных дискретных систем: монография / В. Д. Левчук,

И. В. Максимей; М-во образования РБ; Гом. гос. ун-т им. Ф. Скорины. – Гомель: ГГУ им. Ф. Скорины, 2006. – 203 с.

28. Литвинов, В. В. Математическое обеспечение проектирования вычислительных систем и сетей / В. В. Литвинов. – Киев: Техника, 1982. – 176 с.

29. Максимей, И. В. Возможности агрегатной системы моделирования вероятностных технологических процессов производства / И. В. Максимей [и др.] // Проблемы программирования. – Киев, 2006. – № 2.

30. Максимей, И. В. Задачи и модели исследования операций: в 2 ч. Ч. 2: Методы нелинейного и стохастического программирования: учеб. пособие / И. В. Максимей, В. С. Серегина. – Гомель: БелГУТ, 1999. – 103 с.

31. Максимей, И. В. Имитационная модель технологических процессов на городском транспорте / И. В. Максимей [и др.] // Информатика. – 2006. – № 2. – С. 124–134.

32. Максимей, И. В. Имитационное моделирование на ЭВМ / И. В. Максимей. – М.: «Радио и Связь», 1988. – 232 с.

33. Максимей И. В. Моделирующий комплекс PLSIM / И. В. Максимей [и др.] // Теория сложных систем и методы их моделирования: тр. семинара / ВНИИСИ. – М., 1982. – С. 107–111.

34. Максимей, И. В. Об автоматизации моделирования сетей СМО с динамической структурой / И. В. Максимей, Ю. А. Семишин // Управляющие системы и машины. – 1981. – № 6. – С. 16–22.

35. Максимей, И. В. Система автоматизации моделирования вероятностных технологических процессов, реализующая агрегатный способ имитации / И. В. Максимей [и др.] // Проблемы программирования. – Киев. – 2004. – № 4. – С. 25–31.

36. Максимей, И. В. Средства технологической поддержки имитационного эксперимента: учеб. пособие / И. В. Максимей, Е. И. Сукач. – Гомель: ГГУ им. Ф. Скорины, 2002. – 105 с.

37. Максимей, И. В. Технология имитации и обработка результатов в операционной среде агрегатной системы автоматизации моделирования вероятностных технологических процессов производства / И. В. Максимей, А. Н. Гончаров, В. С. Смородин // Реєстрація, зберігання і обробка інформації. – Киев, 2006.

38. Максимей, И. В. Языковые и технологические возможности МК АСИМ для построения имитационных моделей больших систем / И. В. Максимей, Ю. А. Семишин // Моделирование дискретных управляющих и вычислительных систем: тез. докл. IV Всесоюз. семинара. – Свердловск, 1984. – С. 61–63.

39. Мановицкий, В. И. Система имитационного моделирования дискретных процессов ДИСМ / В. И. Мановицкий, Е. М. Сурков. – Киев; Одесса: Виша школа, 1981. – 94 с.
40. Маркова, Е. В. Комбинаторные планы в задачах многофакторного эксперимента / Е. В. Маркова, А. Н. Лысенков. – М.: Наука, 1979. – 345 с.
41. Марьянович, Т. П. Пакет моделирования НЕДИС-ЕС / Т. Т. Марьянович, М. А. Сахнюк, А. В. Шемшур // Теория и методы математического моделирования: тез. докл. VII Всесоюз. совещания. – Куйбышев: Наука, 1978. – С. 165–166.
42. Нечипуренко, М. И. Язык моделирования ДИС / М. И. Нечипуренко, А. Х. Хайрутдинов, Г. В. Шабров // Тез. докл. 5-й Межвуз. конф. по физическому и математическому моделированию / МЭИ. – М., 1968. – С. 77–84.
43. Окольнішников, В. В. Система автоматизации дискретного моделирования МОДЕЛЬ-6 / В. В. Окольнішников // Математическое обеспечение моделирования сложных систем: тез. докл. Всесоюз. конф. – Киев, 1977. – С. 140–141.
44. Основы теории вычислительных систем / под ред. С. А. Майорова: учеб. пособие для вузов. – М.: Высшая школа, 1978. – 408 с.
45. Поляк, Ю. Г. Вероятностное моделирование на электронных вычислительных машинах / Ю. Г. Поляк. – М.: Сов. Радио, 1971. – 399 с.
46. Поляков, А. К. Архитектура многофункциональной системы моделирования МПЛ/1 / А. К. Поляков // Моделирование дискретных управляющих и вычислительных систем: тез. докл. 3-го Всесоюз. семинара. – Свердловск, 1981. – С. 42–43.
47. Попков, Э. Ф. Экспертные системы: решение неформализованных задач в диалоге с ЭВМ / Э. Ф. Попков. – М.: Наука, 1987. – 283 с.
48. Пранявичюс, Г. Модели и методы исследования вычислительных систем / Г. Пранявичюс. – Вильнюс: Мокслас, 1982. – 227 с.
49. Рыжиков, Ю. И. Имитационное моделирование. Теория и технология / Ю. И. Рыжиков. – СПб.: Корона; М.: Альтекс, 2004. – 384 с.
50. Сергиенко, И. В. Вопросы исследования систем обработки данных и повышения их эффективности / И. В. Сергиенко, В. В. Скопецкий // Кибернетика. – 1977. – 6. – С. 61–72.
51. Советов, Б. Я. Моделирование систем: учеб. для вузов / Б. Я. Советов, С. Л. Яковлев. – М.: Высш. шк., 1985. – 420 с.
52. Технология системного моделирования / под общ. ред. С. В. Емельянова, В. В. Калашникова. – М.: Машиностроение; Берлин: Техника, 1983. – 510 с.

53. Фараонов, П. В. Delphi 5. Руководство разработчика баз данных / П. В. Фараонов, В. В. Шумаков. – М., 2000. – 640 с.
54. Федоров, В. В. Теория оптимального эксперимента / В. В. Федоров. – М.: Наука, 1971. – 270 с.
55. Фейбел, В. Энциклопедия современных сетевых технологий / В. Фейбел. – М.: Комиздат, 1998. – 687 с.
56. Феррари, Д. Оценка производительности вычислительных систем / Д. Феррари. – М.: Мир, 1982. – 576 с.
57. Харин, Ю. С. Основы имитационного и статистического моделирования: уч. пособие / Ю. С. Харин [и др.]. – Мн.: Дизайн ПРО, 1997. – 288 с.
58. Шевченко, Д. Н. Теория вероятностей и математической статистики: уч. пособие / Д. Н. Шевченко. – Гомель: «БелГУТ», 2006. – 318 с.
59. Шеннон, Р. Имитационное моделирование систем / Р. Шеннон // Искусство и наука. – М.: Мир, 1978. – 417 с.
60. Шрайбер, Т. Дж. Моделирование на GPSS / Т. Дж. Шрайбер. – М.: Машиностроение, 1980. – 592 с.

Оглавление

Список условных обозначений и терминов	3
Введение	4
1. ЭЛЕМЕНТЫ ТЕОРИИ ВЕРОЯТНОСТЕЙ	7
1.1. Определение вероятности случайных событий	7
1.2. Типы случайных величин	13
1.3. Дискретные случайные величины	18
1.4. Непрерывные случайные величины	20
1.5. Многомерные случайные величины	27
1.5.1. Распределение многомерной случайной величины ...	27
1.5.2. Функция распределения двумерной случайной величины (ξ, η)	28
1.5.3. Независимые случайные величины	29
2. ОСНОВНЫЕ ПОНЯТИЯ МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ	32
2.1. Выборочный метод при исследовании вероятностных систем	32
2.2. Элементы статистического оценивания результатов исследований	35
2.3. Статистическая проверка гипотез	43
2.4. Проверка непараметрических гипотез	51
2.5. Элементы статистического анализа данных	55
3. МЕТОД СТАТИСТИЧЕСКИХ ИСПЫТАНИЙ КАК ОСНОВА МОДЕЛИРОВАНИЯ ВЕРОЯТНОСТНЫХ СИСТЕМ	71
3.1. Особенности использования вероятностных моделей сложных систем	71
3.2. Виды и алгоритмы реализации единичных жребиев на ЭВМ	78
3.3. Моделирование на ЭВМ псевдослучайных величин	81
3.4. Примеры формирования и использования псевдослучайных величин	85
3.5. Методы исключения и суперпозиции при моделировании случайных величин	87
3.6. Моделирование случайных величин, имеющих мультимодальную функцию плотности распределения	91
3.7. Пример статистического моделирования процесса обстрела ракетами площадной цели сложной формы	93
4. ПЛАНИРОВАНИЕ ВЕРОЯТНОСТНЫХ ЭКСПЕРИМЕНТОВ НА ЭВМ	97

4.1. Кибернетический подход к исследованию на вероятностных моделях сложных систем	97
4.2. Концептуальная модель исследования вероятностной сложной системы	100
4.3. Основные определения планирования экспериментов на ЭВМ с моделями вероятностных систем	102
4.4. Планы построения регрессионной модели поведения сложной системы	107
4.5. Другие планы эксперимента на ЭВМ моделей сложных систем	114
4.6. Планирование имитационного эксперимента	118
5. ОСНОВЫ ОПТИМИЗАЦИИ ВАРИАНТОВ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ	124
5.1. Методы векторной оптимизации сложных систем	124
5.1.1. Выделение главного критерия	124
5.1.2. Метод последовательных уступок	125
5.1.3. Методы приведения откликов к одному типу и масштабу	125
5.1.4. Метод составного критерия	126
5.1.5. Нормативные методы	127
5.1.6. Методы логического объединения критериев	127
5.2. Оптимизация нелинейной функции отклика от одной переменной	128
5.3. Максимизация функции отклика от многих переменных без ограничений	134
5.4. Нахождение экстремума функции отклика от многих переменных при наличии ограничений	141
6. ЭЛЕМЕНТЫ ПРИНЯТИЯ РЕШЕНИЙ ПРИ ВЕРОЯТНОСТНОМ МОДЕЛИРОВАНИИ СЛОЖНЫХ СИСТЕМ	145
6.1. Виды стратегий исследователя при анализе сложных систем.	145
6.2. Принятие решений в условиях неопределенности и риска .	146
6.3. Критерии принятия решений в условиях неопределенности и риска	150
6.4. Методы поиска решений в экспертных системах	153
7. СПОСОБЫ ОРГАНИЗАЦИИ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ	164
7.1. Понятие о модельном времени	164
7.2. Структурная схема имитационных моделей сложных систем.	170
7.3. Организация квазипараллелизма в имитационных моделях путем просмотра активностей	174

7.4. Организация квазипараллелизма в имитационных моделях способом составления расписания событий	177
7.5. Транзактный способ организации квазипараллелизма	180
7.6. Агрегатный способ организации квазипараллелизма	184
7.7. Процессный способ организации квазипараллелизма	189
7.8. Транзактно-процессный способ имитации сложных систем .	195
7.9. Агрегатно-процессный способ имитации сложных систем .	201
8. СРЕДСТВА АВТОМАТИЗАЦИИ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ	207
8.1. Возможности языков и систем моделирования сложных систем	207
8.2. Классификация языков имитационного моделирования сложных систем	214
8.3. Инструментальные и технологические возможности средств автоматизации моделирования сложных систем	218
8.4. Возможности программно-технологических комплексов имитации	231
Заключение	235
Список литературы	236

Научное издание

Максимей Иван Васильевич

**МАТЕМАТИЧЕСКИЕ ОСНОВЫ
ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ
СЛОЖНЫХ СИСТЕМ**

Подписано в печать 2.12.2014. Формат 60x84 1/16
Бумага офсетная. Ризография. Усл. печ. л. 14,4.
Уч.-изд. л. 15,8. Тираж 100 экз. Заказ 674.

Издатель и полиграфическое исполнение:
учреждение образования
«Гомельский государственный университет
имени Франциска Скорины».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 1/87 от 18.11.2013.

Специальное разрешение (лицензия) № 02330 / 450 от 18.12.2013.

Ул. Советская, 104, 246019, Гомель.