

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Р. И. МАКАРОВ Е. Р. ХОРОШЕВА

МАТЕМАТИЧЕСКИЕ ОСНОВЫ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ

Учебное пособие



Владимир 2019

УДК 519.87
ББК 22.183.5
М15

Рецензенты:

Доктор технических наук, профессор
зав. кафедрой вычислительной техники и систем управления
Владимирского государственного университета
имени Александра Григорьевича и Николая Григорьевича Столетовых
В. Н. Ланцов

Доктор технических наук, профессор
профессор кафедры инновационного предпринимательства
Московского государственного технического университета
имени Н. Э. Баумана (национальный исследовательский университет)
Д. В. Александров

Издается по решению редакционно-издательского совета ВлГУ

Макаров, Р. И. Математические основы моделирования
М15 информационных процессов и систем : учеб. пособие / Р. И. Макаров, Е. Р. Хорошева ; Владим. гос. ун-т им. А. Г. и Н. Г. Столетовых. – Владимир : Изд-во ВлГУ, 2019. – 132 с.
ISBN 978-5-9984-1055-0

Приведены общие сведения о математических основах моделирования информационных процессов и систем. Рассмотрены аналитические и имитационные методы моделирования. Теоретический материал подкреплён примерами.

Предназначено для магистрантов первого курса направления подготовки 09.04.02 – Информационные системы и технологии очной формы обучения.

Рекомендовано для формирования профессиональных компетенций в соответствии с ФГОС ВО.

Табл. 6. Ил. 28. Библиогр.: 30 назв.

УДК 519.87
ББК 22.183.5

ISBN 978-5-9984-1055-0

© ВлГУ, 2019

ВВЕДЕНИЕ

Дисциплина «Математические основы моделирования информационных процессов и систем» направлена на формирование представлений магистрантов о математических основах моделирования информационных процессов и систем как метода научного познания, при этом компьютер рассматривается как средство познания и научно-исследовательской деятельности.

Дисциплина знакомит магистрантов с методами построения моделей сложных систем и процессов, возможностями средств моделирования, оценкой качества моделей, проведением экспериментов для оценки эффективности сложных систем, применением моделей в задачах управления.

В результате освоения программы магистратуры у выпускника должны быть сформированы общекультурные, общепрофессиональные и профессиональные компетенции. Общепрофессиональные компетенции следующие: способность воспринимать математические, естественно-научные, социально-экономические и профессиональные знания, умение самостоятельно приобретать, развивать и применять их для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте (ОПК-1).

Выпускник, освоивший программу магистратуры, должен обладать профессиональными компетенциями, соответствующими виду (видам) профессиональной деятельности, на который (которые) ориентирована программа магистратуры.

Научно-исследовательская деятельность предполагает:

– умение проводить разработку и исследование теоретических и экспериментальных моделей объектов профессиональной деятельности в различных областях, а также на предприятиях различного про-

филя и все виды деятельности в условиях экономики информационного общества (ПК-8);

– умение проводить разработку и исследование методик анализа, синтеза, оптимизации и прогнозирования качества процессов функционирования информационных систем и технологий (ПК-9);

– умение моделировать процессы и объекты на базе стандартных пакетов автоматизированного проектирования и исследований (ПК-10).

Глава 1

МОДЕЛИРОВАНИЕ КАК МЕТОД ПОЗНАНИЯ

1.1. Модель и ее свойства

Модели играют в жизни человека чрезвычайно важную роль. Достаточно сказать, что в основе поведения человека как системы разумной лежит субъективная модель мира, создаваемая им на протяжении всей жизни на основе анализа личного и социального опыта. Какими бывают модели? И какие средства формализации используют для представления знаний о системах?

Для начала обратимся к понятию модели и ее свойствам. *Модель* – это совокупность логических, математических или иных объектов, связей и соотношений, отображающих с необходимой или предельно достижимой степенью подобия некоторый фрагмент реальности, подлежащий изучению, а также описание всех существенных свойств моделируемого объекта. Можно рассматривать различные аспекты подобия между моделью и фрагментами реального мира:

- физическое подобие, когда модель и объект имеют близкую физическую сущность;
- функциональное подобие, когда сходны их функции;
- динамическое подобие, проявляющееся в сходстве динамики изменения состояния объекта;
- топологическое подобие, проявляющееся в сходстве пространственной (в том числе организационной) структуры и иные.

Соответственно различают физические, функциональные, динамические, топологические и иные виды моделей. Кроме того, по принципу реализации выделяют натурные, полунатурные, имитационные и теоретические модели. В зависимости от обстоятельств (целей, условий) в практике используют разные модели.

Степень формализации моделей может варьироваться в широких пределах: от моделей, не подвергнутых процедурам формализации, до моделей строго формальных. Выбор формальных средств, ис-

пользуемых для представления моделей, не произволен и определяется двумя аспектами – компонентами модели [8]:

- моделью интерпретации, или интерфейсным компонентом (характеризующим процесс двунаправленного взаимодействия с потребителем, в роли которого может выступать как человек, так и автоматизированная система, реализующая функции ввода и считывания данных);
- сущностным компонентом (характеризующим специфику моделируемого фрагмента реальности, закономерности его функционирования, структуры и т. п.).

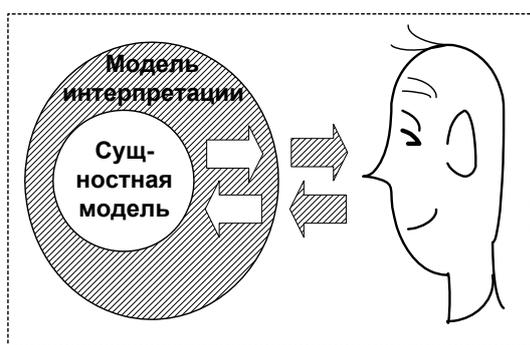


Рис. 1.1. Инкапсуляция моделей

модели, а сущностная модель фрагмента реальности (объекта, процесса, явления и т. п.) заключена внутрь этой оболочки (рис. 1.1).

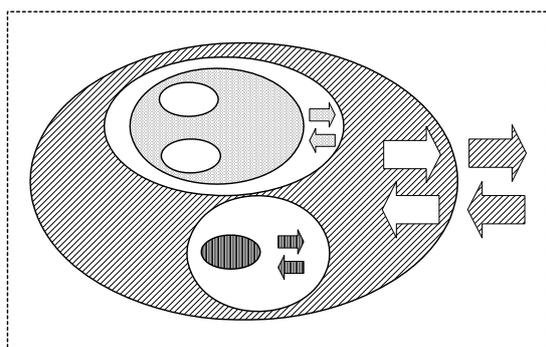


Рис. 1.2. Сложная модель как иерархия модельных объектов

принцип матрешки широко используется при синтезе моделей самой различной семантики.

Сущностный компонент модели – отражение некоторых сущностей, процессов и явлений реального мира. В отличие от модели ин-

Если взглянуть на любую модель с точки зрения специалиста в области разработки программного обеспечения, знакомого с объектным подходом к программированию, то модель предстанет в виде совокупности инкапсулированных (помещенных одна в другую) моделей. При этом модель интерпретации (адаптации, интерфейса) представляет собой внешнюю оболочку

В отличие от простых, одноуровневых моделей сложные модели имеют несколько уровней вложенности, и на каждом уровне вложенности может существовать несколько разнородных моделей, однако и для них изложенный выше подход (выделение интерфейсного и сущностного компонентов) остается справедливым (рис 1.2). Принцип

терпретации сущностная модель не может быть отображена с применением произвольно выбранных средств формализации предметной области. Для каждой предметной области существует некоторый диапазон приемлемых средств формального выражения отношений и сущностей реального мира, различающихся степенью детализации их выражения. Степень же детализации, с одной стороны, определяется спецификой задачи, а с другой – спецификой системы или процесса.

Перечислим наиболее значимые факторы, оказывающие влияние на выбор адекватной степени детализации модели:

- назначение модели и цель исследования (аналитическая, прогностическая модель, исследовательская (научная) модель, кибернетическая (управленческая) модель);
- избирательность исследования (выражению средствами модели подлежат система или процесс в целом или их отдельные аспекты);
- степень полноты знаний о системе или процессах, подлежащих моделированию;
- динамические характеристики моделируемой системы/процесса;
- структура моделируемой системы;
- условия наблюдаемости (непрерывное, кусочно-непрерывное, дискретное);
- характеристика среды и параметры возмущающих воздействий;
- время, доступное для синтеза модели/производства вычислений;
- динамические и точностные характеристики системы сбора информации (точность результатов не может быть выше точности измерений);
- динамические и точностные характеристики системы управления (чаще всего нет смысла анализировать динамические и статические параметры системы или процесса, если отсутствуют средства управления, обеспечивающие необходимую скорость и точность доведения управляющих воздействий);
- точностные характеристики методов, используемых для обработки данных;
- характеристики платформы, используемой для реализации модели (в случае применения специальных программных и технологических средств);

– точностные характеристики реализации методов с учетом ограничений технологической платформы, используемой для их реализации и иные.

Располагая знаниями высшего уровня (зная закон), исследователь менее всего стеснен в выборе средств моделирования. Однако в большинстве случаев такой свободы нет. Например, недостаточный объем знаний о системе не позволяет построить модель более высокой степени формализации, нежели вербальная или логико-лингвистическая модель типа сценария. Такая ситуация возникает тогда, когда причинно-следственные отношения не выявлены, структура системы и отношения между компонентами установлены лишь частично и подлежат уточнению, что соответствует знаниям уровня гипотезы или теории в предложенной иерархии.

В то же время, даже располагая знанием закона, исследователь не всегда может выбрать произвольный способ формального представления системы, поскольку формальный аппарат, как правило, не универсален и привязан к конкретной предметной области и условиям наблюдений.

На практике чаще встречается ситуация, когда некоторая формальная система позволяет адекватно описывать феномены различного происхождения: так обстоит дело со многими математическими формальными системами, полученными в результате развития естественно-научных дисциплин. К числу формальных методов относят аналитические, вероятностные и статистические, теоретико-множественные и логические, лингвистические и семиотические, а также графические и иные методы. Формальные модели, построенные с применением этих методов, получают названия, сходные с названиями использованных методов.

1.2. Моделирование. Принцип моделируемости

Моделирование опирается на принцип изоморфизма (многообразия): замены объекта на адекватную модель. Соотношение объекта и модели определяется степенью ее адекватного описания научными или иными средствами (вербально, графически, математически и т. п.) [9].

Моделирование – основополагающий метод исследования больших и сложных систем. В теории систем утверждается, что ника-

ких других средств для качественного и эффективного описания больших и сложных систем, кроме моделирования, не существует. В современной науке укоренилось представление, что всякое познание является моделированием.

Принцип моделируемости заключается в том, что сложная система может быть представлена конечным множеством моделей, каждая из которых отражает определенную грань сущности сложной системы. Этот важный принцип дает возможность исследовать определенное свойство сложной системы при помощи одной или нескольких упрощенных (узкоориентированных) моделей. Модель, ориентированная на определенную группу свойств сложной системы, всегда проще самой системы. Создание полной модели для сложной системы бесполезно, так как такая модель столь же сложна, как и система.

Принцип моделируемости включает в себя несколько постулатов [6].

Постулат действий. Для изменения поведения системы требуется прирост воздействия, превосходящий некоторое пороговое значение.

Изменение поведения сложной системы может быть связано с энергетикой, веществом и информацией, которые, накапливаясь, проявляют свое влияние скачкообразно, путем качественного перехода. Одновременное энергетическое и информационное воздействие может привести к такому же результату, как энергетическое воздействие более высокого уровня.

Таким образом, порог есть функция трех переменных: количества определенного вещества, количества энергии и количества определенной информации.

Реакция системы на внешнее воздействие носит пороговый характер.

Постулат неопределенности. Существует область неопределенности, в рамках которой свойства системы могут быть описаны только вероятностными характеристиками. Повышение точности определения (измерения) какого-либо количественно описываемого свойства сложной системы сверх некоторого предела влечет за собой понижение возможной точности определения (измерения) другого свойства. Одновременно измерить значения двух (или более) параметров с точностью, превышающей определенный уровень, невозможно.

Максимальная точность определения (измерения) свойств системы зависит от присущей данной системе области неопределенности, внутри которой повышение точности определения (измерения) одного свойства влечет за собой снижение точности определения другого (других).

Постулат дополнительности. Сложные системы, находясь в различных средах (ситуациях), могут проявлять разные системные свойства, в том числе альтернативные (т. е. несовместимые ни в одной из ситуаций по отдельности). Наблюдатель воспринимает одни грани сущности в одних условиях и другие сущности в других.

Постулат многообразия моделей. Характеристики системы на всех уровнях определяются с помощью множества моделей, которые в общем случае различаются используемыми математическими зависимостями и физическими закономерностями. Выбор моделей зависит от цели анализа и синтеза и особенностей исследуемой системы.

Постулат согласования уровней. Требования к системе, формируемые на любом уровне, выступают как условия (или ограничения) выбора частных моделей и предельных возможностей системы на нижележащих уровнях. В случае невозможности выполнения требований корректируются условия.

Постулат внешнего дополнения. Истинность результатов, получаемых на каждом уровне, проверяется с использованием исходных данных, моделей и методов вышележащих уровней. Данный постулат – основополагающий в общей теории систем, и его соблюдение – необходимое условие получения правильных решений на всех уровнях исследования системы.

Постулат достаточности. Последовательность уровней (этапов) определения требуемых характеристик в процессе совершенствования сложной системы выбирается по возрастанию затрат на улучшение системы при проверке достаточности принимаемых решений по заданным критериям эффективности. Постулат достаточности реализуется, как правило, при использовании критериев пригодности и разработке соответствующих моделей, с помощью которых принимаются конструктивные решения на каждом уровне выбора характеристик системы.

Постулат проверенного методического обеспечения. Для анализа и синтеза системы управления необходимо использовать хорошо отработанные и экспериментально проверенные модели и методики, обеспечивающие отдельные характеристики системы в заданные сроки и с требуемой точностью.

1.3. Основные методы формализации предметной области исследований. Поэтапный синтез моделей систем и процессов

Существует много методов формализации предметной области исследований. Рассмотрим некоторые из них и область применения их в задачах моделирования.

Первичная *вербальная модель* представляет собой словесный портрет системы и проблемной ситуации, т. е. документ, аналогичный проекту технического (информационно-поискового и т. п.) задания, разрабатываемого организацией-заказчиком. Процесс синтеза первичной вербальной модели может проходить и при участии сторонних (приглашенных) специалистов. К этому шагу приходится прибегать в тех случаях, когда организация не располагает информацией, достаточной для принятия решения или выявления сущности противоречий.

Вербальная модель создается для сокращения неопределенности, компенсации неполноты знаний и формирования гипотезы или набора гипотез. Но первая и главная задача вербального моделирования – создание вербального описания на материальном носителе. Вербальная модель – это не обязательно исключительно текстовый документ. Она может содержать и количественные характеристики, элементы структуризации (например, таблицы, схемы и графики) [8].

Важный этап вербального моделирования – этап приведения (стандартизации) терминологии и сокращения избыточности описаний. Результат выполнения этой процедуры – вербальная модель, построенная в едином стандартизованном тезаурусе, дальнейшее использование которой упрощает решение задач автоматизации процессов анализа и перевода модели на следующий уровень формального представления.

При решении задачи синтеза баз данных и систем информационного обеспечения деловых процессов полученных на этом этапе

данных зачастую оказывается достаточно для синтеза макета информационной системы.

Логико-лингвистические и семиотические модели характеризуются более высокой степенью формализации. При построении логико-лингвистических моделей широко используется символичный язык логики и формализм теории графов и алгоритмов. Наиболее распространенный способ формального представления логико-лингвистических моделей – граф. В математической интерпретации

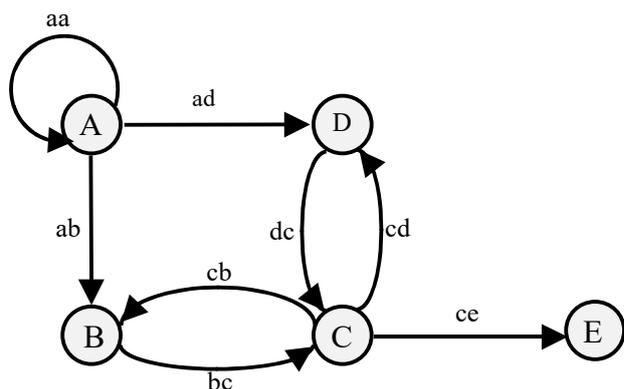


Рис. 1.3. Пример графа переходов

граф представляет собой формальную систему, описываемую как $G = (X, U)$, где X – множество вершин; U – множество ребер (дуг). Граф состоит из упорядоченных пар вершин, причем одна и та же пара может входить в множество U любое число раз, описывая различные виды отношений (рис. 1.3).

Широкое применение логико-лингвистические модели нашли в отраслях разработки программного обеспечения, управления корпоративными информационными ресурсами и многих других отраслях, где требуется определенный уровень формализации, представляющий единство строгости, интуитивной понятности и высокой выразительной способности моделей [8].

Логико-смысловые модели позволяют формировать тематически связанные описания различных аспектов проблемы (равно как и проблемы в целом) и проводить структурный анализ проблемной области. В качестве примера применения логико-смыслового моделирования можно рассматривать гипертекстовые системы, получившие широкое распространение в глобальной телекоммуникационной сети Интернет.

Логико-смысловая модель представляется в виде связного неориентированного графа, в котором вершины соответствуют высказываниям, а ребра – семантическим связям между ними. Характеристики графа используются для исследования логико-смысловой сети.

Логические модели представляют собой следующий уровень формального представления (по сравнению с логико-лингвистическими). В таких моделях естественно-языковые высказывания замещаются на примитивные – литералы, между которыми устанавливаются отношения, предписываемые формальной логикой.

Выделяют логические модели, в которых рассматриваются различные схемы логических отношений: логическое следование, включение и иные, которыми замещаются отношения, характерные для традиционной формальной логики.

Наиболее широкое распространение логические модели получили в области построения систем искусственного интеллекта, где они используются в качестве основы для производства логического вывода из системы посылок, зафиксированных в базе знаний, в ответ на внешний запрос. Неполнота и нечеткость экспертных знаний привели к развитию таких направлений логики, как многозначные логики, вероятностные логики и нечеткие логики (Fuzzy Logic, автор теории Л. Заде, 1960-е гг.) [8].

Лотфи Заде предложил для представления таких знаний математическую теорию нечетких множеств и определил операции нечеткой логики. Системы, использующие модели на базе нечеткой логики, разрабатываются специально для решения плохо определенных задач и задач с использованием неполной и недостоверной информации. Внедрение аппарата нечетких логик в технологию создания экспертных систем привело к созданию нечетких экспертных систем (Fuzzy Expert Systems).

Нечеткая логика позволяет решать широкий класс задач, не поддающихся строгой формализации: методы нечеткой логики используются в системах управления сложными техническими комплексами, функционирующими в непредсказуемых условиях (летательными аппаратами, системами наведения высокоточного оружия и т. д.).

Статистические и теоретико-вероятностные методы составляют методологическую основу одноименных видов моделирования. *Статистическая и теоретико-вероятностная модели* (стохастическая модель) – это модели, обеспечивающие учет влияния случайных факторов в процессе функционирования системы и основанные на применении статистической или теоретико-вероятностной методологии по отношению к повторяющимся феноменам. Данная модель опери-

рует количественными критериями при оценке повторяющихся явлений и позволяет учитывать их нелинейность, динамику, случайные возмущения за счет выдвижения на основе анализа результатов наблюдений гипотез о характере распределения некоторых случайных величин, сказывающихся на поведении системы.

По существу, теоретико-вероятностные и статистические модели различаются уровнем неопределенности знаний о моделируемой системе, существующим на момент синтеза модели. В случае, когда представления о системе носят скорее теоретический характер и основываются исключительно на гипотезах о характере системы и возмущающих воздействий, не подкреплённых результатами наблюдений, теоретико-вероятностная модель единственно возможная. Когда же на этапе синтеза модели уже существуют данные, полученные опытным путем, появляется возможность подкрепления гипотез за счет их статистической обработки.

Статистические модели применимы для изучения массовых явлений любой природы, включая и те, которые не относятся к категории вероятно определенных (математическая статистика приспособлена и для решения детерминированных задач). При моделировании последних статистический процесс вводится в модель искусственно для получения статистических оценок численного решения (например, точности измерения параметров детерминированного процесса). Для обработки результатов наблюдений используются методы корреляционного, регрессионного, факторного, кластерного и иных видов анализа, оперирующие статистическими гипотезами. Особая роль здесь отводится методу статистических испытаний (метод Монте-Карло).

Аналитическое математическое моделирование – это вид моделирования, в ходе которого основная роль принадлежит аналитической математической модели. Важное достоинство аналитического моделирования – возможность получения на его основе фундаментальных результатов и инвариантных зависимостей, которые могут быть распространены как на различные случаи использования моделируемой системы в тех или иных ситуациях, так и на случаи рассмотрения других систем данного класса. Основной недостаток аналитического моделирования заключается в том, что его применение к сложным системам требует существенной идеализации описания си-

стемы. Это связано с разрастанием объемов вычислений даже при несущественном усложнении описаний. Такая идеализация может приводить к неполной адекватности получаемых результатов, к тому, что эти результаты могут использоваться лишь в качестве первого приближения.

Имитационная модель – это комплексное логико-математическое представление системы, реализованное в виде программы, предназначенной для решения на ЭВМ, включающее в себя модели различного типа и рассматривающее аспект функционирования динамической системы во времени. Данный класс моделей применяется при невозможности строгого аналитического решения задачи или проведения натурального эксперимента. Имитационные модели служат для изучения поведения во времени сложной неоднородной динамической системы, относительно структуры которой существуют точные знания или детализированные гипотезы. Для каждого элемента или подсистемы моделируемой системы в памяти ЭВМ формируется блок данных, характеризующих ее текущее и предшествующие состояния, блок логических и вычислительных процедур, описывающих изменения критических параметров во времени.

Комплекс подпрограмм или относительно автономных программных агентов функционирует под управлением программы-супервизора, осуществляющей диспетчеризацию вызовов, активизирующей и приостанавливающей на время выполнение тех или иных процедур в соответствии с планом машинного эксперимента, имитируя тем самым поведение системы. В результате машинного эксперимента формируются массивы данных о состоянии различных параметров системы в разные моменты времени с привязкой к системным событиям, имитируемым в ходе эксперимента. При этом программа-супервизор управляет процессом имитации случайных возмущающих воздействий, от которых зависит функционирование системы в целом, а также ее элементов и подсистем.

Частный случай имитационных моделей – *модели ситуационные*. Эти модели используются при решении задач с неопределенностью, обусловливаемой совокупностью ситуаций. В отличие от других моделей, основанных на заданном графе функционирования системы, для ситуационной модели такой граф неизвестен. Однако существует набор прецедентов, обладающих малым прогностическим

потенциалом. В целом структура ситуационной модели определяется субъективными особенностями восприятия и свойственным аналитику способом разложения ситуации на составляющие. Это связано с тем, что эксперт-аналитик, осуществляющий процедуру синтеза ситуационной модели, формулирует собственные критерии, соответствующие пребыванию системы в том или ином состоянии [8].

Исследование альтернативных стратегий, как правило, проводится на моделях (увы, не всегда это возможно, да и не всякий руководитель в состоянии оценить преимущества моделирования перед непосредственным действием). Как правило, для решения задач многокритериального оценивания необходимо использовать несколько разнородных моделей, отражающих различные аспекты поведения системы и ее элементов. Кроме того, приходится сталкиваться с проблемой изоляции процессов: с одной стороны, модель уже должна существовать (иначе невозможен синтез критериев), с другой – модель необходимо синтезировать. Но есть одно обстоятельство: в одном случае речь идет о модели системы и ситуации в целом, а в другом – о характере ее изменения в ходе реализации альтернативной стратегии (по существу, модель должна быть кибернетической, т. е. учитывать свойства системы с точки зрения анализа управленческих стратегий). На этом этапе оценивается эффективность реализации некоторой альтернативы и выбирается оптимальная (или близкая к оптимальной) альтернатива из множества допустимых.

На этапе собственно моделирования *модель используется* не в качестве объекта синтеза и анализа, а *как инструмент исследования*. То есть модели считаются адекватными и предполагается, что дальнейшие итерации по совершенствованию моделей нецелесообразны. Модели используются в качестве систем, замещающих заданные фрагменты реальности. На них проводятся вычислительные и логические операции, выражающие выявленные на предшествующих этапах отношения и зависимости, определяются значения критериев выбора, обеспечивающие возможность сопоставления альтернативных стратегий. Речь идет о вариации исходных параметров и логике, отображающей стратегию управления. В результате формируется блок исходных данных, включающих в себя значения и оценки критериев выбора, рисков и других данных, используемых на заключительном этапе. В рамках последнего формулируются выводы по результатам моде-

лирования и указания по реализации его результатов. На этом этапе знания, полученные в ходе проведения всего цикла процедур системно-кибернетического исследования, должны приобрести точное и наглядное выражение. Лицу, принимающему решение (осуществляющему выбор альтернативы), должны быть предоставлены аргументированные выводы и рекомендации в той форме и в тех терминах, которые он способен воспринять.

Контрольные вопросы

1. Приведите понятие модели и ее свойства. Назовите виды подобия между моделью и фрагментами реального мира.
2. Перечислите наиболее значимые факторы, оказывающие влияние на выбор адекватной степени детализации модели.
3. Назовите принцип моделируемости сложных систем и постулаты моделируемости.
4. Перечислите методы формализации предметной области исследований.
5. Назовите этапы построения моделей. Поясните собственно моделирование и вывод результатов исследований.

Глава 2

ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МОДЕЛИРОВАНИЯ СИСТЕМ

2.1. Принципы системного подхода к моделированию систем

Системный подход предполагает последовательный переход от общего к частному, когда в основе рассмотрения лежит цель, причем исследуемый объект выделяется из окружающей среды. Специалисты по проектированию и эксплуатации информационных систем имеют дело с системами управления различных уровней, обладающими общими свойствами – стремлением достичь некоторой цели. Эта особенность учитывается в определении системы. Система S – целенаправленное множество взаимно связанных элементов любой природы. Внешняя среда – множество существующих вне системы элементов природы, оказывающих влияние на систему или находящихся под ее воздействием. В зависимости от цели исследований могут рассматриваться разные соотношения между объектом и средой.

Системный подход – это элемент учения об общих закономерностях природы и одно из выражений диалектического учения. При системном подходе к моделированию систем необходимо четко определить цель моделирования. Цель возникает из требуемых задач моделирования, что позволяет подойти к выбору критерия и оценить, какие элементы войдут в создаваемую модель. Необходимо иметь критерий отбора отдельных элементов в создаваемую модель.

Рассмотрим подходы к исследованию систем. Для системного подхода важно определение структуры системы – совокупности связей между элементами системы, отражающих их взаимодействие. Имеется ряд подходов к исследованию структуры системы и ее свойств – это структурный и функциональный подходы.

При *структурном подходе* выявляются состав выделенных элементов системы и связи между ними, что позволяет судить о структуре системы. Наиболее общее описание структуры – топологическое, которое хорошо формализуется на базе теории графов.

При *функциональном описании* рассматриваются отдельные функции, т. е. алгоритмы поведения системы. Функция отображает

свойство, а свойство отображает взаимодействие системы с внешней средой. Свойства могут быть в виде некоторых характеристик либо элементов и подсистем, либо системы в целом. Проявление функций системы во времени, т. е. функционирование системы, означает переход системы из одного состояния в другое, т. е. движение в пространстве состояний. При эксплуатации системы важно качество ее функционирования, определяемое показателем эффективности и являющееся значением критерия оценки эффективности. Система может оцениваться либо совокупностью частных критериев, либо общим интегральным критерием.

Создаваемая модель M с точки зрения системного подхода также является системой, т. е. $S' = S'(M)$, и может рассматриваться по отношению к внешней среде. Правильное понимание как взаимосвязей внутри модели M , так и взаимосвязей ее с внешней средой E определяется тем, на каком уровне находится наблюдатель. Процесс синтеза модели на основе классического (индуктивного) подхода может использоваться при создании простых моделей (рис. 2.1) [1].

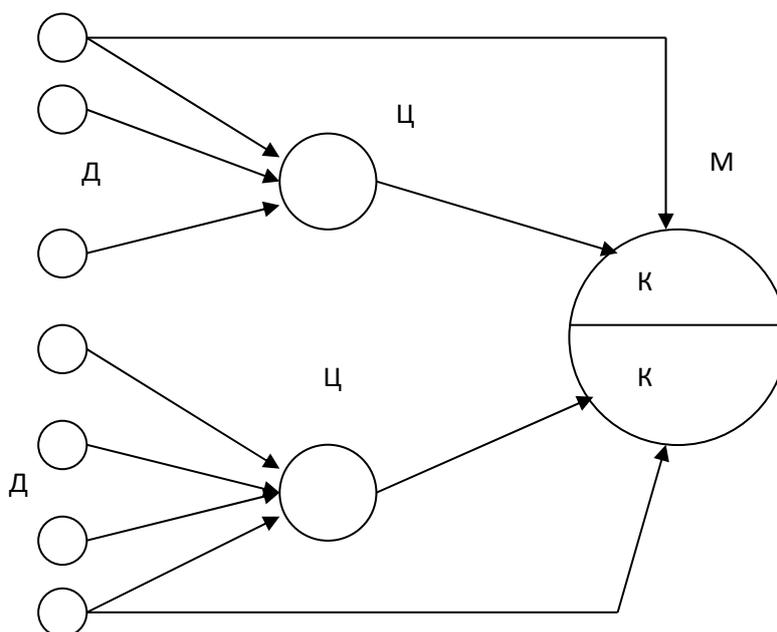


Рис. 2.1. Процесс синтеза модели на основании классического подхода

Реальный объект, подлежащий моделированию, разбивается на подсистемы D , ставятся цели C , отображающие отдельные стороны процесса моделирования. На базе этих целей формируется некоторый

компонент K будущей модели. Совокупности компонентов объединяются в модель M . При классическом подходе наблюдается движение от частного к общему: модель создается путем суммирования отдельных ее компонентов – и не учитывается возникновение нового системного эффекта.

Для моделей сложных объектов такая разобщенность решаемых задач недопустима. Разработчик должен рассматривать анализируемую систему как некоторую подсистему какой-то метасистемы, т. е. системы более высокого уровня. Он вынужден перейти на позиции нового системного подхода, который позволяет построить не только исследуемую систему, решающую совокупность задач, но и систему, являющуюся составной частью метасистемы. Например, при проектировании подсистемы «Бухгалтерия» нельзя забывать, что она составная часть информационной системы предприятия. Процесс синтеза модели на основе системного подхода приведен на рис. 2.2 [1].

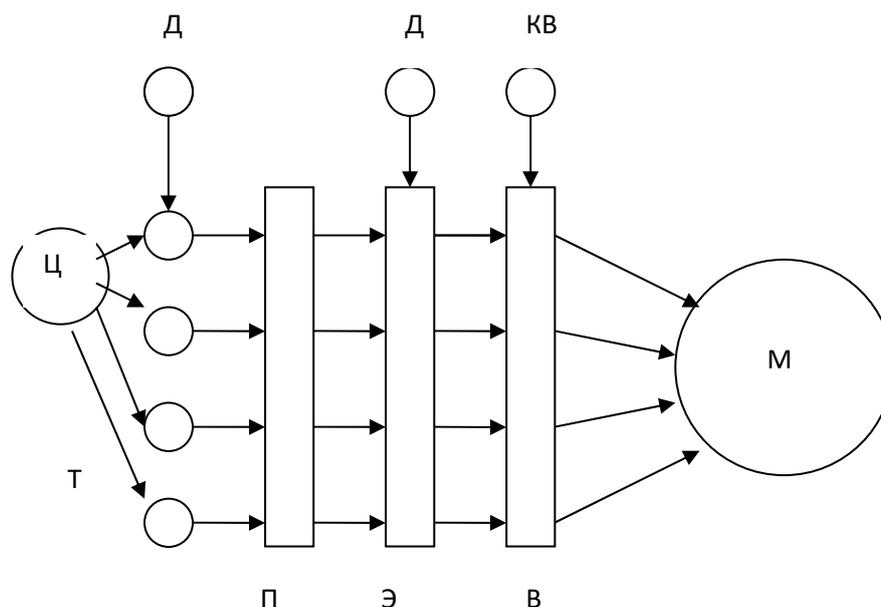


Рис. 2.2. Процесс синтеза модели на основе системного подхода

Системный подход предполагает, что каждая система – это интегрированное целое даже тогда, когда она состоит из отдельных разобщенных подсистем. Построение модели начинается с формулировки цели функционирования. На основе исходных данных, которые из-

вестны из анализа внешней системы, тех ограничений, которые накладываются на систему сверху либо исходя из возможностей ее реализации, и на основе цели функционирования формулируются исходные требования T к модели системы S . На базе этих требований формируются ориентировочно некоторые подсистемы $П$, элементы $Э$ и осуществляется наиболее сложный этап синтеза – выбор B составляющих системы, для чего используется специальный критерий выбора KB .

При моделировании необходимо обеспечить максимальную эффективность модели системы. Эффективность определяется как разность между показателем ценности результатов, полученных в итоге эксплуатации модели, и затратами на разработку и создание модели.

Рассмотрим стадии разработки модели. Предлагается следующая последовательность: макропроектирование и микропроектирование [1].

На *стадии макропроектирования* на основе данных о реальной системе S и внешней среде E строится модель внешней среды, выявляются ресурсы и ограничения для построения модели системы, выбираются модель системы и критерии, позволяющие оценить адекватность модели M реальной системе. Построив модель системы и модель внешней среды, на основе критерия эффективности функционирования системы в процессе моделирования выбирают оптимальную структуру управления, что позволяет реализовать возможности модели по воспроизведению отдельных сторон функционирования реальной системы S .

Стадия микропроектирования зависит от типа выбранной модели. В случае имитационной модели необходимо создать информационное, математическое, техническое и программное обеспечение системы моделирования. На этой стадии устанавливают основные характеристики созданной модели, оценивают время работы с моделью и затраты ресурсов для получения заданного качества соответствия модели процессу функционирования системы S .

При построении модели необходимо руководствоваться рядом принципов системного подхода:

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

- 2) согласование информационных, ресурсных, надежности и других характеристик;
- 3) правильное соотношение отдельных уровней иерархии в системе моделирования;
- 4) целостность отдельных обособленных стадий построения модели.

Использование системного подхода позволяет построить модель реального объекта, на базе этой модели выбрать необходимое количество управляющей информации в реальной системе, оценить показатели ее функционирования и тем самым найти наиболее эффективный вариант построения и выгодный режим функционирования реальной системы.

2.2. Общая характеристика проблемы моделирования систем

Важное значение при создании реальных систем S имеют математические методы анализа и синтеза. Определение качества функционирования большой системы, выбор оптимальной структуры и алгоритмов поведения, построение системы в соответствии с поставленной перед ней целью – основные проблемы при проектировании современных систем, поэтому моделирование можно рассматривать как один из методов, используемых при проектировании и исследовании больших систем. Для объяснения реальных процессов выдвигают гипотезы. Для их подтверждения ставят эксперимент либо приводят теоретические рассуждения, которые логически обосновывают их правильность.

Под экспериментом можно понимать некоторую процедуру организации и наблюдения каких-то явлений в условиях, близких к естественным, либо процедуру имитации их. Различают активный и пассивный эксперименты. В пассивном эксперименте исследователь наблюдает протекающий процесс, в активном – вмешивается и организует протекание процесса.

В основе моделирования лежат информационные процессы, поскольку создание модели базируется на информации о реальном объекте. В процессе реализации модели получают информацию о данном объекте, одновременно в процессе эксперимента с моделью вводят управляющую информацию, важное место занимает обработка

полученных результатов, т. е. информация лежит в основе процесса моделирования.

Характеристики моделей систем. В качестве объекта моделирования выступают сложные организационно-технические системы, которые можно отнести к классу больших систем. По своему содержанию созданная модель также становится системой и ее можно также отнести к классу больших систем.

Для больших систем характерно следующее.

1. *Цель функционирования.* Определяет степень целенаправленности поведения модели. Модели бывают одноцелевые и многоцелевые.

2. *Сложность.* Учитывая, что модель – совокупность отдельных элементов и связей между ними, сложность можно оценить по общему числу элементов в системе и связей между ними. Выделяют ряд уровней иерархии, подсистемы, ряд входов и выходов.

3. *Целостность.* Модель M – одна целостная система $S(M)$, включающая в себя большое количество составных частей, находящихся в сложной взаимосвязи друг с другом.

4. *Неопределенность.* Проявляется в системе в состоянии системы, возможности достижения поставленной цели, методах решения задачи, достоверности исходной информации и т. д. Характеристикой неопределенности служит такая мера информации, как энтропия, позволяющая оценить количество управляющей информации, необходимое для достижения заданного состояния системы.

5. *Поведенческая страта.* Позволяет оценить эффективность достижения системой поставленной цели. В зависимости от случайных воздействий различают детерминированные и стохастические системы, по поведению – непрерывные и дискретные. Поведенческая страта рассмотрения модели дает возможность оценить эффективность построения модели, точность и достоверность полученных результатов моделирования. Поведение модели не обязательно совпадает с поведением реального объекта. Часто моделирование может быть реализовано на базе иного материального носителя.

6. *Адаптивность.* Благодаря ей удается приспособиться к различным возмущающим факторам в широком диапазоне воздействий внешней среды. Поскольку модель – сложная система, весьма важны вопросы живучести, надежности и т. д.

7. *Организационная структура* системы моделирования зависит от сложности модели и степени совершенства средств моделирова-

ния. Имеется возможность использования имитационных моделей для проведения машинных экспериментов. Следует обращать внимание на время моделирования и точность полученных результатов.

8. *Управляемость модели.* Позволяет экспериментаторам рассматривать протекание процессов в различных условиях, имитирующих реальные. Системы моделирования обладают высокой степенью автоматизации процесса моделирования. Наряду с программными средствами управления машинным моделированием используется возможность мультимедийного общения исследователя с процессом моделирования.

9. *Возможность развития модели.* Современный уровень науки и техники позволяет создавать мощные системы моделирования для исследования многих сторон функционирования реального объекта. При создании системы моделирования важна возможность ее развития, т. е. расширения спектра изучаемых функций, числа подсистем, применения новых методов и средств.

Цели моделирования систем. Любую модель строят в зависимости от цели, которую ставит перед ней исследователь. Подобие процесса, протекающего в модели, реальному процессу, – не цель, а условие правильного функционирования модели. В качестве цели должно быть поставлено изучение какой-либо стороны функционирования объекта. Например, для автоматизированной системы управления (АСУ) предприятия – изучение процесса оперативного управления производством, оперативно-календарного планирования, перспективного планирования. Здесь успешно могут использоваться методы моделирования.

Построение модели важно, если имеется информация или выдвинуты гипотезы относительно структуры, алгоритмов и параметров исследуемого объекта. Широко используются различные способы оценки параметров: метод наименьших квадратов, метод максимального правдоподобия, байесовские, марковские оценки.

Если модель построена, то следующей проблемой может стать работа с ней: минимизация времени получения конечных результатов и обеспечение их достоверности. Модель выступает как некоторый «заместитель» оригинала, обеспечивающий фиксацию и изучение лишь некоторых свойств реального объекта.

Характеризуя проблему моделирования в целом, необходимо учитывать, что на этапах от постановки задачи моделирования до ин-

терпретации полученных результатов существует большая группа сложных научно-технических проблем, к основным из которых следует отнести:

- идентификацию реальных объектов;
- выбор вида моделей;
- построение моделей и их машинную реализацию;
- взаимодействие исследователя с моделью в ходе машинного эксперимента;
- проверку правильности полученных в ходе моделирования результатов;
- выявление основных закономерностей, исследованных в процессе моделирования.

В зависимости от объекта моделирования и вида используемой модели эти проблемы могут иметь разную значимость. Важна роль исследователя в процессе моделирования. Постановка задачи, построение содержательной модели реального объекта во многом представляют творческий процесс и базируются на эвристике. Не существует формальных путей выбора оптимального вида модели. Часто отсутствуют формальные методы, позволяющие точно описать реальный процесс. Выбор той или иной аналогии, того или иного математического аппарата моделирования полностью основывается на опыте исследователя, и ошибки исследователя могут привести к ошибочным результатам моделирования.

Средства вычислительной техники, используемые для реализации имитационной модели системы, важны с точки зрения эффективности реализации сложной модели, но они не позволяют подтвердить правильность той или иной модели. Только на основе обработанных данных и опыта исследователя можно с достоверностью оценить адекватность модели реальному процессу.

2.3. Возможности и эффективность моделирования систем на вычислительных машинах

Эффективность экспериментальных исследований сложных систем оказывается крайне низкой, поскольку проведение натуральных экспериментов с реальной системой требует больших материальных затрат и значительного времени либо практически невозможно (на этапе проектирования, когда реальная система отсутствует). Теорети-

ческие исследования эффективны тогда, когда их результаты с требуемой точностью и достоверностью могут быть представлены в виде аналитических соотношений или моделирующих алгоритмов, пригодных для получения соответствующих характеристик процесса функционирования исследуемых систем.

В зависимости от того, удастся ли построить достаточно точную математическую модель реального процесса или вследствие сложности объекта не удастся проникнуть в глубь функциональных связей реального объекта и описать их каким-то аналитическим соотношением, можно рассматривать два основных пути использования ЭВМ:

- 1) как средство расчета по полученным аналитическим моделям;
- 2) как средство имитационного моделирования.

В первом случае перед ЭВМ стоит задача расчета характеристик системы по каким-либо математическим соотношениям при подстановке численных значений. В этом случае вычислительные машины обладают возможностями, зависящими от порядка решаемого уравнения и требований к скорости решения. Основная задача исследований заключается в том, чтобы попытаться описать поведение реального объекта одной из известных математических моделей. Опыт использования вычислительной техники в задачах моделирования показывает, что с усложнением объекта большую эффективность по скорости решения и стоимости выполнения операций дает использование гибридной (аналого-цифровой) техники.

Конкретным техническим средством воплощения имитационной модели могут быть цифровые вычислительные машины (ЦВМ), аналоговые вычислительные машины (АВМ), гибридные вычислительные машины (ГВМ). Обычно модель строится по иерархическому принципу, когда последовательно анализируются отдельные стороны функционирования объекта, и при перемещении центра внимания исследователя рассмотренные ранее подсистемы переходят во внешнюю среду. Иерархическая структура модели может раскрывать последовательность перехода от структурного (топологического) уровня к функциональному (алгоритмическому) и от функционального к параметрическому.

Результаты моделирования в существенной степени зависят от адекватности концептуальной (описательной) модели, от степени по-

добия описания реальному объекту, числа реализаций модели и других факторов.

Имитационная система реализуется на ЭВМ и позволяет исследовать имитационную модель, задаваемую в виде совокупности отдельных блочных моделей и связей между ними с их взаимодействием в пространстве и времени при реализации какого-либо процесса. Можно выделить три основные группы блоков:

1) блоки, характеризующие моделируемый процесс функционирования системы S ;

2) блоки, отображающие внешнюю среду E и ее воздействие на реализуемый процесс;

3) блоки, играющие служебную, вспомогательную роль, обеспечивающие взаимодействие первых двух блоков, а также выполнение дополнительных функций по получению и обработке результатов моделирования.

Кроме того, имитационная система характеризуется набором переменных, с помощью которых удастся управлять изучаемым процессом, и набором начальных условий, при которых можно изменять условия проведения машинного эксперимента.

Машинный эксперимент можно ставить многократно, заранее планировать, определять условия его проведения. При этом необходимо выбрать методику оценки адекватности получаемых результатов и автоматизировать процесс получения и обработки результатов в ходе машинного эксперимента.

Эксперимент с имитационной моделью требует тщательной его подготовки. Имитационная система характеризуется наличием математического, программного, информационного, технического, эргономического и других видов обеспечения.

В качестве критериев целесообразности применения метода имитационного моделирования на ЭВМ можно указать следующее:

1) отсутствие или неприемлемость аналитических, численных и качественных методов решения поставленной задачи;

2) достаточное количество исходной информации о моделируемой системе S для обеспечения возможности построения адекватной имитационной модели;

3) необходимость проведения на базе других возможных методов очень большого количества вычислений, труднореализуемых даже с помощью ЭВМ;

4) возможность поиска оптимального варианта системы при ее моделировании на ЭВМ.

К числу основных достоинств имитационного моделирования сложных систем относят следующие:

– машинный эксперимент с имитационной моделью дает возможность исследовать процесс функционирования системы S в любых условиях;

– существенно сокращается продолжительность испытаний по сравнению с натурным экспериментом;

– имитационная модель позволяет использовать результаты натуральных испытаний реальной системы или ее частей для проведения дальнейших испытаний;

– имитационная модель обладает гибкостью структуры, алгоритмов и параметров моделируемой системы, что важно при поиске оптимального варианта системы;

– имитационное моделирование – практически единственный метод исследования процесса функционирования систем на этапе их проектирования.

Основной недостаток машинной реализации метода имитационного моделирования – то, что решение, полученное при анализе имитационной модели M , всегда носит частный характер, так как оно соответствует фиксированным элементам структуры, алгоритмам поведения и значениям параметров системы S , начальных условий и воздействий внешней среды E .

Для полного анализа характеристик процесса функционирования систем приходится многократно воспроизводить имитационный эксперимент, варьируя исходные данные задачи. При этом увеличиваются затраты машинного времени на проведение эксперимента с имитационной моделью процесса функционирования исследуемой системы S .

Эффективность имитационного моделирования может оцениваться рядом критериев:

– точностью и достоверностью результатов моделирования;

– временем построения и работы с моделью M ;

– затратами машинных ресурсов (времени и памяти);

– стоимостью разработки и эксплуатации модели.

Наилучшая оценка эффективности – сравнение полученных результатов с реальным исследованием (натурным экспериментом).

Это не всегда удается сделать. Статистический подход позволяет с определенной степенью точности при повторяемости машинных экспериментов получить усредненные характеристики поведения системы.

Контрольные вопросы

1. Объясните сущность системного подхода к моделированию сложных систем на ЭВМ.
2. Поясните процесс функционирования системы.
3. Назовите основные характерные черты машинной модели.
4. Обоснуйте цель моделирования системы на ЭВМ.
5. Раскройте особенности имитационного моделирования системы.
6. Поясните, чем обуславливается эффективность моделирования системы на ЭВМ.

Глава 3

МАТЕМАТИЧЕСКИЕ СХЕМЫ МОДЕЛИРОВАНИЯ СИСТЕМ. ФОРМАЛИЗАЦИЯ И АЛГОРИТМИЗАЦИЯ ПРОЦЕССОВ ФУНКЦИОНИРОВАНИЯ СИСТЕМ

Наибольшие затруднения и наиболее серьезные ошибки при моделировании возникают при переходе от содержательного описания объекта исследования к формальному. Этого можно избежать, используя при моделировании язык математических схем, который позволяет обеспечить адекватность при переходе от содержательного описания системы к ее математической схеме. Лишь после этого можно решать вопрос о конкретном методе получения результатов с использованием ЭВМ. Применительно к определенному объекту моделирования используют прошедшие апробацию для данного класса систем математические схемы, показавшие эффективность в прикладных исследованиях и называемые типовыми математическими схемами.

3.1. Основные подходы к построению математических моделей систем

Исходной информацией при построении математических моделей процессов функционирования систем служат данные о назначении и условиях работы исследуемой системы. Эта информация определяет основную цель моделирования и позволяет сформулировать требования к разрабатываемой модели. Уровень абстрагирования зависит от перечня тех вопросов, на которые исследователь хочет получить ответ с помощью модели, что в определенной степени обуславливает выбор математической схемы.

Математическую схему можно рассматривать как звено при переходе от содержательного к формальному описанию процесса функционирования системы с учетом воздействия внешней среды. Каждая система характеризуется набором свойств, под которыми понимаются величины, отражающие поведение моделируемой системы и учитывающие условия ее функционирования во взаимодействии с внешней средой. При построении модели системы необходимо решать вопрос о ее полноте, которая регулируется путем выбора границы «система –

среда». Так же решается задача упрощения модели, которая позволяет выделить основные свойства системы, отбросив второстепенные.

Модель объекта моделирования можно представить в виде множества величин, описывающих функционирование реальной системы и образующих следующие подмножества [1]:

- совокупность входных воздействий на систему $x_i \subset X$, $i = 1, 2, \dots, n_x$;
- совокупность воздействий внешней среды $v_l \subset V$, $l = 1, 2, \dots, n_v$;
- совокупность внутренних параметров системы $h_k \subset H$, $k = 1, 2, \dots, n_h$;
- совокупность выходных характеристик системы $y_j \subset Y$, $j = 1, 2, \dots, n_y$.

В перечисленных подмножествах можно выделить управляемые и неуправляемые переменные. В общем случае x_i, v_l, h_k, y_j – элементы непересекающихся подмножеств, содержат как детерминированные, так и стохастические составляющие.

При моделировании системы входные воздействия, воздействия внешней среды и внутренние параметры – независимые переменные, а выходные характеристики системы – зависимые переменные.

Процесс функционирования системы во времени описывается оператором

$$\vec{y}(t) = F_s(x, v, h, t). \quad (3.1)$$

Соотношение (3.1) – математическое описание поведения объекта моделирования во времени, т. е. оно отражает его динамические свойства (динамическая модель).

Закон функционирования системы (3.1) может быть задан в виде функции, функционала, логических условий, в алгоритмической и табличной формах или в виде словесного правила соответствия. Для описания и исследования системы важен алгоритм ее функционирования, под которым понимается метод получения выходных характеристик с учетом входных воздействий, воздействий внешней среды и собственных параметров системы.

Для статических моделей процесс функционирования системы описывается оператором

$$\vec{y} = (x, v, h). \quad (3.2)$$

Если рассматривать функционирование системы как последовательную смену состояний $z_1(t), z_2(t), \dots, z_k(t)$, то последние могут быть интерпретированы как координаты точки в k -мерном фазовом пространстве. Каждой реализации процесса соответствует некоторая фазовая траектория. Совокупность всех возможных значений состояний Z называется пространством состояний объекта моделирования. Состояние системы в момент времени $t_0 < t^* \leq T$ полностью определяется начальными условиями $z_1(t_0), z_2(t_0), \dots, z_k(t_0)$, входными воздействиями $\vec{x}(t)$, внутренними параметрами $\vec{h}(t)$ и воздействиями внешней среды $\vec{v}(t)$, которые имели место за промежуток времени $t^* - t_0$, с помощью двух векторных уравнений

$$\begin{aligned} \vec{z}(t) &= \varphi(\vec{z}^0, \vec{x}, \vec{v}, \vec{h}, t), \\ \vec{y}(t) &= F[\vec{z}(t)]. \end{aligned} \quad (3.3)$$

Первое условие по начальному состоянию \vec{z}^0 и входным переменным \vec{x}, \vec{v} , внутренним параметрам \vec{h} определяет вектор-функцию $\vec{z}(t)$, а второе по полученному значению состояния $\vec{z}(t)$ – переменную на выходе системы.

Если математическое описание объекта моделирования не содержит элементов случайности или они не учитываются, т. е. отсутствуют стохастические воздействия внешней среды и стохастические внутренние параметры, то модель называется детерминированной, т. е. характеристики определяются детерминированными входными воздействиями $\vec{y}(t) = f(\vec{x}, t)$.

Приведенное математическое описание (3.3) представляет математические схемы общего вида, которые позволяют описать широкий класс систем. Однако на первых этапах исследования системы рационально использовать *типовые математические схемы*: дифференциальные уравнения, конечные и вероятностные автоматы, конечно-разностные схемы, системы массового обслуживания, сети Петри и т. д.

Перечисленные математические схемы не претендуют на описание всех процессов, происходящих в больших информационно-управляющих системах (ИУС). Для таких систем более перспективно применение агрегативных моделей. При агрегативном описании сложный объект расчленяется на конечное число частей, при этом сохраняются связи, обеспечивающие взаимодействие частей.

При построении математических моделей процессов функционирования систем можно выделить следующие основные подходы [1]:

- непрерывно-детерминированные (дифференциальные уравнения);
- дискретно-детерминированные (конечные автоматы);
- дискретно-стохастические (вероятностные автоматы);
- непрерывно-стохастические (системы массового обслуживания);
- обобщенный подход (агрегативные схемы).

3.2. Формализация и алгоритмизация процессов функционирования систем. Методика разработки и машинная реализация моделей систем

Несмотря на многообразие классов моделирующих систем, можно выделить основные закономерности перехода от концептуальной модели объекта моделирования к проведению машинного эксперимента с моделью. Машинная модель позволяет «растягивать» и «сжимать» реальное время протекания процессов в анализируемой системе. Сущность машинного моделирования состоит в проведении на ЭВМ эксперимента с моделью, которая представляет собой программный комплекс, описывающий поведение элементов системы в процессе ее функционирования. Машинное моделирование применяют в тех случаях, когда трудно сформулировать критерий оценки качества функционирования системы и цель системы не поддается полной формализации.

Требования пользователей к модели следующие.

1. Полнота модели должна предоставлять пользователю возможность получения необходимых характеристик системы с требуемой точностью и достоверностью.
2. Гибкость модели должна позволять воспроизводить различные ситуации при варьировании структуры, алгоритмов и параметров системы.
3. Минимальная длительность разработки модели сложной системы.
4. Блочная структура модели, позволяющая замену, добавление и исключение некоторых частей модели.

5. Информационное обеспечение должно позволять модели работать с базой данных систем определенного класса.

6. Программные и технические средства должны обеспечивать эффективную машинную реализацию модели и удобное общение с ней пользователя.

7. Возможность проведения машинных экспериментов с моделью системы при использовании аналитико-имитационного подхода и ограниченных вычислительных ресурсах.

Моделирование с помощью ЭВМ можно использовать:

1) для исследования системы до того, как она спроектирована, с целью определения чувствительности характеристик к изменению структуры, алгоритмов и параметров объекта моделирования и внешней среды;

2) на этапе проектирования системы для анализа и синтеза различных вариантов системы и выбора среди альтернатив наиболее эффективного варианта;

3) после внедрения системы, при ее эксплуатации для получения информации, дополняющей результаты натуральных испытаний реальной системы, и прогноза развития системы во времени.

Основные этапы моделирования систем следующие:

1) построение концептуальной модели системы и ее формализация;

2) алгоритмизация модели системы и ее машинная реализация;

3) получение и интерпретация результатов моделирования системы.

На этапе построения концептуальной модели и ее формализации проводится исследование моделируемого объекта с точки зрения выделения основных составляющих процесса его функционирования, определяются необходимые аппроксимации и схема модели системы. На втором этапе схема модели преобразуется в машинную модель путем последовательной алгоритмизации и программирования модели. На третьем этапе, согласно плану эксперимента, проводят рабочие расчеты на ЭВМ, получают и интерпретируют результаты моделирования системы с учетом воздействий внешней среды.

При построении модели и ее машинной реализации, при получении новой информации возможен пересмотр ранее принятых решений, т. е. процесс моделирования итерационный. Рассмотрим подробнее содержание этапов.

Построение концептуальных моделей систем и их формализация. На первом этапе формулируют концепцию модели и строят ее формальную схему, т. е. происходит переход от содержательного описания к его математической модели. Наиболее ответственно и неформализованно проведение границы между системой и внешней средой, упрощение описания системы, построение сначала концептуальной, а затем формальной модели системы. Модель должна быть адекватной, т. е. с определенной степенью приближения отражать функционирование системы во внешней среде.

Рационально строить модель функционирования системы по блочному принципу. Можно выделить три автономных блока: имитатор воздействия внешней среды на систему, модель процесса функционирования исследуемой системы, вспомогательный блок для машинной реализации двух упомянутых блоков, а также для фиксации и обработки результатов моделирования. Блочная модель предназначена для анализа характеристик процесса функционирования при машинной реализации системы.

После блочной модели строят математическую модель процессов, протекающих в различных блоках. Математическая модель представляет собой совокупность соотношений, определяющих характеристики функционирования системы в зависимости от структуры системы, алгоритмов поведения, параметров системы, воздействий внешней среды, начальных условий и времени. Формализации процесса функционирования любой системы должно предшествовать изучение составляющих его явлений. В результате появится содержательное описание процессов – исходный материал для последующих этапов формализации.

На рис. 3.1 показана концептуальная (когнитивная) модель закалочной установки для производства автомобильных стекол (оригинальная разработка) [4]. В качестве примера приведем математические модели, описывающие функционирование закалочной установки в производстве автомобильных стекол [формулы (3.4) – (3.7)].

В результате обработки экспериментальных данных получена система независимых уравнений регрессии, описывающая напряжения сжатия в контрольных точках вырабатываемых изделий.

$$y_1 = -302 + 0,064T_{14} + 0,0398T_{15} - 0,039T_{16} - 0,49T_{18} + 0,766T_{19} + 0,086T_{27} - 0,1T_{35} + 0,281T_{36} - 0,15T_{39} + 0,11T_{44}, \quad (3.4)$$

оценки уравнения: $R^2 = 39,6 \%$, $F = 5,9$, $S_e = 1,4$ МПа, $\bar{e} = 1,1$ МПа, $dw = 1,7$;

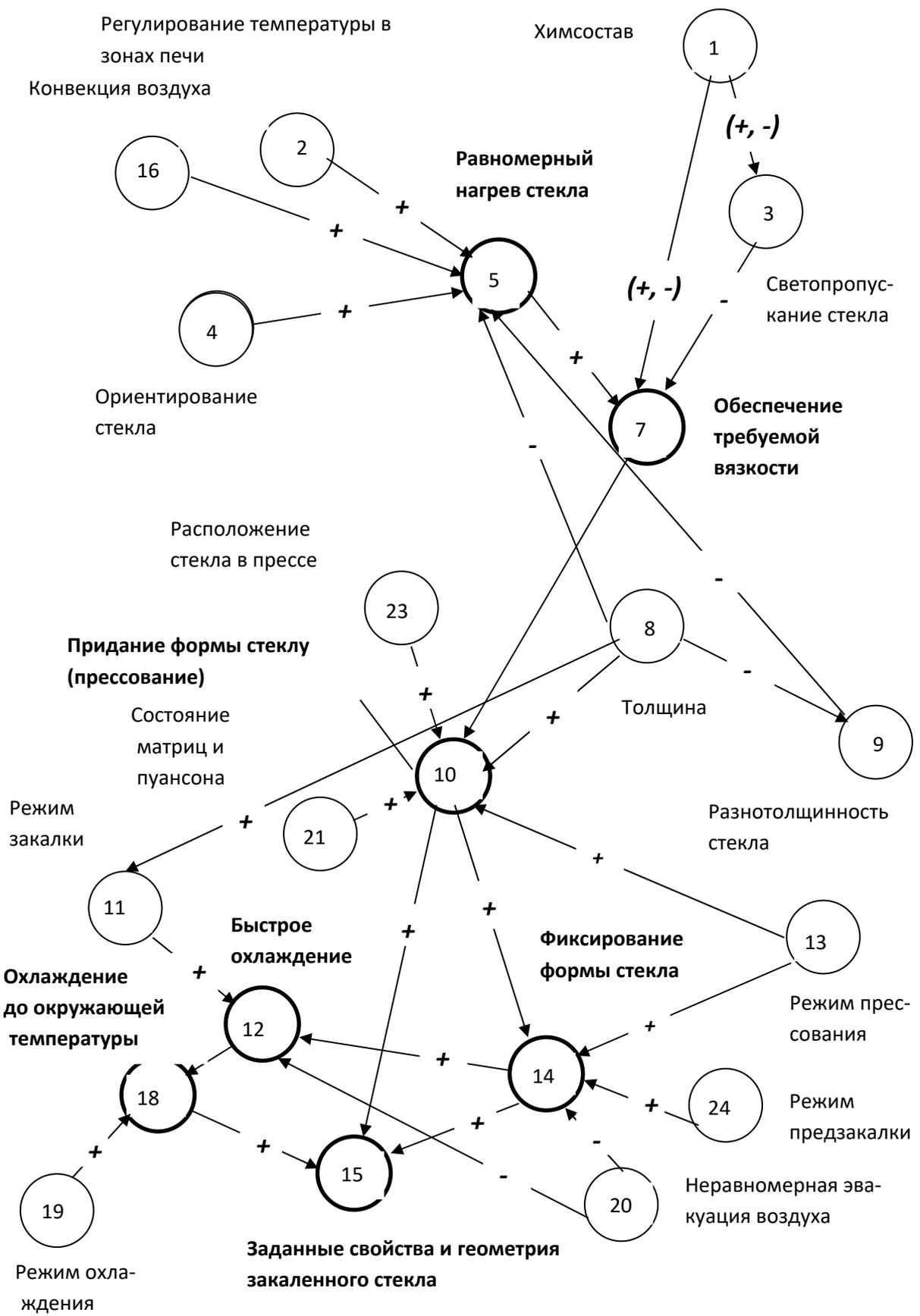


Рис. 3.1. Когнитивная модель процесса закалки стекла

$$y_5 = 1276 + 0,2T_{14} + 0,08T_{15} - 0,036T_{16} - 3,12T_{17} + 1,4T_{19} + 0,94T_{20} + \\ + 0,315T_{25} + 0,173T_{27} + 0,072T_{31} + 0,103T_{34} - 1,32T_{37} - 0,017T_{38} - \\ - 0,161T_{39} - 0,061T_{46}, \quad (3.5)$$

оценки уравнения: $R^2 = 53,6 \%$, $F = 7,1$, $S_e = 2,5$ МПа,
 $\bar{e} = 1,8$ МПа, $dw = 1,8$;

$$y_7 = 186 + 0,507T_{17} + 1,13T_{18} - 1,24T_{19} - 0,9T_{21} + 0,021T_{23} - 0,547T_{26} + \\ + 0,232T_{27} + 0,83T_{28} + 1,08T_{30} - 0,053T_{32} - 0,249T_{33} - 0,812T_{36} - \\ - 0,429T_{40} - 0,298T_{41} - 0,03T_{44} + 0,273T_{49} - 0,443T_{50}, \quad (3.6)$$

оценки уравнения: $R^2 = 43,6 \%$, $F = 3,8$, $S_e = 3,1$ МПа,
 $\bar{e} = 2,3$ МПа, $dw = 1,83$;

$$y_{11} = 964,2 - 0,076T_{12} + 0,493T_{13} - 1,09T_{14} + 0,922T_{15} + 0,139T_{21} - \\ - 0,324T_{22} + 0,228T_{23} + 0,7992T_{26} + 0,051T_{27} - 0,175T_{29} + \\ + 0,243T_{35} + 0,36T_{36} - 0,264T_{37} - 0,03T_{38} - 2,5T_{43}, \quad (3.7)$$

оценки уравнения: $R^2 = 47,4 \%$, $F = 5,2$, $S_e = 2,8$ МПа, $\bar{e} = 2,2$ МПа,
 $dw = 2,4$.

После описания концептуальной модели необходимо проверить достоверность некоторых положений концепции модели, перед тем как перейти к следующему этапу моделирования системы. Проверить достоверность концептуальной модели сложно, так как она построена эвристически и описывается в абстрактных терминах и понятиях. Проверка достоверности должна включать в себя:

- 1) проверку замысла модели;
- 2) оценку достоверности исходной информации;
- 3) рассмотрение постановки задачи моделирования;
- 4) анализ принятых аппроксимаций;
- 5) исследование гипотез и предположений.

После проверки концептуальной модели переходят к этапу машинной реализации модели.

Алгоритмизация моделей систем и их машинная реализация.

Для моделирования процесса функционирования системы на ЭВМ необходимо преобразовать математическую модель процесса в соответствующий моделирующий алгоритм и машинную программу [1].

Рассмотрим принципы построения моделирующих алгоритмов. Наиболее универсальный принцип построения моделирующего алго-

ритма – принцип Δt , позволяющий определить последовательные состояния процесса функционирования системы через заданные интервалы времени Δt . Для этого преобразуют соотношения состояния модели $Z(t) = Z(z_1(t), z_2(t), \dots, z_k(t))$ к такому виду, чтобы сделать удобными вычисления: $z_1(t + \Delta t), z_2(t + \Delta t), \dots, z_k(t + \Delta t)$. Организуют счетчик системного времени, который в начальный момент показывает время t_0 . Для этого момента $Z_i(t_0) = Z_i^0$. Прибавляют интервал времени Δt , тогда счетчик показывает $t_1 = (t_0 + \Delta t)$, вычисляют значение $Z_i(t_0 + \Delta t)$. Затем переходят к моменту времени $t_2 = (t_1 + \Delta t)$ и т. д. Если шаг мал, то можно получить приближенные значения Z . С точки зрения затрат машинного времени принцип Δt может оказаться не экономным.

При рассмотрении процессов моделирования некоторых систем может оказаться, что для них характерны два типа состояний:

1) особые – присущи процессу функционирования системы только в некоторые моменты времени (поступление входных воздействий, воздействий внешней среды и т. п.). Функция состояния $Z_i(t)$ в эти моменты времени изменяется скачком;

2) не особые – процесс находится в них все остальное время, изменение координат $Z_i(t)$ происходит плавно и непрерывно или не происходит совсем.

Для описания систем могут быть построены моделирующие алгоритмы по принципу «особых состояний», определяющихся скачкообразным изменением состояния Z (принцип δz).

При исследовании функционирования больших систем используют комбинированный принцип построения моделирующего алгоритма, сочетающий преимущества рассмотренных принципов Δt и δz .

Удобная форма представления логической структуры моделей процессов функционирования систем и машинных программ – схемы моделирующих алгоритмов и схемы программ.

Проверку достоверности модели системы выполняют на этапе реализации модели. Проверка должна дать ответ на вопрос, насколько логическая схема модели системы и используемые математические соотношения отражают замысел модели, сформированный на первом этапе. При этом проверяются:

- 1) возможность решения поставленной задачи;
- 2) точность отражения замысла в логической схеме;

3) полнота логической схемы модели; правильность используемых математических соотношений.

Перед написанием моделирующей программы составляется план выполнения работ по программированию. При использовании универсальной ЭВМ план должен включать в себя:

- 1) выбор языка программирования;
- 2) указание типа ЭВМ и выбор необходимых устройств;
- 3) оценку объема оперативной и внешней памяти;
- 4) сведения о затратах машинного времени на отладку программы и моделирование.

Получение и интерпретация результатов моделирования системы. Для проведения рабочих расчетов по составленной и отлаженной программе используется ЭВМ. Результаты расчетов позволяют сформулировать и проанализировать выводы о характеристиках процесса функционирования моделируемой системы. При моделировании вырабатывается информация о состоянии процесса функционирования исследуемой системы $z(t) \in Z$. Эта информация – исходный материал для определения оценок характеристик, получаемых в результате машинного эксперимента, т. е. критериев оценки. Последними могут служить показатели, получаемые на основе процессов, протекающих в системе, или на основе специально сформированных функций этих процессов.

Прежде чем приступать к машинному моделированию, необходимо разработать план вычислительного эксперимента. Планирование эксперимента важно для получения максимального объема информации об объекте моделирования при минимальных затратах машинных ресурсов. При этом различают стратегическое и тактическое планирование машинных экспериментов. При *стратегическом планировании* решается задача построения оптимального плана эксперимента для достижения цели, поставленной перед моделированием (оптимизация структуры, алгоритмов и параметров системы, исследуемой методом моделирования). *Тактическое планирование* решает задачу оптимальной организации каждого эксперимента из множества необходимых, например выбор оптимальных планов остановки при статистическом моделировании.

После составления программы модели и плана проведения машинного эксперимента с моделью приступают к рабочим расчетам на ЭВМ, которые включают в себя:

- 1) подготовку исходных данных для ввода в ЭВМ;
- 2) проверку исходных данных, подготовленных для ввода;
- 3) проведение расчетов на ЭВМ;
- 4) получение выходных данных, т. е. результатов моделирования.

Машинное моделирование рационально выполнять в два этапа: контрольные, а затем рабочие расчеты. Контрольные расчеты необходимы для проверки машинной модели и определения чувствительности результатов к изменению исходных данных.

Результаты моделирования могут быть представлены в виде графиков (рис. 3.2), таблиц, диаграмм, схем и т. п. [19].



Рис. 3.2. Изменение плотности стекла (1 – фактическое и 2 – при управлении) в процессе варки

Получив и проанализировав результаты моделирования, их нужно интерпретировать по отношению к моделируемому объекту. Основное содержание этого этапа – переход от информации, полученной в результате машинного эксперимента с моделью, к информации об объекте моделирования, на основании которой и делают выводы относительно характеристик процесса функционирования исследуемой системы.

При подведении итогов моделирования необходимо отметить главные особенности результатов, полученных в соответствии с пла-

ном эксперимента над моделью, провести проверку гипотез и предположений и сделать выводы на основании этих результатов. Все это позволяет сформулировать рекомендации по практическому использованию результатов моделирования.

Пример. Сравнительные результаты моделирования алгоритмов управления с ручным ведением процесса стекловарения в течение года работы линии ЛПС-1 приведены в таблице [19].

Результаты моделирования алгоритмов управления
тепловым режимом работы печи

Режим управления	Плотность стекла, г/см ³	Свильность по образцам	Пузыри, шт/м ²	Оптические искажения по методу «Зебра»	Удельный расход тепла, ккал/кг
Ручной	2,4859/ 0,00064	1,33/0,24	8,2/2,4	48,4°/7,9°	2027,7
Автоматический	2,4851/ 0,00024	1,48/0,11	7,7/0,9	48,6/6,4	2022,3

Примечание. Первое число – среднее, второе число – среднеквадратичное отклонение переменной.

При управлении тепловым режимом печи по суточному изменению плотности достигается стабилизация плотности и однородности вырабатываемого стекла по сравнению с ручным ведением процесса стекловарения. Улучшается качество стекла за счет уменьшения среднеквадратичного отклонения свильности в 2,2 раза, пузырей – в 2,5 – 2,8 раза, оптических искажений – в 1,2 раза. Одновременно уменьшается содержание пузырей на вырабатываемой ленте стекла на 22 %. Разница в удельном расходе тепла на стекловарение не превышает 0,4 %, что находится в пределах погрешностей моделирования, и ею можно пренебречь.

Имитационное моделирование подтвердило эффективность применения алгоритма управления процессом стекловарения по суточному изменению плотности вырабатываемого стекла. Предложенный алгоритм управления рекомендуется для внедрения на стекольном производстве.

По завершении этапа составляется техническая документация. Она включает в себя:

- 1) план проведения эксперимента;
- 2) исходные данные для моделирования;
- 3) результат моделирования системы;
- 4) анализ и оценку результатов моделирования;
- 5) выводы по полученным результатам моделирования; указание путей совершенствования машинной модели и возможных областей ее применения.

Полный комплект документов по моделированию конкретной системы должен содержать техническую документацию по каждому из трех этапов.

Контрольные вопросы

1. Что такое математическая схема?
2. Что понимается под алгоритмом функционирования?
3. Что такое статические и динамические модели объекта?
4. Какие типовые схемы используют при моделировании сложных систем и процессов?
5. В чем суть методики машинного моделирования информационных процессов и систем?
6. Что такое концептуальная модель системы?
7. Назовите основные этапы моделирования систем.
8. Для чего используется ЭВМ при моделировании систем?

Глава 4

МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОЦЕССОВ ОБРАБОТКИ ИНФОРМАЦИИ

4.1. Моделирование информационных процессов

Информационная система какого-либо объекта представляет собой формализованную модель, которая в той или иной степени отражает наиболее существенные его свойства. Проектирование информационных систем становится основой для разработки автоматизированных систем моделирования и управления ресурсами (человеческими, финансовыми, материальными, энергетическими, информационными и др.) различных объектов. В качестве объектов моделирования примем процессы, протекающие в информационных системах управления (ИСУ) [4]. Остановимся на их сущности и содержании, для чего рассмотрим:

- коммуникационную (информационную) схему передачи информации в системе управления;
- классификацию и характеристику информационных процессов;
- структуру информационного процесса.

Несмотря на все многообразие информационных процессов (ИП), протекающих в системах управления, с точки зрения технологии обработки информации они имеют много общего. Это позволяет представить обобщенную схему обработки информации в системах управления (рис. 4.1) [6].

Источниками информации в системах управления могут быть должностные лица (ДЛ) органов управления, автоматические датчики (АД) и вычислительные комплексы (ВК). В общем случае информация до момента выдачи ее пользователю проходит следующие основные технологические этапы преобразования:

- сбор данных (вручную или автоматически);
- формирование сообщения (запроса);
- передача данных по каналам связи с использованием средств автоматизации или традиционными способами;

- выдача данных лицу, принимающему решения, или для их ввода в вычислительный комплекс;
- решение информационных или расчетных задач в ВК;
- выдача результатов решения задач лицом, принимающим решение (ЛПР);
- доведение принятого решения или результатов решения задач до адресатов.

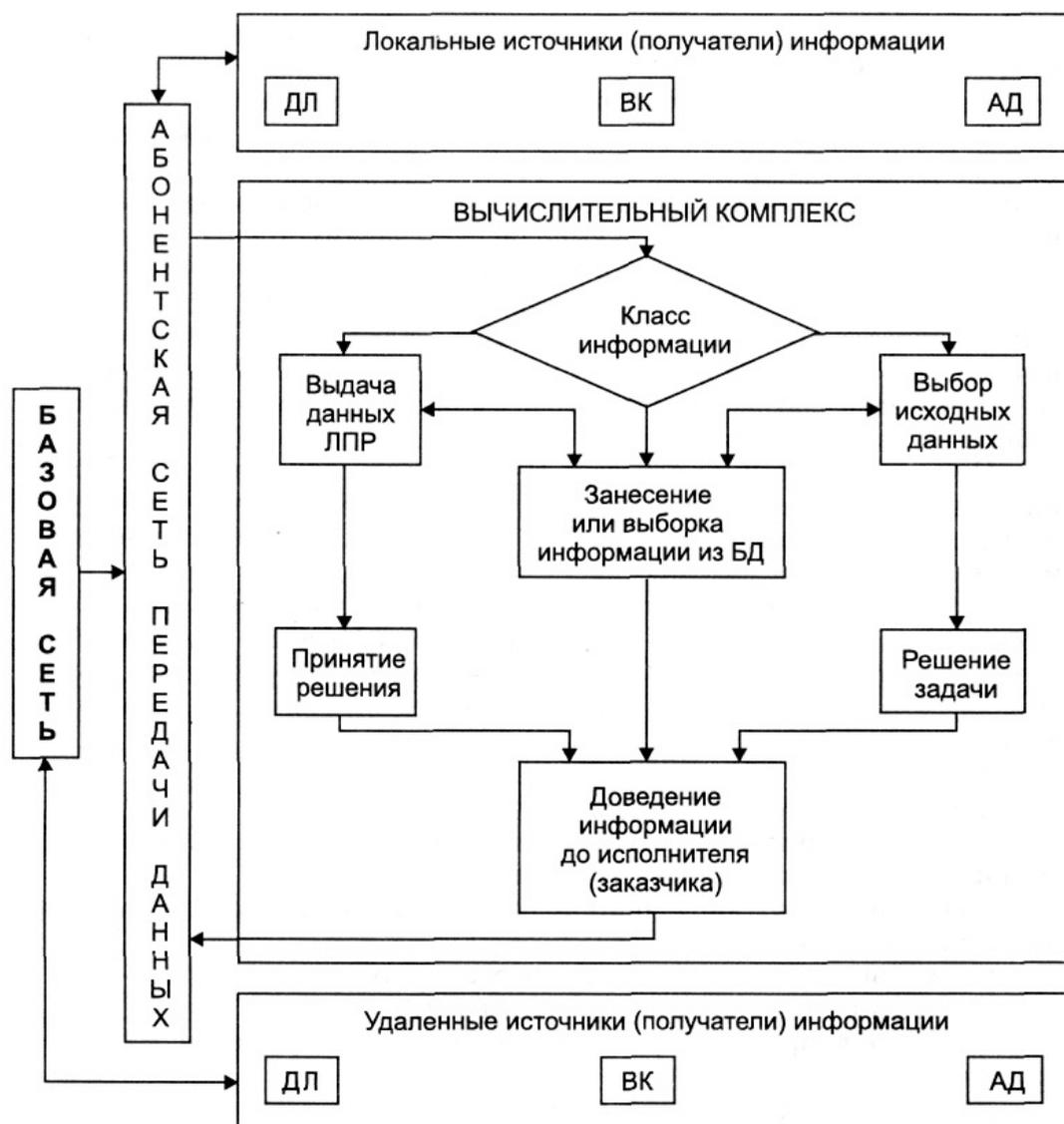


Рис. 4.1. Обобщенная схема обработки информации в системах управления

Для различных категорий пользователей количество этапов, их длительность, а также типы генерируемых ими сообщений (запросов), как правило, различны. Они зависят от конкретного назначения системы, технической реализации ее отдельных элементов, их прио-

ритетов, характера сведений, передаваемых в информационных сообщениях, и т. д.

Структура информационного процесса. Определив понятие информационного процесса как совокупность операций по преобразованию информации, необходимо отметить следующее.

Любая информационная единица (I), обладающая потребительской стоимостью (качеством), характеризуется содержанием (S), формой (F), пространственным расположением (L) и временем (T), т. е. $I = \{S, F, L, T\}$. Каждая из этих характеристик в процессе преобразования (в информационном процессе) может изменяться. При этом различают следующие виды преобразования информации:

- содержательное (семантическая обработка, в результате которой получается новая информация);
- преобразование формы (например, кодирование, декодирование и т. п.);
- преобразование в пространстве и времени (например, передача данных и их хранение).

Элементарное действие в информационном процессе – операция преобразования информации (рис. 4.2) [6].

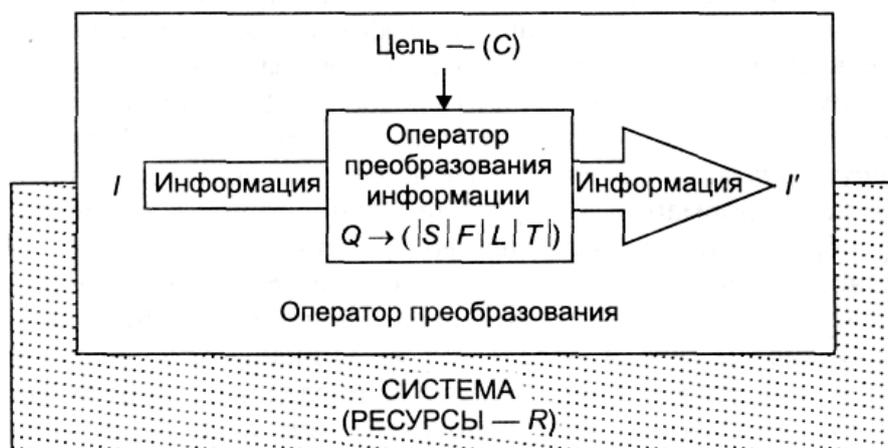


Рис. 4.2. Структура операции преобразования информации:
 $I' = Q(I, C, R)$, где I, I' – кванты информации

Основные атрибуты операции – информация (объект преобразования); оператор (субъект преобразования) и цель (требования к преобразованию).

Операции по преобразованию информации могут быть разных типов, которые определяются типом оператора $Q \rightarrow (|S|F|L|T)$, слож-

ностью (U), зависящей от типа оператора и цели преобразования, $U = \langle Q, C \rangle$, времени (T) ее реализации, $T = \langle U, I \rangle$ и ресурсоемкости (R).

Информационный процесс структурно можно представить множеством подпроцессов. В зависимости от того, какие виды преобразования информации доминируют (т. е. являются основными в системе достижения цели), можно выделить следующие подпроцессы.

1. Процесс формирования (подготовки) информации для преобразования. Доминирующий вид преобразования – преобразование формы (F).

2. Процесс передачи информации от источника к потребителю. Очевидно, что основной вид преобразования здесь – преобразование информации в пространстве (L).

3. Процесс семантической (смысловой) обработки информации – центральный подпроцесс ИП в системе управления. В результате появляется новая информация, на основе которой в конечном счете формируются управляющие воздействия. Именно по этому процессу можно полностью идентифицировать всю систему с управлением, и именно от качества обработки информации зависит эффективность функционирования системы. Цель данного процесса – выработка адекватных управляющих воздействий оптимальным (рациональным) способом. Доминирующий вид преобразования в этом случае – преобразование содержания информации (S).

4. Процесс хранения информации. Цель данного процесса – обеспечить существование информации во времени. Основной вид преобразования здесь – преобразование информации во времени (T).

Очевидно, что все реальные процессы и системы, их реализующие, а следовательно, и структуры ИП уникальны. Специфика функционирования реальных систем (определяющая их уникальность) проявляется в содержании самой информации, составе и последовательности применения конкретных операторов по ее преобразованию и целей, в существовании того или иного процесса, составе и взаимосвязи подпроцессов, а также в конкретных глобальных целях функционирования системы с управлением. На рис. 4.3 пред-

ставлен фрагмент структуры ИП, который обычно присущ большим системам управления.



Рис. 4.3. Фрагмент структуры ИП

Управленческие (информационные) процессы можно реализовать на ЭВМ, если их удастся формализовать. Под формализацией понимается точное описание изучаемого объекта. Формализация процесса управления включает в себя выделение управленческих функций и задач, разработку алгоритмов формализации и проведение алгоритмизации. Процесс считают полностью формализованным, если алгоритмы к задачам представлены математически и переведены на машинные программы и при решении задач уже не нужно учитывать их физическое содержание. Потребность в знании физического содержания появляется лишь при использовании полученных результатов.

Моделирование ИП, протекающих в системах управления, проводится для определения объема и форм представления информации,

методов и средств ее передачи, обработки, хранения, ввода и вывода для известной структуры и алгоритма функционирования системы управления (см. рис. 4.3) [6].

Процедура информационного моделирования включает в себя:

- определение потребности в информации на каждой стадии управления;

- планирование потребностей в информации;

- определение количественных и качественных характеристик коммуникационных процессов;

- определение потребности в информации при оценке эффективности управленческих решений (воздействий).

К показателям (характеристикам) информации относят:

- объем и скорость передачи информации;

- достоверность передаваемых сообщений;

- направление информационных потоков;

- характеристики методов обработки информации и совершаемые при этом ошибки;

- качественный состав информации;

- количество обрабатываемых или передаваемых документов;

- суммарное количество обрабатываемых и передаваемых документов и др.

На основе результатов моделирования вырабатывают предварительные рекомендации по разработке информационного обеспечения системы управления, в том числе по способам передачи, обработки и представления информации, составу информации, необходимой для нормального функционирования системы, структуре информационного обмена и др.

4.2. Характеристики процесса обработки информации

Для анализа процесса обработки информации можно использовать модель (рис. 4.4), представленную граф-схемой алгоритма (ГСА).

В информационных системах, в особенности работающих в реальном масштабе времени, важно, чтобы решения принимались в заданное время и выдаваемая информация была достоверна. Это означает, что все операторы, лежащие на определенном пути от начальной вершины к конечной, должны выполняться с заданной скоростью и

надежностью. При проектировании систем эти характеристики основные и подлежат выбору [4]. На начальных стадиях проектирования допускается, что в системе нет отказов в обработке информации. Необходимо знать количество N_{ij} операций i -го типа в j -м алгоритме, которое задается в виде характеристического вектора

$$N_j = (N_{j1}, N_{j2}, \dots, N_{jm}),$$

где m – число разнообразных типов операций в программах решения функциональных задач.

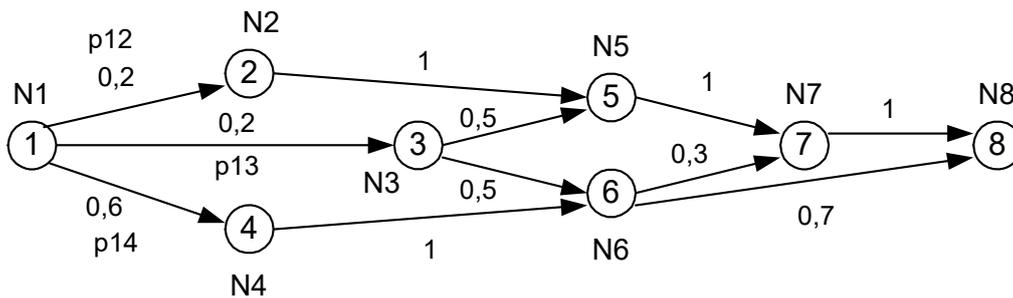


Рис. 4.4. Граф-схема алгоритма вычислительного процесса

Величина N_{ij} зависит от адресности ЭВМ, системы команд, структуры системы обмена. Все многообразие операций разбивают на короткие N_k , длинные N_d операции, операции обращения N_o . На начальных этапах исследования N_j неизвестно, поэтому можно воспользоваться данными аналогичных программ, реализованных на аналогичных ЭВМ. Знание N_k и N_d необходимо для определения требований к арифметическому логическому устройству (АЛУ), N_o используется для выбора структуры устройства обмена и его связей с АЛУ. Для аналитических расчетов требуется ГСА с заданными вероятностями перехода p_{ij} от i -й к j -й вершине и характеристический вектор N для каждого оператора.

Решение выполняется при следующих положениях:

- 1) вероятность перехода к следующему оператору не зависит от того, каким путем пришли к предыдущему оператору;
- 2) вероятности p_{ij} остаются постоянными для системы и не изменяются во времени.

При указанных допущениях процесс выполнения алгоритма представляет собой марковский процесс с k состояниями $s_1, s_2, s_3, \dots, s_k$, которые соответствуют операторам x_1, \dots, x_k (для нашего примера

$$\begin{aligned}
n_4 &= 0,6; \\
n_5 &= 0,2 \cdot 1 + 0,2 \cdot 0,5 = 0,3; \\
n_6 &= 0,7; \\
n_7 &= 0,51; \\
n_8 &= 1.
\end{aligned}$$

Для каждого алгоритма можно просчитать частоту появления каждого типа операции β_l :

$$\beta_l = n_l / (n_1 + n_2 + \dots + n_m), \quad l = 1, m, \quad (4.6)$$

где m – общее число типов операций.

Введем величину частотного вектора для алгоритма

$$\beta = (\beta_1, \beta_2, \dots, \beta_m). \quad (4.7)$$

Для характеристики обмена информацией между алгоритмами используется показатель внешней связности $G_{вш}$:

$$G_{вш} = U_{вх} + U_{вых}, \quad (4.8)$$

где $U_{вх}$ – объем входной информации (в словах); $U_{вых}$ – объем выходной информации, передаваемый другим алгоритмам.

Показатель внутренней связности $G_{вн}$ рассчитывается по формуле

$$G_{вн} = U_k + U_b, \quad (4.9)$$

где U_k – объем констант; U_b – результаты промежуточных вычислений.

Число операций обмена можно вычислить по формуле

$$N_o = a G_{вш} + b G_{вн}, \quad (4.10)$$

где a – число операций, затрачиваемых на обмен одним словом; b – то же на обмен одним словом между внешним запоминающим устройством (ЗУ) и устройствами вычислительной системы при передаче констант и результатов промежуточных вычислений.

Из опыта известно, что $a = 1 \dots 10$; b значительно меньше из-за передачи данных массивами.

4.3. Точность процесса обработки информации

В процессе обработки информации в ИС возникают погрешности вычислений, которые зависят от параметров алгоритма вычислений и технических характеристик устройств. Рассмотрим погрешности вычислений на примере информационно-управляющей системы (рис. 4.5), работающей в реальном масштабе времени. Аналоговая информация от датчика, установленного на объекте управления, через

аналого-цифровой преобразователь (АЦП) поступает в цифровую вычислительную машину. После обработки показаний датчика в ЭВМ управляющая информация через цифро-аналоговый преобразователь (ЦАП) в аналоговой форме поступает на исполнительное устройство. В результате обработки информации в различных устройствах ИС возникают погрешности вычислений.



Рис. 4.5. Структура информационной управляющей системы

Погрешности делятся на погрешности аппроксимации, вычислительные погрешности, трансформационные погрешности, параметрические погрешности, динамические погрешности [4]. Погрешности имеют случайный характер и оцениваются законами распределения. В основном для описания погрешностей пользуются симметричными законами распределений. Чаще всего погрешности аппроксимируются нормальным и равномерным законами распределения. Параметры этих законов распределения – математическое ожидание и дисперсия.

Мерами погрешности вычислений могут быть:

– абсолютная погрешность единичного вычисления; оценивается в физических величинах значений y :

$$\Delta y = A_y - a_y,$$

где A_y – истинное значение; a_y – расчетное значение параметра;

– относительная погрешность вычисления (безразмерная величина)

$$\delta_y = \frac{\Delta y}{a_y} 100 \%$$

или относительная приведенная погрешность вычислений

$$\delta_y = \frac{\Delta y}{a_y^H}$$

где a_y^H – номинальное значение вычисляемого параметра или диапазон изменения параметра.

В зависимости от значения относительной приведенной погрешности вычислений различают вычислители (системы):

- прецизионные: $\delta_y < 0,5 \dots 1,5 \%$;
- точные: δ_y может принимать значения 0,5; 1,0; 2,5 %;
- невысокой точности: δ_y до 10 %.

В технических системах управления используются точные вычислители.

Для оценки точности процесса обработки информации принимаем следующие допущения.

1. В погрешностях вычислений отсутствуют систематические составляющие, т. е. математическое ожидание случайной погрешности равно нулю.

2. Компоненты погрешности взаимно некоррелированы.

При сделанных допущениях общая дисперсия погрешности вычислений равна сумме компонентов (рис. 4.6):

$$\sigma^2 = \sigma_a^2 + \sigma_B^2 + \sigma_T^2 + \sigma_{\Pi}^2 + \sigma_D^2, \quad (4.11)$$

где σ_a^2 – дисперсия погрешности аппроксимации; σ_B^2 – дисперсия вычислительной погрешности; σ_T^2 – дисперсия трансформационной погрешности; σ_{Π}^2 – дисперсия параметрической погрешности; σ_D^2 – дисперсия динамической погрешности.

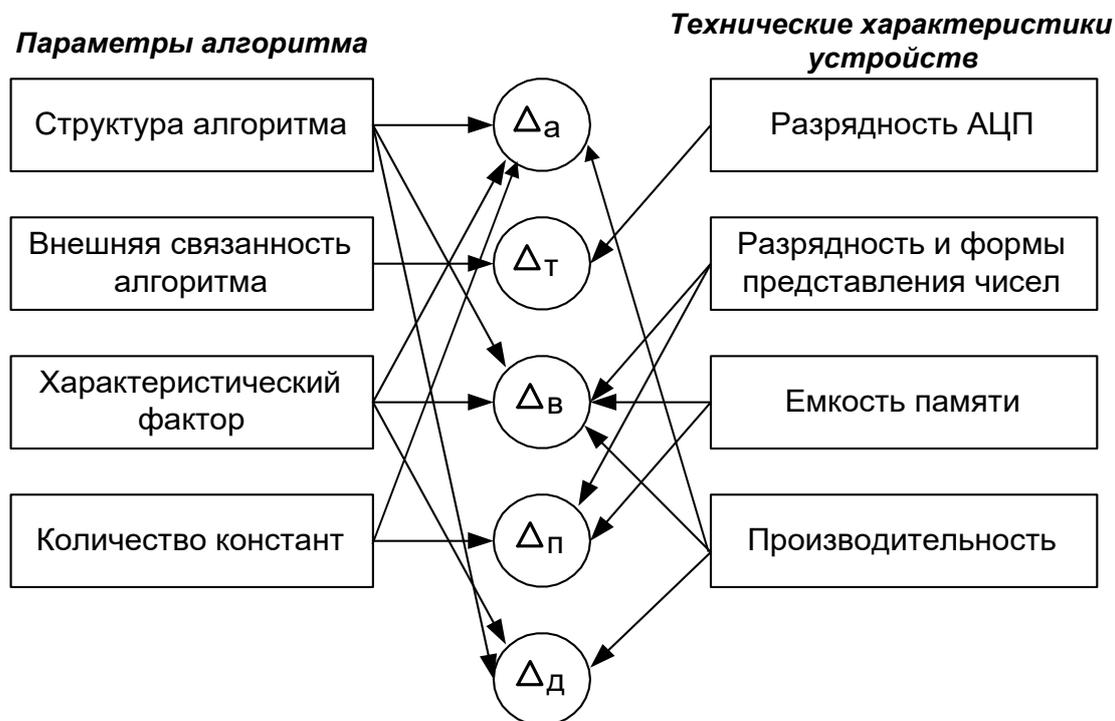


Рис. 4.6. Структура формирования погрешностей вычислений

Параметры ИС определяют инструментальные погрешности системы

$$\sigma_{\text{и}}^2 = \sigma_{\text{в}}^2 + \sigma_{\text{т}}^2 + \sigma_{\text{п}}^2. \quad (4.12)$$

Составляющие погрешности $\sigma_{\text{а}}^2$ и $\sigma_{\text{д}}^2$ мало зависят от технических характеристик ИС.

Рассмотрим отдельные компоненты погрешностей вычисления.

Погрешность аппроксимации $\sigma_{\text{а}}^2$ зависит от структуры алгоритма, т. е. от выбранного численного метода решения задач. При известном законе распределения плотности вероятностей максимальная вероятная ошибка вычисления может быть оценена по формуле

$$\Delta_{\text{МАК}} = \gamma\sigma,$$

где γ – параметр, зависящий от закона распределения и задаваемой вероятности оценки погрешностей. Для вероятности $P = 0,99$ и нормального закона распределения плотности вероятностей ошибки принимается $\gamma = 3$. При этом значение дисперсии погрешности аппроксимации может быть вычислено по формуле

$$\sigma^2 = \left(\frac{\Delta_{\text{МАК}}}{\gamma}\right)^2. \quad (4.13)$$

Трансформационная погрешность $\sigma_{\text{т}}^2$ определяется в основном разрядностью аналого-цифрового преобразователя или же точностью вводимых данных. Погрешность исходных данных пересчитывается к выходной величине, определяемой по формуле

$$y = \varphi(x), \quad (4.14)$$

где y – расчетная выходная величина; x – входная информация; $\varphi(x)$ – алгоритм вычисления.

Дисперсию трансформационной погрешности можно рассчитать, линеаризовав зависимость (4.14):

$$\sigma_{\text{т}}^2 = \left(\frac{\partial\varphi(x)}{\partial x}\right)^2 \sigma_x^2, \quad (4.15)$$

где σ_x^2 – дисперсия погрешности входных данных, поступающих в вычислитель.

При подаче на вход АЦП аналогового сигнала возникает погрешность квантования сигнала по уровню. Последняя описывается равномерным законом распределения плотности вероятностей:

$$\sigma_x^2 = \frac{(\Delta x)^2}{12}, \quad (4.16)$$

где Δx – абсолютная погрешность квантования аналогового сигнала, определяемая ценой младшего разряда АЦП.

Вычислительная погрешность σ_B^2 зависит от количества вычислительных операций в алгоритме решения задачи, от разрядности ЭВМ, а также от формы представления данных и масштабирования данных. Вычисления можно представить совокупностью отдельных вычислительных операций. Дисперсия погрешности одной вычислительной операции при равномерном законе распределения погрешности может быть оценена по формуле

$$\sigma_{01}^2 = \frac{(\Delta y)^2}{12}, \quad (4.17)$$

где Δy – цена младшего разряда расчетной выходной величины.

Если в программе реализуется N_B вычислительных операций, тогда дисперсия погрешности вычислений

$$\sigma_B^2 = N_B \sigma_{01}^2. \quad (4.18)$$

Параметрическая погрешность σ_{Π}^2 связана с погрешностями квантования констант вычислительных алгоритмов и зависит от разрядности и формы представления чисел, имеет равномерный закон распределения. Дисперсия параметрической погрешности рассчитывается по формуле

$$\sigma_{\Pi}^2 = \frac{(\Delta k)^2}{12}, \quad (4.19)$$

где Δk – абсолютная погрешность хранения констант.

Динамическая погрешность σ_D^2 обусловлена конечной продолжительностью времени выполнения вычислений и, следовательно, запаздыванием выдачи управляющих воздействий (решения задачи) в общем случае на k тактов. Абсолютная динамическая погрешность определяется формулой

$$\Delta d(n\Delta t) = y(n\Delta t) - \varphi(x(n-k)\Delta t),$$

где Δt – шаг квантования; $n\Delta t$ – момент определения погрешности.

Зная абсолютное значение динамической погрешности, можно вычислить дисперсию динамической погрешности:

$$\sigma_D^2 = (\Delta d/\gamma)^2. \quad (4.20)$$

Баланс погрешностей [см. формулу (4.11)] позволяет оценить источники погрешностей и принять при необходимости меры по уменьшению величины погрешности вычислений.

Точность процесса обработки информации в ИС, работающих не в режиме реального времени, может оцениваться по вышеописанной методике, за исключением того, что в балансе погрешностей [см. формулу (4.11)] отсутствует динамическая составляющая погрешности.

4.4. Время реализации алгоритма

Важная характеристика процесса обработки информации – время реализации алгоритма в ЭВМ. Этот показатель особенно важен для систем, работающих в реальном времени. Время реализации алгоритма определяется задаваемым периодом выдачи управляющей информации, а также временной диаграммой решения задач. При этом необходимо учитывать влияние неисправностей ИС на время вычислений.

Если за время T необходимо выполнить N вычислительных операций при отсутствии отказов в ИС, скорость вычислений

$$V = \frac{N}{T}. \quad (4.21)$$

В реальной системе могут возникать неисправности, обнаружение которых влечет следующие действия:

- 1) восстановление работоспособности и повторное включение части программы;
- 2) изменение конфигурации системы и повторное выполнение части программы.

Это требует дополнительных затрат времени $T_{\text{п}}$, а следовательно, увеличения скорости вычислений. Оценим увеличение скорости вычислений.

Пусть за время T решается M типов задач. Каждая задача реализует подмножество операторов. Тогда общее количество вычислений составит

$$N_b = \sum_{j=1}^m N(k_j). \quad (4.22)$$

Время, затрачиваемое на реализацию набора задач, определяется соотношением

$$T = \sum_{i=1}^M t_{y_i} + \sum_{i=1}^M t_{Bi}^H, \quad (4.23)$$

где t_{y_i} – среднее время управления при решении i -й задачи; t_{Bi}^H – несовмещенное время решения i -й задачи.

Требуемая скорость вычислений для идеализированного случая, когда в ИС нет отказов,

$$\Pi = \frac{N_b}{T} = \frac{\sum_{j=1}^M N(k_j)}{\sum_{i=1}^M t_{y_i} + \sum_{i=1}^M t_{Bi}^H}. \quad (4.24)$$

Реально в системе могут возникать отказы и сбои, следовательно, необходимо затрачивать время на восстановление работоспособного состояния системы и повторное выполнение части программы. В обоих случаях теряется время T_{Π} , поэтому приходится увеличивать скорость вычислений:

$$V = \frac{N}{T - T_{\Pi}} = \frac{N}{T} \left(\frac{1}{\left(1 - \frac{T_{\Pi}}{T}\right)} \right). \quad (4.25)$$

На начальных этапах проектирования при отсутствии полной информации принимают $T_{\Pi}/T = 0,3 \dots 0,6$, тогда формула (4.25) примет вид

$$V = \frac{N}{T} (1,4 \dots 2,5), \quad (4.26)$$

т. е. у реальной информационной системы производительность должна быть в 1,4 ... 2,5 раза выше по сравнению с безотказным режимом работы.

Контрольные вопросы

1. Назовите и поясните особенности моделирования информационных процессов в системах управления.
2. Приведите характеристики информационных единиц.

3. Раскройте содержание операций по преобразованию информации.

4. Раскройте особенности моделирования алгоритма процесса обработки информации граф-схемами.

5. Объясните понятия динамической длины программы, частотного вектора, внешней и внутренней связности алгоритма.

6. Обоснуйте точность обработки информации. Назовите виды погрешностей вычислений.

7. Перечислите, от чего зависят время реализации алгоритма и требуемая скорость вычислений.

ВВЕДЕНИЕ В ТЕОРИЮ НЕЧЕТКИХ МНОЖЕСТВ

5.1. Общие понятия

В математике давно используется понятие множества – совокупности объектов, выделенных по некоторому признаку. Это понятие базовое в современной математике и потому не определяется строго, формально. Так, если задано некоторое базовое множество X (конечное или бесконечное), то его подмножеством (четким подмножеством) A называется любое множество, содержащее в себе только элементы множества X (хотя, может быть, и не все его элементы).

Любое подмножество A множества X можно описать его функцией принадлежности $\mu_A : X \subset \{0; 1\}$, значение которой для элемента $x \in X$ равно единице в том случае, если этот элемент принадлежит множеству A , и нулю – в противном случае.

Соответствие между подмножествами множества X и всевозможными функциями принадлежности $\mu : X \subset \{0; 1\}$ взаимно однозначно, т. е., определив некоторое подмножество, можно найти его функцию принадлежности и наоборот: задание функции $\mu_A : X \subset \{0; 1\}$ обуславливает и подмножество A множества X .

В четком множестве любой элемент может или принадлежать ему, или не принадлежать, поэтому функция принадлежности принимает лишь два возможных значения: нуль или единица.

В нечетком же множестве (точнее, в нечетком подмноестве базового множества X) любой элемент $x \in X$ может принадлежать множеству с некоторой степенью достоверности, принимающей значения от нуля (элемент достоверно не принадлежит множеству) до единицы (элемент достоверно принадлежит множеству). Соответственно, и функция принадлежности нечеткого множества может принимать любое значение из отрезка $[0, 1]$.

Определим понятие нечеткого множества через его функцию принадлежности [23]. Пусть X – некоторое обыкновенное (четкое)

множество. В дальнейшем будем рассматривать его нечеткие подмножества \tilde{A}^* .

Определение 1. Нечетким множеством \tilde{A} в множестве X называется функция $\mu_{\tilde{A}} : X \subset \{0; 1\}$, которая каждому из элементов множества X ставит в соответствие степень его принадлежности нечеткому множеству \tilde{A} .

Нечеткое множество \tilde{A} называется нормальным, если $\sup \mu_{\tilde{A}}(x) = 1$ для $x \in X$. В противном случае оно называется субнормальным. Носителем $\text{supp } \tilde{A}$ нечеткого множества \tilde{A} называется обычное множество $\text{supp } \tilde{A} := \{x \in X : \mu_{\tilde{A}}(x) > 0\}$.

Пример 1. Пусть множество X – это множество всех действительных чисел. Множество A чисел больше нуля – его четкое подмножество с функцией принадлежности

$$\mu_{\tilde{A}}(x) = \begin{cases} 1, & x > 0, \\ 0, & x \leq 0. \end{cases}$$

Можно определить нечеткое множество \tilde{A} чисел, много больших нуля, задав его функцию принадлежности $\mu_{\tilde{A}}$, например, следующим образом:

$$\mu_{\tilde{A}}(x) = \begin{cases} 1, & x > 100, \\ 0, & x \leq 0, \\ 0,5 - 0,5 \cos(\pi x/100), & 0 < x \leq 100. \end{cases}$$

Действительно, числа меньше нуля достоверно не являются много больше нуля, поэтому функция принадлежности равна нулю.

Числа, большие 100, во многих приложениях (будем считать, что и в примере 1) достоверно считаются много большими нуля, поэтому функция принадлежности для таких чисел равна единице. По поводу же чисел в интервале между 0 и 100 достоверно сказать, что они много больше нуля, нельзя, поэтому функция принадлежности на этом интервале принимает значения между нулем и единицей. В то же время понятно, что чем больше число, тем больше оснований считать его много большим нуля, поэтому на интервале от нуля до ста функция принадлежности монотонно возрастает. Носителем нечеткого множества \tilde{A} является интервал $(0; +\infty)$. Функции принадлежности множеств A и \tilde{A} приведены на рис. 5.1.

* Далее нечеткие объекты (множества, отношения и т. д.) будут обозначаться волной.

Зачем же было введено понятие нечеткого множества? Для того же, для чего вводятся и другие математические объекты, – чтобы описывать окружающий нас мир. В действительности большинство понятий, которые используют люди в повседневной жизни, нечеткие! Когда сапожник ждет около трех минут после нанесения клея перед склеиванием, когда хозяйка в соответствии с рецептом кладет в суп две щепотки соли, когда менеджер в коммерческой фирме выполняет указание руководства существенно повысить объемы продаж, – все они выполняют нечеткие инструкции, сформулированные неформально с помощью разговорного языка.

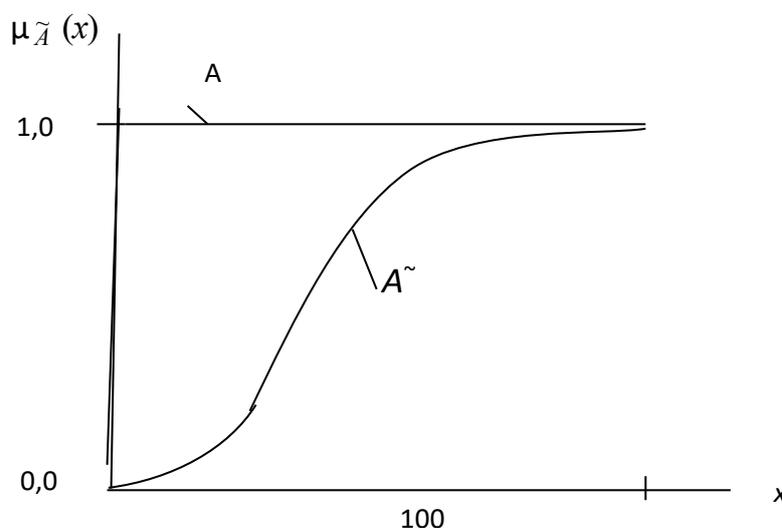


Рис. 5.1. Сравнение функций принадлежности четких и нечетких множеств

Даже формально четкие понятия, используемые в обыденной жизни, воспринимаются людьми как нечеткие. Например, в рецепте может быть четко указано «две чайные ложки соли», но хозяйка понимает, что блюдо не будет испорчено, если положить две с половиной ложки, к тому же чайные ложки бывают разной емкости.

Удобный способ математического описания неформальных понятий, подобных упомянутым выше, – нечеткие множества. Язык нечетких множеств имеет существенные преимущества перед языком теории вероятностей в том случае, когда оценки получают из опроса экспертов. Известно, что люди в большинстве своем неправильно оценивают вероятности (особенно большие и малые), потому требовать от экспертов (специалистов в конкретных предметных областях, а не математиков) оценок в форме распределения вероятности зача-

стую невозможно. Использование же полученных результатов для принятия решений можно квалифицировать как самонадеянное. Описание в форме нечетких множеств гораздо менее требовательно к квалификации экспертов и зачастую гораздо точнее отражает суть дела и имеющуюся у ЛПР информацию.

Конечно, за это удобство приходится платить. Предлагаемые теорией решения, основанные на нечеткой информации, и сами несут на себе печать нечеткости. Они могут рассматриваться лишь как рекомендации для лица, принимающего решение, последний выбирает один из предлагаемых вариантов. Тем не менее даже этот факт можно считать достоинством теории. Он показывает, как увеличение информированности ЛПР сказывается на достоверности и правильности принимаемых решений.

5.2. Операции над нечеткими множествами

Для того чтобы построить содержательную теорию нечетких множеств, необходимо определить операции (такие как объединение, пересечение и т. п.) над нечеткими множествами, аналогичные операциям над обычными, четкими множествами. Сделать это позволяет аналогия между представлением четких и нечетких множеств в форме их функций принадлежности. Большинство операций над обычными множествами могут быть сформулированы через операции над их функциями принадлежности. В то же время функция принадлежности обычного множества – частный случай функции принадлежности нечеткого множества, что позволяет непосредственно обобщать формулы для четких множеств на нечеткий случай. При этом при применении к четким множествам операция дает обычный результат.

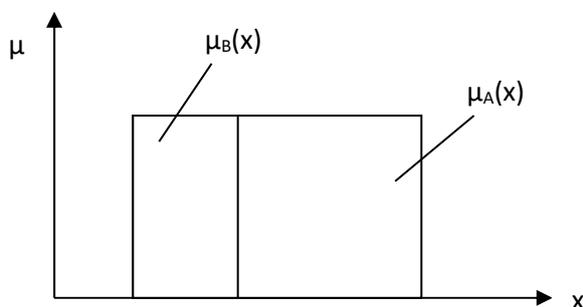


Рис. 5.2. Принадлежность нечетких множеств

Определение 2. Нечеткое множество \tilde{B} в множестве X – подмножество нечеткого множества \tilde{A} в множестве X (\tilde{B} принадлежит \tilde{A} , $\tilde{B} \subset \tilde{A}$), если для всех $x \in X$ $\mu_{\tilde{B}}(x) \leq \mu_{\tilde{A}}(x)$ (рис. 5.2). В теории множеств считается, что пустое множество \emptyset принадлежит любому

множеству. Также, по определению 2, и нечеткое пустое множество с функцией принадлежности $\mu_0(x) \equiv 0$ принадлежит любому нечеткому множеству.

Функцию принадлежности четкого множества $C = A \cap B$ (пересечение множеств A и B) можно записать в виде

$$\mu_C(x) = \min[\mu_A(x); \mu_B(x)].$$

Эту формулу можно использовать и для пересечения нечетких множеств \tilde{A} и \tilde{B} , приняв, по определению 1,

$$\mu_{\tilde{A} \cap \tilde{B}}(x) := \min[\mu_{\tilde{A}}(x); \mu_{\tilde{B}}(x)] \text{ для всех } x \in X.$$

Введем также операции пересечения и объединения произвольного семейства нечетких множеств.

Определение 3. Пусть задано семейство нечетких множеств \tilde{A}_Λ , индексированных параметром $\Lambda \in \Lambda$. Их пересечением будем называть нечеткое множество \tilde{B} с функцией принадлежности

$$\mu_{\tilde{B}}(x) := \inf_{\Lambda \in \Lambda} \mu_{\tilde{A}_\Lambda}(x).$$

Примечание: inference – вывод, заключение, минимальное значение.

Объединением же множеств \tilde{A}_Λ , $\Lambda \in \Lambda$, будем называть нечеткое множество \tilde{C} с функцией принадлежности

$$\mu_{\tilde{C}}(x) := \sup_{\Lambda \in \Lambda} \mu_{\tilde{A}_\Lambda}(x).$$

Примечание: supreme – предельный, высший, крайний.

Заметим, что носитель $\text{supp}(\tilde{A} \cap \tilde{B})$ пересечения нечетких множеств \tilde{A} и \tilde{B} равен пересечению $\text{supp} \tilde{A} \cap \text{supp} \tilde{B}$ их носителей, а носитель объединения – объединению носителей.

Определение 4. Дополнением нечеткого множества \tilde{A} называется нечеткое множество \tilde{A}' с функцией принадлежности $\mu_{\tilde{A}'}(x) := 1 - \mu_{\tilde{A}}(x)^*$.

* Здесь и далее дополнения нечетких множеств обозначаются штрихом.

Дополнение часто используется в теории принятия решений для построения отрицаний нечетких понятий. Например, для нечеткого множества чисел, много больших нуля, его дополнением будет множество чисел, не являющихся много большими нуля.

Для обычных множеств пересечение множества и его дополнения пусты. Как показывает следующий пример, для нечетких множеств это уже не так.

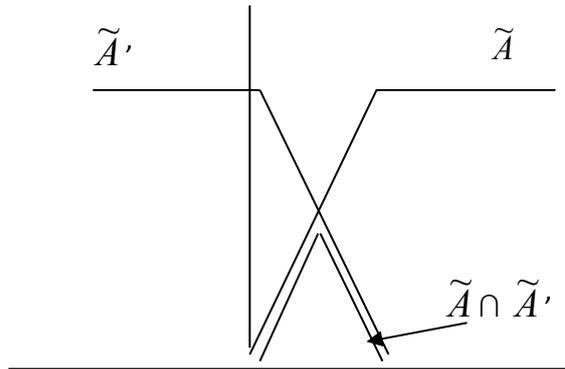


Рис. 5.3. Пересечение нечеткого множества и его дополнения

Пример 2. Найдем пересечение нечеткого множества \tilde{A} чисел, много больших нуля, и его дополнения. На рис. 5.3 изображены функция принадлежности множества \tilde{A} , его дополнения \tilde{A}' и их пересечения $\tilde{A} \cap \tilde{A}'$.

Содержательно нечеткое множество $\tilde{A} \cap \tilde{A}'$ можно интерпретировать

как нечеткое множество чисел, которые одновременно являются много большими нуля и не являются много большими нуля. Непустота этого множества – следствие того, что понятие «быть много большим» описано нечетко и в некотором смысле множество $\tilde{A} \cap \tilde{A}'$ может рассматриваться как «нечеткая граница» между множеством \tilde{A} и его дополнением. Дополнение нечеткого множества – частный случай операции разности двух нечетких множеств.

Определение 5. Разностью $\tilde{A} \setminus \tilde{B}$ нечетких множеств \tilde{A} и \tilde{B} называется нечеткое множество с функцией принадлежности

$$\mu_{\tilde{A} \setminus \tilde{B}}(x) := \max[\mu_{\tilde{A}}(x) - \mu_{\tilde{B}}(x); 0].$$

Таким образом, дополнение нечеткого множества \tilde{A} в множество X – это разность $X \setminus \tilde{A}$.

Для нечетких множеств можно определить и операции, аналогов которых для четких множеств нет. Такова операция умножения нечеткого множества на число.

Определение 6. Произведением нечеткого множества \tilde{A} на число $a \in [0; 1]$ называется нечеткое множество $a\tilde{A}$ с функцией принадлежности

$$\mu_{a\tilde{A}}(x) = a \mu_{\tilde{A}}(x).$$

Определим еще одно понятие, полезное при анализе нечетких множеств.

Определение 7. Множеством уровня $a \in [0; 1]$ нечеткого множества \tilde{A} называется обыкновенное множество $\tilde{A}_a := \{x \in X : \mu_{\tilde{A}}(x) \geq a\}$ – множество точек, степень принадлежности которых нечеткому множеству не меньше, чем a .

С использованием операции умножения на число произвольное нечеткое множество \tilde{A} можно представить в виде объединения его взвешенных множеств уровня a по формуле

$$\tilde{A} = \bigcup_a \tilde{A}_a, \quad a \in [0; 1].$$

Определение 8. Выпуклой комбинацией нечетких множеств $\tilde{A}_1, \dots, \tilde{A}_n$ называется нечеткое множество \tilde{A} с функцией принадлежности

$$\mu_{\tilde{A}}(x) := \sum a_i \mu_{\tilde{A}_i}(x), \quad i = 1, 2, \dots, n,$$

где a_1, \dots, a_n – произвольные неотрицательные числа, сумма которых равна единице.

С помощью понятия выпуклой комбинации можно ввести понятие *выпуклого семейства нечетких множеств* – семейства, содержащего все выпуклые комбинации своих элементов. Например, семейство всех нечетких множеств в X выпукло. Именно возможность рассматривать выпуклые семейства и выпуклые замыкания – основное техническое преимущество нечетких множеств перед обыкновенными множествами.

Определение 9. Декартовым произведением нечетких множеств \tilde{A}_1 из X_1, \dots, \tilde{A}_n из X_n называется нечеткое множество \tilde{A} в $X = X_1 \dots X_n$ с функцией принадлежности

$$\mu_{\tilde{A}}(x) := \min[\mu_{\tilde{A}_1}(x_1); \dots; \mu_{\tilde{A}_n}(x_n)].$$

Итак, основной способ определения операций над нечеткими множествами – обобщение соответствующих операций над обычными множествами.

5.3. Нечеткие отображения и задачи принятия решений

Продолжим обобщать понятия обычной, четкой математики на нечеткий случай. Определим понятия нечеткого отображения, образа и прообраза нечеткого множества при нечетком отображении и с их помощью решим две задачи принятия решений – задачу достижения нечеткой цели и задачу оптимизации при нечетких ограничениях.

Нечеткие отображения. Обычным, четким отображением (многозначным) f множества X во множество Y называется произвольное подмножество декартова произведения $X \times Y$, т. е. $f \subset X \times Y$. Множество X называется областью определения отображения, а множество Y – областью значений. Для фиксированного элемента $x \in X$ области определения отображения его образом при отображении f называется множество $f(x) := \{y \in Y : (x, y) \in f\}$.

Образом множества $A \in X$ при отображении f называется объединение образов всех элементов A , т. е. множество

$$f(A) := \bigcup_{x \in A} f(x) = \{y \in Y : \exists x \in A, (x, y) \in f\}.$$

Пример 3. Рассмотрим четкое многозначное отображение f отрезка $X = [-1, 1]$ в отрезок $Y = [0, 1]$, которое каждой точке $x \in X$ ставит в соответствие множество точек $y \in Y$, больших, чем x^2 . В этом случае $f = \{(x, y) \in X \times Y : y > x^2\}$. Этому отображению соответствует затененная область на рис. 5.4. Образом точки $x = 0,5$ при таком отображении будет интервал $(0,25; 1]$ (см. рис. 5.4, а), а образом точки $y = 0,25$ при обратном отображении – интервал $(-0,5; 0,5)$ (см. рис. 5.4, б).

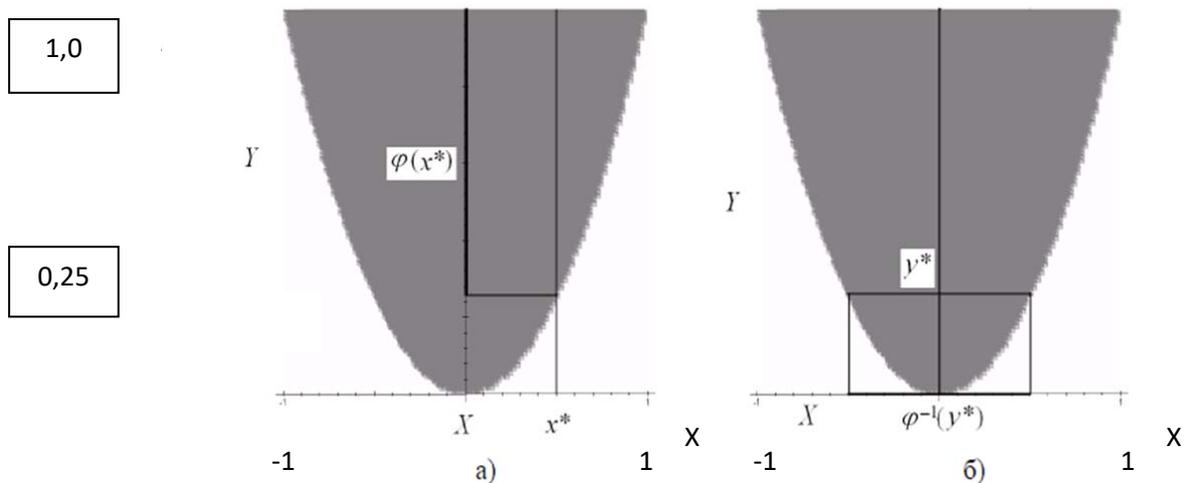


Рис. 5.4. Четкое многозначное отображение

Иногда на отображение накладывается дополнительное условие: образ любого элемента должен состоять не более чем из одного элемента. В этом случае говорят об однозначных (функциональных) отображениях.

Для отображения f из X в Y обратным отображением f^{-1} называется такое отображение из Y в X , что $(y, x) \subset f^{-1}(x, y) \subset f$. Образ элемента $y \subset Y$ при обратном отображении обозначают $f^{-1}(y)$. Очевидно, он является подмножеством множества X .

Пример 4. В моделях принятия решений важную роль играет отображение, которое каждому возможному действию ЛПР ставит в соответствие реакцию системы на это действие. Пусть, например, действие ЛПР (менеджера по продажам) состоит в выборе суммы инвестиций в рекламу из множества X , а состояние системы (фирмы) описывается суммой ее продаж из множества Y . Однозначное отображение f из X в Y каждой сумме инвестиций ставит в соответствие сумму результирующих продаж, таким образом определяется модель системы (фирмы). Типичный вид этого отображения можно задать формулой

$$f(x) = s(1 - \exp(-\Lambda x)),$$

где параметр s – это максимальная емкость рынка; Λ – эластичность спроса.

Пусть $s = 10^6$, $\Lambda = 10^{-5}$. Тогда вложение в рекламу суммы $x = 100\,000$ приводит к сумме продаж $y = 632\,121$, т. е. точка $y = 632\,121$ – образ точки $x = 100\,000$ при отображении $f(x) = s(1 - \exp(-\Lambda x))$.

Если четкое отображение – это подмножество декартова произведения $X \times Y$ области определения и области значений, что же такое нечеткое отображение? Очевидно, нечеткое подмножество $Y \times X$. Тогда нечеткое отображение \tilde{f} множества X во множество Y можно описать его функцией принадлежности $\mu_{\tilde{f}} : X \times Y \subset [0; 1]$. Функция принадлежности $\mu_{\tilde{f}}(x, y)$ определяет степень достоверности того, что точка y принадлежит образу точки x при нечетком отображении \tilde{f} . И как образом элемента $x \subset X$ при четком отображении было четкое подмножество множества Y , так же образом $x \subset X$ при нечетком отображении будет нечеткое подмножество множества Y с функцией принадлежности $\mu_{\tilde{f}}(x, y)$.

Образом четкого множества при нечетком отображении будет объединение образов его элементов:

$$\mu_{\tilde{f}(A)}(y) := \sup_{x \subset A} \mu_{\tilde{f}}(x, y).$$

Однако, чтобы завершить обобщение понятия образа на нечеткий случай, необходимо определить образ нечеткого множества при

нечетком же отображении. Понятно, что образы элементов нечеткого множества должны объединяться с учетом степени принадлежности этих элементов нечеткому множеству. Запишем формулу для образа четкого множества следующим эквивалентным образом через функцию принадлежности четкого множества A :

$$\mu_{\tilde{f}(A)}(y) := \sup_{x \in X} \min[\mu_A(x); \mu_{\tilde{f}}(x, y)].$$

Эта формула уже допускает непосредственное обобщение на нечеткий случай (замена четкого множества A на нечеткое множество \tilde{A}), что позволяет дать следующее определение. Заметим, что x фиксировано и выражение $\mu_{\tilde{f}}(x, y)$ действительно задает функцию принадлежности нечеткого подмножества множества Y .

Определение 10. Образом $\tilde{f}(\tilde{A})$ нечеткого множества $\tilde{A} \subset X$ при нечетком отображении $\tilde{f}: X \rightarrow Y$ называется нечеткое подмножество множества Y с функцией принадлежности

$$\mu_{\tilde{f}(\tilde{A})}(y) := \sup_{x \in X} \min[\mu_{\tilde{A}}(x); \mu_{\tilde{f}}(x, y)]. \quad (*)$$

В частности, если отображение $f: X \rightarrow Y$ четкое, то формулу (*) можно упростить, так как в операции минимума остаются только образы точки y при обратном четком отображении f^{-1} . Действительно,

$$\mu_{f(\tilde{A})}(y) := \sup_{x \in X} \min[\mu_{\tilde{A}}(x); \mu_f(x, y)] = \sup_{x: f(x)=y} \mu_{\tilde{A}}(x) = \sup_{x \in f^{-1}(y)} \mu_{\tilde{A}}(x).$$

Пример 5. Пусть по условиям предыдущего примера реакция системы известна ЛПР лишь неточно, т. е. отображение \tilde{f} нечеткое. Для каждого выбора суммы инвестиций $x \in X$ его образ при отображении \tilde{f} – нечеткое множество возможных продаж с функцией принадлежности $\mu_{\tilde{f}(x)}(y) = \max[0; 1 - (y - f(x))^2/x^2]$, в котором функция «наиболее достоверных продаж» $f(x) = s(1 - \exp(-\Lambda x))$ взята из предыдущего примера ($s = 10^6$, $\Lambda = 10^{-5}$).

Функция принадлежности отображения \tilde{f} показана на рис. 5.5. Большему значению функции принадлежности соответствует более темный тон. На этом же рисунке изображена кривая $f(x) = s(1 - \exp(-\Lambda x))$, на которой функция принадлежности нечеткого отображения принимает максимальное значение, равное единице. На рис. 5.6 изображены функции принадлежности образов точек $x_1 = 1800$, $x_2 = 2000$, $x_3 = 2200$. Они представляют собой нечеткую реакцию системы на выбор соответ-

ствующих объемов инвестиций. Выбор конкретного объема инвестиций можно понимать как четкую инструкцию «что делать?». Пусть, однако, ЛПР (менеджер) хочет узнать реакцию системы (фирмы) на выбор инвестиций в рекламу в размере «примерно 60 000».

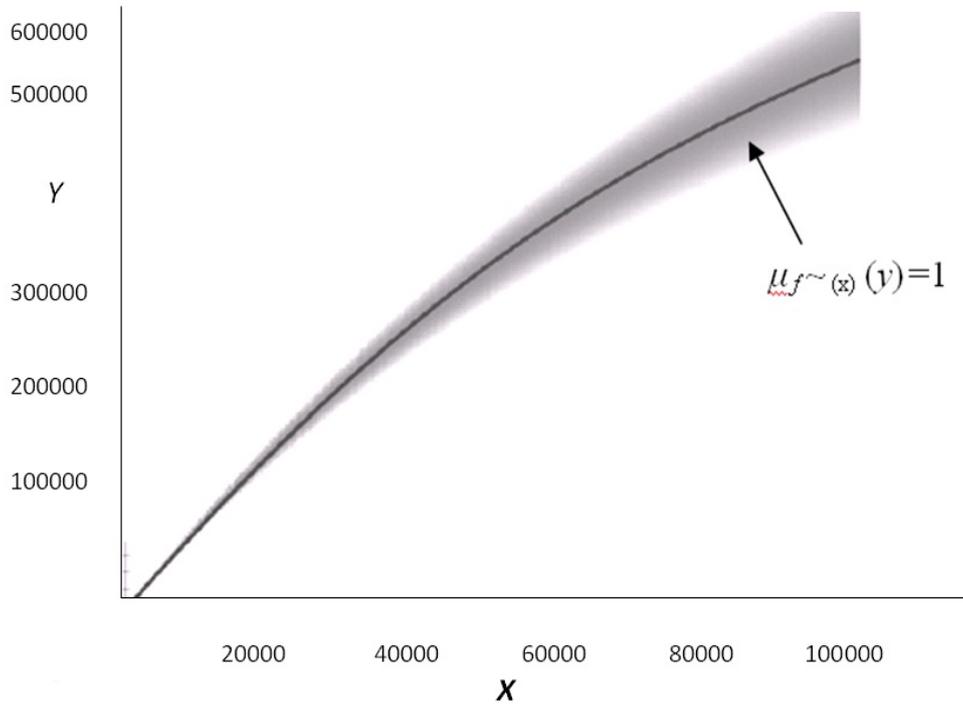


Рис. 5.5. Функция принадлежности нечеткого образа

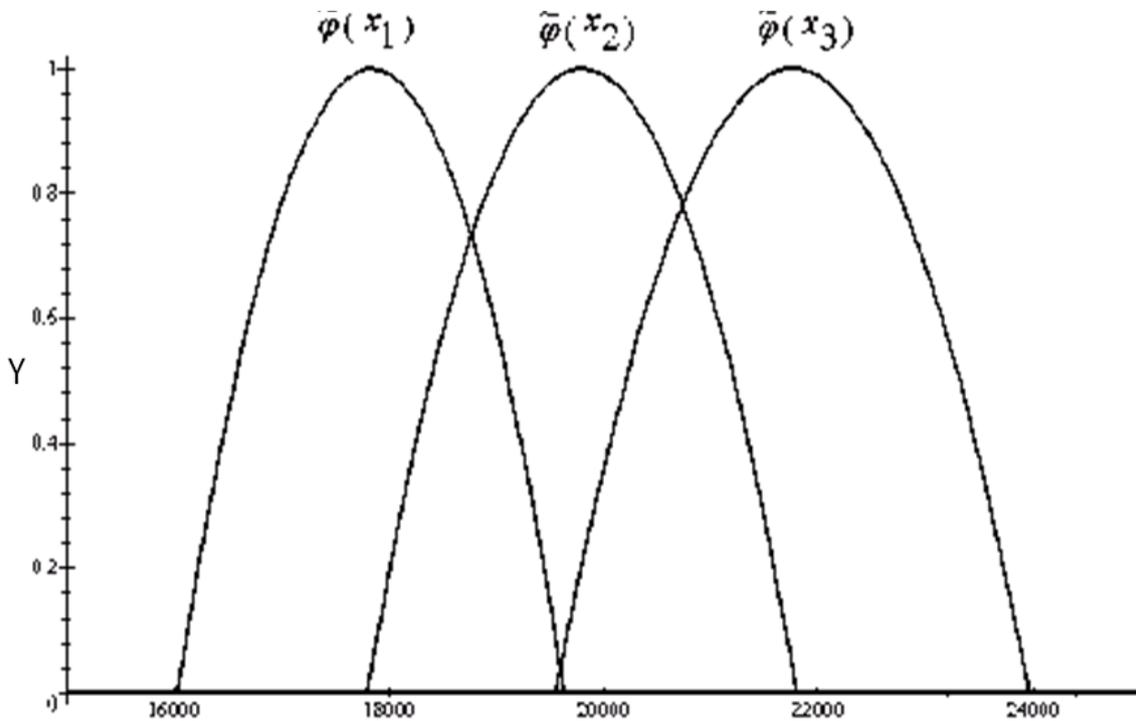


Рис. 5.6. Образы точек при нечетком отображении

Эта нечеткая инструкция описывается некоторым нечетким множеством \tilde{A} , и реакция системы будет образом $\tilde{f}(\tilde{A})$ этого нечеткого множества при нечетком отображении \tilde{f} . Образы нечетких множеств удобно строить, как на рис. 5.7. На нем, так же как и на рис. 5.5, изображена функция принадлежности нечеткого отображения \tilde{f} . Горизонтальная ось соответствует множеству X , и от нее вниз «вверх ногами» строится функция принадлежности нечеткого множества \tilde{A} – «примерно 60 000». Вертикальная ось соответствует множеству Y , и от нее влево (повернутой на 90°) строится функция принадлежности нечеткого образа $\tilde{f}(\tilde{A})$. Стрелки на рис. 5.7 показывают направление «переноса» точек нечеткого множества: образ нечеткого множества – объединение образов точек этого множества с учетом их степени принадлежности.

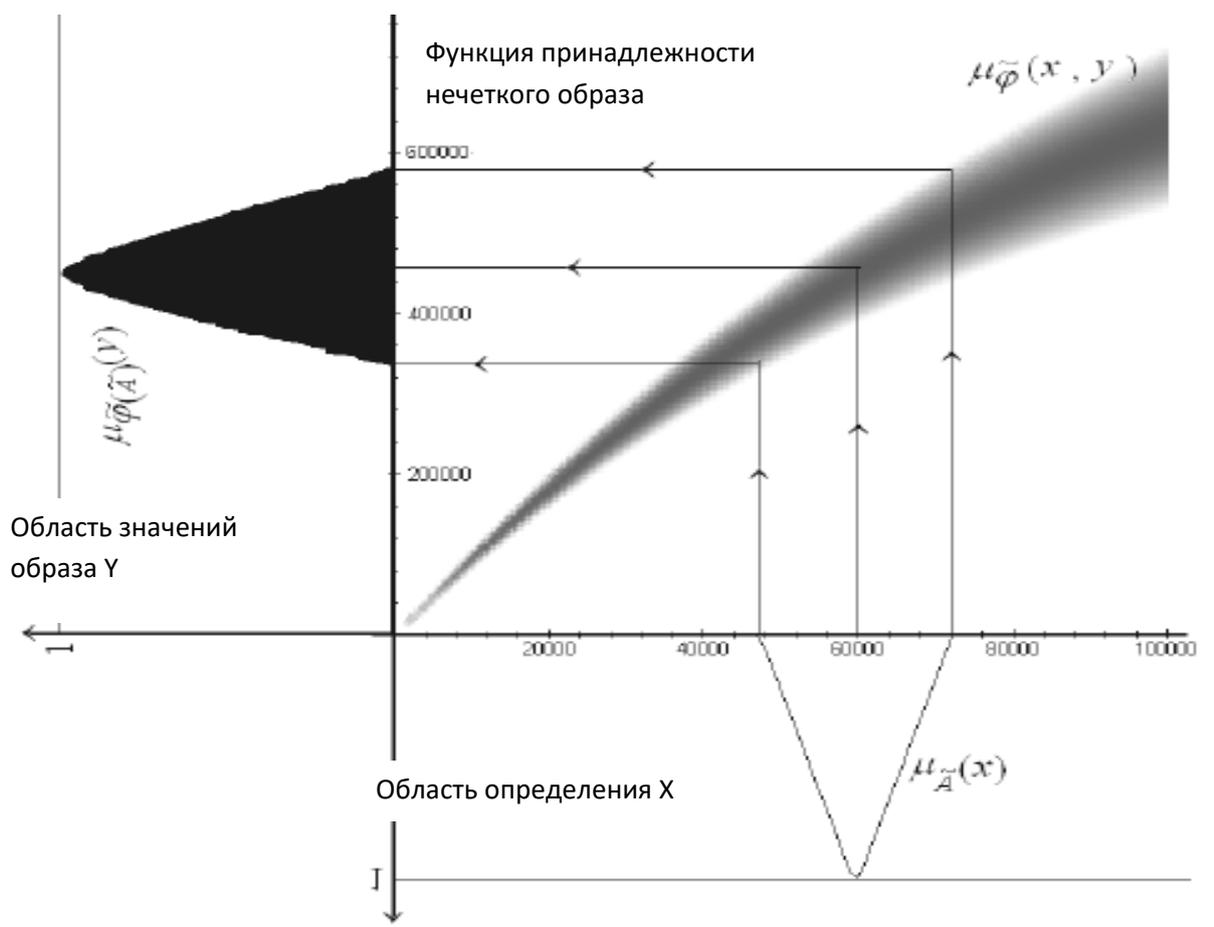


Рис. 5.7. Образ нечеткого множества при нечетком отображении (модель нечеткого объекта управления)

5.4. Прообраз нечеткого множества при нечетком отображении

Понятие образа нечеткого множества при нечетком отображении позволяет ЛПР вычислять нечеткую реакцию системы на нечеткие же управляющие воздействия. Вместе с тем для теории принятия решений гораздо важнее обратная задача – найти действия, которые приводят к желаемому результату.

Решить данную задачу позволяет понятие прообраза нечеткого множества.

Определение 11. Прообразом $\tilde{A} \subset X$ нечеткого множества $\tilde{B} \subset Y$ при нечетком отображении $\tilde{f} : X \rightarrow Y$ называется объединение всех нечетких множеств \tilde{a} таких, что их образ $\tilde{f}(\tilde{a})$ принадлежит нечеткому множеству \tilde{B} , т. е. таких, что

$$\sup_{x \in X} \min[\mu_{\tilde{a}}(x); \mu_{\tilde{f}}(x, y)] \leq \mu_{\tilde{B}}(y) \text{ для всех } y \in Y.$$

Содержательно прообраз нечеткого множества $\tilde{B} \subset Y$ – это «максимальное» нечеткое множество $\tilde{A} \subset X$, переходящее в множество \tilde{B} при нечетком отображении $\tilde{f} : X \rightarrow Y$. Таким образом, чтобы для некоторой нечеткой реакции системы определить то действие (возможно, нечеткое), которое приводит к данной реакции, необходимо найти прообраз нечеткого множества реакции.

Однако не любое нечеткое множество имеет непустой прообраз при нечетком отображении. Пустота прообраза «одноточечного» множества имеет простое содержательное объяснение. Это множество можно интерпретировать, как желание ЛПР получить единственный исход с положительной достоверностью, обеспечив нулевую достоверность остальных исходов. Но это невозможно, так как поведение системы нечетко и выбор любого действия приводит к нескольким возможным исходам. Вывод здесь прост: ЛПР должно ставить перед собой реальные цели, смягчая требования к нечеткому множеству результата.

Вычисление прообразов нечетких множеств имеет важное прикладное значение. В то же время вычисление «по определению» весьма трудоемко. Следующий приводимый без доказательства результат позволяет дать более простую характеристику прообраза нечеткого множества $\tilde{B} \subset Y$, пригодную для численной реализации.

Определим следующие четкие множества.

$N : \{(x, y) \in X \times Y : \mu_{\tilde{f}}(x, y) > \mu_{\tilde{B}}(y)\}$ – множество пар элементов из области определения и области значений отображения, в которых значение функции принадлежности отображения строго превышает значение функции принадлежности множества \tilde{B} .

$N_x : \{y \in Y : (x, y) \in N\}$ – «срез» множества N при фиксированном $x \in X$.

$X^0 := \{x \in X : N_x \neq \emptyset\}$ – множество элементов области определения, для которых множество N_x не пусто.

Теорема. Функция принадлежности прообраза $\tilde{A} \subset X$ нечеткого множества $\tilde{B} \subset Y$ при нечетком отображении $\tilde{f} : X \rightarrow Y$ описывается выражением

$$\mu_{\tilde{A}}(x) = \begin{cases} \inf_{y \in N_x} \mu_{\tilde{B}}(y), & x \in X^0 \text{ – минимальная граница, где } N_x \\ & \text{не пустое,} \\ 1, & x \in X \setminus X^0. \end{cases}$$

Проиллюстрируем нахождение прообраза нечеткого множества с помощью этой теоремы следующим примером.

Пример 6. Рассмотрим нечеткое отображение $\tilde{f} : X \rightarrow Y$ из примера 5. Пусть необходимо найти прообраз нечеткого множества $\tilde{B} \subset Y$, функция принадлежности которого изображена на рис. 5.8 слева (так же как и на рис. 5.7, она изображена повернутой влево на 90°). Также на рисунке для заданных \tilde{f} и \tilde{B} построено множество N (оно изображено черным цветом). Для двух точек x_1, x_2 на оси Y изображены множества N_{x_1}, N_{x_2} – «вертикальные срезы» множества N в этих точках. Из рисунка видно, что множество X^0 в данном примере совпадает с множеством X , так как для любой точки $x \in X$ множество N_x не пусто.

Следовательно, для вычисления значения функции принадлежности прообраза в заданной точке $x \in X$ необходимо найти минимальное значение $\mu_{\tilde{B}}(y)$ на множестве N_x . Так, например, $\mu_{\tilde{A}}(x_1) = \mu_{\tilde{A}}(x_2) = 0$, поскольку соответствующие множества N_{x_1} и N_{x_2} включают в себя точки, в которых $\mu_{\tilde{B}}(\cdot) = 0$. Хотя прообраз в данном примере не пуст, но его образ не совпадает с исходным множеством \tilde{B} (максимальное значение функции принадлежности прообраза равно примерно 0,8, значит, по формуле (*) максимальное значение функции принадлеж-

ности образа не может превышать 0,8), так как отображение \tilde{f} «слишком нечеткое», для того чтобы обеспечить острый пик функции принадлежности множества \tilde{B} .

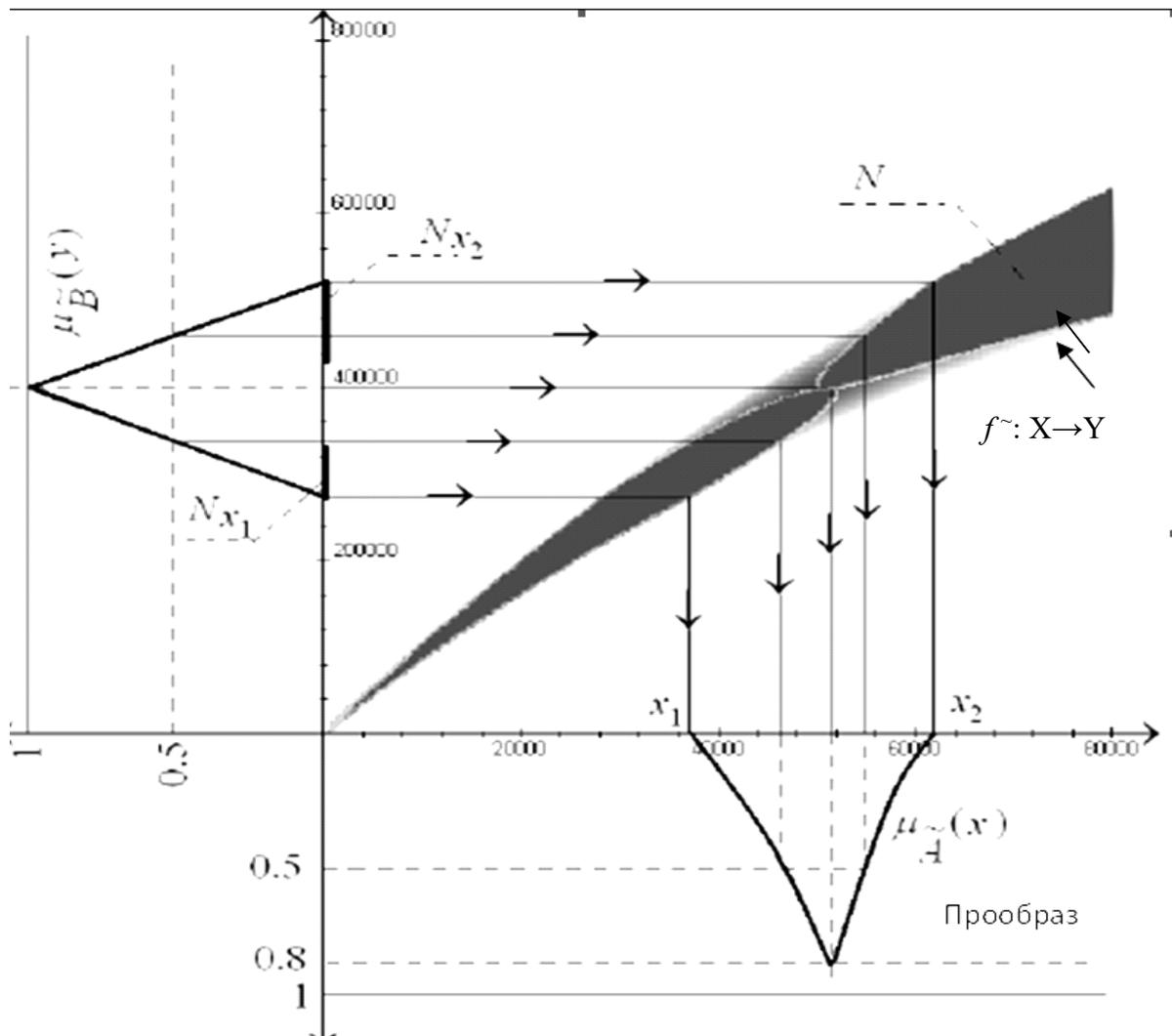


Рис. 5.8. Прообраз нечеткого множества при нечетком отображении

Контрольные вопросы

1. Объясните понятия четких и нечетких множеств. Приведите их определения.
2. Раскройте особенности операций над четкими и нечеткими множествами.
3. Поясните нечеткие отображения в задачах принятия решений.
4. Раскройте понятия образов четкого и нечеткого множеств.
5. Определите понятие прообраза нечеткого множества при нечетком отображении.

Глава 6

ОСНОВНЫЕ ПОЛОЖЕНИЯ ТЕОРИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Под нейронными сетями подразумевают вычислительные структуры, которые моделируют простые биологические процессы, обычно ассоциируемые с процессами человеческого мозга. Они представляют собой распределенные и параллельные системы, способные к адаптивному обучению путем анализа положительных и отрицательных воздействий. Элементарный преобразователь в данных сетях – искусственный нейрон, или просто нейрон, названный так по аналогии с биологическим прототипом.

К настоящему времени предложено и изучено большое количество моделей нейроноподобных элементов и нейронных сетей, ряд из которых будет рассмотрен ниже.

6.1. Структура и свойства искусственного нейрона

Формальный нейрон (рис. 6.1) представляет собой элемент, выполняющий взвешенное суммирование значений координат входного вектора с последующим нелинейным преобразованием суммы. Коэффициенты суммирования называются синаптическими весами, а нелинейная функция – функцией активации нейрона. Такая схема в общих чертах отражает функционирование нейрона головного мозга: функция активации моделирует возбуждение нейрона, а весовые коэффициенты при входных координатах соответствуют обработке в синапсах нейрона. Выходной сигнал формируется на единственном выходе, который в нейрофизиологии называется аксоном. Нейроны группируются в нейронные слои и образуют нейронную сеть [14, 15].

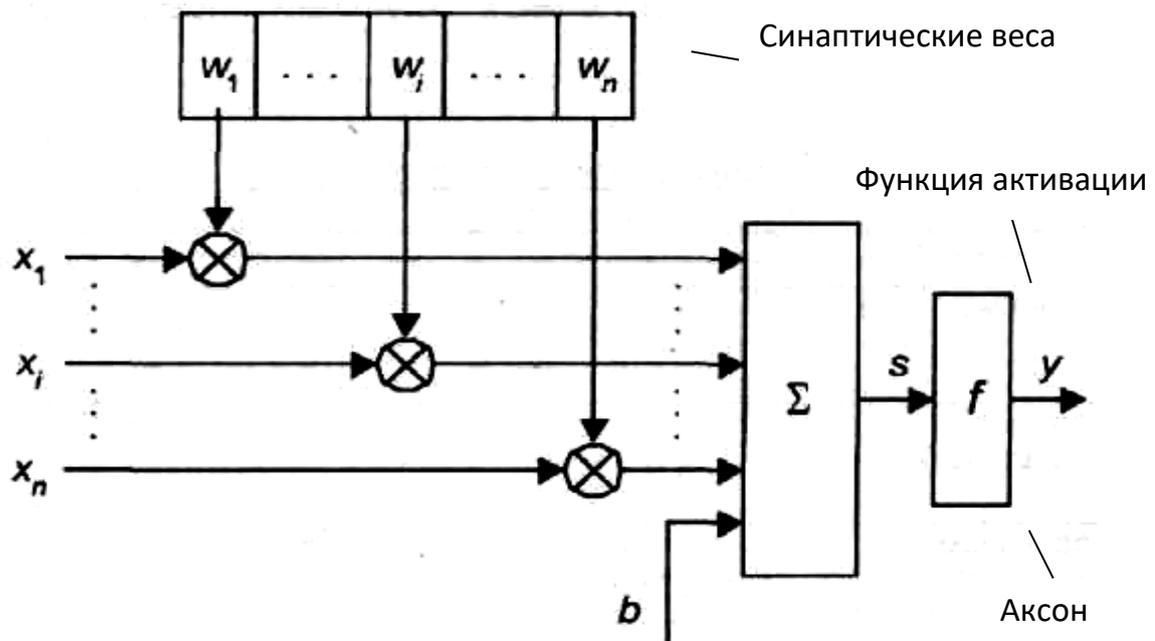


Рис. 6.1. Структура искусственного нейрона

Текущее состояние нейрона определяется как взвешенная сумма его входов:

$$s = \sum_{i=1}^n x_i w_i + b, \quad (6.1)$$

где s – результат суммирования (sum); n – число входов нейрона; x_i – компонент входного вектора (входной сигнал), $i = 1 \dots n$; w_i – вес (weight) синапса, $i = 1 \dots n$; b – значение смещения (bias).

Выход нейрона есть функция его состояния

$$y = f(s), \quad (6.2)$$

где y – выходной сигнал нейрона; f – нелинейное преобразование (функция активации).

В общем случае входной сигнал, весовые коэффициенты и смещение могут принимать действительные значения, а во многих практических задачах – лишь некоторые фиксированные значения. Выход y определяется видом функции активации и может быть как действительным, так и целым. Синаптические связи с положительными весами называют возбуждающими, с отрицательными весами – тормозящими.

Описанный вычислительный элемент можно считать упрощенной математической моделью биологических нейронов. Чтобы подчеркнуть различие нейронов биологических и искусственных, вторые иногда называют нейроноподобными элементами, или формаль-

ными нейронами. На входной сигнал (s) нелинейный преобразователь отвечает выходным сигналом $f(s)$, который представляет собой выход у нейрона.

Простейший вариант нейронной сети – однослойный персептрон Розенблатта. Что же может такая простая схема? Оказывается, уже не так мало – произвести анализ входного вектора данных и распознать среди них вектор, который наилучшим образом соответствует карте синаптических весов. Фактически это решение задачи распознавания. При определенной настройке синаптических весов однослойный персептрон способен выполнять ряд логических функций, однако не все: для реализации функции исключающего «ИЛИ» одного слоя принципиально недостаточно. В этом случае используют многослойные нейронные сети.

Многослойные нейронные сети способны с различной степенью точности аппроксимировать любую из существующих логических функций. Качество аппроксимации функций нейронной сетью, очевидно, зависит от числа слоев и числа нейронов в каждом слое. Интуитивно можно предположить, что с увеличением числа слоев и числа нейронов качество нейронной сети улучшится, и это действительно так, но до некоторого предела, после которого увеличение объема становится неэффективным. Уровень порога связан с размерами входного и выходного векторов, а также с внутренней структурой обрабатываемых данных.

Нелинейная функция f называется активационной и может иметь различный вид, как показано на рис. 6.2 (табл. 6.1). Одна из наиболее распространенных – нелинейная функция с насыщением, так называемая логистическая функция, или сигмоид (т. е. функция S -образного вида).

Таблица 6.1

Функции активации нейронов [15]

Название	Формула	Область значений
Линейная	$f(s) = ks$	$(-\infty, \infty)$
Полулинейная	$f(s) = \begin{cases} ks, & s > 0, \\ 0, & s \leq 0. \end{cases}$	$(0, \infty)$
Логистическая	$f(s) = \frac{1}{1 + e^{-as}}$	$(0, 1)$

Название	Формула	Область значений
Гиперболический тангенс	$f(s) = \frac{e^{as} - e^{-as}}{e^{as} + e^{-as}}$	(-1, 1)
Экспоненциальная	$f(s) = e^{-as}$	(0, ∞)
Синусоидальная	$f(s) = \sin(s)$	(-1, 1)
Сигмоидальная	$f(s) = \frac{s}{a + s }$	(-1, 1)
Линейная с насыщением	$f(s) = \begin{cases} -1, & s \leq -1, \\ s & -1 < s < 1, \\ 1 & s \geq 1. \end{cases}$	(-1, 1)
Пороговая	$f(s) = \begin{cases} 0, & s < 0, \\ 1, & s \geq 0. \end{cases}$	(0, 1)
Модульная	$f(s) = s $	(0, ∞)
Знаковая (сигнатурная)	$f(s) = \begin{cases} 1, & s > 0, \\ -1 & s \leq 0. \end{cases}$	(-1, 1)
Квадратичная	$f(s) = s^2$	(0, ∞)

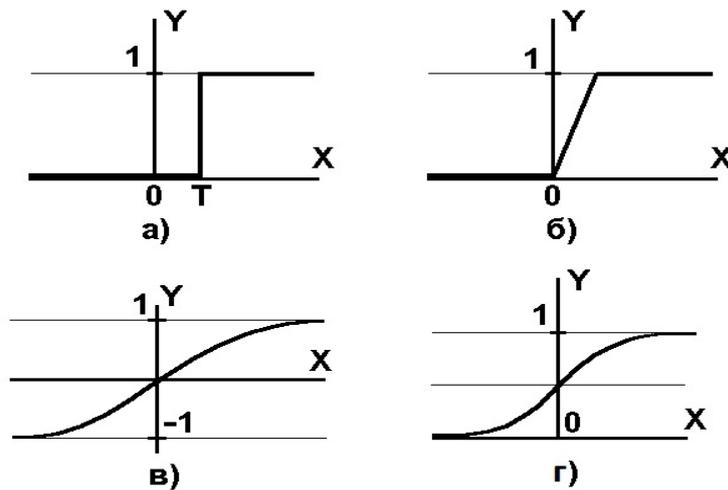


Рис. 6.2. Виды активационных функций:
 а – функция единичного скачка; б – линейный порог (гистерезис); в – сигмоид (гиперболический тангенс); г – сигмоид

Сигмоид определяется по следующей формуле:

$$f(x) = \frac{1}{1 + e^{-\alpha x}}. \quad (6.3)$$

При уменьшении α сигмоид становится более пологим, в пределе при $\alpha = 0$ вырождаясь в горизонтальную линию на уровне 0,5. При увеличении α сигмоид приближается по внешнему виду к функции единичного скачка с порогом T в точке $x = 0$. Из выражения (6.3) для сигмоида очевидно, что выходное значение нейрона лежит в диапазоне $[0, 1]$. Одно из ценных свойств сигмоидной функции – простое выражение для ее производной.

$$f'(x) = \alpha f(x) (1 - f(x)). \quad (6.4)$$

Следует отметить, что сигмоидная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того, она обладает свойством усиливать слабые сигналы и предотвращает насыщение от больших сигналов, так как они соответствуют областям аргументов, где сигмоид имеет пологий наклон.

6.2. Классификация нейронных сетей и их свойства

Нейронная сеть представляет собой совокупность нейроноподобных элементов, определенным образом соединенных друг с другом и с внешней средой с помощью связей, определяемых весовыми коэффициентами. В зависимости от функций, выполняемых нейронами в сети, можно выделить три их типа [2]:

1) *входные нейроны*, на которые подается вектор, кодирующий входное воздействие или образ внешней среды; в них обычно не осуществляются вычислительные процедуры, а информация передается с входа на выход путем изменения активации нейронов;

2) *выходные нейроны*, выходные значения которых представляют выходы нейронной сети; преобразования в них осуществляются по выражениям (6.1) и (6.2);

3) *промежуточные нейроны*, составляющие основу нейронных сетей, преобразования в которых выполняются также по выражениям (6.1) и (6.2).

С точки зрения топологии можно выделить три основных типа нейронных сетей:

- полносвязные (рис. 6.3, а);
- многослойные, или слоистые (рис. 6.3, б);
- слабосвязные (с локальными связями) (рис. 6.3, в).

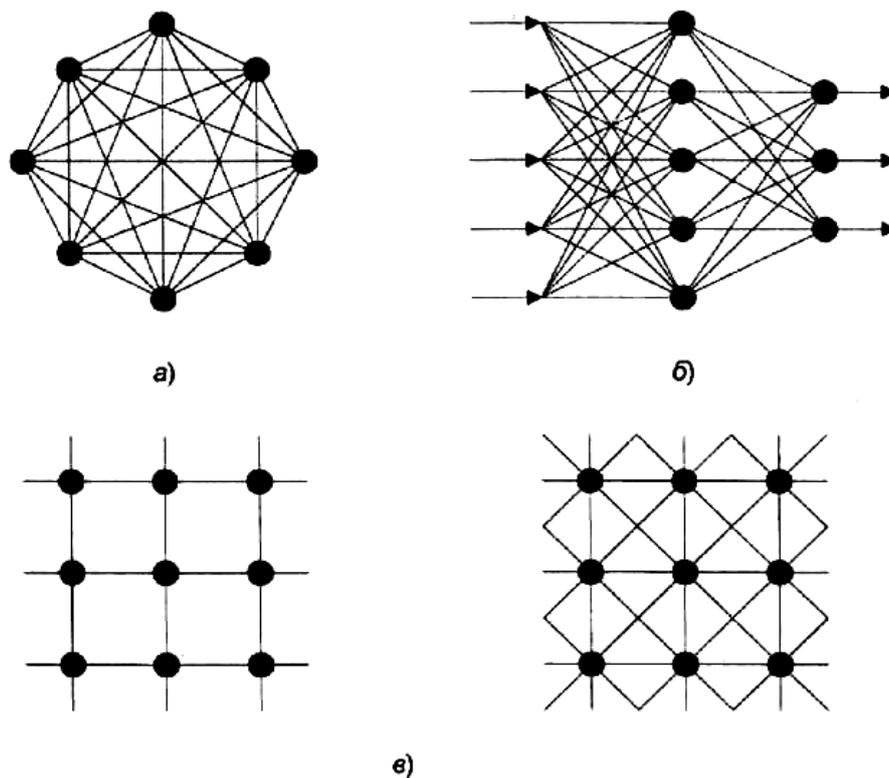


Рис. 6.3. Архитектуры нейронных сетей:
a – полносвязная сеть; *б* – многослойная сеть
с последовательными связями; *в* – слабосвязные сети

В *полносвязных нейронных сетях* каждый нейрон передает свой выходной сигнал остальным нейронам, в том числе и самому себе. Все входные сигналы подаются всем нейронам. Выходными сигналами сети могут быть все или некоторые выходные сигналы нейронов после нескольких тактов функционирования сети.

В *многослойных нейронных сетях* нейроны объединяются в слои. Слой содержит совокупность нейронов с едиными входными сигналами. Число нейронов в слое может быть любым и не зависит от количества нейронов в других слоях. В общем случае сеть состоит из Q слоев, пронумерованных слева направо. Внешние входные сигналы подаются на входы нейронов входного слоя (его часто нумеруют как нулевой), а выходами сети являются выходные сигналы последнего слоя. Кроме входного и выходного слоев в многослойной нейронной сети есть один или несколько скрытых слоев. Связи от выходов нейронов некоторого слоя q к входам нейронов следующего слоя $(q + 1)$ называются последовательными.

Среди многослойных нейронных сетей выделяют следующие типы.

1. Монотонные. Это частный случай слоистых сетей с дополнительными условиями на связи и нейроны. Каждый слой, кроме последнего (выходного), разбит на два блока: возбуждающий и тормозящий. Связи между блоками тоже разделяются на тормозящие и возбуждающие. Если от нейронов блока A к нейронам блока B ведут только возбуждающие связи, это означает, что любой выходной сигнал блока B – это монотонная неубывающая функция любого выходного сигнала блока A . Если же эти связи только тормозящие, то любой выходной сигнал блока B – это невозрастающая функция любого выходного сигнала блока A . Для нейронов монотонных сетей характерна монотонная зависимость выходного сигнала нейрона от параметров входных сигналов.

2. Сети без обратных связей. В таких сетях нейроны входного слоя получают входные сигналы, преобразуют их и передают нейронам первого скрытого слоя и так далее вплоть до выходного слоя, который выдает сигналы для интерпретатора и пользователя. Если не оговорено противоположное, то каждый выходной сигнал q -го слоя будет подан на вход всех нейронов $(q + 1)$ слоя; однако возможен вариант соединения q -го слоя с произвольным $(q + p)$ слоем.

Среди многослойных сетей без обратных связей различают полносвязные (выход каждого нейрона q -го слоя связан с входом каждого нейрона $(q + 1)$ слоя) и частично полносвязные сети. Классический вариант слоистых сетей – полносвязные сети прямого распространения.

3. Сети с обратными связями. В сетях с обратными связями информация с последующих слоев передается на предыдущие. Среди них выделяют следующие:

– слоисто-циклические, отличаются тем, что слои замкнуты в кольцо: последний слой передает свои выходные сигналы первому; все слои равноправны и могут как получать входные сигналы, так и выдавать выходные;

– слоисто-полносвязные, состоят из слоев, каждый из которых представляет собой полносвязную сеть, а сигналы передаются как от слоя к слою, так и внутри слоя; в каждом слое цикл работы распадается на три части: прием сигналов с предыдущего слоя, обмен сиг-

налами внутри слоя, выработка выходного сигнала и передача к последующему слою;

– полносвязно-слоистые, по своей структуре аналогичны слоисто-полносвязным, но функционируют по-другому: в них не разделяются фазы обмена внутри слоя и передачи следующему, на каждом такте нейроны всех слоев принимают сигналы от нейронов как своего слоя, так и последующих.

В *слабосвязных нейронных сетях* нейроны располагаются в узлах прямоугольной или гексагональной решетки. Каждый нейрон связан с четырьмя (окрестность фон Неймана), шестью (окрестность Голея) или восемью (окрестность Мура) своими ближайшими соседями.

Известные нейронные сети можно разделить по типам структур нейронов на гомогенные (однородные) и гетерогенные. *Гомогенные сети* состоят из нейронов одного типа с единой функцией активации, а в *гетерогенную сеть* входят нейроны с различными функциями активации.

Существуют *бинарные и аналоговые сети*. Первые из них оперируют только двоичными сигналами, и выход каждого нейрона может принимать значение либо логического нуля (заторможенное состояние), либо логической единицы (возбужденное состояние).

Еще одна классификация делит нейронные сети на *синхронные и асинхронные*. В первом случае в каждый момент времени лишь один нейрон меняет свое состояние, во втором – состояние меняется сразу у целой группы нейронов, как правило, у всего слоя. Алгоритмически ход времени в нейронных сетях задается итерационным выполнением однотипных действий над нейронами. Далее будут рассматриваться только синхронные сети.

6.3. Алгоритмы обучения нейронных сетей

Процесс обучения нейронной сети может рассматриваться как настройка архитектуры и весов связей для эффективного выполнения поставленной задачи. Обычно итеративная настройка весов связей осуществляется в соответствии с обучающей выборкой. Свойство сети обучаться на примерах делает ее более привлекательной по сравнению с системами, которые следуют набору правил функционирования, сформулированных экспертами [16].

Для организации процесса обучения, во-первых, надо иметь модель внешней среды, в которой функционирует нейронная сеть. Эта модель определяет парадигму обучения. Во-вторых, необходимо понять, как модифицировать весовые параметры сети.

Существуют *три парадигмы обучения* [16]: *с учителем, без учителя* (самообучение) и *смешанная*. В первом случае на каждый входной пример существует требуемый ответ. Веса настраиваются таким образом, чтобы выходы сети были как можно более близкими к требуемым ответам. Более «жесткий» вариант обучения с учителем предполагает, что известна только критическая оценка правильности выхода нейронной сети, а не сами требуемые значения выхода.

Для обучения без учителя не нужно знание требуемых ответов на каждый пример обучающей выборки. В этом случае происходит распределение образцов по категориям (кластерам) в соответствии с внутренней структурой данных или степенью корреляции между образцами.

При смешанном обучении весовые коэффициенты одной группы нейронов настраиваются, как при обучении с учителем, а другой группы – на основе самообучения.

В процессе обучения учитываются следующие свойства нейронных сетей [16]: емкость сети, сложность образцов и вычислительная сложность. Под *емкостью сети* понимается число запоминаемых образцов с учетом сформированных функций и границ принятия решений. Сложность образцов определяет число обучающих примеров, необходимых для достижения способности сети к обобщению.

Известны *четыре основных правила обучения*, обусловленные связанными с ними архитектурами сетей (табл. 6.2): коррекция ошибки, правило Больцмана, правило Хебба и метод соревнования [15].

Коррекция ошибки. Для каждого входного примера задан требуемый выход d , который может не совпадать с реальным u . Правило обучения при коррекции ошибки состоит в использовании разницы $d - u$ для изменения весов с целью уменьшения ошибки рассогласования. Обучение проводится только в случае ошибочного результата. Известны многочисленные модификации этого правила обучения.

Правило Больцмана. Это стохастическое правило обучения обусловлено аналогией с термодинамическими принципами. В результате его выполнения осуществляется настройка весовых коэффициентов нейронов в соответствии с требуемым распределением вероятностей.

Обучение правилу Больцмана может рассматриваться как отдельный случай коррекции ошибки, в котором под ошибкой понимается расхождение корреляций состояний в двух режимах.

Правило Хебба. Это самый известный алгоритм обучения нейронных сетей, суть которого заключается в следующем: если нейроны с обеих сторон синапса возбуждаются одновременно и регулярно, то сила синаптической связи возрастает. Важная особенность состоит в том, что изменение синаптического веса зависит только от активности связанных этим синапсом нейронов. Предложено большое количество разновидностей этого правила, различающихся особенностями модификации синаптических весов.

Метод соревнования. В отличие от правила Хебба, в котором множество выходных нейронов могут возбуждаться одновременно, здесь выходные нейроны соревнуются между собой. И выходной нейрон с максимальным значением взвешенной суммы является «победителем» («победитель забирает все»). Выходы же остальных выходных нейронов устанавливаются в неактивное состояние. При обучении модифицируются только веса нейрона-«победителя» в сторону увеличения близости к данному входному примеру. Это правило позволяет группировать входные данные на категории (кластеры) и представлять их отдельными выходными нейронами.

Таблица 6.2

Известные алгоритмы обучения

Парадигма	Обучающее правило	Архитектура нейронной сети	Алгоритм обучения	Задачи
С учителем	Коррекция ошибки	Однослойный и многослойный персептрон	Алгоритмы обучения персептрона Обратное распространение	Классификация образов Аппроксимация функций Предсказание Управление
	Правило Больцмана	Рекуррентная	Алгоритм обучения Больцмана	Классификация образов
	Правило Хебба	Многослойная, прямого распространения	Линейный дискриминантный анализ	Анализ данных Классификация образов

Окончание табл. 6.2

Парадигма	Обучающее правило	Архитектура нейронной сети	Алгоритм обучения	Задачи
С учителем	Метод соревнования	Соревнование	Векторное квантование	Категоризация внутри класса Сжатие данных
Без учителя	Коррекция ошибки	Многослойная, прямого распространения	Проекция Саммона	Категоризация внутри класса Анализ данных
	Правило Хебба	Прямого распространения или соревнование	Метод главных компонентов	Анализ данных Сжатие данных
	Метод соревнования	Соревнование	Векторное квантование	Категоризация Сжатие данных
Смешанная	Коррекция ошибки и метод соревнования	Сеть RBFN	Алгоритм обучения RBFN	Классификация образов Аппроксимация функций Предсказание Управление

Нейронная сеть считается устойчивой, если после конечного числа итераций обучения ни один из примеров обучающей выборки не изменяет своей принадлежности в кластерах. Однако сеть не перестанет обучаться, если параметр скорости обучения не равен нулю. Но эта искусственная остановка обучения вызывает другую проблему, называемую пластичностью и связанную со способностью сети к адаптации к новым данным. Возникает дилемма стабильности – пластичности Гроссберга.

Список представленных в табл. 6.2 алгоритмов обучения нейронных сетей не исчерпывающий. В последней колонке перечислены задачи, для которых могут быть применены эти алгоритмы. Каждый алгоритм обучения ориентирован на сеть определенной архитектуры и предназначен для ограниченного класса задач.

На основе табл. 6.2 для решения задачи прогнозирования были выбраны следующие виды сетей:

- сеть радиального основания (RBFN);
- многослойный персептрон (MLP);
- однослойный персептрон (Leniar).

Далее они будут рассмотрены более подробно.

6.4. Обратное распространение (Neural Network with Back Propagation Training Algorithm)

Сети обратного распространения (ошибки) (MultiLayer Perceptron with Back Propagation Training Algorithm (MLP, многослойный перцептрон с обучением по методу обратного распространения ошибки)) были предложены и исследованы в 1960-х гг. Ф. Розенблаттом, М. Минским, С. Пейпертом и др. [2]. Лишь в середине 1980-х гг. был предложен эффективный алгоритм обучения многослойных перцептронов, основанный на вычислении градиента функции ошибки и названный обратным распространением ошибки.

Алгоритм обратного распространения – это итеративный градиентный алгоритм, который используется с целью минимизации среднеквадратического отклонения текущего выхода многослойного перцептрона и требуемого выхода. Он используется для обучения многослойных нейронных сетей с последовательными связями. Нейроны в таких сетях делятся на группы с общим входным сигналом – слои. На каждый нейрон первого слоя подаются все элементы внешнего входного сигнала. Все выходы нейронов q -го слоя подаются на каждый нейрон слоя $(q + 1)$. Нейроны выполняют взвешенное суммирование входных сигналов. К сумме элементов входных сигналов, помноженных на соответствующие синаптические веса, прибавляется смещение нейрона. Над результатом суммирования выполняется нелинейное преобразование – функция активации (передаточная функция). Значение функции активации есть выход нейрона.

Характеристики сети следующие. Тип входных сигналов – целые или действительные числа. Тип выходных сигналов – действительные числа из интервала, заданного передаточной функцией нейронов. Тип передаточной функции – сигмоидальная. В нейронных сетях применяют несколько вариантов сигмоидальных передаточных функций.

Функция Ферми (экспоненциальная сигмоида):

$$f(s) = \frac{1}{1 + e^{-2as}},$$

где s – выход сумматора нейрона; a – некоторый параметр.

Рациональная сигмоида:

$$f(s) = \frac{1}{|s| + a}.$$

Гиперболический тангенс:

$$f(s) = \operatorname{th}\left(\frac{s}{a}\right) = \frac{e^{\frac{s}{a}} - e^{-\frac{s}{a}}}{e^{\frac{s}{a}} + e^{-\frac{s}{a}}}.$$

Перечисленные функции относятся к однопараметрическим. Значение функции зависит от аргумента и одного параметра. Используются также и многопараметрические передаточные функции, например

$$f(s) = p_1 \frac{s}{|s| + p_2} + p_3.$$

Сигмоидальные функции – монотонно возрастающие и имеют отличные от нуля производные на всей области определения. Эти характеристики обеспечивают правильное функционирование и обучение сети.

Наиболее эффективная передаточная функция – рациональная сигмоида. Для нахождения гиперболического тангенса требуются большие вычислительные затраты.

Многослойная нейронная сеть функционирует в соответствии с формулами

$$s_{i_q} = \sum_{i_{q-1}}^{N_{q-1}} w_{i_q i_{q-1}} w_{i_{q-1}} - b_{i_q}, \quad i_q = 1, \dots, N_q, \quad m = 1, \dots, L,$$

$$y_{i_q} = f(s_{i_q}), \quad i_q = 1, \dots, N_q, \quad m = 1, \dots, L,$$

где s – выход сумматора; i – номер нейрона; N – число нейронов в слое; w – вес связи; b – смещение; m – номер слоя; L – число слоев; y – выход нейрона; f – функция активации.

Обучение сети разбивается на следующие шаги.

ШАГ 1. Инициализация сети. Весовым коэффициентам и смещениям сети присваиваются малые случайные значения из установленных диапазонов.

ШАГ 2. Определение элемента обучающей выборки: <Текущий вход>, <Требуемый выход>. Текущие входы (x_1, \dots, x_N) должны различаться для всех элементов обучающей выборки. При использовании многослойного персептрона в качестве классификатора требуемый выходной сигнал (d_1, \dots, d_N) состоит из нулей за исключением

одного единичного элемента, соответствующего классу, к которому принадлежит текущий входной сигнал.

ШАГ 3. Вычисление текущего выходного сигнала. Последний определяется в соответствии с традиционной схемой функционирования многослойной нейронной сети.

ШАГ 4. Настройка синаптических весов. Для настройки весовых коэффициентов используется рекурсивный алгоритм, который сначала применяется к выходным нейронам сети, а затем проходит сеть в обратном направлении до первого слоя. Синаптические веса настраиваются в соответствии с формулой

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x_i,$$

где $w_{ij}(t)$ – вес от нейрона i или от элемента входного сигнала i к нейрону j в момент времени t ; η – коэффициент скорости обучения; δ_j – значение ошибки для нейрона j ; x_i – выход нейрона i или i -й элемент входного сигнала.

Если нейрон с номером j принадлежит последнему слою, то:

$$\delta_j = y_j(1 - y_j)(d_j - y_j),$$

где d_j, y_j – соответственно требуемый и текущий выход j -го нейрона.

Если j -й нейрон принадлежит одному из слоев с первого по предпоследний, то

$$\delta_j = x_j(1 - x_j) \sum_k \delta_k y_{jk},$$

где j -й нейрон принадлежит предыдущему слою, а индекс k пробегает все нейроны последующего слоя.

Смещения нейронов b настраиваются аналогичным образом.

Области применения сети: распознавание образов, классификация, прогнозирование, синтез речи, контроль, адаптивное управление, построение экспертных систем.

Недостатки сети: многокритериальная задача оптимизации в методе обратного распространения рассматривается как набор однокритериальных: на каждой итерации происходят изменения значений параметров сети, улучшающие работу лишь с одним примером обучающей выборки. Такой подход существенно уменьшает скорость обучения.

Классический метод обратного распространения относится к алгоритмам с линейной сходимостью. Для увеличения скорости сходи-

мости необходимо использовать матрицы вторых производных функции ошибки.

Преимущества сети: обратное распространение – первый эффективный алгоритм обучения многослойных нейронных сетей, один из самых популярных алгоритмов обучения, с помощью которого решены и решаются многочисленные практические задачи.

Модификации сети связаны с использованием различных функций ошибки, различных процедур определения направления и величины шага. Функции ошибки следующие:

- интегральные функции ошибки по всей совокупности обучающих примеров;
- функции ошибки целых и дробных степеней; процедуры определения величины шага на каждой итерации;
- дихотомия;
- инерционные соотношения, например

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x_i + \alpha(w_{ij}(t) - w_{ij}(t-1)),$$

где α – некоторое положительное число меньше единицы;

- отжиг.

Процедуры определения направления шага:

- с использованием матрицы производных второго порядка (метод Ньютона и др.);
- с использованием направлений на нескольких шагах (партан-метод и др.).

6.5. Сеть радиального основания (Radial Basis Function Network)

Под парадигмой RBFN (Radial Basis Function Network) понимается архитектура, предложенная J. Moody и C. Darken в 1989 г. К классу RBFN относят также вероятностные и регрессионные нейронные сети.

В общем случае это нейронная сеть, которая содержит слой скрытых нейронов с радиально симметричной активационной функцией, каждый из которых предназначен для хранения отдельного эталонного вектора (в виде вектора весов) [14; 17]. Для построения RBFN необходимо выполнение следующих условий.

Во-первых, наличие эталонов, представленных в виде весовых векторов нейронов скрытого слоя.

Во-вторых, наличие способа измерения расстояния входного вектора от эталона. Обычно это стандартное евклидово расстояние.

В-третьих, специальная функция активации нейронов скрытого слоя, задающая выбранный способ измерения расстояния. Обычно используется функция Гаусса, существенно усиливающая малую разницу между входным и эталонным векторами.

Другими словами, выходной сигнал эталонного нейрона скрытого слоя y_i – это функция (гауссиан) только от расстояния ρ_i между входным и эталонным векторами:

$$\rho_i = \sqrt{\sum_{j=1}^N (x_j - c_{ij})^2}, \quad y_i = \exp\left[-\left(\frac{\rho_i - R}{\sigma_i}\right)^2\right], \quad i = 1, \dots, L,$$

где x_j – входной вектор; $C_i = (c_{i1}, \dots, c_{iN})$ – весовой вектор i -го эталонного нейрона скрытого слоя; R, σ_i – параметры активационной функции; L – число эталонов.

Обучение слоя образцов-нейронов сети подразумевает предварительное проведение кластеризации для нахождения эталонных векторов и определенных эвристик для вычисления значений σ_i параметров активационной функции.

Нейроны скрытого слоя соединены по полносвязной схеме с нейронами выходного слоя, которые осуществляют взвешенное суммирование. Для нахождения значения весов от нейронов скрытого слоя к выходному слою используется линейная регрессия.

В общем случае активационные функции нейронов скрытого слоя могут отражать законы распределения случайных величин (вероятностные нейронные сети) либо характеризовать различные аналитические зависимости между переменными (регрессионные нейронные сети).

Поверхность отклика радиального элемента представляет собой гауссову функцию (колоколообразной формы) с вершиной в центре и понижением к краям.

Области применения сети: распознавание образов, классификация.

Недостатки сети: заранее должно быть известно число эталонов, а также эвристики для построения активационных функций нейронов скрытого слоя.

Преимущества сети: отсутствие этапа обучения в принятом смысле этого слова.

6.6. Сравнение сетей RBFN с MLP

Элемент многослойного персептрона полностью задается значениями своих весов и порогов, которые в совокупности определяют уравнение разделяющей прямой и скорость изменения функции при отходе от этой линии. До действия сигмоидной функции активации уровень активации такого элемента обусловлен гиперплоскостью, поэтому в системе Neural Networks такой элемент называется линейным (хотя функция активации, как правило, нелинейна).

В отличие от линейного элемента радиальный элемент задается своим центром и радиусом. Положение точки в N -мерном пространстве определяется N числовыми параметрами, т. е. их ровно столько же, сколько весов у линейного элемента, и поэтому координаты центра радиального элемента в пакете Neural Networks хранятся как веса, а его радиус (отклонение) – как порог.

Следует четко понимать, что веса и пороги радиального элемента принципиально отличаются от весов и порогов линейного элемента, и если забыть об этом, термин может ввести в заблуждение. Радиальные веса на самом деле представляют точку, а радиальный порог – отклонение.

Сеть типа радиальной базисной функции (RBFN) имеет промежуточный слой из радиальных элементов, каждый из которых воспроизводит гауссову поверхность отклика. Поскольку эти функции нелинейные, для моделирования произвольной функции нет необходимости брать более одного промежуточного слоя. Для моделирования любой функции необходимо лишь взять достаточное число радиальных элементов. Остается решить вопрос, как следует скомбинировать выходы скрытых радиальных элементов, чтобы получить из них выход сети. Оказывается, достаточно взять их линейную комбинацию (т. е. взвешенную сумму гауссовых функций). Сеть RBFN имеет выходной слой, состоящий из элементов с линейными функциями активации.

Сети RBFN имеют ряд преимуществ перед сетями MLP. Во-первых, как уже сказано, они моделируют произвольную нелинейную функцию с помощью всего одного промежуточного слоя и тем самым избавляют от необходимости решать вопрос о числе слоев. Во-вторых, параметры линейной комбинации в выходном слое можно полностью оптимизировать с помощью хорошо известных методов

линейного моделирования, которые работают быстро и не испытывают трудностей с локальными минимумами, так мешающими при обучении MLP, поэтому сеть RBFN обучается очень быстро (на порядок быстрее MLP).

Однако до того как применять линейную оптимизацию в выходном слое сети RBFN, необходимо определить число радиальных элементов, положение их центров и величины отклонений. Соответствующие алгоритмы, хотя и работают быстрее алгоритмов обучения MLP, в меньшей степени пригодны для отыскания субоптимальных решений. В качестве компенсации «Автоматический конструктор» сети пакета Neural Networks сможет выполнить за пользователя все необходимые действия по экспериментированию с сетью.

Другие отличия работы RBFN от MLP связаны с различным представлением пространства модели: «групповым» в RBFN и «плоскостным» в MLP.

Опыт показывает, что для правильного моделирования типичной функции сеть RBFN с ее более эксцентричной поверхностью отклика требует несколько большего числа элементов. Модель, основанная на RBFN, работает медленнее, ей нужно больше памяти, чем модели MLP (однако сеть RBFN гораздо быстрее обучается, а в некоторых случаях это важнее).

С «групповым» подходом связано и неумение сетей RBFN экстраполировать свои выводы за область известных данных. При удалении от обучающего множества значение функции отклика быстро спадает до нуля. Напротив, сеть MLP выдает более определенные решения при обработке сильно отклоняющихся данных. Достоинство это или недостаток – зависит от конкретной задачи, однако в целом склонность MLP к некритическому экстраполированию результата считается его слабостью. Экстраполяция на данные, лежащие далеко от обучающего множества, как правило, опасна и необоснованна.

Сети RBFN более чувствительны к «проклятию размерности» и испытывают значительные трудности, когда число входов велико.

Как уже говорилось, обучение RBFN-сети происходит в несколько этапов. Сначала определяются центры и отклонения для радиальных элементов, после этого оптимизируются параметры линейного выходного слоя. Расположение центров должно соответствовать

кластерам, реально присутствующим в исходных данных. Рассмотрим два наиболее часто используемых метода.

Выборка из выборки. В качестве центров радиальных элементов берут несколько случайно выбранных точек обучающего множества. В силу случайности выбора они представляют распределение обучающих данных в статистическом смысле. Однако, если число радиальных элементов невелико, такое представление может быть неудовлетворительным.

Алгоритм K -средних. Этот алгоритм стремится выбрать оптимальное множество точек, являющихся центроидами кластеров в обучающих данных. При K радиальных элементах их центры располагают таким образом:

- чтобы каждая обучающая точка относилась к одному центру кластера и лежала к нему ближе, чем к любому другому центру;
- каждый центр кластера был центроидом множества обучающих точек, относящихся к этому кластеру.

После того как определено расположение центров, нужно найти отклонения. Величина отклонения (ее также называют сглаживающим фактором) определяет, насколько «острой» будет гауссова функция. Если эти функции выбраны слишком «острыми», сеть не будет интерполировать данные между известными точками и потеряет способность к обобщению. Если же гауссовы функции взяты чересчур широкими, сеть не будет воспринимать мелкие детали. Это еще одна форма проявления дилеммы пере- и недообучения. Как правило, отклонения выбирают таким образом, чтобы колпак каждой гауссовой функции захватывал несколько соседних центров. Для этого имеется несколько методов.

Явный. Отклонения задаются пользователем.

Изотропный. Отклонение задают одинаковым для всех элементов и определяют эвристически с учетом количества радиальных элементов и объема покрываемого пространства.

K ближайших соседей. Отклонение каждого элемента устанавливается (индивидуально) равным среднему расстоянию до его K ближайших соседей. Тем самым отклонения будут меньше в тех частях пространства, где точки расположены густо. Здесь будут хорошо учитываться детали. А там, где точек мало, отклонения будут большими (и будет проводиться интерполяция).

После того как выбраны центры и отклонения, параметры выходного слоя оптимизируются с помощью стандартного метода линейной оптимизации – алгоритма псевдообратных матриц (сингулярного разложения).

Могут быть построены различные гибридные разновидности радиальных базисных функций. Например, выходной слой может иметь нелинейные функции активации, и тогда для его обучения используется какой-либо из алгоритмов обучения многослойных персептронов, например метод обратного распространения. Можно также обучать радиальный (скрытый) слой с помощью алгоритма обучения сети Кохонена. Это еще один способ разместить центры так, чтобы они отражали расположение данных.

Контрольные вопросы

1. Расскажите об особенностях структуры, свойствах искусственного нейрона и нейронной сети.
2. Приведите варианты структуры нейронных сетей. Опишите однослойные и многослойные сети.
3. Перечислите виды активационных функций и их характеристики.
4. Назовите типы многослойных нейронных сетей и их характеристики.
5. Объясните алгоритмы обучения нейронных сетей. Как организован процесс обучения?
6. Поясните особенности сети обратного распространения. Опишите алгоритм обратного распространения и перечислите характеристики сети.
7. Перечислите особенности сети радиального основания. Назовите область ее применения.
8. Сравните характеристики сетей RBFN и MLP и области их применения.

Глава 7

СИТУАЦИОННОЕ МОДЕЛИРОВАНИЕ, ИЛИ СИТУАЦИОННОЕ УПРАВЛЕНИЕ

7.1. Основы ситуационного моделирования, или ситуационного управления

Направление предложено Д. А. Поспеловым.

История возникновения. Метод ситуационного управления возник в связи с необходимостью моделирования процессов принятия решений в системах с активным элементом (человеком). В его основе лежат три основные предпосылки.

Первая предпосылка – изучение в рамках психологии принципов и моделей принятия решений человеком в оперативных ситуациях. Известны работы советских психологов в этой области: В. Н. Пушкина, Б. Ф. Ломова, В. П. Зинченко и др. [3]. Вениамин Ноевич Пушкин сформулировал так называемую модельную теорию мышления. Он показал, что психологический механизм регулирования актов поведения человека тесно связан с построением в структурах мозга информационной модели объекта и внешнего мира, в рамках которого осуществляется процесс управления на основе восприятия человеком информации извне и уже имеющихся опыта и знаний. Основа построения модели – понятийные представления об объектах и отношениях между ними, отражающие семантику выделенной сферы деятельности человека (предметной области). Модель объекта имеет многоуровневую структуру и определяет тот информационный контекст, на фоне которого протекают процессы управления. Чем богаче такая информационная модель объекта и выше способности манипулирования знаниями, тем выше качество принимаемых решений, тем многообразнее поведение человека. Вениамин Ноевич Пушкин впервые выделил три важные особенности процесса принятия решений [14]: наличие возможности классификации ситуаций в соответствии с типовыми решениями по управлению; принципиальная открытость больших систем; существенная ограниченность языка описания пространства состояний и решений объекта управления.

Второй предпосылкой метода ситуационного управления стали представления, полученные в исследованиях по семиотике (науке о

знаковых системах). Это работы Ю. А. Шрейдера, Ю. Д. Апресяна. В них была определена трехаспектная структура знака в любой знаковой системе: имя знака, отражающее его синтаксический аспект; содержание знака, выражающее его семантический аспект; назначение знака, определяющее его прагматический аспект (треугольник Фреге). В прикладной семиотике знаки, вариантами которых являются слова, предложения, тексты, стали рассматриваться как системы, замещающие реальные объекты, процессы, события внешнего мира [3]. Совокупности знаков с отношениями между ними, таким образом, стали моделирующими псевдофизическими аналогами реальных систем функционирования и управления. Именно поэтому ситуационное управление называли еще и семиотическим моделированием, поскольку знаковый язык достаточен для описания и процессов функционирования объекта с требуемой степенью приближения.

Третья предпосылка связана с разработками в области информационно-поисковых систем и попытками создания формального языка описания и представления технических наук с целью автоматизации работ по реферированию научных публикаций и организации процессов поиска, хранения и представления информации. В рамках этих исследований Э. Ф. Скороходько был разработан и исследован язык, получивший затем название языка RX-кодов [3]. Свою реализацию этот язык нашел в информационно-поисковой системе БИТ, которая успешно и довольно долго эксплуатировалась в Институте кибернетики АН УССР.

На основе модельной теории мышления В. Н. Пушкина, языка RX-кодов Э. Ф. Скороходько и семиотики Д. А. Пospelов, а затем Ю. И. Клыков в 1965 г. сформулировали новую кибернетическую концепцию управления большими системами в виде метода ситуационного управления.

Сущность метода. За основу управления принято понятие ситуации как основного объекта описания, анализа и принятия решений. Следовательно, необходимы соответствующие средства описания, классификации, обучения и трансформации ситуаций в соответствии с принимаемыми решениями.

Классификация ситуаций, исходя из анализа структуры задач управления в больших системах, обосновывалась существованием на каждом уровне управления множества ситуаций, число которых

несоизмеримо велико по сравнению с множеством возможных решений по управлению.

Задача принятия решений трактовалась как задача поиска такого разбиения множества ситуаций на классы, при котором каждому классу соответствует решение, наиболее целесообразное с позиции заданных критериев функционирования. При наличии такого разбиения поиск решения в конкретной ситуации сводился к поиску класса и соотнесения с ним решения по управлению. Однако такая постановка задачи справедлива для систем управления, в которых число потенциально возможных ситуаций (ПВС) существенно превышает (иногда на несколько порядков) число возможных решений по управлению. Этот случай соответствует контекстно-независимому способу вывода решений, когда все множество ПВС разбивается на классы таким образом, чтобы каждому классу в соответствие ставилось решение по управлению.

Случай, когда множества ситуаций и решений были либо соизмеримы по мощности, либо достаточно больше, чтобы этот факт можно было установить, был рассмотрен и разработан затем Л. С. Загадской и ее школой. За основу языка описания всего множества ситуаций были взяты идеи языков RX -кодов и синтагматических цепей. Роль множества объектов предметной области играли их знаковые эквиваленты в естественном языке, т. е. слова-имена, а в роли отношений выступали слова-имена, соответствующие реальным связям между объектами или процессами. В качестве грамматики языка ситуационного управления (ЯСУ) выступали правила порождения новых понятий и отношений, их преобразования и классификации.

Важнейшая идея метода Л. С. Загадской – формирование семиотической модели объекта путем обучения принятию решений. При этом рассматривались два режима обучения: экспертом, хорошо знающим исследуемую предметную область, либо на основе анализа множества конкретных ситуаций и решений по управлению. Очевидно, что последний случай более длительный, не гарантирует полноту описания, требует наличия статистики ситуаций и принятых в них решений, что далеко не всегда возможно, поэтому всеобщей практикой стало использование первого подхода к обучению. Тем не менее наличие в ЯСУ средств обобщения и классификации ситуаций обеспечивает принципиальную возможность создания моделей, способ-

ных к усовершенствованию функций принятия решений в изменяющихся условиях работы объекта управления. Другими словами, создается возможность «выращивания» модели объекта для заданных условий функционирования.

Развитие ситуационного моделирования. В 1973 г. Л. С. Загадская [3] разработала еще один новый тип систем ситуационного управления, рассматривавший класс систем управления, в котором мощности множеств возможных ситуаций и решений по управлению сопоставимы или неизвестны. Предлагалось все множество ситуаций разбивать на классы таким образом, чтобы каждому классу в соответствие ставилась структура типового решения. На следующем этапе решения эта структура доопределялась в процессе интерпретации и конкретизации решения и с учетом имеющихся ограничений на ресурсы. Таким образом, каждому типовому решению по управлению U_j в соответствие ставилась его структура M_j , и, следовательно, кроме множества $U = \{U_1, \dots, U_n\}$ строилось множество структур типовых решений $M = \{M_1, M_2, \dots, M_n\}$.

Затем для каждой структуры выявлялся необходимый контекст – пласт знаний, имеющий фреймовую структуру и включающий в себя правила интерпретации ситуаций в пределах данной структуры и множество процедур для их трансформации и имитации. Была разработана также логико-семиотическая модель вывода решений на иерархии структур принятия решений [3].

Очевидно, что во втором случае существенно усложняется проблема построения модели предметной области (МПО). Разработка МПО до сих пор – это искусство, которое требует высочайшей квалификации системных аналитиков. Необходимо ответить на ряд вопросов.

– Каким образом задаются границы выделенной предметной области?

– Каким образом формируется непротиворечивый язык описания всех множеств ситуаций и процессов для МПО со сложной, иерархической и распределенной структурой?

– Каким образом формируется система знаний о МПО, достаточная для достижения поставленных целей?

– Каким образом «проявляются» необходимые взаимодействия между участниками процессов управления и принятия решений, как они описываются?

– Каким образом принимаются решения в условиях неполноты, неопределенности и неоднозначности?

В результате исследования и разработки прикладных систем ситуационного управления была создана сквозная методология и технология проектирования систем ситуационного управления большими системами, включая необходимые инструментальные средства и системы на базе языков РЕФАЛ и ЛИСП [3].

Как следует из описания языка ситуационного управления и организации ситуационной модели управления, уже тогда, в 70-е гг. XX в., системы ситуационного управления (ССУ) имели все признаки современных экспертных систем (ЭС) по меньшей мере 2-го поколения, т. е. динамических ЭС. Это и наличие семиотической модели объекта управления и процессов его функционирования в виде системы правил продукционного типа, и естественно-языковой интерфейс с разработчиками и пользователями, и наличие встроенной логики времени, обеспечивающей работу ССУ в режиме реального времени и моделирования. Это и инструментальные программные средства реализации ССУ на базе языков ЛИСП и РЕФАЛ. Более того, отечественные специалисты создавали большие системы и даже внедряли их в практику в составе промышленных АСУ. Например:

– система ситуационного управления «Авиаремонт», выполненная Одесским отделением Института экономики АН УССР как часть АСУ «Авиаремонт» для ЦНИИАСУ (Рига);

– система ситуационного диспетчерского управления взлетом и посадкой самолетов, разработанная для ВНИИРА (Ленинград);

– система планирования сеансов спутниковой связи;

– ряд систем специального назначения и др.

В начале 80-х гг. XX в. появились экспертные системы. Выяснилось, что по своей сути они, вроде бы, совпадали с ССУ. И термин этот оказался более удачным, быстро вошел в моду. В результате уже к началу 90-х гг. XX в. почти все «ситуационщики» занимались ЭС. Таким образом, ситуационное управление стало основой для разработки искусственного интеллекта.

7.2. Моделирование процессов принятия решений в системах с активным элементом (человеком)

Рассмотрим методику разработки ситуационной модели на основе анализа и формализации действий ЛПР на примере управления технологическим процессом варки-выработки листового стекла [19]. Отклонения режимных переменных рассматриваются как появление ситуаций (начальных) s_H на объекте управления, а нахождение режимных переменных в пределах допусков – как конечная ситуация s_K . Управление можно представить как перевод объекта из начальной ситуации $s_H \subset S_H$ в конечную ситуацию $s_K \subset S_K$ под действием управляющих воздействий:

$$\begin{array}{cccc} u_1 & u_2 & u_3 & u_n \\ s_H \rightarrow s_1 \rightarrow s_2 \rightarrow \dots \rightarrow s_K, \end{array} \quad (7.1)$$

где S_H, S_K – множества начальных и конечных ситуаций; $s_i, i = 1, \dots, n$ – множества промежуточных ситуаций; $u_i, i = 1, \dots, n$ – управляющие воздействия по шагам управления.

В зависимости от числа последовательно выполняемых управляющих воздействий по переводу объекта из начальной ситуации в конечную управление может быть многошаговым (7.1) или одношаговым:

$$\begin{array}{c} u_1 \\ s_H \rightarrow s_K. \end{array} \quad (7.2)$$

Возможность классификации ситуаций в соответствии с типовыми решениями по управлению позволяет использовать машинную процедуру формирования понятий CLS-9 для построения ситуационной модели управления [20]. Понятия представляются в виде деревьев классификаций, в которых устанавливаются соответствия между дискретными представлениями произвольных ситуаций и выбираемыми управляющими воздействиями. Действия ЛПР формализуются на основе высказываний экспертов о принимаемых управляющих решениях в тех или иных технологических ситуациях, возникающих на объекте управления.

Формирование понятий может рассматриваться как задача распознавания образов, когда по признакам (ситуации) определяется значение управляющего воздействия, т. е. из множества критериальных классов управляющего воздействия выбирается нужный класс. Совокупность всевозможных признаков зависит от количества переменных, характеризующих признаки, и числа возможных значений

каждого признака. При трех значениях каждого признака $V_i = 3$, учитывающих знак отклонения переменной от задаваемого интервала:

«-1» – выход за пределы нормы в сторону уменьшения (меньше);

«0» – значение переменной в допустимых пределах (норма);

«+1» – выход за пределы нормы в сторону увеличения (больше),

и числе переменных ситуаций n генеральная совокупность признаков будет иметь размерность 3^n . При этом каждая ситуация будет кодироваться n -разрядным (троичным) числом.

Множество критериальных классов для каждого управляющего воздействия ограничивается тремя элементами:

$$K = \{ -, =, + \}, \quad (7.3)$$

где «-» – отрицательное приращение управляющего воздействия (уменьшить); «=» – нулевое приращение (не изменять); «+» – положительное приращение (увеличить).

Процедура CLS-9 формирует деревья по каждому управляющему воздействию. От вершин деревьев, характеризующих признак, отходят V_i ветвей. Алгоритм предусматривает использование в качестве критерия того признака, который наиболее полезен при классификации относящихся к данной вершине объектов. Для описания алгоритма формирования понятий вводится следующая система обозначений:

$A = \{A_j\}$ – множество признаков, не рассматривавшихся ни в одной из вершин, расположенных выше исследуемой;

$K = \{k\}$ – множество критериальных классов;

V_i – количество возможных значений признака i ;

n_{ijk} – количество относящихся к рассматриваемой вершине объектов, которые характеризуются значением j признака i и принадлежат к классу k ;

n_{ij}^* – максимальное значение параметра $\{n_{ijk}\}$ для любого из k классов.

Процедура CLS-9 определяет все члены множества $\{n_{ijk}\}$ и для каждого признака находит значение величины

$$H_i = \sum_{j=1}^{V_i} n_{ij}^*. \quad (7.4)$$

В качестве критерия выбирается тот признак, для которого значение H_i оказывается максимальным. Затем алгоритм формирует

вершину, от которой вниз отходят ветви по числу V_i возможных значений выбранного признака i . Построенное дерево «наилучшее из возможных» в смысле классификации всех объектов, описания которых были введены в память ЭВМ к моменту формирования дерева. Классификация всех объектов оказывается правильной.

Пример. Дана выборка для формирования понятий по выбору управляющих воздействий для процесса стекловарения в ванной печи средней производительности (табл. 7.1) [19].

Таблица 7.1

Выборка для разработки ситуационной модели
управления процессом стекловарения

Ситуация	Признаки							Принимаемые решения по управлению		
	$\Theta_{гс}$	$\Theta_{см}$	$Q_{г}$	P	$L_{ш}$	$L_{п}$	M	$\mu_{г}$	$\mu_{в}$	$\mu_{ш}$
1	+1	-1	0	0	0	0	0	=	-	=
2	+1	-1	0	+1	-1	0	0	=	+	-
3	+1	-1	0	-1	-1	-1	0	=	=	+
4	+1	-1	0	1	-1	+1	0	=	-	-
5	+1	-1	-1	+1	0	0	-1	+	=	-
6	+1	-1	+1	+1	-1	0	0	=	=	-
7	-1	-1	+1	+1	-1	0	0	=	+	-
8	-1	-1	-1	+1	-1	0	-1	+	+	-
9	+1	-1	-1	-1	-1	0	-1	+	=	+
10	+1	-1	+1	+1	-1	0	-1	=	=	-
11	+1	-1	-1	0	-1	-1	-1	+	=	=
12	+1	-1	+1	+1	-1	-1	0	=	=	-
13	+1	-1	-1	-1	-1	-1	-1	+	=	+
14	-1	-1	-1	-1	-1	-1	-1	+	+	+
15	+1	-1	+1	+1	-1	+1	-1	=	-	-
16	+1	-1	-1	-1	-1	+1	-1	+	-	=

Обозначения признаков: $\Theta_{гс}$ – температура газовой среды в ванной печи; $\Theta_{см}$ – температура стекломассы в ванной печи; $Q_{г}$ – расход газа на ванную стекловаренную печь; P – давление газового пространства в ванной печи; $L_{ш}$ – граница варочной шихты в ванной печи; $L_{п}$ – граница варочной пены в ванной печи; M – число работающих машин вертикального вытягивания стекла.

Обозначения управляющих воздействий: $\mu_{г}$ – расход природного газа на горелки; $\mu_{в}$ – расход воздуха на горение; $\mu_{ш}$ – положение шиберов дымовой трубы (тяга дымовых газов).

Рассмотрим построение ситуационной модели $\mu_{\text{в}}$ – расхода воздуха на горение. Алгоритм построения модели состоит из следующих шагов.

1. Построить таблицу частот n_{ij} распределения значений $(-1, 0, 1)$ признаков по критериальным классам $\mu_{\text{в}}$ $(-, =, +)$ (табл. 7.2).

Таблица 7.2

Определение значения управляющего воздействия $\mu_{\text{в}}$
из множества критериальных классов по выборке $n = 16$

Признаки	$\Theta_{\text{Гс}}$			$\Theta_{\text{СМ}}$			$Q_{\text{Г}}$			P			$L_{\text{Ш}}$			$L_{\text{П}}$			M			
Значения	-1	0	+1	-1	0	+1	-1	0	+1	-1	0	+1	-1	0	+1	-1	0	+1	-1	0	+1	
Критериальные классы $\mu_{\text{в}}$	-	0	0	4	4	0	0	1	2	1	1	1	2	3	1	0	0	1	3	2	2	0
	=	0	0	8	8	0	0	4	1	3	3	1	4	7	1	0	4	4	0	5	3	0
	+	3	0	1	4	0	0	2	1	1	1	0	3	4	0	0	1	3	0	2	2	0
n_{ij}^*	3	0	8	8	0	0	4	2	3	3	1	4	7	1	0	4	4	3	5	3	0	
$H_i = \sum n_{ij}^*$	11			8			9			8			8			11			8			

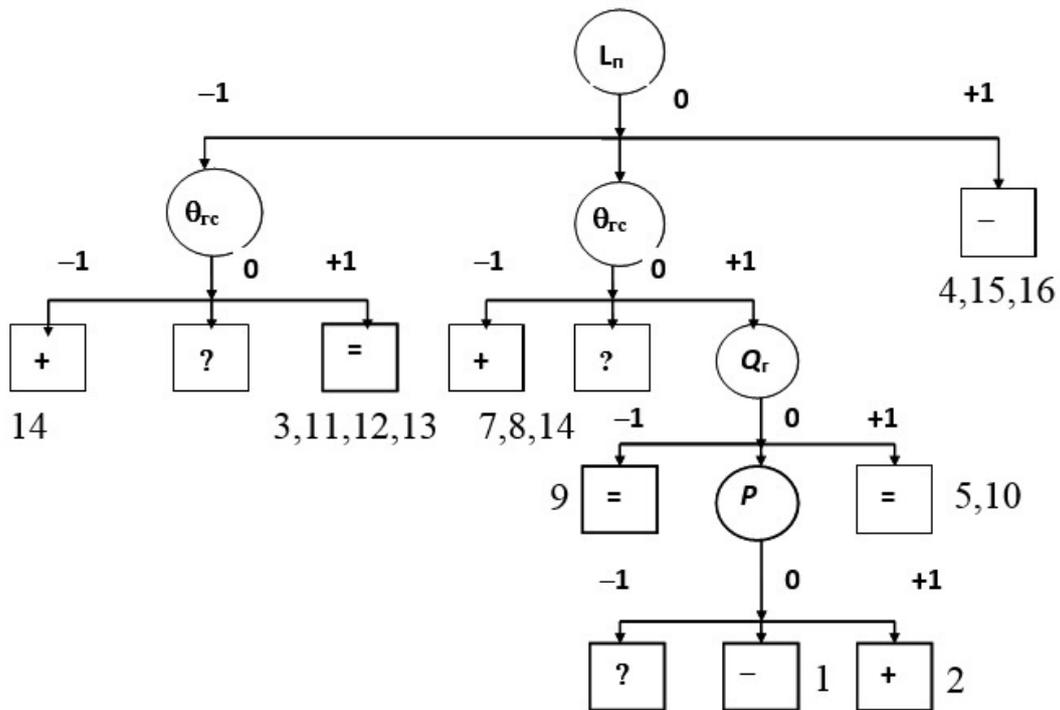
2. Определить максимальное число попаданий n_{ij}^* для каждого значения признака в один из критериальных классов n_i .

3. Для каждого признака рассчитать значение величины $H_i = \sum n_{ij}^*$.

4. В качестве критерия выбирать тот признак, для которого значение H_i оказывается максимальным. При одинаковых значениях $H_i = 11$, как в нашем случае, критерий выбирается экспертным путем либо случайным образом. Таким признаком в нашем случае выбран признак $L_{\text{П}}$.

5. Построить дерево классификаций, в вершину записать выбранный признак $L_{\text{П}}$ с наибольшим критерием. От вершины вниз отходят ветви по числу значений выбранного признака слева направо: $i = -1, 0, +1$. При $i = +1$ находится решение для трех ситуаций, записанных в строках 4, 15, 16 табл. 7.1, выбирается критериальный класс $\mu_{\text{в}} = (-)$. В «лист» дерева записывается символ $(-)$. Из табл. 7.1 удаляются строки 4, 15, 16, в которых ситуации распознаны.

При значениях признака $i = -1, i = 0$ критериальные классы не различимы, нужна дополнительная информация для получения решения (см. рисунок).



Дерево классификации управляющего воздействия $\mu_{\text{в}}$

6. Для продвижения по ветви $i = -1$ признака L_{π} необходимо образовать узел для записи второго признака. Выбирается тот признак, который не использовался выше создаваемого узла и имеет наибольшее значение H_i (см. табл. 7.2). Таким признаком является $\Theta_{гс}$. Из созданного узла выходят три ветви со значениями признака $\Theta_{гс} i = -1, 0, +1$.

7. Найти решение при значении признака $\Theta_{гс} i = -1$. Находится решение $\mu_{\text{в}} = (+)$ для одной ситуации в строке 14 табл. 7.1.

При $i = 0$ классификация не определена, в «листе» дерева записывается символ (?).

При $i = +1$ решения находят по табл. 7.1. Это непустое пересечение данных, содержащихся в столбцах $L_{\pi} = (-1)$, $\Theta_{гс} = (+1)$ и $\mu_{\text{в}} = (=)$. Классифицируют четыре ситуации $\mu_{\text{в}} = (=)$ в строках 3, 11, 12, 13 табл. 7.1.

8. Найти решение для ветви $i = 0$ признака L_{π} , т. е. вершины дерева. В качестве признака повторно выбирается $\Theta_{гс}$. При значении $i = -1$ находят решение $\mu_{\text{в}} = (+)$ для двух ситуаций в строках 7, 8 табл. 7.1.

При $i = 0$ классификация не определена, в «листе» дерева записывается символ (?), а при $i = +1$ критериальные классы не различимы, нужна дополнительная информация для нахождения решения.

9. В качестве третьего признака выбирать тот, который не использовался выше по дереву, но в то же время он имеет большее зна-

чение H_i (см. табл. 7.2). Таким признаком является Q_r , который записывается в очередную вершину (см. рисунок). Из вершины Q_r исходят три ветви со значениями $i = -1, 0, +1$.

10. Найти решение для ветви $i = -1$. По табл. 7.1 определить решение $\mu_v = (=)$ для одной ситуации 9 (см. табл. 7.1).

При значении $i = +1$ находят решение $\mu_v = (=)$ еще для двух ситуаций 5, 10 (см. табл. 7.1).

При $i = 0$ классификация не однозначная, нужна дополнительная информация для нахождения решения.

11. Для продвижения по ветви $i = 0$ признака Q_r создается узел, в который записывается один из четырех признаков со значением $H_i = 8$ (см. табл. 7.2). Выбрать признак P . Из созданного узла выходят три ветви со значениями $i = -1, 0, +1$.

12. Найти решения для значения $i = -1$ признака P . Решение отсутствует, в «листе» записывается вопросительный знак.

При значении $i = 0$ признака P по табл. 7.1 находят решение $\mu_v = (-)$ для 1-й ситуации.

При значении $i = +1$ признака P по табл. 7.1 находят решение $\mu_v = (+)$ для 2-й ситуации.

Дерево (см. рисунок) правильно классифицирует все 16 ситуаций. В «листьях» дерева записано множество критериальных классов для управляющего воздействия μ_v , которое ограничено тремя элементами $\{-, =, +\}$.

Рядом с «листьями» указаны номера классифицированных ситуаций. В трех «листьях» записана неопределенная классификация (?). Это может быть следствием относительно малого количества ситуаций в выборке, либо может отражать тот факт, что некоторые ситуации не встречались в ходе технологического процесса и такие опыты отсутствуют в выборке.

Приведенное на рисунке дерево правильно классифицирует управляющие воздействия для всех ситуаций обучающей выборки (см. табл. 7.1). Процедура CLS-9 позволяет выявить сложную логическую структуру принимаемых решений ЛПР по управлению объектом, оперируя сравнительно небольшим количеством технологических ситуаций.

Сформированные понятия могут использоваться для классификации всех ситуаций генеральной совокупности в количестве $3^7 = 2187$. При этом правильность классификации новых ситуаций не гаранти-

руется из-за отсутствия в исходной выборке примеров выбора управляющих воздействий по таким ситуациям, определяемым значениями признаков. Ошибочная классификация вновь встретившихся ситуаций требует перестройки дерева классификации управляющих воздействий. Любая возникшая в технологическом процессе ситуация, для которой обнаруживается несоответствие, требует классификации ее экспертом и внесения дополнений в исходную выборку.

Следует ожидать, что достаточен некоторый объем выборки (заранее неопределенный, но малый по сравнению с полной совокупностью ситуаций), который приведет к формированию понятий, правильно классифицирующих всю совокупность ситуаций. Такой вывод согласуется с литературными [20] и экспериментальными [19] данными.

Задача принятия решений в конкретной ситуации трактуется как поиск решения по дереву классификации ситуаций. Так, ситуация характеризуется значениями следующих признаков: температура газовой среды в ванной печи ниже нормы, $\Theta_{гс} = -1$; температура стекломассы в норме, $\Theta_{см} = 0$; расход газа в норме, $Q_{г} = 0$; граница варочной шихты укоротилась, $L_{ш} = -1$; граница варочной пены в норме, $L_{п} = 0$; число работающих машин вертикального вытягивания в норме, $M = 0$. По дереву классификации управляющего воздействия (см. рисунок) находим решение: увеличить расход воздуха $\mu_{в} = (+)$. Такой ситуации в табл. 7.1 нет, что подтверждает возможность использования сформулированного понятия (дерева классификации ситуаций) для классификации ситуаций, отсутствующих в обучающей выборке.

Контрольные вопросы

1. Назовите предпосылки, лежащие в основе ситуационного моделирования.
2. Раскройте сущность метода ситуационного моделирования.
3. Приведите формулировку задачи принятия решений.
4. Раскройте содержание машинной процедуры формирования понятий CLS-9 и применение ее для построения ситуационной модели управления.
5. Поясните особенности формирования дерева для управляющего воздействия. Опишите выбор признаков и их значений.
6. Объясните алгоритм построения дерева классификации управляющего воздействия.

Глава 8

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ

Существует два подхода к построению модели: аналитическое и имитационное моделирование. *Аналитическое моделирование* основано на косвенном описании моделируемого объекта с помощью набора математических формул. При этом предполагается использование математической модели реального объекта в форме алгебраических, дифференциальных, интегральных и других уравнений, связывающих выходные переменные с входными. Вводится система ограничений. Обычно предполагается однозначная вычислительная процедура получения точного решения уравнений.

Язык аналитического описания содержит следующие основные группы семантических элементов: критерий, данные, математические операции, ограничения. Наиболее существенно то, что аналитическая модель, вообще говоря, не является структурно подобной объекту моделирования. Под структурным подобием понимается однозначное соответствие элементов и связей модели элементам и связям моделируемого объекта. К аналитическим моделям относят модели, построенные с использованием математического программирования, корреляционного, регрессионного видов анализа.

Имитационное моделирование основано на прямом описании моделируемого объекта. Существенная характеристика таких моделей – структурное подобие объекта и модели. Это значит, что каждому существенному с точки зрения решаемой задачи элементу объекта ставится в соответствие элемент модели. При этом описываются законы функционирования каждого элемента объекта и связи между ними. Работа с имитационной моделью заключается в проведении имитационного эксперимента. Процесс, протекающий в модели в ходе эксперимента, подобен процессу в реальном объекте, поэтому исследование объекта по его имитационной модели сводится к изучению характеристик процесса, протекающего в ходе эксперимента.

8.1. Сущность метода имитационного моделирования

Определим метод имитационного моделирования в общем виде как экспериментальный метод исследования реальной системы по ее имитационной модели, который сочетает особенности экспериментального подхода и специфические условия использования вычислительной техники. В этом определении подчеркивается, что имитационное моделирование – машинный метод моделирования. Развитие информационных технологий привело к появлению этого вида компьютерного моделирования. В определении также акцентируется внимание на экспериментальной природе имитации, применяется имитационный метод исследования (осуществляется эксперимент с моделью). В имитационном моделировании важную роль играет не только проведение, но и планирование эксперимента с моделью. Однако данное определение не проясняет, что собой представляет сама имитационная модель.

Имитационная модель – это комплексное логико-математическое представление системы, реализованное в виде программы, предназначенной для решения на ЭВМ, включающее в себя модели различного типа и рассматривающее аспект функционирования динамической системы во времени. Данный класс моделей применяется при невозможности строгого аналитического решения задачи или проведения натурального эксперимента.

Имитационные модели служат для изучения поведения во времени сложной неоднородной динамической системы, относительно структуры которой существуют точные знания или детализированные гипотезы. Для каждого элемента или подсистемы моделируемой системы в памяти ЭВМ формируется блок данных, характеризующих ее текущее и предшествующие состояния, блок логических и вычислительных процедур, описывающих изменения критических параметров во времени, а также проводятся вычисления этих параметров на основе заданных значений.

Комплекс подпрограмм или относительно автономных программных агентов функционирует под управлением программы-супервизора, осуществляющей диспетчеризацию вызовов, активизирующей и приостанавливающей на время выполнение тех или иных процедур в соответствии с планом машинного эксперимента, имитируя тем самым поведение системы. В результате машинного экспери-

мента формируются массивы данных о состоянии различных параметров системы в разные моменты времени с привязкой к системным событиям, имитируемым в ходе эксперимента. При этом программасупервизор управляет процессом имитации случайных возмущающих воздействий, от которых зависит функционирование системы в целом, ее элементов и подсистем.

Имитационная модель – это инструмент исследования, посредством которого могут осуществляться и манипуляции с масштабом времени функционирования модели. Различают имитационные модели, функционирующие как в натуральном, так и в замедленном или ускоренном масштабе времени. Это крайне важно при анализе поведения систем, для наблюдения которых отсутствует возможность воспользоваться натуральным масштабом времени. К разряду таких систем могут быть отнесены экосистемы, популяции, системы, в которых протекают скоротечные физические процессы, и иное.

Ответим на вопрос: в чем же состоит сущность имитационного моделирования?

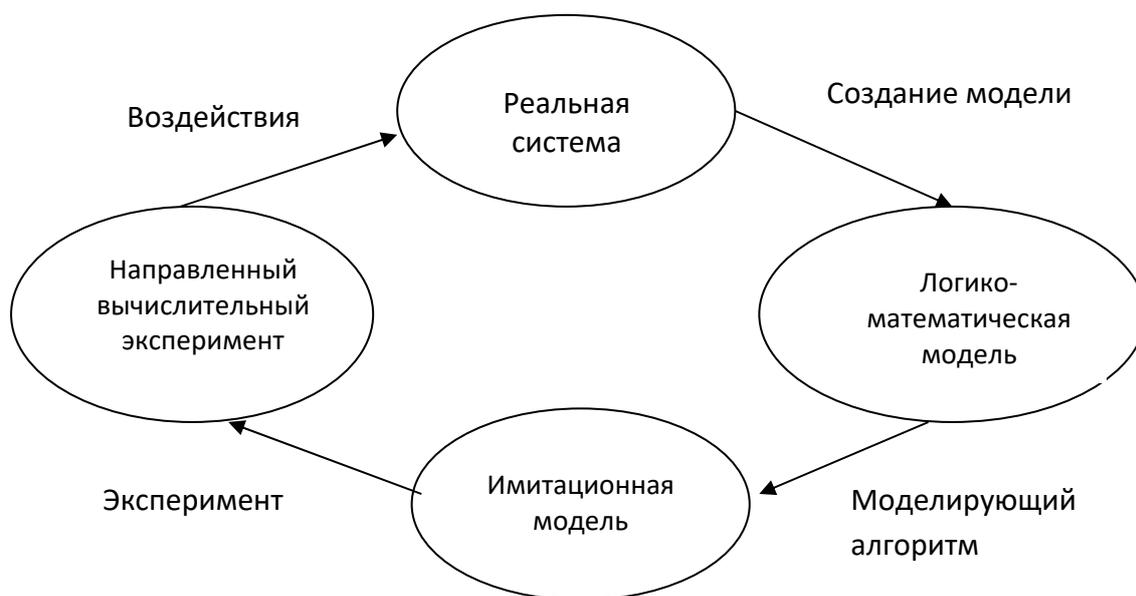


Рис. 8.1. Процесс имитационного исследования

В процессе имитационного моделирования (рис. 8.1) исследователь имеет дело с четырьмя основными элементами [30]:

- реальная система;
- логико-математическая модель моделируемого объекта;
- имитационная (машинная) модель;

– ЭВМ, на которой осуществляется имитация – направленный вычислительный эксперимент.

Исследователь изучает реальную систему, разрабатывает логико-математическую модель реальной системы. Имитационный характер исследования предполагает наличие логических или логико-математических моделей, описывающих изучаемый процесс.

Реальная система определяется как совокупность взаимодействующих элементов, функционирующих во времени. Составной характеристикой сложной системы описывает представление ее модели в виде трех множеств (A, S, T) , где A – множество элементов (в их число включается и внешняя среда); S – множество допустимых связей между элементами (структура модели); T – множество рассматриваемых моментов времени.

Первая особенность имитационного моделирования состоит в том, что имитационная модель позволяет воспроизводить моделируемые объекты:

- с сохранением их логической структуры;
- с сохранением поведенческих свойств (последовательности чередования во времени событий, происходящих в системе), т. е. динамики взаимодействий.

При имитационном моделировании структура моделируемой системы адекватно отображается в модели, а процессы ее функционирования проигрываются (имитируются) на построенной модели. Отсюда построение имитационной модели заключается в описании структуры и процессов функционирования моделируемого объекта или системы. В описании имитационной модели выделяют две составляющие:

- статическое описание системы, которое по существу является описанием ее структуры. При разработке имитационной модели необходимо применять структурный анализ моделируемых процессов;
- динамическое описание системы, или описание динамики взаимодействий ее элементов. При его составлении фактически требуется построение функциональной модели моделируемых динамических процессов.

Идея метода с точки зрения его программной реализации состоит в следующем: элементам системы ставят в соответствие некоторые программные компоненты, а состояния этих элементов описывают с

помощью переменных состояния. Элементы взаимодействуют (или обмениваются информацией), значит, может быть реализован алгоритм функционирования отдельных элементов, т. е. моделирующий алгоритм. Кроме того, элементы существуют во времени, значит, надо задать алгоритм изменения переменных состояния. Динамика в имитационных моделях реализуется с помощью механизма продвижения модельного времени.

Вторая особенность метода имитационного моделирования – возможность описания и воспроизведения взаимодействия между различными элементами системы. Таким образом, чтобы составить имитационную модель, надо:

- представить реальную систему (процесс) как совокупность взаимодействующих элементов;
- алгоритмически описать функционирование отдельных элементов;
- описать процесс взаимодействия различных элементов между собой и с внешней средой.

Ключевой момент в имитационном моделировании – выделение и описание состояний системы. Система характеризуется набором переменных состояний, каждая комбинация которых описывает конкретное состояние. Следовательно, путем изменения значений этих переменных можно имитировать переход системы из одного состояния в другое. Таким образом, имитационное моделирование – это представление динамического поведения системы посредством продвижения ее от одного состояния к другому в соответствии с определенными правилами. Эти изменения состояний могут происходить либо непрерывно, либо в дискретные моменты времени. Имитационное моделирование есть динамическое отражение изменений состояния системы с течением времени.

При имитационном моделировании логическая структура реальной системы отображается в модели, а также имитируется динамика взаимодействий подсистем в моделируемой системе. Для описания динамики моделируемых процессов в имитационном моделировании реализован механизм задания модельного времени. Этот механизм встроен в управляющие программы системы моделирования. Если бы на ЭВМ имитировалось поведение одного компонента системы, то выполнение действий в имитационной модели можно было бы осу-

шествить последовательно, по пересчету временной координаты. Чтобы обеспечить имитацию параллельных событий реальной системы, вводят некоторую глобальную переменную t (обеспечивающую синхронизацию всех событий в системе), которую называют модельным (или системным) временем.

Существуют два основных способа изменения t :

- пошаговый (применяются фиксированные интервалы изменения модельного времени);
- пособытийный (применяются переменные интервалы модельного времени, при этом величина шага определяется интервалом до следующего события).

В случае пошагового метода продвижение времени происходит с минимально возможной постоянной длиной шага (*принцип Δt*). Эти алгоритмы не очень эффективны с точки зрения использования машинного времени на их реализацию.

Способ фиксированного шага применяют в случаях:

- если закон изменения параметров от времени описывается интегро-дифференциальными уравнениями. Характерный пример – решение интегро-дифференциальных уравнений численным методом. В подобных методах шаг моделирования равен шагу интегрирования. Динамика модели представляет собой дискретное приближение реальных непрерывных процессов;

- когда события распределены равномерно и можно подобрать шаг изменения временной координаты;
- когда сложно предсказать появление определенных событий;
- когда событий очень много и они появляются группами.

В остальных случаях применяют пособытийный метод, например когда события распределены неравномерно на временной оси и появляются через значительные временные интервалы.

В рамках пособытийного метода (принцип «особых состояний») координаты времени меняются тогда, когда изменяется состояние системы. В пособытийных методах длина шага временного сдвига максимально возможная. Модельное время с текущего момента изменяется до ближайшего момента наступления следующего события. Применение пособытийного метода предпочтительнее в том случае, если частота наступления событий невелика. Тогда большая длина шага позволит ускорить ход модельного времени. На практике пособытийный метод получил наибольшее распространение.

Таким образом, последовательный характер обработки информации в ЭВМ обуславливает то, что параллельные процессы, происходящие в модели, преобразуются с помощью рассмотренного механизма в последовательные. Такой способ представления носит название квазипараллельного процесса.

Простейшая классификация основных видов имитационных моделей связана с применением двух этих способов продвижения модельного времени.

Различают имитационные модели:

- непрерывные;
- дискретные;
- непрерывно-дискретные.

В непрерывных имитационных моделях переменные изменяются непрерывно, состояние моделируемой системы рассматривается как непрерывная функция времени, и, как правило, это изменение описывается системами дифференциальных уравнений. Соответственно, продвижение модельного времени зависит от численных методов решения дифференциальных уравнений.

В дискретных имитационных моделях переменные изменяются дискретно в определенные моменты имитационного времени (наступление событий). Динамика дискретных моделей представляет собой процесс перехода от момента наступления очередного события к моменту наступления следующего события.

Поскольку в реальных системах непрерывные и дискретные процессы часто невозможно разделить, были разработаны непрерывно-дискретные модели, в которых совмещаются механизмы продвижения времени, характерные для этих двух процессов.

Имитационный характер исследования предполагает наличие логических или логико-математических моделей, описывающих изучаемый процесс (систему). Логико-математическая модель сложной системы может быть как алгоритмической, так и неалгоритмической. Для машинной реализации на основе логико-математической модели сложной системы строится моделирующий алгоритм, который описывает структуру и логику взаимодействия элементов в системе.

Имитационная модель – это программная реализация моделирующего алгоритма. Она составляется с применением средств автоматизации моделирования.

В общем виде технологическая схема имитационного моделирования представлена на рис. 8.2.

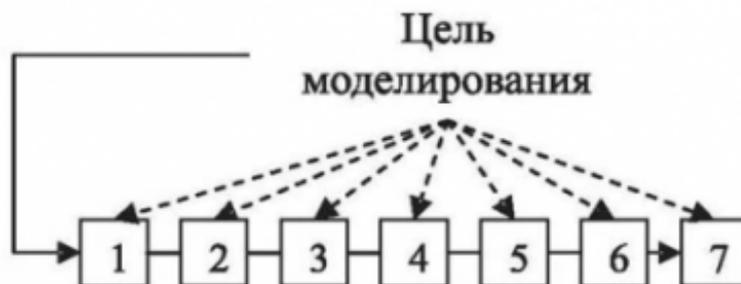


Рис. 8.2. Технологическая схема имитационного моделирования: 1 – реальная система; 2 – построение логико-математической модели; 3 – разработка моделирующего алгоритма; 4 – построение имитационной (машинной) модели; 5 – планирование и проведение имитационных экспериментов; 6 – обработка и анализ результатов; 7 – выводы о поведении реальной системы (принятие решений)

Метод имитационного моделирования позволяет решать задачи высокой сложности, обеспечивает имитацию сложных и многообразных процессов с большим количеством элементов. Отдельные функциональные зависимости в таких моделях могут описываться громоздкими математическими соотношениями. Имитационное моделирование эффективно используется в задачах исследования систем со сложной структурой с целью решения конкретных проблем. Имитационная модель содержит элементы непрерывного и дискретного действия, поэтому применяется для исследования динамических систем, когда требуется анализ узких мест, исследование динамики функционирования, когда желательно наблюдать на имитационной модели ход процесса в течение определенного времени.

Имитационное моделирование – эффективный аппарат исследования стохастических систем, когда исследуемая система может быть подвержена влиянию многочисленных случайных факторов сложной природы. Имеется возможность проводить исследование в условиях неопределенности, при неполных и неточных данных.

При имитационном моделировании широкое применение находит метод Монте-Карло [1]. В его основе лежит метод статистических

испытаний. Суть его состоит в том, что результат испытания ставится в зависимость от значения некоторой случайной величины, распределенной по заданному закону, поэтому результат каждого отдельного испытания также носит случайный характер. Проведя серию испытаний, получают множество частных значений наблюдаемой характеристики (т. е. выборку). Полученные статистические данные обрабатываются и представляются в виде соответствующих численных оценок интересующих исследователя величин (характеристик системы).

Теоретическая основа метода статистических испытаний – предельные теоремы теории вероятностей (теорема Чебышева, теорема Пуассона, теорема Бернулли). Принципиальное значение предельных теорем состоит в том, что они гарантируют высокое качество статистических оценок при весьма большом числе испытаний.

Важно отметить, что метод статистических испытаний применим для исследования как стохастических, так и детерминированных систем. Еще одна важная особенность данного метода – то, что его реализация практически невозможна без использования компьютера.

Таким образом, статистический эксперимент – основа для оценки характеристик системы при имитационном моделировании. Он представляет собой наблюдение за поведением системы в течение некоторого промежутка времени. Имитационное моделирование – важный фактор в системах поддержки принятия решений, так как позволяет исследовать большое число альтернатив (вариантов решений), проигрывать различные сценарии при любых входных данных. Главное преимущество имитационного моделирования состоит в том, что исследователь для проверки новых стратегий и принятия решений, при изучении возможных ситуаций всегда может получить ответ на вопрос: «Что будет, если...?». Имитационная модель позволяет прогнозировать, когда речь идет о проектируемой системе или исследуются процессы развития (т. е. в тех случаях, когда реальной системы еще не существует).

В имитационной модели может быть обеспечен различный, в том числе и высокий, уровень детализации моделируемых процессов. При этом модель создается поэтапно, эволюционно.

8.2. Область применения имитационных моделей

При разработке имитационной модели остаются справедливыми принципы моделирования, рассмотренные ранее. В качестве следствия из принципов моделирования необходимо отметить два важных обстоятельства:

- взаимосвязь между отдельными элементами системы, описанными в модели, а также между некоторыми величинами (параметрами) может быть представлена в виде аналитических зависимостей;

- модель можно считать реализуемой и имеющей практическую ценность только в том случае, если в ней отражены те свойства реальной системы, которые влияют на значение выбранного показателя эффективности.

Поскольку основа имитационного моделирования – метод статистических испытаний, то наибольший эффект от его применения достигается при исследовании сложных систем, на функционирование которых существенное влияние оказывают случайные факторы.

Применение имитационного моделирования целесообразно также в следующих случаях:

- если не существует законченной постановки задачи на исследование и идет процесс познания объекта моделирования;

- если характер протекающих в системе процессов не позволяет описать их в аналитической форме;

- если необходимо наблюдать за поведением системы (или отдельных ее компонентов) в течение определенного периода времени, в том числе при изменении скорости протекания процессов;

- при изучении новых ситуаций в системе либо при оценке функционирования ее в новых условиях;

- если исследуемая система – элемент более сложной системы, другие элементы которой имеют реальное воплощение;

- когда необходимо исследовать поведение системы при введении в нее новых компонентов;

- при подготовке специалистов и освоении новой техники (в качестве тренажеров).

Среди различных способов проверки гипотез, часто полученных эмпирическим путем, эффективно воспроизведение, имитация интересующих исследователя свойств и сторон анализируемого объекта

искусственным путем с учетом закономерностей и особенностей, установленных при предварительном анализе.

В процессе анализа выявляются различные параметры – признаки или характеристики объекта. Часто эти параметры (поскольку они обнаружены в исследуемом объекте) принимаются за существенные. На самом деле, однако, они могут быть и несущественными: они не мешают этому объекту быть самим собой, не меняют его сущности. В то же время существенные для него параметры могут оказаться скрытыми для исследователя и не обнаруживаться при анализе.

Таким образом, возникает задача установления существенности выбранных исходных закономерностей, а также выделения среди них основных (определяющих) и несущественных, которые либо не влияют на исследуемое явление, либо вытекают из остальных закономерностей. Решить эту задачу также можно методами имитационного моделирования.

Используемые во многих творческих процессах теории и знания, как правило, имеют интуитивный характер, не подтверждены формально, логически; интуитивно устанавливаются и области применения, способ их использования. Однако если эти интуитивные знания приводят к полезным результатам, то есть основания предполагать, что они базируются на объективных закономерностях, которые, будучи выявлены, могут лечь в основу автоматизации соответствующего творческого процесса или, иными словами, послужить основой для его моделирования. Выявление таких объективных закономерностей эвристических, на первый взгляд, неформализуемых процессов, каким, в частности, представляется и творчество, осуществляется методами эвристического программирования, являющегося разновидностью имитационного моделирования.

Научная ценность и объективная достоверность результатов имитационного моделирования тем выше, чем полнее исключены из них субъективные факторы и чем формальнее строятся такие модели. В связи с этим при решении задач проверки гипотез, выявления объективных существенных закономерностей, лежащих в основе изучаемого явления или процесса, наиболее удобно имитационное моделирование с помощью ЭВМ, называемое также математическим моделированием.

Приведенные возможные области применения имитационных моделей можно рассматривать и как перечень их достоинств, но, к

сожалению, имитационные модели имеют и ряд недостатков. Первый, и весьма существенный, заключается в том, что разработка имитационных моделей, как правило, требует больших затрат времени и сил. Кроме того, любая имитационная модель сложной системы значительно менее объективна, чем аналитическая модель, поскольку первая прежде всего отражает субъективные представления разработчика о моделируемой системе. И, наконец, результаты имитационного моделирования, как и при любом численном методе, всегда носят частный характер. Для получения обоснованных выводов необходимо проведение серии модельных экспериментов, а обработка результатов требует применения специальных статистических процедур.

Контрольные вопросы

1. Определите понятие имитационной модели и ее назначение.
2. Объясните сущность имитационного моделирования.
3. Назовите составляющие, присутствующие в описании имитационной модели.
4. Укажите, что необходимо для составления имитационной модели.
5. Перечислите основные способы изменения модельного времени.
6. Поясните особенности технологической схемы имитационного моделирования.
7. Объясните, что лежит в основе метода Монте-Карло, применяемого при имитационном моделировании.
8. Назовите область применения имитационных моделей.
9. Раскройте недостатки имитационных моделей.

ИНФОРМАЦИОННОЕ МОДЕЛИРОВАНИЕ

В современной статистической теории информации формула количества информации выражает то разнообразие сведений, которое один объект содержит о другом. Исходным является понятие условной энтропии объекта X при заданном Y – $H(X/Y)$, которое интерпретируется как количество информации, необходимое для задания объекта X в ситуации, когда объект Y уже задан.

В настоящее время в теории информации соединились три научных направления [26]:

- статистическая термодинамика как математическая теория;
- шенноновская теория информации;
- теория оптимальных статистических решений.

Сегодня происходит становление не только понятия информации, но и связанного с ним общенаучного метода исследования – теоретико-информационного. Работы У. Эшби о возможностях различного построения кибернетики – отправная точка теории управления и моделирования [27]. Он показал, что процесс связи можно интерпретировать как передачу информации, а управление – как ограничение разнообразия.

Информационная теория моделирования как составная часть теории управления определяет условия подобия целей, информационных структур, информационных потоков, а также условия подобия информационных функций преобразователей информации в узлах управления. Эффективность функционирования систем связывается с количеством информации, вносимой в контур управления, а также с затратами на информационную часть системы. Теория информации позволяет исследовать объекты сложной природы на относительно простых и наглядных математических моделях. В работе [28] показана возможность оценки взаимосвязей технологических параметров при помощи понятия энтропии, а в работе [29] доказано, что энтропия имеет нормальное распределение. Обобщение результатов этих работ дало возможность получить не только точечные, но и интервальные оценки как самой энтропии, так и параметров, определяемых на ее основе.

Согласно К. Шеннону, энтропией называется величина, рассчитываемая по формуле

$$H = -\sum_{i=1}^k p_i \ln p_i,$$

где p_i – вероятность i -го состояния дискретной случайной величины X ; k – число состояний величины X .

В приведенной формуле вероятности заменяют их оценками:

$$p_i = f_i/n,$$

где f_i – частоты наблюдения случайной величины X в i -м состоянии; n – объем выборки.

$$\hat{H} = -\sum_{i=1}^k \hat{p}_i \ln \hat{p}_i.$$

Оценка \hat{H} – состоятельная, смещенная, асимптотически нормальная оценка энтропии $H(X)$ при фиксированных p_i и $n \rightarrow \infty$.

Математическое ожидание и дисперсия $H(X)$

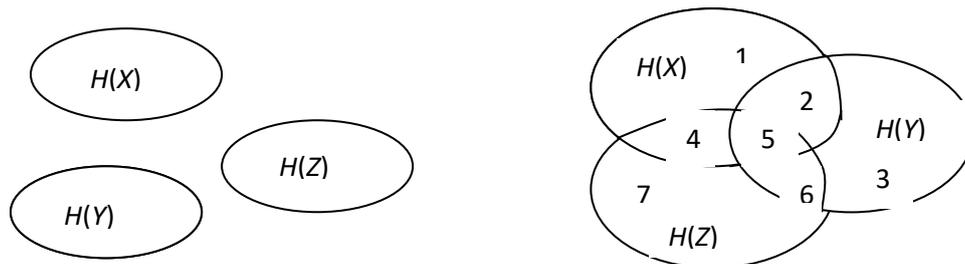
$$\begin{aligned} M(\hat{H}) &= H - (k-1)/2n + 0(1/n^2), \\ D(\hat{H}) &= (1/n) \left[\sum_{i=1}^k p_i \ln^2 p_i - H^2 \right] + 0(1/n^2), \end{aligned}$$

где $0(1/n^2)$ – величина порядка n^{-a} ($a > 0$).

9.1. Моделирование технологических операций

Информационный анализ параметров позволяет провести количественную оценку зависимости результатов выполненной операции (качество изделия) от влияющих факторов, например режима обработки, материала, состояния оборудования и другое, выявить параметр, обладающий наибольшей информативностью.

Допустим, что в результате обработки заготовки на отдельной операции сформировались три параметра: X , Y , Z . Рассмотрим возможные случаи взаимосвязи между этими параметрами (см. рисунок) [26].



Диаграммы информационной связи между параметрами

На рисунке слева отражена полная информационная независимость параметров. На рисунке справа характеризуется информационная связь между всеми параметрами. Сумма секторов 1, 2, 4, 5 отражает энтропию параметра X :

$$S(1, 2, 5, 4) = H(X);$$

$$S(2, 5, 6, 3) = H(Y);$$

$$S(4, 5, 6, 7) = H(Z).$$

$S(2, 5) = I(X \rightarrow Y) = I(Y \rightarrow X)$ – количество взаимной информации между параметрами X и Y .

Аналогично можно показать $S(4, 5)$, $S(5, 6)$.

$S(1, 2, 3, 6, 5, 4) = H(X, Y)$ – совместное количество информации о системе X и Y .

$S(2, 5, 6) = H(XZ \rightarrow Y)$ – количество информации о параметрах X и Z , заключенное в параметре Y .

Используя принятые обозначения, можно вывести следующие зависимости количества взаимной информации:

$$\begin{aligned} I(X \rightarrow Y) &= S(1,2,5,4) + S(2,5,6,3) - S(1,2,3,6,5,4) = H(X) + H(Y) - H(X, Y); \\ I(X \rightarrow Z) &= H(X) + H(Z) - H(X, Z); \\ I(Y \rightarrow Z) &= H(Y) + H(Z) - H(Y, Z); \\ I(XY \rightarrow Z) &= H(X, Y) + H(Z) - H(X, Y, Z); \\ I(XZ \rightarrow Y) &= H(X, Z) + H(Y) - H(X, Y, Z); \\ I(YZ \rightarrow X) &= H(Y, Z) + H(X) - H(X, Y, Z). \end{aligned} \tag{9.1}$$

Информационный анализ трехпараметрического случая заключается в последовательном определении количества информации с помощью вышезаписанной системы уравнений.

Для количественной оценки зависимостей между параметрами рассчитывают коэффициенты информационной связи:

$$\begin{aligned}
 R_I(X \rightarrow Y) &= I(X \rightarrow Y)/H(Y); \\
 R_I(X \rightarrow Z) &= I(X \rightarrow Z)/H(Z); \\
 R_I(Y \rightarrow Z) &= I(Y \rightarrow Z)/H(Z); \\
 R_I(XY \rightarrow Z) &= I(XY \rightarrow Z)/H(Z); \\
 R_I(XZ \rightarrow Y) &= I(XZ \rightarrow Y)/H(Y); \\
 R_I(YZ \rightarrow X) &= I(YZ \rightarrow X)/H(X).
 \end{aligned}
 \tag{9.2}$$

Обобщают результаты на n параметров, зависимость между ними вычисляют по формуле

$$I(X_1, X_2, X_3, \dots, X_{n-1} \rightarrow X_n) = H(X_1, X_2, X_3, \dots, X_{n-1}) + H(X_n) - H(X_1, X_2, X_3, \dots, X_n).$$

Таким образом, математическое моделирование технологических операций заключается в составлении и решении систем уравнений (9.1), (9.2).

9.2. Анализ моделей. Значимость оценок и доверительные интервалы

Информация и коэффициенты информации вычисляются с использованием эмпирических данных и являются случайными величинами. Оценка информации $I(X \rightarrow Y)$ с точностью до постоянного множителя имеет χ^2 -распределение [26]:

$$2n \hat{I} = \chi^2 m,$$

где n – число опытов; $m = (k_1 - 1)(k_2 - 1)$ – число степеней свободы; k_1, k_2 – количество интервалов разбиения входного и выходного параметров.

Формула позволяет определить значимость информационного взаимодействия двух параметров. Информация, передаваемая от одного параметра к другому, считается значимой при выполнении соотношения

$$2n \hat{I} \geq \chi^2_{m, \alpha},$$

где $\chi^2_{m, \alpha}$ – квантиль χ^2 -распределения с уровнем значимости α .

Значимость коэффициента информационной связи R_I определяется значимостью информации. При $m > 25$ распределение Пирсона можно заменить распределением Гаусса с дисперсией $\sigma_I^2 = 2m$, что позволяет определить доверительные интервалы для информации [26]:

$$\hat{I} - \frac{t_\alpha \sqrt{2m}}{2n} \leq I \leq \hat{I} + \frac{t_\alpha \sqrt{2m}}{2n},$$

где t_α – α -квантиль нормального распределения.

Доверительный интервал для коэффициента информационной связи определяют по формуле [26]

$$\hat{R}_I - \frac{t_\alpha \sqrt{2m}}{2nH(Y)} \leq R_I \leq \hat{R}_I + \frac{t_\alpha \sqrt{2m}}{2nH(Y)}.$$

Минимальный объем выборки, необходимый для построения информационной модели с заданной точностью ΔI , определяют по формуле [26]

$$n_{\min} = \frac{t_\alpha}{\Delta I} \sqrt{\frac{m}{2}}.$$

Идентичность информационной модели реальному объекту оценивается по значению коэффициента информационной связи R_I , который может использоваться в качестве меры определенности процесса аналогично коэффициенту детерминации в регрессионной модели [26].

Пример. Исследование парных причинно-следственных взаимодействий часто встречается при анализе систем, в частности технологических операций. Рассмотрим методику построения информационной модели технологической операции [4].

Для анализа используется выборка, состоящая из 100 измерений и отражающая зависимости показателя Y от причины X . Для построения таблицы частот проводят дискретизацию данных. Количество интервалов разбиения рассчитывают по формуле $k = 3,748 + 0,012n$ [26].

Выбираемая длина интервалов должна в два раза и более превышать абсолютную погрешность измерений для исключения влияния погрешностей измерения на результаты анализа. Для рассматриваемого примера таблица частот приведена ниже.

Подсчет частот

Y	X					f _y
	100 – 120	120 – 140	140 – 160	160 – 180	180 – 200	
65 – 85	13	3	0	0	0	16
85 – 105	14	23	2	0	0	39
105 – 125	0	11	23	1	0	35
125 – 145	0	0	3	6	0	9
145 – 165	0	0	0	0	1	1
f _x	27	37	28	7	1	100

Рассчитаем оценку энтропии X:

$$\begin{aligned} \hat{H}(X) &= -\sum_{i=1}^k \hat{p}_i \ln \hat{p}_i = -\sum_i \frac{f_i}{n} \ln \frac{f_i}{n} + \sum_i \frac{f_i}{n} \ln n = \\ &= \frac{-(27 \ln 27 + 37 \ln 37 + 28 \ln 28 + 7 \ln 7 + 1 \ln 1) + 100 \ln 100}{100} = 1,21. \end{aligned}$$

Аналогично рассчитаем оценку энтропии Y:

$$\begin{aligned} \hat{H}(Y) &= \frac{-(16 \ln 16 + 39 \ln 39 + 35 \ln 35 + 9 \ln 9 + 1 \ln 1) + 100 \ln 100}{100} = \\ &= 1,2906. \end{aligned}$$

Оценку совместной энтропии H(XY) рассчитаем по формуле

$$\begin{aligned} \hat{H}(XY) &= -\sum_i \sum_j \hat{p}_{ij} \ln \hat{p}_{ij} = -\sum_i \sum_j \frac{f_{ij}}{n} \ln \frac{f_{ij}}{n} + (n \ln n) / n = \\ &= \frac{-(13 \ln 13 + 14 \ln 14 + 2 \cdot 3 \ln 3 + 2 \cdot 23 \ln 23 + 11 \ln 11 + 2 \ln 2 + \\ &+ 2 \cdot 1 \ln 1 + 6 \cdot \ln 6 + 100 \ln 100)}{100} = 2,0089. \end{aligned}$$

Параметры искомой информационной модели

$$I(X \rightarrow Y) = H(X) + H(Y) - H(X, Y) = 1,21 + 1,2906 - 2,0089 = 0,4917.$$

Коэффициент информационной связи

$$R_I(X \rightarrow Y) = I(X \rightarrow Y) / H(Y) = 0,4917 / 1,2906 = 0,381.$$

Проведем анализ полученной модели, для чего рассчитаем доверительные границы для информации $I(X \rightarrow Y)$. Оценка информации значима при выполнении соотношения

$$2n\hat{I} \geq \chi^2_{m, \alpha},$$

$$2n\hat{I} = 2 \cdot 100 \cdot 0,4917 = 98,34.$$

Для уровня значимости 0,05, числа степеней свободы $m = (k_1 - 1) \times (k_2 - 1) = 4 \cdot 4 = 16$ находим табличное значение $\chi^2_{m, \alpha} = 26,296$.

Следовательно, информация значима, так как отличается от нуля.

Определим доверительные границы для $I(X \rightarrow Y)$:

$$\hat{I} - \frac{\chi^2_{m, \alpha/2}}{2n} \leq I(X \rightarrow Y) \leq \hat{I} + \frac{\chi^2_{m, 1-\alpha/2}}{2n}.$$

По таблице находим значения

$$\chi^2_{16; 0,05/2} = 28,845; \quad \chi^2_{16; 1-0,05/2} = 6,908.$$

После подстановки в формулу получаем доверительные интервалы (границы) для $I(X \rightarrow Y)$:

$$0,4917 - \frac{28,845}{200} \leq I(X \rightarrow Y) \leq 0,4917 + \frac{6,908}{200},$$

$$0,3474 \leq I(X \rightarrow Y) \leq 0,5262.$$

Поскольку коэффициент информационной связи – мера определенности процесса, то по результатам моделирования можно сделать вывод, что на зависимую переменную Y на 38,1 % оказывает влияние переменная X . Влияние неучтенных факторов значительное и составляет 61,9 %.

Контрольные вопросы

1. Назовите особенности информационной теории моделирования и решаемые ею задачи.
2. Дайте определение энтропии по К. Шеннону.
3. Приведите оценки энтропии, математическое ожидание и дисперсии, вычисляемые по эмпирическим данным.

4. Объясните особенности моделирования технологических операций, информационную связь между параметрами.
5. Раскройте содержание информационного анализа для трехпараметрического случая.
6. Дайте определение коэффициента информационной связи. Количественная оценка зависимостей между параметрами.
7. Приведите оценку значимости информационного взаимодействия двух параметров.
8. Определите величину доверительного интервала для коэффициента информационной связи между двумя параметрами.
9. Обоснуйте идентичность информационной модели реальному объекту.
10. Определите минимальный объем выборки, необходимой для построения информационной модели с заданной точностью ΔI .
11. Рассчитайте количество интервалов разбиения при дискретизации экспериментальных данных.

ЗАКЛЮЧЕНИЕ

Очевидно, что невозможно полно описать в рамках одного учебного пособия математические основы моделирования информационных процессов и систем. В издании освещены основные, наиболее широко используемые математические методы моделирования информационных процессов и систем.

Магистранты знакомятся с моделированием как методом познания, основами теории моделирования, математическими схемами моделирования систем и информационных процессов. Рассмотрены аналитические и имитационные методы моделирования, математические основы моделирования с использованием нечетких множеств и нейронных сетей, ситуационное, имитационное и информационное моделирование процессов и систем. Теоретический материал подкреплен примерами, которые дают возможность разобраться в особенностях использования математических методов при моделировании информационных процессов и систем.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Советов, Б. Я.* Моделирование систем : учеб. для вузов / Б. Я. Советов, С. Ф. Яковлев. – 3-е изд., перераб. и доп. – М. : Высш. шк., 2001. – 343 с. – ISBN 5-06-003860-2.

2. *Осовский, С.* Нейронные сети для обработки информации / С. Осовский / пер. с пол. И. Д. Рудинского. – М. : Финансы и статистика, 2004. – 344 с. – ISBN 5-279-02567-4.

3. Теория систем и системный анализ в управлении организациями : справочник / под ред. В. Н. Волковой, А. А. Емельянова. – М. : Финансы и статистика, 2006. – 848 с. – ISBN 5-279-02933-5.

4. Информационные технологии в управлении качеством автомобильного стекла : учеб. пособие / Р. И. Макаров [и др.] ; Владим. гос. ун-т. – Владимир : Изд-во Владим. гос. ун-та, 2010. – 276 с. – ISBN 978-5-9984-0038-4.

5. *Бильфельд, Н. В.* Программирование в Matlab : учеб.-метод. пособие / Н. В. Бильфельд, Е. В. Иванова. – Пермь : Изд-во Перм. нац. исслед. политехн. ун-та, 2011. – 235 с. – ISBN 978-5-398-00672-8.

6. *Малинин, А. С.* Исследование системы управления / А. С. Малинин, В. И. Мухин. – М. : ГУ ВШЭ, 2002. – 397 с. – ISBN 5-7598-0192-9.

7. *Коротков, Э. М.* Исследование систем управления / Э. М. Коротков. – М. : ДеКА, 2000. – 285 с. – ISBN 5-89645-014-1.

8. *Курносов, Ю. В.* Аналитика: методология, технология и организация информационно-аналитической работы / Ю. В. Курносов, П. Ю. Конотопов. – М. : Русаки, 2004. – 550 с. – ISBN 5-93347-151-8.

9. *Алексеева, М. Б.* Основы теории систем и системного анализа : учеб. пособие / М. Б. Алексеева, С. Н. Балан. – СПб. : СПбГИЭУ, 2002. – 88 с. – ISBN 5-88996-229-X.

10. *Макаров, Р. И.* Методология проектирования информационных систем : учеб. пособие / Р. И. Макаров, Е. Р. Хорошева ; Владим. гос. ун-т. – Владимир : Изд-во Владим. гос. ун-та, 2008. – 334 с. – ISBN 978-5-89368-817-7.

11. Основы теории систем и системного анализа : конспект лекций / Институт делового администрирования. – Кривой Рог, 1996. – 77 с.

12. *Орловский, С. А.* Проблемы принятия решений при нечеткой исходной информации / С. А. Орловский. – М. : Наука, 1981. – 206 с.

13. *Медведев, В. С.* Нейронные сети. MATLAB 6 / В. С. Медведев, В. Г. Потемкин ; под общ. ред. В. Г. Потемкина. – М. : ДИАЛОГ-МИФИ, 2002. – 496 с. – ISBN 5-86404-163-7.

14. *Пушкин, В. Н.* Оперативное мышление в больших системах / В. Н. Пушкин. – М. : Энергия, 1965. – 375 с.

15. *Круглов, В. В.* Искусственные нейронные сети. Теория и практика / В. В. Круглов, В. В. Борисов. – 2-е изд. – М. : Горячая линия – Телеком, 2002. – 382 с. – ISBN 5-93517-031-0.

16. *Круглов, В. В.* Нейронные сети: конфигурации, обучение, применение / В. В. Круглов, В. В. Борисов, Е. В. Харитонов. – Смоленск : Изд-во Моск. энергет. ин-та, 1998.

17. Нейронные сети. STATISTICA Neural Networks : пер. с англ. / под ред. В. П. Боровикова. – М. : Горячая линия – Телеком, 2008. – 392 с. – ISBN 978-5-9912-0015-8.

18. *Барский, А. Б.* Нейронные сети: распознавание, управление, принятие решений / А. Б. Барский. – М. : Финансы и статистика, 2004. – 176 с. – ISBN 5-279-02757-X.

19. *Макаров, Р. И.* Автоматизация производства листового стекла : учеб. пособие / Р. И. Макаров, Е. Р. Хорошева, С. А. Лукашин. – М. : Изд-во АСВ, 2002. – 192 с. – ISBN 5-93093-116-X.

20. *Хант, Э.* Моделирование процесса формирования понятий на вычислительной машине : пер. с англ. / Э. Хант, Дж. Марин, Ф. Стоун. – М. : Мир, 1970. – 301 с.

21. *Ермольев, Ю. М.* Методы стохастического программирования / Ю. М. Ермольев. – М. : Наука, 1976. – 240 с.

22. *Райбман, Н. С.* Построение моделей процессов производства / Н. С. Райбман, В. М. Чадаев. – М. : Энергия, 1975. – 376 с.

23. *Штовба, С. Д.* Введение в теорию нечетких множеств и нечеткую логику [Электронный ресурс] / С. Д. Штовба. – Режим доступа: <http://ru.b-ok.cc/book/3258436/f3e8f4> (дата обращения: 15.09.2019).

24. *Круглов, В. В.* Нечеткая логика и искусственные нейронные сети [Электронный ресурс] / В. В. Круглов, М. И. Дли, Р. Ю. Голунов. – Режим доступа: <http://b-ok2.org/book/2461354/218267> (дата обращения: 15.09.2019).

25. Описание MatLab : Fuzzy Logic ToolBox (Eng) [Электронный ресурс]. – Режим доступа: <http://matlab.ru/products/fuzzy-logic-toolbox> (дата обращения: 15.09.2019).

26. Информационные методы в управлении качеством / В. Г. Григорович [и др.]. – М. : Стандарты и качество, 2001. – 208 с. – ISBN 5-901397-03-7.

27. Эшби, У. Р. Введение в кибернетику / У. Р. Эшби. – М. : Изд-во иностр. лит., 1959. – 432 с.

28. Шеридан, Т. Б. Системы человек – машина : Модели обработки информации, управления и принятия решения человеком-оператором / Т. Б. Шеридан, У. Р. Феррелл. – М. : Машиностроение, 1980. – 400 с.

29. Башарин, Г. П. О статистической оценке энтропии независимых случайных величин / Г. П. Башарин // Теория вероятностей и ее применения. – 1959. – Т. 4, № 3. – С. 361 – 364.

30. Лычкина, Н. Н. Имитационное моделирование экономических процессов [Электронный ресурс] : учеб. пособие / Н. Н. Лычкина. – Режим доступа: <https://www.hse.ru> (дата обращения: 15.09.2019).

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
Глава 1. МОДЕЛИРОВАНИЕ КАК МЕТОД ПОЗНАНИЯ	5
1.1. Модель и ее свойства	5
1.2. Моделирование. Принцип моделируемости	8
1.3. Основные методы формализации предметной области исследований. Поэтапный синтез моделей систем и процессов ..	11
<i>Контрольные вопросы</i>	17
Глава 2. ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МОДЕЛИРОВАНИЯ СИСТЕМ	18
2.1. Принципы системного подхода к моделированию систем	18
2.2. Общая характеристика проблемы моделирования систем	22
2.3. Возможности и эффективность моделирования систем на вычислительных машинах.....	25
<i>Контрольные вопросы</i>	29
Глава 3. МАТЕМАТИЧЕСКИЕ СХЕМЫ МОДЕЛИРОВАНИЯ СИСТЕМ. ФОРМАЛИЗАЦИЯ И АЛГОРИТМИЗАЦИЯ ПРОЦЕССОВ ФУНКЦИОНИРОВАНИЯ СИСТЕМ	30
3.1. Основные подходы к построению математических моделей систем.....	30
3.2. Формализация и алгоритмизация процессов функционирования систем. Методика разработки и машинная реализация моделей систем	33
<i>Контрольные вопросы</i>	42
Глава 4. МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОЦЕССОВ ОБРАБОТКИ ИНФОРМАЦИИ	43
4.1. Моделирование информационных процессов.....	43
4.2. Характеристики процесса обработки информации... ..	48
4.3. Точность процесса обработки информации	51
4.4. Время реализации алгоритма.....	56
<i>Контрольные вопросы</i>	57

Глава 5. ВВЕДЕНИЕ В ТЕОРИЮ НЕЧЕТКИХ МНОЖЕСТВ.....	59
5.1. Общие понятия	59
5.2. Операции над нечеткими множествами	62
5.3. Нечеткие отображения и задачи принятия решений	66
5.4. Прообраз нечеткого множества при нечетком отображении.....	71
<i>Контрольные вопросы</i>	73
Глава 6. ОСНОВНЫЕ ПОЛОЖЕНИЯ ТЕОРИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ	74
6.1. Структура и свойства искусственного нейрона	74
6.2. Классификация нейронных сетей и их свойства	78
6.3. Алгоритмы обучения нейронных сетей.....	81
6.4. Обратное распространение (Neural Network with Back Propagation Training Algorithm).....	85
6.5. Сеть радиального основания (Radial Basis Function Network)	88
6.6. Сравнение сетей RBFN с MLP.....	90
<i>Контрольные вопросы</i>	93
Глава 7. СИТУАЦИОННОЕ МОДЕЛИРОВАНИЕ, ИЛИ СИТУАЦИОННОЕ УПРАВЛЕНИЕ.....	94
7.1. Основы ситуационного моделирования, или ситуационного управления	94
7.2. Моделирование процессов принятия решений в системах с активным элементом (человеком)	99
<i>Контрольные вопросы</i>	105
Глава 8. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ	106
8.1. Сущность метода имитационного моделирования.....	107
8.2. Область применения имитационных моделей	115
<i>Контрольные вопросы</i>	117
Глава 9. ИНФОРМАЦИОННОЕ МОДЕЛИРОВАНИЕ	118
9.1. Моделирование технологических операций	119
9.2. Анализ моделей. Значимость оценок и доверительные интервалы.....	121
<i>Контрольные вопросы</i>	124
ЗАКЛЮЧЕНИЕ.....	126
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	127

Учебное издание

МАКАРОВ Руслан Ильич
ХОРОШЕВА Елена Руслановна

МАТЕМАТИЧЕСКИЕ ОСНОВЫ МОДЕЛИРОВАНИЯ
ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ

Учебное пособие

Редактор Т. В. Евстюничева
Технический редактор С. Ш. Абдуллаева
Корректор Н. В. Пустовойтова
Компьютерная верстка Е. А. Кузьминой
Выпускающий редактор А. А. Амирсейидова

Подписано в печать 13.12.19.
Формат 60×84/16. Усл. печ. л. 7,67. Тираж 50 экз.

Заказ

Издательство

Владимирского государственного университета
имени Александра Григорьевича и Николая Григорьевича Столетовых.
600000, Владимир, ул. Горького, 87.