

Министерство образования и науки Украины
Винницкий государственный технический университет

Ю. И. Митюшкин

Б. И. Мокин

А. П. Ротштейн

Soft Computing:
идентификация
закономерностей
нечеткими базами знаний

УНІВЕРСУМ-Вінниця

2002

УДК 681.3
М66

Рецензенти:

доктор технічних наук, професор **П. Д. Лежнюк**
доктор технічних наук, професор **В. М. Лисогор**

Рекомендовано до видання Вченою радою Вінницького державного технічного університету Міністерства освіти і науки України (протокол № 2 від 26.09.2002)

Мітюшкін Ю. І., Мокін Б. І., Ротштейн О. П.

М66 Soft Computing: ідентифікація закономірностей нечіткими базами знань. Монографія. - Вінниця: УНІВЕРСУМ-Вінниця, 2002. - 145 с.

ISBN 966-

Монографія є розвитком методів ідентифікації складних нелінійних закономірностей на основі сучасних інтелектуальних технологій (Soft Computing), а саме - комплексного застосування нечіткої логіки, нейронних мереж та генетичних алгоритмів. Основний акцент робиться на способі отримання нечітких баз знань із експериментальних даних без участі експертів шляхом застосування генетичних алгоритмів оптимізації. Наводиться відповідний математичний апарат, методологія настроювання отриманих моделей нейронними мережами, результати комп'ютерного моделювання, а також розв'язання прикладної задачі з галузі електроенергетики - прогнозування добових графіків споживання електричної потужності.

Книга розрахована на фахівців з математичного моделювання та обчислювальних методів, а також тих галузей, де постійно виникає необхідність прийняття рішення на основі великої кількості даних - медицина, бізнес, соціологія тощо.

УДК 681.3

ISBN 966-

© Ю. Мітюшкін, Б. Мокін, О. Ротштейн, 2002

Наукове видання

**Юрій Ігоревич Мітюшкін
Борис Іванович Мокін
Олександр Петрович Ротштейн**

**SOFT COMPUTING:
ІДЕНТИФІКАЦІЯ ЗАКОНОМІРНОСТЕЙ
НЕЧІТКИМИ БАЗАМИ ЗНАНЬ**

Монографія

Оригінал-макет підготовлено авторами.

Видається в авторській редакції.

Видавництво ВДТУ «УНІВЕРСУМ-Вінниця»
Свідоцтво Держкомінформу України
серія ДК № 746 від 25.12.2001
21021, м. Вінниця, Хмельницьке шосе, 95, ВДТУ, ГНК, к. 114
Тел. (0432) 44-05-32

Підписано до друку . . . 2002. Гарнітура Times New Roman
Формат 29,7×42 1/4 . Папір офсетний
Друк різнографічний. Ум. др. арк. 8,51
Наклад 300 прим.
Зам. № 2002-

Віддруковано в комп'ютерному інформаційно-видавничому центрі
Вінницького державного технічного університету
Свідоцтво Держкомінформу України
серія ДК № 746 від 25.12.2001
21021, м. Вінниця, Хмельницьке шосе, 95

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	6
ГЛАВА 1 МЕТОДЫ ИДЕНТИФИКАЦИИ МНОГОМЕРНЫХ ЗАВИСИМОСТЕЙ.....	11
1.1. Идентификация нелинейных объектов.....	11
1.1.1. Системы управления.....	11
1.1.2. Распознавание образов.....	13
1.1.3. Нейронные сети как средства идентификации.....	14
1.1.4. Нечеткая логика.....	15
1.2. Задача извлечения знаний.....	16
1.3. Методы самоорганизации моделей	21
ГЛАВА 2 ИЗВЛЕЧЕНИЕ НЕЧЕТКИХ БАЗ ЗНАНИЙ	24
2.1. Основные понятия теории нечетких множеств.....	24
2.2. Принципы идентификации на базе нечеткой логики.	27
2.3. Нечеткие базы знаний как модели нелинейных объектов.....	29
2.3.1. Оценки входов и выходов объекта моделирования.....	30
2.3.2. Входы и выход объекта как лингвистические переменные.....	31
2.3.3. Структура нечеткой базы знаний.....	34
2.3.4. Модели функций принадлежности входных переменных.....	38
2.3.5. Постановка задачи моделирования.....	39
2.3.6. Система нечетких логических уравнений.	41
2.3.7. Решение системы нечетких логических уравнений и дефаззификация результатов.....	42
2.4. Извлечение нечетких баз знаний как задача оптимизации	46
2.4.1. Нечеткие логические уравнения с весами правил	47
2.4.2. Задача оптимизации для объекта с дискретным выходом.....	49
2.4.3. Задача оптимизации для объекта с непрерывным выходом.....	52
2.5. Генетические алгоритмы оптимизации.....	54

2.5.1. Принципы построения.....	55
2.5.2. Основные операции.....	57
2.6. Генетические алгоритмы извлечения нечетких баз знаний	58
2.6.1. Кодирование нечеткой модели.....	59
2.6.2. Скрещивание хромосом-моделей.....	60
2.6.3. Мутация генов хромосом-моделей.....	62
2.6.4. Функции соответствия хромосом.....	63
2.6.5. Выбор хромосом-родителей.....	64
2.6.6. Инициализация популяции хромосом.....	66
2.6.7. Генетический алгоритм извлечения нечеткой базы знаний.....	66
ГЛАВА 3 НАСТРОЙКА НЕЧЕТКИХ БАЗ ЗНАНИЙ.....	68
3.1. Постановка задачи.....	68
3.2. Нейронные модели многомерных зависимостей.....	69
3.2.1. Нейронная сеть как универсальный аппроксиматор.....	69
3.2.2. Обучение нейронной сети.....	74
3.2.3. Компьютерное моделирование.....	76
3.3. Нейро-нечеткие модели многомерных зависимостей.....	82
3.3.1. Лингвистический аппроксиматор	82
3.3.2. Нейро-нечеткая сеть	83
3.3.3. Обучение нейро-нечеткой сети.....	86
3.3.4. Компьютерное моделирование.....	89
ГЛАВА 4 ЭКСПЕРИМЕНТЫ С ЭТАЛОННЫМИ МОДЕЛЯМИ.....	96
4.1. Зависимости с непрерывным выходом.....	96
4.1.1. Объект «один вход - один выход».....	96
4.1.2. Объект «два входа - один выход».....	100
4.2. Зависимости с дискретным выходом.....	104
4.2.1. Объект «два входа - один выход» 1-го типа.....	104
4.2.2. Объект «два входа - один выход» 2-го типа.....	108

ГЛАВА 5 ПРИМЕНЕНИЕ НЕЧЕТКИХ БАЗ ЗНАНИЙ ДЛЯ ПРОГНОЗИРОВАНИЯ В ЭЛЕКТРОЭНЕРГЕТИКЕ	113
5.1. Общая характеристика задачи.....	113
5.2. Структура модели прогнозирования.....	116
5.2.1. Построение нечеткой базы знаний.....	119
5.2.2. Прогнозирование суточных графиков потребления электрической мощности.....	121
5.3. Алгоритм реализации процесса прогнозирования.....	122
5.3.1. Обучающая выборка.....	122
5.3.2. Предварительный анализ экспериментальных данных.	123
5.3.3. Генетический алгоритм извлечения матрицы знаний из обучающей выборки.....	127
5.4. Результаты компьютерного моделирования.....	132
ЗАКЛЮЧЕНИЕ.....	135
ЛИТЕРАТУРА.....	137

ВВЕДЕНИЕ

Идентификация многомерных объектов в кибернетике и других областях предусматривает построение с помощью средств математики соответствующих им операторов связи между функциями входных и выходных сигналов на основе доступной информации [51]. При этом процесс создания математической модели того или иного объекта состоит из двух основных этапов [78]:

1) структурная идентификация - установление переменных величин и элементов структуры модели;

2) параметрическая идентификация - этап, во время которого на основе наблюдения входных и выходных сигналов объекта определяются значения параметров и устанавливается окончательная структура модели путем устранения лишних элементов.

В основе современной теории идентификации лежит моделирование исследуемых объектов при помощи уравнений (дифференциальных, разностных и т. п.). При этом сложность того или иного объекта самым непосредственным образом влияет на качество построения его модели. Если для описания некоторых объектов применяется информация, которая не может быть выражена количественно - так называемая семантическая, т. е. смысловая, качественная информация, то классическая теория оказывается плохо приспособленной для таких случаев. Основные причины малоэффективности или же вообще неприспособленности традиционных методов моделирования к подобным ситуациям состоят в следующем [48]:

- не все входные и выходные параметры объекта могут описываться количественно;
- между рядом входных и выходных параметров невозможно установить количественные зависимости;

- существующие способы моделирования объектов приводят к таким громоздким конструкциям, что их практическое применение оказывается невозможным.

А если объект имеет, скажем, экономическую, социальную или биологическую природу, то можно также добавить следующее:

- объект эволюционирует во времени, его структура и функции изменяются;

- элементы, входящие в состав объекта, имеют активную природу; их поведение временами может противоречить здравому смыслу.

Таким образом, при моделировании объектов, которые характеризуются вышеуказанными особенностями, возникает проблема построения так называемых *логико-лингвистических моделей*, то есть моделей, в которых средства обработки информации основаны на логике, а экспериментальные данные представляются в лингвистической форме [48]. Такие модели должны основываться на *системах знаний* об исследуемом объекте, которые представляют собой концентрацию опыта специалистов (экспертов) в данной области. Для систем знаний, в свою очередь, должен быть выбран язык представления знаний, при помощи которого в модели можно было бы адекватно воссоздавать сведения о структуре объекта, и который по возможности должен быть близким к естественному языку (в простейшем варианте - быть ее определенным подмножеством) и характеризоваться достаточной формальностью и логичностью с целью построения компактной, строгой и четкой системы знаний.

Одним из таких современных формальных аппаратов для обработки экспертной естественно-языковой информации является теория лингвистической переменной, базирующаяся на нечетких множествах. В соответствии с этой теорией, модель объекта представляет собой так называемую нечеткую базу знаний в виде совокупности логических высказываний «ЕСЛИ-ТО», которые связывают лингвистические оценки входных и выходных параметров объекта, например: ЕСЛИ скорость автомобиля *очень высокая*, ТО затраты бензина

большие; ЕСЛИ температура *низкая* и давление *низкое*, ТО положение дросселя *среднее*; ЕСЛИ *высокая* температура больного держится *несколько дней*, ТО *возможно* воспаление легких [55]. Адекватность таких моделей к данным эксперимента определяется качеством *функций принадлежности* [48], при помощи которых лингвистические оценки превращаются в количественную форму. Но поскольку функции принадлежности определяются экспертными методами, адекватность таких нечетких моделей целиком зависит от квалификации экспертов. Иначе говоря, проблема адекватности известных нечетких моделей остается открытой. А когда привлечение экспертов для построения модели оказывается невозможным по причине их отсутствия, в таком случае возникает проблема извлечения лингвистических знаний об объекте из экспериментальных данных.

Таким образом, актуальность поднятой проблемы обусловлена тем, что для моделирования многих объектов есть смысл применять логико-лингвистические модели, дающие возможность преодолеть трудности моделирования классическими методами. Однако в многих случаях повышенная сложность того или иного объекта или его новизна (и как следствие - недостаточное освоение) делают невозможным привлечение квалифицированных экспертов для построения таких моделей. Это порождает проблему отыскания закономерностей, которые бы легли в основу системы лингвистических знаний об объекте моделирования, из имеющихся статистических (экспериментальных) данных, характеризующих исследуемый объект. Такие модели обычно являются «грубыми», поэтому не менее важен вопрос о поиске эффективных методов их тонкой настройки.

Данная монография посвящена исследованиям в области идентификации нелинейных многомерных объектов нечеткими базами знаний. Этот метод идентификации впервые был научно обоснован в работе [59], и получил развитие в [55]. В настоящей же работе для решения задачи идентификации предлагается комплексное использование *нечетких баз знаний, нейронных сетей и ге-*

нетических алгоритмов, объединенных под термином *интеллектуальные технологии* или *Soft Computing* (термин, введенный Лотфи Заде).

В монографии представлены результаты исследований, проводимых в научных школах д.т.н., проф. Б.И. Мокина и д.т.н., проф. А.П. Ротштейна. Книга состоит из пяти глав.

Первая глава содержит краткое описание известных методов моделирования сложных нелинейных объектов (и их идентификации в частности), включая общую характеристику нейронных сетей и нечетких баз знаний. Также вкратце характеризуется проблема извлечения знаний, в качестве примера приводятся методы самоорганизации математических моделей как один из возможных вариантов решения данной проблемы. При этом указываются как достоинства, так и недостатки рассматриваемых методов.

Во *второй главе* описывается метод извлечения знаний, систематизированных в виде нечетких баз знаний о нелинейных объектах, с использованием генетических алгоритмов оптимизации. После краткого освещения основных понятий теории нечетких множеств, принципов идентификации на базе нечеткой логики и построения нечетких баз знаний, в терминах математического программирования ставятся задачи их извлечения из экспериментальных данных как задачи оптимизации. Затем дается краткая характеристика генетических алгоритмов оптимизации и описывается методика их применения для решения задачи извлечения нечетких баз знаний.

Третья глава посвящена описанию метода настройки синтезированных «грубых» нечетких баз знаний посредством их заключения в структуру, функционирующую по принципу нейронной сети и обладающую механизмом обучения. На основании анализа моделирования с помощью традиционной нейронной сети предлагается конструкция, обеспечивающая нечеткий логический вывод согласно содержанию правил ЕСЛИ-ТО нечеткой базы знаний, и обладающая механизмом тонкой настройки функций принадлежности лингвистических термов входных переменных по алгоритму Back Propagation - так называемая

нейро-нечеткая сеть. Дееспособность такого метода тонкой настройки нечетких баз знаний подтверждена соответствующими компьютерными экспериментами.

В *четвертой главе* демонстрируются результаты компьютерного моделирования, подтверждающие эффективность предложенного метода извлечения нечетких баз знаний из экспериментальных данных. При этом в качестве эталонных исследовались зависимости как с непрерывным, так и дискретным характером выходной переменной.

В *пятой главе* описывается реальный пример синтеза нечетких баз знаний путем их извлечения из данных эксперимента для моделирования процессов в электроэнергетике. В качестве практической задачи решается задача прогнозирования суточных графиков потребления электрической мощности на основании таких нечетких влияющих факторов, как среднесуточная температура окружающей среды, среднесуточная облачность, время суток, тип суток и время года (сезон).

Текст книги написан Ю.И. Митюшкиным по материалам его кандидатской диссертации, постановка задачи в которой и руководство на начальном этапе работы, а перед этим на этапе магистерской подготовки осуществлялось профессором А.П. Ротштейном, а руководство на заключительном этапе и адаптация к задачам электроэнергетики - профессором Б.И. Мокиным.

Б.И. Мокиным осуществлено также и общее редактирование книги.

ГЛАВА 1

МЕТОДЫ ИДЕНТИФИКАЦИИ МНОГОМЕРНЫХ ЗАВИСИМОСТЕЙ

Данная глава посвящена обзору классических методов идентификации и методов работы с логико-лингвистической информацией. В ней также рассматривается проблема извлечения знаний.

1.1. Идентификация нелинейных объектов

Задачи моделирования нелинейных объектов возникали и возникают в различных сферах человеческой деятельности. С целью решения этих задач параллельно и независимо друг от друга разрабатывались и развивались методы идентификации. Далее будут рассмотрены основные области, в которых идентификация нелинейных объектов занимает ведущее место.

1.1.1. Системы управления.

Наибольшее количество исследований в области идентификации, а также разработок соответствующих методов было проведено именно в рамках теории систем управления. При этом для большинства работ, посвященных данной проблеме, общим является предположение о возможности формального представления любого объекта управления в виде довольно простой математической модели, построенной на основе дифференциальных уравнений. Для более сложных объектов моделирование проводится с применением систем интегро-дифференциальных уравнений, смешанных систем, где наравне с дифференциальными присутствуют также алгебраические или трансцендентные уравнения, которые объединяют отдельные параметры, описывают структуру или принципы функционирования объекта. При этом необходимым условием поиска способа управления объектом является возможность его моделирования математическими средствами [47].

Достаточно полный обзор работ, посвященных решению проблемы идентификации методами классической математики, проведен в работе [79].

В зависимости от характеристик объекта управления, методы его идентификации имеют определенные особенности. Среди прочих следует выделить [78]:

- идентификацию линейных детерминированных объектов;
- непараметрическую идентификацию линейных стохастических систем;
- параметрическую идентификацию стохастических систем.

Для идентификации линейных детерминированных объектов наиболее частое применяются:

а) уравнения связи между входными и выходными переменными объекта в виде систем дифференциальных или разностных уравнений;

б) методы передаточных функций [10];

в) частотные методы [71].

Для непараметрической идентификации линейных стохастических систем наиболее широко применяется уравнение Винера-Хопфа [13,14,30, 79].

В реальных прикладных задачах идентификации наиболее широко применяются методы параметрической идентификации. Необходимым условием для применения этих методов является изначальное наличие уравнения или системы уравнений, формализующих модель исследуемого объекта, в которых могут быть неизвестными некоторые параметры. Методы параметрической идентификации различаются в зависимости от алгоритма вычисления неизвестных параметров или критерия идентификации [77]:

- метод наименьших квадратов и его модификации;
- метод максимального правдоподобия;
- метод байесовских оценок [31];
- метод фильтра Калмана;
- метод стохастической аппроксимации [43];

- метод усредненных невязок [78];
- метод регуляризации [72, 73].

Однако для всех методов, разработанных в рамках теории управления [78], характерны следующие недостатки:

- 1) отсутствие возможности непосредственной работы с переменными, имеющими качественную природу;
- 2) нет возможности использования логико-лингвистической экспертной информации о структуре исследуемого объекта.

1.1.2. Распознавание образов.

Для теории распознавания образов характерным является наличие многих математических методов обработки информации, служащих для установления нелинейных зависимостей между входными и выходными параметрами объекта распознавания по экспериментальным данным [12,74]. Среди основных, в частности, следует выделить:

- аппроксимацию многомерных зависимостей путем разложения по системе ортогональных функций;
- классификацию образов с помощью функций расстояния (метод фазового интервала);
- классификацию образов с помощью функций правдоподобия (методы теории статистических решений [75], байесовский подход [12]);
- методы настраиваемых классификаторов (перцептроны [54], метод потенциальных функций [2], стохастическая аппроксимация [43]).

Недостатками вышеупомянутых методов следует считать те же, что являются характерными для методов теории управления - во-первых, отсутствие возможности непосредственно использовать логико-лингвистическую информацию об объекте, и во-вторых, невозможность работы с качественными переменными (правда, в данном случае это касается лишь входных переменных).

1.1.3. Нейронные сети как средства идентификации.

В настоящее время исследование нейронных сетей [28,34,92] считается одним из наиболее перспективных направлений в области искусственного интеллекта [29,44,70,76]. Нейронные сети являются обобщением идеи перцептрона Розенблата [54]. В основу их функционирования воплощены принципы моделирования работы человеческого мозга и высших позвоночных животных.

С формальной точки зрения нейронная сеть представляет собой универсальную модель-аппроксиматор в виде графа. Моделируя реальный объект, такой граф способен путем так называемого обучения повышать свою адекватность этому объекту за счет модификации весов межэлементных связей. Именно наличие формальных методик обучения (при условии существования обучающей выборки в виде экспериментальных пар «входы-выход») является главным преимуществом нейронных сетей. Среди разнообразных методов обучения в настоящее время наиболее популярным является *алгоритм обратного распространения ошибки* [1,82,92], в основу которого положен градиентный метод поиска оптимума. Суть данного алгоритма состоит в изменении весов межнейронных связей пропорционально мере их влияния (первой производной) на различие между модельными и экспериментальными значениями выходной переменной.

Однако в качестве средства идентификации нелинейных объектов нейронные сети характеризуются определенными недостатками:

1) веса межнейронных связей обученной нейронной сети не подлежат ясной и содержательной интерпретации;

2) градиентный метод, который применяется для обучения нейронной сети, не всегда дает возможность достичь глобального минимума в различии модельных и экспериментальных значений выхода. Поэтому соответствующую задачу оптимизации при обучении сети обычно приходится решать с использованием нескольких различных комбинаций переменных параметров - весов межнейронных связей, что в значительной мере влияет на увеличение времени обу-

чения;

3) на сегодняшний день модели нейронных сетей, использующих качественные входы, развиты недостаточно, чтобы успешно моделировать соответствующие объекты.

1.1.4. Нечеткая логика.

Среди всех известных средств моделирования теория лингвистической переменной, базирующаяся на нечетких множествах [20], позволяет наилучшим образом осуществлять строгую математическую формализацию логико-лингвистической информации, которая применяется при описании сложных нелинейных объектов. Как правило, моделирование таких объектов сводится к построению нечетких баз знаний, которые воплощают в себе экспертные знания об объекте в виде лингвистических высказываний ЕСЛИ-ТО.

Основным формализмом теории нечетких множеств, при помощи которого экспертные знания ЕСЛИ-ТО превращаются в строгие математические модели, является понятие функции принадлежности [3], которая характеризует субъективную меру уверенности эксперта в том, что некоторая величина принадлежит определенному нечеткому понятию (терму), которым характеризуется та или иная входная (выходная) переменная. С целью логической связи функций принадлежности входных и выходных переменных в рамках теории нечетких множеств сформулированы методы нечеткого логического вывода.

Особенность такого моделирования состоит в том, что, поскольку функции принадлежности имеют субъективный характер, качество построенной нечеткой модели полностью зависит от квалификации эксперта, который сформулировал правила ЕСЛИ-ТО и выбрал те или иные формы функций принадлежности. В связи с этим, результаты нечеткого логического вывода иногда могут заметно отличаться от реальных экспериментальных данных. Таким образом, возникает задача настройки параметров нечеткой модели по экспериментальным данным по аналогии с классической теорией идентификации.

На сегодняшний день существует огромное количество источников, из которых можно почерпнуть разнообразную информацию касательно теоретических [8,20,27,45,50,55,96] и прикладных [3-5,21,32,33,35,48,55,59,96] аспектов теории нечетких множеств. В последние годы в ВГТУ научные школы д.т.н., проф. Ротштейна А.П. и д.т.н., проф. Мокина Б.И. активно занимаются развитием теории моделирования нелинейных объектов при помощи нечеткой логики. В частности, были предложены эффективные методы настройки нечетких моделей, которые нашли свое отображение в целом ряде научных работ [26,39-41,55-57,64,83,87,90,95] и в решении ряда прикладных задач [6,18,36-38,42,55,58,63,65,66,86,88,89,91]. Данная монография является развитием этого направления исследований, кроме того, в ней предлагается принципиально новая методика проектирования нечетких баз знаний, основанная на извлечении лингвистических знаний из экспериментальных данных без участия эксперта.

1.2. Задача извлечения знаний

Проблема извлечения знаний (в более широком смысле - проблема обучения) является составной частью инженерии знаний - области информационных технологий, занимающейся решением задачи преобразования знаний (которые ранее мог накапливать и применять на практике лишь человек) в объект обработки на ЭВМ. Особое внимание при этом уделяется технике *извлечения знаний*, которое является составной частью совокупной техники обработки знаний.

Главными функциями обучения являются извлечение необходимой информации из окружающей среды и ее накопление с целью дальнейшего использования. Среди основных аспектов проблемы обучения можно выделить следующие:

- функции объекта обучения;
- свойства среды - источника информации;

- взаимодействие объекта обучения и среды.

Две основные цели, которые ставятся в процессе проведения обучения, это:

- 1) извлечение информации из среды;
- 2) повышение ее уровня и структурирование.

Большинство созданных на сегодня систем обучения ориентированы на анализ механизма получения информации человеком и ее воспроизведения в ЭВМ. Что касается сложного эвристического обучения, присущего человеку, то программными средствами оно практически еще не реализовано.

Что касается извлечения знаний в узком смысле, то в инженерии знаний под данным понятием понимается работа, связанная с получением знаний, которыми владеет конкретное лицо (эксперт). Понятие инженерии знаний в данном случае тоже имеет узкий смысл и означает процесс организации знаний в так называемую базу знаний и построение на ее основе *экспертных систем* - то есть работу по оснащению компьютерных программ специальными экспертными знаниями из определенной проблемной области. Такая работа может выполняться или человеком, или компьютерной программой.

Методы оснащения программ экспертными знаниями можно условно распределить на 3 уровня [50]:

I-й: создание алгоритма и преобразование его в программу разработчиком экспертной системы;

II-й: заполнение пробелов в знаниях усилиями самой программы;

III-й: самостоятельное обретение программой алгоритмических знаний.

В настоящее время большинство экспертных систем - это программы первого уровня. Задача извлечения знаний, которая ставится в контексте их построения, предусматривает применение определенной методики проведения процесса получения необходимых знаний от эксперта (экспертов) с целью их дальнейшей программной обработки.

Современные экспертные системы являются широко распространенной

разновидностью систем обработки информации, которые основаны на знаниях. При этом их особенность состоит в том, что они, как правило, предусматривают активное участие человека-эксперта, используют главным образом основные положения конкретной проблемной области, взятые от того же эксперта, и предназначены для решения сложных задач, которые обычно не решаются неспециалистами в данной области, имея при этом возможность общаться с последними на естественном языке.

Таким образом, в рассматриваемом контексте суть задачи состоит в *получении экспертных знаний от человека*. А методы получения знаний для построения экспертных систем являются альтернативой известным методам «получения того, что знает человек» [50].

Основными атрибутами задачи получения знаний для экспертных систем являются:

- 1) источники знаний;
- 2) сами знания;
- 3) методы их получения.

В качестве классического источника знаний обычно выступает *эксперт*, то есть человек, который владеет достаточными знаниями, навыками и опытом в проблемной области. Относительно того, какие же конкретно знания подлежат извлечению, то здесь определяющее место занимают правила вывода, необходимые для реализации функций системы. Правила вывода обеспечивают связь между объектами, понятиями и атрибутами, которые формируют базовую структуру проблемной области. Второе по значимости место занимают знания по применению средств, доступных лишь экспертам, при этом имеется в виду облегчение их применения неспециалистами. На третьем месте - знания и правила вывода, которые помещаются непосредственно в базу знаний системы.

Прежде чем применять для получения знаний определенный метод, следует установить *критерии целесообразности* его использования, которые должны давать ответ на вопросы: почему применяется именно данный метод? В ка-

кой мере ему присуща эвристичность, или он взят за основу в качестве крайнего средства? Какая необходима поддержка данному методу?

Наиболее распространенными *методами получения знаний* для экспертных систем являются следующие [50]:

- 1) получение знаний путем создания долгодействующей рабочей группы;
- 2) оперативное создание прототипа;
- 3) применение подхода «особое внимание анализу знаний».

Долгодействующая рабочая группа набирается из экспертов в исследуемой области и инженеров по знаниям. Для оперативного создания прототипа подбирается инициативная рабочая группа экспертов, которые опрашивают инженеров по знаниям и в сжатые сроки создают программное обеспечение - прототип экспертной системы, которое с течением времени оценивается и модифицируется. В третьем случае проектирование и реализация экспертной системы начинается лишь после всестороннего подробного предварительного анализа знаний и методов решения задач в проблемной области. Цель такого подхода - уменьшение риска и затрат на построение экспертной системы.

По сравнению с первыми двумя методами, «особое внимание анализу знаний» является наиболее эффективным. Условно его можно разделить на два этапа:

1) четкое ограничение функций экспертной системы и пользователя, предусматривающее анализ проблемной области, определение задач, которые необходимо решить, и сбор информации о рабочей среде экспертов и экспертной системы;

2) четкое разграничение работ, которые возлагаются на экспертную систему, а также ее функций, знаний для реализации экспертных способностей и правил логического вывода. После этого создается так называемая *интерпретационная модель*, предназначенная для интерпретации полученных знаний, и в терминах этой модели описывается информация от экспертов и других источников.

Важным элементом методики извлечения знаний является *техника извлечения знаний*, подразделяющаяся на 6 основных классов [50]:

- 1) опрашивание с наводящими вопросами;
- 2) структурированное опрашивание;
- 3) самонаблюдение;
- 4) самоотчет;
- 5) диалог;
- 6) критический обзор.

Первый метод близок к традиционному диалогу. Основными его структурными элементами являются предложение темы и поиск информации. Главное условие использования этого метода - достаточный терминологический запас у опросчика, а основная цель - «зондирование почвы».

При структурированном опрашивании с помощью заведомо подготовленных вопросов ставится цель узнать о понятиях и моделях («корнях» и «листьях»). Условие применения - владение опросчиком в совершенстве инженерной терминологией, а также усовершенствование техники опрашивания.

Самонаблюдение предусматривает анализ экспертом собственных действий и решений о типе и методах с целью прояснения общей стратегии, их достоверности (а также идей как результатов логического вывода), и общей информации о типе знаний, которые применяются для решения задачи, так как будто они осуществлены кем-то другим, а эксперт их наблюдает и фиксирует «со стороны».

Четвертый метод - это отчет эксперта самому себе о своих действиях и мыслях. Диалог предусматривает наблюдение общения эксперта и неспециалиста, на основе которого получают информацию о процессе решения задачи общими усилиями. Такая информация должна давать возможность делать выводы, что является необходимым условием удобного интерфейса между создаваемой экспертной системой и пользователем. И в конце концов критический обзор используется для получения так называемых метазнаний, что осуществляется пу-

тем повторного изучения знаний, добытых другими методами, совместно с экспертом.

Таким образом, проанализировав вышеописанную проблему и пути ее решения, можно коротко охарактеризовать главную их особенность: с целью получения знаний для построения экспертных систем предполагается участие по меньшей мере двух лиц, одно из которых выступает в роли источника знаний, а второе - их получателя с целью создания на их основе соответствующего программного обеспечения в виде экспертной системы. Такой подход обычно является довольно времязатратным, а отсутствие одной из сторон вообще делает невозможным его применение. Как следствие - поднимается вопрос о возможности автоматизации процесса синтеза систем принятия решений, иначе говоря - их проектирования средствами ЭВМ. Примером реализации такого подхода может служить теория самоорганизации моделей.

1.3. Методы самоорганизации моделей

Вопросом самоорганизации математических моделей уже много лет занимается научная школа А.Г. Ивахненко [22-25] при Институте кибернетики НАН Украины. Ядром теории самоорганизации является метод группового учета аргументов (МГУА). Данный метод дает возможность синтеза математической модели исследуемого объекта по экспериментальным данным наблюдений путем рационализированного перебора многих вариантов с помощью ЭВМ.

Основной принцип самоорганизации состоит в том, что оптимальная модель должна соответствовать минимуму предварительно выбранного внешнего критерия или «ансамбля» критериев. Также основу метода составляют:

а) подход А. Геделя: только те внешние критерии, которые основанные на новой информации, позволяют получить истинную модель объекта, скрытую в зашумленных данных;

б) принцип неконечных решений Д. Габора: всякая сложная однорядная

процедура может быть заменена многорядной (то есть, требующей меньше расчетного времени) лишь при условии сохранения достаточной «свободы выбора» нескольких лучших решений от каждого предыдущего ряда.

Основываясь на этих двух принципах, МГУА позволяет синтезировать *единую* модель оптимальной сложности по минимуму внешнего критерия выбора.

Детерминистический подход, который является по сути господствующим в науке и который основан на анализе причинно-следственных связей, эффективен для относительно простых задач моделирования. Решение же сложных задач целесообразно возложить на ЭВМ, запрограммированную на рационализированный перебор моделей. В таких случаях роль человека (эксперта) ограничивается лишь указанием общих критериев выбора и возможных переменных параметров модели, взятых с запасом, а ЭВМ должна подобрать наиболее эффективное множество выходных и системных переменных, класс уравнений и адекватную опорную функцию (линейные или степенные полиномы, гармоничные раскладки и т. п., классы уравнений - алгебраические, дифференциальные, интегральные или интегро-дифференциальные).

По сравнению с другими существующими методами, которые основаны на детерминистическом подходе, а также методами имитационного моделирования, методы самоорганизации значительно более объективны, поскольку при их применении результат моделирования не зависит от априорного представления эксперта об объекте моделирования. Самоорганизация - это естественное дополнение к существующим методам там, где в результате сложности задачи интуиция человека становится недостаточно продуктивной.

Методы самоорганизации математических моделей, в частности, МГУА, в определенной степени свободны от жестких требований к объекту моделирования. Они не требуют достаточно высокого уровня знаний о нем с целью вывода достоверных математических уравнений, которые описывают все элементы, и выбора оптимальной области моделирования, то есть количества уравне-

ний - иначе говоря, глубокого изучения объекта и значительного объема высококачественной априорной информации. Поскольку МГУА основаны на переборе вариантов по внешним критериям, соответствующий вычислительный алгоритм легко программируется на ЭВМ и позволяет самостоятельно отыскать необходимую информацию путем обработки выборок экспериментальных данных (даже очень незначительных). Главная же особенность МГУА - индуктивный характер выбора модели или системы моделей по указанным внешними критериями и последовательности их применения. Многокритериальность, или применение некоторого комбинированного критерия обеспечивают регуляризацию, то есть однозначность выбора.

Таким образом, подытожив вышесказанное, отметим, что методы самоорганизации предназначены для решения задачи *объективного выбора* модели оптимальной сложности с помощью перебора вариантов по ансамблю внешних критериев, которые указываются автором модели. Именно под этим углом их можно рассматривать в контексте проблемы автоматизированного извлечения знаний. Также преимуществом методов самоорганизации следует считать то, что они позволяют получать модели, на которых можно смело экспериментировать и тем самым осуществлять объективный (не зависящий от представлений автора) системный анализ и идентификацию сложных объектов.

Что касается ограничений, присущих методам самоорганизации (и МГУА в частности), то можно отметить следующее:

1) отсутствие ясного физического смысла и сложность интерпретации, что, как правило, присуще моделям объектов, которые синтезируются в процессе селекции различных моделей;

2) невозможность непосредственного использования нечетких и логико-лингвистических входных переменных исследуемого объекта.

ГЛАВА 2

ИЗВЛЕЧЕНИЕ НЕЧЕТКИХ БАЗ ЗНАНИЙ

Настоящая глава посвящена описанию метода построения нечетких баз знаний путем их извлечения из данных эксперимента. В первых подразделах приведены основные понятия теории нечетких множеств и принципы идентификации нелинейных объектов нечеткими базами знаний, то есть базовый теоретический материал. Затем представляется оптимизационный подход к задаче извлечения нечетких баз знаний, излагаются принципы построения генетических алгоритмов для решения оптимизационных задач, и наконец, раскрывается метод извлечения нечетких баз знаний при помощи генетических алгоритмов для объектов, которые описываются нелинейными зависимостями с непрерывным и дискретным выходом.

2.1. Основные понятия теории нечетких множеств

В качестве базового теоретического материала приведем основные понятия теории нечетких множеств. При этом будем опираться на работу [20] одного из основоположников данной теории Лотфи Заде.

Пусть U - универсальное множество, то есть полное множество, которое охватывает всю проблемную область.

Нечеткое подмножество F множества U определяется функцией принадлежности $\mu^F(u)$, где u - элемент универсального множества, то есть $u \in U$.

Функция принадлежности отображает элементы из множества U на множество чисел в диапазоне $[0,1]$, которые характеризуют степень принадлежности каждого элемента $u \in U$ к нечеткому множеству $F \subset U$.

Если универсальное множество U охватывает конечное число множеств (элементов) u_1, u_2, \dots, u_n , то нечеткое множество F можно представить в виде:

$$F = \mu^F(u_1)/u_1 + \mu^F(u_2)/u_2 + \dots + \mu^F(u_n)/u_n = \sum_{i=1}^n \mu^F(u_i)/u_i . \quad (2.1)$$

Если же множество U представляет собой непрерывный интервал, в таком случае используется следующее обозначение:

$$F = \int_U \mu^F(u)/u . \quad (2.2)$$

Символы \sum и \int в вышеприведенных формулах означают не операцию нахождения дискретной или непрерывной суммы, а символизируют вхождение пар $\mu^F(u)/u$ в общую совокупность.

Основными операциями, которые выполняются над нечеткими множествами, являются операции дополнения, объединения и пересечения, которые определяются следующим образом:

1. Дополнение множества:

$$\bar{F} = \sum_{i=1}^n (1 - \mu^F(u_i))/u_i , \quad \mu^{\bar{F}}(u) = 1 - \mu^F(u). \quad (2.3)$$

2. Объединение множеств:

$$F \cup G = \sum_{i=1}^n \{ \mu^F(u_i) \vee \mu^G(u_i) \} / u_i , \quad \mu^{F \cup G}(u) = \mu^F(u) \vee \mu^G(u), \quad (2.4)$$

где \vee - знак операции взятия максимума.

3. Пересечение множеств:

$$F \cap G = \sum_{i=1}^n \left\{ \mu^F(u_i) \wedge \mu^G(u_i) \right\} / u_i, \quad \mu^{F \cap G}(u) = \mu^F(u) \wedge \mu^G(u), \quad (2.5)$$

где \wedge - знак операции взятия минимума.

Важным также является понятие нечеткого отношения. С его помощью выполняется операция нечеткого логического вывода. Пусть нечеткие множества $F \subset U$ и $G \subset V$, которые заданы на следующих универсальных множествах:

$$U = \{x_1, x_2, \dots, x_l\}; \quad V = \{v_1, v_2, \dots, v_m\}, \quad (2.6)$$

входят в нечеткое правило «ЕСЛИ F , ТО G ». Согласно [20], нечеткое отношение между множествами $F \subset U$ и $G \subset V$ представляет собой матрицу вида:

$$R = F \times G = \sum_{i=1}^l \sum_{j=1}^m \left\{ \mu^F(x_i) \wedge \mu^G(v_j) \right\} / (x_i, v_j), \quad (2.7)$$

в которой элементы, расположенные на пересечении i -й строки и j -го столбца, определяются следующим образом:

$$\mu^R(x, v) = \mu^F(x) \wedge \mu^G(v). \quad (2.8)$$

А операция нечеткого логического вывода имеет следующий вид:

$$\begin{array}{c} F \longrightarrow G \\ \frac{F'}{G'} \end{array} \quad (2.9)$$

Данная запись расшифровывается так: ЕСЛИ факт G следует из факта F ,

То факт G' следует из факта F' , где F, G, F', G' - нечеткие множества. Для определения G' используется формула:

$$G' = F' \circ R = F' \circ (F \times G), \quad (2.10)$$

где \circ - операция *min-max* - композиции, в соответствии с которой

$$G' = \sum_{j=1}^m \vee_{x_i \in U} \left\{ \mu^{F'}(x_i) \wedge \mu^R(x_i, v_j) \right\} / v_j, \quad F, F' \subset U, \quad G, G' \subset V. \quad (2.11)$$

2.2. Принципы идентификации на базе нечеткой логики

В работе [59] был сформулирован ряд принципов идентификации нелинейных объектов нечеткими базами знаний:

1) *Принцип лингвистичности входных и выходных переменных*, в соответствии с которым входы и выходы объекта идентификации описываются лингвистическими переменными, которые в свою очередь оцениваются не количественными, а качественными терминами (от англ. term - называть). Согласно Заде [20], лингвистической переменной называется такая переменная, значениями которой являются слова или словосочетания естественного языка, которые будем называть качественными терминами. Например:

- лингвистическая переменная - РАССТОЯНИЕ; термы - *большое, среднее, маленькое*;
- лингвистическая переменная - СКОРОСТЬ; термы - *высокая, выше средней, средняя, ниже средней, низкая*.

Каждый терм, оценивающий лингвистическую переменную, можно формализовать в виде нечеткого множества, заданного на соответствующем универсальном множестве.

2) *Принцип формирования структуры зависимости “вход-выход” в виде нечеткой базы знаний.* Нечеткая база знаний представляет собой совокупность правил ЕСЛИ <входы> - ТО <выход>, которые отображают опыт эксперта и его понимание причинно-следственных связей, характерных для объекта идентификации. Формирование нечеткой базы знаний является своеобразным аналогом этапа структурной идентификации [78], на котором строится грубая модель объекта с переменными параметрами, которые необходимо настроить. В данном случае переменными параметрами являются параметры функций принадлежности нечетких термов, которые оценивают входы и выходы объекта.

3) *Принцип иерархичности баз знаний.* Данный принцип позволяет избежать проблемы «проклятия размерности», которая обусловлена тем, что при большом количестве входных переменных построение системы высказываний о неизвестной зависимости «вход-выход» значительно усложняется. В связи с этим целесообразно провести классификацию входных переменных и построить на ее основе так называемое дерево вывода, которое будет представлять собой систему иерархически связанных нечетких баз знаний меньшей размерности. Используя принцип иерархичности, можно учитывать практически неограниченное количество входных переменных, влияющих на оценку выхода. При этом рекомендуется, чтобы количество аргументов (входов) в каждом узле дерева - нечетких мини-баз знаний - равнялось 7 ± 2 [85].

4) *Принцип термометра в оценке качественных переменных.* Данный принцип удобно применять тогда, когда эксперт не в состоянии оценить некоторую переменную ни числом, ни качественным термом, а лишь интуитивно ощущает ее уровень. Удобство такого подхода заключается в том, что он позволяет рассматривать различные по природе лингвистические переменные на едином универсальном множестве $U = [\underline{u}, \bar{u}]$. Суть принципа термометра состоит в определении экспертной оценки некоторой переменной путем закрашивания части шкалы (рис. 2.1), у которой левая и правая границы соответствуют минимальному и максимальному уровням той или иной переменной.



Рис. 2.1. Оценка переменной u по принципу термометра.

5) *Принцип двухэтапной настройки баз нечетких знаний.* В соответствии с данным принципом, построение модели нелинейного объекта осуществляется в два этапа, которые аналогичны этапам структурной и параметрической идентификации, характерным для классических методов [77,78].

На первом этапе осуществляется формирование и грубая настройка модели объекта путем построения базы знаний по доступной экспертной информации, что является традиционным для технологии нечетких экспертных систем [59]. Грубую настройку весов правил и форм функций принадлежности можно осуществлять, в частности, при помощи модифицированного метода парных сравнений Саати [69]. На втором этапе осуществляется тонкая настройка нечеткой модели путем ее обучения с помощью экспериментальных данных. Суть тонкой настройки состоит в подборе таких весовых коэффициентов правил ЕСЛИ-ТО и таких параметров функций принадлежности, при которых минимизировалось бы различие между желательным (экспериментальным) и модельным (теоретическим) поведением объекта.

2.3. Нечеткие базы знаний как модели нелинейных объектов

Одним из методов моделирования многомерных зависимостей в задачах кибернетики является их описание нечеткими базами знаний, предназначенными для формализации причинно-следственных связей между переменными «вход-выход», которые характеризуют ту или иную конкретную зависимость. Нечеткие базы знаний воплощают в себе описание этих связей на естественном языке с применением теории нечетких множеств и лингвистических перемен-

ных [20, 27]. В данном подразделе поясняются основные термины и элементы, необходимые для определения нечетких баз знаний.

2.3.1. Оценки входов и выходов объекта моделирования.

Рассмотрим объект с одним выходом и n входами типа:

$$y = f_y(x_1, x_2, \dots, x_n), \quad (2.12)$$

где y - выходная переменная;

x_1, x_2, \dots, x_n - входные переменные.

Переменные x_1, x_2, \dots, x_n и y могут быть как количественными, так и качественными.

Примеры количественных переменных:

СКОРОСТЬ АВТОМОБИЛЯ = [0, 160] км/ч ;

ТЕМПЕРАТУРА ОКРУЖАЮЩЕЙ СРЕДЫ = [-25, 35] °С;

ВОЗРАСТ ТРУДОСПОСОБНОГО ЧЕЛОВЕКА = [17, 60] лет и прочие,

которые легко измеряются в принятых для них количественных диапазонах.

Примером переменной, для которой не существует стандартной количественной шкалы, может служить, скажем, УРОВЕНЬ КАЧЕСТВА ДИПЛОМНОГО ПРОЕКТА, который может оцениваться или качественными терминами (низкий, средний, высокий), или измеряться в шкалах условных величин, например, 5-балльной, 12-балльной и т. п..

Количественные переменные характеризуются известными областями определения:

$$U_i = [x_i, \bar{x}_i], \quad i = \overline{1, n}, \quad (2.13)$$

$$Y = [\underline{y}, \overline{y}], \quad (2.14)$$

где \underline{x}_i (\overline{x}_i) - нижнее (верхнее) значение входной переменной x_i , $i = \overline{1, n}$;

\underline{y} (\overline{y}) - нижнее (верхнее) значение выходной переменной y .

Качественные переменные $x_1 \div x_n$ и y характеризуются множеством всех возможных значений:

$$U_i = \{v_i^1, v_i^2, \dots, v_i^{q_i}\}, \quad i = \overline{1, n}, \quad (2.15)$$

$$Y = \{y^1, y^2, \dots, y^{q_m}\}, \quad (2.16)$$

где v_i^1 ($v_i^{q_i}$) - балльная оценка, которая соответствует минимальному (максимальному) значению входной переменной x_i ;

y^1 (y^{q_m}) - балльная оценка, которая соответствует минимальному (максимальному) значению выходной переменной y ;

q_i , $i = \overline{1, n}$ и q_m - мощности множеств (2.15) и (2.16), причем в общем случае $q_1 \neq q_2 \neq \dots \neq q_n \neq q_m$.

2.3.2. Входы и выход объекта как лингвистические переменные.

Пусть $X^* = \langle x_1^*, x_2^*, \dots, x_n^* \rangle$ - вектор фиксированных значений входных переменных некоторого объекта, где $x_i^* \in U_i$, $i = \overline{1, n}$. Пусть на основе информации о входном векторе X^* необходимо определить выход $y^* \in Y$. Для формального решения данной задачи необходимым условием является наличие за-

зависимости (2.12). Чтобы установить такую зависимость, рассмотрим входные переменные x_i , $i = \overline{1, n}$ и выходную переменную y как лингвистические переменные [20], заданные на универсальных множествах (2.13), (2.14) или (2.15), (2.16).

Для оценки лингвистических переменных x_i , $i = \overline{1, n}$ и y будем использовать качественные термы из следующих терм-множеств:

$A_i = \{a_i^1, a_i^2, \dots, a_i^{l_i}\}$ - терм-множество переменной x_i , $i = \overline{1, n}$;

$D = \{d_1, d_2, \dots, d_m\}$ - терм-множество переменной y , где

a_i^p - p -й лингвистический терм переменной x_i , $p = \overline{1, l_i}$, $i = \overline{1, n}$;

d_j - j -й лингвистический терм переменной y ;

m - количество различных возможных значений переменной y в установленной области ее значений.

Мощности терм-множеств A_i , $i = \overline{1, n}$ в общем случае могут быть разными, то есть $l_1 \neq l_2 \neq \dots \neq l_n$.

Названия отдельных термов $a_i^1, a_i^2, \dots, a_i^{l_i}$ также могут отличаться в зависимости от характера лингвистических переменных x_i , $i = \overline{1, n}$. Например, СКОРОСТЬ АВТОМОБИЛЯ {низкая, средняя, высокая, очень высокая}, ТЕМПЕРАТУРА ВОЗДУХА {холодно, тепло, жарко}, ЧАСТОТА ПУЛЬСА БОЛЬНОГО {замедленная, нормальная, ускоренная}.

Рассмотрим лингвистические термы $a_i^p \in A_i$ и $d_j \in D$, $p = \overline{1, l_i}$, $i = \overline{1, n}$, $j = \overline{1, m}$ как нечеткие множества [27], заданные на универсальных множествах U_i и Y , которые определяются соотношениями (2.13) - (2.16).

Если переменные x_i , $i = \overline{1, n}$ и y являются количественными, нечеткие

множества a_i^p и d_j будут определяться соотношениями:

$$a_i^p = \int_{\underline{x}_i}^{\overline{x}_i} \mu^{a_i^p}(x_i) / x_i, \quad (2.17)$$

$$d_j = \int_{\underline{y}}^{\overline{y}} \mu^{d_j}(y) / y, \quad (2.18)$$

где $\mu^{a_i^p}(x_i)$ - функция принадлежности значения входной переменной

$x_i \in [\underline{x}_i, \overline{x}_i]$ к терму $a_i^p \in A_i$, $p = \overline{1, l_i}$, $i = \overline{1, n}$;

$\mu^{d_j}(y)$ - функция принадлежности значения выходной переменной $y \in [\underline{y}, \overline{y}]$

к терму $d_j \in D$, $j = \overline{1, m}$.

Если переменные x_i , $i = \overline{1, n}$ и y являются качественными, нечеткие множества a_i^p и d_j будут определяться следующим образом:

$$a_i^p = \sum_{k=1}^{q_i} \mu^{a_i^p}(v_i^k) / v_i^k, \quad (2.19)$$

$$d_j = \sum_{r=1}^{q_m} \mu^{d_j}(y^r) / y^r, \quad (2.20)$$

где $\mu^{a_i^p}(v_i^k)$ - степень принадлежности элемента $v_i^k \in U_i$ к терму $a_i^p \in A_i$,

$p = \overline{1, l_i}$, $i = \overline{1, n}$, $k = \overline{1, q_i}$;

$\mu^{d_j}(y^r)$ - степень принадлежности элемента $y^r \in Y$ к терму $d_j \in D$,

$j = \overline{1, m}$.

U_i и Y определяются соотношениями (2.15) и (2.16).

Напомним, что, как и в формулах (2.1), (2.2), знаки интеграла и суммы в соотношениях (2.17) - (2.20) обозначают вхождение пар $\mu(u)/u$ в общую совокупность.

Определение лингвистических оценок переменных и необходимых для их формализации функций принадлежности является первым этапом построения нечеткой модели объекта, который исследуется. В литературе по нечеткой логике [96] он получил название *фаззификации переменных* (от англ. fuzzification).

2.3.3. Структура нечеткой базы знаний.

Пусть для рассматриваемого объекта известно N экспериментальных данных, которые связывают его входы и выход. При этом данную выборку можно упорядочить следующим образом:

$$N = k_1 + k_2 + \dots + k_j + \dots + k_m, \quad (2.21)$$

где k_j - количество экспериментальных данных, которые соответствуют значению d_j выходной переменной y , $j = \overline{1, m}$, а знак «+» в данном случае означает не объединение элементов, а их арифметическую сумму;

m - общее количество значений выходной переменной, причем в общем случае $k_1 \neq k_2 \neq \dots \neq k_j \neq \dots \neq k_m$.

Будем считать, что $N < l_1 \cdot l_2 \cdot \dots \cdot l_n$, то есть количество имеющихся экспериментальных данных меньше полного перебора различных комбинаций возможных значений входных переменных объекта l_i , $i = \overline{1, n}$.

Экспериментальную выборку пронумеруем следующим образом:

$11, 12, \dots, 1k_1$ - номера комбинаций значений входных переменных, соответствующих значению выхода d_1 ;

$21, 22, \dots, 2k_2$ - номера комбинаций значений входных переменных, соответствующих значению выхода d_2 ;

...

$j1, j2, \dots, jk_j$ - номера комбинаций значений входных переменных, соответствующих значению выхода d_j ;

...

$m1, m2, \dots, mk_m$ - номера комбинаций значений входных переменных, соответствующих значению выхода d_m .

На основании введенных обозначений сформируем так называемую матрицу знаний (табл. 2.1), при этом будем руководствоваться следующими правилами:

- размерность матрицы должна равняться $(n+1) \times N$, где $(n+1)$ - количество столбцов, а $N = k_1 + k_2 + \dots + k_m$ - количество строк;

- первые n столбцов матрицы соответствуют входным переменным x_i , $i = \overline{1, n}$, а $(n+1)$ -и столбец соответствует значениям d_j выходной переменной y , $j = \overline{1, m}$;

- каждая строка матрицы представляет собой комбинацию значений входных переменных, отнесенную экспертом к одному из возможных значений выходной переменной y , при этом первые k_1 строк соответствуют значению выходной переменной $y = d_1$, следующие k_2 строк - значению $y = d_2$ и т. д., а последние k_m строк - значению $y = d_m$;

Матрица знаний

Номер входной комбинации	Входные переменные				Выходная переменная
	x_1	x_2	$\dots x_i \dots$	x_n	y
11	a_1^{11}	a_2^{11}	$\dots a_i^{11} \dots$	a_n^{11}	d_1
12	a_1^{12}	a_2^{12}	$\dots a_i^{12} \dots$	a_n^{12}	
...	
$1 k_1$	$a_1^{1k_1}$	$a_2^{1k_1}$	$\dots a_i^{1k_1} \dots$	$a_n^{1k_1}$	
...
$j1$	a_1^{j1}	a_2^{j1}	$\dots a_i^{j1} \dots$	a_n^{j1}	d_j
$j2$	a_1^{j2}	a_2^{j2}	$\dots a_i^{j2} \dots$	a_n^{j2}	
...	
jk_j	$a_1^{jk_j}$	$a_2^{jk_j}$	$\dots a_i^{jk_j} \dots$	$a_n^{jk_j}$	
...
$m1$	a_1^{m1}	a_2^{m1}	$\dots a_i^{m1} \dots$	a_n^{m1}	d_m
$m2$	a_1^{m2}	a_2^{m2}	$\dots a_i^{m2} \dots$	a_n^{m2}	
...	
mk_m	$a_1^{mk_m}$	$a_2^{mk_m}$	$\dots a_i^{mk_m} \dots$	$a_n^{mk_m}$	

- элемент a_i^{jp} , который находится на пересечении i -го столбца и jp -й строки, соответствует лингвистической оценке параметра x_i в строке матрицы знаний с номером jp . Лингвистическая оценка a_i^{jp} выбирается из термножества, соответствующего переменной x_i , то есть $a_i^{jp} \in A_i$, $i = \overline{1, n}$, $j = \overline{1, m}$,

$$p = \overline{1, k_j}.$$

Введенная матрица знаний определяет систему логических высказываний типа «ЕСЛИ-ТО, ИНАЧЕ», которые связывают значение входных переменных $x_1 \div x_n$ с одним из возможных значений выхода d_j , $j = \overline{1, m}$:

$$\begin{aligned} &\text{ЕСЛИ } (x_1 = a_1^{11}) \text{ И } (x_2 = a_2^{11}) \text{ И } \dots \text{ И } (x_n = a_n^{11}) \text{ ИЛИ} \\ &\quad (x_1 = a_1^{12}) \text{ И } (x_2 = a_2^{12}) \text{ И } \dots \text{ И } (x_n = a_n^{12}) \text{ ИЛИ } \dots \\ &\quad (x_1 = a_1^{1k_1}) \text{ И } (x_2 = a_2^{1k_1}) \text{ И } \dots \text{ И } (x_n = a_n^{1k_1}), \end{aligned}$$

ТО $y = d_1$, ИНАЧЕ

$$\begin{aligned} &\text{ЕСЛИ } (x_1 = a_1^{21}) \text{ И } (x_2 = a_2^{21}) \text{ И } \dots \text{ И } (x_n = a_n^{21}) \text{ ИЛИ} \\ &\quad (x_1 = a_1^{22}) \text{ И } (x_2 = a_2^{22}) \text{ И } \dots \text{ И } (x_n = a_n^{22}) \text{ ИЛИ } \dots \\ &\quad (x_1 = a_1^{2k_2}) \text{ И } (x_2 = a_2^{2k_2}) \text{ И } \dots \text{ И } (x_n = a_n^{2k_2}), \end{aligned}$$

ТО $y = d_2$, ИНАЧЕ ...

$$\begin{aligned} &\text{ЕСЛИ } (x_1 = a_1^{m1}) \text{ И } (x_2 = a_2^{m1}) \text{ И } \dots \text{ И } (x_n = a_n^{m1}) \text{ ИЛИ} \\ &\quad (x_1 = a_1^{m2}) \text{ И } (x_2 = a_2^{m2}) \text{ И } \dots \text{ И } (x_n = a_n^{m2}) \text{ ИЛИ } \dots \\ &\quad (x_1 = a_1^{mk_m}) \text{ И } (x_2 = a_2^{mk_m}) \text{ И } \dots \text{ И } (x_n = a_n^{mk_m}), \end{aligned}$$

ТО $y = d_m$, (2.22)

где a_i^{jp} - лингвистическая оценка входной переменной x_i в p -й строке j -й дизъюнкции, которая выбирается из соответствующего терм-множества A_i ($i = \overline{1, n}$, $j = \overline{1, m}$, $p = \overline{1, k_j}$);

k_j - количество правил, которые определяют значение выходной перемен-

ной $y = d_j$;

$d_j (j = \overline{1, m})$ - лингвистическая оценка выходной переменной y , которая определяется из терм-множества D .

Полученная система логических высказываний представляет собой так называемую *нечеткую базу знаний*.

Если в данную систему ввести операции \cup (ИЛИ) и \cap (И), это даст возможность записать ее более компактно:

$$\bigcup_{p=1}^{k_j} \left[\bigcap_{i=1}^n (x_i = a_i^{jp}) \right] \longrightarrow y = d_j, \quad j = \overline{1, m}. \quad (2.23)$$

Таким образом, искомое соотношение (2.12), устанавливающее связь между входными параметрами x_i и выходной переменной y , формализовано в виде системы нечетких логических высказываний (2.23), которая базируется на соответствующей матрице знаний (см. табл. 2.1).

2.3.4. Модели функций принадлежности входных переменных.

По определению, данному в [20], функция принадлежности $\mu^T(x)$ характеризует субъективную меру (в диапазоне $[0,1]$) уверенности эксперта в том, что четкое значение x соответствует нечеткому терму T . На практике наиболее распространенными являются треугольные, трапециевидные, квазиколоколообразные и гауссовы функции принадлежности [83,96], поскольку они имеют возможность изменять свою форму путем изменения собственных параметров.

В работах [56,64] была предложена аналитическая модель функции принадлежности переменной x к произвольному нечеткому терму T в виде:

$$\mu^T(x) = \frac{1}{1 + \left(\frac{x-b}{c}\right)^2}, \quad (2.24)$$

где b и c - параметры настройки: b - координата максимума функции, $\mu^T(b) = 1$; c - коэффициент концентрации-растяжения функции (рис. 2.2). Фактически число b представляет собой наиболее прагматичное значение переменной x для нечеткого термина T .

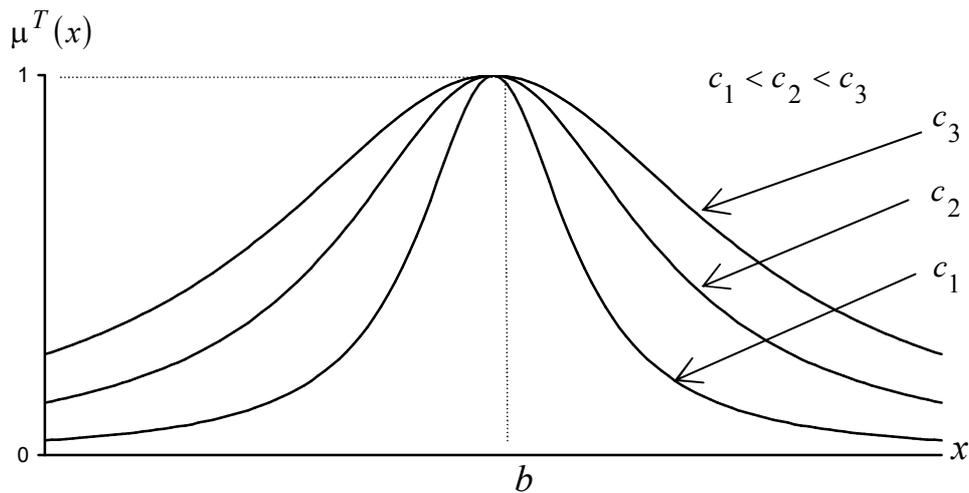


Рис. 2.2. Модель квазиколоколообразной функции принадлежности.

Основными преимуществами модели функций принадлежности такого типа являются, во-первых, простота, и во-вторых, удобство настройки.

Данную методику моделирования изложим, исходя из работ [26,55,56,59, 64], чтобы показать в дальнейшем, каким образом ее удалось усовершенствовать.

2.3.5. Постановка задачи моделирования.

Пусть для объекта типа (2.12) известно:

- множество входных переменных $X = (x_1, x_2, \dots, x_n)$;

- диапазоны количественного изменения каждой входной переменной

$$x_i \in [\underline{x}_i, \overline{x}_i], \quad i = \overline{1, n};$$

- функции принадлежности, которые позволяют представлять переменные x_i , $i = \overline{1, n}$ в виде нечетких множеств (2.17) или (2.19);

- матрица знаний, построенная в соответствии с правилами, введенным в п. 2.3.3.

В зависимости от характера выходной переменной, исследуемый объект может рассматриваться как *объект с дискретным выходом* или *объект с непрерывным выходом*.

Для объекта с дискретным выходом допускается известным множество решений $D = \{d_1, d_2, \dots, d_m\}$, соответствующих выходной переменной y .

Объект с непрерывным выходом характеризуется интервалом $[\underline{y}, \bar{y}]$, на котором изменяется выходная переменная y . С целью описания такого объекта нечеткой базой знаний интервал выходной переменной разбивается на m подинтервалов следующим образом:

$$[\underline{y}, \bar{y}] = \underbrace{[\underline{y}, y_1]}_{d_1} \cup \underbrace{[y_1, y_2]}_{d_2} \cup \dots \cup \underbrace{[y_{j-1}, y_j]}_{d_j} \cup \dots \cup \underbrace{[y_{m-1}, \bar{y}]}_{d_m}. \quad (2.25)$$

Необходимо: разработать методику принятия решения, при помощи которой фиксированному вектору входных переменных $X^* = \langle x_1^*, x_2^*, \dots, x_n^* \rangle$, $x_i^* \in [\underline{x}_i, \bar{x}_i]$ однозначно соответствовало бы решение $y \in D$ (для объекта с дискретным выходом) или $y \in [\underline{y}, \bar{y}]$ (для объекта с непрерывным выходом).

С целью решения поставленной задачи в работе [55] был предложен метод, основанный на использовании нечетких логических уравнений, которые строятся на базе матрицы знаний или изоморфной к ней системы логических высказываний (2.22) и позволяют вычислять значения функций принадлежности выходной переменной при фиксированных значениях входов объекта.

2.3.6. Система нечетких логических уравнений.

Лингвистические оценки a_i^{jp} переменных $x_1 \div x_n$, входящих в логические высказывания (2.22), будем рассматривать как нечеткие множества, определенные на универсальных множествах $U_i = [x_i, \overline{x_i}]$, $i = \overline{1, n}$, $j = \overline{1, m}$.

Введем следующие обозначения:

$\mu^{a_i^{jp}}(x_i)$ - функция принадлежности параметра $x_i \in [x_i, \overline{x_i}]$ к нечеткому терму a_i^{jp} , $i = \overline{1, n}$, $j = \overline{1, m}$, $p = \overline{1, k_j}$;

$\mu^{d_j}(x_1, x_2, \dots, x_n)$ - функция принадлежности вектора входных переменных $X = (x_1, x_2, \dots, x_n)$ значению выходной переменной $y = d_j$, $j = \overline{1, m}$.

Таким образом, имеем два типа функций, связь между которыми определяется нечеткой базой знаний согласно (2.23), на основе чего можно вывести систему следующих уравнений:

$$\begin{aligned} \mu^{d_1}(x_1, x_2, \dots, x_n) &= \left[\mu^{a_1^{11}}(x_1) \wedge \mu^{a_2^{11}}(x_2) \wedge \dots \wedge \mu^{a_n^{11}}(x_n) \right] \vee \\ &\vee \left[\mu^{a_1^{12}}(x_1) \wedge \mu^{a_2^{12}}(x_2) \wedge \dots \wedge \mu^{a_n^{12}}(x_n) \right] \vee \dots \\ &\dots \vee \left[\mu^{a_1^{1k_1}}(x_1) \wedge \mu^{a_2^{1k_1}}(x_2) \wedge \dots \wedge \mu^{a_n^{1k_1}}(x_n) \right], \\ \mu^{d_2}(x_1, x_2, \dots, x_n) &= \left[\mu^{a_1^{21}}(x_1) \wedge \mu^{a_2^{21}}(x_2) \wedge \dots \wedge \mu^{a_n^{21}}(x_n) \right] \vee \\ &\vee \left[\mu^{a_1^{22}}(x_1) \wedge \mu^{a_2^{22}}(x_2) \wedge \dots \wedge \mu^{a_n^{22}}(x_n) \right] \vee \dots \\ &\dots \vee \left[\mu^{a_1^{2k_2}}(x_1) \wedge \mu^{a_2^{2k_2}}(x_2) \wedge \dots \wedge \mu^{a_n^{2k_2}}(x_n) \right], \end{aligned}$$

$$\begin{aligned}
& \dots \\
\mu^{d_m}(x_1, x_2, \dots, x_n) = & \left[\mu^{a_1^{m1}}(x_1) \wedge \mu^{a_2^{m1}}(x_2) \wedge \dots \wedge \mu^{a_n^{m1}}(x_n) \right] \vee \\
& \vee \left[\mu^{a_1^{m2}}(x_1) \wedge \mu^{a_2^{m2}}(x_2) \wedge \dots \wedge \mu^{a_n^{m2}}(x_n) \right] \vee \dots \\
& \dots \vee \left[\mu^{a_1^{mk_m}}(x_1) \wedge \mu^{a_2^{mk_m}}(x_2) \wedge \dots \wedge \mu^{a_n^{mk_m}}(x_n) \right], \quad (2.26)
\end{aligned}$$

где \vee - логическое ИЛИ, \wedge - логическое И.

Данные нечеткие логические уравнения получены из нечеткой базы знаний (2.23) путем замены лингвистических термов a_i^{jp} и d_j на соответствующие функции принадлежности, а операций \cup и \cap - на операции \vee и \wedge . В более компактном виде данную систему логических уравнений можно записать следующим образом:

$$\mu^{d_j}(x_1, x_2, \dots, x_n) = \bigvee_{p=1}^{k_j} \left[\bigwedge_{i=1}^n \mu^{a_i^{jp}}(x_i) \right], \quad j = \overline{1, m}. \quad (2.27)$$

2.3.7. Решение системы нечетких логических уравнений и дефаззификация результатов.

Теперь у нас есть все необходимое для разработки методики определения значения выходной переменной y , соответствующего вектору фиксированных значений входных переменных $X^* = \langle x_1^*, x_2^*, \dots, x_n^* \rangle$.

Представим методику моделирования нелинейных объектов нечеткими базами знаний в виде алгоритма, который состоит из пяти шагов.

ШАГ 1. Зафиксировать вектор значений входных переменных

$$X^* = (x_1^*, x_2^*, \dots, x_n^*).$$

ШАГ 2. Задать функции принадлежности нечетких термов, которые используются в нечеткой базе знаний (2.22), и вычислить значение этих функций для заданных значений входных переменных $x_1^* \div x_n^*$.

ШАГ 3. Согласно логическим уравнениям (2.27) вычислить значение многомерных функций принадлежности $\mu^{d_j}(x_1^*, x_2^*, \dots, x_n^*)$ вектора X^* для всех значений d_j , $j = \overline{1, m}$ выходной переменной y . Логические операции И (\wedge) и ИЛИ (\vee) над функциями принадлежности заменяются на операции \min и \max [27]:

$$\mu(a) \wedge \mu(b) = \min[\mu(a), \mu(b)], \quad (2.28)$$

$$\mu(a) \vee \mu(b) = \max[\mu(a), \mu(b)]. \quad (2.29)$$

ШАГ 4. ЕСЛИ выход объекта дискретный, ТО определить значение d_j^* , функция принадлежности которого максимальна:

$$\mu^{d_j^*}(x_1^*, x_2^*, \dots, x_n^*) = \max_{j=1, m} \left(\mu^{d_j}(x_1^*, x_2^*, \dots, x_n^*) \right) - \quad (2.30)$$

данное значение функции принадлежности является искомым решением для вектора значений входных переменных $X^* = (x_1^*, x_2^*, \dots, x_n^*)$ - перейти на КОНЕЦ АЛГОРИТМА;

ИНАЧЕ ЕСЛИ выход объекта непрерывный, ТО имеем нечеткое множество, которое соответствует выходной переменной y :

$$\bar{y} = \left\{ \frac{\mu^{d_1}(y)}{[y, y_1]}, \frac{\mu^{d_2}(y)}{[y_1, y_2]}, \dots, \frac{\mu^{d_m}(y)}{[y_{m-1}, \bar{y}]} \right\}. \quad (2.31)$$

ШАГ 5. Для определения четкого числа из интервала $[y, \bar{y}]$, который соответствует нечеткому множеству (2.31), необходимо применить операцию дефаззификации [96] (от англ. defuzzification); в общем случае данная операция предназначена для преобразования нечеткой (fuzzy) информации в четкую форму. Итак, четкое число y^* , которое соответствует нечеткому множеству (2.31), определим следующим образом:

$$y^* = \frac{y\mu^{d_1}(y) + y_1\mu^{d_2}(y) + \dots + y_{m-1}\mu^{d_m}(y)}{\mu^{d_1}(y) + \mu^{d_2}(y) + \dots + \mu^{d_m}(y)}. \quad (2.32)$$

При вероятностной интерпретации степеней принадлежности формула (2.32) может рассматриваться как аналог математического ожидания дискретной случайной величины. Если интервал $[y, \bar{y}]$ разбить на m равных отрезков, то есть

$$y_1 = y + \Delta, \quad y_2 = y + 2\Delta, \quad \dots, \quad y_{m-1} = \bar{y} - \Delta, \quad \Delta = \frac{\bar{y} - y}{m-1}, \quad (2.33)$$

то формула (2.32) упрощается и принимает вид, удобный для расчетов:

$$y^* = \frac{\sum_{j=1}^m [y + (j-1)\Delta] \mu^{d_j}(y)}{\sum_{j=1}^m \mu^{d_j}(y)}. \quad (2.34)$$

КОНЕЦ АЛГОРИТМА.

Предложенная методика использует алгоритм идентификации лингвистического термина по максимуму функции принадлежности [20] (в случае дискретного выхода), и обобщает эту идею на всю матрицу знаний.

Вычислительную часть алгоритма можно легко реализовать, построив на основе матрицы знаний матрицу значений функций принадлежности и выполнив над ними операции \min и \max (рис. 2.3).

$\mu^{11}(x_1)$	$\mu^{11}(x_2)$	\dots	$\mu^{11}(x_n)$	} min	}	max
$\mu^{12}(x_1)$	$\mu^{12}(x_2)$	\dots	$\mu^{12}(x_n)$	} min		
\dots	\dots	\dots	\dots			
$\mu^{1k_1}(x_1)$	$\mu^{1k_1}(x_2)$	\dots	$\mu^{1k_1}(x_n)$	} min	}	max
\dots	\dots	\dots	\dots			
$\mu^{21}(x_1)$	$\mu^{21}(x_2)$	\dots	$\mu^{21}(x_n)$	} min		
$\mu^{22}(x_1)$	$\mu^{22}(x_2)$	\dots	$\mu^{22}(x_n)$	} min	}	max
\dots	\dots	\dots	\dots			
$\mu^{2k_2}(x_1)$	$\mu^{2k_2}(x_2)$	\dots	$\mu^{2k_2}(x_n)$	} min		
\dots	\dots	\dots	\dots		}	max
$\mu^{m1}(x_1)$	$\mu^{m1}(x_2)$	\dots	$\mu^{m1}(x_n)$	} min		
$\mu^{m2}(x_1)$	$\mu^{m2}(x_2)$	\dots	$\mu^{m2}(x_n)$	} min		
\dots	\dots	\dots	\dots		}	max
$\mu^{mk_m}(x_1)$	$\mu^{mk_m}(x_2)$	\dots	$\mu^{mk_m}(x_n)$	} min		

Рис. 2.3. Матричная реализация алгоритма принятия решения.

Приведенный алгоритм нахождения как дискретных значений $\{d_1, d_2, \dots, d_m\}$, так и четких числовых значений из диапазона $[y, \bar{y}]$ выходной переменной y по заданному вектору фиксированных значений входных переменных $X^* = \langle x_1^*, x_2^*, \dots, x_n^* \rangle$ и матрицей знаний позволяет создать при определенных условиях методику моделирования нелинейного объекта $y = f_y(x_1, x_2, \dots, x_n)$ любой сложности.

2.4. Извлечение нечетких баз знаний как задача оптимизации

Как следует из вышеприведенного, адекватность моделей, которые описываются нечеткими базами знаний, данным эксперимента определяется типом и качеством настройки функций принадлежности, с помощью которых лингвистические оценки параметров модели превращаются в количественную форму. Но поскольку функции принадлежности в свою очередь определяются экспертными методами [59,69], то адекватность нечеткой модели в целом также зависит от квалификации экспертов. Кроме того, далеко не всегда существует возможность привлечения эксперта, который бы промоделировал тот или иной объект (процесс), что может быть связано с его сложностью или определенной новизной и как следствие - недостаточным освоением. В связи с этим возникает вопрос о возможности автоматизации процесса построения нечетких баз знаний, которые моделируют указанные объекты, на основе имеющихся экспериментальных данных, полученных в результате их исследования. Так, преобразование экспериментальной информации в нечеткие базы знаний могло бы быть полезным в медицине, менеджменте, электроэнергетике и других областях, где лица, принимающие решения, вместо строгих количественных соотношений (которых они зачастую не имеют по объективным причинам) отдают предпочтение прозрачным и легким для интерпретации словесным правилам.

Суть метода, предложенного в данной главе, состоит в том, что в результате обработки экспериментальной выборки данных с помощью ЭВМ определяются такие параметры функций принадлежности нечетких термов и такие веса нечетких правил, которые минимизируют расхождение между результатами нечеткого логического вывода и экспериментальными данными. Для решения соответствующей задачи оптимизации используем технику генетических алгоритмов.

2.4.1. Нечеткие логические уравнения с весами правил.

При формулировании экспертом лингвистических правил типа ЕСЛИ-ТО, которые образуют нечеткую базу знаний о некотором объекте, уверенность эксперта в том или ином правиле может быть разной. Если одно правило, по мнению эксперта, может служить в качестве бесспорной истины, то относительно другого правила у того же эксперта могут быть некоторые сомнения. С целью отображения этих разных степеней уверенности в базу нечетких знаний в работе [64] вводятся *весы правил* - числа из интервала $[0,1]$, характеризующие уверенность эксперта в каждом выбранном для принятия решения конкретном правиле.

Учитывая веса правил, нечеткая база знаний (2.22) о неизвестной зависимости (2.12) примет следующий вид:

ЕСЛИ $(x_1 = a_1^{11})$ И $(x_2 = a_2^{11})$ И... И $(x_n = a_n^{11})$ (с весом w_{11})
ИЛИ $(x_1 = a_1^{12})$ И $(x_2 = a_2^{12})$ И... И $(x_n = a_n^{12})$ (с весом w_{12})
ИЛИ ...
ИЛИ $(x_1 = a_1^{1k_1})$ И $(x_2 = a_2^{1k_1})$ И... И $(x_n = a_n^{1k_1})$ (с весом w_{1k_1}),
ТО $y = d_1$, ИНАЧЕ
ЕСЛИ $(x_1 = a_1^{21})$ И $(x_2 = a_2^{21})$ И... И $(x_n = a_n^{21})$ (с весом w_{21})
ИЛИ $(x_1 = a_1^{22})$ И $(x_2 = a_2^{22})$ И... И $(x_n = a_n^{22})$ (с весом w_{22})
ИЛИ ...
ИЛИ $(x_1 = a_1^{2k_2})$ И $(x_2 = a_2^{2k_2})$ И... И $(x_n = a_n^{2k_2})$ (с весом w_{2k_2}),
ТО $y = d_2$, ИНАЧЕ ...
ЕСЛИ $(x_1 = a_1^{m1})$ И $(x_2 = a_2^{m1})$ И... И $(x_n = a_n^{m1})$ (с весом w_{m1})
ИЛИ $(x_1 = a_1^{m2})$ И $(x_2 = a_2^{m2})$ И... И $(x_n = a_n^{m2})$ (с весом w_{m2})

ИЛИ ...

ИЛИ $(x_1 = a_1^{mk_m})$ И $(x_2 = a_2^{mk_m})$ И ... И $(x_n = a_n^{mk_m})$ (с весом w_{mk_m}),

ТО $y = d_m$, (2.35)

где a_i^{jp} - лингвистическая оценка входной переменной x_i , которая выбирается

из соответствующего терм-множества A_i ($i = \overline{1, n}$, $j = \overline{1, m}$, $p = \overline{1, k_j}$),

в p -й строке j -й дизъюнкции;

w_{jp} - вес правила;

d_j ($j = \overline{1, m}$) соответствует:

а) одному из типов (классов) решений, если речь идет об объекте с дискретным выходом;

б) подинтервалу значений выходной переменной y , если речь идет об объекте с непрерывным выходом;

k_j - количество правил, которые определяют значение выходной переменной $y = d_j$.

Данной модифицированной нечеткой базе знаний будет соответствовать следующая система нечетких логических уравнений:

$$\mu^{d_j}(x_1, x_2, \dots, x_n) = \bigvee_{p=1}^{k_j} \left\{ w_{jp} \left[\bigwedge_{i=1}^n \mu^{a_i^{jp}}(x_i) \right] \right\}, \quad j = \overline{1, m}. \quad (2.36)$$

Учитывая то, что операциям \vee и \wedge в теории нечетких множеств [20] соответствуют операции \max и \min , из (2.36) получим:

$$\mu^{d_j}(x_1, x_2, \dots, x_n) = \max_{p=1, \overline{k_j}} \left\{ w_{jp} \min_{i=1, \overline{n}} \left[\mu^{a_i^{jp}}(x_i) \right] \right\}, \quad j = \overline{1, m}. \quad (2.37)$$

2.4.2. Задача оптимизации для объекта с дискретным выходом.

При изложении материала этого параграфа используем результаты работы [55].

Пусть для объекта типа (2.12) известно следующее:

а) интервалы изменения входов и выхода:

$$x_i \in \left[\underline{x}_i, \overline{x}_i \right], \quad i = \overline{1, n}, \quad y \in \left[\underline{y}, \overline{y} \right], \quad (2.38)$$

при этом выход является дискретным, то есть разбитым на классы решений d_j ($j = \overline{1, m}$) следующим образом:

$$\left[\underline{y}, \overline{y} \right] = \underbrace{\left[\underline{y}, y_1 \right]}_{d_1} \cup \underbrace{\left[y_1, y_2 \right]}_{d_2} \cup \dots \cup \underbrace{\left[y_{j-1}, y_j \right]}_{d_j} \cup \dots \cup \underbrace{\left[y_{m-1}, \overline{y} \right]}_{d_m}; \quad (2.39)$$

б) выборка экспериментальных данных в виде M пар «входы-выход»:

$\left\{ X_p, d_p \right\}$, где $X_p = \left\{ x_1^p, x_2^p, \dots, x_n^p \right\}$ - вектор значений входных переменных в p -й паре, $p = \overline{1, M}$.

Необходимо синтезировать знания об объекте в виде системы нечетких логических высказываний типа (2.35), то есть в виде нечеткой базы знаний.

Модель объекта с дискретным выходом, которому соответствует нечеткая база знаний (2.35), можно представить в виде аналитических выражений (2.37) и (2.24), или в более компактном виде:

$$\mu^{dj}(y) = \mu^{dj}(X, B, C, W), \quad j = \overline{1, m}, \quad (2.40)$$

где $X = (x_1, x_2, \dots, x_n)$ - вектор входных переменных;

$B = (b_1, b_2, \dots, b_q)$ и $C = (c_1, c_2, \dots, c_q)$ - векторы параметров настройки функций

принадлежности (2.24);

$W = (w_1, w_2, \dots, w_N)$ - вектор весов правил нечеткой базы знаний (2.35);

q - общее количество термов в (2.35);

N - общее количество строк в нечеткой базе знаний (2.35).

Синтезированная нечеткая база знаний должна соответствовать определенным ограничениям на свой объем. Зададим эти ограничения следующим образом:

а) $N = k_1 + k_2 + \dots + k_m \leq \overline{N}$;

б) $k_1 \leq \overline{k_1}, k_2 \leq \overline{k_2}, \dots, k_m \leq \overline{k_m}$,

где \overline{N} - максимально допустимое общее количество строк-конъюнкций в нечеткой базе знаний типа (2.35);

$\overline{k_j}$ - максимально допустимое общее количество строк-конъюнкций в пра-

вилах, соответствующих j -му классу решений, $j = \overline{1, m}$.

Однако ни содержание, ни количество лингвистических термов a_i^{jp} ($i = \overline{1, n}$, $p = \overline{1, k_j}$, $j = \overline{1, m}$) в будущей базе знаний (2.35) заведомо не являются известными. Поэтому их интерпретацию предлагается проводить на основе значений параметров (b_i^{jp}, c_i^{jp}) функций принадлежности (2.24). Таким образом, синтез базы знаний (2.35) сводится к извлечению матрицы параметров, сведенных в табл. 2.2.

Таблица 2.2

Матрица параметров базы знаний

Номер правила	ЕСЛИ			Вес	ТО
	x_1	$\dots x_i \dots$	x_n		y
11	$\left(b_1^{11}, c_1^{11}\right)$	$\left(b_i^{11}, c_i^{11}\right)$	$\left(b_n^{11}, c_n^{11}\right)$	w_{11}	d_1
12	$\left(b_1^{12}, c_1^{12}\right)$	$\left(b_i^{12}, c_i^{12}\right)$	$\left(b_n^{12}, c_n^{12}\right)$	w_{12}	
...	
$1k_1$	$\left(b_1^{1k_1}, c_1^{1k_1}\right)$	$\left(b_i^{1k_1}, c_i^{1k_1}\right)$	$\left(b_n^{1k_1}, c_n^{1k_1}\right)$	w_{1k_1}	
...	
$j1$	$\left(b_1^{j1}, c_1^{j1}\right)$	$\left(b_i^{j1}, c_i^{j1}\right)$	$\left(b_n^{j1}, c_n^{j1}\right)$	w_{j1}	d_j
$j2$	$\left(b_1^{j2}, c_1^{j2}\right)$	$\left(b_i^{j2}, c_i^{j2}\right)$	$\left(b_n^{j2}, c_n^{j2}\right)$	w_{j2}	
...	
jk_j	$\left(b_1^{jk_j}, c_1^{jk_j}\right)$	$\left(b_i^{jk_j}, c_i^{jk_j}\right)$	$\left(b_n^{jk_j}, c_n^{jk_j}\right)$	w_{jk_j}	
...	
$m1$	$\left(b_1^{m1}, c_1^{m1}\right)$	$\left(b_i^{m1}, c_i^{m1}\right)$	$\left(b_n^{m1}, c_n^{m1}\right)$	w_{m1}	d_m
$m2$	$\left(b_1^{m2}, c_1^{m2}\right)$	$\left(b_i^{m2}, c_i^{m2}\right)$	$\left(b_n^{m2}, c_n^{m2}\right)$	w_{m2}	
...	
mk_m	$\left(b_1^{mk_m}, c_1^{mk_m}\right)$	$\left(b_i^{mk_m}, c_i^{mk_m}\right)$	$\left(b_n^{mk_m}, c_n^{mk_m}\right)$	w_{mk_m}	

Суть задачи оптимизации состоит в нахождении вектора неизвестных параметров (B, C, W) , минимизирующих различие между модельными результатами и экспериментальными данными. Если для достижения этой цели воспользоваться методом наименьших квадратов, то данная задача в терминах математического программирования для объекта с дискретным выходом может быть сформулирована так:

найти такую матрицу (см. табл. 2.2), которая бы удовлетворяла следую-

шим ограничениям на диапазоны изменения параметров (B, C, W) :

$$b_j \in [\underline{b}_j, \overline{b}_j], c_j \in [\underline{c}_j, \overline{c}_j], j = \overline{1, q}, w_i \in [\underline{w}_i, \overline{w}_i], i = \overline{1, N}, \quad (2.41)$$

а также количество строк, и при этом обеспечивала бы:

$$\sum_{p=1}^M \left\{ \sum_{j=1}^m \left[\mu^{d_j} (X_p, B, C, W) - \mu_p^{d_j} (y) \right]^2 \right\} = \min_{B, C, W}, \quad (2.42)$$

где $\mu_p^{d_j} = \begin{cases} 1, & \text{если } d_j = d_p \\ 0, & \text{если } d_j \neq d_p \end{cases}$.

2.4.3. Задача оптимизации для объекта с непрерывным выходом.

Пусть для объекта типа (2.12) известно следующее:

а) интервалы изменения входов и выхода:

$$x_i \in [\underline{x}_i, \overline{x}_i], i = \overline{1, n}, y \in [\underline{y}, \overline{y}], \quad (2.43)$$

при этом выход является непрерывным;

б) выборка экспериментальных данных в виде M пар «входы-выход»:

$$\left\{ X_p, y_p \right\}, \text{ где } X_p = \left\{ x_1^p, x_2^p, \dots, x_n^p \right\} - \text{вектор значений входных пере-}$$

менных в p -й паре, $p = \overline{1, M}$ -

и необходимо синтезировать знания об объекте в виде системы нечетких логических высказываний типа (2.35), то есть в виде нечеткой базы знаний.

Тогда модель объекта с непрерывным выходом, которому соответствует нечеткая база знаний (2.35), можно представить в виде аналитических выражений (2.37), (2.24) и (2.32), или в более компактном виде:

$$y = F(X, B, C, W), \quad (2.44)$$

где F - оператор связи «входы-выход», что соответствует использованию соотношений (2.37), (2.24) и (2.32);

$X = (x_1, x_2, \dots, x_n)$ - вектор входных переменных;

$B = (b_1, b_2, \dots, b_q)$ и $C = (c_1, c_2, \dots, c_q)$ - векторы параметров настройки функций

принадлежности (2.24);

$W = (w_1, w_2, \dots, w_N)$ - вектор весов правил нечеткой базы знаний (2.35);

q - общее количество термов в (2.35);

N - общее количество строк в нечеткой базе знаний (2.35).

Синтезированная нечеткая база знаний должна удовлетворять ограничениям, которые аналогичны сформулированным при постановке предыдущей задачи. Это же касается и матрицы параметров (см. табл. 2.2), соответствующей данной нечеткой базе знаний. Исходя из этого, задача оптимизации для объекта с непрерывным выходом может быть сформулирована следующим образом:

найти такую матрицу (см. табл. 2.2), которая бы удовлетворяла ограничениям на диапазоны изменения параметров (B, C, W) , а также количество строк, и обеспечивала:

$$\sum_{p=1}^M \left[F(X_p, B, C, W) - y_p \right]^2 = \min_{B, C, W}. \quad (2.45)$$

Поставленные нелинейные задачи оптимизации можно решить при по-

мощи различных методов [46]. Но, как показано в дальнейших параграфах этой главы, генетические алгоритмы дают наилучшие результаты именно для решения задач извлечения оптимальных по объему нечетких баз знаний с оптимальными параметрами как для объектов с дискретным, так и с непрерывным выходом.

Прежде чем перейти к рассмотрению метода генетических алгоритмов, введем некоторые основные понятия из их теории.

2.5. Генетические алгоритмы оптимизации

В современных человеко-машинных системах возникает необходимость решать большой объем оптимизационных задач, среди которых попадаются очень сложные по своей сути. Применение для их решения традиционных классических методов оптимизации приводит к значительным затратам времени, а в многих случаях вообще безрезультатно. В связи с этим возникает необходимость разработки альтернативных методов, способных преодолевать сложности при нахождении оптимума. Это касается как усовершенствования классических, так и разработки принципиально новых методик.

Среди многих разработок, осуществленных в этом направлении, следует особо выделить новые стохастические методы оптимизации, называемые *эволюционными* или *генетическими* алгоритмами. Данные алгоритмы были сформулированы на основе наблюдения эволюционных процессов, постоянно происходящих в природе. Они представляют собой обобщение методов случайного поиска на тот случай, когда нахождение оптимального решения ведется одновременно с нескольких начальных точек. Это приводит к существенному уменьшению времени поиска оптимума, а также преодолеваются трудности, связанные с проблемами локального экстремума и «проклятия размерности» [84]. В научной литературе бывшего СССР одно из первых упоминаний о генетических алгоритмах встречается в работах известного специалиста в области

случайного поиска Л.А. Растригина [52,53].

Применение генетических алгоритмов показало, что при решении сложных практических задач эти методы имеют преимущество перед традиционными. Так, возможным применением генетических алгоритмов являются такие задачи, как составление расписаний, проектирование надежных систем, составление маршрутов движения, расположение оснащения, планирование транспортных перевозок и т.п.

В данном подразделе излагаются общие принципы построения генетических алгоритмов оптимизации и их применения к решению задачи извлечения нечетких баз знаний из экспериментальных данных.

2.5.1. Принципы построения.

Генетический алгоритм (ГА) представляет собой одну из разновидностей методик случайного поиска и основан на механизмах, напоминающих естественный отбор и размножение [94].

В отличие от существующих методик, ГА начинает работу с некоторого случайного набора начальных решений, носящего название *популяции*. Каждый элемент из популяции называется *хромосомой* и представляет собой некоторое решение поставленной задачи оптимизации в первом приближении. Хромосома изображается в виде последовательности символов определенной природы, например, бинарных.

ГА по своей сути является итерационным. Каждая итерация носит название *поколения* (или *генерации*). На протяжении выполнения определенного количества итераций происходит *эволюция* хромосом из начальной популяции. При этом в ходе каждой итерации хромосомы оцениваются с помощью некоторой меры соответствия, или так называемой *функции соответствия* (англ. fitness function). Создание каждого последующего поколения предусматривает формирование новых хромосом, так называемых *отпрысков*, которые получают либо путем *скрещивания* (англ. crossover) двух хромосом-родителей из те-

кущей популяции, либо путем случайного изменения (*мутации*) элементов одной хромосомы (*генов*). Для формирования новой популяции применяется отбор некоторых родителей и отпрысков согласно функции соответствия и удаления остальных с целью сохранения размера популяции постоянным.

Хромосомы, которым соответствует большее значение функции соответствия, имеют больше шансов быть выбранными (*выжить*). После выполнения определенного количества итераций алгоритм сходится к наилучшей хромосоме, которая будет представлять собой или оптимальное, или близкое к оптимальному (субоптимальное) решение.

Обозначим через $P(t)$ набор родительских хромосом, а через $C(t)$ - набор хромосом-отпрысков из текущей генерации t . Тогда общую структуру генетического алгоритма можно представить следующим образом [55]:

Процедура: Генетический алгоритм

начало

$t:=0$;

Задать начальное значение $P(t)$;

Оценить $P(t)$ с помощью функции соответствия;

пока (не достигнуто условие завершения)

 Скрещивать $P(t)$ чтобы получить $C(t)$;

 Оценить $C(t)$ с помощью функции соответствия ;

 Выбрать $P(t+1)$ из $P(t)$ и $C(t)$;

$t:=t+1$;

конец

конец.

Таким образом, при выполнении генетического алгоритма используются два основных вида операций:

- 1) генетические операции: скрещивание и мутация;
- 2) эволюционная операция: отбор.

Генетические операции напоминают процесс наследования генов при

создании нового отпрыска в каждом поколении. Эволюционная операция осуществляет переход от одной популяции к следующей и напоминает процесс Дарвиновской эволюции.

2.5.2. Основные операции.

Операция скрещивания. Скрещивание - главная генетическая операция. Она выполняется с участием двух хромосом-родителей и в результате ее выполнения создается отпрыск, который представляет собой комбинацию особенностей обеих родителей.

Продемонстрируем выполнение данной операции на простейшем примере. Вначале для всех хромосом из популяции необходимо установить (проще всего - выбрать случайным образом) некоторую точку (англ. cut-point), относительно которой будет проводиться скрещивание. Потом выбирается первая пара хромосом-родителей, и путем комбинирования сегмента первой родительской хромосомы, находящегося по левую сторону от выбранной точки скрещивания, с сегментом второй родительской хромосомы, который находится по правую сторону от точки скрещивания, создается хромосома-отпрыск. Данный процесс наглядно представлен на рис. 2.4.

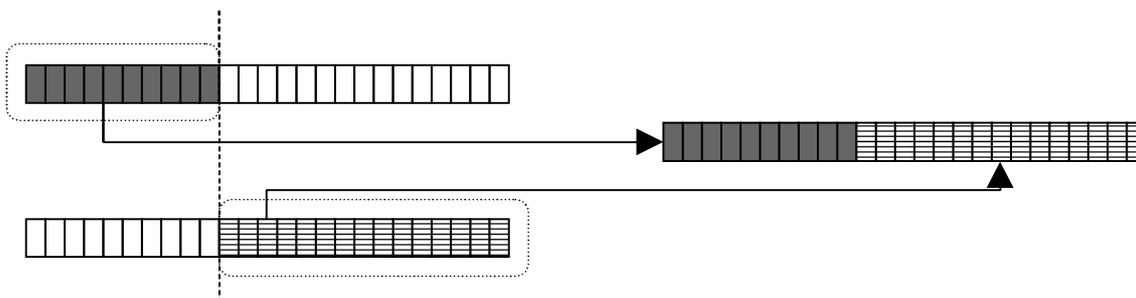


Рис. 2.4. Операция скрещивания.

Подобная методика проведения операции скрещивания очень эффективна при применении к хромосомам, представляющим собой битовые строки. Кроме того, производительность всего генетического алгоритма в первую очередь зависит от производительности способа проведения операции скрещивания.

Процент отпрысков, которые продуцируются на каждой итерации, назы-

вается коэффициентом скрещивания (p_c). Число, которое равняется произведению p_c на количество хромосом в популяции, равняется количеству полученных отпрысков. Чем больше значение коэффициента p_c , тем больше областей поискового пространства (или пространства решений) есть возможность исследовать и тем меньше шанс попадания в локальный экстремум. Однако при очень большом значении p_c возрастают затраты времени вычислений при исследовании бесперспективных областей.

Операция мутации. В результате выполнения операции мутации в различных хромосомах определенным образом происходят случайные изменения. Простейший вариант мутации состоит в случайном изменении одного или нескольких генов. В ГА мутация играет важную роль для:

а) восстановления генов, которые были удалены из популяции в ходе операции отбора и теперь могут быть исследованы в новом контексте;

б) получения генов, которые не были представлены в начальной популяции.

Интенсивность мутации определяется коэффициентом мутации (p_m), характеризующим процент генов от общего их количества, которые на данной итерации поддаются мутации. При небольшом значении этого коэффициента многие гены, которые могли бы быть полезными, никогда не будут рассмотрены, однако довольно большое значение p_m может привести к значительным случайным возбуждениям, в результате чего отпрыски будут непохожими на родителей и алгоритм утратит возможность поиска оптимальной хромосомы эволюционным путем.

2.6. Извлечение нечетких баз знаний с помощью генетических алгоритмов

Поставленные в пп. 2.5.2. и 2.5.3. задачи оптимизации могут быть решены различными методами, наиболее универсальным из которых является наиско-

рейший спуск [11]. Однако недостаток этого метода состоит в том, что найденный оптимум может оказаться локальным. Поэтому для нахождения глобального оптимума следует осуществлять перебор начальных точек, что приводит к значительным затратам машинного времени. Генетические алгоритмы оптимизации помогают устранить указанные недостатки.

2.6.1. Кодирование нечеткой модели.

Чтобы реализовать генетический алгоритм извлечения нечеткой базы знаний, параметры синтезированной модели нужно определенным образом закодировать. С этой целью сведем неизвестные параметры B, C, W искомой матрицы (см. табл. 2.2) в один вектор:

$$S = (B, C, W) = \left(b_{11}, c_{11}, \dots, b_{l_1}, c_{l_1}, \dots, b_{n1}, c_{n1}, \dots, b_{nl_n}, c_{nl_n}, w_1, w_2, \dots, w_N \right), \quad (2.46)$$

где l_i - количество термов-оценок входной переменной x_i , $l_1 + l_2 + \dots + l_n = q$, $i = \overline{1, n}$;

q - общее количество термов в (2.35);

N - общее количество строк в нечеткой базе знаний (2.35).

Вектор S единственным образом определяет некоторую нечеткую модель $F(X, B, C, W)$, и наоборот, любая модель $F(X, B, C, W)$ однозначно определяет некоторый вектор S , причем это касается модели нелинейного объекта как с дискретным, так и с непрерывным выходом. Поэтому вектор S можно рассматривать в качестве кода нечеткой модели $F(X, B, C, W)$ и поставить ему в соответствие хромосому, которая воплощает искомую матрицу параметров (см. табл. 2.2) и определяется строкой, показанной на рис. 2.5.

На данном рисунке r_{jp} - код правила ЕСЛИ-ТО с номером jp , $p = \overline{1, k_j}$,

$$j = \overline{1, m}.$$

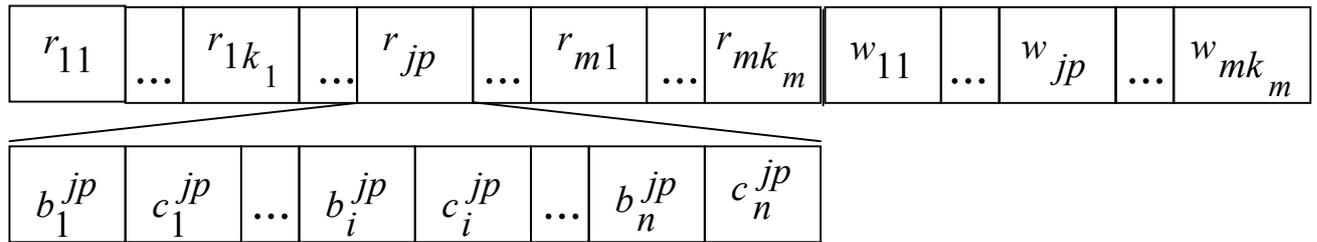


Рис. 2.5. Кодирование матрицы параметров.

2.6.2. Скрещивание хромосом-моделей.

Операция скрещивания - основная операция генетического алгоритма, от которой в первую очередь зависит его производительность. Таким образом, чем проще данная операция будет выполняться, тем большей будет производительность генетического алгоритма.

Технику проведения операции скрещивания рассмотрим на примере. Пусть в качестве родительских хромосом взято две хромосомы S_1 и S_2 . Провести скрещивание этих хромосом означает получить две хромосомы-отпрыска Ch_1 и Ch_2 путем обмена генов хромосом-родителей относительно выбранной точки (точек) скрещивания (рис. 2.6).

На рис. 2.6 показан один из простых вариантов проведения операции скрещивания. Он состоит в обмене частей хромосом-родителей в каждом правиле r_{jp} ($p = \overline{1, k_j}$, $j = \overline{1, m}$) и векторе весов правил. При этом точки обмена случайным образом выбираются по одной в каждом правиле и одна - в векторе весов правил.

Введем следующие обозначения:

$w_{jp}^{S_1}$ - jp -тый вес правила в родительской хромосоме S_1 ;

$w_{jp}^{S_2}$ - jp -тый вес правила в родительской хромосоме S_2 ;

$w_{jp}^{Ch_1}$ - jp -тый вес правила в хромосоме-отпрыске Ch_1 ;

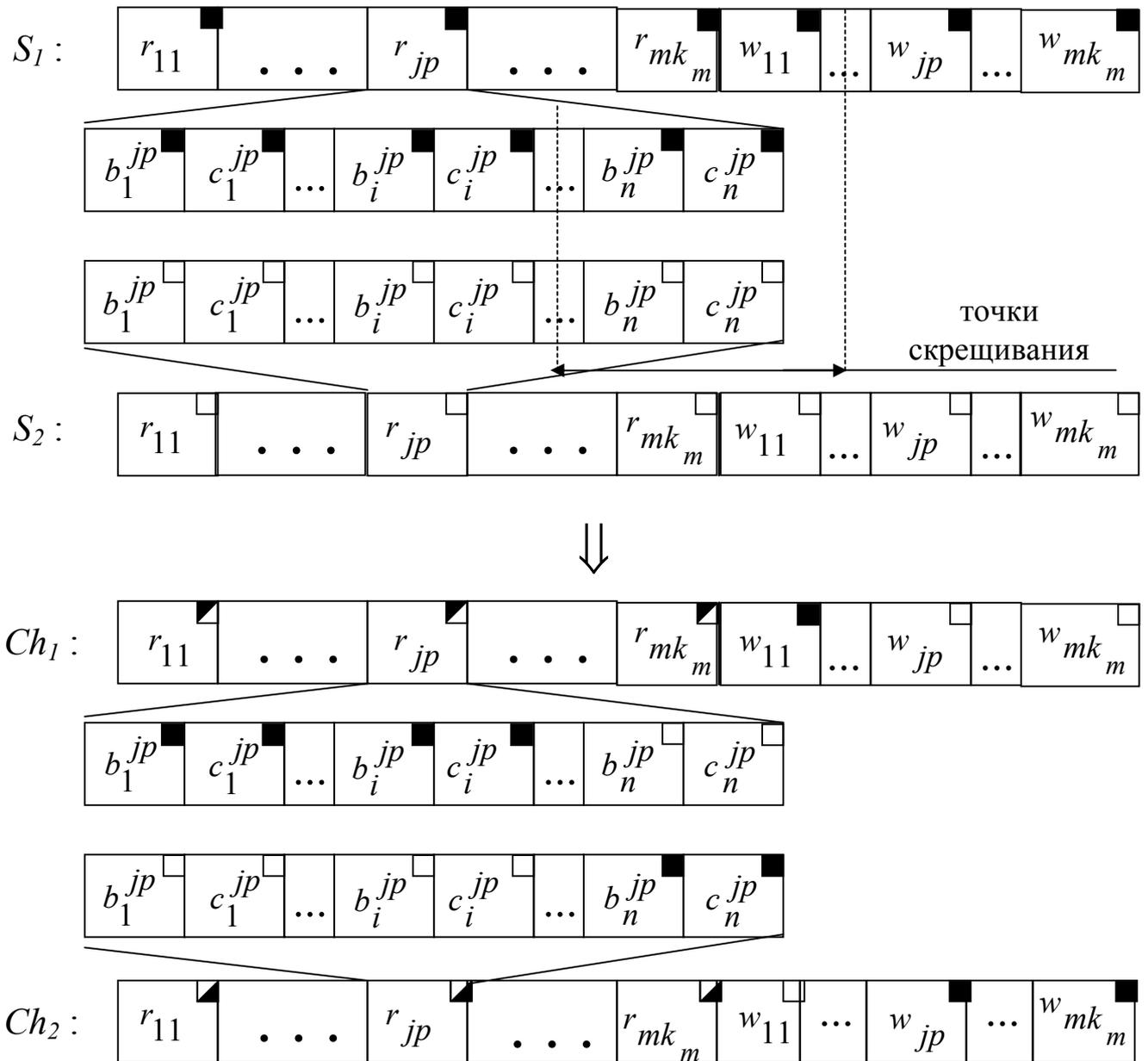


Рис. 2.6. Операция скрещивания (■, □ - символы родителей, ◐, ◑ - символы отпрысков)

$w_{jp}^{Ch_2}$ - jp -тый вес правила в хромосоме-отпрыске Ch_2 , $p = \overline{1, k_j}$, $j = \overline{1, m}$;

$b_{ijp}^{S_1}, c_{ijp}^{S_1}$ - ijp -тые параметры b, c в родительской хромосоме S_1 ;

$b_{ijp}^{S_2}, c_{ijp}^{S_2}$ - ijp -тые параметры b, c в родительской хромосоме S_2 ;

$b_{ijp}^{Ch_1}, c_{ijp}^{Ch_1}$ - ijp -тые параметры b, c в хромосоме-отпрыске Ch_1 ,

$b_{ijp}^{Ch_2}, c_{ijp}^{Ch_2}$ - ijp -тые параметры b, c в хромосоме-отпрыске Ch_2 .

Алгоритм операции скрещивания двух хромосом-родителей S_1 и S_2 , в результате которой появляются отпрыски Ch_1 и Ch_2 , имеет следующий вид:

ШАГ 1. Сгенерировать $N + 1$ случайных целых чисел z_{jp} :

$1 \leq z_{jp} < n - 1$, где jp - номер текущей строки-правила, $p = \overline{1, k_j}$, $j = \overline{1, m}$, n -

количество входных переменных x_i , $i = \overline{1, n}$;

$1 \leq z_{N+1} < N$, где N - общее количество строк-правил и, соответственно,

весов правил в нечеткой базе знаний (2.35).

ШАГ 2. Осуществить обмен генов в соответствии с установленными значениями точек обмена z_{jp} :

$$b_{ijp}^{Ch_1} = \begin{cases} b_{ijp}^{S_1}, & i \leq z_{jp} \\ b_{ijp}^{S_2}, & i > z_{jp} \end{cases}, \quad b_{ijp}^{Ch_2} = \begin{cases} b_{ijp}^{S_2}, & i \leq z_{jp} \\ b_{ijp}^{S_1}, & i > z_{jp} \end{cases}, \quad (2.47)$$

$$c_{ijp}^{Ch_1} = \begin{cases} c_{ijp}^{S_1}, & i \leq z_{jp} \\ c_{ijp}^{S_2}, & i > z_{jp} \end{cases}, \quad c_{ijp}^{Ch_2} = \begin{cases} c_{ijp}^{S_2}, & i \leq z_{jp} \\ c_{ijp}^{S_1}, & i > z_{jp} \end{cases}, \quad (2.48)$$

$$w_{jp}^{Ch_1} = \begin{cases} w_{jp}^{S_1}, & jp \leq z_{N+1} \\ w_{jp}^{S_2}, & jp > z_{N+1} \end{cases}, \quad w_{jp}^{Ch_2} = \begin{cases} w_{jp}^{S_2}, & jp \leq z_{N+1} \\ w_{jp}^{S_1}, & jp > z_{N+1} \end{cases}, \quad (2.49)$$

$$p = \overline{1, k_j}, \quad j = \overline{1, m}, \quad i = \overline{1, n}.$$

2.6.3. Мутация генов хромосом-моделей.

Каждый элемент вектора S может «подвергнуться» мутации с некоторой вероятностью p_m . Если операцию мутации элемента S обозначить как $Mu(s)$,

то для элементов хромосомы будет справедливо:

$$Mu(w_{jp}): w_{jp} = RANDOM(\underline{w}, \bar{w}), \quad (2.50)$$

$$Mu(b_{ijp}): b_{ijp} = RANDOM(\underline{x}_i, \bar{x}_i), \quad (2.51)$$

$$Mu(c_{ijp}): c_{ijp} = RANDOM(\underline{c}_i, \bar{c}_i), \quad (2.52)$$

где $\underline{w}(\bar{w})$ - нижняя (верхняя) граница интервала возможных значений весов правил, $[\underline{w}, \bar{w}] \in [0,1]$;

$[\underline{c}_i, \bar{c}_i]$ - интервал возможных значений коэффициента концентрации-растяжения функций принадлежности, которые соответствуют термам-оценкам входной переменной x_i , $[\underline{c}_i, \bar{c}_i] \in (0, +\infty)$, $i = \overline{1, n}$;

$RANDOM(\underline{\xi}, \bar{\xi})$ - операция определения случайного числа, равномерно распределенного на интервале $[\underline{\xi}, \bar{\xi}]$.

Таким образом, можно сформулировать алгоритм проведения операции мутации:

ШАГ 1. Для каждого элемента s вектора S (2.46) сгенерировать случайное число $z = RANDOM([0,1])$.

ЕСЛИ $z > p_m$, то данный элемент мутации не подвергать

ИНАЧЕ перейти на ШАГ 2.

ШАГ 2. Совершить операцию мутации элемента $s \in S$ в соответствии с формулами (2.50)-(2.52).

2.6.4. Функции соответствия хромосом.

Функцию соответствия хромосомы S обозначим через $FF(S)$ (от англ. fitness function). В качестве такой функции вполне можно использовать критерий оптимизации, взятый с знаком «минус». Таким образом, для нечетких мо-

делей $\mu^{d_j}(y) = \mu^{d_j}(X, B, C, W)$ объектов с дискретным выходом функция соответствия хромосомы S (2.46), полученная из критерия (2.42), будет иметь вид:

$$FF(S) = - \sum_{l=1}^M \left[\sum_{j=1}^m \left[\mu^{d_j}(X^l, B, C, W) - \mu^{d_j}(X^l) \right]^2 \right]. \quad (2.53)$$

Для нечетких моделей $F(X, B, C, W)$ объектов с непрерывным выходом функция соответствия хромосомы S (2.46), полученная из критерия (2.45), будет иметь вид:

$$FF(S) = - \sum_{l=1}^M \left[F(X^l, B, C, W) - y^l \right]^2. \quad (2.54)$$

Знак «минус» нужен для сохранения физического смысла функции соответствия, то есть чем хуже нечеткая модель описывает выборку экспериментальных данных, тем меньшей должна быть функция соответствия такой модели.

2.6.5. Выбор хромосом-родителей.

В соответствии с принципами генетических алгоритмов, выбор родителей для операции скрещивания должен осуществляться не случайно, а в зависимости от значений функций соответствия хромосом в популяции: чем большее значение функции соответствия некоторой хромосомы, тем большей должна быть вероятность того, что данная хромосома «даст потомство».

С целью определения хромосом-родителей можно воспользоваться следующим способом: каждой хромосоме S_i из популяции ставится в соответствие

число p_i такое, что: $p_i \geq 0$, $\sum_{i=1}^K p_i = 1$, $FF(S_i) > FF(S_j) \Rightarrow p_i > p_j$, где K - ко-

личество хромосом в популяции. При этом набор чисел p_i можно интерпретировать как закон распределения дискретной случайной величины.

С помощью ряда чисел p_i , хромосомы-родители для операции скрещивания определяются по следующему алгоритму:

ШАГ 1. Отложить ряд чисел p_i на горизонтальную ось (рис. 2.7).

ШАГ 2. Сгенерировать случайное число z (рис. 2.7), имеющее равномерный закон распределения на интервале $[0, 1]$.

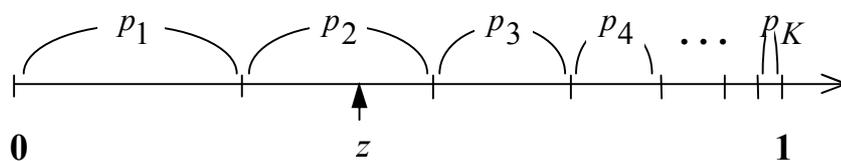


Рис. 2.7. Принцип выбора родительской хромосомы.

ШАГ 3. Выбрать в качестве родительской ту хромосому S_i , которая соответствует подинтервалу p_i , в который попадает число z . Так, на рис. 2.7. случайное число z в качестве родительской определяет хромосому S_2 .

ШАГ 4. Повторить ШАГИ 1-3 для определения второй родительской хромосомы.

Для определения ряда чисел p_i целесообразно воспользоваться формулой:

$$p_i = \frac{FF'(S_i)}{\sum_{j=1}^K FF'(S_j)}, \quad (2.55)$$

где $FF'(S_i) = \max_{j=1, K} |FF(S_j)| + FF(S_i)$.

2.6.6. Инициализация популяции хромосом.

Хромосомы начальной популяции можно определять разными способами. Один из таких способов состоит в том, что популяция задается случайным образом в соответствии с формулами:

$$w_i^0 = RANDOM\left(\left[\underline{w}_i, \overline{w}_i\right]\right), \quad (2.56)$$

$$b_i^0 = RANDOM\left(\left[\underline{x}_i, \overline{x}_i\right]\right), \quad (2.57)$$

$$c_i^0 = RANDOM\left(\left[\underline{c}_i, \overline{c}_i\right]\right). \quad (2.58)$$

2.6.7. Генетический алгоритм извлечения нечеткой базы знаний.

При выполнении генетического алгоритма на каждой его итерации размер популяции будет увеличиваться на $p_c \cdot K$ хромосом-отпрысков, где p_c - коэффициент скрещивания. С целью сохранения постоянного размера популяции K возникает необходимость перед каждой следующей итерацией удалять $p_c \cdot K$ наихудших согласно значениям функций соответствия хромосом.

Учитывая вышеописанные формальности, можно сформулировать генетический алгоритм извлечения нечеткой базы знаний из экспериментальных данных:

ШАГ 1. Сформировать начальную популяцию согласно п. 2.7.6.

ШАГ 2. Вычислить значения функций соответствия $FF(S_i)$, $i = \overline{1, K}$ для каждой хромосомы, используя соотношения (2.53)-(2.54).

ШАГ 3. По алгоритму из п. 2.7.5 определить пары хромосом-родителей в количестве $\frac{p_c \cdot K}{2}$.

ШАГ 4. Провести операцию скрещивания каждой пары хромосом-родителей по алгоритму, приведенному в п. 2.7.2.

ШАГ 5. Осуществить мутацию полученных хромосом-отпрысков по алгоритму, приведенному в п. 2.7.3, учитывая коэффициент мутации (вероятность мутации) p_m .

ШАГ 6. Удалить из полученной популяции размером $K + p_c \cdot K$ хромосом $p_c \cdot K$ ее членов, которые имеют наихудшие значения функций соответствия $FF(S_i)$ (соотношения (2.53)-(2.54)).

ШАГ 7. ЕСЛИ получена хромосома S_i , для которой $FF(S_i) = 0$ (максимум функции соответствия), ТО - конец алгоритма, ИНАЧЕ - перейти на ШАГ 8.

ШАГ 8. ЕСЛИ заданное число итераций алгоритма не исчерпано, ТО перейти на ШАГ 2, ИНАЧЕ - конец алгоритма. При этом хромосома, которая имеет наибольшее значение функции соответствия $FF(S_i)$, будет представлять собой найденное субоптимальное решение.

Таким образом, полученная в результате выполнения вышеописанного генетического алгоритма хромосома будет соответствовать матрице параметров (см. табл. 2.2) нечеткой базы знаний, представляющей собой нечеткую модель исследуемого объекта. Если при этом решение является субоптимальным, то такую нечеткую модель можно улучшить путем дальнейшей настройки. Эту дополнительную настройку можно осуществлять, применяя как традиционные методы оптимизации, так и генетические алгоритмы в варианте, предложенном в работах [26,55], но даже такая дополнительная комбинация методов поиска оптимума дает лучшие результаты на пути к нахождению его глобального значения по сравнению с любой другой.

ГЛАВА 3

НАСТРОЙКА НЕЧЕТКИХ БАЗ ЗНАНИЙ

3.1. Постановка задачи

Как вытекает из предыдущей главы, путем применения генетического алгоритма оптимизации для обработки экспериментальной информации возникает возможность построить нечеткую базу знаний, которая будет соответствовать математической модели того ли иного объекта идентификации. Но, как правило, получить идеальную модель, то есть такую, которая бы характеризовалась максимумом функции соответствия, довольно сложно, ведь подбор переменных параметров модели проводится путем случайного поиска, который заложен в основу генетических алгоритмов. Поэтому полученная в результате процедуры извлечения нечеткая база знаний является, как правило, субоптимальной, иначе говоря, этап извлечения знаний соответствует этапу грубой настройки нечеткой базы знаний. Таким образом, возникает необходимость проведения следующего этапа - тонкой настройки переменных параметров. Методам тонкой настройки посвящены работы [26,55-57,64,87,90]. В данной главе предлагается метод с применением такого средства интеллектуальных технологий, как нейронные сети. Результаты исследований в этом направлении представлены в работах [60,67].

Следует отметить, что нейронные сети и нечеткая логика - это принципиально разные математические конструкции. Вместе с тем они оба могут применяться в качестве универсальных аппроксиматоров сложных (нелинейных) функциональных зависимостей. При этом главным преимуществом нейронных сетей является их способность к так называемому обучению. Данная способность реализуется с помощью специально разработанных алгоритмов, среди которых наиболее популярно правило «обратного распространения ошибки» [93]. Также к преимуществам следует отнести ту особенность, что обучение нейрон-

ной сети не требует никакой априорной информации о структуре искомой функциональной зависимости - нужна лишь обучающая выборка в виде экспериментальных пар «входы-выход». Основным же недостатком нейронных сетей состоит в том, что обученная нейронная сеть, которую можно считать аналогом графа со взвешенными дугами, не поддается содержательной интерпретации.

Преимуществом же нечеткой логики, как вытекает из предыдущих глав, является возможность использования экспертных знаний о структуре объекта в виде лингвистических высказываний типа ЕСЛИ <входы> ТО <выход>. Однако такой особенностью, как механизм обучения, аппарат нечеткой логики не обладает. Поэтому результаты нечеткого логического вывода в значительной степени зависят от характера функций принадлежности, которыми формализуются нечеткие термы типа «низкий», «высокий», «средний», «выше среднего» и т. п.

В связи с вышеизложенным возникает вопрос: а можно ли объединить преимущества этих двух структур, получив таким образом граф, который воплощает в себе экспертные знания об исследуемом объекте и при этом обладает возможностью тонкой настройки переменных параметров, то есть механизмом обучения? Решению этой проблемы и посвящена данная глава.

3.2. Нейронные модели многомерных зависимостей

3.2.1. Нейронная сеть как универсальный аппроксиматор.

Рассмотрим нелинейный объект типа:

$$y = f(x_1, x_2, \dots, x_n), \quad (3.1)$$

где y - ВЫХОД, x_1, x_2, \dots, x_n - ВХОДЫ.

Покажем, что такой объект можно промоделировать при помощи нейронной сети.

Формальная модель нейронной сети представляет собой совокупность формальных нейронов, которые определенным образом соединены между собой и внешней средой. На рис. 3.1 представлена модель формального нейрона, которая взята из работы [16].

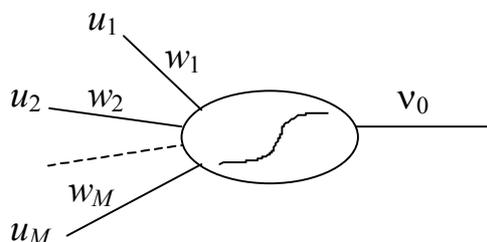


Рис. 3.1. Формальный нейрон.

В простейшем случае, как показано в работе [16], нейрон можно рассматривать как элемент с несколькими входами и одним выходом, предназначенный для выполнения нелинейного параметрического преобразования суммарного взвешенного входного сигнала в скалярную величину:

$$v_0 = \psi \left(\sum_{i=1}^M u_i w_i \right), \quad (3.2)$$

где v_0 - выходной сигнал;

u_i - входной i -й сигнал;

w_i - вес i -го входа;

M - количество входов;

$\psi(\bullet)$ - характеристическая функция (функция активации) нейрона.

В качестве функции активации обычно используется сигмоидная функция типа:

$$\psi(\tau) = \frac{1}{1 + e^{-k\tau}}, \quad (3.3)$$

где k - коэффициент сжатия-растяжения функции вдоль оси абсцисс [16].

Особенностью данной функции является то, что ее производная легко представляется в аналитическом виде:

$$\psi'(\tau) = \psi(\tau)(1 - \psi(\tau)), \quad (3.4)$$

что очень важно для процесса настройки сети.

Наиболее общей моделью нейронной сети можно считать модель типа *многослойный персептрон* [1,16]. Именно эту модель было решено взять за основу для идентификации нелинейных объектов. На рис. 3.2 представлена модель многослойного персептрона, содержащего три слоя.

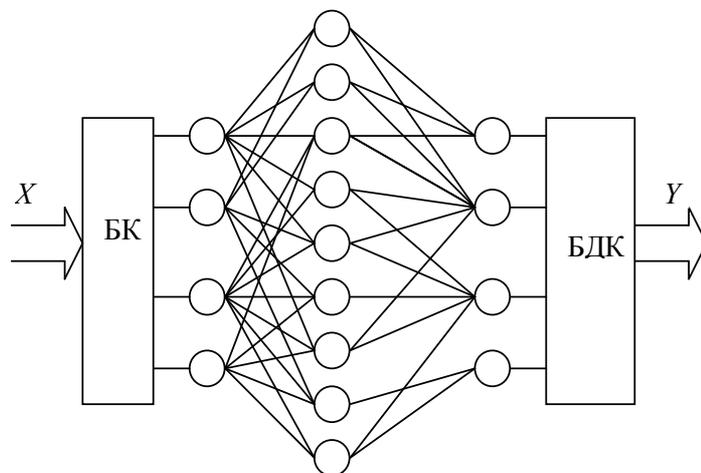


Рис. 3.2. Многослойный персептрон.

В моделях такого типа первый слой нейронов служит для ввода входных сигналов, последний - для вывода выходных сигналов, а внутренний (один или несколько) - для обработки входной информации и сохранения информации о внутренней структуре объекта, который моделируется. Межэлементные связи в такой сети организовываются лишь между нейронами соседних слоев: отдельно взятый нейрон может соединяться с одним, несколькими или всеми нейронами из соседнего слоя (в последнем случае такая нейронная сеть называется *полно-*

связной). При этом на входы того или иного нейрона поступают сигналы от нейронов предыдущего слоя, а выходной сигнал нейрона передается на входы нейронов следующего слоя.

Поскольку входные и выходные величины объекта идентификации в общем случае могут иметь различную природу, с целью их согласования с сигналами, которые способна обрабатывать нейронная сеть, на входе и на выходе следует использовать блоки кодирования (БК) и соответственно декодирования (БДК) информации. В простейшем случае эти блоки могут выполнять преобразование чисел, например, из десятичной системы счисления в двоичную (на входе) и из двоичной системы счисления в десятичную (на выходе).

Каждая межнейронная связь в нейронной сети характеризуется определенным коэффициентом усиления (ослабления) сигнала, или иначе говоря - весом. Веса межнейронных связей и являются переменными параметрами модели объекта идентификации, которые подлежат настройке. Эти веса удобно представлять в виде матриц:

$$W^{\mu} = \left\{ w_{ij}^{\mu} \right\},$$

$$\mu = \overline{1, K}, \quad i = \overline{1, N_{\mu}}, \quad j = \overline{1, N_{\mu-1}}, \quad (3.5)$$

где μ - номер матрицы межнейронных связей;

w_{ij}^{μ} - вес связи между j -тым нейроном предыдущего слоя и i -тым нейроном следующего слоя для μ -той матрицы связей;

N_{μ} - количество нейронов μ -го слоя;

$N_{\mu-1}$ - количество нейронов $(\mu-1)$ -го слоя.

Как известно [1,16], принцип функционирования многослойного персептрона состоит в следующем. На нейроны входного слоя поступает вектор вход-

ных величин (переменных) $X = \{x_1, x_2, \dots, x_n\}$, который предварительно кодируется в БК. Принято считать, что входной слой нейронов служит лишь для приема входных сигналов сети и их передачи в следующие слои, а функции активации для нейронов этого слоя не вычисляются. В каждом же следующем слое осуществляется вычисление выходных сигналов (значений функций активации) для каждого нейрона с использованием известных выходов нейронов предыдущего слоя, которые являются одновременно входными сигналами для нейронов текущего слоя, и значений весов межнейронных связей. В результате таких вычислений в каждом слое информация, полученная из выходного слоя сети, декодируется в БДК и выдается как выход нелинейного объекта:

$$Y = F(W^1, W^2, \dots, W^K, x_1, x_2, \dots, x_n). \quad (3.6)$$

Учитывая все вышеприведенные предположения и формализмы, сформулируем задачу идентификации нелинейных объектов с помощью нейронных сетей.

Дано: (X^P, \hat{y}^P) , $p = \overline{1, P}$ - выборка экспериментальных данных (обучающая выборка) в виде P пар «входы-выход», где

$X^P = \{x_1^p, x_2^p, \dots, x_n^p\}$ - вектор входных переменных;

\hat{y}^P - выходная переменная.

Необходимо: определить такие параметры (3.5) нейронной сети, которые бы обеспечивали минимальное отклонение значений выходных величин модели от эталонных значений:

$$\sum_{p=1}^P (F(W^1, W^2, \dots, W^K, x_1^p, x_2^p, \dots, x_n^p) - \hat{y}^p)^2 \rightarrow \min_{W^\mu}. \quad (3.7)$$

3.2.2. Обучение нейронной сети.

Исходя из постановки задачи, с целью идентификации исследуемого объекта нейронной сетью следует определить ее неизвестные параметры - веса межнейронных связей. Иначе говоря, необходимо провести обучение нейронной сети. Данную процедуру целесообразно выполнить с использованием вышеупомянутого метода обратного распространения ошибки (back-propagation method) [16,93].

Разобьем каждую итерацию процедуры обучения на два этапа - прямой и обратный ход и сформулируем соответствующие алгоритмы.

Алгоритм прямого хода

ШАГ 1. Вычислить суммарный взвешенный входной сигнал u_j для каждого нейрона текущего слоя μ :

$$u_j = \sum_{i=1}^{N_{\mu-1}} v_i w_{ij} . \quad (3.8)$$

ШАГ 2. Вычислить выходной сигнал v_j для каждого нейрона текущего слоя:

$$v_j = \frac{1}{1 + e^{-ku_j}} . \quad (3.9)$$

ШАГ 3. Если текущий слой не является выходным, то перейти к следующему слою и перейти на ШАГ 1.

ШАГ 4. Вычислить погрешность Q нейронной сети:

$$Q = \frac{1}{2} \sum_{j=1}^{N_K} (v_j - \hat{v}_j)^2, \quad (3.10)$$

где N_K - количество нейронов в выходном слое,

\hat{v}_j - эталонное значение выхода j -го нейрона выходного слоя.

КОНЕЦ АЛГОРИТМА.

Алгоритм обратного хода

ШАГ 1. Определить скорость изменения погрешности при изменении выходного сигнала для каждого нейрона в выходном слое (EA):

$$EA_j^K = \frac{\partial Q}{\partial v_j} = (\hat{v}_j - v_j). \quad (3.11)$$

ШАГ 2. Определить скорость изменения погрешности при изменении суммарного входного сигнала для каждого нейрона в текущем слое (EI):

$$EI_j^\mu = \frac{\partial Q}{\partial u_j} = \frac{\partial Q}{\partial v_j} \frac{\partial v_j}{\partial u_j} = EA_j^\mu v_j (1 - v_j). \quad (3.12)$$

ШАГ 3. Определить скорость изменения погрешности при изменении веса входной дуги каждого нейрона текущего слоя (EW):

$$EW_{ij}^\mu = \frac{\partial Q}{\partial w_{ij}} = \frac{\partial Q}{\partial u_j} \frac{\partial u_j}{\partial w_{ij}} = EI_j^\mu v_i. \quad (3.13)$$

ШАГ 4. Определить скорость изменения погрешности при изменении активности нейрона предыдущего слоя (EA):

$$EA_i^{\mu-1} = \frac{\partial Q}{\partial v_i} = \sum_{j=1}^{N_\mu} \left(\frac{\partial Q}{\partial u_j} \frac{\partial u_j}{\partial v_i} \right) = \sum_{j=1}^{N_\mu} EI_j^\mu w_{ij} \quad (3.14)$$

ШАГ 5. Провести модификацию межнейронных связей по правилу градиента:

$$w_{ij}[t+1] = w_{ij}[t] + \gamma EW_{ij}^\mu, \quad (3.15)$$

где t - номер итерации (шага обучения);

γ - скорость обучения (шаг итерации).

Перейти к следующему слою.

ШАГ 6. Если текущий слой - не входной, перейти на ШАГ 2.

КОНЕЦ АЛГОРИТМА.

Обучение нейронной сети по данным алгоритмам продолжается до тех пор, пока не будет достигнута приемлемая погрешность.

3.2.3. Компьютерное моделирование.

С целью исследования возможности идентификации нелинейных объектов с помощью нейронных сетей был проведен ряд компьютерных экспериментов с эталонными нелинейными зависимостями, заданными аналитически. В этих экспериментах обучающая выборка пар «входы-выход», с помощью которой определялись параметры нейронной сети, генерировалась из заданной исследуемой модели-эталона. По окончании процесса обучения проводилось сравнения результатов, которые выдавались полученной моделью нелинейного объекта - наученной нейронной сетью, с эталонными.

Эксперимент 1

В качестве нелинейного объекта-эталона примем следующую зависимость:

$$y = f(x) = \frac{3(4x-2)(4x-3)(4x-3.7)(4x-1.3)(4x-0.2)}{40}, \quad x \in [0,1]. \quad (3.16)$$

Геометрический образ такого объекта показан на рис. 3.3 (а, б) сплошной линией.

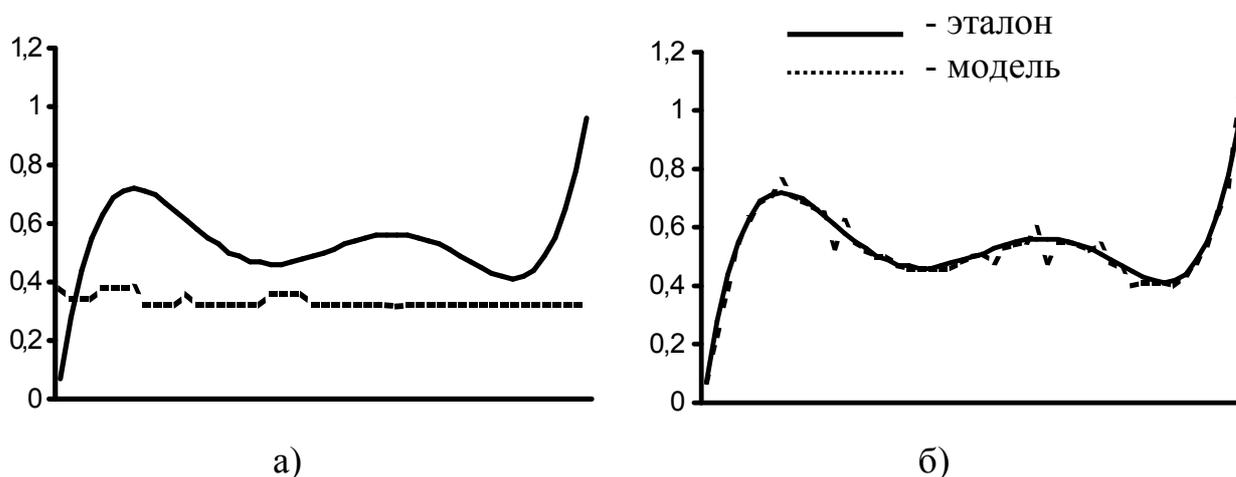


Рис. 3.3. Геометрические образы характеристик модели и эталона:
а - до обучения сети-аппроксиматора;
б - после обучения сети-аппроксиматора.

В качестве модели данного объекта используем соответствующую нейронную сеть-аппроксиматор, изображенную на рис. 3.4. Данная сеть является полносвязной, то есть обладает свойством полной взаимосвязи нейронов соседних слоев при отсутствии связей в слое.

Для моделирования объекта (3.16) установим следующую структуру сети-аппроксиматора: первый (входной) слой – 7 нейронов (7 разрядов двоичного кода входной переменной); второй (внутренний) слой (память сети) – 10 нейронов; третий (выходной) слой – 7 нейронов (7 разрядов двоичного кода выходной переменной).

Перед началом обучения сети-аппроксиматора веса его межнейронных связей определялись случайным образом. В связи с этим начальный вид исследуемой модели, показанный пунктиром на рис. 3.3 (а), был совсем не похож на эталон. После проведения обучения сети была получена модель, геометриче-

ский образ характеристики которой в сравнении с эталоном показан на рис. 3.3 (б).

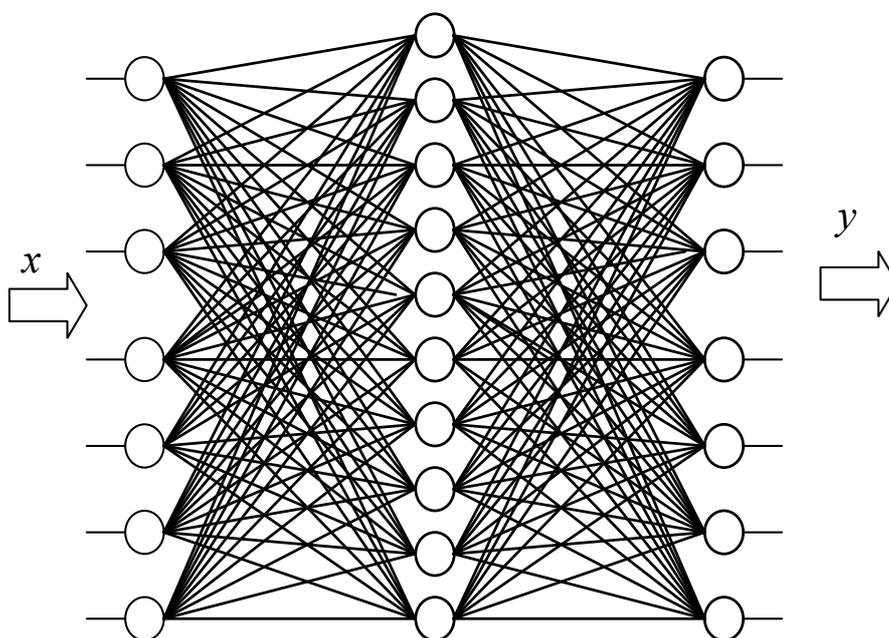


Рис. 3.4. Нейронная сеть-аппроксиматор характеристики объекта (3.16).

В табл. 3.1 и 3.2 приведены весовые коэффициенты сети, которые установились после ее обучения.

Таблица 3.1

Веса межнейронных связей входного (a_i)
и внутреннего (b_j) слоев

	a_1	a_2	a_3	a_4	a_5	a_6	a_7
b_1	0.34	-2.87	-4.53	1.13	-2.61	8.51	-10.04
b_2	0.74	-0.53	-0.45	-7.35	-3.15	5.13	-8.25
b_3	0.62	1.23	-2.78	-2.14	-8.37	5.11	-5.66
b_4	0	1.88	0.51	-0.40	-9.87	6.00	-4.98
b_5	0.65	-1.75	-0.97	3.61	-9.41	5.60	-3.72
b_6	0.31	-2.09	-2.03	-9.48	4.95	6.20	-5.73
b_7	0.44	3.87	-3.11	-3.35	-2.74	-1.75	-2.25
b_8	0.50	-0.07	3.61	5.53	-5.47	3.29	-1.32
b_9	0.94	-3.62	-5.26	2.12	-3.91	-1.0	-2.85
b_{10}	0.54	0.99	0.59	0.81	0.66	0.67	0.18

Таблица 3.2

Веса межнейронных связей внутреннего (b_j) и выходного (c_i) слоев

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b	b_{10}
c_1	0,40	-2,11	-2,15	0,58	-0,26	1,28	1,93	0,04	0,26	-0,92
c_2	1,95	1,71	2,38	0,26	-2,02	-2,70	1,38	0,93	0,01	-0,62
c_3	1,50	-2,97	-0,04	0,60	-1,27	0,21	1,19	1,26	2,01	-1,02
c_4	1,25	1,31	1,85	4,42	-0,41	-5,47	-5,32	-4,72	-0,16	3,57
c_5	3,69	4,13	1,48	-5,36	-5,94	-2,60	3,86	4,53	3,29	-2,16
c_6	4,34	6,79	2,34	-2,68	-3,59	-6,17	-4,65	0,34	-0,05	4,66
c_7	-4,52	-7,71	-3,59	3,00	4,61	6,27	2,54	0	-3,27	4,19

Эксперимент 2

Рассмотрим нелинейный объект-эталон, характеристика которого задана следующей аналитической зависимостью:

$$y = \frac{1}{40} (2z - 0.9)(7z - 1)(17z - 19)(15z - 2),$$

$$z = \frac{(x_1 - 3)^2 + (x_2 - 3)^2}{18}, \quad x_1 \in [0,6], \quad x_2 \in [0,6]. \quad (3.17)$$

Геометрический образ характеристики такого объекта показан на рис. 3.5 (а).

Для идентификации исследуемого объекта применим нейронную сеть-аппроксиматор, которая изображена на рис. 3.6. Эта сеть также является полно-связной. Определим ее структуру: первый (входной) слой - 12 нейронов (по 6 разрядов двоичного кода на каждую входную переменную); второй (внутренний) слой (память сети) - 15 нейронов; третий (выходной) слой - 7 нейронов (6 разрядов - цифровые, 1 - знаковый).

Перед началом обучения сети-аппроксиматора веса его межнейронных связей устанавливались по аналогии с экспериментом 1. Поведение исследуемой модели до обучения сети-аппроксиматора показано на рис. 3.5 (б), после обучения - на рис. 3.5 (в).

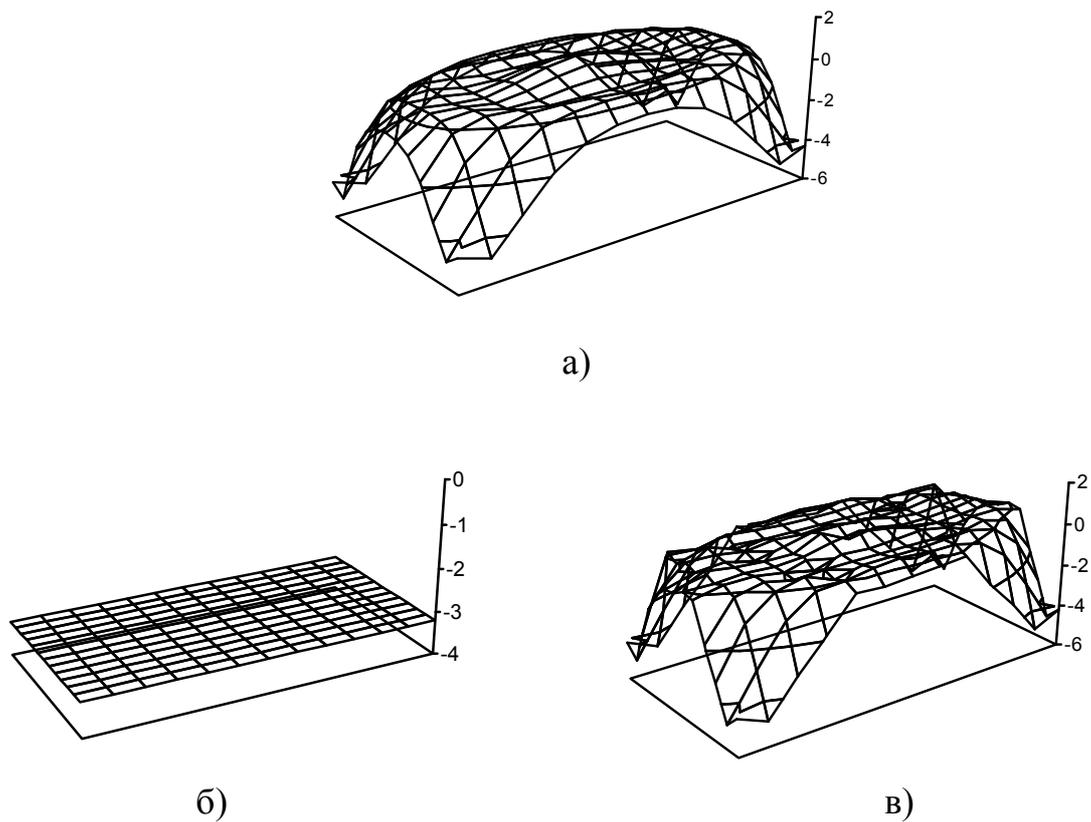


Рис. 3.5. Геометрические образы характеристик объекта (3.17):
 а - эталон;
 б - модель до обучения сети-аппроксиматора;
 в - модель после обучения сети-аппроксиматора.

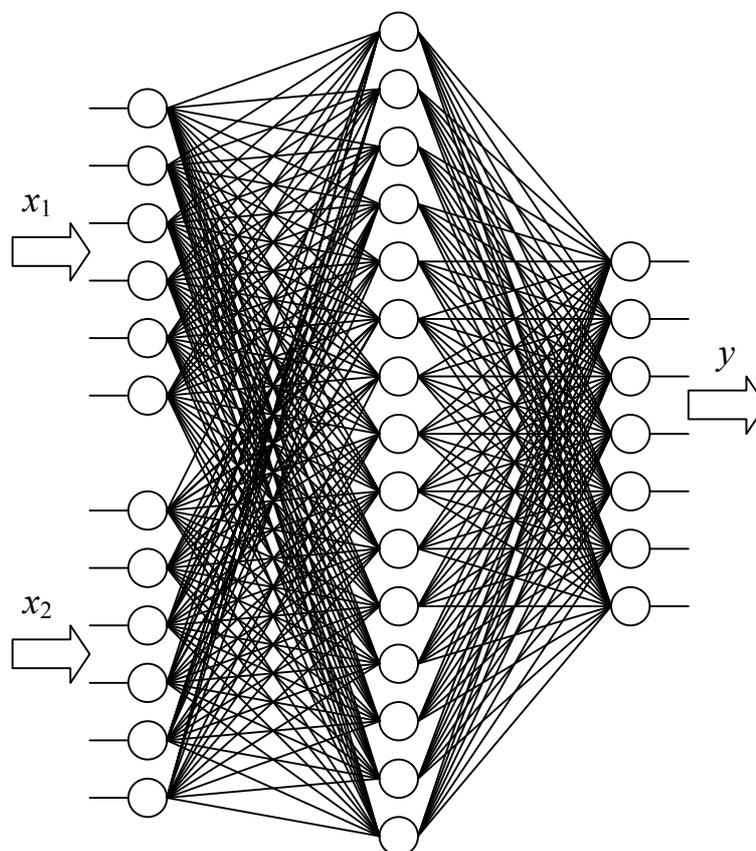


Рис. 3.6. Нейронная сеть-аппроксиматор характеристики объекта (3.17).

В табл. 3.3 и 3.4 приведены значения весовых коэффициентов сети, которые установились после ее обучения.

Таблица 3.3

Веса межнейронных связей входного (a_i) и внутреннего (b_j) слоев

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
b_1	-7,71	1,64	3,87	-6,42	5,93	-7,77	-1,86	1,58	3,38	-2,59	3,29	-7,91
b_2	4,69	-3,62	-4,18	2,74	7,32	2,49	3,58	8,63	-4,88	5,47	-8,64	-0,77
b_3	2,45	10,09	-7,48	14,44	-5,97	-1,01	-6,55	2,13	1,30	-1,23	-8,40	-2,81
b_4	-5,37	-2,29	-4,41	-0,38	4,17	-0,34	5,68	2,81	-15,18	3,82	2,48	-1,39
b_5	3,16	1,31	-14,66	-1,22	-1,66	-4,19	-5,23	-4,46	-7,22	-1,52	-1,42	-2,29
b_6	-0,94	1,03	-6,80	-4,62	-0,08	-2,27	1,83	6,34	-11,07	4,54	6,59	1,86
b_7	-9,17	5,85	-2,40	5,32	2,22	-2,17	-2,04	-2,17	-2,92	2,99	-7,20	-4,25
b_8	-5,16	-9,35	-6,67	4,26	6,59	-4,18	-4,28	-0,88	8,02	-5,55	1,54	2,43
b_9	-6,28	-9,09	-2,28	-1,87	-0,31	-1,40	-2,89	4,43	-3,34	2,77	-3,16	-5,72
b_{10}	-0,59	6,99	10,99	-3,11	-5,00	-0,40	-10,08	1,77	-9,87	1,09	2,77	2,31
b_{11}	-11,22	2,77	-5,51	-0,35	1,30	-3,68	12,20	3,55	-2,52	1,12	1,55	4,63
b_{12}	1,00	0,10	-6,46	-6,54	-2,85	-3,22	5,33	0,57	1,68	-3,89	-1,61	-4,80
b_{13}	3,39	0,30	0,70	-4,36	3,47	0,25	-2,59	-2,87	-4,16	-4,85	-7,96	-8,59
b_{14}	0,64	0,71	4,99	4,79	-1,59	1,01	1,08	-7,51	-11,16	-7,28	-4,81	-4,15
b_{15}	0,93	0,27	0,72	0,68	0,07	0,39	0,99	0,66	0,06	0,99	0,88	0,87

Таблица 3.4

Веса межнейронных связей внутреннего (b_j) и выходного (c_i) слоев

	b_1	b_2	b_3	b_4	b_5	b_6	b_7
c_1	7,78	3,63	4,62	0,67	1,06	4,34	-0,03
c_2	0	0,71	-1,63	-6,19	-6,55	2,54	4,10
c_3	-3,53	-2,55	-2,94	3,09	10,22	-2,76	-9,27
c_4	-2,53	-1,45	-5,70	-3,18	-2,11	4,61	8,92
c_5	-6,93	-2,16	-7,17	-11,96	-9,88	-6,39	-3,31
c_6	-11,68	-5,72	-7,99	-1,09	-0,28	-2,66	3,79
c_7	-11,39	-6,10	-9,39	-3,93	-1,05	-1,69	2,53

Продолжение табл. 3.4

	b_8	b_9	b_{10}	b_{11}	b_{12}	b_{13}	b_{14}	b_{15}
c_1	-6,79	-7,58	-8,67	-8,15	-8,93	-5,88	-1,92	0,02
c_2	2,13	2,40	2,16	-0,02	-0,76	0,07	-1,64	-2,54
c_3	-3,85	-6,65	-5,31	2,55	2,77	5,03	1,22	0,17
c_4	3,22	-1,07	-1,63	-5,63	4,46	3,88	-0,97	-0,48
c_5	0,27	4,34	2,77	4,28	7,33	6,04	5,83	-1,12
c_6	6,91	9,82	6,03	4,54	3,18	3,79	3,72	-4,92
c_7	4,14	8,58	1,14	3,93	7,58	8,82	13,97	6,63

3.3. Нейро-нечеткие модели многомерных зависимостей

3.3.1. Лингвистический аппроксиматор.

Рассмотрим объект с характеристикой вида:

$$y = f(x_1, x_2, \dots, x_n). \quad (3.18)$$

Предположим, что для данного объекта связь «входы (x_i) - выход (y)» можно представить в виде экспертной матрицы знаний, которой будет соответствовать нечеткая база знаний типа (2.35). Классы выходной переменной d_j , $j = \overline{1, m}$, формируются аналогично (2.25).

На основе алгоритмов и соотношений, изложенных в подразделах 2.3-2.4, можно получить соотношения, которые обеспечивают аппроксимацию объекта с характеристикой (3.18) с помощью нечеткой базы знаний и адекватно описывают данный объект. Эти соотношения имеют вид:

$$y = \frac{y \mu^{d_1}(y) + y_1 \mu^{d_2}(y) + \dots + y_{m-1} \mu^{d_m}(y)}{\mu^{d_1}(y) + \mu^{d_2}(y) + \dots + \mu^{d_m}(y)}, \quad (3.19)$$

$$\mu^{d_j}(y) = \max_{p=1, k_j} \left\{ w_{jp} \min_{i=1, n} [\mu^{jp}(x_i)] \right\}, \quad (3.20)$$

$$\mu^{jp}(x_i) = \frac{1}{1 + \left(\frac{x_i - b_i^{jp}}{c_i^{jp}} \right)^2}, \quad i = \overline{1, n}, \quad j = \overline{1, m}, \quad p = k_j, \quad (3.21)$$

где $\mu^{d_j}(y)$ - функция принадлежности выходной переменной y к классу $d_j \in$

$$[y_{j-1}, y_j];$$

$\mu^{j^p}(x_i)$ - функция принадлежности входной переменной x_i к лингвистическому терму a_i^p ;

$b_i^{j^p}, c_i^{j^p}$ - параметры настройки функций принадлежности.

3.3.2. Нейро-нечеткая сеть.

Покажем, как доступную лингвистическую информацию об объекте (3.18) можно представить в виде специальной конструкции, которую назовем нейро-нечеткой сетью и которая является изоморфной нечеткой базе знаний о данном объекте.

На рис. 3.7 представлена структура нейро-нечеткой сети, а функции ее узлов (аналоги нейронов в обычной нейронной сети) показаны в табл. 3.5.

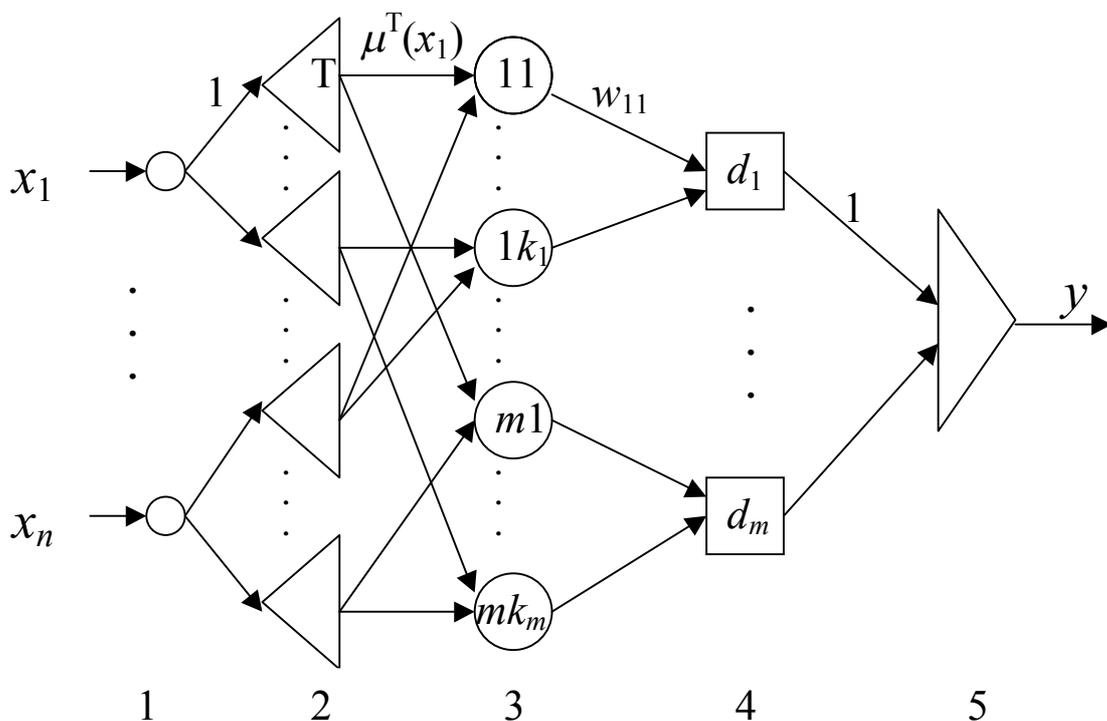
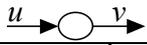
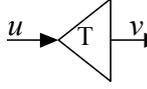
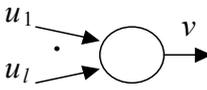
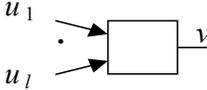
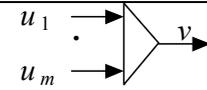


Рис. 3.7. Структура нейро-нечеткой сети.

Элементы нейро-нечеткой сети

Узел	Название	Функция
	Вход	$v = u$
	Нечеткий терм	$v = \mu^T(u)$
	Нечеткое правило	$v = \prod_{i=1}^l u_i$
	Класс правил	$v = \sum_{i=1}^l u_i$
	Дефаззификация	$v = \frac{\sum_{j=1}^m u_j \bar{d}_j}{\sum_{j=1}^m u_j}$

Как можно видеть из рис. 3.7, нейро-нечеткая сеть всегда имеет пять слоев:

слой 1 - входы объекта идентификации;

слой 2 - нечеткие термы, которые используются в нечеткой базе знаний, описывающей исследуемый объект;

слой 3 - строки-конъюнкции (нечеткие правила) нечеткой базы знаний исследуемого объекта;

слой 4 - классы выходной переменной d_j , $j = \overline{1, m}$, объединяющие нечеткие правила;

слой 5 - блок дефаззификации (3.19), т. е. преобразования результатов нечеткого логического вывода в четкое число.

Количество же узлов в нейро-нечеткой сети является переменным и определяется следующим образом:

слой 1 - по количеству входов объекта идентификации;

слой 2 - по количеству нечетких термов входных переменных в нечеткой базе знаний, которая описывает исследуемый объект;

слой 3 - по количеству строк-конъюнкций (нечетких правил) в базе знаний;

слой 4 - по количеству классов, на которые разбивается диапазон (множество значений) выходной переменной.

По аналогии с обычной нейронной сетью, межэлементным связям в нейро-нечеткой сети также присваиваются определенные весовые коэффициенты, которые являются переменными параметрами модели. Межэлементные связи взвешиваются следующим образом:

между 1-м и 2-м слоями - единицы (всегда константы);

между 2-м и 3-м слоями - параметрами b и c функций принадлежности входов к нечетким термам;

между 3-м и 4-м слоями - весами нечетких правил;

между 4-м и 5-м слоями - единицы (всегда константы).

В табл. 3.5 введены также следующие обозначения:

$\mu^T(u)$ - функция принадлежности входной переменной u к нечеткому терму T ;

\bar{d}_j - центр класса $d_j \in [\underline{y}, \bar{y}]$ (для объектов с непрерывным выходом).

Опираясь на обоснование, которое изложено в работе [96], при описании в табл. 3.5 функций элементов «нечеткое правило» и «класс правил» нечеткологические операции \min и \max в формуле (3.20) заменены на арифметические операции умножения и суммы. Проведенная замена дает возможность получить аналитические выражения, удобные для дифференцирования при обучении нейро-нечеткой сети.

Таким образом, нами получена структура, которая воплощает в себя нечеткую базу знаний об объекте идентификации, построенную экспертным путем или методом извлечения знаний, (что соответствует этапу грубой настройки), а также механизмы нечеткого логического вывода. Однако главным предназначением данной структуры является именно тонкая настройка полученной

базы знаний, и в данном случае эта проблема решается путем применения механизмов обучения нейронных сетей.

3.3.3. Обучение нейро-нечеткой сети.

Процесс обучения нейро-нечеткой сети аналогичен процедуре обучения традиционных нейронных сетей по правилу «back-propagation», согласно которому алгоритм обучения состоит из двух фаз - прямого и обратного хода. Опишем эти фазы для нейро-нечеткой сети с учетом соответствующих особенностей и соотношений.

Алгоритм прямого хода

ШАГ 1. Подать на входы сети значения входных величин.

ШАГ 2. Вычислить степени принадлежности значений входных величин к лингвистическим термам по формуле (3.21).

ШАГ 3. Вычислить степени принадлежности значения выходной величины к соответствующим классам по формуле (3.20) (нечеткий логический вывод) с учетом особенностей, приведенных в табл. 3.5.

ШАГ 4. Вычислить модельное значение выходной величины y путем дефаззификации (формула (3.19)).

ШАГ 5. Рассчитать ошибку сети:

$$E_t = \frac{1}{2} (\hat{y}_t - y_t)^2, \quad (3.22)$$

где \hat{y}_t и y_t - соответственно модельное и экспериментальное значения выхода объекта (3.18) на t -м шаге обучения.

КОНЕЦ АЛГОРИТМА.

Алгоритм обратного хода

ШАГ 1. Определить скорость изменения ошибки сети при изменении значения выходной величины:

$$\frac{\partial E_t}{\partial y} = \varepsilon_1 = y_t - \hat{y}_t. \quad (3.23)$$

ШАГ 2. Определить скорость изменения ошибки при изменении степени принадлежности значения выходной величины к соответствующим классам:

$$\frac{\partial E_t}{\partial \mu^{d_j}(y)} = \frac{\partial E_t}{\partial y} \frac{\partial y}{\partial \mu^{d_j}(y)} = \varepsilon_1 \varepsilon_2, \quad (3.24)$$

$$\varepsilon_2 = \frac{\partial y}{\partial \mu^{d_j}(y)} = \frac{\bar{d}_j \sum_{j=1}^m \mu^{d_j}(y) - \sum_{j=1}^m \bar{d}_j \mu^{d_j}(y)}{\left(\sum_{j=1}^m \mu^{d_j}(y) \right)^2}. \quad (3.25)$$

ШАГ 3. Определить скорость изменения ошибки при изменении веса того или иного правила:

$$\frac{\partial E_t}{\partial w_{jp}} = \varepsilon_1 \varepsilon_2 \frac{\partial \mu^{d_j}(y)}{\partial w_{jp}}, \quad (3.26)$$

$$\frac{\partial \mu^{d_j}(y)}{\partial w_{jp}} = \prod_{i=1}^n \mu^{jp}(x_i), \quad p = k_j. \quad (3.27)$$

ШАГ 4. Определить скорость изменения ошибки при изменении параметров b и c функций принадлежности лингвистических термов:

$$\frac{\partial E_t}{\partial c_i^{jp}} = \varepsilon_1 \varepsilon_2 \varepsilon_3 \varepsilon_4 \frac{\partial \mu^{jp}(x_i)}{\partial c_i^{jp}}, \quad (3.28)$$

$$\frac{\partial E_t}{\partial b_i^{jp}} = \varepsilon_1 \varepsilon_2 \varepsilon_3 \varepsilon_4 \frac{\partial \mu^{jp}(x_i)}{\partial b_i^{jp}}, \quad (3.29)$$

где

$$\varepsilon_3 = \frac{\partial \mu^{dj}(y)}{\partial \left(\prod_{i=1}^n \mu^{jp}(x_i) \right)} = w_{jp}, \quad (3.30)$$

$$\varepsilon_4 = \frac{\partial \left(\prod_{i=1}^n \mu^{jp}(x_i) \right)}{\partial \mu^{jp}(x_i)} = \frac{1}{\mu^{jp}(x_i)} \prod_{i=1}^n \mu^{jp}(x_i), \quad (3.31)$$

$$\frac{\partial \mu^{jp}(x_i)}{\partial c_i^{jp}} = \frac{2c_i^{jp}(x_i - b_i^{jp})^2}{\left((c_i^{jp})^2 + (x_i - b_i^{jp})^2 \right)^2}, \quad (3.32)$$

$$\frac{\partial \mu^{jp}(x_i)}{\partial b_i^{jp}} = \frac{2(c_i^{jp})^2(x_i - b_i^{jp})}{\left((c_i^{jp})^2 + (x_i - b_i^{jp})^2 \right)^2}, \quad j = \overline{1, m}, \quad i = \overline{1, n}, \quad p = k_j. \quad (3.33)$$

ШАГ 5. Провести модификацию переменных параметров сети по правилу градиента:

$$w_{jp}(t+1) = w_{jp}(t) - \eta \frac{\partial E_t}{\partial w_{jp}(t)}, \quad (3.34)$$

$$c_i^{jp}(t+1) = c_i^{jp}(t) - \eta \frac{\partial E_t}{\partial c_i^{jp}(t)}, \quad (3.35)$$

$$b_i^{jp}(t+1) = b_i^{jp}(t) - \eta \frac{\partial E_t}{\partial b_i^{jp}(t)}, \quad j = \overline{1, m}, \quad i = \overline{1, n}, \quad p = k_j, \quad (3.36)$$

где t - номер шага итерации;

η - параметр, характеризующий скорость обучения и который может выбираться в соответствии с рекомендациями работы [77].

КОНЕЦ АЛГОРИТМА.

3.3.4. Компьютерное моделирование.

Для проведения компьютерных экспериментов по идентификации нелинейных объектов нейро-нечеткими сетями воспользуемся теми же эталонными зависимостями, которые исследовались в п. 3.1.3.

Эксперимент 1

В качестве эталонного объекта с одной входной переменной примем объект, который описывается зависимостью (3.16). Поведение этого объекта показано на рис. 3.8 жирной линией.

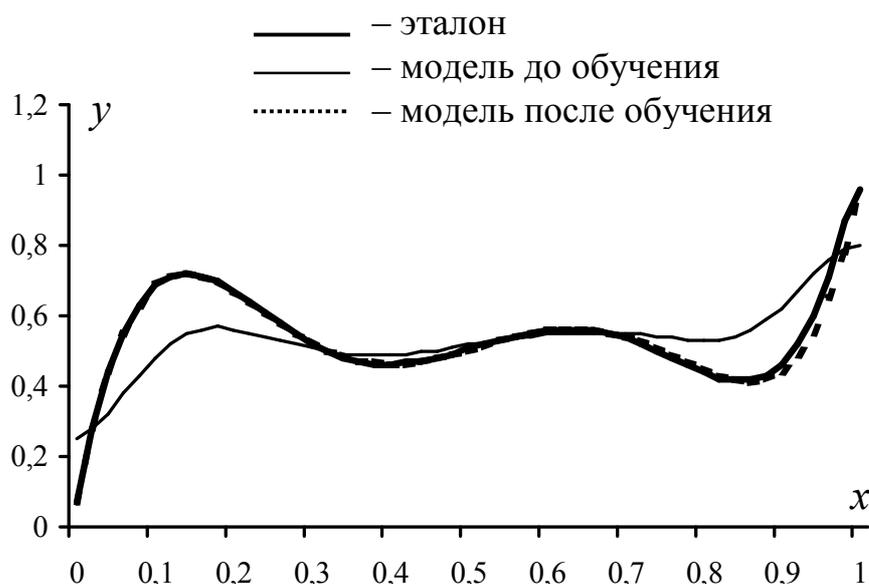


Рис. 3.8. Характеристики модели объекта (3.16).

Наблюдая поведение объекта, опишем его нечеткой базой знаний:

ЕСЛИ $x = A_1 = \text{около } 0$ ТО $y = d_1 \in [0, 0.4]$;

ЕСЛИ $x = A_2 = \text{около } 0,15$ ТО $y = d_5 \in [0.7, 0.9]$;

ЕСЛИ $x = A_3 = \text{около } 0,4$ ТО $y = d_3 \in [0.45, 0.55]$;

ЕСЛИ $x = A_4 = \text{около } 0,6$ ТО $y = d_4 \in [0.55, 0.7]$;

ЕСЛИ $x = A_5 = \text{около } 0,85$ ТО $y = d_2 \in [0.4, 0.45]$;

ЕСЛИ $x = A_6 = \text{около } 1$ ТО $y = d_6 \in [0.9, 1]$,

где $A_1 \div A_6$ - лингвистические термы входной переменной;

$d_1 \div d_6$ - классы выходной переменной.

Функции принадлежности нечетких лингвистических термов были выбраны экспертным путем. Их графики до обучения сети-аппроксиматора показаны на рис. 3.9 (а), а начальные значения параметров b и c представлены в табл. 3.6.

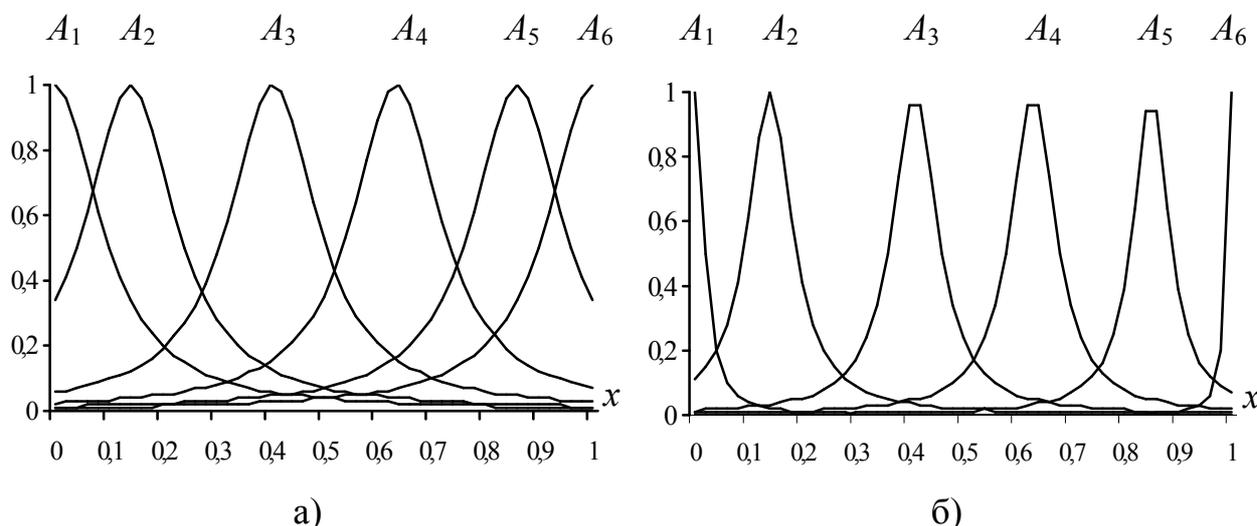


Рис. 3.9. Графики функций принадлежности нечетких термов входной переменной:
 а - до обучения сети-аппроксиматора;
 б - после обучения сети-аппроксиматора.

Таблица 3.6

Параметры функций принадлежности нечетких термов входной переменной

Термы		A_1	A_2	A_3	A_4	A_5	A_6
До обучения	b	0	0.15	0.4	0.6	0.85	1
	c	0.3	0.3	0.3	0.3	0.3	0.3
После обучения	b	0	0.13	0.41	0.63	0.85	1
	c	0.02	0.04	0.05	0.05	0.04	0.01

Структура нейро-нечеткой сети, которая соответствует исследуемому объекту, показана на рис. 3.10. Данная сеть до обучения использует соответст-

вующие параметры функций принадлежности, а также веса правил, которые равняются 1, и таким образом обеспечивает грубую аппроксимацию объекта. Геометрический образ модели до настройки показан на рис. 3.8 тонкой линией.

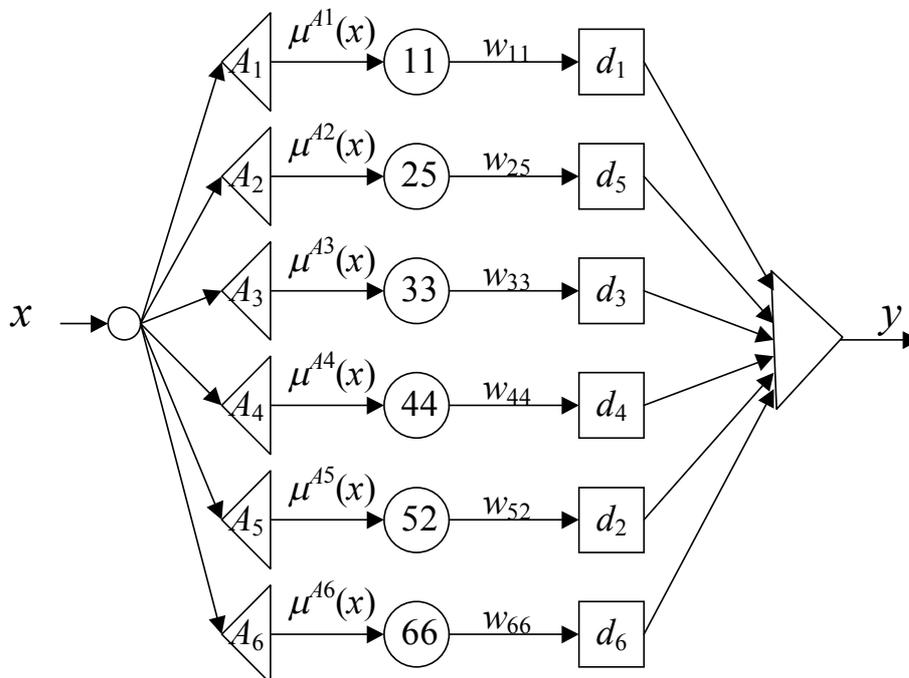


Рис. 3.10. Нейро-нечеткая сеть для идентификации объекта (3.16).

В результате обучения сети по вышеописанным алгоритмам была получена модель объекта с параметрами b и c функций принадлежности и весами нечетких правил, которые приведены в табл. 3.6 и 3.7. Геометрический образ модели после настройки показан на рис. 3.8 пунктиром, а графики функций принадлежности нечетких термов после обучения изображены на рис. 3.9 (б).

Таблица 3.7

Веса правил						
Вес правила	w_{11}	w_{25}	w_{33}	w_{44}	w_{52}	w_{66}
До обучения	1	1	1	1	1	1
После обучения	0.9997	0.9985	0.9988	0.9985	0.9984	0.9975

Эксперимент 2

В качестве эталонного объекта с двумя входными переменными был использован объект, который описывается зависимостью (3.17). Геометрический образ характеристики такого объекта показан на рис. 3.11 (а).

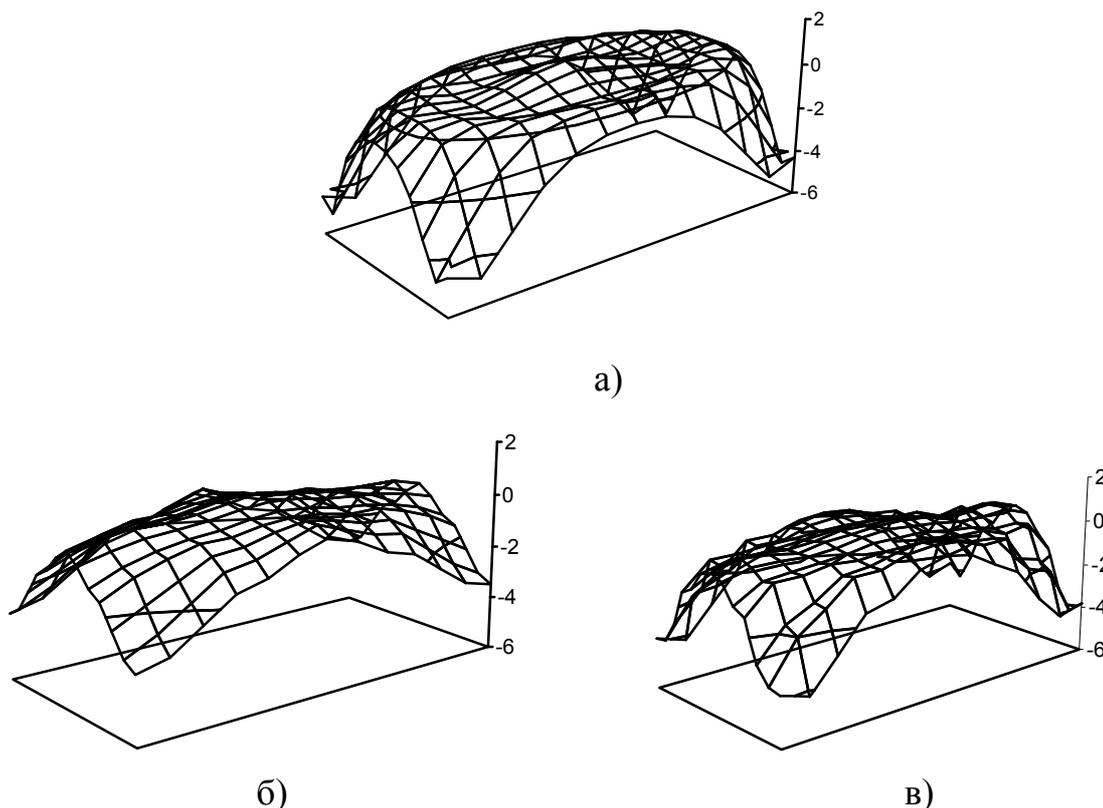


Рис. 3.11. Геометрические образы характеристик объекта (3.17):
 а - эталон;
 б - модель до обучения сети-аппроксиматора;
 в - модель после обучения сети-аппроксиматора.

Для описания поведения данного объекта сформируем экспертным путем нечеткую базу знаний из 49 правил вида: ЕСЛИ $x_1 = A_i$ И $x_2 = B_i$ ТО $y = C_j$.

Изобразим эту базу знаний схематически в виде матрицы:

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
B_1	C_2	C_1	C_3	C_4	C_3	C_1	C_2
B_2	C_1	C_3	C_4	C_5	C_4	C_3	C_1
B_3	C_3	C_4	C_4	C_4	C_4	C_4	C_3
B_4	C_4	C_5	C_4	C_5	C_4	C_5	C_4
B_5	C_3	C_4	C_4	C_4	C_4	C_4	C_3
B_6	C_1	C_3	C_4	C_5	C_4	C_3	C_1
B_7	C_2	C_1	C_3	C_4	C_3	C_1	C_2

где:

1) $A_1 = B_1 = \text{около } 0$, $A_2 = B_2 = \text{около } 0.5$, $A_3 = B_3 = \text{около } 1.5$, $A_4 = B_4 = \text{около } 3$, $A_5 = B_5 = \text{около } 4.5$, $A_6 = B_6 = \text{около } 5.5$, $A_7 = B_7 = \text{около } 6$ - лингвистические термы входных переменных x_1 и x_2 ;

2) $C_1 = [-5.08, -4.5]$, $C_2 = [-4.5, -3]$, $C_3 = [-3, -0.5]$, $C_4 = [-0.5, 0]$, $C_5 = [0, 0.855]$ - классы выходной переменной y .

Начальные веса всех правил, так же, как и в предыдущем эксперименте, равняются 1. Графики функций принадлежности нечетких термов входных переменных до настройки показаны на рис. 3.12 (а), а значения их параметров b и c приведены в табл. 3.8 и 3.9.

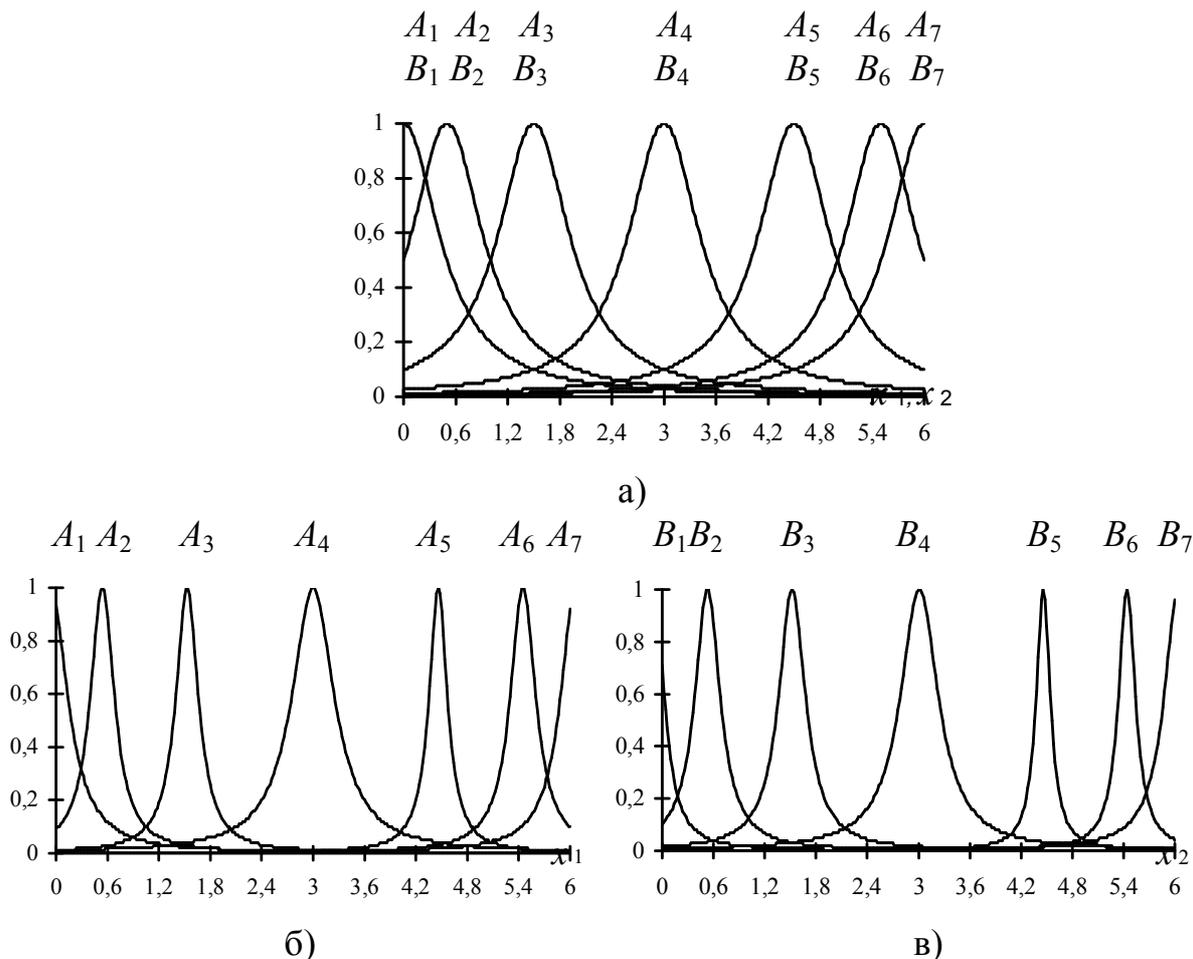


Рис. 3.12. Графики функций принадлежности термов входных переменных: а - до обучения сети-аппроксиматора; б, в - после обучения сети-аппроксиматора.

Таблица 3.8

Параметры функций принадлежности термов переменной x_1

Термы		A_1	A_2	A_3	A_4	A_5	A_6	A_7
До обучения	b	0	0.5	1.5	3	4.5	5.5	6
	c	0.5	0.5	0.5	0.5	0.5	0.5	0.5
После обучения	b	-0.16	0.54	1.53	3	4.46	5.45	6.16
	c	0.24	0.17	0.16	0.3	0.13	0.18	0.21

Таблица 3.9

Параметры функций принадлежности термов переменной x_2

Термы		B_1	B_2	B_3	B_4	B_5	B_6	B_7
До обучения	b	0	0.5	1.5	3	4.5	5.5	6
	c	0.5	0.5	0.5	0.5	0.5	0.5	0.5
После обучения	b	-0.19	0.53	1.52	3.01	4.46	5.44	6.14
	c	0.14	0.18	0.19	0.27	0.1	0.12	0.2

Структура нейро-нечеткой сети, адекватной по характеристикам исследуемому объекту, показана на рис. 3.13. Данная сеть до обучения использует вышеуказанные параметры b , c и w и обеспечивает грубую аппроксимацию объекта. Геометрический образ модели до настройки показан на рис. 3.11 (б).

В результате обучение сети по соответствующим алгоритмам была получена модель объекта, геометрический образ которой показан на рис. 3.11 (в). Графики функций принадлежности нечетких термов после обучения показаны на рис. 3.12 (б, в), а значения их параметров b и c , а также модифицированных весов нечетких правил приведены в табл. 3.8 - 3.10.

Таблица 3.10

Итоговые веса правил

Термы	A_1	A_2	A_3	A_4	A_5	A_6	A_7
B_1	0.9995	0.9954	0.9479	0.9989	0.9761	0.9738	0.9785
B_2	0.9969	0.9014	0.9998	0.9989	0.9997	0.9366	0.9451
B_3	0.9894	0.9999	1	1	1	0.9998	0.958
B_4	0.9862	1	1	0.9834	1	0.9993	0.9976
B_5	0.9938	0.9995	1	1	1	0.9995	0.9688
B_6	0.997	0.8971	0.9994	0.9983	0.9991	0.934	0.9245
B_7	0.9995	0.9986	0.9536	0.9988	0.98	0.9831	0.9806

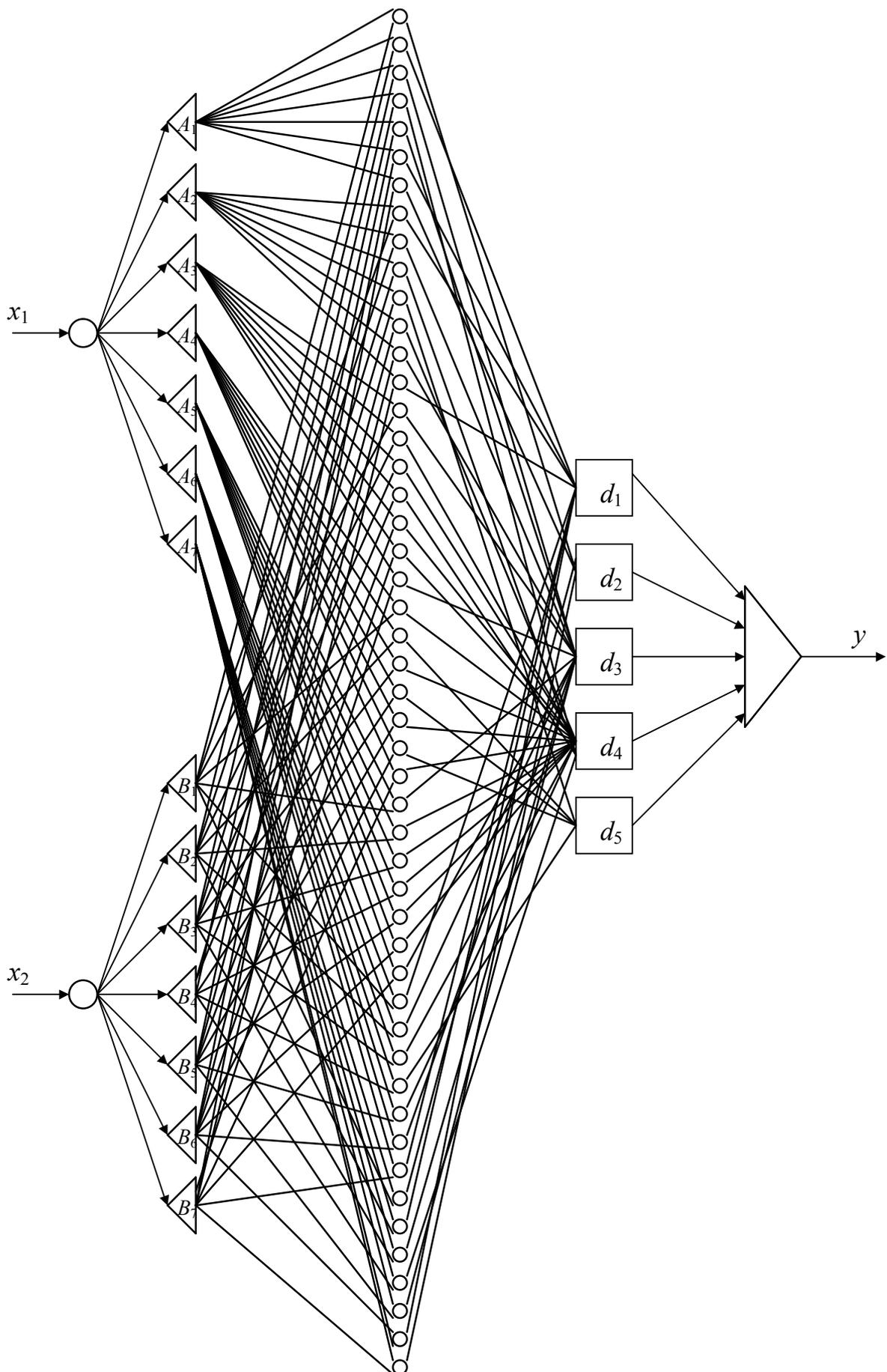


Рис. 3.13. Нейро-нечеткая сеть для идентификации объекта (3.17).

ГЛАВА 4

ЭКСПЕРИМЕНТЫ С ЭТАЛОННЫМИ МОДЕЛЯМИ

Данная глава посвящена описанию результатов компьютерных экспериментов, проведенных с целью проверки возможности извлечения из экспериментальных данных нечетких баз знаний о нелинейных объектах с непрерывным [61] и дискретным выходом с помощью генетических алгоритмов.

4.1. Зависимости с непрерывным выходом

4.1.1. Объект «один вход - один выход».

Пусть для объекта $y = f(x)$ модель-эталон задана следующим аналитическим выражением:

$$y = f(x) = e^{\frac{-x}{4}} \cdot \sin\left(\frac{\pi}{2}x\right), \quad x \in [0,10], \quad y \in [-0.47,0.79]. \quad (4.1)$$

График характеристики данного объекта изображен на рис.4.1.

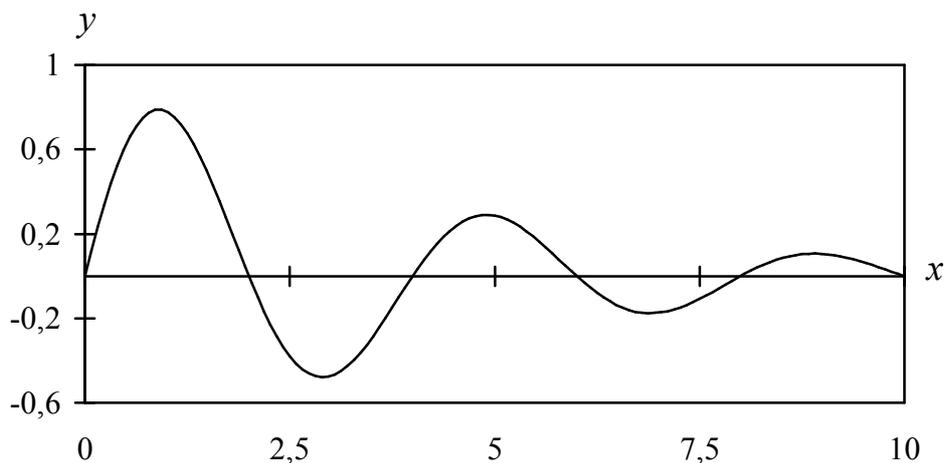


Рис. 4.1. График характеристики объекта «один вход - один выход».

Разобьем интервал изменения выходной переменной исследуемого объекта на семь подинтервалов следующим образом:

$$y \in \underbrace{[-0.47, -0.30)}_{d_1} \cup \underbrace{[-0.30, -0.05)}_{d_2} \cup \underbrace{[-0.05, 0.15)}_{d_3} \cup \underbrace{[0.15, 0.30)}_{d_4} \cup \\ \cup \underbrace{[0.30, 0.45)}_{d_5} \cup \underbrace{[0.45, 0.65)}_{d_6} \cup \underbrace{[0.65, 0.78]}_{d_7}$$

Задачу извлечения нечеткой базы знаний об исследуемом объекте сформулируем так:

синтезировать совокупность правил ЕСЛИ-ТО, описывающих объект (4.1), по 5 правил на каждый подинтервал выходной переменной. В процессе синтеза веса правил должны приравниваться лишь к двум значениям - 0 (ошибочное правило) или 1 (истинное правило).

В результате применения для решения данной задачи генетического алгоритма оптимизации были получены параметры матрицы знаний, представленные в табл. 4.1.

Из представленной таблицы следует удалить строки, соответствующие нулевым весам правил, поскольку такие правила не будут влиять на воспроизведение характеристики исследуемого объекта при помощи извлеченной нечеткой базы знаний.

Основываясь лишь на правилах с единичными весами, по извлеченным параметрам b и c можно построить функции принадлежности термов входной переменной, графики которых для соответствующих правил показаны в табл. 4.2.

Если термы, которым соответствуют представленные в табл. 4.2 функции принадлежности, интерпретировать лингвистически, то получим нечеткую базу знаний, правила которой будут сформулированы следующим образом:

ЕСЛИ $x = \text{около } 2.8$ ТО $y \in d_1$

ЕСЛИ $x = \text{около } 6.9$ ТО $y \in d_2$

ЕСЛИ $x = \text{около } 0$ ИЛИ $x = \text{около } 8.8$ ИЛИ $x = \text{около } 10$ ТО $y \in d_3$

Таблица 4.1

Параметры матрицы знаний для объекта (4.1)

ЕСЛИ		вес	ТО	ЕСЛИ		вес	ТО
<i>b</i>	<i>c</i>			<i>b</i>	<i>c</i>		
2.13	0.72	0	d_7	0.13	0.87	1	d_3
0.7	1.25	0		9.1	1.25	0	
0.92	0.7	1		8.62	2.2	1	
0.93	1.12	0		9.92	1.12	1	
1.01	1.9	1		8.7	1.33	1	
1.52	0.64	0	d_6	3.71	0.25	0	d_2
4	1.17	0		6.11	1.13	1	
1.82	1.41	0		6.91	2.05	1	
3.12	1.9	0		6.83	0.72	1	
0.92	1.65	0		1.13	0.92	0	
2.25	2.01	0	d_5	2.81	1.12	1	d_1
2.19	1.36	0		2.72	0.7	1	
3.01	1.39	0		2.93	0.85	1	
5.01	0.71	0		0.13	0.64	0	
1.1	0.72	0		2.81	1.17	1	
4.91	0.21	1	d_4				
5.2	1.5	1					
5.01	0.9	1					
5.12	0.83	0					
9.17	1.19	0					

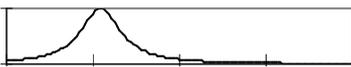
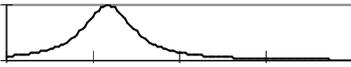
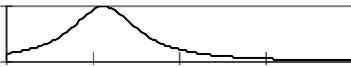
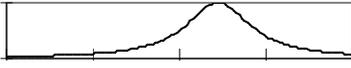
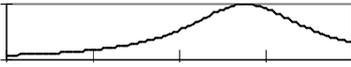
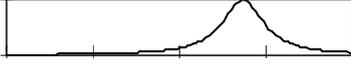
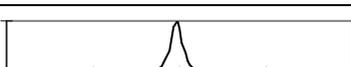
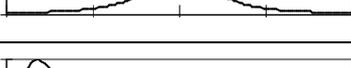
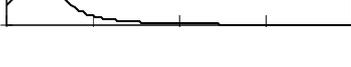
ЕСЛИ $x = \text{около } 5$ ТО $y \in d_4$

ЕСЛИ $x = \text{около } 0.9$ ТО $y \in d_7$

На рис. 4.2 изображена модель объекта, полученная в результате его аппроксимации извлеченной нечеткой базой знаний, в сравнении с эталоном.

Таблица 4.2

Параметры и функции принадлежности синтезированных правил

ЕСЛИ x		функции принадлежности	ТО
параметры термов			y
b	c		
2.81	1.12		d_1
2.72	0.7		
2.93	0.85		
2.81	1.17		
6.11	1.13		d_2
6.91	2.05		
6.83	0.72		
0.13	0.87		d_3
8.62	2.2		
9.92	1.12		
8.7	1.33		
4.91	0.21		d_4
5.2	1.5		
5.01	0.9		
0.92	0.7		d_7
1.01	1.9		

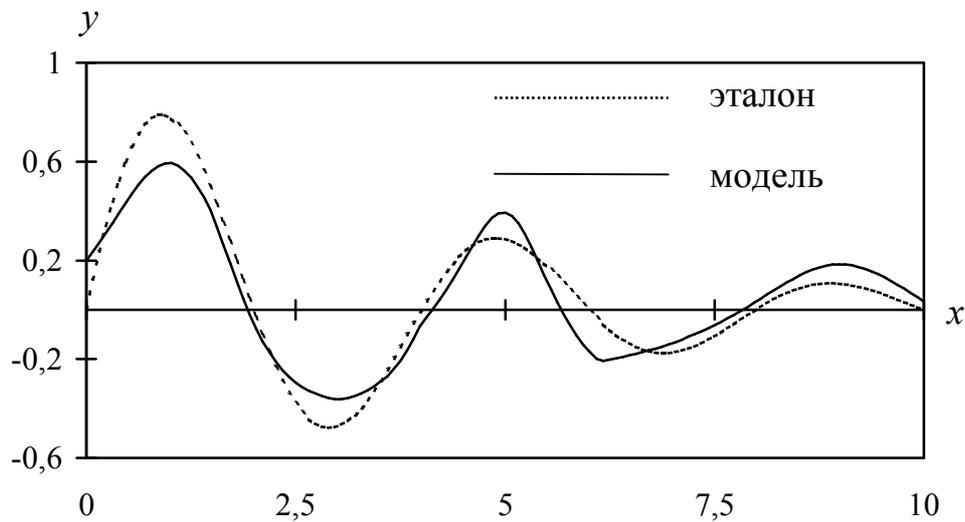


Рис. 4.2. Графическое сравнение модели с эталоном.

4.1.2. Объект «два входа - один выход».

Для объекта $y = f(x_1, x_2)$ модель-эталон в аналитической форме задана следующим образом:

$$y = \frac{1}{40}(2z - 0.9)(7z - 1)(17z - 19)(15z - 2), \quad z = \frac{(x_1 - 3)^2 + (x_2 - 3)^2}{18},$$

$$x_1 \in [0, 6], \quad x_2 \in [0, 6], \quad y \in [-5.08, 0.855]. \quad (4.2)$$

На рис. 4.3(а) показан геометрический образ характеристик исследуемого объекта.

Интервал изменения выходной переменной объекта разобьем на пять подинтервалов: $y \in \underbrace{[-5.08, -4.5]}_{d_1} \cup \underbrace{[-4.5, 3]}_{d_2} \cup \underbrace{[-3, -0.5]}_{d_3} \cup \underbrace{[-0.5, 0]}_{d_4} \cup \underbrace{[0, 0.855]}_{d_5}$.

Задача извлечения нечеткой базы знаний об исследуемом объекте аналогична предыдущей:

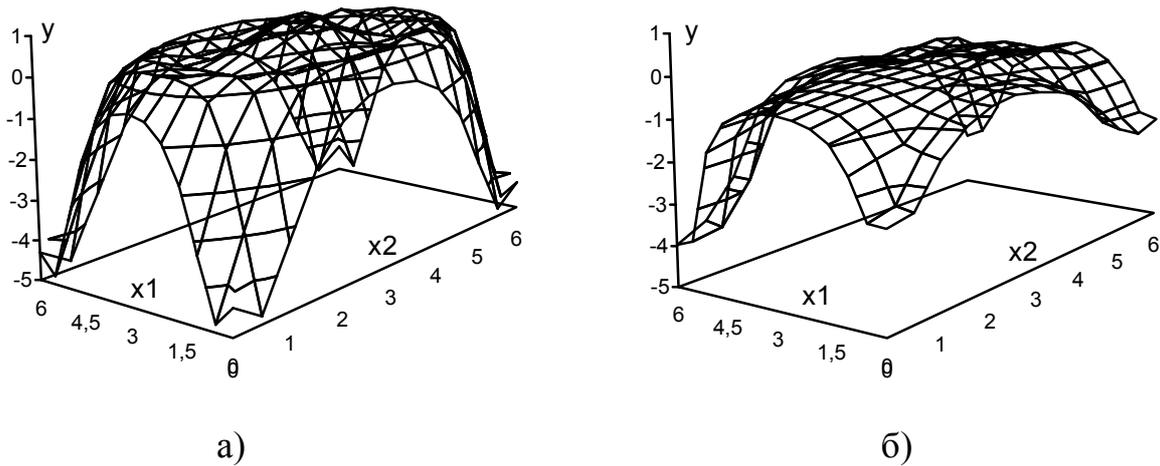


Рис. 4.3. Геометрический образ объекта (4.2):
 а - эталон;
 б - модель.

синтезировать совокупность правил ЕСЛИ-ТО, описывающих объект (4.2), по 5 правил на каждый подинтервал выходной переменной. Веса правил должны приравниваться лишь значениям 0 или 1.

В результате применения для решения данной задачи генетического алгоритма оптимизации были получены параметры матрицы знаний, представленные в табл. 4.3.

Данную таблицу следует сократить так же, как и в п. 4.2.1. В табл. 4.4 приведены параметры функций принадлежности правил, веса которых равняются единице, а также построены функции принадлежности термов входных переменных.

После лингвистической интерпретации термов, которым соответствуют функции принадлежности из табл. 4.4, получим нечеткую базу знаний, которая состоит из следующих правил:

ЕСЛИ $x_1 = \text{около } 5.7$ И $x_2 = \text{около } 0.2$ ТО $y \in d_1$

ЕСЛИ $x_1 = \text{около } 5.6$ И $x_2 = \text{около } 6$ ТО $y \in d_2$

ЕСЛИ $x_1 = \text{около } 0.2$ И $x_2 = \text{около } 0.3$

Таблица 4.3

Параметры синтезированной матрицы знаний
для объекта «два входа - один выход»

ЕСЛИ				вес	ТО
x_1		x_2			y
b	c	b	c		
5.74	0.21	0.18	0.54	1	d_1
2.84	0.68	1.74	0.79	0	
4.78	0.77	5.95	1.37	0	
1.73	0.36	5.64	1.24	0	
0.37	1.44	1.00	0.92	0	
4.64	1.22	0.70	0.23	0	d_2
4.07	0.76	2.88	1.09	0	
0.53	1.36	0.73	1.13	0	
5.61	1.14	6.00	0.24	1	
5.11	1.28	0.48	0.55	0	
3.74	1.28	0.57	0.33	0	d_3
2.86	0.52	3.85	0.82	0	
3.88	0.23	5.63	0.88	0	
0.22	0.32	0.27	0.31	1	
0.10	0.39	5.52	0.47	1	
2.50	0.33	5.57	1.09	1	d_4
4.40	0.35	5.64	1.38	1	
4.12	0.83	4.70	0.56	0	
5.68	1.02	4.61	0.91	0	
4.00	1.41	2.57	0.53	1	
5.14	0.25	4.66	0.64	1	d_5
3.16	1.26	5.60	1.32	1	
2.21	0.47	4.20	1.45	1	
3.26	0.72	2.93	1.08	1	
4.31	1.37	2.37	0.63	1	

ИЛИ $x_1 = \text{около } 0$ И $x_2 = \text{около } 5.5$ ТО $y \in d_3$

ЕСЛИ $x_1 = \text{около } 2.5$ И $x_2 = \text{около } 5.6$

ИЛИ $x_1 = \text{около } 4.4$ И $x_2 = \text{около } 5.6$

Таблица 4.4

Параметры и функции принадлежности синтезированных правил

ЕСЛИ						ТО
x_1			x_2			y
параметры термов		функции принадлежности	параметры термов		функции принадлежности	
b	c		b	c		
5.74	0.21		0.18	0.54		d_1
5.61	1.14		6.00	0.24		d_2
0.22	0.32		0.27	0.31		d_3
0.10	0.39		5.52	0.47		
2.50	0.33		5.57	1.09		d_4
4.40	0.35		5.64	1.38		
4.00	1.41		2.57	0.53		
5.14	0.25		4.66	0.64		d_5
3.16	1.26		5.60	1.32		
2.21	0.47		4.20	1.45		
3.26	0.72		2.93	1.08		
4.31	1.37		2.37	0.63		

ИЛИ $x_1 = \text{около } 4$ И $x_2 = \text{около } 2.6$ ТО $y \in d_4$
 ЕСЛИ $x_1 = \text{около } 5.1$ И $x_2 = \text{около } 4.7$
 ИЛИ $x_1 = \text{около } 3.1$ И $x_2 = \text{около } 5.6$
 ИЛИ $x_1 = \text{около } 2.2$ И $x_2 = \text{около } 4.2$
 ИЛИ $x_1 = \text{около } 3.3$ И $x_2 = \text{около } 2.9$
 ИЛИ $x_1 = \text{около } 4.3$ И $x_2 = \text{около } 2.4$ ТО $y \in d_5$.

Геометрический образ модели объекта, которая получена в результате ее аппроксимации извлеченной нечеткой базой знаний, вместе с эталоном изображены на рис. 4.3(б).

4.2. Зависимости с дискретным выходом

4.2.1. Объект «два входа - один выход» 1-го типа.

В качестве такого объекта с дискретным выходом рассмотрим объект, для которого известно:

а) два входа: $x_1 \in [0,1]$ и $x_2 \in [0,1]$;

б) один выход y , который характеризуется 5-ю возможными значениями:
 $\{d_1, d_2, d_3, d_4, d_5\}$.

Таким образом, исследуемый объект представляет собой двумерное пространство, разделенное на пять областей - классов решений (рис. 4.4).

Границы областей-классов заданы следующими аналитическими выражениями:

$$\textcircled{1}: x_1 \in [0.5, 0.625], x_2 = 32(x_1 - 0.5)^2 + 0.5;$$

$$\textcircled{2}: x_1 \in [0, 0.25], x_2 = -16(x_1 - 0.25)^4 + 0.8125;$$

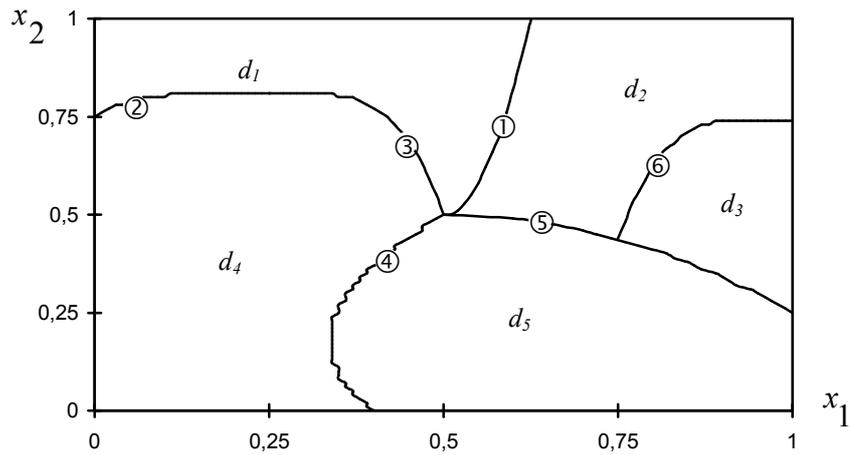


Рис. 4.4. Объект «два входа - один выход» 1-го типа с дискретным выходом.

$$\textcircled{3}: x_1 \in [0.25, 0.5], x_2 = -80(x_1 - 0.25)^4 + 0.8125;$$

$$\textcircled{4}: x_2 \in [0, 0.5], x_1 = (0.5 - x_2)(2 - x_2)(-0.1 - x_2) + 0.5;$$

$$\textcircled{5}: x_2 \in [0.25, 0.5], x_1 = \sqrt{0.5 - x_2} + 0.5;$$

$$\textcircled{6}: x_1 \in [0.75, 1], x_2 = -80(x_1 - 1)^4 + 0.75.$$

Сформулируем задачу извлечения нечеткой базы знаний об исследуемом объекте:

синтезировать совокупность правил ЕСЛИ-ТО, по 5 правил для каждого класса выходной переменной, при помощи которых проводилось бы определение соответствия значений пар входных переменных тому или другому классу выхода.

Так же, как и в предыдущих экспериментах, веса правил должны равняться лишь 0 или 1.

В результате применения для решения данной задачи генетического алгоритма оптимизации были получены параметры матрицы знаний, представленные в табл. 4.5. В табл. 4.6 приведены параметры функций принадлежности правил с единичными весовыми коэффициентами, а также показаны формы функций принадлежности термов входных переменных.

Параметры синтезированной матрицы знаний

ЕСЛИ				вес	ТО
x_1		x_2			
b	c	b	c		y
0.41	0.21	0.87	0.08	0	d_1
0.12	0.05	0.89	0.09	1	
0.13	0.26	0.98	0.14	1	
0.58	0.14	0.1	0.09	0	
0.43	0.10	0.79	0.25	0	
0.12	0.20	0.80	0.25	0	d_2
0.42	0.17	0.57	0.22	0	
0.9	0.18	0.92	0.09	1	
0.29	0.28	0.37	0.17	0	
0.46	0.22	0.25	0.10	0	
0.96	0.09	0.52	0.16	1	d_3
0.24	0.09	0.76	0.27	0	
0.14	0.14	0.61	0.09	0	
0.73	0.24	0.31	0.14	0	
0.41	0.27	0.35	0.22	0	
0.61	0.09	0.26	0.22	0	d_4
0.39	0.06	0.64	0.07	1	
0.05	0.24	0.97	0.12	0	
0.19	0.16	0.12	0.27	1	
0.16	0.25	0.60	0.15	1	
0.49	0.12	0.13	0.28	1	d_5
0.98	0.08	0.86	0.13	0	
0.43	0.22	0.86	0.12	0	
0.68	0.19	0.51	0.14	0	
0.95	0.22	0.09	0.18	1	

После лингвистической интерпретации термов, которым соответствуют функции принадлежности из табл. 4.6, получим нечеткую базу знаний, состоящую из следующих правил:

Таблица 4.6

Параметры и функции принадлежности синтезированных правил

ЕСЛИ						ТО
x_1			x_2			y
параметры термов		функции принадлежности	параметры термов		функции принадлежности	
b	c		b	c		
0.12	0.05		0.89	0.09		d_1
0.13	0.26		0.98	0.14		
0.9	0.18		0.92	0.09		d_2
0.96	0.09		0.52	0.16		d_3
0.39	0.06		0.64	0.07		d_4
0.19	0.16		0.12	0.27		
0.16	0.25		0.60	0.15		
0.49	0.12		0.13	0.28		d_5
0.95	0.22		0.09	0.18		

ЕСЛИ $x_1 = \text{около } 0.12$ И $x_2 = \text{около } 0.95$ ТО $y \in d_1$

ЕСЛИ $x_1 = \text{около } 0.9$ И $x_2 = \text{около } 0.9$ ТО $y \in d_2$

ЕСЛИ $x_1 = \text{около } 0.95$ И $x_2 = \text{около } 0.5$ ТО $y \in d_3$

ЕСЛИ $x_1 = \text{около } 0.4$ И $x_2 = \text{около } 0.65$

ИЛИ $x_1 = \text{около } 0.2$ И $x_2 = \text{около } 0.12$

ИЛИ $x_1 = \text{около } 0.15$ И $x_2 = \text{около } 0.6$ ТО $y \in d_4$

ЕСЛИ $x_1 = \text{около } 0.5$ И $x_2 = \text{около } 0.13$

ИЛИ $x_1 = \text{около } 0.95$ И $x_2 = \text{около } 0.1$ ТО $y \in d_5$.

В результате применения извлеченной нечеткой базы знаний для классификации выборки пар входов были получены результаты, представленные на рис. 4.5, где различными символами обозначается тот или иной класс выхода, которому поставила в соответствие данную пару входов извлеченная модель.

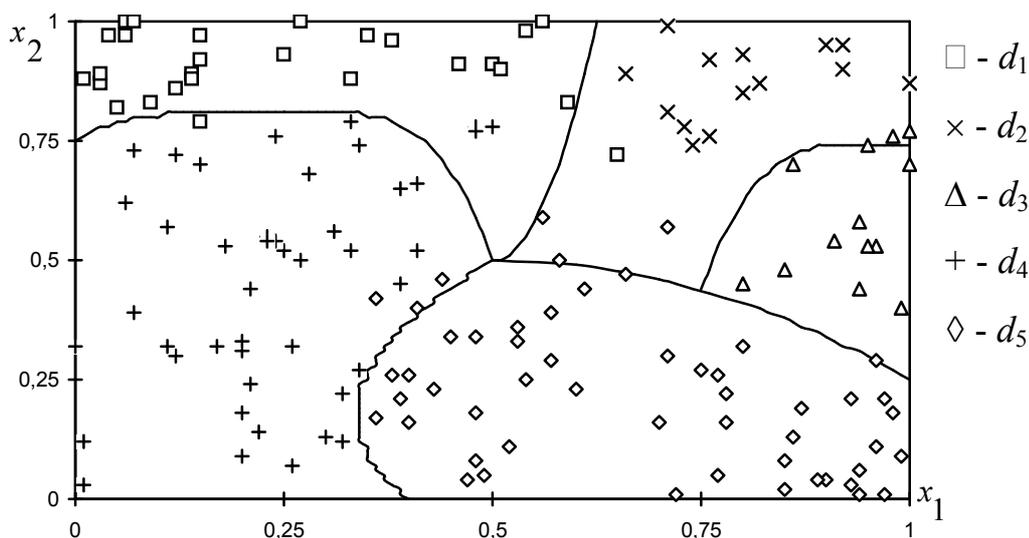


Рис. 4.5. Геометрический образ модели объекта «два входа - один выход» с дискретным выходом 1-го типа.

4.2.2. Объект «два входа - один выход» 2-го типа.

Объект «два входа - один выход» 2-го типа также характеризуется дискретными значениями выхода. При этом от предыдущего объекта он отличается лишь границами классов выходной переменной, которые заданы следующими аналитическими выражениями:

$$\textcircled{1}: x_2 \in [0.5, 0.75], x_1 = \frac{1}{2} \sqrt[3]{4(0.75 - x_2)} + 0.25;$$

$$\textcircled{2}: x_2 \in [0.25, 0.75], x_1 = (x_2 - 0.5)^2 + 0.1875;$$

$$\textcircled{3}: x_1 \in [0.25, 0.75], x_2 = 0.5 - 0.0625(4(0.75 - x_1))^2;$$

$$\textcircled{4}: x_1 \in [0.25, 0.5], x_2 = 0.25(4(x_1 - 0.25))^{0.6} + 0.75;$$

$$\textcircled{5}: x_1 \in [0, 0.1875], x_2 = 0.25 \left(1 - \frac{16}{3} x_1 \right)^{0.34} + 0.5;$$

$$\textcircled{6}: x_1 \in [0.25, 0.5], x_2 = 0.25 - 0.125 \sqrt{x_1 - 0.25};$$

$$\textcircled{7}: x_1 \in [0.5, 0.8125], x_2 = 0.1875 + 0.25 \left(1 - 0.0625^{\frac{4}{5}(x_1 - 0.5)} \right);$$

$$\textcircled{8}: x_1 \in [0.5, 1], x_2 = 0.3125 + 0.125 \left(\frac{16}{5} (x_1 - 0.8125) \right).$$

Вид двумерного пространства, разделенного на пять областей - классов решений, которые соответствуют характеристикам объекта 2-го типа, показан на рис. 4.6.

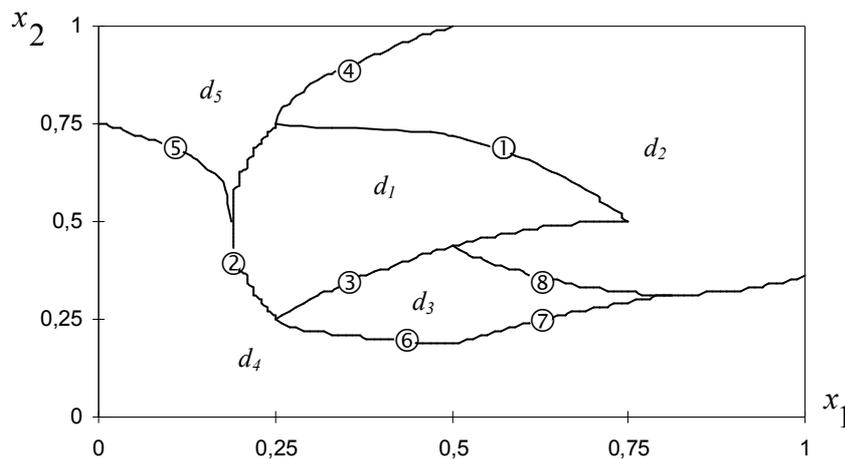


Рис. 4.6. Объект «два входа - один выход» с дискретным выходом 2-го типа.

Постановка задачи для данного эксперимента абсолютно аналогична той, что была сформулирована в п. 4.3.1.

В результате применения для решения данной задачи генетического алгоритма оптимизации были получены параметры матрицы знаний, которые представлены в табл. 4.7.

В табл. 4.8 приведены параметры функций принадлежности правил с единичными весовыми коэффициентами, а также показаны формы функций принадлежности термов входных переменных.

После лингвистической интерпретации термов, которым соответствуют функции принадлежности из табл. 4.8, получим нечеткую базу знаний, состоящую из следующих правил:

Таблица 4.7

Параметры синтезированной матрицы знаний

ЕСЛИ				вес	ТО
x_1		x_2			
b	c	b	c		y
0.61	0.29	0.05	0.27	0	d_1
0.89	0.25	0.37	0.15	0	
0.80	0.12	0.11	0.08	0	
0.44	0.17	0.62	0.09	1	
0.47	0.22	0.48	0.06	1	
0.42	0.11	0.40	0.07	0	d_2
0.88	0.20	0.66	0.28	1	
0.16	0.28	0.88	0.17	0	
0.86	0.22	0.59	0.11	1	
0.97	0.22	0.87	0.28	1	
0.10	0.21	0.34	0.19	0	d_3
0.45	0.16	0.31	0.08	1	
0.13	0.07	0.81	0.23	0	
0.09	0.21	0.41	0.24	0	
0.77	0.29	0.88	0.06	0	
0.43	0.24	0.09	0.17	1	d_4
0.14	0.11	0.23	0.25	1	
0.85	0.24	0.03	0.24	0	
0.32	0.10	0.02	0.09	1	
0.95	0.22	0.16	0.06	1	
0.86	0.11	0.84	0.16	0	d_5
0.18	0.05	0.77	0.17	1	
0.06	0.12	0.85	0.12	1	
0.31	0.26	0.93	0.09	0	
0.68	0.09	0.43	0.15	0	

ЕСЛИ $x_1 = \text{около } 0.45$ И $x_2 = \text{около } 0.6$

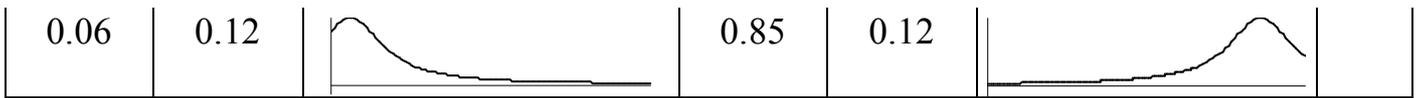
ИЛИ $x_1 = \text{около } 0.45$ И $x_2 = \text{около } 0.5$ ТО $y \in d_1$

ЕСЛИ $x_1 = \text{около } 0.9$ И $x_2 = \text{около } 0.65$

Таблица 4.8

Параметры и функции принадлежности синтезированных правил

ЕСЛИ						ТО
x_1			x_2			y
параметры термов		функции принадлежности	параметры термов		функции принадлежности	
b	c		b	c		
0.44	0.17		0.62	0.09		d_1
0.47	0.22		0.48	0.06		
0.88	0.20		0.66	0.28		d_2
0.86	0.22		0.59	0.11		
0.97	0.22		0.87	0.28		
0.45	0.16		0.31	0.08		d_3
0.43	0.24		0.09	0.17		d_4
0.14	0.11		0.23	0.25		
0.32	0.10		0.02	0.09		
0.95	0.22		0.16	0.06		
0.18	0.05		0.77	0.17		d_5



- ИЛИ $x_1 = \text{около } 0.95$ И $x_2 = \text{около } 0.9$ ТО $y \in d_2$
- ЕСЛИ $x_1 = \text{около } 0.45$ И $x_2 = \text{около } 0.3$ ТО $y \in d_3$
- ЕСЛИ $x_1 = \text{около } 0.45$ И $x_2 = \text{около } 0.1$
- ИЛИ $x_1 = \text{около } 0.15$ И $x_2 = \text{около } 0.25$
- ИЛИ $x_1 = \text{около } 0.3$ И $x_2 = \text{около } 0$
- ИЛИ $x_1 = \text{около } 0.95$ И $x_2 = \text{около } 0.15$ ТО $y \in d_4$
- ЕСЛИ $x_1 = \text{около } 0.2$ И $x_2 = \text{около } 0.75$
- ИЛИ $x_1 = \text{около } 0.05$ И $x_2 = \text{около } 0.85$ ТО $y \in d_5$.

В результате применения извлеченной нечеткой базы знаний для классификации выборки пар входов были получены результаты, приведенные на рис. 4.7.

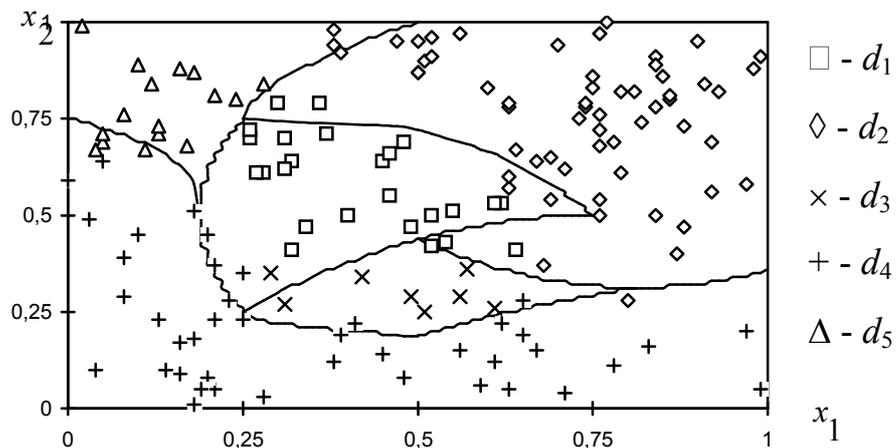


Рис. 4.7. Модель объекта с дискретным выходом 2-го типа.

Для лингвистической интерпретации функций принадлежности, синтезированных в вышеописанных экспериментах, могут быть использованы и другие термы, например, «низкий», «средний», «высокий» и т. д.

Дальнейшее повышение точности полученных лингвистических моделей возможно за счет их тонкой настройки с помощью методов, подробно изложенных в работах [26, 55, 56, 64, 86] и в главе 3.

ГЛАВА 5

ПРИМЕНЕНИЕ НЕЧЕТКИХ БАЗ ЗНАНИЙ ДЛЯ ПРОГНОЗИРОВАНИЯ В ЭЛЕКТРОЭНЕРГЕТИКЕ

Данная глава посвящена решению задачи прогнозирования суточных графиков потребления электрической мощности с помощью нечетких баз знаний, для проектирования которых применяется метод извлечения знаний на базе генетических алгоритмов оптимизации.

5.1. Общая характеристика задачи

Актуальность проблемы энергосбережения как в рамках всего народнохозяйственного комплекса Украины, так и для отдельных потребителей на местах, не вызывает сомнений. Решение задач энергосбережения в системах электропитания невозможно без надежного коротко- и долгосрочного прогнозирования электропотребления [81]. Результаты такого прогнозирования выступают в качестве необходимой базовой информации для многих задач энергосбережения.

Организация прогнозов электропотребления усложняется тем, что процесс электропотребления не является детерминированным и не всегда статистически устойчив [80]. Неопределенность этого процесса в особенно проявляется в условиях энергетического и финансового кризиса. Также на уровень электропотребления влияет большое количество факторов, таких как технологические и экономические характеристики потребителей, условия окружающей среды, социальные и демографические особенности региона и т.п.

Трудность построения прогнозирующих моделей состоит в том, что между внешними влияющими факторами существуют внутренние взаимосвязи, которые полностью не выявлены и сложно поддаются формализации. То же можно сказать и об исследовании связей между влияющими факторами и величи-

ной нагрузки. Среди различных методик, которые предлагались для решения задачи прогнозирования суточных графиков потребления электрической мощности, следует отметить моделирование методами экстраполяции временных рядов и регрессионного анализа [7]. Недостатком таких моделей является необходимость для их построения большого количества статистической информации по многим взаимосвязанным технологическим, коммерческим, экологическим и социальным факторам, которая не всегда имеется в наличии. Неплохие результаты были достигнуты при применении имитационной модели [17], построенной методом самоорганизации на основе МГУА - метода группового учета аргументов [22-25], однако данная модель не была приспособлена для учета текущих значений влияющих факторов. При моделировании методами теории вероятностей [19] необходимым условием является статистическая устойчивость параметров прогнозирования (как, собственно, и для предыдущих двух методов), которое в условиях действия нестационарных факторов сложно удовлетворить. То же касается и методики моделирования графиков энергонагрузок с помощью линейного периодического случайного процесса [49]. Комплексная методика авторов работы [9], которая использует методы математической статистики и положена в основу специализированного программного обеспечения, также характеризуется аналогичными недостатками. Для решения задачи был предложен также подход, основанный на использовании нейронных сетей [15]. Основным преимуществом такого подхода по сравнению с другими является возможность непосредственного прогноза суточных графиков электрических нагрузок с учетом заданных влияющих факторов (тип суток, сезонность, среднесуточная температура воздуха, переход на летнее/зимнее время) с перспективой на десять суток, что обеспечивается предварительным обучением прогнозирующей нейронной сети с помощью обучающей выборки суточных графиков годовой предыстории. Такой подход позволяет обойти трудности описания статистических закономерностей традиционными методами, однако при этом имеет и ряд недостатков. Во-первых, для обучения нейронной сети

применяется упоминавшийся выше алгоритм back-propagation [93], что не позволяет решить проблему локального оптимума. Во-вторых - некоторая громоздкость модели, обусловленная большим количеством нейронов, которые обеспечивают разные комбинации влияющих факторов. В-третьих, вызывает сомнение степень доверия к прогнозным графикам, полученным в виде детерминированных временных рядов. Поскольку даже очень совершенная модель не в состоянии учесть все факторы, влияющие на точность прогноза, то такая форма прогноза не обеспечивается инструментарием оценки его погрешности.

Подытожив вышеизложенный анализ, можно выделить такие основные проблемы, присущие задаче прогнозирования суточных графиков потребления электрической мощности:

1) практическое отсутствие статистической устойчивости для данных электропотребления в современных условиях, что ставит под сомнение степень доверия к моделям, построенным методами теории вероятностей и классической математической статистики;

2) недостаточную достоверность представления результатов прогнозирования в виде детерминированных графиков электрических нагрузок, поскольку многие влияющие факторы имеют качественную природу и сложно поддаются формализации при применении традиционных методов моделирования.

Решение вышеописанной проблемы было решено получить путем применения методики извлечения знаний из экспериментальных данных, что представляет собой принципиально новый подход к решению задачи прогнозирования. Применение данного подхода дает возможность устранить ряд недостатков, характерных для традиционных методов. Во-первых, нечеткие множества являются удобным средством формализации величин, имеющих качественную природу, и их применение для формирования нечеткого прогноза наиболее приемлемо в условиях неопределенности параметров прогнозирования. Во-вторых, большое значение имеет эффективность модели прогнозирования и достоверность прогноза, что требует выбора соответствующих методов для ее

построения. В связи с этим, генетические алгоритмы, которые являются воплощением методов случайного поиска и характеризуются соответствующими преимуществами (преодоление проблем локального оптимума и «проклятия размерности»), представляют собой эффективный инструмент проектирования нечетких баз знаний (в качестве моделей коротко- и долгосрочного прогнозирования электропотребления) путем их извлечения из экспериментальных данных.

5.2. Структура модели прогнозирования

Общая структура предложенной модели процесса прогнозирования показана на рис. 5.1. Первый этап предусматривает формирование нечеткой базы знаний на основе имеющихся экспериментальных данных по суточным графикам потребления электрической мощности за минувший период и внешних влияющих факторов. На уровень и характер потребления электрической мощности влияет очень большое количество факторов, которые во всей совокупности учесть невозможно. По результатам предварительного анализа были отобраны лишь те факторы, влияние которых на уровень электропотребления наиболее существенно и для которых существует реальная возможность сбора экспериментальных данных:

1) *тип дня*, представляется лингвистической переменной $d \in \{\text{день после выходного или праздника, середина недели, день перед выходным или праздником, выходной (праздник)}\}$; влияние данного фактора проявляется в том, что при прочих равных условиях электропотребление в выходные (праздничные) дни являются наименьшим, со следующего дня после выходного или праздника начинает нарастать, максимальное потребление наблюдается в середине недели, и в конце недели или накануне праздника идет на спад;

2) *сезон естественного освещения*, представляется лингвистической переменной $L \in \{\text{зимний (ноябрь-январь), весенний (февраль-апрель), летний (май-июль), осенний (август-октябрь)}\}$; данный фактор влияет на длительность

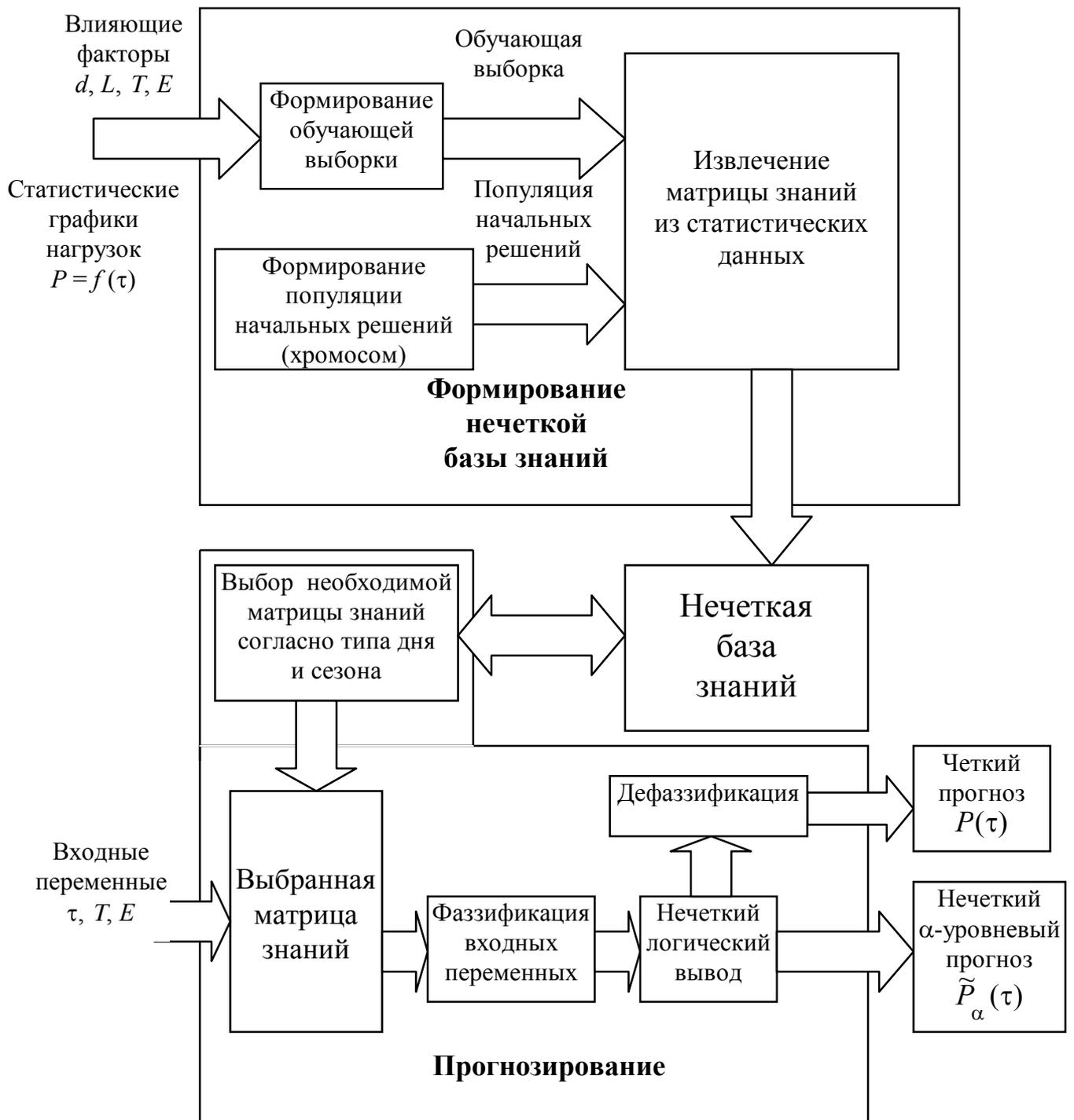


Рис. 5.1. Общая структура процесса прогнозирования.

светового дня, которая в свою очередь обуславливает неодинаковую потребность в искусственном освещении в разные времена года, что влияет на уровень электропотребления; в данном случае следует обратить внимание на тот факт, что в электроэнергетике сезоны, созвучные с временами года, по месячному составу немного отличаются;

3) *среднесуточная температура воздуха*, представляется соответствующей действительной числовой переменной T , °С; влияние этого фактора определяется зависимостью электропотребления от необходимости применения дополнительных средств электрообогрева или наоборот - электроохлаждения;

4) *облачность (естественная освещенность)*, представляется соответствующей действительной числовой переменной $E \in [0,10]$, балл; в зависимости от данного фактора изменяется продолжительность использования дополнительного искусственного освещения в течение суток.

Если проанализировать природу данных факторов влияния, то можно сделать вывод, что параметры d и L носят дискретный характер, а величины T и E являются непрерывными, однако их значения можно указать лишь приближенно. При таких условиях осуществляемый прогноз также может носить лишь приближенный (или нечеткий) характер. Учитывая данные обстоятельства, можно сделать вывод, что для формализации соответствующей модели прогнозирования имеет смысл воспользоваться нечеткими базами знаний.

Однако в данном случае мы имеем большое количество экспериментальных данных разной природы и разных по физическому смыслу. В связи с этим довольно сложно найти эксперта, который смог бы сформировать всю необходимую совокупность нечетких логических правил, положенных в основу нечеткой базы знаний как ядра прогнозирующей модели. В такой ситуации возникает необходимость решения задачи извлечения совокупности нечетких логических правил, которые наилучшим образом будут отображать закономерности связей «входы-выход» на множестве имеющихся экспериментальных данных, без участия эксперта. Для решения этой задачи следует применить метод, основанный на генетических алгоритмах оптимизации.

Охарактеризуем более подробно последовательность действий на каждом этапе процесса прогнозирования.

5.2.1. Построение нечеткой базы знаний.

Формирование нечеткой базы знаний осуществляется путем реализации определенной последовательности процедур, первой из которых является процедура *формирования обучающей выборки*. При помощи этой процедуры производится подготовка необходимого массива экспериментальных данных, что включает в себя:

- сортировку временных рядов суточных графиков нагрузок $P_{\text{эксп}} = f(\tau)$ (τ - время суток, $\tau \in [1, 24]$, ч) и соответствующих им метеорологических показателей (T и E) в массивы данных, которые соответствуют определенным комбинациям характеристик дня d и сезона L . Поскольку области определения лингвистических переменных d и L состоят из четырех термов каждая, то очевидно, что всего отдельных выборок-массивов будет 16;

- анализ сформированного массива данных на предмет наличия графиков, которые соответствуют идентичным метеорологическим условиям, и приведения таких графиков к одному по определенному критерию.

Второй процедурой принятой последовательности является *формирование популяции начальных решений*. В еще не синтезированной модели прогнозирования заведомо неизвестны ни количество нечетких правил, ни их содержание. Поэтому, во-первых, необходимо задаться определенными ограничениями на количество согласно п. 2.5.2. Во-вторых, содержание нечетких правил будет определяться переменными параметрами функций принадлежности термов входных переменных, для оценки которых воспользуемся функцией (2.24). Данный вид функции принадлежности используется из следующих соображений. Фаззификации в нашей задаче подлежат величины T и E , погрешность определения которых можно считать одинаковой в обе стороны от установленного значения. С этой точки зрения применение симметричных функций принадлежности для фаззификации данных величин вполне логично. Выражение (2.24) соответствует указанному свойству, а кроме того, оно довольно простое и удобное с точки зрения упрощения алгоритма расчетов.

Таким образом, действуя аналогично п. 2.5.2, сформируем случайным образом набор переменных параметров модели прогнозирования (переменных параметров функций принадлежности и весов правил) с учетом ограничений на их диапазоны и сведем их в соответствующую матрицу знаний (табл. 5.1).

Таблица 5.1

Матрица знаний модели прогнозирования мощности

Номер правила	ЕСЛИ			Вес	ТО
	$\tau, \text{ч}$	$T, \text{°C}$	$E, \text{балл}$		$P, \text{МВт}$
11	(b_1^{11}, c_1^{11})	(b_2^{11}, c_2^{11})	(b_3^{11}, c_3^{11})	w_{11}	<i>около</i>
...	\bar{P}_1
$1k_1$	$(b_1^{1k_1}, c_1^{1k_1})$	$(b_2^{1k_1}, c_2^{1k_1})$	$(b_3^{1k_1}, c_3^{1k_1})$	w_{1k_1}	
...	
$j1$	(b_1^{j1}, c_1^{j1})	(b_2^{j1}, c_2^{j1})	(b_3^{j1}, c_3^{j1})	w_{j1}	<i>около</i>
...	\bar{P}_j
jk_j	$(b_1^{jk_j}, c_1^{jk_j})$	$(b_2^{jk_j}, c_2^{jk_j})$	$(b_3^{jk_j}, c_3^{jk_j})$	w_{jk_j}	
...	
$m1$	(b_1^{m1}, c_1^{m1})	(b_2^{m1}, c_2^{m1})	(b_3^{m1}, c_3^{m1})	w_{m1}	<i>около</i>
...	\bar{P}_m
mk_m	$(b_1^{mk_m}, c_1^{mk_m})$	$(b_2^{mk_m}, c_2^{mk_m})$	$(b_3^{mk_m}, c_3^{mk_m})$	w_{mk_m}	

В данной таблице $\bar{P}_1 \dots \bar{P}_m$ - центры подинтервалов, на которые разбивается область значений выходной величины P - активной нагрузки, МВт:

$$[P_{\min}, P_{\max}] = \underbrace{[P_{\min}, P_1]}_{\bar{P}_1} \cup \underbrace{[P_1, P_2]}_{\bar{P}_2} \cup \dots \cup \underbrace{[P_{m-1}, P_{\max}]}_{\bar{P}_m}. \quad (5.1)$$

С целью применения для решения задачи извлечения нечетких правил методики генетических алгоритмов полученную матрицу знаний следует закодировать в хромосому (рис. 5.2) и сгенерировать аналогичным образом начальную популяцию таких хромосом.

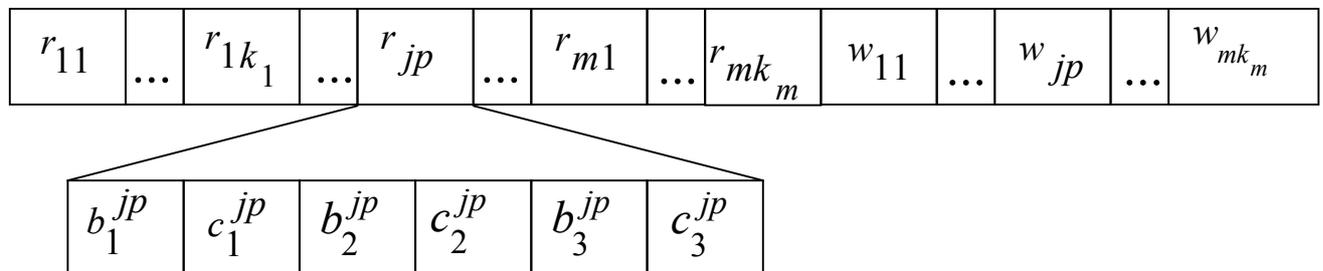


Рис. 5.2. Хромосома-код переменных параметров модели прогнозирования

Следующая процедура формирования нечеткой базы знаний - применение методики генетических алгоритмов согласно п. 2.6.7 для *извлечения матрицы знаний из экспериментальных данных*. В качестве базового материала для решения этой задачи используется выборка экспериментальных данных, выбранная в зависимости от сезона и характера дня, на который планируется строить прогноз, и сформированная начальная популяция хромосом.

Полученная на данном этапе хромосома будет представлять собой оптимальное (субоптимальное) решение и воплощать в себе матрицу знаний, которой соответствует совокупность нечетких логических правил ЕСЛИ (входы) - ТО (выход), формализующих связи между входными переменными (τ , T , E) и значениями нагрузок P . А в целом нечеткая база знаний будет представлять собой совокупность матриц знаний, при этом для проведения прогноза на основе алгоритма, который приводится ниже, будет выбираться определенная, в зависимости от сезона и характера дня.

5.2.2. Прогнозирование суточных графиков потребления электрической мощности.

Прогнозирование суточных графиков потребления электрической мощности на основе модели, синтезированной на предыдущем этапе, осуществляется

в такой последовательности:

1) определяется тип дня, на который проводится прогноз, сезон, и выбирается соответствующая матрица знаний;

2) входным величинам E и T присваиваются соответственно значения среднесуточной облачности и температуры по данным метеопрогноза (E_{np} и T_{np}), а $\tau \in [1, 24]$, ч;

3) рассчитываются значения функций принадлежности входных переменных $\mu^{jp}(E_{np})$, $\mu^{jp}(T_{np})$, $\mu^{jp}(\tau)$ (фаззификация) согласно (2.24);

4) определяются степени принадлежности выходной величины P к соответствующим классам $\mu^{\bar{P}_1}(P), \mu^{\bar{P}_2}(P), \dots, \mu^{\bar{P}_m}(P)$ (нечеткий логический вывод) по формуле:

$$\mu^{\bar{P}_j}(P) = \max_{p=1, k_j} \left\{ w_{jp} \min \left[\mu^{jp}(\tau), \mu^{jp}(T_{np}), \mu^{jp}(E_{np}) \right] \right\}, \quad j = \overline{1, m}; \quad (5.2)$$

5) выдается конечный прогноз P_{mod} на каждый час, который рассчитывается методом дефаззификации:

$$P_{mod} = \frac{\bar{P}_1 \mu^{\bar{P}_1}(P) + \bar{P}_2 \mu^{\bar{P}_2}(P) + \dots + \bar{P}_m \mu^{\bar{P}_m}(P)}{\mu^{\bar{P}_1}(P) + \mu^{\bar{P}_2}(P) + \dots + \mu^{\bar{P}_m}(P)}. \quad (5.3)$$

5.3. Алгоритм реализации процесса прогнозирования

5.3.1. Обучающая выборка.

В соответствии с вышеописанной структурой модели процесса прогнозирования, определяющим этапом является формирование нечеткой базы знаний. Первая процедура реализации этого этапа - формирование обучающей выборки

- выборки экспериментальных данных за минувший период с целью ее анализа и извлечения закономерностей-правил, которые лягут в основу модели прогнозирования потребления электрической мощности.

Обучающая выборка представляет собой массив экспериментальных данных «входы - выход», который можно представить в виде табл. 5.2.

Таблица 5.2

Структура обучающей выборки экспериментальных данных

Сутки (№ п/п)	Входы			Выход
	$T, ^\circ\text{C}$	$E, \text{балл}$	$\tau, \text{ч}$	$P, \text{МВт}$
1	T_1	E_1	1	P_1^1
			2	P_2^1
		
			24	P_{24}^1
...
M	T_M	E_M	1	P_1^M
			2	P_2^M
		
			24	P_{24}^M

В этой таблице T_p и E_p , $p = \overline{1, M}$ - значения соответственно среднесуточной температуры и облачности, взятые из архивных ведомостей Винницкого центрального гидрометеобюро, $P_1^p - P_{24}^p$ - почасовые значения суточных графиков потребления электрической мощности, взятые из суточных ведомостей предприятия «Винницкие центральные высоковольтные электрические сети».

5.3.2. Предварительный анализ экспериментальных данных.

Структурированную вышеприведенным способом обучающую выборку необходимо предварительно проанализировать с целью:

- формирования обобщенных суточных графиков потребления электрической мощности, которые соответствуют идентичным метеорологическим условиям;

- определения диапазона температур, характерных для того или иного сезона, а также интервала почасовых показателей потребления электрической мощности $[P_{\min}, P_{\max}]$.

Данный анализ нужно проводить по следующим причинам. На практике почти невозможно, чтобы идентичным погодным влияющим факторам соответствовали также идентичные графики потребления электрической мощности, что является следствием влияния побочных факторов, не все из которых возможно учесть. Поэтому с целью устранения данных расхождений необходимо проанализировать обучающую выборку на предмет наличия графиков, которые соответствуют идентичным погодным влияющим факторам, и заменить каждую группу таких графиков одним обобщенным графиком.

Признаки идентичности графиков нагрузок соответствуют так называемым *зонам нечувствительности*, определенным на шкале температуры и облачности, которые в свою очередь привязаны к определенным диапазонам значений. Изменение внутри данных диапазонов той или иной величины практически не влияет на величину потребления электрической мощности (на конкретный час суток). На основании опыта экспертов, для решения поставленной задачи были приняты такие зоны нечувствительности. Для среднесуточной облачности область определения $[0, 10]$ баллов разбивается на интервалы по 2 балла. Для среднесуточной температуры существует определенный «порог», выше которого потребление электрической мощности менее чувствительно к изменению температуры, вместе с тем при температуре ниже этого «порога» чувствительность заметно возрастает. Таким пороговым значением, по оценкам экспертов, было принято $10\text{ }^{\circ}\text{C}$, в соответствии с чем были установлены следующие зоны нечувствительности: для температур $10\text{ }^{\circ}\text{C}$ и выше - $3\text{ }^{\circ}\text{C}$ (то есть

имеем интервалы [10, 13) °С, [13, 16) °С и т. д.), для температур ниже 10 °С - 2 °С (соответственно интервалы [8, 10) °С, [6, 8) °С и т. д.).

Алгоритм определения в обучающей выборке групп графиков потребления электрической мощности, соответствующих идентичным с точностью до установленных зон их нечувствительности влияющим метеофакторам, и замены их одним обобщенным предусматривает выполнение следующих шагов.

ШАГ 1. Зафиксировать значения среднесуточной температуры и облачности, которые соответствуют первому графику потребления электрической мощности из обучающей выборки.

ШАГ 2. Определить, в какие зоны нечувствительности попадает данный график согласно соответствующих ему значений T и E .

ШАГ 3. Зафиксировать значения T и E , соответствующие следующему графику.

ШАГ 4. ЕСЛИ данному графику соответствуют аналогичные зоны нечувствительности согласно T и E , ТО отметить данный график.

ШАГ 5. ЕСЛИ обучающая выборка исчерпана, ТО перейти на ШАГ 6, ИНАЧЕ перейти на ШАГ 3.

ШАГ 6. Заменить первый из отмеченных графиков на обобщенный среднеквадратичный график, почасовые значения потребления электрической мощности у которого рассчитываются следующим образом:

$$P_j = \sqrt{\frac{\sum_{i=1}^n P_{ij}^2}{n}}, \quad j = \overline{1, 24}, \quad (5.4)$$

где n - количество отмеченных графиков. Удалить из обучающей выборки остальные отмеченные графики.

ШАГ 7. ЕСЛИ обучающая выборка исчерпана, ТО - КОНЕЦ АЛГОРИТМА, ИНАЧЕ зафиксировать значения T и E , соответствующие следующему

графику потребления электрической мощности из обучающей выборки, и перейти на ШАГ 2.

КОНЕЦ АЛГОРИТМА.

Большинству электротехнических процессов присущи квадратичные зависимости мощности от токов. Поэтому выбор формулы (5.4) для усреднения графиков представляется вполне адекватным.

Определение диапазона температур того или иного сезона, для прогнозирования на который синтезируется модель, также имеет смысл. Входными переменными для прогнозирующей модели, как показано выше, являются среднесуточная температура, облачность и время суток. Области определения двух последних величин не зависят от времени года, с температурой же ситуация иная. Если для этой величины принять какой-то универсальный диапазон (скажем, от $-30\text{ }^{\circ}\text{C}$ до $+30\text{ }^{\circ}\text{C}$), то в процессе выполнения генетического алгоритма на этапе извлечения матрицы знаний параметры b функций принадлежности (2.24) переменной T во время начальной инициализации популяции хромосом или мутации хромосом-отпрысков будут определяться случайным образом на всем этом диапазоне. Это приведет к исследованию бесперспективных областей, то есть областей температур, не характерных для данного сезона, что значительно снизит эффективность алгоритма. Поэтому в массиве среднесуточных температур из обучающей выборки следует определить минимальную (T_{\min}) и максимальную (T_{\max}) температуру и принять область определения переменной T для соответствующего сезона в диапазоне $[T_{\min}, T_{\max}]$.

Определение интервала $[P_{\min}, P_{\max}]$ - области значений выходной величины P (активная нагрузка, МВт) - необходимо для его разбивки на термы (классы) (5.1), и проводится аналогично определению интервала температур.

5.3.3. Генетический алгоритм извлечения матрицы знаний из обучающей выборки.

После того, как установлен сезон и день недели, для прогнозирования на которые планируется синтезировать модель, и по методу, описанному выше, сформирована соответствующая обучающая выборка, необходимо задаться следующими важными величинами для выполнения процедуры извлечения матрицы знаний путем применения генетического алгоритма: количество термов выходной переменной m ; максимально допустимое количество правил $\overline{k_j}$, которые соответствуют j -му классу выходной переменной, $j = \overline{1, m}$; границы диапазонов изменения параметров c (коэффициентов концентрации-растяжения) функций принадлежности входных переменных (2.24); размер начальной популяции хромосом-решений; коэффициент скрещивания p_c (доля хромосом из популяции, подлежащих скрещиванию) и коэффициент (вероятность) мутации p_m ; количество итераций генетического алгоритма.

На современном этапе развития теории нечетких баз знаний и генетических алгоритмов не существует общепринятых критериев для установления вышеназванных параметров, и эта проблема требует дополнительных исследований. Фактически все эти параметры должны удовлетворять двум основным требованиям, которые конфликтуют между собой - скорость выполнения генетического алгоритма и точность результатов. В результате проведения значительного количества экспериментов были определены наиболее оптимальные с точки зрения эффективности алгоритма и точности прогноза значения этих величин. Так, предварительные исследования обучающих выборок показали, что величина диапазона $[P_{\min}, P_{\max}]$ для каждого сезона и типа дня приблизительно одинакова, и для предприятия электросетей, которое исследовалось, разница $P_{\max} - P_{\min} \approx 200$ Мвт. Это дает возможность установить одинаковое количество выходных термов (то есть разбить область значений выходной переменной на одинаковое количество классов) для всех моделей, подлежащих синтезу.

Учитывая требования диспетчерских служб, в соответствии с которыми интервал, соответствующий классу типа «около $\overline{P_j}$ », где $\overline{P_j}$ - центр соответствующего класса, $j = \overline{1, m}$, не должен превышать $15 \div 20$ МВт, принимаем количество термов выходной переменной $m = 13$. Остальным величинам присвоим такие значения: $\overline{k_j} = 5$; диапазон изменения параметров c функций принадлежности для переменной T : $c \in [1, 25]$, для переменной E : $c \in [0.1, 4]$, для переменной τ : $c \in [0.5, 8]$; размер начальной популяции хромосом-решений $K = 40$; коэффициенты $p_c = 0.25$ и $p_m = 0.17$. Количество итераций генетического алгоритма зависит от размера обучающей выборки, поскольку чем больше массив экспериментальных данных, тем сложнее отыскать в нем необходимые закономерности. Величины обучающих выборок, относящихся к различным типам дня и соответствующих одинаковому периоду, будут разными. Так, дней с характеристикой «начало (конец) рабочей недели» получается в среднем по одному в неделю, «середина рабочей недели» - по 3, а «выходной (праздник)» - по 2. Поэтому для построения соответствующих моделей установим такое количество итераций генетического алгоритма: на «начало (конец) рабочей недели» - 2000 итераций, на «середину рабочей недели» - 4000 итераций, на «выходной (праздник)» - 3000 итераций.

Таким образом, имеется весь необходимый материал для выполнения процедуры извлечения матрицы знаний с помощью генетического алгоритма, который приводится ниже.

ШАГ 1. Провести инициализацию начальной популяции хромосом-решений, рассчитав значения их генов по соотношениям:

$$w_{jp} = RANDOM\left(\left[\underline{w}, \overline{w}\right]\right), \quad b_{ijp} = RANDOM\left(\left[\underline{x_i}, \overline{x_i}\right]\right), \quad c_{ijp} = RANDOM\left(\left[\underline{c_i}, \overline{c_i}\right]\right), \quad (5.5)$$

где $j = \overline{1, 3}$ - номер термина выходной переменной;

$p = \overline{1, 5}$ - номер правила, которое соответствует j -му терму выходной переменной;

$\underline{w}(\overline{w})$ - нижняя (верхняя) граница интервала возможных значений весов правил, $[\underline{w}, \overline{w}] \in [0, 1]$;

$[\underline{b}_i, \overline{b}_i]$ - область определения входной переменной x_i , $i = \overline{1, 3}$ (для T , °C - $[T_{\min}, T_{\max}]$, для E , балл - $[0, 10]$, для τ , ч - $[1, 24]$);

$[\underline{c}_i, \overline{c}_i]$ - интервалы возможных значений коэффициентов концентрации-растяжения функций принадлежности, которые соответствуют термам-оценкам входной переменной x_i , $i = \overline{1, 3}$.

Напомним, что $RANDOM(\underline{\xi}, \overline{\xi})$ - операция определения случайного числа, равномерно распределенного на интервале $[\underline{\xi}, \overline{\xi}]$.

ШАГ 2. Рассчитать функции соответствия (fitness functions) каждой хромосомы, используя соотношение:

$$FF(S_{ch}) = - \sum_{l=1}^M [P_{\text{мод}}^l - P_{\text{эм}}^l]^2, \quad (5.6)$$

где $ch = \overline{1, K}$ - номер хромосомы S ;

$l = \overline{1, M}$ - номер элемента обучающей выборки;

$P_{\text{эм}}^l$ - значения показателя потребляемой электрической мощности на определенный час как элемент обучающей выборки.

Для расчета модельного значения выхода $P_{\text{мод}}^l$ выполняются операции фаззификации значений входных переменных T^l , E^l и τ^l как элементов обучающей выборки (формула (2.24)), нечеткого логического вывода (расчет степеней принадлежности выходной переменной P к собственным термам - формула (5.2)) и дефаззификации (расчет четкого значения выхода $P_{\text{мод}}^l$ - формула (5.3)).

ШАГ 3. ЕСЛИ получена хромосома с функцией соответствия $FF(S_{ch}) = 0$, ТО - КОНЕЦ АЛГОРИТМА, ИНАЧЕ - перейти на ШАГ 4.

ШАГ 4. Определить количество хромосом-родителей, которые будут отобраны для операции скрещивания, по формуле $p_c \cdot K$.

ШАГ 5. Для каждой хромосомы определить число p_{ch} , характеризующее вероятность отбора данной хромосомы для операции скрещивания:

$$p_{ch} = \frac{FF'(S_{ch})}{\sum_{j=1}^K FF'(S_j)}, \quad ch = \overline{1, K}, \quad \sum_{ch=1}^K p_{ch} = 1, \quad (5.7)$$

где

$$FF'(S_{ch}) = \max_{j=1, K} |FF(S_j) + FF(S_{ch})|. \quad (5.8)$$

ШАГ 6. Сгенерировать случайное число z , имеющее равномерный закон распределения на интервале $[0, 1]$, и выбрать в качестве родительской ту хромосому S_k , которая удовлетворяет соотношению (см. рис. 2.7):

$$z < \sum_{j=1}^k p_j, \quad z > \sum_{j=1}^{k-1} p_j, \quad k \leq K. \quad (5.9)$$

ШАГ 7. Повторить ШАГ 6 для определения второй родительской хромосомы - эта хромосома будет скрещиваться с предыдущей.

ШАГ 8. ЕСЛИ отобраны не все $p_c \cdot K$ хромосом-родителей, ТО повторить ШАГИ 6-7 для определения следующей пары хромосом-родителей.

ШАГ 9. Синтезировать $p_c \cdot K$ хромосом-отпрысков путем скрещивания хромосом-родителей в соответствующих парах по алгоритму п. 2.6.2.

ШАГ 10. Для каждого гена каждой хромосомы-отпрыска сгенерировать случайное число $z = RANDOM([0,1])$.

ШАГ 11. ЕСЛИ $z < p_m$, ТО провести мутацию соответствующего гена каждой хромосомы-отпрыска в соответствии с формулами (5.5).

ШАГ 12. Рассчитать функции соответствия новообразовавшихся хромосом-отпрысков по формуле (5.6).

ШАГ 13. Сократить полученную популяцию хромосом до предыдущего размера, удалив из нее $p_c \cdot K$ хромосом с наименьшими значениями функций соответствия.

ШАГ 14. ЕСЛИ получена хромосома с функцией соответствия $FF(S_{ch}) = 0$, ТО считать ее искомым (оптимальным) решением и перейти на КОНЕЦ АЛГОРИТМА, ИНАЧЕ - перейти на ШАГ 15.

ШАГ 15. ЕСЛИ количество итераций исчерпано, ТО считать искомым (субоптимальным) решением хромосому, имеющую наибольшее значение функции соответствия, и перейти на КОНЕЦ АЛГОРИТМА, ИНАЧЕ - перейти на ШАГ 5.

КОНЕЦ АЛГОРИТМА.

Таким образом, решающим признаком прекращения работы алгоритма и определения необходимого решения является прохождение предварительно определенного оптимального (для этой задачи) количества итераций алгоритма. При этом возможна ситуация, когда алгоритм поиска оптимального решения будет завершен раньше - при достижении функцией соответствия одной из хромосом своего оптимального значения.

Синтезированная таким образом модель дает нам возможность получать прогнозные варианты графиков потребления электрической мощности на определенный день недели и сезон на основании таких влияющих факторов, как прогнозируемая среднесуточная температура T_{np} и прогнозируемая среднесуточная облачность E_{np} , вместе с которыми в качестве значений третьей входной переменной будет использоваться временной ряд часов суток (τ , ч).

5.4. Результаты компьютерного моделирования

Представленные в данной работе методы и алгоритмы были положены в основу специализированного программного обеспечения АСПСГ (Автоматизированная система прогнозирования суточных графиков), основными функциями которой является синтез новых прогнозирующих моделей и перестройка старых, получение прогнозов на перспективу, а также обновление массивов экспериментальных данных. С помощью разработанной программы был проведен ряд компьютерных экспериментов. Эти исследования имели целью проверить адекватность прогнозов, полученных с помощью синтезированных на основе специально сформированной обучающей выборки моделей, путем их сравнения с данными специально сформированной тестовой выборки суточных графиков потребления электрической мощности и соответствующих им влияющих факторов. В обучающую выборку вошли данные за 1993-97 года, в тестовую - аналогичные данные за 2000 год. На рис. 5.3 представлен ряд диаграмм сравнения прогнозов с реальными суточными графиками.

Как можно видеть из этих диаграмм, в среднем прогнозы характеризуются достаточно высокой точностью. Существенные расхождения наблюдаются в отдельные часы, а для зимнего периода характерно завышение прогнозных показателей над реальными вследствие отключения потребителей в соответствии

————— - эталон
 - модель

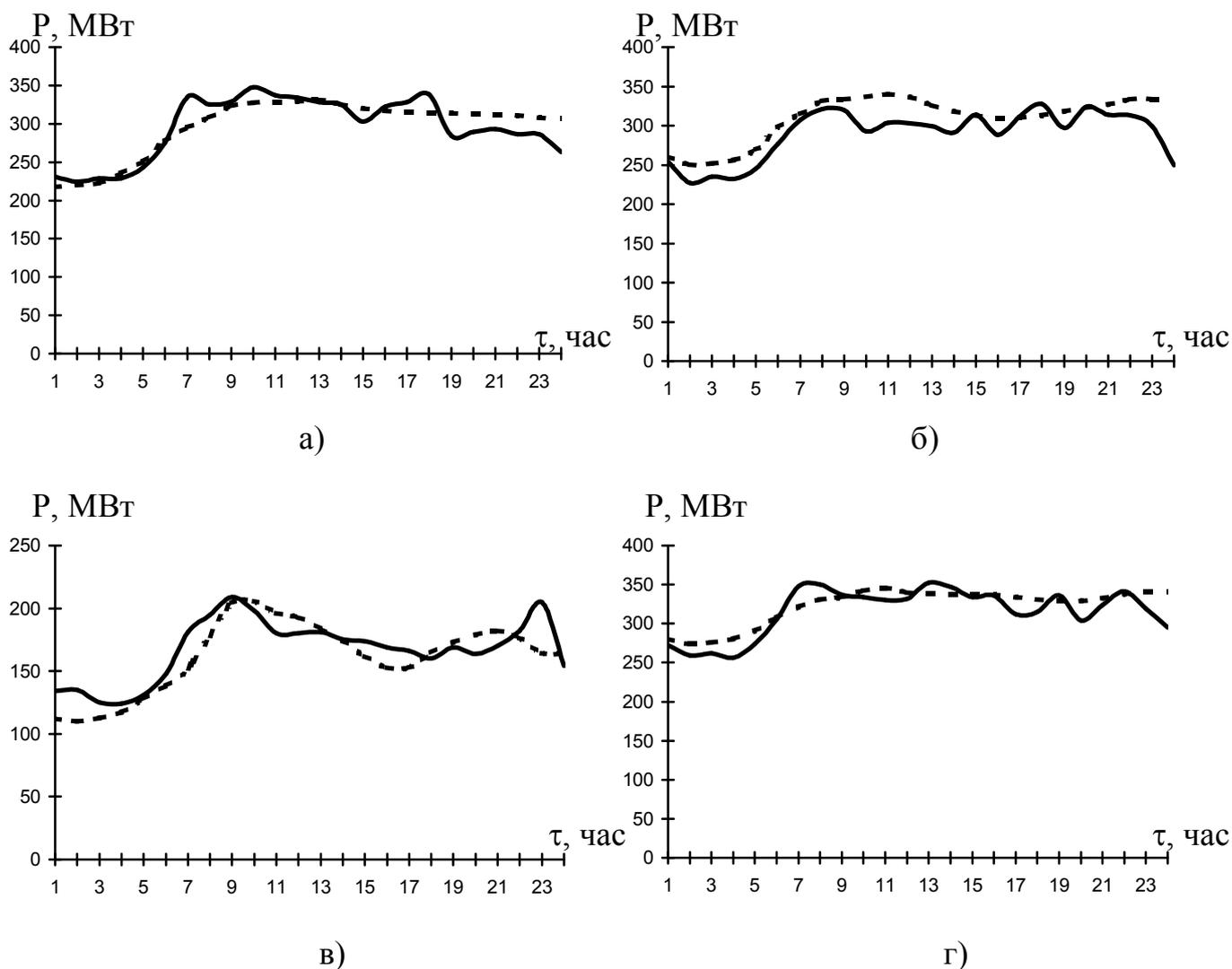


Рис. 5.3. Результаты компьютерных экспериментов:

- а - начало недели (весенний сезон);
- б - середина недели (зимний сезон);
- в - конец недели (летний сезон);
- г - выходной (зимний сезон).

с плановыми ограничениями потребления электроэнергии. Это подтверждает целесообразность организации прогнозов в виде не детерминированных, а зонных графиков. Так, например, зоны, которые характеризуют почасовые интервалы возможных значений потребления мощности, можно будет определять на некотором α -уровне нечеткого множества выходной переменной. Результаты проведенных исследований дают нам основания принять разработанную методику за основу для синтеза таких моделей.

ЗАКЛЮЧЕНИЕ

При идентификации сложных нелинейных зависимостей классическими методами не всегда существует возможность учета экспертной информации об исследуемом объекте и использования определенных нечетко или лингвистически заданных параметров, которые его характеризуют. Кроме того, в тех случаях, когда объект моделирования характеризуется лишь выборкой экспериментальных данных и для его идентификации невозможно привлечь экспертов, возникает проблема извлечения знаний, то есть выявления на основе анализа имеющейся статистики функциональных зависимостей между входными и выходными параметрами объекта.

Эта книга посвящена исследованиям в области идентификации объектов такого типа. При этом в качестве базового инструмента для решения поставленной задачи были комплексно использованы средства интеллектуальных технологий - нечеткие базы знаний, нейронные сети и генетические алгоритмы оптимизации. Нечеткие базы знаний служат весьма удобным средством для формализации логико-лингвистической информации об объекте моделирования, которой владеют эксперты. При отсутствии последних построение грубой нечеткой модели представляется возможным путем извлечения знаний из экспериментальных данных, характеризующих тот или иной объект. Дальнейшая тонкая настройка синтезированной модели производится путем обучения нейроподобной структуры, воплощающей в себя грубую нечеткую базу знаний - нейро-нечеткой сети.

Авторы считают, что перспективными направлениями дальнейших исследований в области развития методов идентификации сложных нелинейных зависимостей средствами интеллектуальных технологий могут быть следующие:

- 1) использование для моделирования сложных закономерностей нечетких логических правил, в которых лингвистические термы входных и выходных переменных оцениваются двумерными функциями принадлежности;

2) разработка генетических алгоритмов извлечения нечетких баз знаний для идентификации объектов, имеющих иерархическую структуру;

3) разработка методов извлечения нечетких баз знаний, а также их тонкой настройки на основании выборок экспериментальных данных, которые содержат нечеткие (размытые) значения входов и выходов.

ЛИТЕРАТУРА

1. Аведьян Э.Д. Алгоритмы настройки многослойных нейронных сетей // Автоматика и телемеханика. - 1995. - № 4 - С. 31-39.
2. Айзерман М.А., Браверман Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. - М.: Наука, 1970. - 384 с.
3. Алиев Р.А., Абдикеев Н.М., Шахназаров М.М. Производственные системы с искусственным интеллектом. - М.: Радио и связь, 1990. - 264 с.
4. Алиев Р.А., Церковный А.З., Мамедова Г.А. Управление производством при нечеткой исходной информации. - М.: Энергоатомиздат, 1991. - 240 с.
5. Борисов А.И., Вилюмс Э.Р., Сукур Л.Я. Диалоговые системы принятия решений на базе миниЭВМ. - Рига.: Зинатне, 1986. - 195 с.
6. Буяльская Т.Б., Мокин Б.И., Мокин А.Б. Нечеткие математические модели в задачах экспертизы художественных произведений // Праці Міжнар. конф. «Автоматика-2000». - Том 2. - Львів (Україна). - 2000. - С. 43-48.
7. Бэнн Д.В., Фармер Е.Д. Сравнительные модели прогнозирования электрической нагрузки: Пер. с англ. - М.: Энергоатомиздат, 1987. - 200 с.
8. Вопросы анализа и процедуры принятия решений: Сборник переводов / Под ред. И.Ф. Шахнова. - М.: Мир, 1976. - 228 с.
9. Воропаев Л.Г., Коневский М.Б., Мясников В.А., Демура А.В., Исаев К.Н., Надтока И.И., Седов А.В. Программное обеспечение краткосрочного прогнозирования электрической нагрузки энергосистемы // Изв. вузов. Электромеханика. - 1996. - № 3-4. - С. 119-120.
10. Гарднер М.Ф., Бэрнс Дж. Переходные процессы в линейных системах: Пер. с англ. - М.: Физматгиз, 1961. - 285 с.
11. Глушков В.М. Введение в АСУ. - К: Тэхніка, 1974. - 320 с.
12. Горелик А.Л., Скрипкин В.А. Методы распознавания. - М.: Высшая школа, 1977. - 224 с.

13. Гропп Д. Методы идентификации систем: Пер. с нем. - М.: Мир, 1979. - 199 с.
14. Дейч А.М. Методы идентификации динамических объектов. - М.: Энергия, 1979. - 225 с.
15. Демура А.В. Краткосрочное прогнозирование суточных графиков нагрузки на основе искусственных нейронных сетей // Изв. вузов. Электромеханика. - 1998. - № 2-3. - С. 69-71.
16. Джеффри Е. Хинтон. Как обучаются нейронные сети // В мире науки. - 1992.- № 11-12. - С.103-110.
17. Едемский С.Н. Прогнозирование электропотребления нагрузки на основе моделей с самоорганизацией // Изв. вузов. Энергетика. - 1990. - № 2. - С. 17-22.
18. Жупанова М.А., Кательников Д.И., Кудрик В.В. Нечеткая экспертная система для психо-физиологического тестирования человека-оператора // Тези доповідей IV-ї Міжнародної науково-технічної конференції «Контроль і управління в технічних системах» (КУТС-97). - Том 3. - Вінниця: УНІВЕРСУМ.- 1997. - С. 162-168.
19. Забелло Е.П. Возможный алгоритм оперативного прогноза электрических нагрузок // Изв. вузов. Энергетика. - 1990. - № 1. - С. 36-39.
20. Заде Л. Понятие лингвистической переменной и ее применение к принятию приближенных решений: Пер. с англ. - М.: Мир, 1976. - 167 с.
21. Згуровский М.З. Интегрированные системы оптимального управления и проектирования. - К.: Вища школа, 1990. - 348 с.
22. Ивахненко А.Г. Индуктивный метод самоорганизации моделей сложных систем. - К.: Наукова думка, 1982. - 296 с.
23. Ивахненко А.Г., Лапа В.Г. Предсказание случайных процессов. - К: Наукова думка, 1971. - 416 с.
24. Ивахненко А.Г., Мюллер Й.А. Самоорганизация прогнозирующих моделей.- К.: Техніка, 1985. - 223 с.

25. Ивахненко А.Г., Юрачковский Ю.П. Моделирование сложных систем по экспериментальным данным. - М.: Радио и связь, 1987. - 120 с.
26. Кательников Д.И. Разработка метода идентификации нелинейных объектов для принятия решений на базе нечеткой логики: Дис... канд. техн. наук: 05.13.06. - Винница, 1998. - 195 с.
27. Кофман А. Введение в теорию нечетких множеств: Пер. с англ. - М.: Радио и связь, 1982.- 432 с.
28. Кузин Л.Т. Основы кибернетики: В 2 т. - М.: Наука, 1979. - Т. 2. - 583с.
29. Куссуль Э.М. Ассоциативные нейрноподобные структуры. - К.: Техніка, 1992.- 210 с.9. Ли Р. Оптимальные оценки, определение характеристик и управление. - М.: Наука, 1966. - 311 с.
30. Ли Р. Оптимальные оценки, определение характеристик и управление. - М.: Наука, 1966. - 311 с.
31. Липцер Р.Ш., Ширяев А.Н. Статистика случайных процессов. - М.: Наука, 1974. - 205 с.
32. Малышев Н.Г., Бернштейн Л.С., Боженьюк А.В. Нечеткие модели для экспертных систем в САПР. - М.: Энергоатомиздат, 1991. - 136 с.
33. Методы и системы принятия решений. Системы, основанные на знаниях / Под ред. А.Н. Борисова. - Рига: РПИ, 1989. - 175 с.
34. Мкртчян С.О. Нейроны и нейронные сети. - М.: Энергия. - 1989. - 178с.
35. Модели принятия решений на основе лингвистической переменной / Под ред. А.Н. Борисова. - Рига: Зинатне, 1982. - 256 с.
36. Мокин Б.И., Грабко В.В., Динь Тхань Вьет. Математические модели и информационно-измерительные системы для технической диагностики трансформаторных вводов. - Винница: ВГТУ, УНІВЕРСУМ-Вінниця, 1997. - 130 с.

37. Мокін Б.І., Мітюшкін Ю.І. Моделювання процесу прогнозування добових графіків споживання електричної потужності на основі баз нечітких знань // Вісник ВПІ. - 2001. - № 5. - С. 58-63.
38. Мокін Б.І., Мітюшкін Ю.І. Застосування баз нечітких знань для прогнозування добових графіків споживання електричної потужності // Тези доповідей VI-ї Міжнародної науково-технічної конференції «Контроль і управління в складних системах» (КУСС-2001). - Вінниця: УНІВЕРСУМ-Вінниця. - 2001. - С. 180.
39. Мокін Б.І., Мітюшкін Ю.І. Прогнозування добових графіків споживання електричної потужності на основі баз нечітких знань // Вісник ВПІ. - 2001. - № 3. - С. 47 - 54.
40. Мокін Б.І., Камінський В.В., Кацев С.Ш. Нетрадиційні операції та принципи узагальнення в теорії нечітких множин (основні ідеї та перспективи застосування в прикладних задачах) // Вісник ВПІ. - 2000. - № 5. - С. 83-88.
41. Мокін Б.І., Камінський В.В., Кацев С.Ш. Властивості слабких операцій в теорії нечітких множин // Вісник ВПІ. - 2001. - № 5. - С. 106-113 .
42. Мокін О.Б. Особливості комп'ютерної реалізації нечіткої експертизи мистецьких творів // Вісник ВПІ. - 2001. - № 3. - С. 86 - 89.
43. Невельсон М.Б., Хасьминский Р.З. Стохастическая аппроксимация и рекуррентное оценивание. - М.: Наука, 1972. - 310 с.
44. Нейрокомпьютеры и интеллектуальные роботы / Под ред. Н.М. Амосова. - К.: Наукова думка, 1991. - 330 с.
45. Орлов А.И. Задачи оптимизации и нечеткие переменные. - М.: Знание, 1980.- 53 с.
46. Поляк Б.Т. Введение в оптимизацию. - М.: Наука, 1983. - 384 с.
47. Поспелов Д.А. Большие системы. Ситуационное управление. - М.: Знание, 1975. - 64 с.
48. Поспелов Д.А. Логико-лингвистические модели в системах управления. - М.: Энергоатомиздат, 1981. - 229 с.

49. Приймак М.В. Побудова прогнозних графіків енергонавантажень на основі періодичного лінійного випадкового процесу // Технічна електродинаміка. - 2000. - № 3. - С. 50-52.
50. Приобретение знаний: Пер. с японск. / Под ред. С. Осуги, Ю. Саэки. - М.: Мир, 1990. - 330 с.
51. Райбман Н.С. Что такое идентификация? - М.: Наука, 1970. - 123 с.
52. Растрингин Л.А. Этот случайный, случайный, случайный мир. - М.: Молодая гвардия, 1974. - 207 с.
53. Растрингин Л.А. Случайный поиск. - М.: Знание, 1979. - 64 с.
54. Розенблат Ф. Принципы нейродинамики (перцептрон и теория механизмов мозга): Пер. с англ. - М.: Мир, 1965. - 480 с.
55. Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткие множества, генетические алгоритмы, нейронные сети. - Винница: УНІВЕРСУМ-Вінниця, 1999. - 320 с.
56. Ротштейн А.П., Кательников Д.И. Идентификация нелинейных объектов нечеткими базами знаний // Кибернетика и системный анализ. - 1998. - № 5. - С. 53-61.
57. Ротштейн А.П., Кательников Д.И. Проектирование и настройка нечетких правил ЕСЛИ-ТО для принятия решений // Тези доповідей IV-ї Міжнародної науково-технічної конференції «Контроль і управління в технічних системах» (КУТС-97). - Том 1. - Вінниця: УНІВЕРСУМ.- 1997. - С. 50-56.
58. Ротштейн А.П., Лойко Е.Е., Кательников Д.И. Прогнозирование количества заболеваний на основе экспертно-лингвистической информации // Кибернетика и системный анализ. - 1999. - № 2. - С. 178-185.
59. Ротштейн А.П. Медицинская диагностика на нечеткой логике. - Винница: Континент-ПРИМ, 1996. - 132 с.
60. Ротштейн А.П., Митюшкин Ю.И. Идентификация нелинейных зависимостей нейронными сетями // Проблемы бионики. - 1998. - № 49. - С. 168-174.

61. Ротштейн А.П., Митюшкин Ю.И. Извлечение нечетких баз знаний из экспериментальных данных с помощью генетических алгоритмов // Кибернетика и системный анализ. - 2001. - № 4. - С. 45-53.
62. Ротштейн А.П., Митюшкин Ю.И. Нейролингвистическая идентификация нелинейных зависимостей // Кибернетика и системный анализ. - 2000. - № 2. - С. 37-44.
63. Ротштейн О.П., Єгоров С.О., Черноволик Г.О. Оцінка якості дипломного проектування на основі нечіткої логіки // Вісник ВПІ. - 1995. - № 4. - С. 52-58.
64. Ротштейн О.П., Кательніков Д.І. Ідентифікація нелінійних об'єктів нечіткими базами знань // Вісник ВПІ. - 1997. - № 4. - С. 98-103.
65. Ротштейн О.П., Ларюшкін Є.П., Кательніков Д.І. Багатофакторний аналіз технологічного процесу біоконверсії на основі лінгвістичної інформації // Вісник ВПІ. - 1997. - № 3. - С. 38-45.
66. Ротштейн О.П., Лойко Є.Є., Кательніков Д.І., Лойко Л.С. Прогнозування кількості захворювань на базі нечіткої логіки // Вісник ВДМУ. - 1998. - № 2. - С. 486-489.
67. Ротштейн О.П., Митюшкін Ю.І. Застосування нейронних мереж для ідентифікації нелінійних залежностей // Вимірювальна та обчислювальна техніка в технологічних процесах. - 1998. - № 3. - С. 9-15.
68. Ротштейн О.П., Митюшкін Ю.І. Нейро-лінгвістична ідентифікація нелінійних залежностей // Вимірювальна та обчислювальна техніка в технологічних процесах. - 1998. - № 4. - С. 5-12.
69. Ротштейн О.П., Черноволик Г.О., Ларюшкін Є.П. Метод побудови функцій належності нечітких множин // Вісник ВПІ. - 1996. - № 3. - С. 72-75.
70. Соколов Е.Н., Шмелев Л.А. Нейробионика: Организация нейроподобных элементов и систем / АН СССР. Научный совет по комплексной проблеме «Кибернетика». - М.: Наука, 1983. - 280 с.

71. Стефани Е.П. Основы расчета настройки регуляторов теплоэнергетических процессов. - М.: ГЭИ, 1960. - 195 с.
72. Тихонов А.Н. О методах регуляризации задач оптимального управления // ДАН СССР. - 1965. - № 4. - С. 34-39.
73. Тихонов А.Н., Арсенин В.Я., Тимонов А.А. Математические задачи компьютерной томографии. - М.: Наука, 1987. - 159 с.
74. Ту Дж., Гонсалес Р. Принципы распознавания образов: Пер. с англ. - М.: Мир, 1978. - 411 с.
75. Хелстром К. Статистическая теория обнаружения сигналов: Пер. с англ. - М.: ИЛ, 1963. - 235 с.
76. Цыганков В.Д. Нейрокомпьютер и его применение. - М.: «Сод Систем», 1993. - 117 с.
77. Цыпкин Я.З. Основы информационной теории идентификации. - М.: Наука, 1984. - 293 с.
78. Штейнберг Ш.Е. Идентификация в системах управления. - М.: Энергоатомиздат, 1987. - 81 с.
79. Эйкхофф П. Основы идентификации систем управления. - М.: Мир, 1975. - 321 с.
80. Экспериментальные исследования режимов энергосистем / Л.М. Горбунова, М.Г. Портной, Р.С. Рабинович и др. / Под ред. С.А. Совалова. - М.: Энергоатомиздат, 1985. - 448 с.
81. Электрические системы и сети / Н.В. Буслова, В.Н. Винославский, Г.И. Денисенко и др. - К.: Вища школа, 1986. - 584 с.
82. Beale R., Jackson T. Neural Computing: An Introduction. - Adam Hilger, IOP Publishing Ltd: Bristol. - 1990. - 350 p.
83. Buyalskaya T.B., Mokin B.I. About one method of the artistic images perception levels measurement // Proc. XIV IMEKO World Congr. «New measurement - challenges and visions». - Tampere (Finland). - 1997. - Vol. V. - P. 18-21.

84. Gen M., Cheng R. Genetic Algorithms and Engineering Design. - John Wiley & Sons, New York, 1997. - 352 p.

85. Miller G.A. The Magic Number Seven Plus or Minus Two: Some Limits on Our Capacity for Processing Information // Psychological Review. - 1956. - ¹ 63. - Д. 81-97.

86. Rotshtein À. Design and tuning of fuzzy if-then rules for medical diagnosis. In N.-H. Teodorescu (ed): Fuzzy and Neural-Fuzzy Systems in Medical and Biomedical Engineering. - CRC Press, 1998. - P. 35 - 97.

87. Rotshtein A.P., Katelnikov D.I., Design and Tuning of Fuzzy If-Then Rules for Automatic Classification // Proc. International Conf. «Annual Meeting of North American Fuzzy Information Processing Society - NAFIPS '98». - Tampa (USA). - 1998. - Д. 50-55.

88. Rotshtein A.P., Katelnikov D.I. Fuzzy rule-based system for psychophysiological testing of the man-operator // Proc. 5th European Congr. on Intelligent Techniques and Soft Computing (EUFIT '97). - Aachen (Germany). - 1997. - P. 984-986.

89. Rotshtein A.P., Katelnikov D.I., Goldenberg L.N. Design and tuning of the fuzzy expert systems for medical diagnostics // Proc. International Workshop «Biomedical Engineering & Medical Informatics» (BEMI '97). - Gliwice (Poland). - 1997. - P. 106-110.

90. Rotshtein A.P., Katelnikov D.I. Tuning of Fuzzy Rules for Nonlinear Objects Identification with Discrete and Continuous Output // Proc. 6th European Congr. on Intelligent Techniques and Soft Computing (EUFIT '98). - Aachen (Germany). - 1998. - Д. 929-934.

91. Rotshtein A.P., Katelnikov D.I., Zlepko S.M. Fuzzy rule-based expert system for differential diagnosis of ischemia heart disease // Proc. International Conf. «Fuzzy Logic and Its Applications - FUZZY '97». - Zichron (Israel). - 1997. - P. 367-372.

92. Rummelhart, D.E. and McClelland, J.L. Parallel Distributed Processing: Explorations in the Microstructure of Cognition (volumes 1 & 2). The MIT Press. - 1986. - 320 p.
93. Rummelhart D.E., Hinton G.E., Williams R.J. Learning Internal Representation by Back-Propagation Errors // Nature. - 1986. - 323. - Д. 533-536.
94. Tang K.S, Man K.F., Kwong S., He Q. Genetic Algorithms and their Applications // IEEE Signal Processing Magazine. - 1996. - Nov. - Д. 22-36.
95. T.B. Buyalskaya, B.I. Mokin Formalization of art products expertise in fuzzy sets // Proc. XVI IMEKO World Congr. Vol. IX. Topic 23 - Expert Systems in Decision Making. - Vienna (Austria). - 2000. - P. 193-197.33.
96. Zimmermann H.-J. Fuzzy Set Theory and Its Applications. - Kluwer, Academic Publisher, Dordrecht, Boston, MA, 2nd ed., 1991. - 315 p.