

Министерство образования Российской Федерации
Владимирский государственный университет

В.Г. ЧЕРНОВ

НЕЧЕТКИЕ КОНТРОЛЛЕРЫ. ОСНОВЫ ТЕОРИИ И ПОСТРОЕНИЯ

Учебное пособие по курсу
«Интеллектуальные системы управления»

Рекомендовано

*УМО по образованию в области радиотехники, биомедицинской техники
и автоматизации в качестве учебного пособия для студентов высших
учебных заведений*

Владимир 2003

УДК 658.562.3

Ч49

Рецензенты:

Доктор физико-математических наук, профессор зав.кафедрой электро-
техники и микропроцессорной техники Московского государственного
института стали и сплавов (технологического университета)

Ф.И. Маняхин

Доктор технических наук, профессор зав кафедрой прикладной математи-
ки и САПР Ковровской государственной академии

А.С. Шалумов

Печатается по решению редакционно-издательского совета
Владимирского государственного университета

Чернов В.Г.

Ч49 Нечеткие контроллеры. Основы теории и построения: Учеб. пособие
по курсу «Интеллектуальные системы управления» / Владим. гос. ун-т.
Владимир, 2003. 148 с.

ISBN 5-89368-384-6

Изложены основы теории нечетких множеств, нечеткой логики, методы выпол-
нения арифметических операций над нечеткими числами. Описаны основные принци-
пы построения нечетких контроллеров и варианты их реализации, а также средства для
проектирования и моделирования нечетких контроллеров. Приведены примеры при-
менения нечетких контроллеров.

Предназначено для студентов, обучающихся по специальности 210100 – управ-
ление и информатика в технических системах, а также может быть использовано сту-
дентами родственных специальностей.

Табл 3. Ил. 103. Библиогр.: 38 назв.

УДК 658.562.3

ISBN 5-89368-384-6

© Владимирский государственный
университет, 2003

Оглавление

Введение	5
Глава 1. ОСНОВЫ ТЕОРИИ НЕЧЕТКИХ МНОЖЕСТВ	7
1.1. Нечеткие множества. Определение	7
1.2. Лингвистическая и нечеткая переменные	10
1.3. Основные методы построения функций принадлежности	13
1.4. Прямые методы одного эксперта.....	17
1.5. Операции над нечеткими множествами	24
1.6. Формализованное представление отношений.....	29
1.7. Нечеткая логика.....	34
1.8. Нечеткие числа. Математика нечетких чисел.....	37
1.9. Нечеткие выводы.....	46
Глава 2. НЕЧЕТКИЕ КОНТРОЛЛЕРЫ	51
2.1. Общие принципы построения.....	51
2.2. Фазификация. Средства реализации	58
2.3. Средства воспроизведения функций принадлежности	61
2.4. Организация обработки правил условного логического вывода	72
2.5. Организация процессоров обработки правил условного вывода	84
2.6. Формирование управляющего воздействия (дефазификация).....	88
2.7. Составление правил нечеткого управления	92
Глава 3. ТЕХНИЧЕСКИЕ И ПРОГРАММНЫЕ СРЕДСТВА ДЛЯ РЕАЛИЗАЦИИ НЕЧЕТКОГО УПРАВЛЕНИЯ	94
3.1. Контроллеры для реализации нечеткого управления	94
3.2. Программные средства нечеткого управления	96

Глава 4. ПРИМЕРЫ ПРАКТИЧЕСКОЙ РЕАЛИЗАЦИИ НЕЧЕТКИХ АЛГОРИТМОВ И СИСТЕМ УПРАВЛЕНИЯ.....	112
4.1. Нечеткая система управления доменной печью	112
4.2. Использование нечетких алгоритмов управления в автомобильных системах.....	128
4.3. Нечеткое управление концентрационным режимом алюминиевых электролизеров	131
4.4. Принципы адаптации в нечетком управлении	139
Библиографический список.....	144

Общепринятые количественные методы непригодны для задач принятия решений, содержащих нечеткость. Поскольку теория решений все больше проникает и в организационные, и в сложные технические системы, нечеткость становится преобладающим фактором при их описании.

Р. Ягер

Введение

Для развития новой отрасли науки необходимо четыре условия: общественная потребность, новая методология (идеи, техника), активность исследователей и интерес потенциальных пользователей.

В отношении теории нечетких множеств все эти условия присутствуют в полной мере. Общественная потребность возникла в связи с тем, что попытки решения ряда задач управления на основе строгих математических методов в классической теории управления оказались неудачными. Объяснить это можно опираясь на сформулированный Л. Заде принцип несовместимости, согласно которому с увеличением размерности и сложности системы существенно усложняется ее моделирование с помощью известных математических соотношений. Другими словами, при использовании формул существенно возрастает число переменных и параметров, при этом их измерение и определение сильно затрудняются. В этой связи создание полностью адекватной модели становится практически невозможным.

В то же время было отмечено, что опытные операторы технологических установок, опытные руководители, опираясь на опыт, интуицию,

собственные представления о целях управления, осуществляют процесс управления так, что результаты оказываются достаточно близкими к тем, которые теоретически могли бы обеспечить самые мощные оптимизационные методы. Важным обстоятельством при этом является то, что указанные лица формулируют модели поведения управляемых ими систем и принципы выработки управляющих воздействий в виде словесных конструкций.

В 1965 году профессор Лотфи Заде предложил лингвистическую модель, в основу которой положены не математические выражения, а слова, отражающие качество. Теоретическое оформление этого подхода получило в виде теории нечетких множеств, которая позволила описывать нечеткие понятия и знания, оперировать ими и делать выводы. За прошедшее время теория нечетких множеств прошла путь от почти антинаучной теории, практически отвергнутой в Европе и США, до ситуации, когда системы нечеткого управления стали применяться, начиная от бытовых приборов до сложных объектов управления типа космических кораблей и ракет-перехватчиков.

Вместе с тем все заметнее стал тот факт, что освоение этих методов осуществлялось бы гораздо проще и быстрее, если бы в распоряжении потенциальных пользователей было бы больше доступной литературы по этим вопросам. К сожалению, большинство работ в этой области написано профессиональными математиками и посвящено в большей степени теоретическим вопросам. Примеры, приводимые в этих работах, либо слишком условны, либо схематичны.

Вышедшая в издательстве "Мир" в 1993 году работа Т. Тэрано и соавторов во многом устраняет эти недостатки. Но, с другой стороны, призванная стимулировать интерес к прикладным аспектам теории нечетких множеств, она все же требует для изучения и освоения изложенного материала достаточно серьезной подготовки, и кроме этого она практически недоступна в связи с ограниченностью тиража.

Предлагаемое пособие представляет собой попытку уменьшить дефицит в учебной литературе по тематике нечеткого управления, так как в основной массе издания соответствующего направления представлены публикациями в периодической печати, а известные монографии имеют либо теоретический характер, либо просто недоступны студентам из-за ограниченных по объему тиражей.

Глава 1

ОСНОВЫ ТЕОРИИ НЕЧЕТКИХ МНОЖЕСТВ

1.1. Нечеткие множества

К классической теории множеств в общем случае относятся аксиоматическая теория множеств и элементарная теория множеств. Первая – одна из фундаментальных теорий математики. В нашем случае нам достаточно некоторых положений элементарной теории, которую в дальнейшем будем называть теорией четких множеств.

Рассматривая управление в самом общем смысле этого слова, с позиции теории четких множеств можно представить процесс управления как задание правил соответствия между элементами множеств входных параметров $X = \{x_i : i = \overline{1, I}\}$ и элементами множества выходных параметров $Y = \{y_j : j = \overline{1, J}\}, I \neq J$. В теории множеств в этом случае считается заданным отображение

$$\Gamma : X \rightarrow Y. \quad (1.1)$$

В различных приложениях отображение (1.1) может быть определено самыми различными способами: в виде таблиц, графиков, алгебраических или дифференциальных уравнений, при этом общим для них всех является то обстоятельство, что в отображении (1.1) участвуют только элементы, обязательно принадлежащие множествам X и Y . В теории четких множеств это определяется заданием характеристической функции. Пусть U – универсальное множество* и X – множество в пространстве U , тогда характеристическая функция множества X определится следующим образом (рис. 1.1).

* Множество U называется универсальным, если для любого $X \subset U$ выполняется условие $X \cap U = X$. Универсальное множество описывается функцией принадлежности $\mu_u(x) = 1, \forall x \in U$.

Например, для множества X чисел $2 \leq x \leq 4$ характеристическая функция имеет вид, представленный рис. 1.2.

$$\chi(x) = \begin{cases} 1, & x \in X \\ 0, & x \notin X \end{cases}$$

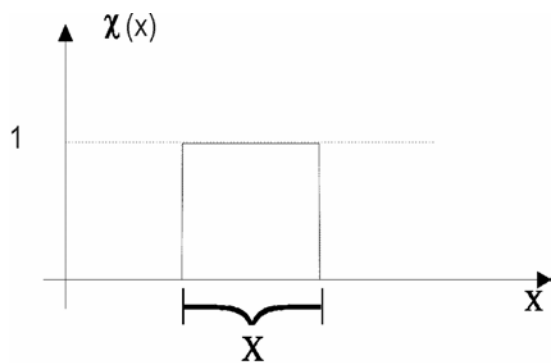


Рис. 1.1

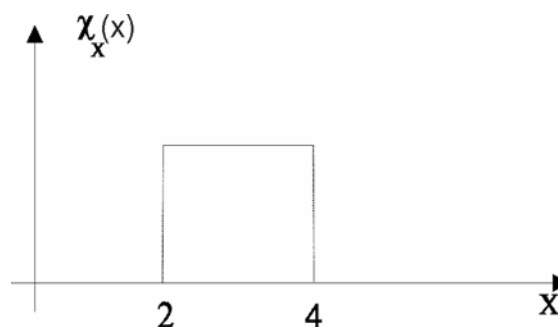


Рис. 1.2.

Естественно, что при таком подходе нет места предположению, что x находится приблизительно в пределах от 2 до 4. Для разрешения этой ситуации Л. Заде [16] расширил двузначную оценку 0 или 1 до неограниченной многозначной оценки выше 0 и ниже 1 на $[0,1]$ и впервые ввел понятие нечеткого множества, заменив характеристическую функцию на функцию принадлежности (рис. 1.3), которая может принимать любые значения в интервале $[0,1]$ для $x \in X$. В соответствии с этим элемент x_i множества U может не принадлежать X ($\mu_x = 0$), может быть элементом

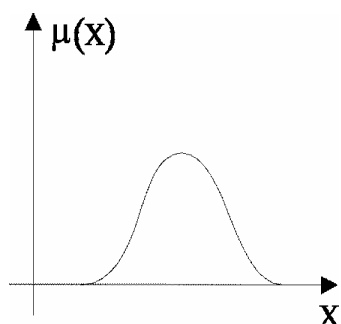


Рис. 1.3

X в небольшой степени (μ_x близко к нулю), может более или менее принадлежать X (μ_x не слишком близко к 0, не слишком близко к единице), может быть в значительной степени элементом X (μ_x близко к единице) или, наконец, может быть элементом X ($\mu_x = 1$). Таким образом, мы можем создать математическую структуру, которая позволяет оперировать с относительно неполно определенными элементами,

принадлежность которых к данному подмножеству лишь в какой-то мере иерархически упорядочена. Множество значений x , на котором определена функция принадлежности, получило название нечеткого множества. Более строгое определение имеет следующую формулировку.

Пусть U – универсальное множество, тогда нечетким множеством \tilde{X} на множестве U называется совокупность пар вида $\tilde{X} = \{\mu_X(x)/x\}$, где $\mu_X(x)$ – функция принадлежности.

Чаще всего определение нечеткого множества объясняют следующим образом: величина $\mu_X(x)$ обозначает субъективную оценку степени принадлежности x множеству X , например, $\mu_X(x)=0,8$ означает, что x на 80 % принадлежит X .

Теперь предположение, что x приблизительно лежит в пределах от 2 до 4, может быть представлено соответствующей функцией принадлежности (рис.1.4).

Носителем нечеткого множества называется множество элементов $x \in U$, такое, что для любого $x \in U$, $\mu_X(x) > 0$:

$$\text{Supp}X = \{x/\forall x \in U, \mu_{\tilde{X}} > 0\}.$$

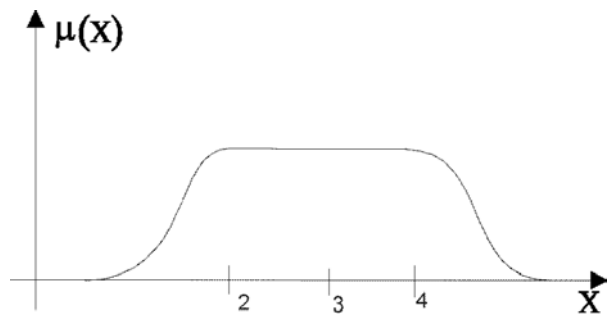


Рис. 1.4

Иными словами, носителем нечеткого множества \tilde{X} является подмножество универсального множества U , для элементов которого функция принадлежности строго больше нуля. Нечеткое множество называют нормальным, если

$$\text{Sup}(\mu_X(x)) = 1.$$

В противном случае нечеткое множество субнормальное.

Очень важным результатом введения функции принадлежности явилось то, что она позволила отразить субъективный взгляд на ту или иную ситуацию.

Например, разные лица могут представить приблизительное равенство $x \approx 2$ совершенно по-разному (рис. 1.5).

Следовательно, могут и должны существовать «моя функция принадлежности», «твоя функция принадлежности», «функция принадлежности эксперта» и т.д.

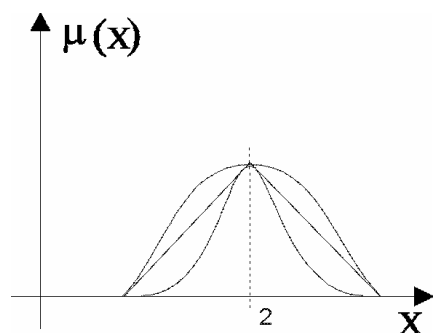


Рис. 1.5

Обратим внимание на связь четкого и нечеткого множеств. Два значения $\{0,1\}$ принадлежат замкнутому интервалу $[0,1]$. Следовательно, четкое множество является частным случаем нечеткого множества, а понятие нечеткого множества – расширенным понятием, охватывающим и понятие четкого множества.

Таким образом, понятие принадлежности получает интересное обобщение, приводящее, как мы это увидим, к очень полезным результатам.

Нечеткое множество строго определяется с помощью функции принадлежности. Понятие функции принадлежности, по существу, является основным в теории нечетких множеств, и большинство приложений этой теории основано на различных операциях над функциями принадлежности.

Кроме функции принадлежности к основным понятиям теории нечетких множеств относится понятие «лингвистическая и нечеткая переменные».

1.2. Лингвистическая и нечеткая переменные

Одной из областей применения теории нечетких множеств являются человеко-машинные системы управления. Осуществление диалога в таких системах невозможно без использования языков, близких к естественному, способных описывать нечеткие категории, приближенные к человеческим понятиям и представлениям. В этой связи целесообразно использовать понятие лингвистической переменной, введенной впервые Л. Заде [16]. Подобные лингвистические переменные позволяют адекватно отразить приблизительное словесное описание предметов и явлений в том случае, когда точное детерминированное описание отсутствует. При этом следует учесть, что многие нечеткие категории, описанные лингвистически, зачастую не менее информативны, чем точное описание.

В качестве примера конкретная фраза: «температура воды равна $+5^{\circ}\text{C}$ » может быть замена приблизительной фразой: «температура воды низкая». В этом смысле слово «низкая» можно рассматривать как лингвистическое значение переменной «температура», имея в виду при этом, что лингвистическое значение играет такую же роль, как и численное значение « $+5^{\circ}\text{C}$ ». То же самое можно сказать о лингвистических значениях «очень низкая», «чуть больше, чем низкая», «почти средняя» и т.д., если их сопоставить с численными значениями $+3$, $+6.5$, $+12$,

Совокупность значений лингвистической переменной составляет терм-множество этой переменной. Это множество может иметь, вообще говоря, бесконечное число элементов, но на практике, естественно, оно конечно. Например, терм-множество лингвистической переменной «температура» можно записать так:

$$(\text{температура}) = \{ \text{очень низкая} \vee \text{почти низкая} \vee \text{низкая} \vee \text{почти средняя} \vee \text{средняя} \vee \dots \vee \text{высокая} \vee \text{очень высокая} \}.$$

Отметим, что в случае лингвистической переменной «температура» числовая переменная «температура», принимающая, например, значения $[+3, +5, +6.5, +12, +17, \dots, +50, +70]$, является так называемой базовой переменной лингвистической переменной «температура». Соответствующее множество значений называется множеством базовых значений, или базовым множеством. При этом такое, например, лингвистическое значение как «высокая» можно интерпретировать как название некоторого нечеткого ограничения на значение базовой переменной. Именно это ограничение будем считать смыслом лингвистического значения «высокая». Функция принадлежности представляет числовую характеристику, количественно определяющую представление субъекта относительно нечеткого ограничения.

Таким образом, нечеткую переменную определяют ее название, область определения, описание ограничений на возможные значения нечеткой переменной, которые задаются функцией принадлежности.

Формальное описание нечеткой переменной будет представлено тройкой

$$\langle L, D, C_L \rangle,$$

где L – наименование нечеткой переменной;

D – область ее определения;

$C_L = \{\mu_L(x)/x\}$ – нечеткое множество на D .

Пример. Пусть температура среды оценивается с помощью понятий «низкая», «средняя», «высокая», при этом минимальная температура оценивается как $+3^{\circ}\text{C}$, а максимальная $+60^{\circ}\text{C}$.

Функции принадлежности, соответствующие этим понятиям, приведены на рис. 1.6.

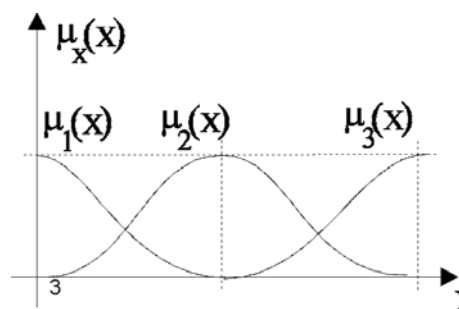


Рис. 1.6

Тогда нечеткая переменная будет задана следующей совокупностью:

$$\langle \text{температура, } [3,60], \mu_1(x)/x, \mu_2(x)/x, \mu_3(x)/x \rangle$$

Чтобы определить лингвистическую переменную, необходимо задать ее имя, множество значений (терм-множество),

представляющих собой наименование нечетких переменных областью определения, каждой из которых является множество D . Кроме этих определений необходимо задать правила, с помощью которых из имеющихся элементов терм-множеств могут получаться новые, а также правила, согласно которым значениям лингвистической переменной ставятся в соответствие нечеткие множества. Формально это представляется так:

$$\langle L, T, D, G, M \rangle,$$

где L – наименование лингвистической переменной;

T – множество значений лингвистической переменной (терм-множество), определенное на D ;

G – грамматика, совокупность правил, позволяющая оперировать элементами терм-множества T , в частности генерировать новые осмысленные термы;

М – процедура, позволяющая установить соответствие между лингвистическим значением и нечетким множеством, т.е. правила вычисления функции принадлежности нового значения, определенного G.

Вернемся к примеру. Пусть определяется новое значение – «малая или средняя температура». Грамматика G определяет правило построения нового значения (рис. 1.7), а процедура М – значения новой функции принадлежности $\mu'(x) = \mu_1(x) \cup \mu_2(x)$ (утолщенная кривая).

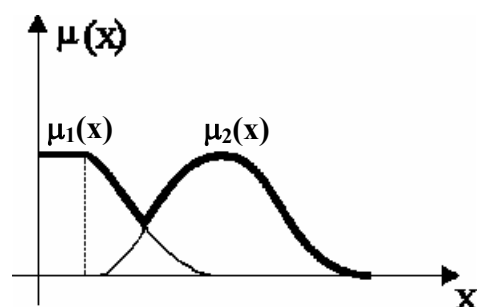


Рис. 1.7

1.3. Основные методы построения функций принадлежности

В основе теории из любой области естествознания лежит очень важное основополагающее для ее построения понятие элементарного объекта. Например, для механики – это материальная точка, для электродинамики – это вектор напряженности поля, для теории автоматического управления – передаточная функция. Для теории нечетких множеств основополагающим понятием является понятие нечеткого множества, которое характеризуется функцией принадлежности. С помощью нечетких множеств можно строго описывать присущие для языка человека расплывчатые понятия, «без формализации которых нет надежды существенно продвинуться вперед в моделировании интеллектуальных процессов» [16]. Основной трудностью, мешающей интенсивному применению теории нечетких множеств при решении практических задач, является то, что функция принадлежности должна быть задана вне самой теории и, следовательно, ее адекватность не может быть проверена непосредственно средствами теории. В каждом в настоящее время известном методе построения функции принадлежности формулируются свои требования и обоснования к выбору именно такого построения.

Фиксирование конкретных значений из интервала [0,1], которыми оценивается степень принадлежности, имеет субъективный характер. С одной стороны, для экспертных методов существенным является характер

измерений (первичный или производный) и тип шкалы [11], в которой получают информацию от эксперта и которая определяет допустимый вид операций, применяемых к экспертной информации [18]. С другой стороны, имеется два типа свойств: те, которые можно непосредственно измерить, и те, которые являются качественными и требуют попарного сравнения объектов, обладающих рассматриваемыми свойствами, чтобы определить их относительное место по отношению к рассматриваемому понятию. Таким образом, построение функции принадлежности выполняется по экспертным оценкам. При этом можно выделить две группы методов – прямые и косвенные.

Прямые методы определяются тем, что эксперт непосредственно задает правила определения значений функции принадлежности. Целесообразность прямых методов обосновывается в [16]: "По своей природе оценка является приближением. Во многих случаях достаточно весьма приближенная характеристика набора данных, поскольку в большинстве основных задач, решаемых человеком, не требуется высокая точность. Человеческий мозг использует допустимость такой неточности, кодируя информацию, достаточную для задачи (или достаточную для решения), элементами нечетких множеств, которые приближенно описывают исходные данные. Поток информации, поступающий в мозг через органы зрения, слуха, осязания и др., суживается таким образом в тонкую струйку информации, необходимой для решения поставленной задачи с минимальной степенью точности".

В косвенных методах значения функции принадлежности выбираются таким образом, чтобы удовлетворить заранее сформулированным условиям. Экспертная информация является только исходной информацией для дальнейшей обработки. Дополнительные условия могут налагаться как на вид получаемой информации, так и на процедуру обработки.

Как правило, прямые методы используются для описания понятий, которые характеризуются измеряемыми параметрами. Однако следует помнить о возможных субъективных искажениях, и поэтому прямые методы должны использоваться только в том случае, когда такие ошибки незначительны или маловероятны.

Косвенные методы более трудоемкие, но они менее чувствительны относительно искажений в ответах. И, наконец, последнее замечание.

Функция принадлежности может отражать мнение одного (уникального) эксперта или же мнение группы экспертов, следовательно, круг методов может быть расширен, так как возможны прямые и косвенные методы для одного эксперта, прямые и косвенные – для группы экспертов. Подробная классификация методов построения функций принадлежности приведена в [16].

Требования к функциям принадлежности

Пусть $T = \{\tau_i\}, i = \overline{1, I}$ – базовое множество лингвистической переменной; a_i – соответствующая ему нечеткая переменная; S_i – носитель нечеткого множества $X_i = \{\mu_{X_i}(x)/x\}$. Договоримся о естественной упорядоченности множества T , при которой терм, имеющий носитель, расположенный левее на числовой оси, имеет меньший номер.

Тогда относительно функции принадлежности можно выдвинуть следующие условия.

1. Функция принадлежности должна быть положительной, т.е.

$$(\forall x \in S_i, i = \overline{1, I}, \mu_{X_i}(x) \geq 0).$$

2. Если это не оговаривается дополнительно, функция принадлежности должна быть нормальной

$$\text{Sup} \mu_{X_i}(x) = 1 \quad (1.2)$$

Если условие нормальности принято, то запрещается использование функций принадлежности, не удовлетворяющих условию (1.2) (рис.1.8). Следует отметить, что это условие относится к исходным функциям принадлежности, так как при выполнении различных операций над функциями принадлежности условие 2 может быть нарушено. Функция принадлежности 3 относится к запрещенным (рис. 1.8).

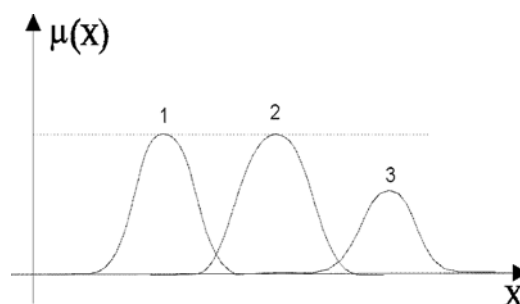


Рис 1.8

3. В базовом множестве термов T запрещается использование пар термов, представленных на рисунке (1.9, а, б). В первом случае отсутствует (см. рис. 1.9, а) естественная разграничиваемость по-

нятий, представленных соседними термами τ_i и τ_{i+1} , во втором (см. рис. 1.9, б) – участку $[c, d]$ из области определения не поставлено в соответствие какое-либо понятие.

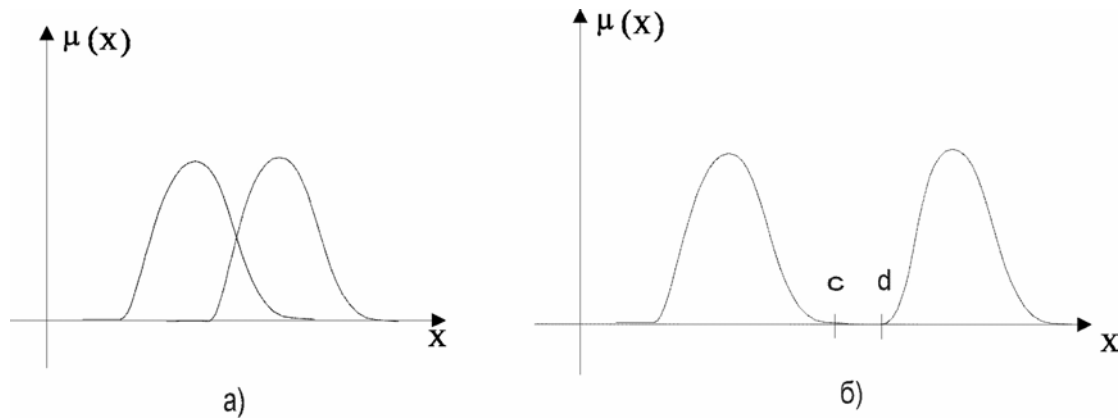


Рис. 1.9

4. Термы с минимальными и максимальными номерами не могут соответствовать колоколообразным функциям принадлежности. Для этих термов функции принадлежности имеют S-образный вид (рис. 1.10).

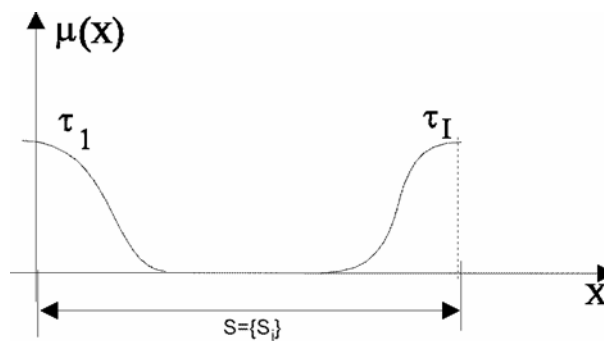


Рис. 1.10

5. Функция принадлежности может задаваться на непрерывном или дискретном носителе.

В практике нечеткого управления наиболее часто используются прямые методы построения функций управления.

Описание более сложных методов можно найти в работе [26].

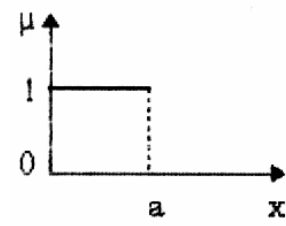
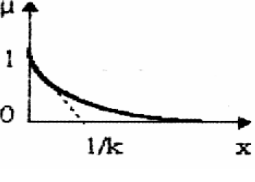
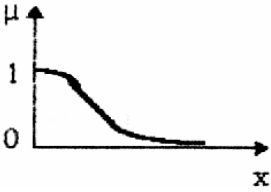
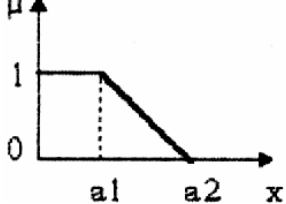
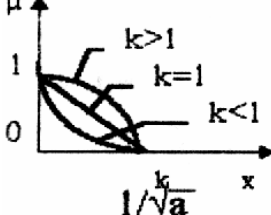
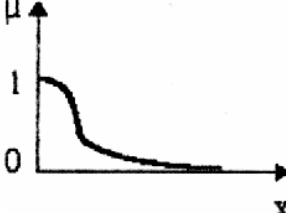
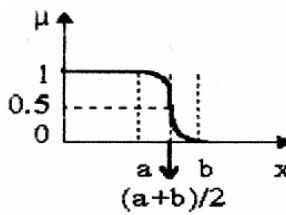
1.4. Прямые методы одного эксперта

Прямые методы для одного (уникального) эксперта состоят в непосредственном назначении степени принадлежности для исследуемых объектов или непосредственном назначении функции (правила), позволяющей вычислить ее значения.

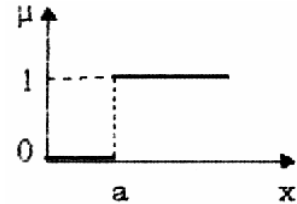
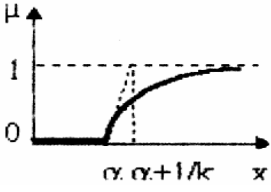
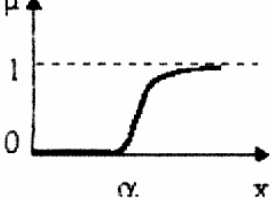
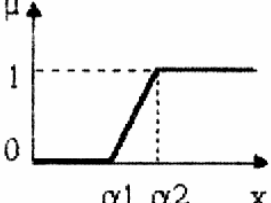
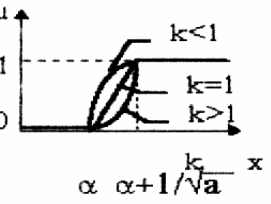
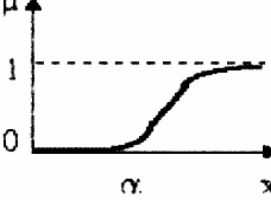
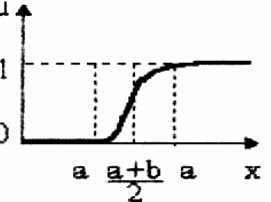
Использование типовых функций принадлежности

К настоящему времени накоплен достаточно широкий набор различных вариантов функций принадлежности для самых разнообразных нечетких утверждений [19, 20] (см. таблицу). Безусловно, выбор функции принадлежности и их параметров определяется в большой степени опытом, интуицией и другими субъективными факторами лица, принимающего решения. Именно здесь возникают новые, связанные с неоднозначностью и другого рода нечеткостью неопределенности, которые носят субъективный характер. Тем не менее, имея некоторый набор типовых функций принадлежности, можно подобрать ту, которая будет в достаточной мере отвечать представлениям лица, её выбирающего. Существенным является то, что для этих функций заранее известны их аналитические представления, что позволяет вычислить их значения в любой точке области определения. В то же время определенные трудности возникают при вычислении параметров аналитического представления функции принадлежности, соответствующих конкретным лингвистическим значениям.

Для вычисления параметров функции принадлежности при известном аналитическом представлении в [7] предложен достаточно сложный метод расчета, отдельные моменты которого представляются слишком формальными и даны без достаточных обоснований. Хотя этот метод и позволяет получить результат, вопрос: почему надо действовать именно так? – на наш взгляд, остается без должного ответа.

График	Формула
	$\mu(x) = \begin{cases} 1, & 0 \leq x \leq a, \\ 0, & x > a \end{cases}$
	$\mu(x) = e^{-kx}; k > 0$
	$\mu(x) = e^{-kx^2}; k > 0$
	$\mu(x) = \begin{cases} 1, & 0 \leq x \leq a1, \\ \frac{a2 - x}{a2 - a1}, & a1 \leq x \leq a2, \\ 0, & a2 < x \end{cases}$
	$\mu(x) = \begin{cases} 1 - ax^k, & 0 \leq x \leq 1/\sqrt[k]{a}, \\ 0, & 1/\sqrt[k]{a} < x \end{cases}$
	$\mu(x) = 1/(1 + kx^2); k > 1$
	$\mu(x) = \begin{cases} 1, & 0 \leq x \leq a, \\ 0.5 - 0.5 \cdot \sin\{\pi[x - (a+b)/2]/(b-a)\}, & a \leq x \leq b, \\ 0, & b \leq x \end{cases}$

Продолжение таблицы

График	Формула
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq a, \\ 1, & x > a \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq \alpha, \\ 1 - e^{-k(x-\alpha)}, & \alpha \leq x, k > 0 \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq \alpha, \\ 1 - e^{-k(x-\alpha)^2}, & \alpha \leq x, k > 0 \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq \alpha_1, \\ (x - \alpha_1) / (\alpha_2 - \alpha_1), & \alpha_1 \leq x \leq \alpha_2, \\ 1, & \alpha_2 < x \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq \alpha_1, \\ a(x - \alpha)^k, & \alpha \leq x \leq \alpha + 1/\sqrt[k]{a}, \\ 1, & \alpha + 1/\sqrt[k]{a} \leq x \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq \alpha, \\ \frac{k(x - \alpha)^2}{1 + k(x - \alpha)}, & \alpha \leq x \leq \infty \end{cases}$
	$\mu(x) = \begin{cases} 0, & 0 \leq x \leq a, \\ 0.5 + 0.5 \cdot \sin\{\pi[x - (a + b)/2] / (b - a)\}, & a \leq x \leq b, \\ 1, & a \leq x \end{cases}$

Продолжение таблицы

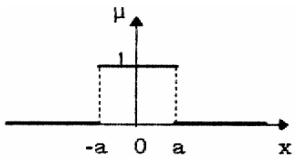
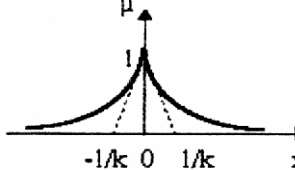
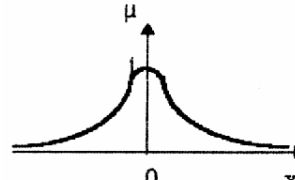
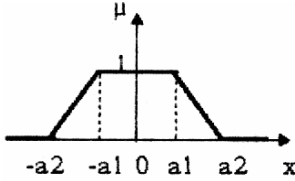
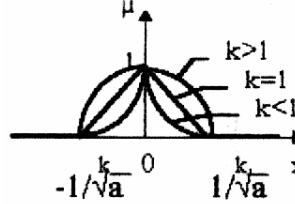
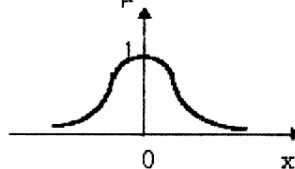
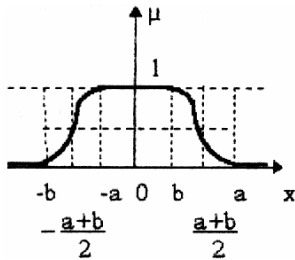
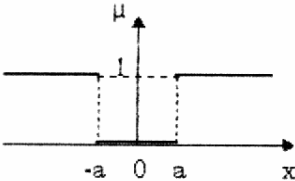
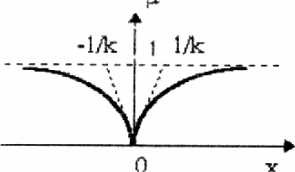
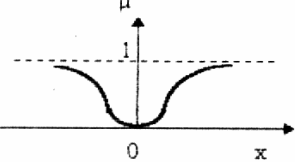
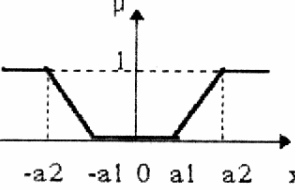
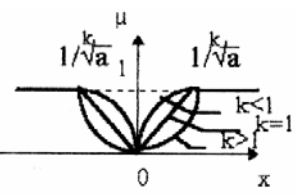
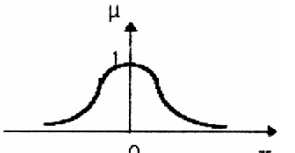
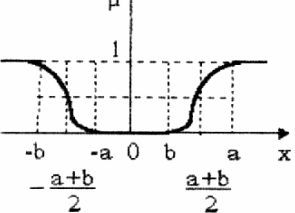
График	Формула
	$\mu(x) = \begin{cases} 0, & -\infty < x < -a, \\ 1, & -a \leq x \leq a, \\ 0, & a < x < \infty \end{cases}$
	$\mu(x) = \begin{cases} e^{kx}, & -\infty < x \leq 0, \\ e^{-kx}, & 0 \leq x < \infty, k > 1 \end{cases}$
	$\mu(x) = e^{-kx^2}$
	$\mu(x) = \begin{cases} 0, & -\infty \leq x \leq -a_2, \\ (a_2 + x)/(a_2 - a_1), & -a_2 \leq x \leq -a_1, \\ 1, & -a_1 \leq x \leq a_1, \\ (a_2 - x)/(a_2 - a_1), & a_1 \leq x \leq a_2, \\ 0, & a_2 \leq x < \infty \end{cases}$
	$\mu(x) = \begin{cases} 0, & -\infty < x \leq -1/\sqrt[k]{a}, \\ 1 - a(-x)^k, & -1/\sqrt[k]{a} \leq x \leq 0, \\ 1 - a(x)^k, & 0 \leq x \leq 1/\sqrt[k]{a}, \\ 0, & 1/\sqrt[k]{a} \leq x < \infty \end{cases}$
	$\mu(x) = 1/(1+kx^2); k > 1$
	$\mu(x) = \begin{cases} 0, & -\infty < x \leq -b, \\ 0.5 + 0.5 \cdot \sin\{\pi[x + (a+b)/2]/(b-a)\}, & -b \leq x \leq -a, \\ 1, & -a \leq x \leq a, \\ 0.5 - 0.5 \cdot \sin\{\pi[x - (a+b)/2]/(b-a)\}, & a \leq x \leq b, \\ 0, & b \leq x < \infty \end{cases}$

График	Формула
	$\mu(x) = \begin{cases} 1, & -\infty \leq x \leq -a, \\ 0, & -a \leq x \leq a, \\ 1, & a \leq x < \infty \end{cases}$
	$\mu(x) = \begin{cases} 1 - e^{kx}, & -\infty < x \leq 0, \\ 1 - e^{-kx}, & 0 \leq x < \infty, k > 1 \end{cases}$
	$\mu(x) = 1 - e^{-kx^2}, k > 1$
	$\mu(x) = \begin{cases} 1, & -\infty < x \leq -a2, \\ -(a1 + x)/(a2 - a1), & -a2 \leq x \leq -a1, \\ 0, & -a1 < x \leq a1, \\ (x - a1)/(a2 - a1), & a1 \leq x \leq a2, \\ 1, & a2 \leq x < \infty \end{cases}$
	$\mu(x) = \begin{cases} 1, & -\infty < x \leq -1/\sqrt[k]{a}, \\ a(-x)^k, & -1/\sqrt[k]{a} \leq x \leq 0, \\ a(x)^k, & 0 \leq x \leq 1/\sqrt[k]{a}, \\ 1, & 1/\sqrt[k]{a} \leq x < \infty \end{cases}$
	$\mu(x) = kx^2 / (1 + kx^2); k > 1$
	$\mu(x) = \begin{cases} 1, & -\infty < x \leq -b, \\ 0.5 - 0.5 \cdot \sin \{ \pi [x + (a+b)/2] / (b-a) \}, & -b \leq x \leq -a, \\ 0, & -a \leq x \leq a, \\ 0.5 + 0.5 \cdot \sin \{ \pi [x - (a+b)/2] / (b-a) \}, & a \leq x \leq b, \\ 1, & b \leq x < \infty \end{cases}$

Можно предложить более простую методику, которая вытекает из рассмотрения функций принадлежности, приведенных в таблице.

Все функции могут быть разбиты на два класса:

- с конечным носителем, т.е. когда точно можно указать элемент x , при котором $\mu_X(x) = 0$;

- с бесконечным носителем, для которых

$$\lim_{|x| \rightarrow \infty} \mu_X(x) = 0.$$

В первом случае эксперт или лицо, принимающее решение, однозначно определяет носитель нечеткого множества или базовое множество, соответствующее определенному лингвистическому значению.

Во втором случае эксперт должен ответить на вопрос типа: Какое минимальное значение должна иметь функция принадлежности, чтобы не считать элемент « x » принадлежащим данному множеству? Ответ на этот вопрос в дальнейшем определит параметры функции принадлежности.

Затем необходимо указать координаты плато функции принадлежности (c, d) (рис. 1.11).

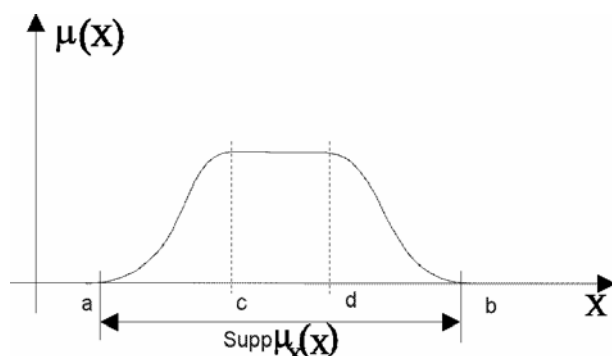


Рис. 1.11

Для многих аналитических представлений (см. таблицу этих параметров достаточно для расчета значений функций принадлежности в любой точке. Однако для экспоненциальных, параболических и гиперболических форм представления функций принадлежности этого оказывается недостаточно. Тогда экс-

перту можно предложить следующий вопрос: Для какого значения x его принадлежность нечеткому множеству оценивается равной 0,5? Пусть

эксперт выбрал функцию принадлежности вида $\mu_X(x) = e^{-kx^2}$.

Тогда при $x = x_{0.5}$, $\mu_X(x) = 0.5$, соответственно $k = -\frac{\ln 0.5}{(x_{0.5})^2}$.

Если при ответе на предыдущий вопрос эксперт указал для функции принадлежности допустимое значение погрешности ε_1 , с которой опреде-

ляется указанное значение, то носитель нечеткого множества будет определяться из соотношения

$$x_H = \pm \sqrt{\frac{\ln \varepsilon_1}{\ln 0.5}} (x_{0.5})^2.$$

И наконец, может быть использована методика, предложенная в [5], которая основана на анализе двух соседних лингвистических значений. Пусть рассматриваются два лингвистических значения $j-1, j$ и соответствующие термы (рис. 1.12). Определяются носители $(a_{j-1}, b_{j-1}), (a_j, b_j)$, плато $(c_{j-1}, d_{j-1}), (c_j, d_j)$. Для построения нисходящей (для терма $j-1$) и восходящей (для терма j) ветвей эксперта просят указать точку, относительно которой он испытывает наибольшие трудности при соотнесении ее с термом $j-1$ или j . Нетрудно видеть, что в этой точке $\mu_X(x) = 0.5$ для обоих термов. Дальнейшие расчеты трудностей не представляют.

Достаточно часто встречается задача построения функции принадлежности, соответствующей некоторому произвольному значению. Для этой цели удобно использовать представление функций принадлежности в

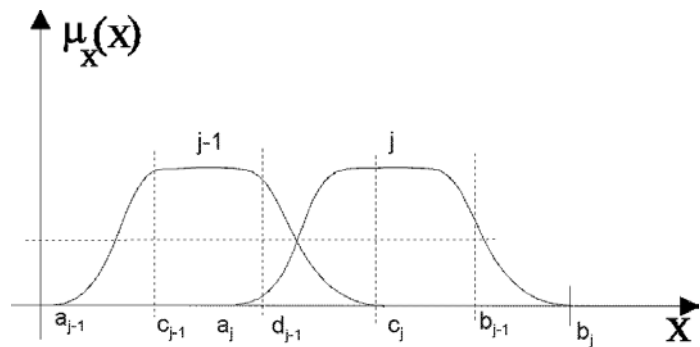


Рис. 1.12

виде стандартных S-образных функций. Не обсуждая варианты аналитического представления этих функций, покажем чисто качественно возможность их использования. S-образную функцию можно определить тремя точками (l, m, r) (рис. 1.13).

При $x=l$ $S(x)=0$, при $x = m$ $S(x) = 0.5$, при $x = r$ $S(x) = 1$.

Таким образом, $S(x) = S(x, l, m, n)$.

Функция вида $1 - S(x)$ представлена на рис. 1.14. Нетрудно видеть, что, комбинируя эти две функции, можно построить функции принадлежности, удовлетворяющие ранее сформулированным условиям.

Пусть определены функции $S(x_1)$ и $S(x_2)$, соответствующие исходному базовому множеству термов $T = \{\tau_i\}, i = \overline{1, I}$.

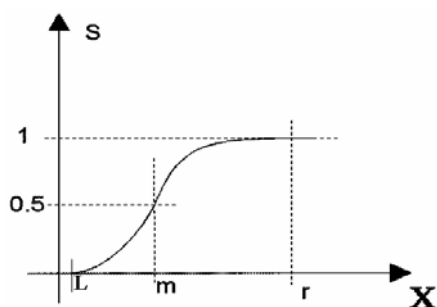


Рис. 1.13

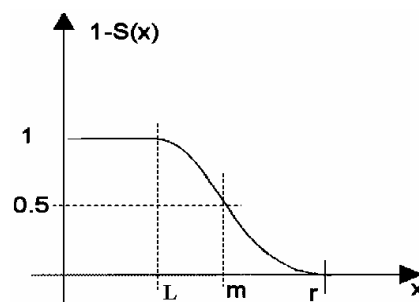


Рис. 1.14

Если произвольное значение $x' \in [x_1, x_2]$, то можно предположить, что $m' \in [m_1, m_2]$.

Положив, что справедливо равенство

$$\frac{(m' - m_1)}{(m' - m_2)} = \frac{(x' - x_1)}{(x' - x_2)}$$

и, обозначая через $\lambda = \frac{(x' - x_1)}{(x' - x_2)}$, $\lambda \in [0, 1]$, получим

$$m' = \lambda m_1 + (1 - \lambda) m_2.$$

Последнее соотношение позволяет при непрерывном носителе рассчитывать значения функции принадлежности для произвольного значения лингвистической переменной, не включенного в исходное базовое множество.

Рассмотренные выше примеры, естественно, не исчерпывают всего множества прямых методов построения функций принадлежности для одного эксперта. Целый ряд других методов этого класса рассмотрен в [26], там же рассматриваются и прямые методы для нескольких экспертов.

1.5. Операции над нечеткими множествами

Операции над нечеткими множествами такие, например, как объединение и пересечение можно определить различными способами. Выбор конкретного из них зависит от специфики решаемой задачи, т.е. от конкретного смысла, вкладываемого в эти операции.

Объединением нечетких множеств A , B и U называют нечеткое множество вида

$$A \cup B = \int_U (\mu_A(x) \vee \mu_B(x)) / x, \quad (1.3)$$

где $(\mu_A(x) \vee \mu_B(x)) = \max\{\mu_A(x), \mu_B(x)\}, x \in U$.

Знак интеграла здесь и для всех последующих операций обозначает их выполнение над всеми элементами множеств A и B .

Графическая интерпретация операции объединения приведена на рис. 1.15.

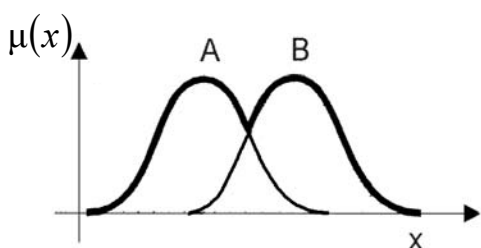


Рис. 1.15

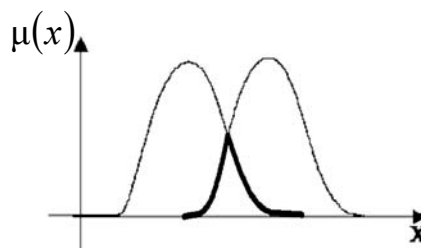


Рис. 1.16

Пересечением нечетких множеств A и B из U называют нечеткое множество вида (рис.1.16)

$$A \cap B = \int_U (\mu_A(x) \wedge \mu_B(x)) / x, \quad (1.4)$$

где $(\mu_A(x) \wedge \mu_B(x)) = \min\{\mu_A(x), \mu_B(x)\}, x \in U$.

Дополнением или отрицанием нечеткого множества A называют нечеткое множество вида (рис. 1.17)

$$\bar{A} = \int_U (1 - \mu_A(x)) / x, x \in U. \quad (1.5)$$

Концентрирование нечеткого множества A из U , определяют в виде (рис. 1.18)

$$\text{CON}(A) = \int_U (\mu_A(x))^2 / x, x \in U. \quad (1.6)$$

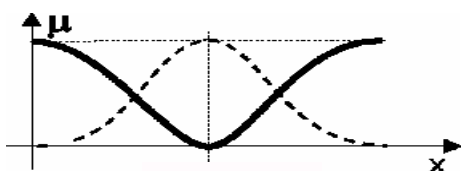


Рис. 1.17

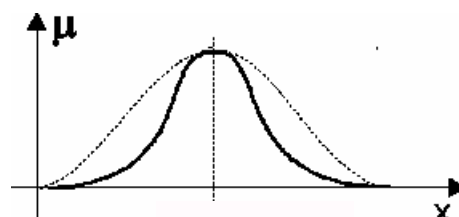


Рис. 1.18

Растяжение нечеткого множества A из U , определяют в виде (рис. 1.19)

$$DIL(A) = \int_U (\mu_A(x))^{0.5} / x, \quad x \in U. \quad (1.7)$$

Множеством уровня α нечеткого множества (α -срезом) A называют нечеткое множество, составленное из элементов $x \in U$, степень принадлежности которых нечеткому множеству A не меньше α (рис. 1.20)

$$\forall \alpha \in [0,1], \quad A_\alpha = \{x/x \in U, \mu_A(x) \geq \alpha\}.$$

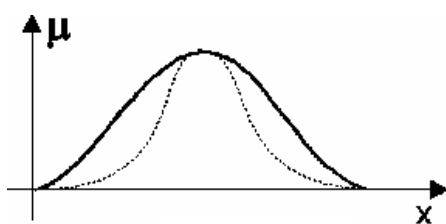


Рис. 1.19

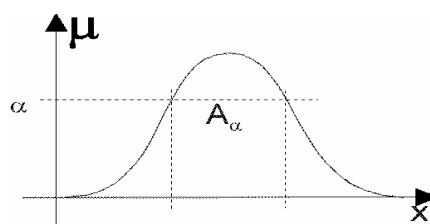


Рис. 1.20

Строгое множество уровня определяется как

$$\forall \alpha \in [0,1], A_\alpha^- = \{x/x \in U, \mu_A(x) > \alpha\}.$$

Тогда функцию принадлежности можно определить для произвольного нечеткого множества A с помощью его α -сечения в виде

$$\mu_A(x) = \sup_{\alpha \in [0,1]} \min(\alpha, \mu_{A_\alpha}(x)),$$

где $\mu_{A_\alpha}(x) = \begin{cases} 1, & \text{если } x \in A_\alpha, \\ 0, & \text{если } x \notin A_\alpha. \end{cases}$

Нечеткое множество уровня α нечеткого множества A определяется следующим образом:

$$\tilde{A}_\alpha = (x, \mu_x(x), x \in A_\alpha).$$

Преимуществом такого определения является то, что в прикладных задачах целесообразно использовать не сами нечеткие множества, а их множества уровня, что позволяет экономить время вычислений и память ЭВМ.

Пусть A и B – произвольные нечеткие множества из U . Говорят, что A включает в себя B ($B \subseteq A$), если

$$\forall x \in U, \mu_B(x) \leq \mu_A(x).$$

Когда последнее неравенство строгое, тогда говорят, что включение строгое. Очевидно, что $A=B$, если $(A \subseteq B), (B \subseteq A)$.

Если функции принадлежности двух нечетких множеств A и B из U равны, то A и B – равные нечеткие множества, т.е. если

$$\mu_B(x) = \mu_A(x), \quad \forall x \in U, \text{ то } A=B.$$

Нетрудно убедиться, что введенные операции над нечеткими множествами являются более общими, чем аналогичные операции над обычными множествами.

Известно определение функции принадлежности объединения нечетких множеств через их алгебраическую сумму

$$\mu_{A \cup B}(x) = \begin{cases} 1 & \text{при } \mu_A(x) + \mu_B(x) \geq 1, \\ \mu_A(x) + \mu_B(x) & \text{в противном случае,} \end{cases}$$

а функции принадлежности пересечения нечетких множеств через алгебраическое произведение

$$\mu_{A \cap B}(x) = \mu_A(x) \mu_B(x), \quad \forall x \in U.$$

Нечеткие подмножества некоторого универсального множества относительно операций объединения, пересечения и дополнения, определенных соотношениями (1.3) – (1.6) удовлетворяют следующим свойствам:

1. Идемпотентность: $A \cup A = A \cap A \neq 0$ при $A \neq 0$. Отметим, что нечеткое подмножество универсального множества U называется пустым при условии $\mu_0(x) = 0$ для $\forall x \in U$.

2. Коммутативность: $A \cup B = B \cup A, A \cap B = B \cap A$.

Данные свойства с очевидностью вытекают из приведенных выше определений операций над нечеткими множествами.

3. Ассоциативность: $(A \cup B) \cup C = A \cup (B \cup C), (A \cap B) \cap C = A \cap (B \cap C)$.

4. Дистрибутивность:

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C), \quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$$

5. Поглощаемость: $A \cup (B \cap A) = A$. Это свойство можно записать в другой форме, а именно

$$\max(\mu_A, \min(\mu_A, \mu_B)) = \mu_A.$$

6. Единственность обратного: $(\overline{\overline{A}}) = A$.

7. Правила Моргана:

$$\begin{aligned} \overline{(A \cup B)} &= \overline{A} \cap \overline{B}, \\ \overline{(A \cap B)} &= \overline{A} \cup \overline{B}. \end{aligned}$$

Перейдем к рассмотрению следующих операций над нечеткими множествами.

Пусть A_1 и A_2 – нечеткие подмножества универсальных множеств U_1 и U_2 соответственно. Тогда декартово произведение нечетких подмножеств A_1 и A_2 обозначается $A_1 \times A_2$ и определяется как нечеткое подмножество множества U . Последнее определяется декартовым произведением

$$U = U_1 \times U_2.$$

При этом функция степеней принадлежности декартова произведения $A_1 \times A_2$ определяется выражением

$$\mu_{A_1 \times A_2}(x_1, x_2) = \min\{\mu_{A_1}(x_1), \mu_{A_2}(x_2)\}, \quad x_1 \in U_1, \quad x_2 \in U_2.$$

Например, пусть имеем универсальные множества $U_1=U_2=(3,5,7)$ и нечеткие подмножества $A_1=(0.5/3, 1/5, 0.6/7)$ и $A_2=(1/3, 0.6/5)$.

В этом случае декартово произведение нечетких подмножеств A_1 и A_2 будет равно $A_1 A_2 = \{0.5/(3,3), 1/(5,3), 0.6/(7,3), 0.5/(3,5), 0.6/(5,5), 0.6/(7,5)\}$.

Декартово произведение нечетких множеств тесно связано с понятием нечеткого отношения, которое будет рассмотрено ниже.

В ряде приложений теории нечетких множеств требуется проводить их сравнение. Формализацией сравнения нечетких подмножеств A и B универсального множества U с конечным числом элементов может быть вычисление расстояния Хэмминга, которое определяется выражением

$$d(A, B) = \sum_{i=1}^n |\mu_A(x_i) - \mu_B(x_i)|.$$

Знак « \sum » обозначает арифметическое сложение.

Отметим, что

$$|\mu_A(x_i) - \mu_B(x_i)| = \max\{\mu_A(x_i), \mu_B(x_i)\} - \min\{\mu_A(x_i), \mu_B(x_i)\},$$

$$0 \leq d(A, B) \leq n.$$

Например, $A=(0/0, 0.1/1, 0.2/2, 0.5/3, 0.8/4, 1/5)$,

$B=(1/0, 1/1, 0.8/2, 0.6/3, 0.4/4, 0.2/5)$.

Найдем расстояние Хэмминга

$$\begin{aligned} d(A, B) &= |1-0| + |1-0.1| + |0.8-0.2| + |0.6-0.5| + |0.8-0.4| + |10.2|= \\ &= 1+0.9+0.6+0.1+0.4+0.8=3.8. \end{aligned}$$

Кроме линейных форм расстояние между нечеткими подмножествами рассматривают как квадратичные формы, а именно

$$e(A, B) = \sqrt{\sum_{i=1}^n [\mu_A(x_i) - \mu_B(x_i)]^2},$$

для которых $0 \leq e(A, B) \leq \sqrt{n}$.

1.6. Формализованное представление отношений

Параметры различных систем могут быть связаны между собой различного рода отношениями. Выделение отношений осуществляется по заранее выбранному признаку. Если нас интересует влияние параметра системы на ее производительность или качество выпускаемой продукции, то данная связь может быть описана различного вида отношениями, “влияет”, “ не влияет”, “сильно влияет”, “ слабо” и т.д. Наиболее распространенной формой задания отношений является словесное описание. Обще-принятая формализация отношений осуществляется в соответствии со следующим определением.

Отношением R на множестве U называется подмножество R множества, определяемого декартовым произведением.

Существует несколько форм задания отношений. Задание отношения R на множестве U может быть выполнено перечислением всех пар $(u_i, u_j) \in U, (i, j = \overline{1, n})$, для которых выполняется отношение R . Кроме того, отношения задаются в виде матриц и графов. Простейшими отношениями являются такие, для которых можно четко указать, выполняются они или нет для параметров X_1 и X_2 .

В тех случаях, когда связи между параметрами системы выражены нечетко, целесообразно формализовать отношения в соответствии со следующим определением.

Пусть U_1 и U_2 – универсальные множества. Если U является декартовым произведением $U=U_1 \times U_2$, то нечеткое отношение R определяется как нечеткое подмножество универсального множества U :

$$\mu_R : u_1 \times u_2 \rightarrow [0,1].$$

Значение $\mu_R(u_i, u_j)$ для конкретной пары $(u_i, u_j) \in u_1 \times u_2$ характеризует субъективную степень выполнения отношения

$$u_i \tilde{R} u_j.$$

Если множество U конечно и невелико, нечеткое отношение R удобно задавать в матричном виде. В этом случае матрица $M(\tilde{R})$ отношения \tilde{R} представляет собой квадратную матрицу, строки и столбцы которой помечены элементами $u \in U$, и на пересечении строки u_i и столбца u_j записано значение $r_{ij} = \mu_R(u_i, u_j)$.

Пример. Пусть $U = \{1, 3, 5, 7, 9\}$. Определим на множестве U нечеткое отношение \tilde{R} “намного больше”. Матрица такого отношения может иметь следующий вид:

	1	3	5	7	9
1	0	0	0	0	0
3	0,2	0	0	0	0
5	0,1	0	0	0	0
7	0,8	0,4	0	0	0
9	1	0,8	0,5	0	0

Наглядностью обладает задание нечеткого отношения в виде нечеткого графа $\tilde{G} = (\tilde{U}, \tilde{V})$, где $\tilde{U} = \{u_1, u_2, u_3, u_4, u_5\}$ – множество вершин; $\tilde{V} = \left\{ \frac{\mu_R(u_i, u_j)}{(u_i, u_j)} \mid \mu_R(u_i, u_j) > 0 \right\} (u_i, u_j) \in U$ (рис. 1.21) – множество нечетких дуг. Очевидно, что, как и в случае нечетких множеств, обычное четкое отношение можно рассматривать как частный случай нечеткого отношения, функция принадлежности которого принимает два значения: 0 и 1.

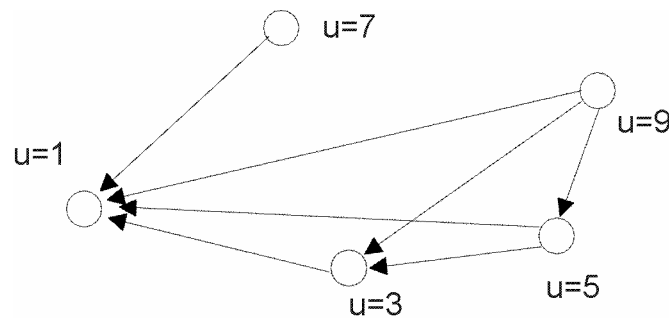


Рис. 1.21

Дадим некоторые определения, характеризующие нечеткие отношения. Носителем нечеткого отношения R на множестве U называют подмножество декартова произведения $U \times U$ вида

$$\text{Supp} \tilde{R} = \left\{ \frac{(u_i, u_j)}{(u_i, u_j)} \in U \times U, \mu_R(u_i, u_j) > 0 \right\}.$$

Носитель нечеткого отношения следует понимать как отношение на множестве U , связывающее все пары (u_i, u_j) , для которых степень выполнения данного нечеткого отношения не равна нулю.

Для нашего примера

$$(u_3, u_1), (u_5, u_1), (u_7, u_1), (u_9, u_1), (u_5, u_3), (u_7, u_3), (u_9, u_3), (u_9, u_5).$$

По аналогии с нечеткими множествами определяется и множество α -уровня нечеткого отношения, т.е.

$$R_\alpha = \left\{ \frac{(u_i, u_j)}{(u_i, u_j)} \in U \times U, \mu_R(u_i, u_j) \geq \alpha \right\}.$$

Перейдем к рассмотрению операций над нечеткими отношениями, некоторые из которых являются аналогами операций над нечеткими множествами, а некоторые присущи только нечетким отношениям.

Пересечением нечетких отношений P и Q на $U \times U$ называют нечеткое отношение $P \cap Q$, определяемое функцией принадлежности

$$\mu_{P \cap Q}(u_i, u_j) = \mu_P(u_i, u_j) \wedge \mu_Q(u_i, u_j) = \min_{u_i, u_j \in U} \{ \mu_P(u_i, u_j), \mu_Q(u_i, u_j) \}.$$

Объединением нечетких отношений P и Q на $X \times X$ называют нечеткое бинарное отношение $P \cup Q$, определяемое функцией принадлежности

$$\mu_{P \cup Q}(x_i, x_j) = \mu_P(x_i, x_j) \vee \mu_Q(x_i, x_j) = \max \{ \mu_P(x_i, x_j), \mu_Q(x_i, x_j) \}.$$

Дополнением нечеткого отношения $R \subseteq X \times X$ называют отношение \bar{R} с функцией принадлежности

$$\begin{aligned} \mu_{\bar{R}}(x_i, x_j) &= 1 - \mu_R(x_i, x_j), \\ \forall x_i, x_j &\in X \times X. \end{aligned}$$

Обратным отношением к отношению R называют отношение R^{-1} с функцией принадлежности

$$\begin{aligned} \mu_{R^{-1}}(x_i, x_j) &= \mu_R(x_j, x_i), \\ \forall x_i, x_j &\in X \times X. \end{aligned}$$

Очевидно, что матрица $M^{-1}(R)$ является транспонированной $M(R)$.

Говорят, что отношение R_1 включено в отношение R , если множество пар, для которых выполняется отношение R_1 , находится в множестве пар, для которых выполняется отношение R . Так, например, отношение между параметрами Z_1 и Z_2 , характеризуемое термином “много меньше”,

включено в отношение, характеризуемое понятием “меньше”. Отметим, что обратное утверждение может не выполняться. Тот факт, что отношение R_1 включено в отношение R , обозначают следующим образом: $R_1 \subseteq R$.

На этапе формализации качественной информации важную роль играют отношения эквивалентности, порядка и доминирования. С помощью отношения эквивалентности могут выделяться классы свойств исследуемых объектов или систем, которые являются в некотором смысле равноценными. Это отношение оказывается полезным для выявления в множестве первичных терминов подмножества терминов-синонимов.

Отношение эквивалентности обладает свойствами рефлексивности, симметричности и транзитивности. Рефлексивность отношения R означает выполнение условия $U \subseteq R$, U -диагональное отношение. Данное отношение используется для формализации понятий типа “похоже на”, “подобен” и т.п. На главной диагонали матрицы рефлексивного отношения стоят единицы. В понятиях типа “похоже на”, “подобен” выделяют свойства симметричности.

Отношение R симметрично, если выполняется условие $R \subseteq R^{-1}$, т.е. если выполняется связь $z_1 R z_2$, то должно выполняться $z_2 R z_1$.

Транзитивность подразумевает то, что если параметры z_1 и z_2 связаны отношением R , а также этим же отношением связаны z_2 и z_3 , то параметры z_1 и z_3 связаны этим же отношением. Формально данное свойство записывается следующим образом: если $z_1 R z_2$ и $z_2 R z_3$, то $z_1 R z_3$.

Наряду с рассмотренными свойствами отношений выполняются свойства антирефлексивности и асимметричности. Первое выполняется, если пересечение $R \cap U = 0$, а второе – при $R \cap R^{-1} = 0$. По своему смысловому содержанию свойства антирефлексивности и асимметричности противоположны рефлексивности и симметричности соответственно.

Отношение доминирования характеризуется свойствами антирефлексивности и асимметричности. Это отношение используется для формализации связей между z_1 и z_2 в случаях, когда z_1 превосходит или предпочтительнее в некотором смысле z_2 .

Частным случаем отношения доминирования является отношение порядка, для которого дополнительно выполняется свойство транзитивности. Примером отношения порядка является понятие “больше”.

В теории нечетких множеств имеется ряд операций над нечеткими отношениями, которые не имеют аналогов для нечетких множеств, рассмотренных в п. 1.5.

Максиминное произведение нечетких отношений R_1 и R_2 , которые определены на множестве U , обозначается $R_1 \circ R_2$ и задается функцией принадлежности $\mu_{R_1 \circ R_2}(u_1, u_2)$, вычисляемой следующим образом:

$$\mu_{R_1 \circ R_2}(u_1, u_2) = \text{Sup}\{\min(\mu_{R_1}(u_1, z), \mu_{R_2}(z, u_2))\},$$

где μ_{R_1}, μ_{R_2} – функции принадлежности нечетких отношений R_1 и R_2 соответственно.

Например, пусть заданы два универсальных множества $U_1=U_2=(1,2,3)$. На множестве $U_1 \times U_2$ определены нечеткие отношения

$$R_1 = \begin{bmatrix} 1 & 0.8 & 0.4 \\ 0.8 & 1 & 0.8 \\ 0.4 & 0.8 & 1 \end{bmatrix}; R_2 = \begin{bmatrix} 0 & 0.3 & 0.7 \\ 0.3 & 0 & 0.3 \\ 0.7 & 0.3 & 0 \end{bmatrix}.$$

Так как нечеткие отношения заданы в матричном виде, то максиминное произведение в данном случае представляет собой операцию, аналогичную умножению матриц, но вместо арифметических операций умножения и сложения используют операции нахождения минимального (\wedge – пересечение) и максимального (\vee – объединение) элементов соответственно. Поэтому максиминное произведение имеет вид

$$R_1 \circ R_2 = \begin{bmatrix} 0.4 & 0.3 & 0.7 \\ 0.7 & 0.3 & 0.7 \\ 0.7 & 0.3 & 0.4 \end{bmatrix}.$$

Минимаксное произведение нечетких отношений R_1 и R_2 на множестве U определяется функцией принадлежности, вычисляемой по соотношению

$$\mu_{R_1 \circ R_2}(u_1, u_2) = \text{inf}\{\max(\mu_{R_1}(u_1, z), \mu_{R_2}(z, u_2))\}.$$

Если нечеткие отношения R_1 и R_2 заданы в виде матриц, то минимаксное произведение представляет собой операцию, аналогичную операции умножения матриц, но вместо арифметических операций умножения и сложения используются операции \wedge и \vee .

Например, для предыдущего примера минимаксное произведение в матричном виде равно

$$R_1 \circ R_2 = \begin{bmatrix} 0.7 & 0.4 & 1 \\ 0.8 & 0.8 & 0.8 \\ 0.4 & 0.4 & 0.7 \end{bmatrix}.$$

Максимумпликативное произведение нечетких отношений R_1 и R_2 , заданных на множестве U , определяется функцией принадлежности

$$\mu_{R_1 \circ R_2}(u_1, u_2) = \text{Sup}\{\mu_{R_1}(u_1, z) \times \mu_{R_2}(z, u_2)\}.$$

Для исходных данных примера максимумпликативное произведение нечетких отношений равно

$$R_1 \circ R_2 = \begin{bmatrix} 0.28 & 0.3 & 0.7 \\ 0.56 & 0.24 & 0.56 \\ 0.7 & 0.3 & 0.28 \end{bmatrix}.$$

Выбор той или иной композиции при решении практических задач определяется требованиями, которым должно удовлетворять решение задачи. Поэтому в каждом конкретном случае данный вопрос требует особого рассмотрения.

1.7. Нечеткая логика

В сочетании слов «нечеткая» и «логика» есть что-то необычное. Логика в обычном смысле слова есть представление механизмов мышления, то, что никогда не может быть нечетким, но всегда строгим и формальным. Однако математики, исследовавшие эти механизмы мышления, заметили, что в действительности существует не одна логика (например булева), а столько логик, сколько мы пожелаем, потому что все определяется выбором соответствующей системы аксиом. Конечно, как только аксиомы выбраны, все утверждения, построенные на их основе, должны быть строго, без противоречий увязаны друг с другом согласно правилам, установленным в этой системе аксиом.

Булева логика – это логика, связанная с булевой теорией множеств, аналогично, нечеткая логика связана с теорией нечетких множеств.

В бинарной булевой алгебре определены следующие основные логические операции:

$$a \wedge b \text{ – операция “И” } a = [0,1], b = [0,1]; \quad (1.7)$$

$$a \vee b \text{ – операция “ИЛИ”}; \quad (1.8)$$

$$\bar{a} \text{ – операция “НЕ”}; \quad (1.9)$$

$$a \oplus b = \bar{a}b \vee a\bar{b} \text{ – операция “исключающее ИЛИ”}. \quad (1.10)$$

Соотношения (1.8 – 1.10) будут справедливы и для нечетких переменных.

Пусть $x \in U$, A и B – нечеткие подмножества U . Обозначим

$$\tilde{a} = \mu_a(x), \tilde{b} = \mu_b(x).$$

Тогда

$$\tilde{a} \wedge \tilde{b} = \min(\tilde{a}, \tilde{b}) \text{– нечеткий аналог операции “И”};$$

$$\tilde{a} \vee \tilde{b} = \max(\tilde{a}, \tilde{b}) \text{– нечеткий аналог операции “ИЛИ”};$$

$$\bar{\tilde{a}} = 1 - \tilde{a} \text{– нечеткий аналог операции “НЕ”};$$

$$\tilde{a} \oplus \tilde{b} = (\bar{\tilde{a}} \wedge \tilde{b}) \vee (\tilde{a} \wedge \bar{\tilde{b}}) \text{– нечеткий аналог операции “исключающее ИЛИ”}.$$

Можно доказать, что эти операции удовлетворяют следующим свойствам:

$$\left. \begin{aligned} \tilde{a} \wedge \tilde{b} &= \tilde{b} \wedge \tilde{a} \\ \tilde{a} \vee \tilde{b} &= \tilde{b} \vee \tilde{a} \end{aligned} \right\} \text{– коммутативность};$$

$$\left. \begin{aligned} (\tilde{a} \wedge \tilde{b}) \wedge \tilde{c} &= \tilde{a} \wedge (\tilde{b} \wedge \tilde{c}) \\ (\tilde{a} \vee \tilde{b}) \vee \tilde{c} &= \tilde{a} \vee (\tilde{b} \vee \tilde{c}) \end{aligned} \right\} \text{– ассоциативность};$$

$$\left. \begin{aligned} \tilde{a} \wedge \tilde{a} &= \tilde{a} \\ \tilde{a} \vee \tilde{a} &= \tilde{a} \end{aligned} \right\} \text{– идемпотентность};$$

$$\tilde{a} \wedge (\tilde{b} \vee \tilde{c}) = (\tilde{a} \wedge \tilde{b}) \vee (\tilde{a} \wedge \tilde{c}) \quad (1.11)$$

$$\tilde{a} \vee (\tilde{b} \wedge \tilde{c}) = (\tilde{a} \vee \tilde{b}) \wedge (\tilde{a} \vee \tilde{c}) \quad (1.12)$$

$$\tilde{a} \wedge 0 = 0,$$

$$\tilde{a} \vee 0 = \tilde{a},$$

$$\tilde{a} \wedge 1 = \tilde{a},$$

$$\tilde{a} \vee 1 = 1,$$

$$\overline{\overline{\tilde{a}}} = \tilde{a}$$

$$\overline{\tilde{a} \wedge \tilde{b}} = \bar{\tilde{a}} \vee \bar{\tilde{b}} \quad (1.13)$$

$$\overline{\tilde{a} \vee \tilde{b}} = \bar{\tilde{a}} \wedge \bar{\tilde{b}} \quad (1.14)$$

– обобщение теорем де Моргана на случай нечетких множеств.

Доказательства всех этих формул тривиальны, за исключением, может быть, формул (1.11 – 1.14).

На рис. 1.22 – 1.23 приведены графические построения, доказывающие соотношения (1.11), (1.12). Рис. 1.22, а, 1.23, а представляют результаты выполнения преобразований в левой части соотношений (1.13), (1.14), рис. 1.22, б, 1.23, б – в правой части этих же соотношений. Соответствующие результаты представлены заштрихованными областями.

За исключением двух свойств

$$a \wedge \bar{a} = 0,$$

$$a \vee \bar{a} = 1,$$

для которых, кроме случая $\tilde{a} = 0$ или $\tilde{a} = 1$, соответствующие соотношения для нечетких множеств не выполняются:

$$\tilde{a} \wedge \bar{\tilde{a}} \neq 0,$$

$$\tilde{a} \wedge \bar{\tilde{a}} \neq 1.$$

Перечисленные выше соотношения составляют все свойства биполярной булевой алгебры. Читателю, желающему более подробно изучить этот вопрос, рекомендуем обратиться к работе [13].

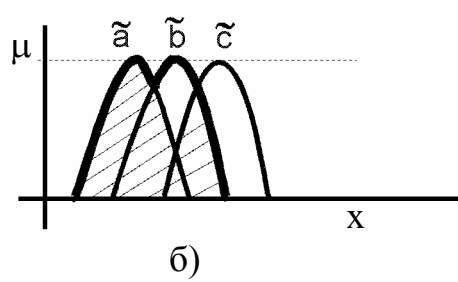
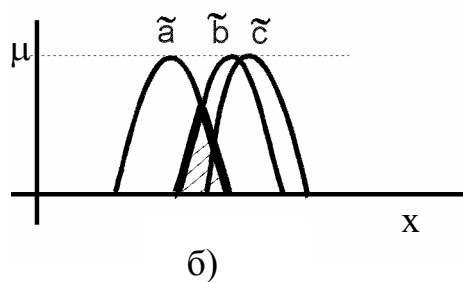
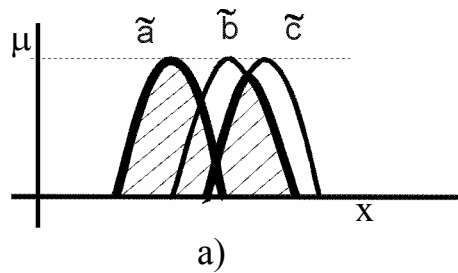
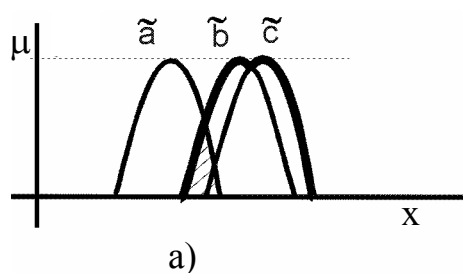


Рис. 1.22

Рис. 1.23

1.8. Нечеткие числа. Математика нечетких чисел

Нечеткие числа широко используются в повседневной жизни. Когда мы говорим: “приблизительно три”, “приблизительно двадцать пять” и т.п., то тем самым предполагаем использование нечеткого числа.

Формально определение нечеткого числа строится аналогично определению нечеткой переменной. Действительно, если считать, что универсальное множество – это множество действительных чисел, то нечеткое подмножество $X \in U$, определяемое функцией принадлежности $\mu_X(x) \in [0,1]$, можно рассматривать как нечеткое число.

Нечеткое число \tilde{X} на действительной прямой выпукло, если для каких-либо чисел $x_i, x_j, x_k, x_i \leq x_j \leq x_k$

$$\mu_{\tilde{X}}(x_j) \geq \min(\mu_{\tilde{X}}(x_i), \mu_{\tilde{X}}(x_k)).$$

Нечеткое число \tilde{X} на действительной прямой называется нормальным, если $\max \mu_{\tilde{X}}(x) = 1$. На рис. 1.24 приведены \tilde{X}_1 – нечеткое выпуклое число, \tilde{X}_2 – нечеткое нормальное число, \tilde{X}_3 – нечеткое нормальное выпуклое число.

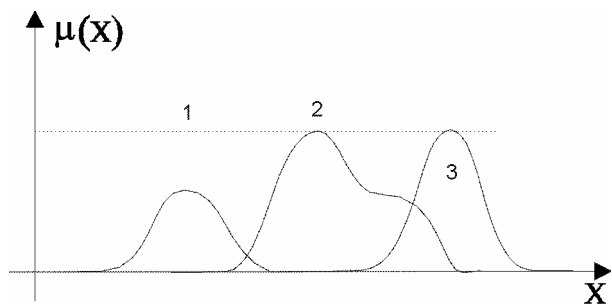


Рис. 1.24

Вводя понятие нечеткого числа (нечетких чисел), мы, естественно, должны определить некоторые, хотя бы простейшие, математические операции над этими числами. Сразу же следует указать, что это далеко не тривиальная задача, которая к тому же в некоторых аспектах еще не решена.

Математические операции, вводимые для нечетких чисел, впрочем, как и для четких, должны решать, по крайней мере, две задачи:

- первая – это находить результат применения некоторой комбинации математических операций к заранее известной совокупности чисел, т.е. находить значение соотношения

$$y = \varphi(x_1, x_2, \dots, x_3),$$

где φ – некоторый функционал;

- вторая – это находить значение некоторой переменной по известным значениям других, т. е. решения различного рода уравнений, например для простейшего случая

$$A = X + B \text{ или } A = X / B,$$

где A, B известные значения; X – неизвестная переменная. При этом очевидно, что должно выполняться условие, если $X = A - B$ или $X = AB$:

$$A = (A - B) + B = A \quad A = (AB) / B = A. \quad (1.15)$$

Для четких чисел выполнение указанных условий и соответственно решение уравнений каких-либо проблем не составляют. Кроме того, две перечисленные задачи в этом случае как бы объединяются. В какой-то мере это можно объяснить тем, что для четких чисел определены противоположное число \bar{A} , такое что $A + \bar{A} = 0$ или $\bar{A} = 0 - A$, и обратное число A^* :

$$A \cdot A^* = 1, \quad A^* = \frac{1}{A},$$

т. е. эти числа могут быть вычислены с помощью стандартных операций вычитания и деления.

Для нечетких чисел все обстоит значительно сложнее. Сразу же укажем, что согласно исследованиям, проведенным в [28], нечеткое число не имеет противоположного и обратного чисел, а арифметические операции умножения и сложения для нечетких чисел коммутативны

$$\tilde{A} \times \tilde{B} = \tilde{B} \times \tilde{A}, \quad \tilde{A} + \tilde{B} = \tilde{B} + \tilde{A},$$

ассоциативны

$$\begin{aligned} \tilde{A} \times (\tilde{B} \times \tilde{C}) &= (\tilde{A} \times \tilde{B}) \times \tilde{C}, \\ \tilde{A} + (\tilde{B} + \tilde{C}) &= (\tilde{A} + \tilde{B}) + \tilde{C} \end{aligned}$$

и в общем виде недистрибутивны: $\tilde{A} \times (\tilde{B} + \tilde{C}) \neq \tilde{A} \times \tilde{B} + \tilde{A} \times \tilde{C}$.

В связи с этим первая и вторая задачи для нечетких чисел приобретают определенную несвязанность. Проблема заключается в том, что методы, относящиеся к первой задаче, не дают точного решения второй, в то же время методы, используемые при решении второй задачи, могут применяться в первой с весьма существенными ограничениями.

Для решения первой задачи нам нужно определить алгоритмы выполнения арифметических операций над нечеткими числами, для решения второй – алгоритмы решения нечетких уравнений.

Алгоритмы выполнения арифметических операций над нечеткими числами

Конечной целью любого алгоритма для выполнения арифметических операций над нечеткими числами является вычисление функции принадлежности для любой точки носителя результата выполнения заданной арифметической операции над исходными нечеткими числами. Если \tilde{A} и \tilde{B} – нечеткие числа с областью определения в виде интервала на действительной оси и их носители соответственно определены как $S_A = (a_2, a_1)$, $S_B = (b_2, b_1)$, $a_2 > a_1$, $b_2 > b_1$ и $g(a, b)$ – некоторая функция, определяющая одну из четырех арифметических операций “+”, “-”, “·”, “x”, то в общем виде справедливо соотношение

$$\begin{aligned} \mu_D(x) &= \text{Sup} \min\{\mu_A(a), \mu_B(b)\}, \\ g(a, b) &= x, \\ a &\in S_A, b \in S_B, \end{aligned} \quad (1.16)$$

которое отражает принцип обобщения, впервые сформулированный Л. Заде. Если функция g является функцией большего числа аргументов, то принцип обобщения записывается аналогично.

Соотношение (1.16) в той или иной мере является основой различных арифметических алгоритмов для нечетких чисел. Значительное их количество может быть объяснено различной точностью вычисления функции принадлежности и вычислительными затратами. Ниже будут рассмотрены несколько наиболее простых, но в то же время имеющих наибольшее практическое применение алгоритмов. Желаящим познакомиться с более сложными можно рекомендовать работы [7, 28] и др.

Арифметические операции над нечеткими числами при L-R-аппроксимации

Одним из вариантов представления нечетких чисел является так называемая L-R-аппроксимация, при которой нечеткое число \tilde{X} определяется его левой x_L и правой x_R границами, а также центральным значением x^* . Наиболее просто L-R-аппроксимация выполняется для нечетких чисел, функции принадлежности которых за пределами носителя тождественно равны нулю, например треугольные, трапецеидальные.

Для треугольной функции принадлежности (рис. 1.25) нечеткое число $x \approx x^*$ представляется следующим соотношением:

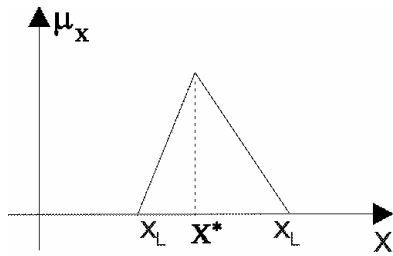


Рис. 1.25

$$\tilde{x}^* = \int_{x_L}^{x^*} (x - x_L)/x + \int_{x^*}^{x_R} (x_R - x)/x,$$

где знак \int означает объединение по всем $x \in [x_L, x^*]$ и $x \in [x^*, x_R]$ соответственно.

Пусть имеются два нечетких числа

$$\tilde{x}_1^* = \int_{x_{L1}}^{x_1^*} (x - x_{L1})/x + \int_{x_1^*}^{x_{R1}} (x_{R1} - x)/x$$

$$\text{и } \tilde{x}_2^* = \int_{x_{L2}}^{x_2^*} (x - x_{L2})/x + \int_{x_2^*}^{x_{R2}} (x_{R2} - x)/x.$$

Суммой чисел \tilde{x}_1^* и \tilde{x}_2^* назовем число \tilde{x}_3^* такое, что

$$\tilde{x}_3^* = \int_{x_{L3}}^{x_3^*} \mu_{x_3}(x)/x + \int_{x_3^*}^{x_{R3}} \mu_{x_3}(x)/x,$$

$$\begin{aligned} \text{где } x_{L3} &= x_{L1} + x_{L2}; \\ x_{R3} &= x_{R1} + x_{R2}; \\ x_3^* &= x_1^* + x_2^*. \end{aligned}$$

Для треугольных функций принадлежности слагаемых функция принадлежности суммы также будет треугольной и представляться уравнением $\mu(x) = a_0 + kx$. Для определения значений a_0 и k необходимо для $x_{L3} \leq x \leq x_3^*$ решить систему уравнений

$$\begin{cases} a_0 + kx = 1 & \text{при } x = x_3^* = x_1^* + x_2^*, \\ a_0 + kx = 0 & \text{при } x = x_{L3} = x_{L1} + x_{L2}, \end{cases}$$

$$\text{откуда } a_0 = -\frac{x_{L3}}{x_3^* - x_{L3}}, k = \frac{1}{x_3^* - x_{L3}}$$

$$\text{и соответственно } \mu_{x_3}(x) = \frac{x - x_{L3}}{x_3^* - x_{L3}}.$$

$$\begin{aligned} \text{Для } x_3^* \leq x \leq x_{R3} \quad a_0 + kx &= 1 & \text{при } x = x_3^*; \\ a_0 + kx &= 0 & \text{при } x = x_{R3}. \end{aligned}$$

Тогда $a_0 = -\frac{x_{R3}}{x_3^* - x_{R3}}$, $k = \frac{1}{x_3^* - x_{R3}}$ и $\mu_{x_3}(x) = \frac{x - x_{R3}}{x_3^* - x_{R3}}$.

Таким образом,

$$\tilde{x}_3^* = \int_{x_{L3}}^{x_3^*} \frac{x - x_{L3}}{x_3^* - x_{L3}} \Big/ x + \int_{x_3^*}^{x_{R3}} \frac{x_{R3} - x}{x_{R3} - x_3^*} \Big/ x.$$

Аналогично для операции вычитания

$$\tilde{x}_3^* = \int_{x_{L3}}^{x_3^*} \frac{x - x_{L3}}{x_3^* - x_{L3}} \Big/ x + \int_{x_3^*}^{x_{R3}} \frac{x_{R3} - x}{x_{R3} - x_3^*} \Big/ x,$$

$$x_{L3} = x_{L1} - x_{R2}, \quad x_{R3} = x_{R1} - x_{L2}, \quad x_3^* = x_1^* - x_2^*.$$

Для операций умножения и деления, предполагая сохранность треугольной функции принадлежности, в работе [7] получены следующие соотношения:

$$\tilde{x}_3^* = \int_{x_{L3}}^{x_3^*} \frac{\sqrt{x} - \sqrt{x_{L3}}}{\sqrt{x_3^*} - \sqrt{x_{L3}}} \Big/ x + \int_{x_3^*}^{x_{R3}} \frac{\sqrt{x_{R3}} - \sqrt{x}}{\sqrt{x_{R3}} - \sqrt{x_3^*}} \Big/ x,$$

$$x_{L3} = x_{L1}x_{L2}, \quad x_{R3} = x_{R1}x_{R2}, \quad x_3^* = x_1^*x_2^*,$$

результатирующая функция принадлежности $\mu_{x_3}(x) = a_0 + k_1\sqrt{x}$:

$$\tilde{x}_3^* = \int_{x_{L3}}^{x_3^*} \frac{(x - x_{L3})x_3^*}{(x_3^* - x_{L3})x} \Big/ x + \int_{x_3^*}^{x_{R3}} \frac{(x_{R3} - x)x_3^*}{(x_{R3} - x_3^*)x} \Big/ x,$$

где $x_{L3} = \frac{x_{L1}}{x_{R2}}$, $x_{R3} = \frac{x_{R1}}{x_{L2}}$, $x_3^* = \frac{x_1^*}{x_2^*}$,

результатирующая функция принадлежности $\mu_{x_3}(x) = a_0 + \frac{k_1}{x}$.

В отношении двух последних операций целесообразно сделать следующее замечание. Операции умножения и деления являются операциями нелинейными, поэтому предположение о сохранении вида функции принадлежности представляются достаточно искусственными.

Отметим еще одну особенность нормальных выпуклых непрерывных нечетких чисел: найти нечеткое число и его правую и левую границы можно не проводя лингвистического анализа, поскольку точно известно, при каком x функция принадлежности равна 1, а при каких x она равна нулю.

Операции над нечеткими числами с использованием уровней множеств

В основу данного подхода к выполнению операций над нечеткими числами положена возможность дискретизации нечеткого числа по конечному числу α -уровней, когда каждому уровню α_i ставится в соответствие множество

$$X_{\alpha_i} = \{x_{\alpha_i 1}, x_{\alpha_i 2}, \dots, x_{\alpha_i n}\},$$

$$\mu(x_{\alpha_i j}) \geq \alpha_i, j = \overline{1, n},$$

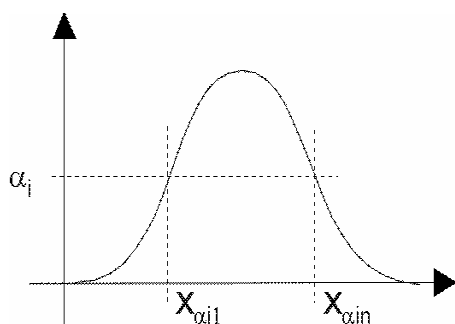


Рис. 1.26

а также разложения на выпуклые, возможно, ненормализованные нечеткие подмножества (рис. 1.26). Для операций с использованием уровней множеств накладывается дополнительное ограничение: эти нечеткие подмножества должны иметь функции принадлежности либо строго убывающие, либо строго возрастающие, либо постоянные (рис. 1.27).

Эти ограничения объясняются тем, что если рассматривать только монотонные (только возрастающие или только убывающие) операции*, то для этих операций на участках одинаковой монотонности функций принадлежности результат может быть получен без дополнительного лингвистического анализа.

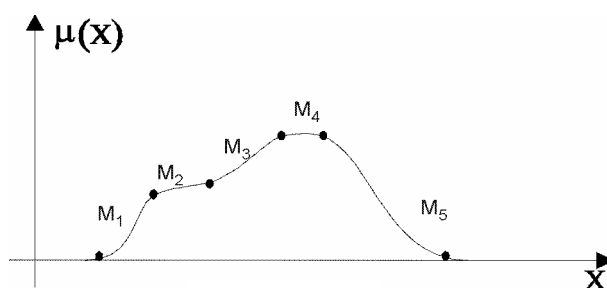


Рис. 1.27

* Бинарная операция \diamond называется возрастающей, если для любых $x_1 > y_1$ и $x_2 > y_2 \Rightarrow x_1 \diamond x_2 > y_1 \diamond y_2$, и убывающей, если для $x_1 > y_1$ и $x_2 > y_2 \Rightarrow x_1 \diamond x_2 < y_1 \diamond y_2$.

Пусть нечеткие числа \tilde{X} и \tilde{Y} представлены в виде α -уровневых подмножеств, для которых функции принадлежности имеют одинаковый характер монотонности:

$$\begin{aligned} \tilde{X} &= \{X_{\alpha_i}\}, \tilde{Y} = \{Y_{\alpha_i}\}. \\ \tilde{X} &= \left\{ \left(x_{\alpha_i 1}, \dots, x_{\alpha_i n_i} \right) \right\} \quad i = 1, \bar{I}, \\ \tilde{Y} &= \left\{ \left(y_{\alpha_j 1}, \dots, y_{\alpha_j m_j} \right) \right\} \quad j = 1, J, \\ \mu(x_{\alpha_i k}) &\geq \alpha_i \quad i = 1, \bar{I}, k = 1, n_i, \\ \mu(y_{\alpha_j q}) &\geq \alpha_j \quad i = 1, \bar{I}, q = 1, m_j. \end{aligned}$$

Операции выполняются над абсциссами точек, находящихся на одинаковых α -уровнях и имеющих одинаковые участки монотонности функций принадлежности.

Рассмотрим для простоты один α -уровень α_i и три значения аргумента (рис. 1.28)

$$x_{\alpha_{ij}} \text{ и } y_{\alpha_{ij}} \quad j = 1, 2, 3.$$

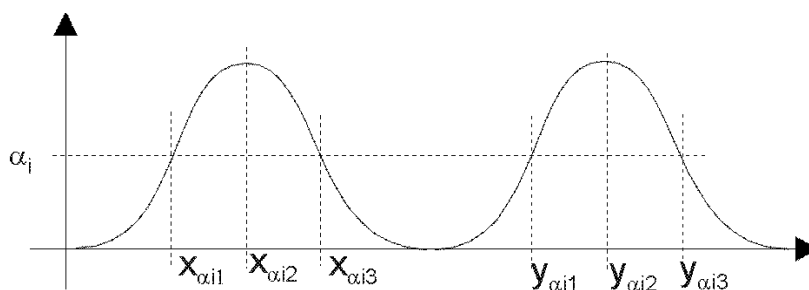


Рис. 1.28

$$\begin{aligned} \tilde{X}_{\alpha_i} &= \left\{ \mu_{\alpha_i 1}(x_{\alpha_i 1}) / x_{\alpha_i 1}, \mu_{\alpha_i 2}(x_{\alpha_i 2}) / x_{\alpha_i 2}, \mu_{\alpha_i 3}(x_{\alpha_i 3}) / x_{\alpha_i 3} \right\}, \\ \tilde{Y}_{\alpha_i} &= \left\{ \mu_{\alpha_i 1}(y_{\alpha_i 1}) / y_{\alpha_i 1}, \mu_{\alpha_i 2}(y_{\alpha_i 2}) / y_{\alpha_i 2}, \mu_{\alpha_i 3}(y_{\alpha_i 3}) / y_{\alpha_i 3} \right\}. \end{aligned}$$

Тогда для строго возрастающих или строго убывающих операций, какими являются сложение и умножение, справедливы соотношения

$$\tilde{Z}_{\alpha_i} = \tilde{X}_{\alpha_i} \times \tilde{Y}_{\alpha_i} = \left\{ \mu_{\alpha_i 1} / (x_{\alpha_i 1} \times y_{\alpha_i 1}), \mu_{\alpha_i 2} / (x_{\alpha_i 2} \times y_{\alpha_i 2}), \mu_{\alpha_i 3} / (x_{\alpha_i 3} \times y_{\alpha_i 3}) \right\}, \quad (1.17)$$

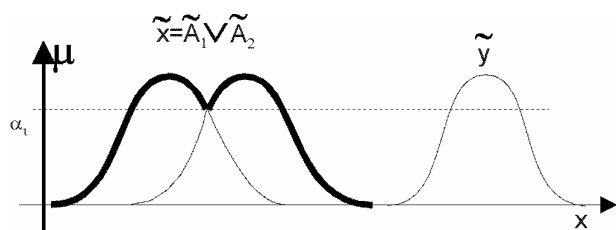
$$\tilde{Z}_{\alpha_i} = \tilde{X}_{\alpha_i} \times \tilde{Y}_{\alpha_i} = \left\{ \mu_{\alpha_i 1} / (x_{\alpha_i 1} + y_{\alpha_i 1}), \mu_{\alpha_i 2} / (x_{\alpha_i 2} + y_{\alpha_i 2}), \mu_{\alpha_i 3} / (x_{\alpha_i 3} + y_{\alpha_i 3}) \right\}. \quad (1.18)$$

Операции вычитания и деления не являются строго возрастающими или строго убывающими, поэтому их вначале надо представить в виде

$$\begin{aligned} \tilde{X} - \tilde{Y} &= \tilde{X} + (-\tilde{Y}), \\ \tilde{X} / \tilde{Y} &= \tilde{X} \times \left(\frac{1}{\tilde{Y}} \right), \end{aligned}$$

а затем может использоваться или соотношение (1.17), или (1.18).

Основным ограничением при использовании данного метода реализации нечетких множеств является



требование участков одинаковой монотонности функций принадлежности для участков операций.

На рис. 1.29 представлен случай, когда это условие не выполняется.

Рис. 1.29

Алгоритм использования принципа обобщения при выполнении арифметических операций над нечеткими числами

Арифметические операции над нечеткими числами можно рассматривать как функциональное преобразование возможных значений вектора

$$\vec{X} = \{x_i\} \quad i = \overline{1, n}$$

$$y = \varphi(x_1, \dots, x_n). \quad (1.19)$$

Например, $y = x_1 + x_2$, $y = x_1 x_2$, $y = \frac{x_1}{x_2}$, $y = x_1 - x_2$.

Считая известным значение функции принадлежности $\mu_x(x_i)$, необходимо найти функцию принадлежности $\mu_y(y)$, которая будет определять результат функционального преобразования (1.20). В литературе, изла-

гающей основные операции над нечеткими множествами (числами), ограничиваются преимущественно лишь заданием основного правила для определения функции принадлежности:

$$\mu_Y(y) = \max_{\bar{X}: \varphi(\bar{X})=y} \min\{\mu_x(x_i)\}. \quad (1.20)$$

При непосредственном использовании такого правила для решения практических задач возникают трудности в разработке методов и алгоритмов определения результирующей функции принадлежности, вызванные математической сложностью описания и определения множества X , удовлетворяющего условию (1.20).

В работе [24] предложены два варианта алгоритмов определения функции принадлежности (1.20): алгоритм перебора и рекуррентный алгоритм. В нашем случае мы остановимся только на алгоритме перебора, который, несмотря на громоздкость, позволяет получить в большинстве случаев результаты, представляющие практический интерес.

Рекуррентный алгоритм предполагает решение нелинейного уравнения, которое только в отдельных случаях имеет аналитическое решение [18].

При использовании метода перебора на множестве возможных значений нечетких переменных формируется множество различных их комбинаций

$$\{x_i\}_l \quad l = \overline{1, L},$$

где l – номер комбинации.

Для каждой комбинации рассчитывают значение функции $y_l = \varphi(\{x_i\}_l)$ и функции принадлежности $\mu_{Y_l}(Y_l) = \min_i \{\mu_x(x_i)\}$.

В общем случае для различных комбинаций l_i, l_j, l_k возможно, что $y_{l_i} = y_{l_j} = y_{l_k}$, при этом $\mu_Y(Y_{l_i}) \neq \mu_Y(Y_{l_j}) \neq \mu_Y(Y_{l_k})$.

Тогда результирующее значение $\mu_Y(Y_l) = \max_{i,j,k} \{\mu_Y(y_l)\}$.

В заключение полученные значения y_l упорядочивают и затем разбивают на некоторое количество k групп с постоянным числом членов n_k .

Для каждой группы определяют $\mu_k(Y_e) = \max \{ \mu_Y(y_e) \}$. Значения $\mu_k(y_e)$ рассматриваются как n_k значений функции принадлежности результата. Очевидно, что точность вычисления $\mu_Y(y)$ определяется дискретностью разбиения множеств исходных значений x_i . После получения дискретного множества значений $\mu_Y(y)$ имеет смысл воспользоваться интерполяционными процедурами.

Выбор количества дискретных отсчетов исходных значений x_i можно осуществить, задаваясь точностью представления функций принадлежности.

Отметим, что данный алгоритм может применяться и для многоместных арифметических операций, однако вычислительные трудности в этом случае существенно возрастают.

1.9. Нечеткие выводы

В системах управления, использующих методы теории нечетких множеств, алгоритм управления описывается множеством правил типа:

$$\text{«если } \langle \text{условия} \rangle, \text{ то } \langle \text{действие} \rangle\text{»}, \quad (1.21)$$

которые часто называют продукционными правилами, а также правилами условного логического вывода, или правилами нечеткого вывода.

Логическое выражение, стоящее после «если», называют антецедентом, предпосылкой и т.п., а стоящее после «то», – заключением операции, выводом, следствием и т.п.

В общем случае при проектировании системы создается некоторая система правил вида (1.21), в которых правая и левая части содержат комбинации лингвистических значений соответствующих переменных. Для этих лингвистических значений определяются соответствующие функции принадлежности, которые подвергаются обработке, характер которой определяется структурой правила.

Пусть существуют знания эксперта о том, что необходимо открыть регулирующий клапан, если уровень воды поднимается, т.е. существует правило:

$$\text{«Если уровень воды высокий, то открыть»}. \quad (1.22)$$

Лингвистические значения “высокий” (рис. 1.30, а) и “открыть” (рис. 1.30, б) представляются функциями принадлежности.

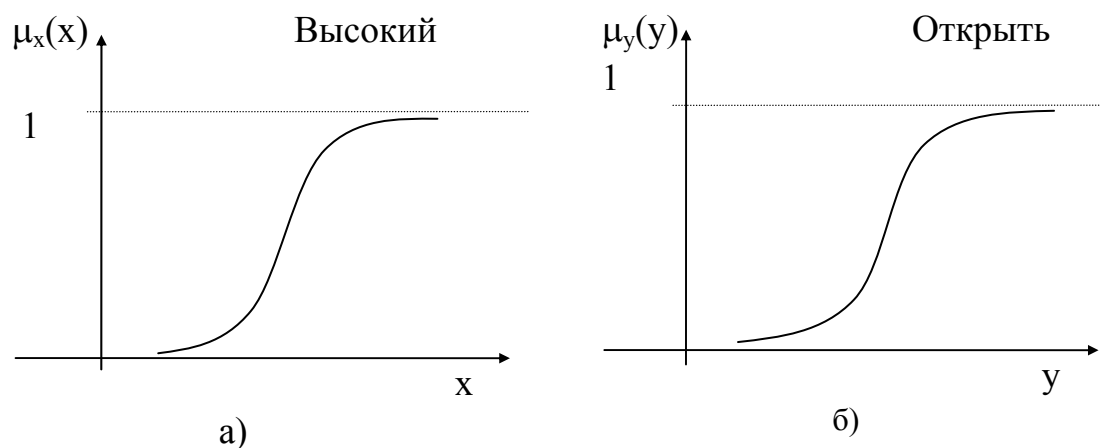


Рис. 1.30

Предположим теперь, что уровень воды соответствует предположению “Довольно высокий”, которое представляется соответствующей функцией принадлежности (рис. 1.31).

Какую операцию необходимо выполнить в данном случае над регулирующим клапаном? Очевидно, что решение следует искать на основе сопоставления уже имеющегося правила (1.22) и наблюдения, характеризуемого лингвистическим значением “Довольно высокий”, т.е. решить задачу

“Если уровень высокий, то открыть”

“Если уровень довольно высокий, то?”

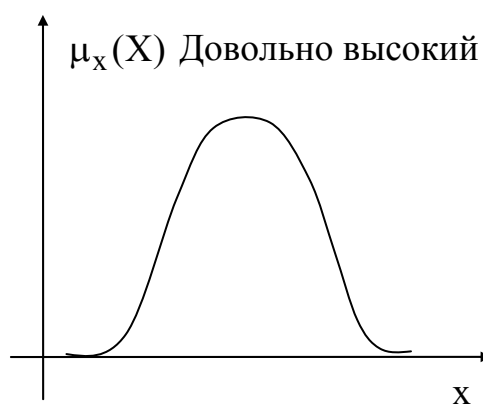


Рис. 1.31

Обычная человеческая логика дает следующий результат:

“Если уровень высокий, то открыть”

“Если уровень довольно высокий, слегка открыть”.

Однако для практической реализации последнего вывода надо указать конкретное значение угла поворота регулирующего клапана, которому соответствует лингвистическое значение “слегка открыть”. На рис. 1.32 представлена графическая интерпретация операций над функциями при-

надлежности, с помощью которых можно получить конкретное управляющее воздействие.

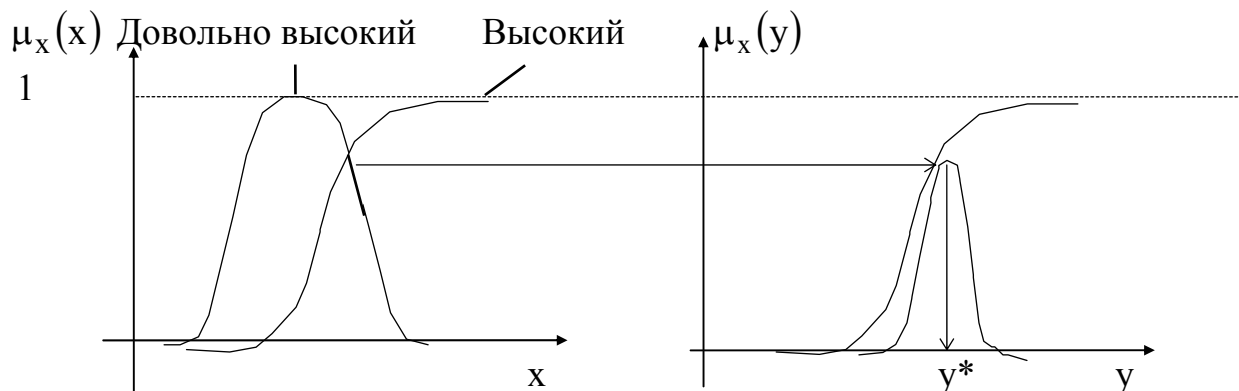


Рис. 1.32

Эти преобразования можно записать в виде формул, используя рассмотренные ранее операции над нечеткими множествами.

$$\mu_z(z) = \min\{\mu_{\tilde{x}'}(x), \mu_{\tilde{x}}(x)\},$$

где $\mu_{\tilde{x}}(x)$ – функция принадлежности терма “Высокий”;

$\mu_{\tilde{x}'}(x)$ – функция принадлежности терма “Довольно высокий”.

$$Z = x' \wedge x; z \in Z; \mu_{y'}(y) = \min\{\mu_{\tilde{y}}(y), \mu_z(z)\},$$

где $\mu_y(y)$ – функция принадлежности терма “Открыть”.

$$y' = y \wedge Z; y^* \rightarrow \max\{\mu_{y'}(y)\},$$

где y^* – управляющее воздействие, соответствующее рассмотренной ситуации.

Полученное правило вывода значения управляющего воздействия назвали «максиминное правило». В более формальной записи приведенные преобразования представляют отношения предпосылки и заключения: $R: A \rightarrow B$, которое можно рассматривать как нечеткое множество на декартовом произведении полного пространства предпосылок и полного пространства заключений. Таким образом, процесс получения вывода B' с использованием наблюдения A' и значения $A \rightarrow B$ можно представить в виде следующей формулы. – Предпосылка 1: Если есть A , то $y=B$; предпосылка 2: x есть A' ; следствие: y есть B' .

В приведенных примерах рассматривалась простейшая форма нечеткого продукционного правила. Возможны и более сложные:

Если $\langle \quad \rangle$, то $\langle \quad \rangle$, иначе $\langle \quad \rangle$.

Если $\langle \alpha \rangle$ и $\langle \beta \rangle$, то $\langle \alpha \wedge \beta \rangle$.

Если $\langle \alpha \rangle$ или $\langle \beta \rangle$, то $\langle \alpha \vee \beta \rangle$,

где между знаками $\langle \rangle$ заключены некоторые нечеткие высказывания: β есть α , где β – имя лингвистической переменной; α – её лингвистическое значение. В дальнейшем для простоты слово “есть” будем заменять знаком равенства “=”.

Необходимо отметить, что в строгом смысле отождествление союзов и, или, если...то, если...то...иначе с операциями конъюнкции, дизъюнкции, отрицания и импликации справедливы только при выполнении свойств коммутативности, ассоциативности и дистрибутивности высказываний, образующих предложения. Обычно эти условия выполняются, что позволяет нам в дальнейшем отказаться от специальных проверок.

В общем случае в условной части правил вывода могут присутствовать сложные логические конструкции, поэтому первый этап обработки правил заключается в упрощении условной части путем применения к ней в соответствии с ее структурой операций, определенных для нечетких множеств.

Нечеткое продукционное правило “Если..., то...” можно рассматривать как высказывание имплицативной формы. Тогда нечеткому продукционному правилу

$$\text{если } x_1 = A, \text{ то } y_1 = B, \quad (29)$$

где x_1 и y_1 – лингвистические переменные; A и B – лингвистические значения x_1 и y_1 , которым соответствуют нечеткие множества $A = \{\mu_A(x)/x\}$, $B = \{\mu_B(y)/y\}$, согласно логике Лукасевича может быть поставлена в соответствии результирующая функция принадлежности:

$$\mu(x, y) = 1 \wedge (1 - \mu_A(x) + \mu_B(y))$$

или согласно Л. Заде

$$\mu(x, y) = (1 - \mu_A(x) \vee \mu_B(y)).$$

Итак, мы получили два варианта преобразования нечетких продукционных правил “Если..., то...”. Первые варианты этих преобразований были предложены Л. Заде, затем В. Мамдани [24] предложил свои. В настоящее время известно более десятка различных преобразований. Подробный анализ с учетом выполнения различных условий проведен в [26]. В настоящее время наиболее часто используются следующие преобразования:

1. $\mu(x, y) = \min\{\mu_A(x), \mu_B(y)\} = \mu_A(x) \wedge \mu_B(y)$.

2. $\mu(x, y) = \min\{\mu_A(x), \mu_B(y)\} \vee \min\{1 - \mu_A(x), 1\}$.
3. $\mu(x, y) = \min\{1, (1 - \mu_A(x)) + \mu_B(y)\}$.
4. $\mu(x, y) = \max\{(1 - \mu_A(x)), \mu_B(y)\}$.

Преобразование 1 удобно тем, что оно сохраняет ширину значения функции принадлежности и позволяет выделить каждое преобразование и процесс его построения даже из информации в табличной форме. Среди недостатков этого правила можно отметить коммутативность, отсутствие разницы между выражениями типа $(A \wedge B) \rightarrow C$ и $A \rightarrow (B \rightarrow C)$ и невозможность использовать связку “ИЛИ” вместо “И” для интерпретации связки “ИНАЧЕ” для получения протокола применения правил:

Правило 1, иначе Правило 2, иначе...

Другие правила лишены этого недостатка за счет того, что каждому правилу нельзя поставить в соответствие его область влияния. Например, арифметические связки в Правиле 3 приводят к получению новых значений функции принадлежности, что требует выполнения аппроксимации. Правило 4 лишено всех указанных недостатков и является наиболее “человеческим” [26] по природе, так как, если предпосылка А дает следствие Б, то предпосылка А, близкая к А, дает следствие Б, близкое к Б. Это свойство особенно важно для систем с участием “человеческого фактора”, где все ситуации не могут быть заданы с помощью набора правил.

Контрольные вопросы

1. Что характеризует функция принадлежности?
2. Укажите основные требования по построению функций принадлежности.
3. Какие методы построения функций принадлежности позволяют проверить корректность экспертных оценок?
4. Приведите варианты определения операций пересечения и объединения в теории нечетких множеств.
5. В чем заключаются основные особенности нечеткой арифметики?
6. Как строятся правила условного логического вывода?
7. Укажите основные методы вычисления импликации при обработке правил условного логического вывода.
8. Какие методы используют для нахождения решения при обработке правил условного логического вывода?

Глава 2

НЕЧЁТКИЕ КОНТРОЛЛЕРЫ

2.1. Общие принципы построения

Расширение сферы применения автоматизированных систем управления, повышение уровня ответственности этих систем при решении различных задач управления в промышленной и непромышленной сфере поставило перед разработчиками ряд сложных и практически противоречивых проблем. С одной стороны, необходимо усиливать возможности этих систем, реализовывать всё более сложные алгоритмы управления, с другой – всё более жёсткие требования предъявляются к уровню их надёжности.

Повышение возможностей и качества управления за счёт только усовершенствования аппаратной части дало определённый эффект, но в то же время указало на наличие существенных ограничений на этом пути. Известно, что повышение гибкости систем управления за счёт усложнения аппаратной реализации приводит к ухудшению отказоустойчивости и живучести систем. Существенный рост сложности объектов управления создал определённые трудности и теоретического характера.

Разработка систем управления невозможна без построения модели объекта управления. Основное содержание данной задачи состоит в решении следующих проблем [30]:

- описании процессов, происходящих в системах управления;
- выборе соответствующих методов формализации и установлении адекватности получаемых моделей с исходным объектом, а также с методами исследования (в зависимости от уровня физической и математической строгости).

Следует отметить, что традиционный подход к решению задач теории управления на основе существующих в прикладной математике формально-логических методов ставит своей целью создание точных (в широком смысле) моделей, строгих рассуждений и выводов. Основное внимание при этом приходится уделять вопросам корректности, полноты, непротиворечивости, замкнутости, устойчивости, управляемости и многим другим качественным аспектам описания моделей объектов и алгоритмов управления.

Следует отметить, что процесс построения моделей физических процессов носит сложный эволюционный характер, связан с неизбежной аппроксимацией реального объекта и приводит к потере информации при его описании. При этом гипотезы и аксиомы, по которым осуществляются аппроксимация и представление реального объекта соответствующей моделью, могут не учитывать всей реальной сущности физического процесса, что приводит к дополнительному приращению риска и неопределённости в описании объекта управления. Здесь уместно напомнить принцип несовместимости Л.Заде: "По мере возрастания сложности системы наша способность формулировать точные, содержащие смысл утверждения о её поведении уменьшаются вплоть до некоторого порога, за которым точность и смысл становятся взаимоисключающими".

Таким образом, вопрос оценки истинности утверждения: "Математическая модель адекватна реальному объекту управления" – остаётся открытым и не решается только в рамках принятого метода исследований. Оценка адекватности математической модели реальному объекту управления носит относительный характер и существенно зависит от принятых гипотез описания исследуемой динамической системы.

Одним из возможных направлений разрешения перечисленных выше проблем является создание "интеллектуальных" систем управления, в которых управление и принятие решений реализуются на моделях нечётких регуляторов и экспертных систем с использованием нечётких индукций и алгоритмов управления, обобщённых правил нечёткого логического вывода. С точки зрения нечётких моделей рассматриваются два варианта:

1) нечёткость описания как аппроксимация слабоструктурированной модели реального объекта управления из-за его сложности и нечёткости информации о его свойствах;

2) реальный объект обладает объективной внутренней нечёткостью описания функционирования [33].

В первом случае оценка адекватности модели реальному объекту устанавливается нечёткой мерой отношения между исследуемыми объектами или системами и методами имитационного моделирования. Особое значение такой подход имеет при построении промышленных интеллектуальных САУ, основанных на знаниях и использующих в структуре контуров управления соответствующие устройства с искусственным интеллектом [25, 26, 3, 29, 27, 21, 33].

Во втором случае исследования полноты соответствующей оценки адекватности объекта управления показали, что существует класс динамических систем, для которых истинность суждений об адекватности систем, принадлежащих этому классу, своим моделям, не может принципиально принимать булевы значения $\{0,1\}$.

В работе [17] сформулированы пять принципов организации интеллектуальной управляющей структуры.

Первый принцип. Наличие взаимодействия управляющих систем с реальным внешним миром с использованием информационных каналов связи.

Второй принцип. Принципиальная открытость систем с целью повышения интеллектуальности и совершенствования собственного поведения.

Третий принцип. Наличие механизмов прогноза изменений внешнего мира и собственного поведения системы в динамически меняющемся внешнем мире.

Четвёртый принцип. Наличие у управляющей системы многоуровневой иерархической структуры, построенной в соответствии с правилом повышения интеллектуальности и снижения требований к точности моделей по мере повышения ранга иерархии в системе (и наоборот).

Пятый принцип. Сохраняемость функционирования (возможно, с некоторой потерей качества или эффективности, иначе с некоторой деградацией) при разрыве связей или потере управляющих воздействий от высших уровней иерархии управляющей структуры.

Подробное рассмотрение этих принципов представлено в [17]. В соответствии с этими принципами были определены два типа интеллектуальных управляющих систем.

Определение 1. Управляющие системы, организованные и функционирующие в соответствии со сформулированными пятью принципами (в полном их объёме), называются управляющими системами, обладающими свойством "интеллектуальности в большом".

Определение 2. Управляющие системы, структурно не организованные в соответствии с приведёнными выше пятью принципами, но исполь-

зующие при функционировании знания (например в виде правил) как средство преодоления неопределённости входной информации, модели управляемого объекта или его поведения, называются управляющими системами, обладающими свойством "интеллектуальности в малом".

Примером управляющих систем со свойством "интеллектуальности в малом" служат нечёткие контроллеры.

Определение 3. Нечётким регулятором (контроллером) называется иерархическая двухуровневая система управления, "интеллектуальная в малом", на нижнем (исполнительном) уровне которой находится традиционный ПИД-регулятор, а на верхнем координационном уровне используется база знаний (с блоком нечёткого вывода в виде продукционных правил с нечёткой импликацией) и устройства перевода в лингвистические и в чёткие значения (фазификатор и дефазификатор соответственно).

На рис. 2.1 приведена типовая структура нечёткого контроллера.

Как правило, во всех нечётких регуляторах используется основной принцип регулирования – принцип регулирования по отклонению [2].

Выходная переменная объекта управления сравнивается с заданным значением x_r , ошибка рассогласования

$$E = x_r - x$$

обычно подвергается различным масштабным преобразованиям. Кроме самого значения рассогласования вычисляется скорость изменения рассогласования \dot{E} . Полученные числовые значения преобразуются фазификатором в соответствующие лингвистические значения.

Используя эти значения и знания, хранящиеся в базе знаний, процессор вывода определяет лингвистический эквивалент управляющего воздействия, который с помощью дефазификатора преобразуется в числовую форму. Во всех этих операциях участвует база знаний контроллера, которая, можно считать, состоит из двух частей: базы данных и базы правил. База данных содержит лингвистические значения всех используемых переменных и соответствующие базовые множества. При работе фазификатора определяется принадлежность E и \dot{E} конкретным базовым множествам с соответствующими лингвистическими значениями. Дефазификатор решает обратную задачу, при которой для лингвистического значения

управления определяется базовое множество и находится точка единственного управляющего воздействия.

Координационный уровень

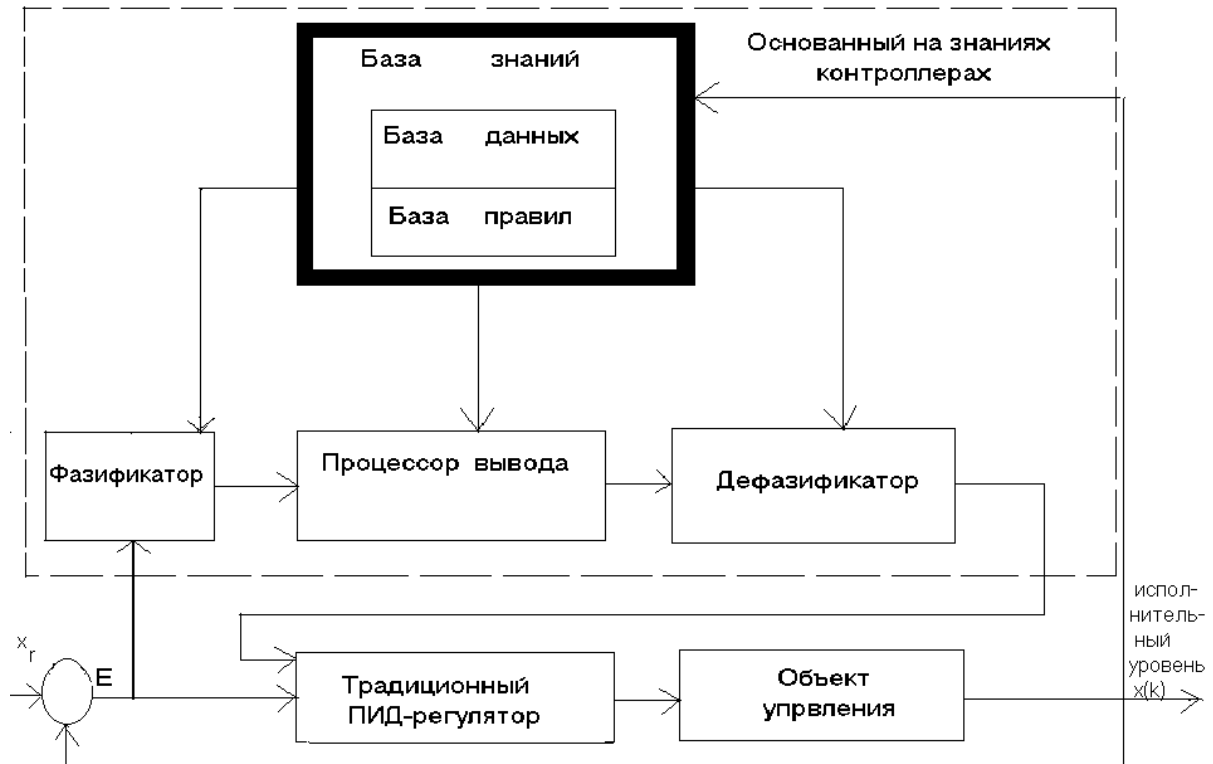


Рис. 2.1

База правил нечётких регуляторов строится на основе продукционной модели знаний, имеющей конструкцию вида "если...,то...". Каждая продукция представляется в виде множества пар "ситуация – действие" и позволяет ставить в соответствие со сложившейся ситуацией действие регулятора в виде значения регулирующего воздействия.

Обычно левая часть каждой продукции представляется конъюнкцией элементарных условий, а правая часть – как множество действий. Для регулятора, представленного рис. 2.1, любое правило в базе знаний может быть представлено в виде:

$$\text{если } (E \text{ есть } E^*_1) \text{ и } (\dot{E} \text{ есть } \dot{E}^*_1), \text{ то } (U \text{ есть } U^*_1),$$

где $E^*_1, \dot{E}^*_1, U^*_1$ – лингвистические значения переменных E, \dot{E}, U .

Основной проблемой создания нечёткого регулятора является конструирование базы знаний, содержащей опыт и знания человека – оператора.

Заполнение базы знаний может выполняться различными способами [7]: оператор-эксперт управляет технологическим процессом, за которым "наблюдает" регулятор, запоминая все действия эксперта и заполняя свою базу знаний; оператор-эксперт описывает своё действие при каждой наблюдаемой ситуации в виде продукции "если..., то... ", которые и будут образовывать базу знаний регулятора; перед самоорганизующимся нечётким регулятором ставится цель обеспечить желаемую переходную характеристику проектируемой системы и одновременно сообщается некоторая информация о технологическом процессе (объекте управления). Регулятор самостоятельно (методом проб и ошибок) накапливает знания без эксперта.

Для построения базы знаний нечёткого контроллера может использоваться и более сложная структура: "ситуация – стратегия управления – действие"[25].

Кроме рассмотренной структуры возможны и более сложные, способные адаптировать к изменениям окружающей среды путем переключений на другие множества лингвистических значений, продукционные правила, базы знаний [2].

Совокупность – фазификатор, процесс вывода, дефазификатор – можно рассматривать как нечёткий процессор (НПР). В реализации последнего в настоящее время возможны следующие направления:

- 1) программная эмуляция нечётких процессоров на персональных компьютерах или промышленных контроллерах;
- 2) применение плат-ускорителей для ПК, реализующих нечёткие алгоритмы управления;
- 3) создание специализированных нечётких процессоров.

Эмуляция нечётких процессоров на персональных компьютерах не вызывает в настоящее время каких-либо затруднений. Аппаратные и программные возможности современных ПК позволяют воспроизводить все необходимые операции.

Однако требование управления в реальном масштабе времени может быть обеспечено традиционными ПК на ограниченном количестве продукционных правил.

Значительно лучшие перспективы здесь имеют ЭВМ на RISC-процессорах. Общим недостатком этих решений является аппаратная и программная избыточность подобных решений. В то же время высокий

уровень развития ПК делает их удобным средством разработки систем, использующих методы теории нечётких множеств.

Платы-ускорители могли бы взять на себя основные функции по выполнению операций фазификации, обработки продукционных правил и дефазификации, при этом ПК использовался бы как интеллектуальное устройство ввода-вывода и местонахождение базы данных и правил. При таком решении большое значение имеет рациональная организация обмена между ПК и платой-ускорителем.

Платы-ускорители должны представлять собой сопроцессоры, ориентированные на выполнение определённого класса операций с нечёткими множествами (числами).

В первую очередь это должны быть системы, работающие с продукционными правилами.

При разработке соответствующих плат, видимо, следует решить несколько проблем, связанных:

1) с самой структурой продукционных правил, имея в виду, что правила могут быть:

простыми: если $\langle \text{простое условие} \rangle$, то $\langle \text{действие} \rangle$;

сложными: если $\langle \text{простое условие} \rangle$, то $\langle \text{действие} 1 \rangle$,

иначе $\langle \text{действие} 2 \rangle$;

составными: если $\langle \text{простое условие} 1 \rangle$, то

если $\langle \text{простое условие} 2 \rangle$, то $\langle \text{действие} 1 \rangle$.

2) со структурой причинной части продукционных правил, которая может представлять собой некоторую композицию, вообще говоря, произвольной сложности, простых условий;

3) решение может выбираться по одному имеющему наибольший уровень выполнения продукционному правилу или по комбинации нескольких продукционных правил;

4) с типом алгоритма обработки продукционного правила: максимальный алгоритм, имплицитивный (Заде, Лукасевич, Мамдани).

В то же время, независимо от конкретной реализации нечеткого процессора и содержания пунктов 1-4, необходима реализация основных этапов:

- фазификация – отображение множества значений входных переменных на множество их лингвистических значений;

- процедура вывода, когда по установленным лингвистическим значениям из базы правил выбирается продукционное правило или правила с наибольшим уровнем истинности и затем вычисляется результирующая свертка функций принадлежности, входящих в правило (правила);
- дефазификация, когда происходит определение единственного управляющего воздействия.

Следует иметь в виду, что эти этапы последовательно выполняются в каждом цикле управления и эффективность их реализации будет непосредственно сказываться на процессе управления.

2.2. Фазификация. Средства реализации

В общем виде фазификация – это выполнение отображения множества числовых значений управляющих параметров на множество лингвистических значений, определенных для этих параметров

$$X \xrightarrow{\Gamma} L_x,$$

где X – множество числовых значений;

L_x – множество лингвистических значений;

Γ – оператор отображения.

Для задания отображения Γ диапазон возможных значений X разбивается на совокупность базовых множеств $B = \{b_l; l = \overline{1, I_L}\}$, количество которых I_L определяется числом используемых лингвистических значений.

Функции принадлежности лингвистических значений

$$\mu_l(x), \quad x \in X, \quad l \in L$$

определяются на базовых множествах с учетом требований, которые должны быть выполнены при построении функций принадлежности [36].

Таким образом, в процессе фазификации решается вопрос о принадлежности конкретного значения управляющего параметра конкретному базовому множеству. В качестве критерия принадлежности используются значения соответствующих функций принадлежности. В ситуации, пред-

ставленной на рис. 2.2, переменная x' может быть отнесена к базовому множеству b_{L+1} , так как

$$\mu_{L+1}(x) > \mu_L(x). \quad (2.1)$$

Числовому значению x' ставится в соответствие лингвистическое значение “ $L+1$ ”.

Отметим, что согласно правилам построения функций принадлежности значение x' может быть соотнесено не более чем с двумя лингвистическими значениями.

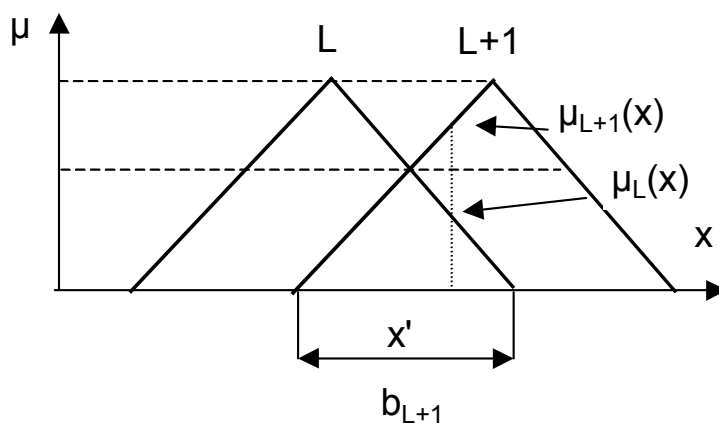


Рис. 2.2

Как правило, для установления соответствия между числовым значением и лингвистическим используют соотношение (2.1). Однако могут возникнуть ситуации, когда следует учесть то, что числовой оценке управляющего параметра в данный момент времени могут соответствовать несколько (два) лингвистических значения. Тогда значения функций принадлежности можно рассматривать как степень истинности соответствующего решения. Очевидно, что это обстоятельство должно учитываться в дальнейшем.

Построение базовых множеств и выбор вида функций принадлежности выполняются экспертным путем. Их количество можно выбирать исходя из условий полноты и избыточности. Следует отметить, что в нечетких контроллерах можно предусмотреть вариант пересмотра в процессе эксплуатации структуры базовых множеств. Относительно вида функций принадлежности можно высказать предположение, что при многократном выполнении одних и тех же операций вид функции принадлежности принципиального значения не имеет и на первый план выходят вопро-

сы простоты соответствующих технических решений. Видимо, этими соображениями можно объяснить тот факт, что в большинстве технических приложений используются простейшие треугольные функции, для которых имеется очень простое (L, R) представление. При условии использования нормальных функций принадлежности (что практически всегда имеет место)

$$\text{Sup}[\mu(x)] = 1 \text{ или } \max[\mu(x)] = 1,$$

треугольная функция принадлежности однозначно задается тройкой (x_L, x^*, x_R) , где x_L – левая граница носителя нечеткого множества; x_R – правая; x^* – точка максимума функции принадлежности:

$$\mu(x_L) = \mu(x_R) = 0, \mu(x^*) = 1.$$

Достаточно важным вопросом является выбор шкалы, в которой представлены аргументы функций принадлежности.

Возможны два варианта:

- использование абсолютных шкал в физических единицах, в которых представляются переменные, участвующие в управлении;
- использование относительных шкал.

В отношении первого варианта можно отметить следующее. В процессе управления могут участвовать переменные различной физической природы с различными масштабами абсолютных шкал. Их обработка в рамках одного правила условного логического вывода будет сопряжена с использованием большого количества масштабных коэффициентов с различными операциями пересчета и т.п. Это создает дополнительные и мало обоснованные сложности. Значительно удобнее приведение всех переменных к относительной шкале $[0,1]$ или $[-1,1]$. Формулы приведения тривиальны:

для шкалы $[0,1]$

$$\bar{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}};$$

для шкалы $[-1,1]$

$$\bar{x} = \frac{x - \frac{x_{\max} + x_{\min}}{2}}{x_{\max} - x_{\min}}.$$

Хотя принципиальных ограничений на выбор этих шкал нет, шкалу $[-1,1]$ целесообразно использовать, если переменные, участвующие в процессе управления, могут принимать отрицательные значения и характер правил условного логического вывода зависит от знака используемых в нем переменных.

2.3. Средства воспроизведения функций принадлежности

Основным и естественным требованием к нечеткому контроллеру является обеспечение управления в реальном масштабе времени.

Пусть $T_{упр}$ время, в течение которого необходимо сформировать сигнал управления нечеткого регулятора, имеющего структуру, приведенную на рис. 2.1.

Представим это время в виде суммы

$$T_{упр} = T_F + T_S,$$

где T_S – время, используемое стандартным каналом управления (ПИД – регулятор, объект управления (см. рис. 2.1)). Это время определяется методами классической теории автоматического управления; T_F – время, используемое нечетким контроллером.

В свою очередь

$$T_F = t_{ff} + t_p + t_{df},$$

где t_{ff} – время фазификации;

t_p – время, затрачиваемое на поиск и обработку правил условного логического вывода;

t_{df} – время дефазификации.

В данном пункте остановимся подробнее на времени фазификации t_{ff} и его компонентах.

Нечеткие контроллеры в отличие от традиционных контроллеров являются объектно- (процессорно) ориентированными. В большинстве случаев множества лингвистических значений, правила условного логического вывода и их алгоритмы обработки, а также алгоритм дефазификации достаточно уникальны. Они проектируются под конкретный вид объекта (процесса) управления или весьма узкий класс объектов (процессов). В связи с этим настройка нечеткого контроллера не сводится к изменению нескольких параметров контроллера (коэффициенты передачи, постоянные времени, какие-то константы), а требует замены используемых лингвистических значений, правил условного логического вывода.

Очевидным фактом является то, что для процесса фазификации необходимо воспроизведение функций принадлежности. Современный уровень электронной технологии позволяет говорить о том, что наиболее целесообразным является хранение значений функций принадлежности в микросхемах памяти. Учитывая объем хранимых данных, время доступа стандартных БИС памяти, можно решать задачу хранения значений функций принадлежности с помощью традиционных микросхем памяти. Основным недостатком этого решения будет их неэффективное использование, так как функциональные возможности, ориентированные на вычислительные процессы, передачу больших массивов данных, будут использоваться в нечетких контроллерах далеко не в полной мере.

Более предпочтительным является вариант использования специализированных устройств памяти, учитывающих особенности процесса фазификации. Такие микросхемы приобретают характер заказных. Но опять же, учитывая уровень современных технологий, а также то, что особенности специализированных устройств памяти проявляются только на структурно-функциональном уровне, а не на технологическом, можно считать, что проблемы заказных, специализированных микросхем памяти практически нет.

Анализ различных вариантов использования нечетких контроллеров показал, что в подавляющем большинстве случаев размер области носителя нечеткой переменной, как правило, постоянен для различных лингвистических значений. Исключение могут составлять лишь крайнее левое и крайнее правое значения (рис. 2.3).

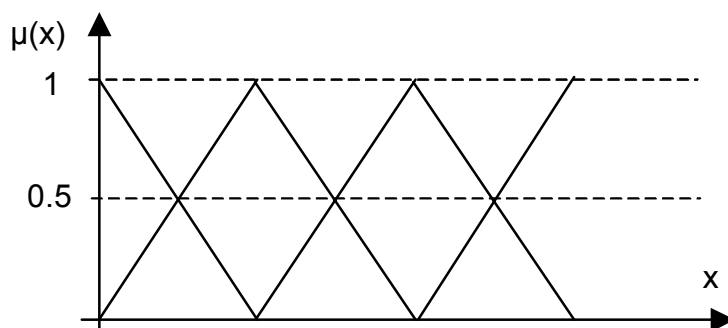


Рис. 2.3

Очевидно, что для генерации любого количества функций принадлежности достаточно хранить один эталон, из которого путем несложных функциональных преобразований аргумента (рис. 2.4. – 2.6.) можно получать все остальные, т.е. если заданы $\mu_0(x_i)$, то при замене x_i на $x_i' = f_k(x_i)$ могут быть получены новые функции принадлежности $\mu_0(x_i')$ (см. рис. 2.4. – 2.6.).

В простейшем случае

$$f_k(x_i) = \alpha_{ok} + \beta_k x. \quad (2.2)$$

Параметры этой функции находятся из очевидных соображений.

Так как для $\mu_0(x_i)$ известны (x_i, x^*, x_R) , то

$$\begin{cases} x_L' = \alpha_{ok} + \beta_k x_L, \\ x_R' = \alpha_{ok} + \beta_k x_R. \end{cases}$$

Откуда

$$\beta_k = \frac{x_R' - x_L'}{x_R - x_L}, \quad \alpha_{ok} = \frac{x_L' x_R - x_R' x_L}{x_R - x_L}.$$

Нетрудно проверить, что при найденных значениях α_{ok} и β_k

$$(x^*) = \alpha_{ok} + \beta_k x^*.$$

В общем виде эта идея реализуется следующей схемой (рис. 2.7).

РЭ – процессорный элемент, реализующий функциональное преобразование (2.2).

В зависимости от реализации элементов этой схемы возможны различные варианты нечеткой памяти, один из которых приведен на рис. 2.8 [21]. Его основу составляет нечеткое постоянное запоминающее устройство, в котором значения функций принадлежности $\mu_0(x)$ и ее аргумента хранятся в виде единичных кодов. Преобразование аргумента эталонной функции принадлежности выполняется по схемному принципу путем соединения вертикальных линий (линий аргумента) с соответствующими ячейками выходного регистра (см. рис. 2.8).

К недостаткам данного решения можно отнести неудобную систему адресации запоминающей матрицы и необходимость преобразования единичных кодов в двоичные.

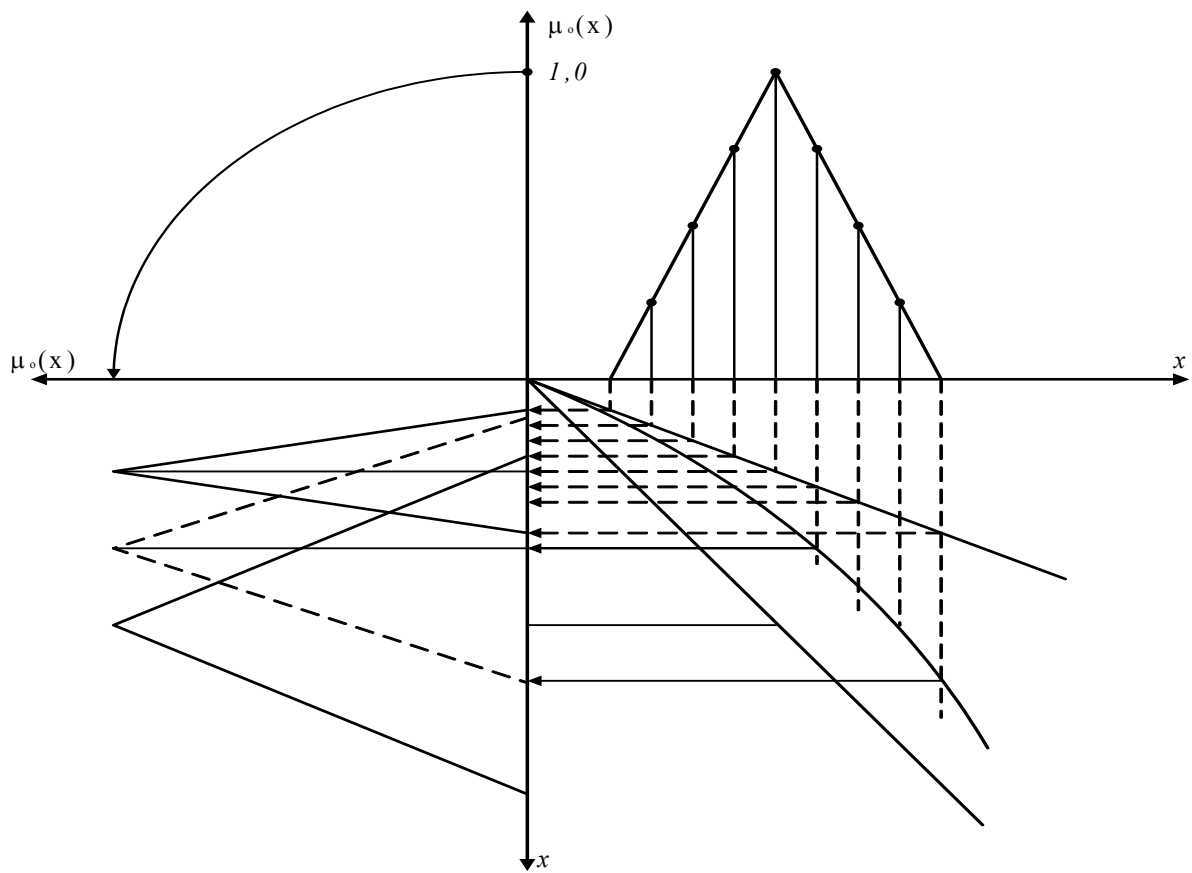


Рис. 2.4

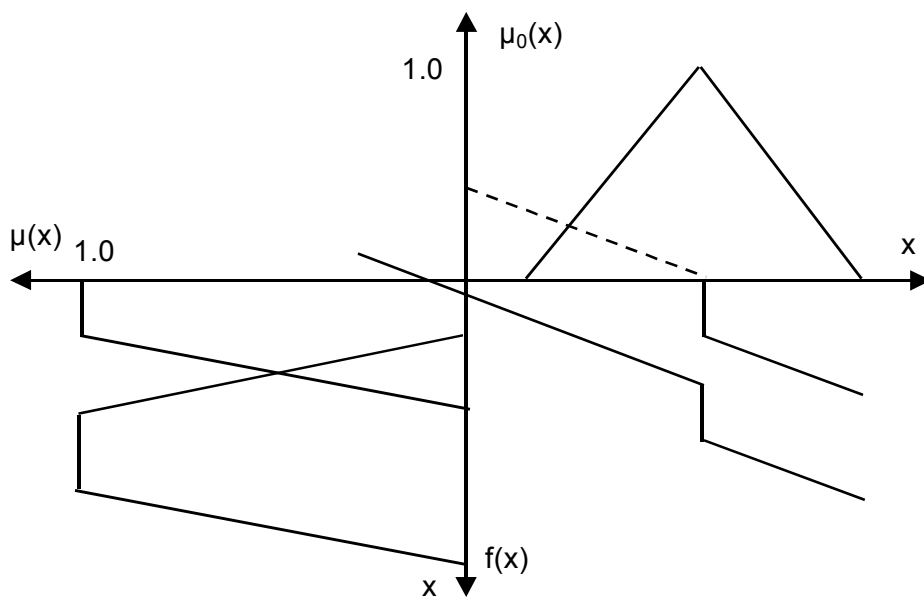


Рис. 2.5

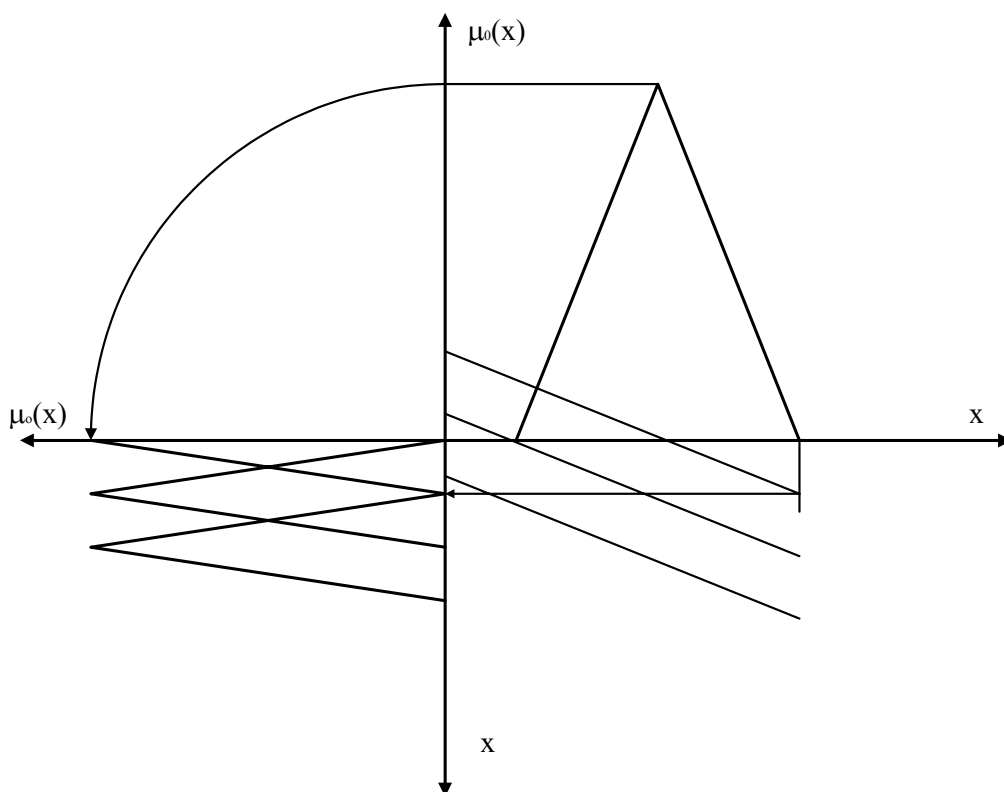


Рис. 2.6. Генерация семейства треугольных функций принадлежности по одному образцу

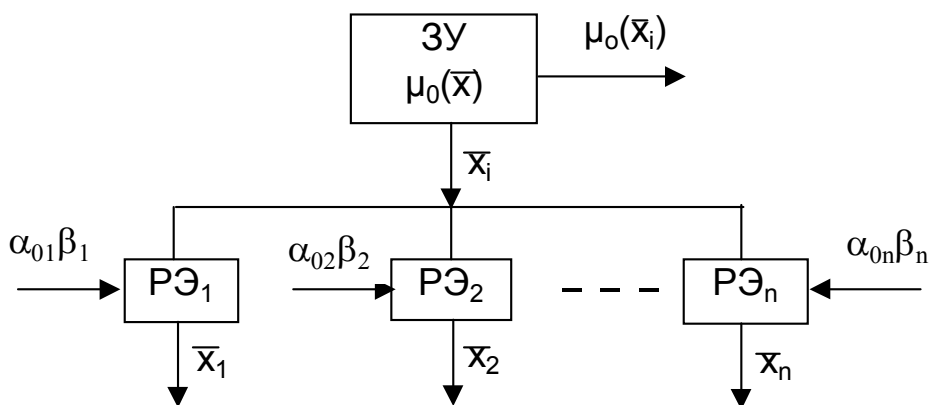


Рис. 2.7

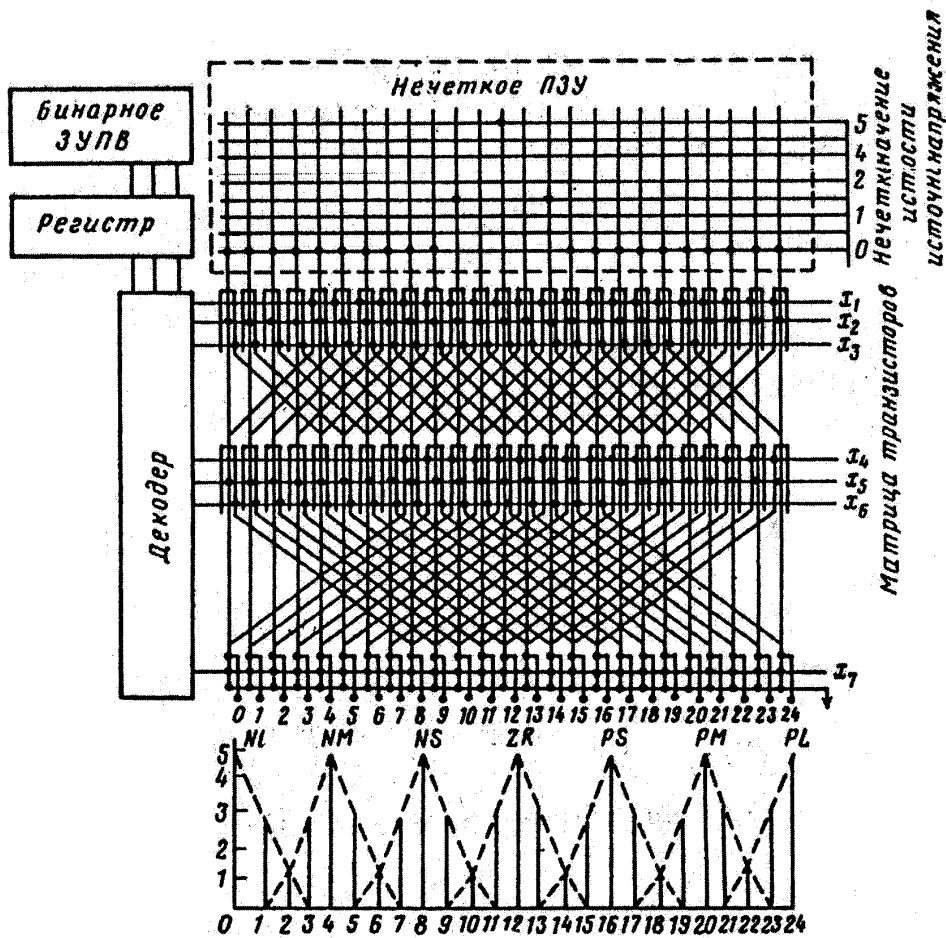


Рис. 2.8

Важным вопросом при проектировании нечетких контроллеров и, соответственно, схем нечеткой памяти является точность предоставления функции принадлежности по ее значению и по аргументу. В большинстве случаев рекомендации имеют эмпирический характер и решения принимаются на основе экспериментов. Объяснить это можно самим характером нечеткого управления. Тем не менее некоторые рекомендации общего характера могут быть все-таки сформулированы. Причем в их основу должны быть положены конечные результаты.

Предположим, что в результате выполнения некоторых процедур обработки правил условного логического вывода была получена результирующая функция принадлежности $\mu_R(x)$, по которой будет приниматься решение по управляющему воздействию. Пусть также значение аргумен-

та, что, собственно говоря, предопределяется методом генерации функций принадлежности, представляется множеством дискретных значений с шагом дискретизации δ_x . Допустим, что решение принимается по максимуму функции $\mu_R(x)$. В силу дискретности представления $\mu_R(x)$ координата максимума x^* может оказаться между точками x_1^* и x_2^* . Вполне реальна ситуация, когда выбор x_1^* или x_2^* не имеет принципиального значения. Однако могут быть и ситуации, когда соответствующий выбор может повлиять на качество управления. Введем переменную P_x , которую назовем порогом выбора (рис. 2.9) и которая задается так, что отсчет x_1^* и x_2^* находятся в середине соответствующих P_x . Тогда если $x^* \in P_{x_i}$, то выбирается x_1^* , если $x^* \in P_{x_{i+1}}$, то x_2^* .

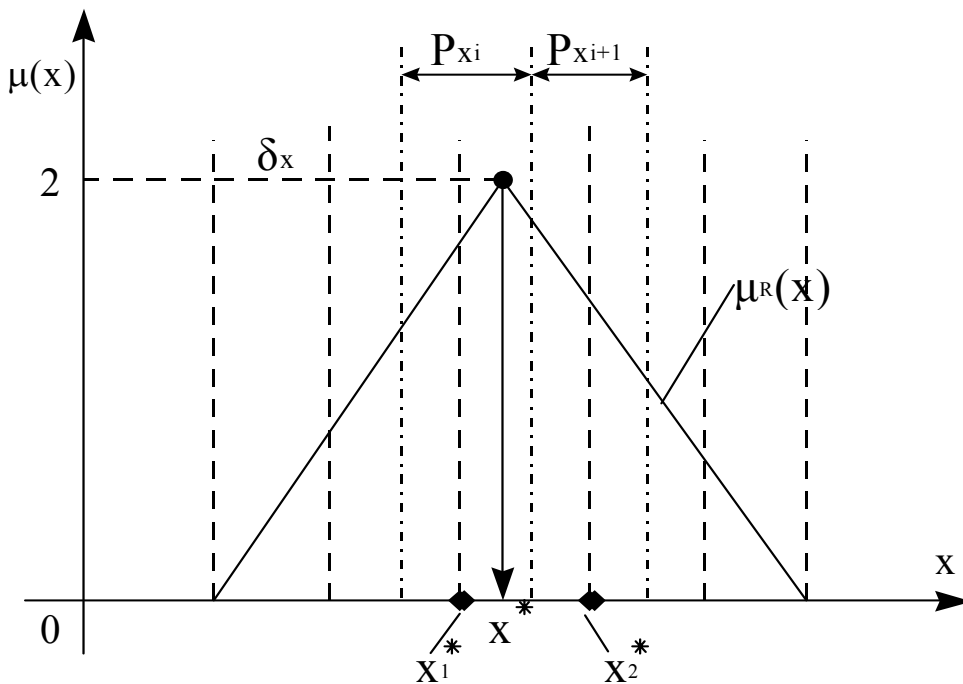


Рис. 2.9

Максимальная погрешность определения управляющего воздействия в этом случае равна

$$\frac{P_x}{2} = d_{\max}$$

Отсюда, если известен размер носителя соответствующего лингвистического значения Δ_i , количество отсчетов

$$n_i = \frac{\Delta_i}{d_{\max}}$$

Если используется m лингвистических значений, то

$$N_x = \sum_{i=1}^m n_i$$

определяет общее количество отсчетов по аргументу функций принадлежности и соответственно количество выходов нечеткой памяти.

Обычно $n_i = n$ и $N_x = \min$.

Аналогичный подход можно использовать и для самих значений функций принадлежности. Дискретное представление $\mu_R(x)$ также приводит к соответствующей погрешности. Однако здесь возможны две ситуации.

В первой при использовании выбора управляющего воздействия по максимуму $\mu_R(x)$ погрешность дискретизации не играет существенной роли, поскольку важно лишь наличие глобального максимума функций принадлежности, а не его абсолютное значение.

Вторая ситуация возникает:

- при выборе управляющего воздействия из нескольких альтернативных;
- при определении управляющего воздействия как средневзвешенного из нескольких близко расположенных друг к другу.

В последнем случае в качестве весовых коэффициентов используются значения соответствующих функций принадлежности.

В этих ситуациях существенны уже абсолютные значения функций принадлежности и свою роль может сыграть погрешность дискретизации. Вводя порог P_m (рис. 2.10) аналогично P_x , можно добиться, что максимальная погрешность

$$\max(d_\mu) = P_m/2,$$

соответственно количество отсчетов по значениям μ_x

$$N_\mu = 1/d_\mu.$$

Таким образом, количество выводов нечеткой памяти N_{FM} для генерации значений функций принадлежности равно:

$$N_{FM} = N_x + N_\mu$$

и оно зависит от требований к точности воспроизведения функций принадлежности, обусловленных необходимым качеством управления.

Если позволяет динамика процесса, то сдвиг функции принадлежности по оси аргумента можно выполнять с помощью так называемых быстрых сдвигателей, комбинационных схем, позволяющих за один такт выполнять сдвиг на произвольное число разрядов в обоих направлениях.

Но в этом случае функции принадлежности будут генерироваться последовательно во времени, хотя и с высокой скоростью. Достоинством такого решения можно считать более простую структуру соединений.

Как уже отмечалось, недостатком рассмотренного варианта нечеткой памяти является ее усложнение из-за необходимости преобразования единичных кодов значений функции принадлежности и ее аргумента в двоичный код. Ограниченное количество значений функции использование для хранения и генерации схем, аналогов ПЗУ принадлежности делает целесообразным с линейной выборкой (рис. 2.11). Значения функции принадлежности представляются в целочисленном формате и снимаются с вертикальных шин, горизонтальные используются для задания аргумента. В данном случае значения функций принадлежности представляются в двоичном формате, а значения аргумента – в единичном.

При разработке нечеткой памяти, представленной на рис. 2.8, руководствовались, по всей вероятности, стремлением

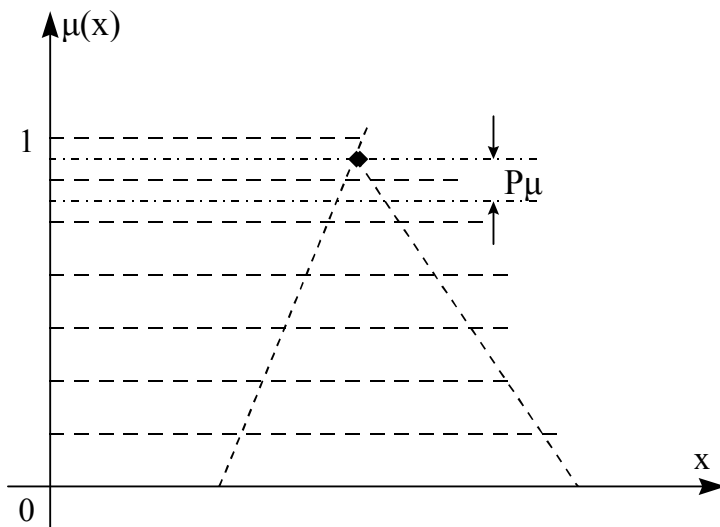


Рис. 2.10

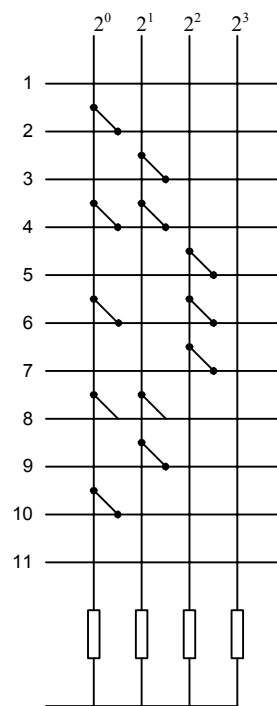


Рис. 2.11

реализовать все в одной микросхеме, применение ПЗУ и современный уровень электронной технологии делают целесообразным страничный принцип организации нечеткой памяти, когда каждая функция принадлежности занимает свою страницу в виде отдельной микросхемы, а адресные линии (линии аргумента) объединены на одном регистре (рис. 2.12).

Такое решение требует одного преобразования двоичного кода в единичный для линий аргументов, кроме того, позволяет просто модифицировать функции принадлежности, а также использовать функции принадлежности разного вида, например трапецеидальные и треугольные.

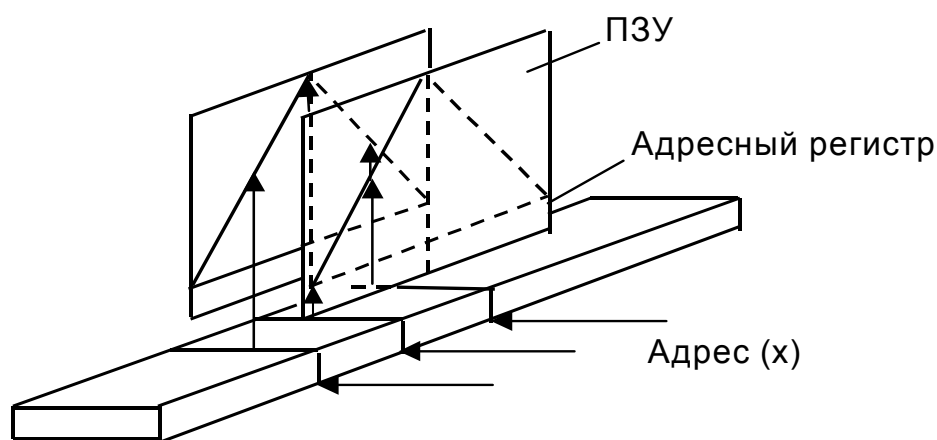


Рис. 2.12

Рассматривая реализацию процесса фазификации, не следует ограничиваться лишь генерацией функций принадлежности. Необходимо также рационально организовать и первичное преобразование сигналов, участвующих в управлении. Для адресации нечеткой памяти, вообще говоря, необходимы коды аргументов, т.е. коды сигналов управления (ошибка, приращение ошибки и т.п.). Эти сигналы имеют преимущественно аналоговый характер, и поэтому нуждаются в аналого-цифровом преобразовании. Особенностью нечеткого управления является то, что здесь не предъявляются высокие требования к точности этого преобразователя, более существенными являются ограничения по времени.

В этих условиях наиболее подходящим является параллельное аналого-цифровое преобразование с практически минимальной разрядностью выходного кода, равной четырем битам.

Не исключается использование АЦП последовательной аппроксимации. Но они явно избыточны по точности преобразования.

Отметим, что высокая скорость АЦП параллельного действия делает допустимым мультиплексирование его входа.

Итак, будем полагать, что переменные управления подвергаются параллельному аналого-цифровому преобразованию, коды которого используются для адресации нечеткой памяти либо непосредственно, либо после преобразования в единичный код.

Дальнейшие схемные решения будут зависеть от того, какой алгоритм определения степени выполнимости правил условного логического вывода принят.

Как уже отмечалось, возможны два варианта:

1. По наибольшим значениям функций принадлежности отбираются лингвистические значения, для которых получены наибольшие значения функций принадлежности, и выбирается правило, в которое входят только эти лингвистические значения. Управляющее решение находится после обработки этого правила. Все остальные лингвистические значения и правила не рассматриваются.

2. Учитываются все лингвистические значения, которые будут установлены в процессе фазификации, определяются все правила вывода, где участвуют эти значения, и управляющее воздействие выбирается из взвешенной свертки этих правил.

Принципиальным моментом является то, что при первом варианте устройство работает, по существу, по пороговому принципу, фиксируя достижения максимумов. Во втором надо запоминать уровни, на которых зафиксировалась та или иная величина, и значения этих уровней впоследствии учитывать, т.е. появляются дополнительные элементы хранения и обработки.

Рассмотрим первый вариант. Известно, что при построении исходных функций принадлежности должны быть выполнены определенные условия [36] (гл. 1). В данной ситуации наиболее существенным является то, что функции принадлежности соседних лингвистических значений должны пересекаться на уровне 0,5 и координата минимума одной должна совпадать с координатой максимума другой (рис. 2.13).

Нетрудно видеть из рис. 2.11 – 2.12, что адресные линии ПЗУ для соседних лингвистических значений будут перекрываться.

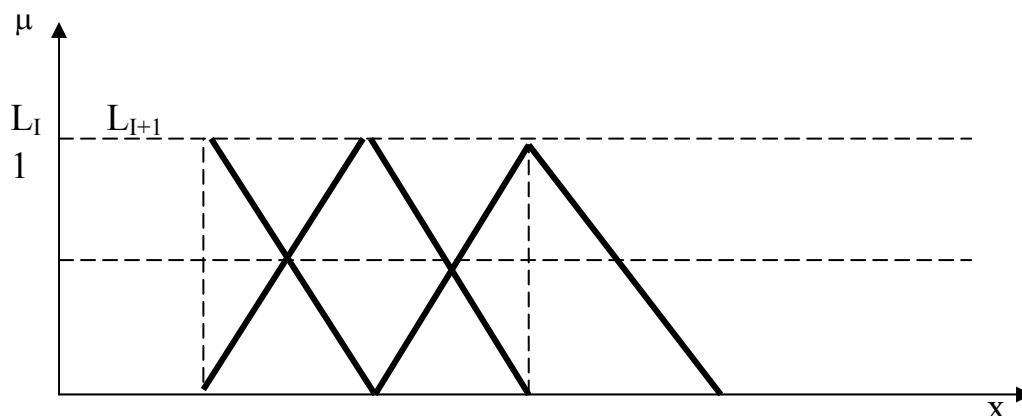


Рис. 2.13

2.4. Организации обработки правил условного логического вывода

Обработка правил условного логического вывода происходит в два этапа:

- поиск по итогам фазификации соответствующего правила или правил;
- обработка установленных правил с целью формирования результирующей функции принадлежности, на основе которой будет формироваться управляющее воздействие.

Рассмотрим случай, когда выбирается только одно правило вывода. Это не ограничивает общность, так как специфика использования нескольких правил вывода проявляется главным образом на этапе выбора управляющего воздействия.

При выполнении поиска правил условного логического вывода главным ограничивающим фактором является время поиска. В зависимости от сложности задачи управления база правил может включать от десятка до сотни, а то и нескольких сотен правил. Поэтому целесообразно вести речь только о параллельных вариантах поиска, когда анализируется несколько правил, а то и все одновременно.

Поиск и обработку правил условного логического вывода будет выполнять специальное устройство, которое назовем *процессором правил*.

Здесь возможны также различные варианты. Первый вариант предполагает разделение устройств хранения правил и последующей их обра-

ботки. В этом случае каждое из правил может рассматриваться как программа для устройства обработки. Каждое правило может храниться в виде двоичного слова, определенной длины, в котором выделяются поля по их функциональному назначению. Практически это аналог микропрограммного управления. Обработывающее устройство может быть построено по двум вариантам. Первый – использование универсального устройства, управляемого микропрограммой-правилом.

Естественно, что обрабатывающее устройство должно быть проблемно-ориентированным.

Это решение может рассматриваться как аналог универсального арифметико-логического устройства с микропрограммным управлением, широко используемым в компьютерах общего назначения. Известный недостаток – последовательное выполнение команд – очевидно, будет иметь место и в рассматриваемом случае. Поскольку обрабатывать предполагается ограниченное число команд, то более близким аналогом будет RISC-архитектура (архитектура с сокращенным набором команд).

Другое решение – это использование набора процессорных модулей, из которых под выбранное правило конфигурируется соответствующая структура. Такое решение также известно в вычислительной технике. Основные трудности будут связаны с ростом аппаратных затрат, необходимых для перестройки структуры. Очевидно, что они будут тем больше, чем сложнее правила.

Общим недостатком этих вариантов является разделение во времени процессов поиска подходящего правила и его обработки.

Процесс поиска правил вывода может быть организован либо по адресному методу, эффективность которого будет падать с увеличением сложности правил, либо по ассоциативному, т.е. поиску по содержанию, что более эффективно, но в то же время требует более сложных технических решений.

В заключение отметим, что рассмотренные решения наиболее эффективны для универсальных контроллеров.

В то же время специфика нечетного управления такова, что более рациональным является создание специализированных нечетких контроллеров, поэтому более целесообразными являются отказ от программных

методов обработки правил условного логического вывода и совмещение в одном устройстве операций хранения и обработки.

Иными словами, для каждого правила предназначается свой процессор, структура которого однозначно отражает структуру правила.

При таком подходе обеспечивается параллельная обработка нескольких правил вывода, но в то же время требует серьезного внимания проблема минимизации аппаратных затрат.

Наиболее рациональным представляется подход, когда для построения процессоров используется некоторый элементарный базовый элемент, на основе которого реализуются все необходимые операции. Это позволит в определенной мере выполнить требования однородности и регулярности, что сделает контроллер более технологичным. Таким образом, возникает задача определения некоторого функционального преобразования, которое должен выполнять базовый элемент.

Ограниченная разность как базовое функциональное преобразование

Функциональное преобразование базового элемента должно отвечать, по крайней мере, двум основным условиям:

- простота реализации;
- функциональная полнота, т.е. с его помощью путем суперпозиции должны реализоваться все определенные в теории нечетких множеств преобразования над функциями принадлежности.

Одним из возможных вариантов является использование ограниченной разности, которая определяется следующим образом (рис. 2.14) [51]:

$$\mu_{A-B} = 0 \vee (\mu_A - \mu_B) = \mu_A - \mu_B = \begin{cases} 0, & \mu_A - \mu_B \leq 0 \\ \mu_A - \mu_B & \end{cases}$$

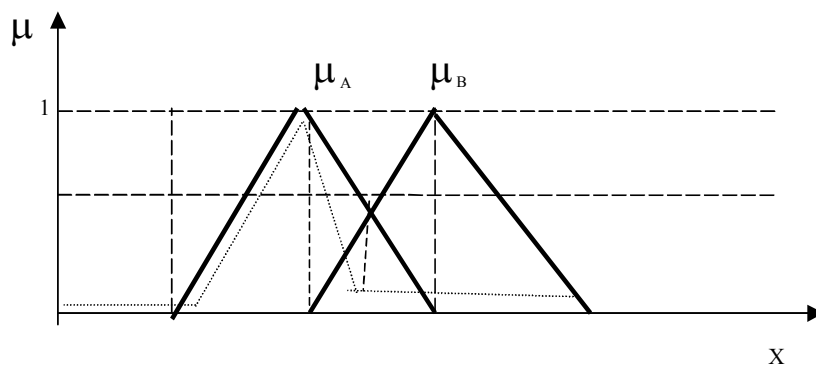


Рис. 2.14

Используя простые графические построения, докажем справедливость следующих тождеств (рис. 2.15 – 2.18):

$$\bar{\mu}_A = 1 - \mu_A = 1 - \mu_A.$$

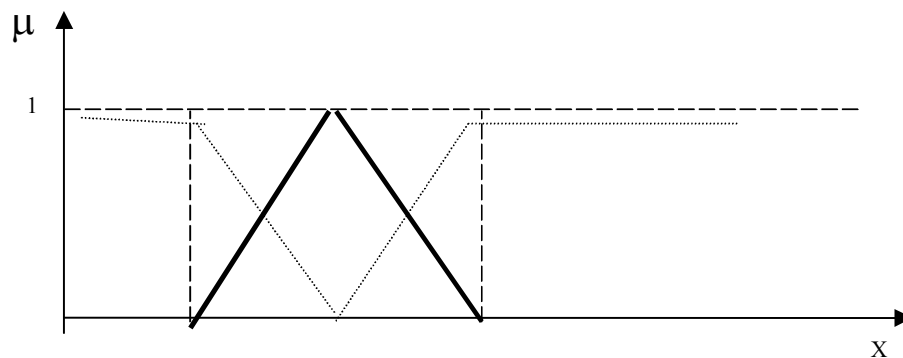


Рис. 2.15

$$\mu_{A \vee B} = (\mu_A - \mu_B) + \mu_B = (\mu_B - \mu_A) + \mu_A.$$

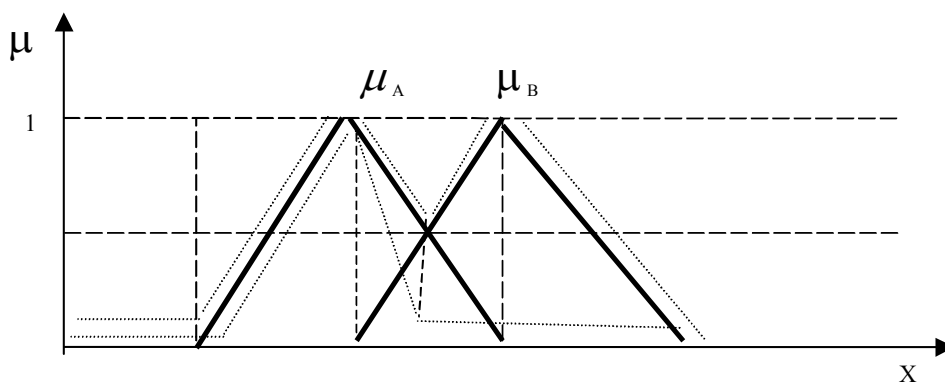


Рис. 2.16

$$\mu_{A \wedge B} = \mu_A - (\mu_A - \mu_B) = \mu_B - (\mu_B - \mu_A).$$

Приведенные тождества указывают на то, что все операции могут быть выполнены через ограниченную разность [21] и алгебраическую сумму. Необходимо отметить, что при реализации операций пересечения и

объединения через ограниченную разность теряется свойство коммутативности этих операций.

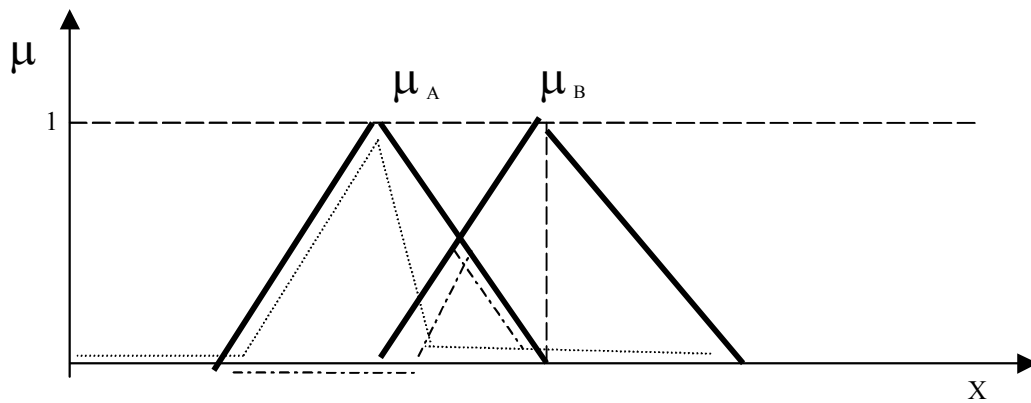


Рис. 2.17

$$\mu_{A \rightarrow B} = 1 \wedge (1 - \mu_A + \mu_B) = 1 - (\mu_A - \mu_B).$$

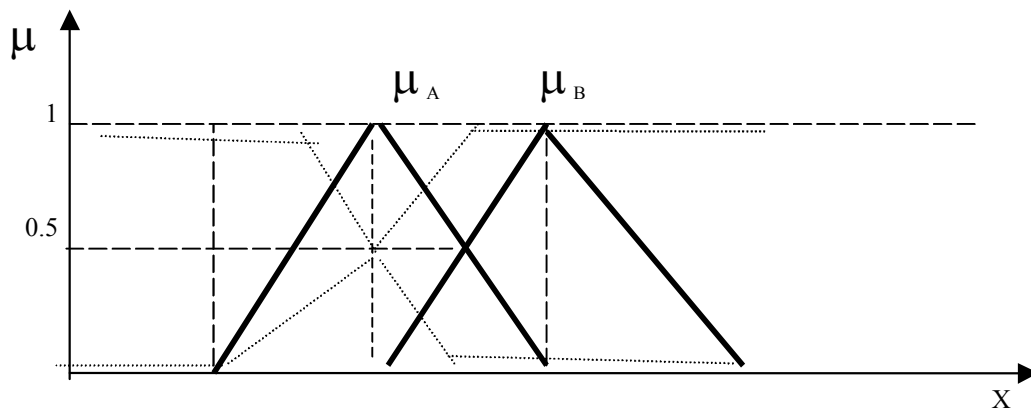


Рис. 2.18

Без доказательства приведем еще четыре операции с их представлением через ограниченную разность [21]:

- ограниченное произведение

$$\mu_{A \circ B} = 0 \vee (\mu_A + \mu_B - 1) = (\mu_A - \mu_B) - 1;$$

- ограниченная сумма

$$\mu_{A \oplus B} = 1 \wedge (\mu_A + \mu_B) = 1 - [1 - (\mu_A + \mu_B)];$$

- эквивалентность

$$\mu_{A \leftrightarrow B} = \mu_{A \rightarrow B} + \mu_{B \rightarrow A} = 1 - [(\mu_A - \mu_B) + (\mu_B - \mu_A)];$$

- абсолютная разность

$$\mu_{[A - B]} = \begin{cases} \mu_A - \mu_B \text{ при } \mu_A \geq \mu_B, \\ \mu_B - \mu_A \text{ при } \mu_A < \mu_B \end{cases} = (\mu_A - \mu_B) + (\mu_B - \mu_A).$$

В качестве примера рассмотрим преобразование правила:

$$\text{если } \langle P1=A \rangle \text{ и } \langle P2=BVC \rangle \text{ то } \langle R=D \rangle,$$

где P1, P2, R-имена нечетких переменных;

A, B, C, D-их лингвистические значения.

Рассмотрим два варианта импликации:

$$1) \mu(Z) = \min\{\mu_D(Z), \min[\mu_A(Z), \max[\mu_B(Z), \mu_C(Z)]]\};$$

$$2) \mu(Z) = \min\{1, 1 - \mu_1(Z) + \mu_D(Z)\},$$

$$\mu_1(Z) = \min\{\mu_A(Z), \max[\mu_B(Z), \mu_C(Z)]\},$$

Z – формальная переменная.

Тогда

$$\mu_1(Z) = \max[\mu_B(Z), \mu_C(Z)] = [\mu_B(Z) - \mu_C(Z)] + \mu_C(Z)$$

и для первого варианта

$$\mu_2(Z) = \min[\mu_A(Z), \mu_1(Z)] = \mu_A(Z) - (\mu_A(Z) - \mu_1(Z)),$$

для второго варианта

$$\mu_3(Z) = \min(\mu_2(Z), \mu_D(Z)) = \mu_2(Z) - (\mu_2(Z) - \mu_D(Z)) = \mu_D \div (\mu_D - \mu_2)$$

либо

$$\mu_3(Z) = 1 - (\mu_2(Z) - \mu_D(Z)).$$

Нетрудно видеть, что и более сложные правила условного логического вывода могут быть представлены через ограниченную разность.

Схемотехническая реализация ограниченной разности

В настоящее время при выборе схемотехнической реализации обычно рассматривают две альтернативы – цифровую и аналоговую. Если речь идет о схемотехнике ЭВМ общего назначения и промышленных контроллеров на основе этой архитектуры, то бесспорное преимущество имеет цифровая схемотехника. В нечетких контроллерах это скорее всего не так. Видимо поэтому нейросетевые технологии и нечеткие контроллеры рассматриваются как ренессанс аналоговой схемотехники. Исследования [21] показали, что различные преобразования функций принадлежности достаточно просто выполняются с помощью аналоговых схем. Например, операции объединения (рис. 2.19) и пересечения (рис. 2.20) реализуют с помощью схем, названных эмиттерами с нечеткими логическими вентилями [21].

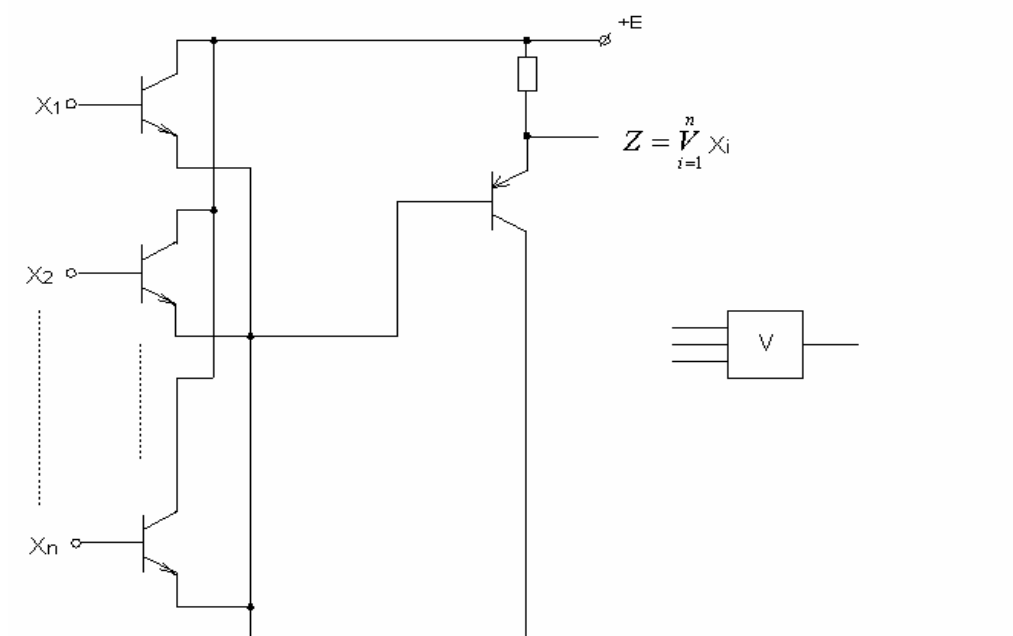


Рис. 2.19

Таким образом, для обработки условной части правил условного логического вывода должна быть построена определенная последовательность из приведенных выше элементов. Пусть, например, имеет место следующая комбинация условий:

$$\text{Если } \langle P1=A \rangle \text{ и } \langle P2=B \vee C \rangle, \text{ то } \dots \quad (2.3)$$

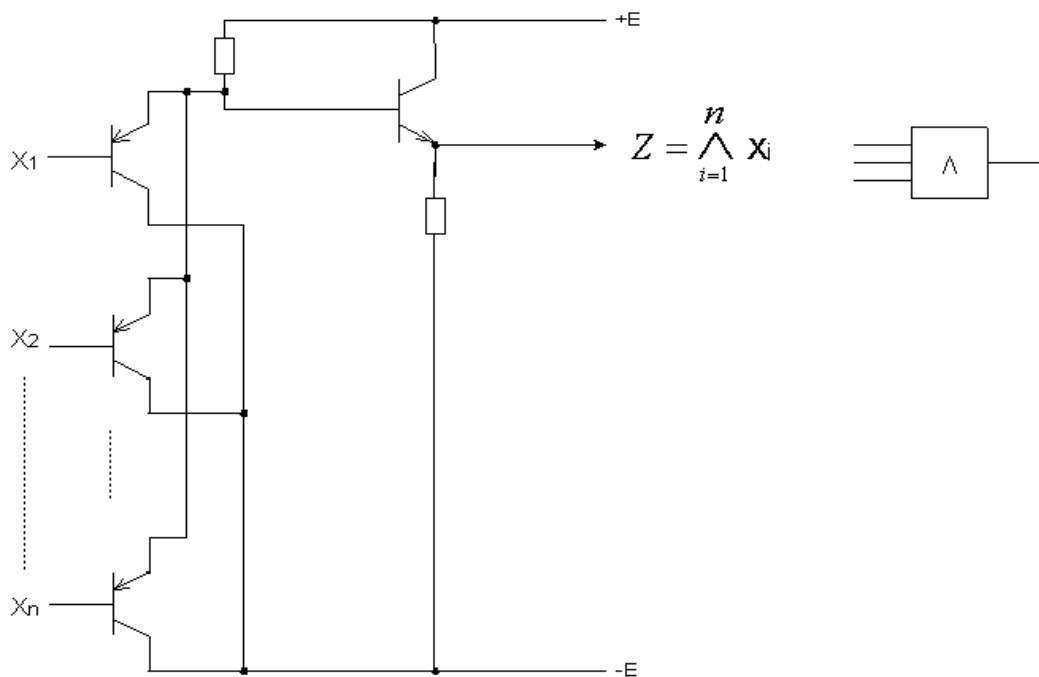


Рис. 2.20

Этому соответствуют следующие преобразования над функциями принадлежности:

$$\mu(Z) = \min \{ \mu_A(Z), \max[\mu_B(Z), \mu_C(Z)] \},$$

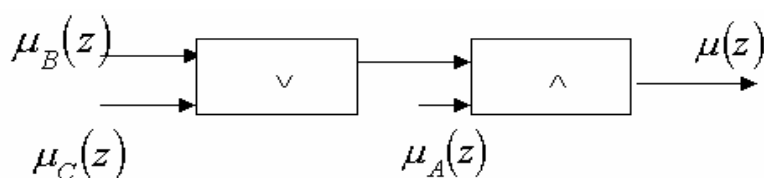


Рис. 2.21

Подобную совокупность можно рассматривать как процессор обработки условной части правила вывода. Очевидно, что их число должно быть равно общему количеству правил.

Для того чтобы получить законченный процессор обработки правил условного логического вывода, необходимо дополнить его блоком обработки импликации. Здесь то и возникают определенные сложности, свя-

занные с различной трактовкой этой операции. Например, для правила вида (2.3.).

Если $\langle P1=A \rangle$ и $\langle P2=B \vee C \rangle$, то $\langle R=D \rangle$,

$$\mu(z) = \min[\mu_D(Z), \min[\mu_A(Z), \max[\mu_B(z), \mu_C(z)]]]. \quad (2.4)$$

Возможна и другая интерпретация

$$\mu(z) = \min[1, (1 - \mu_1(Z) + \mu_D(z))], \quad (2.5)$$

где $\mu_1(Z) = \min\{\mu_A(Z), \max[\mu_B(z), \mu_C(z)]\}$.

Если реализация соотношения (2.4) требует введения еще одной схемы пересечения в рис. 2.21 (рис. 2.22), то импликация (2.5), получившая широкое распространение, требует более сложных элементов.

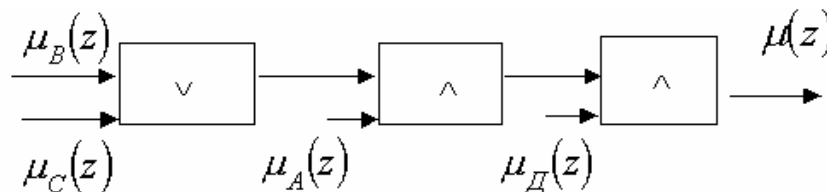


Рис. 2.22

Использование ограниченной разности, позволяющей преодолеть эти затруднения, предполагает наличие соответствующей электронной схемы. В работе [21] рассматривается так называемое токовое зеркало, которое может быть реализовано по биполярной или МОП-технологии. Предпочтительней является МОП-структура, которая свободна от ошибок биполярного токового зеркала и сама по себе является более технологичной, особенно при изготовлении заказных или полузаказных интегральных схем.

На рис 2.23 представлено токовое зеркало (см. рис. 2.23, а) и его условное обозначение (см. рис. 2.23, б). На рис. 2.24 представлена характеристика вход-выход токового зеркала. Комбинация токового зеркала и

диода, который тоже реализуется как МОП-структура, дает базовый элемент (рис. 2.25). На рисунке обозначены: I_0 – выходной ток; I_i – входной ток.

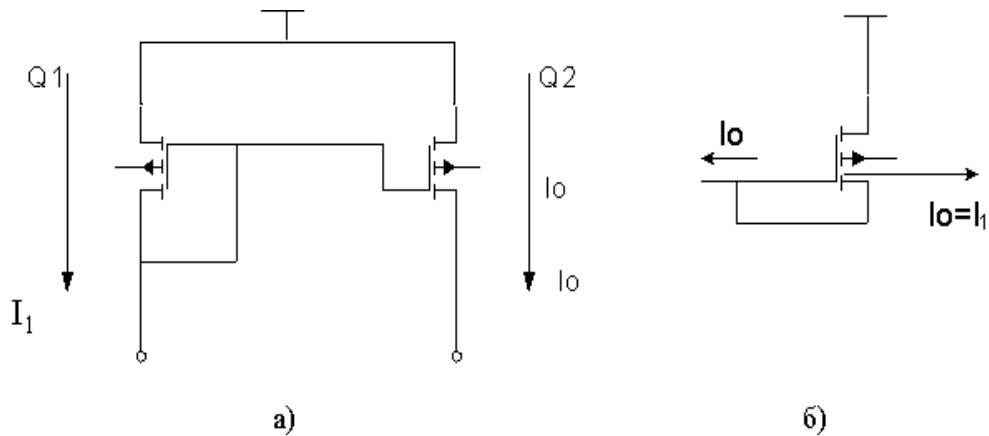


Рис. 2.23

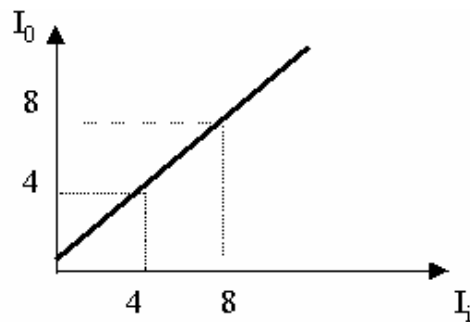


Рис. 2.24

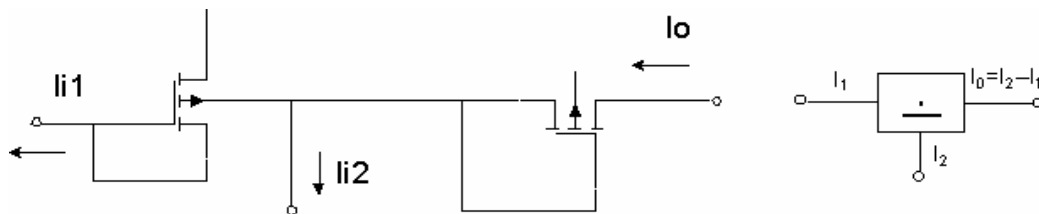


Рис. 2.25

Выходной ток базового элемента:

$$I_0 = \begin{cases} I_{i2} - I_{i1}, & \text{при } I_{i2} > I_{i1} \\ 0, & \text{при } I_{i2} < I_{i1} \end{cases} = 0 \vee (I_{i2} - I_{i1}).$$

Заменяя $I_{i1} \rightarrow \mu_B$ и $I_{i2} \rightarrow \mu_A$, получаем, что выходной ток $I_0 = \mu_A \dot{-} \mu_B$, т.е. базовый элемент работает как ограниченная разность.

Использование токовых сигналов позволяет очень просто выполнять операции суммирования и вычитания. Достаточно соответствующим образом в нужную точку подключить определенный источник тока. Реализация операции объединения (MAX) представлена на рис. 2.26, а операций эквивалентности – на рис. 2.27 (принципиальная схема) и рис. 2.28 (ее условное обозначение).

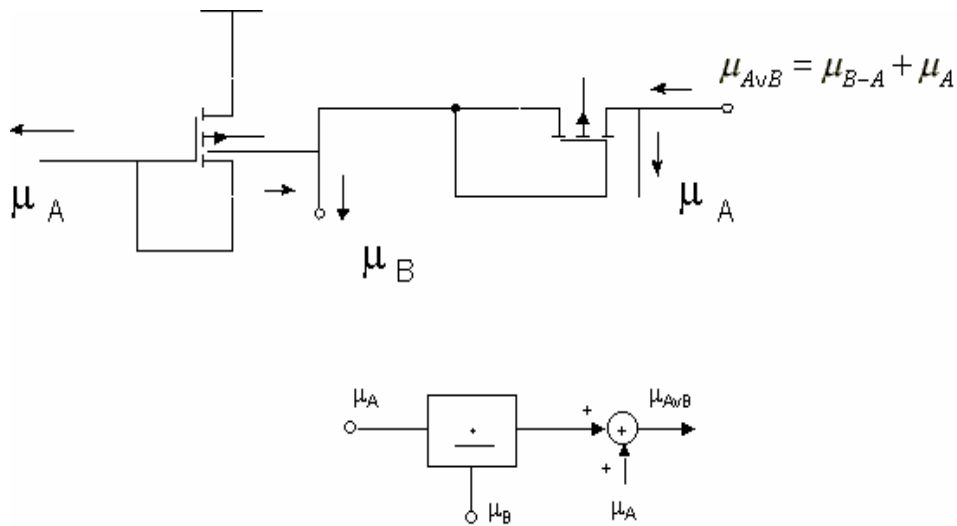


Рис. 2.26

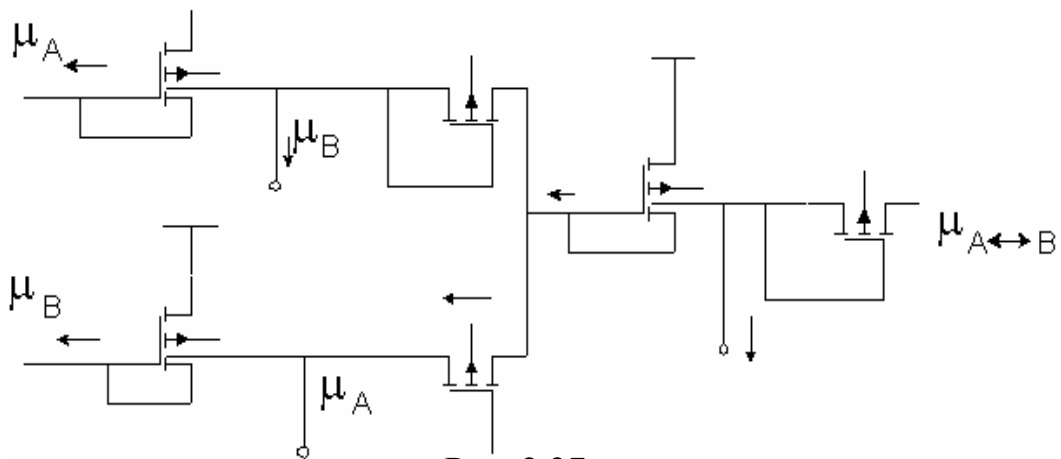


Рис. 2.27

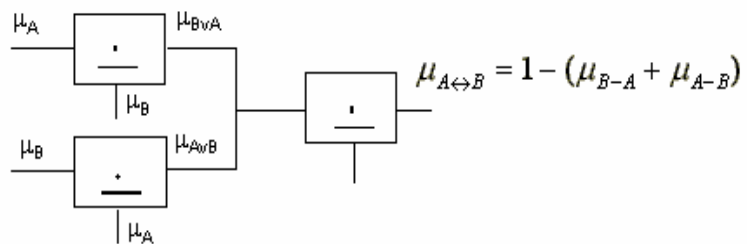


Рис. 2.28

Для обработки правил условного логического вывода надо выполнить каскадное соединение базовых логических элементов. Так, для соотношения (2.4) может быть построена следующая цепочка (рис 2.29).

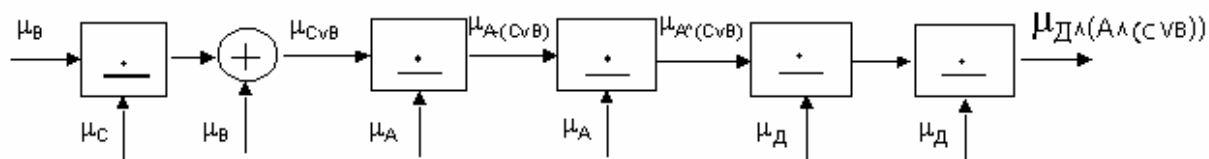


Рис. 2.29

Для соотношения 2.5 цепочка обработки будет та же самая, только в последнем элементе ограниченной разности надо произвести замену $\mu_D \rightarrow 1$.

Весьма важными достоинствами схемы являются её однородность и регулярность (одинаковые элементы соединены одинаковым образом).

Для схем основных нечетких логических операций, описанных выше, нужны один или два источника тока, которые обеспечат два входных, равных друг другу тока. При большом количестве переменных большое количество идентичных источников тока трудно реализуемо. Кроме этого в системах, использующих токовый принцип, каждая схема должна иметь некоторые выходные терминалы, число которых равно числу последующих управляемых схем. Для этой цели используется токовое зеркало с многотерминальным выходом (рис. 2.30).

Такая схема подключается либо к выходу соответствующего базового элемента, либо к источнику тока, представляющему конкретную переменную.

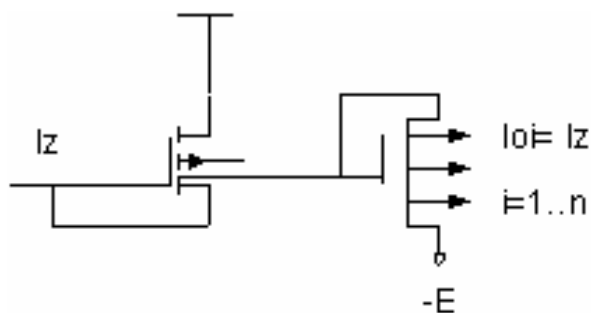


Рис. 2.30

Таким образом, используя рассмотренные элементы, можно построить необходимое число процессоров для обработки правил условного логического вида, создав процессорную матрицу. Использование эмиттеров

с нечеткими логическими вентилями или базовых МОП-элементов, реализующих ограниченную разность, предполагает преобразование кодов значений функций принадлежности в аналоговые сигналы (напряжение или ток). Это преобразование выполняется с помощью стандартных схем цифро-аналоговых преобразователей с выходом по напряжению или с токовым выходом. В последнем случае на выходе ЦАП следует включить многотерминальное токовое зеркало.

2.5. Организация процессоров обработки правил условного логического вывода

В предыдущем пункте было показано, что операция ограниченной разности и МОП-схема, ее реализующая, могут быть использованы для процессоров обработки правил условного вывода.

Наиболее простым решением является полностью аппаратная реализация с фиксированной структурой, когда каждому правилу соответствует отдельный процессор, работа которого инициируется соответствующими результатами фазификации.

Наиболее простыми процессоры получаются, если правила однородные и имеют одноуровневую структуру.

Например, стратегия 2 в нечетком контроллере подогрева воды [26] состоит из правил:

если $\langle x = \text{незначительно малое} \rangle$, то $\langle F1 = \text{очень малое} \rangle$, то $\langle F2 = \text{в устойчивом состоянии} \rangle$;

если $\langle x = \text{немалое} \rangle$, то $\langle \text{увеличение } F1 = \text{большое} \rangle$, то $\langle F2 = \text{в устойчивом состоянии} \rangle$ и т.п.,

где x – отклонение температуры воды от заданного значения; $F1$ – поток горячей воды; $F2$ – поток холодной воды.

Алгоритм управления давлением в контроллере паровой машины [26] состоит из сложных условных выражений, например:

$$\text{если } P_E = NB, \text{ то (если } C_{PE} = HE(NB \text{ или } NM), \text{ то } H_C = PB); \quad (2.6)$$

...

$$\text{если } P_E = PO, \text{ то (если } C_{PE} = PS, \text{ то } H_C = NS) \text{ и т.п.,} \quad (2.7)$$

где P_E – отклонение давления пара; C_{PE} – изменение отклонения; NB – большое отрицательное; NM – среднее отрицательное; NS – малое отрицательное; PO – выше нормы; PS – положительное среднее; PB – положительное большое.

Очевидно, что процессоры обработки правил для приведенных примеров будут заметно отличаться.

Для первого примера структура процессоров будет не только однородной, но и регулярной и легко реализуется матричной схемой (рис. 2.31).

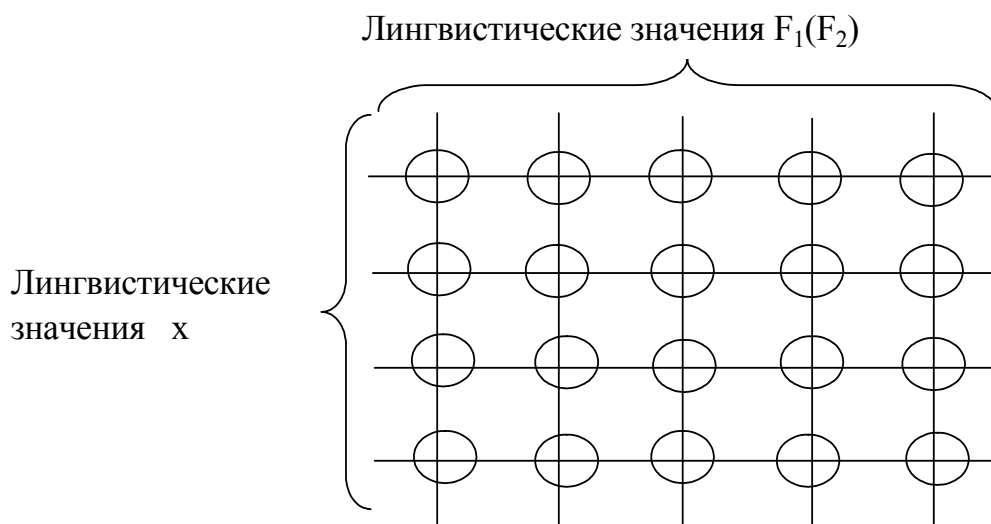


Рис. 2.31

На вертикальные и горизонтальные линии матрицы подаются лингвистические значения соответствующих переменных в виде значений функций принадлежности. В узлах матрицы находятся процессорные элементы, состав которых зависит от принятой процедуры обработки.

Если используется операция “пересечение”, то в узлах матрицы включается элемент, представленный рис. 2.32, а, если “импликация”, то рис. 2.32, б.

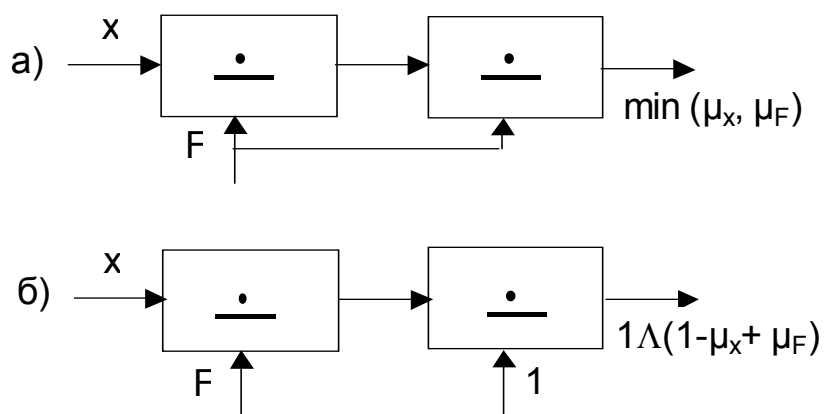


Рис. 2.32

Увеличение переменных в условной части правил с простой структурой можно обеспечить каскадной схемой включения таких матриц.

Например, правила вида:

если $\langle x=\dots \rangle$ и $\langle y=\dots \rangle$, то $\langle P=\dots \rangle$ при трех лингвистических значениях каждой из переменных будут обрабатываться схемой, приведенной на рис. 2.33. С целью упрощения рисунка соединения показаны только для трех элементов.

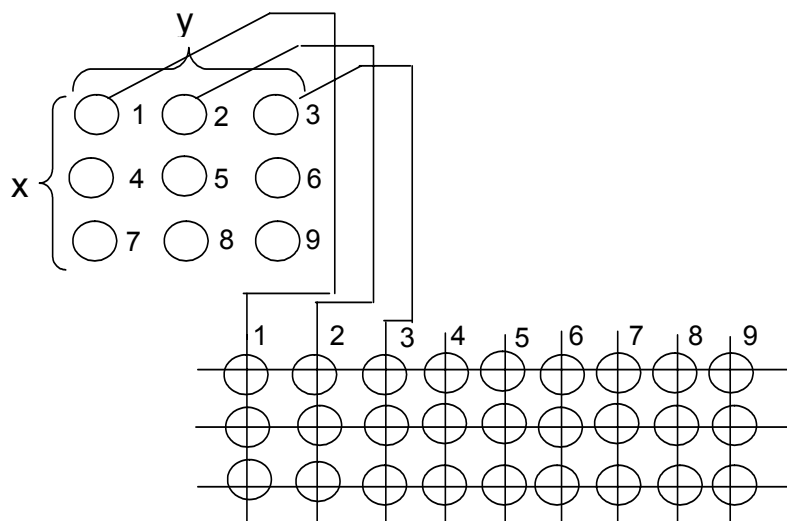


Рис. 2.33

К недостаткам рассмотренной структуры следует отнести быстрый рост необходимого числа соединений и процессорных элементов.

Нетрудно видеть, что в матрице последней степени максимальное число процессорных элементов равно

$$N = \prod_{i=1}^k N_i^2,$$

где N_i^2 – количество лингвистических значений i -й переменной;

k – число переменных в правиле.

Общее число процессорных элементов равно сумме процессоров, входящих в матрицы каждой степени.

Поскольку в практике при построении правил условного вывода не используются все возможные сочетания всех лингвистических значений переменных, то размерность матриц и количество процессорных элементов будут меньше.

Для неоднородных сложных условных выражений рассмотренные выше решения теряют свою целесообразность.

В этом случае каждое правило должно обрабатываться своим процессором, структура которого отражает структуру правила.

Например, для правила (2.6) при использовании операции \min результирующая функция принадлежности вычисляется по соотношению

$$\begin{aligned} \mu(z) &= \min \{ \min \{ [1 - \max(\mu_{NB}(z), \mu_{NM}(z))], \mu_{PB}(z) \}, \mu_{NB}(z) \} = \\ &= \mu_{NB} \dot{\div} [\mu_{NB} \dot{\div} \{ \mu_{PB} \dot{\div} \{ \mu_{PB} \dot{\div} [1 \dot{\div} ((\mu_{NB} \dot{\div} \mu_{NM}) + \mu_{NM})] \} \} \}]. \end{aligned}$$

Соответствующая структура процессора представлена на рис. 2.34.

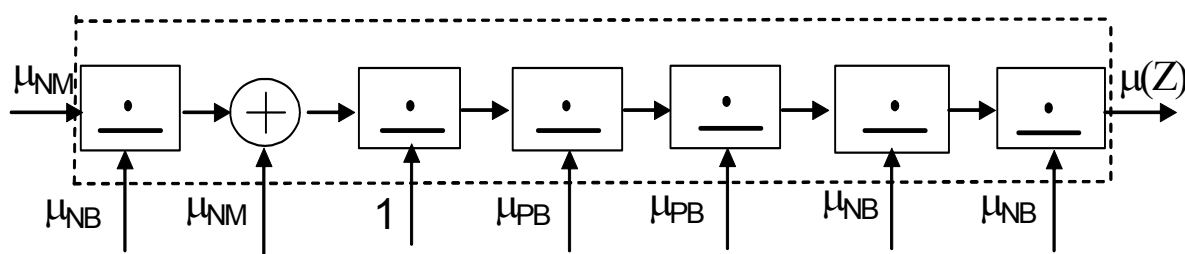


Рис. 2.34

Инициирование процессора осуществляется путем подачи соответствующих сигналов на его входы.

Выходной сигнал в схеме последующей обработки подается через вентиль, который открывается при наличии разрешающих сигналов (рис. 2.35).

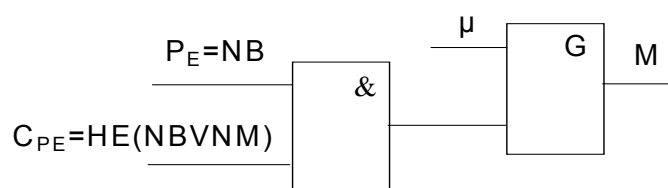


Рис. 2.35

Для правила (2.7)

$$\mu(z) = \min \{ \min [\mu_{PS}^{(2)}, \mu_{NS}^{(2)}], \mu_{PO} \} = \mu_{PO} \dot{\div} \{ \mu_{PO} \dot{\div} [\mu_{PS} \dot{\div} (\mu_{PS} \dot{\div} \mu_{NS})] \} .$$

Соответствующий процессор представлен на рис. 2.36.

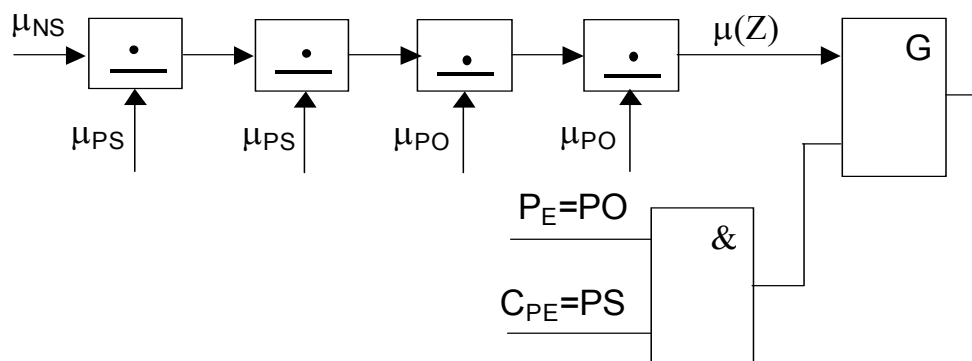


Рис. 2.36

Сопоставляя рисунки, нетрудно видеть, что свойство однородности сохраняется, но регулярность нарушается.

Количество процессоров обработки правил условного логического вывода, очевидно, равно количеству правил. Для отдельных видов процессов оно может быть порядка 100, обычное количество около 20.

Учитывая весьма простую схему реализации ограниченной разности и ее технологичность, можно полагать, что создание подобных контроллеров серьезных трудностей не представляет.

2.6. Формирование управляющего воздействия (дефазификация)

Наряду с общими моментами в подходах к созданию нечетких контроллеров существует и ряд различий в определении единственного управляющего воздействия из выходного нечеткого подмножества, представляемого функцией принадлежности, полученной после обработки правил условного логического вывода. Один из применяемых способов – выбор такого значения, для которого функция принадлежности имеет наибольшее значение (рис. 2.37, а), при наличии нескольких экстремумов на функции принадлежности (рис. 2.37, б) выбирается среднее значение для этих точек.

В последнем случае имеет место так называемый метод mean of maxima (среднего по максимуму).

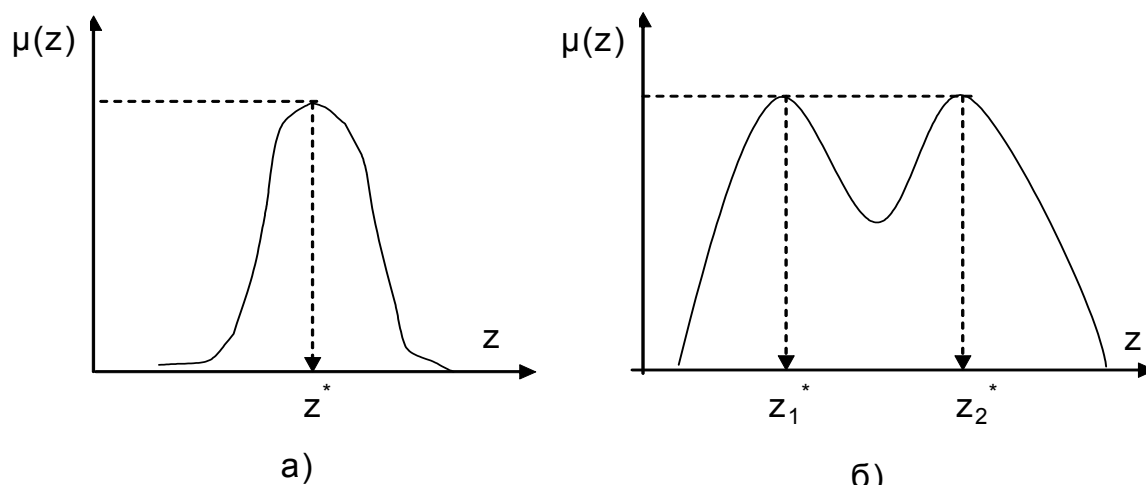


Рис. 2.37

Разновидностью метода среднего по максимуму является метод средневзвешенного по максимуму, так как могут возникнуть ситуации, когда экстремумы функции принадлежности выходного нечеткого подмножества не равны. Но это неравенство не столь значительно, чтобы можно было одним из экстремумов пренебречь (рис. 2.38).

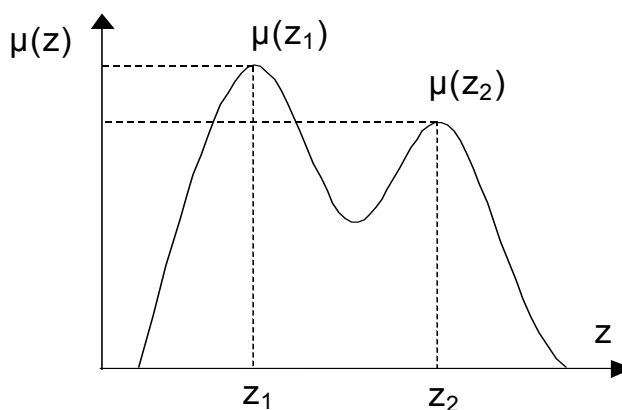


Рис. 2.38

В этом случае управляющее воздействие можно определить по соотношению

$$z^* = \frac{\mu(z_1)z_2 + \mu(z_2)z_1}{\mu(z_1) + \mu(z_2)}.$$

Исследования, проведенные по методам выбора управляющих воздействий, показали, что при использовании метода *mean of maxima* нечеткий алгоритм ведет себя как позиционное реле и для анализа применимы методы классической теории нелинейных САУ. В то же время при использовании метода центра площади нечеткий алгоритм идентичен ПИ-регулятору и считается, что этот метод предпочтительнее.

При выборе управляющего воздействия следует контролировать уровень функций принадлежности, т.е. необходимо установить некоторый порог решения, ниже которого выбор управляющего воздействия производить не следует (рис. 2.39, а – выбор управляющего воздействия можно выполнять; б – выбор управляющего воздействия производить не следует).

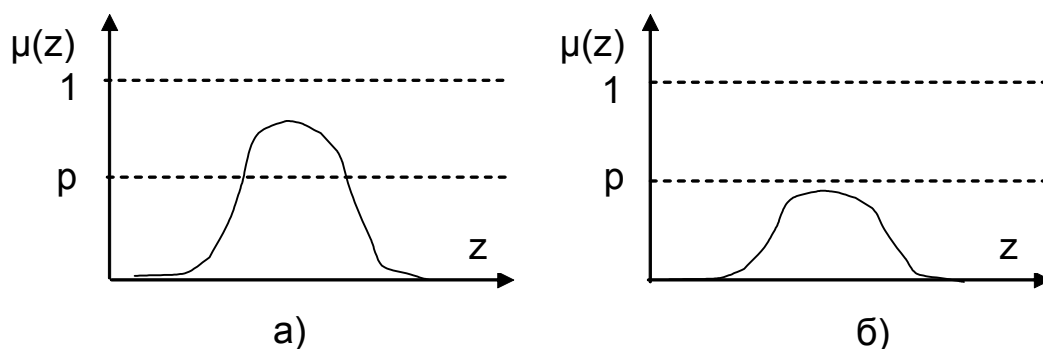


Рис. 2.39

В большинстве источников, посвященных нечеткой схемотехнике, подробно рассматриваются схемы обработки правил условного логического вывода и практически отсутствуют описания реализации такого важного узла, как дефазификатор.

Особенностью метода дефазификации является то, что собственно значение функции принадлежности в характерных точках не используется в управлении. Управляющее воздействие формируется по координатам этих точек независимо от того, какой метод используется.

Методы центра тяжести и центра площади требуют проведения вычислений. Например, центр тяжести вычисляется по формуле

$$z^* = \frac{\sum \mu(z_i) z_i}{\sum \mu(z_i)}. \quad (2.8)$$

Расчет по формуле (2.8) может выполняться с помощью специализированного вычислителя, который может быть реализован на базе существующих микропроцессорных наборов.

Наиболее просто реализуется метод mean of maxima, т.е. определение управляющего воздействия по максимуму функции принадлежности, определяющей выходное нечеткое подмножество.

Предположим, что значения функции принадлежности представлены двоичным кодом некоторой разрядности. На рис. 2.40 представлена схема, определяющая координату максимума функции принадлежности.

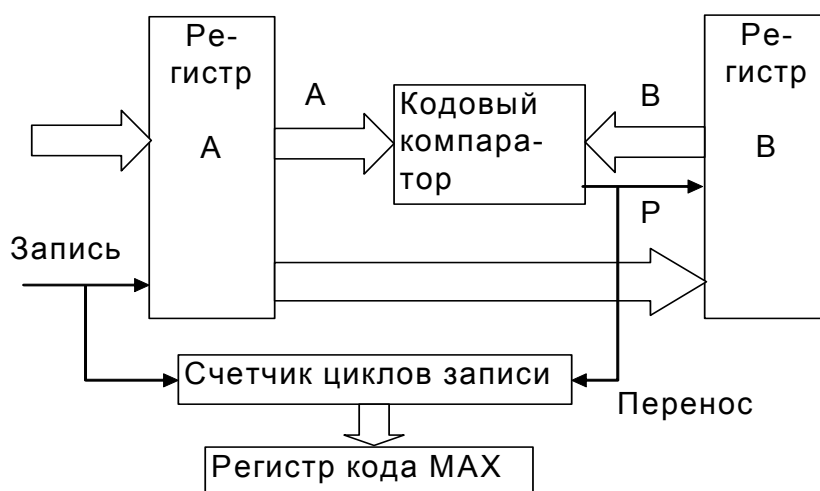


Рис. 2.40

В начальном состоянии регистр В устанавливается в ноль. В регистр А последовательно заносятся коды значений функции принадлежности. Импульсы “Запись” подсчитываются в счетчике циклов записи.

Содержимое регистров А и В сравниваются в кодовом компараторе, и если $A > B$, то формируется сигнал “Перенос”, по которому содержимое регистра А переносится в регистр В, а содержимое счетчика циклов записи – в регистр кода МАХ. Нетрудно видеть, что после ввода всех значений функции принадлежности в регистре В будет код ее максимума, в регистре кода МАХ – код координаты этого максимума, который с определенным коэффициентом пропорциональности может использоваться в качестве управляющего воздействия.

2.7. Составление правил нечеткого управления

Алгоритмы нечеткого управления, представляющие качественный процесс решения задач, описываются набором правил условного логического вывода.

Известны четыре способа получения этих правил:

- 1) на основе опыта и знаний экспертов;
- 2) на основе создания модели действия оператора;
- 3) посредством обучения;
- 4) с использованием нечеткой модели оборудования.

При использовании первого способа, как и при создании экспертных систем, вырабатывается словесное описание опыта квалифицированного специалиста в данной проблемной области, который затем формализуются в виде набора правил «если... то».

Так же, как и при разработке экспертных систем, необходим инженер по знаниям, т.е. специалист, способный квалифицированно работать с экспертом. Этот способ был реализован при управлении доменной печью [31].

Второй способ используется тогда, когда эксперт не может описать свои действия, а запоминает манипуляции механически. В этом случае разработчик нечеткого регулятора (инженер по знаниям) делает попытку описать действия оператора в виде правил условного вывода. Этот метод может потребовать несколько итераций, для того чтобы получить приемлемый результат.

Способ обучения целесообразно применять при экспериментах на реальном оборудовании, или когда существует приближенная его модель. Правила нечеткого управления формируются путем обучения, начиная с ситуации, когда вообще нет ни одного правила. Обучение можно вести практически непрерывно, улучшая управление в соответствии с изменением условий и не привлекая при этом экспертов. Этот метод удобен при разработке систем управления интеллектуальными роботами. Примером применения этого метода может служить нечеткий контроллер для управления скоростью автомобиля [13].

Четвертый способ используется тогда, когда модель оборудования может быть создана только в нечеткой форме в виде набора правил «если..., то...». Правила выводятся теоретически, исходя из целей управления.

Контрольные вопросы

1. Перечислите пять принципов организации интеллектуальной управляющей структуры.
2. Дайте определение нечеткого регулятора.
3. Укажите основные направления реализации нечетких процессоров.
4. Дайте определение ограниченной разности нечетких множеств.
5. В чем заключаются преимущества использования ограниченной разности в процессорах обработки правил условного вывода?

Глава 3

ТЕХНИЧЕСКИЕ И ПРОГРАММНЫЕ СРЕДСТВА ДЛЯ РЕАЛИЗАЦИИ НЕЧЕТКОГО УПРАВЛЕНИЯ

3.1. Контроллеры для реализации алгоритмов нечеткого управления

В настоящее время разработкой и производством технических и программных средств активно занимаются различные фирмы Японии, США, Западной Европы.

Среди японских фирм в этой области наиболее активно работает фирма «Omron», которая имеет свои отделения в ряде европейских стран.

Номенклатура технических и программных средств этой фирмы, обеспечивающих нечеткое управление, состоит из следующих средств [22]:

- фази-(нечетких) процессоров;
- интеллектуальных модулей контроллера, реализующих управление с нечеткой логикой;
- специализированного температурного фази-контроллера;
- инструментальных технических средств и программного обеспечения для IBM совместных ПЭВМ, поддерживающих нечеткое управление.

Реализация нечеткого управления обеспечивается двумя типами цифровых процессоров FP1000, FP3000, которые различаются количеством разрядов и контактов в корпусе: FP1000 – 8 разрядов, 44 – вывода, FP3000 – 12 разрядов, 64 вывода. Остальные характеристики процессоров одинаковы.

Основные характеристики нечетких процессоров семейства FP

Тактовая частота, МГц	24
Число обрабатываемых правил в режимах:	
типовом	20
расширенном	128
Тип функций принадлежности	Треугольная, трапецеидальная
Максимальное число идентификаторов	7
Формат правила	5 условий и 2 заключения в правиле
Время обработки, мкс	125
Алгоритм обработки	min «и»/max «или»
Алгоритм преобразования выходных переменных	Метод «центра тяжести»

Специальный температурный фази-контроллер ESAF реализует гибридное управление на базе ПИД и нечеткого регулятора. Последний включается во входную цепь обратной связи, дает сверхдинамичную реакцию на изменение задания. Точность отслеживания задания составляет $\pm 0,3$ %. Контроллер обладает повышенной помехоустойчивостью (на 50 % выше, чем обычный ПИД-регулятор.)

Создание и перенос базы знаний поддерживается специальным инструментальным программным обеспечением – Fuzzy Support Software (FSS), реализованным на ПЭВМ с дополнительным процессором.

Программное обеспечение FSS реализуется в форме меню и обеспечивает:

- создание объектного программного обеспечения (ОПО);
- запись ОПО в файл и его распечатку;
- загрузку ОПО в интеллектуальные фази-модули и его выгрузку;
- управление процессом создания заключений для правил;
- протоколирование данных в формате Lotus 1, 2, 3 или Excel;
- генерирование функций помощи при создании ОПО.

На базе процессоров FP1000, FP3000 созданы интеллектуальные фази-модули C200H-F2001 и C500-F2001.

Подключение внешних устройств производится через полудуплексный канал RS-232C со старт-стопной синхронизацией. Подключение оптоволоконного кабеля осуществляется через модуль связи Link Unit.

Для сохранения правил и функций принадлежности в оперативной памяти при отключении питания используется резервная батарея со сроком действия 5 лет, со временем замены 5 мин.

Фирма «Echelon» разработала, а фирма «Motorola» производит 8-битовый микроконтроллер MC143150/20. Максимальное количество входов контроллера равно четырем и может программироваться пользователем. На каждый вход приходится восемь функций принадлежности, которые имеют трапецеидальный вид и параметры которых определяются пользователем.

Контроллер рассчитан на полосу пропускания 30 Гц, что является результатом программной реализации основных процедур, но для многих применений полоса пропускания 30 Гц является вполне достаточной.

Время выполнения (включая время, затраченное на операции фазификации, обработки правил и дефазификации) лежит в пределах от 19 до 24 мс и зависит от количества правил, функций принадлежности и синглтонов, т.е. возможных выходных значений.

Для нечеткого контроллера MC143150 разработаны средства для параллельного обмена с микропроцессором MC684C11.

Для организации аналоговых выходов используется шестиканальный цифро-аналоговый преобразователь MC144110, шесть аналоговых входов образуются с помощью 8-битового аналого-цифрового преобразователя MC14443. Разработка программного обеспечения нечеткого контроллера выполняется на ПЭВМ в среде Visual Basic, связь с ПЭВМ осуществляется через интерфейсную плату MM3221EVK. Для расширения пространства выходов используется программируемый интерфейсный расширитель PSD3XX.

Нечеткий контроллер MC143150 может использоваться в системах управления теплообменом, контроля уровня жидкостей, управления электрическими двигателями.

3.2. Программные средства нечеткого управления

Средства нечеткого управления в пакете Matlab

Пакет *Matlab* компании «MathWorks» (США) содержит широкий набор готовых функций, используемых при анализе и синтезе систем автоматического управления.

В пакете имеется набор алгоритмов, образующих так называемый инструментарий, которые могут использоваться для проектирования, анализа и моделирования САУ. Кроме этого в пакете *Matlab* имеется набор блоков *Simulink*, позволяющий в графической форме выполнять моделирование достаточно сложных систем, в том числе и с использованием блоков, реализующих нечеткое управление.

Для решения задач методами теории нечетких множеств в пакете *Mathlab* предусмотрен пакет нечеткой логики *Fuzzy Logic Toolbox* [23].

Основные возможности пакета:

- построение систем нечеткого вывода (экспертных систем, регуляторов, аппроксиматоров зависимостей);
- построение адаптивных нечетких систем (гибридных нейронных сетей);
- интерактивное динамическое моделирование в среде *Simulink*.

Пакет обеспечивает работу:

- в режиме графического интерфейса;
- в режиме командной строки;
- с использованием блоков и примеров пакета *Simulink*.

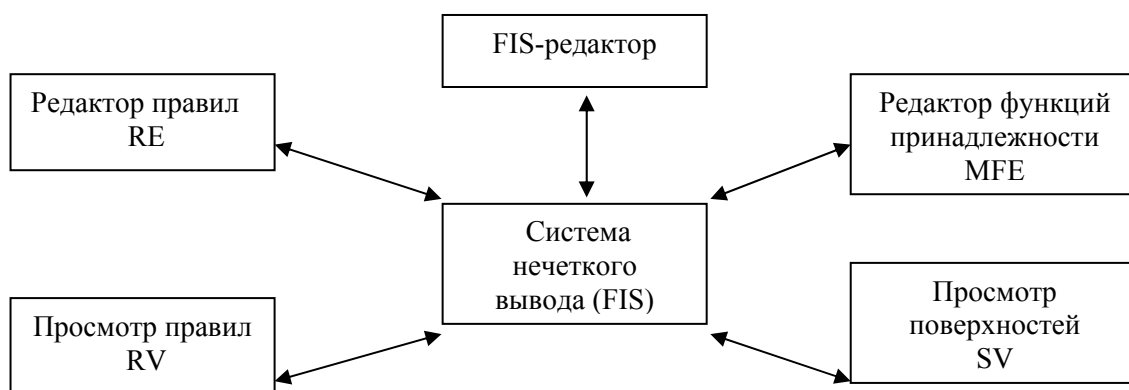


Рис. 3.1

Моделирование нечеткого управления выполняется с помощью системы нечеткого вывода FIS (Fuzzy Inference System) (рис. 3.1), включающей редактор системы нечеткого вывода (FIS-Editor), редактор функций принадлежности (The Member Ship Function Editor), редактор правил (The Rule Editor), подсистемы для просмотра правил, схем нечетких выводов (The Rule Viewer) и полученных поверхностей (The Surface Viewer).

FIS-редактор обеспечивает высокий уровень общения с системой, не имеет ограничений на число входных и выходных переменных, которое ограничивается лишь доступным объемом памяти применяемой ЭВМ.

Редактор функций принадлежности используется для задания вида функций принадлежности для каждой переменной.

Редактор правил применяется для редактирования текста правил условного логического вывода при описании поведения моделируемой системы.

Просмотрщики правил и поверхностей используются для визуального контроля. Просмотрщик правил отображает схему нечеткого вывода на последнем этапе и используется как средство диагностики. С его помощью можно, например, увидеть, какие правила активны, или оценить влияние формы отдельной функции принадлежности на результат.

Просмотрщик поверхностей используется для представления на экране зависимости одного выхода от одного или двух входов, а также генерации и построения картины поверхности выхода для системы.

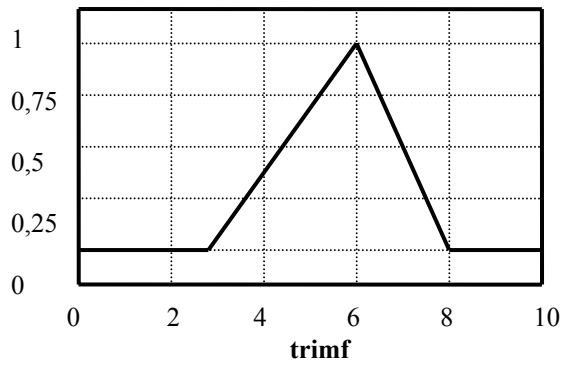
Все компоненты FIS могут взаимодействовать и обмениваться данными в процессе моделирования.

В пакете *Mathlab* возможно использование шести видов функции принадлежности (рис.3.2):

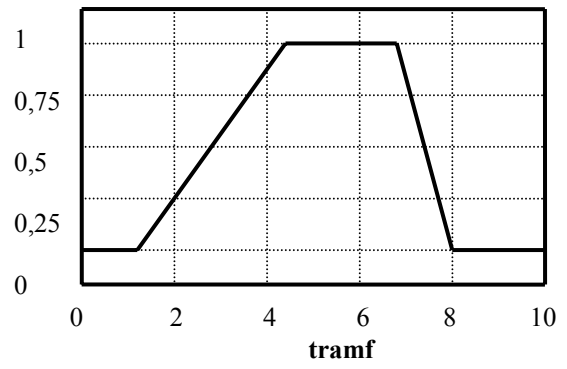
- треугольной (*trimf*) (см. рис. 3.2, а);
- трапецеидальной (*trmf*) (см. рис. 3.2, б);
- функции принадлежности в виде кривой Гаусса (*gaussmf*) или составленной из двух кривых Гаусса (*gauss2mf*) (см. рис. 3.2, в,г);
- колоколообразной (*bellmf*) (см. рис. 3.2, д);
- сигма-функций, предназначенных для воспроизведения несимметричных функций принадлежности, *sigmf*-функция принадлежности (см. рис. 3.2, е), открытая справа, *dsigmf* – закрытая функция принадлежности (см. рис. 3.2, ж), составленная из разности двух сигма-функций; *psigmf* – закрытая функция принадлежности, образованная из произведения двух сигма-функций (см. рис. 3.2, з);
- три функции принадлежности, основанные на полиномиальных кривых: *zmf* – несимметричная функция принадлежности, открытая слева; *smf* – несимметричная функция принадлежности, открытая справа; *pmf* – закрытая функция принадлежности.

Кроме этого в пакете *Mathlab* для пользователя имеется возможность конструировать собственные функции принадлежности.

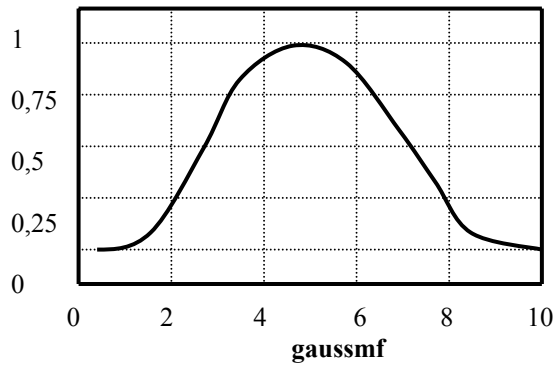
Система нечеткого моделирования поддерживает два основных оператора «И» и «ИЛИ». Импликация реализуется через оператор «И».



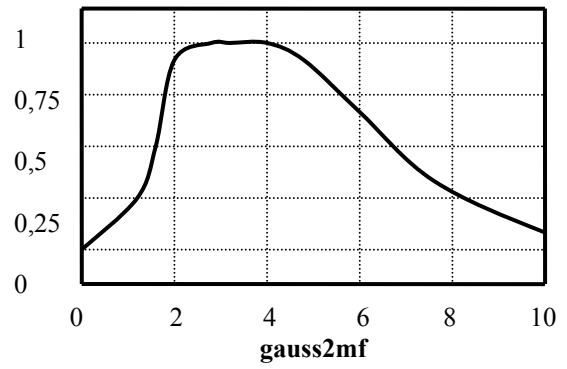
a)



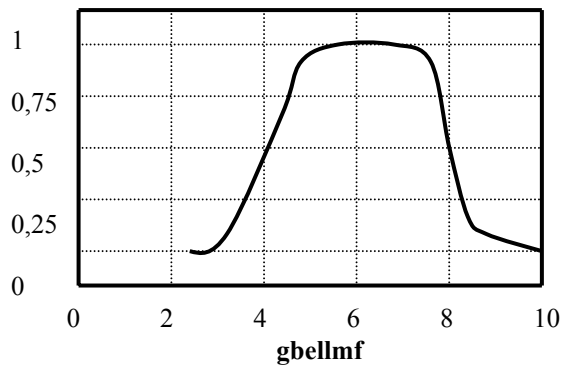
б)



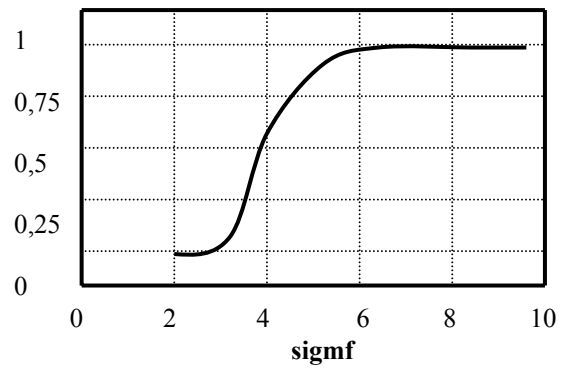
в)



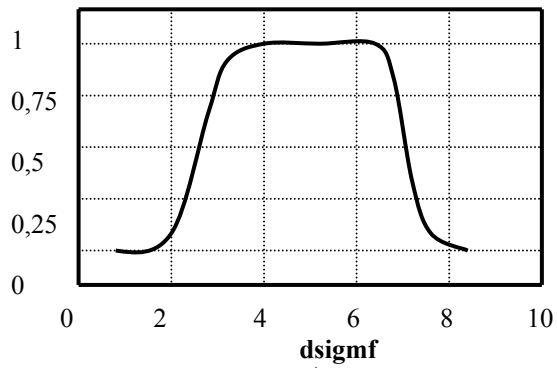
г)



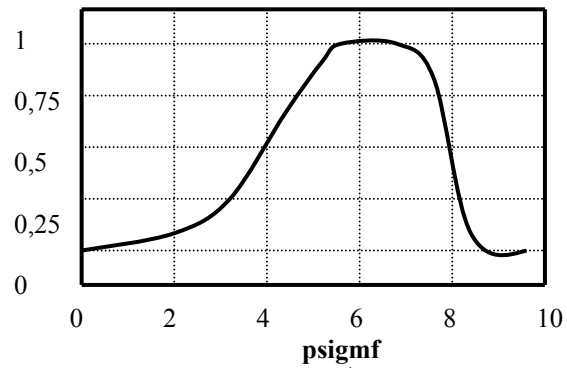
д)



е)



ж)



з)

Рис. 3.2

Оператор «И» представлен в двух видах: \min и произведение (prod), «ИЛИ» – \max и probor -оператор вероятного «ИЛИ», известный еще как алгебраическая сумма и вычисляемый по уравнению:

$$\text{probor}(a, b) = a + b - ab.$$

Кроме этих операций в пакете нечеткой логики *Fuzzy Logic Toolbox* представлены операции концентрирования и растяжения.

Пакет нечеткой логики поддерживает также все известные операции над нечеткими отношениями.

Для реализации нечетких выводов используются алгоритмы Мамдани (Mamdani) и Сугэно (Sugeno).

Для упрощения рассмотрения предположим, что базу знаний организуют два нечетких правила вывода:

П1: если X есть A_1 и Y есть B_1 , тогда Z есть C_1 ;

П2: если X есть A_2 и Y есть B_2 , тогда Z есть C_2 ,

где X и Y – имена входных переменных;

Z – имя переменной вывода;

$A_1, B_1, C_1, A_2, B_2, C_2$ – некоторые лингвистические значения соответствующих переменных, для которых определены функции принадлежности $\mu_{A_1}(x), \mu_{B_1}(y), \mu_{C_1}(z), \mu_{A_2}(x), \mu_{B_2}(y), \mu_{C_2}(z)$.

Алгоритм Мамдани. Данный алгоритм состоит из нескольких этапов:

1. Для значений X_0 и Y_0 определяется степень истинности для предпосылок каждого правила: $\mu_{A_1}(x_0), \mu_{B_1}(y_0), \mu_{A_2}(x_0), \mu_{B_2}(y_0)$.

2. Находятся уровни "отсечения" для предпосылок каждого из правил (используется операция \min):

$$\alpha_1 = A_1(x_0) \wedge B_1(y_0);$$

$$\alpha_2 = A_2(x_0) \wedge B_2(y_0).$$

Затем находятся "усеченные" функции принадлежности

$$c'_1(z) = \alpha_1 \wedge c_1(z);$$

$$c'_2(z) = \alpha_2 \wedge c_2(z).$$

3. Производится объединение усеченных функций с использованием операции \max , что дает итоговое нечеткое подмножество для переменной выхода с функцией принадлежности

$$\mu_{\Sigma}(z) = \max \{ \min [\mu_{A_1}(x_0), \mu_{B_1}(y_0)], \min [\mu_{A_2}(x_0), \mu_{B_2}(y_0)] \}.$$

Алгоритм Сугэно

Набор правил представляется в следующем виде:

П1: если X есть A1 и Y есть B1, тогда $z_1=a_1x+b_1y$;

П2: если X есть A2 и Y есть B2, тогда $z_2=a_2x+b_2y$.

Алгоритм состоит из нескольких этапов:

1. Первый этап аналогичен первому этапу алгоритма Мамдани.
2. На втором этапе находятся

$$\alpha_1 = \mu_{A1}(x_0) \wedge \mu_{B1}(y_0);$$

$$\alpha_2 = \mu_{A2}(x_0) \wedge \mu_{B2}(y_0)$$

и индивидуальные выходы правил:

$$z_1^* = a_1x_0 + b_1y_0 ;$$

$$z_2^* = a_2x_0 + b_2y_0 .$$

3. Определяется четкое значение переменной вывода (рис. 3.3):

$$z_0 = \frac{\alpha_1 z_1^* + \alpha_2 z_2^*}{\alpha_1 + \alpha_2} .$$

Рассмотренный алгоритм относится к алгоритму Сугэно 1-го порядка.

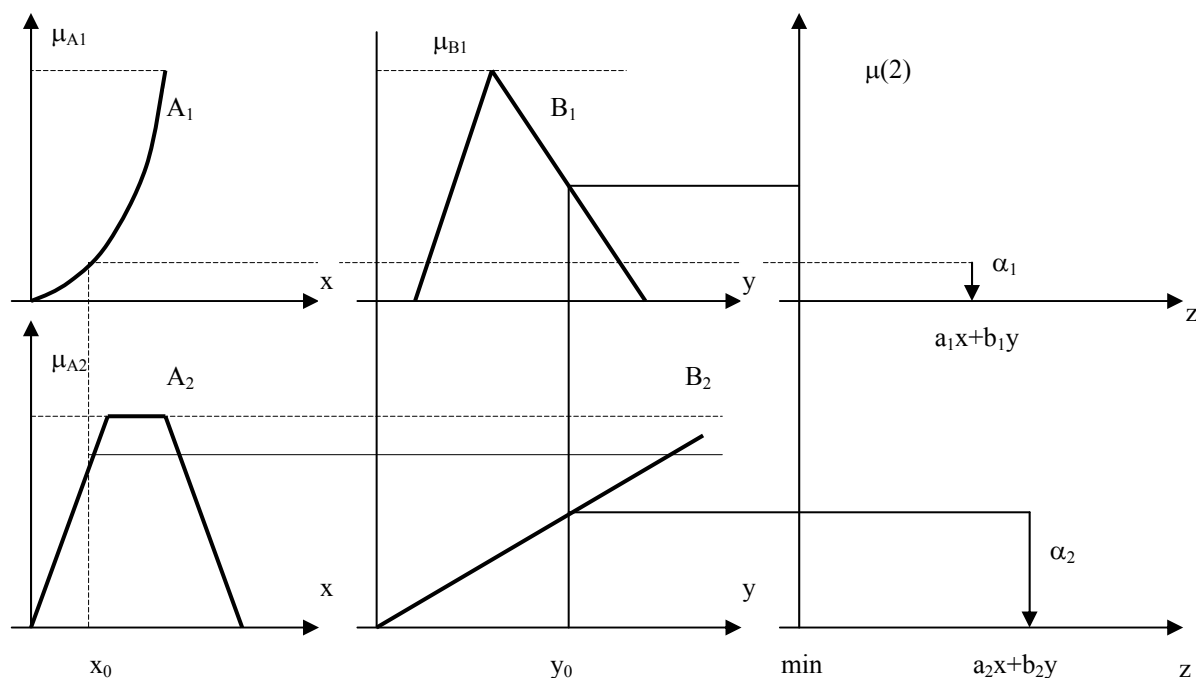


Рис. 3.3

Если $z_1=c_1$ и $z_2=c_2$, то получим алгоритм Сугэно нулевого порядка:

$$\alpha_1 = \mu_{A1}(x_0) \wedge \mu_{B1}(y_0);$$

$$\alpha_2 = \mu_{A2}(x_0) \wedge \mu_{B2}(y_0),$$

и находятся усеченные функции принадлежности

$$\mu'_{C1}(z) = \alpha_1 \wedge \mu_{C1}(z);$$

$$\mu'_{C2}(z) = \alpha_2 \wedge \mu_{C2}(z).$$

4. С использованием операции \max производится объединение найденных усеченных функций, что приводит к получению регулирующего нечеткого подмножества для переменного выхода с функцией принадлежности (рис. 3.4):

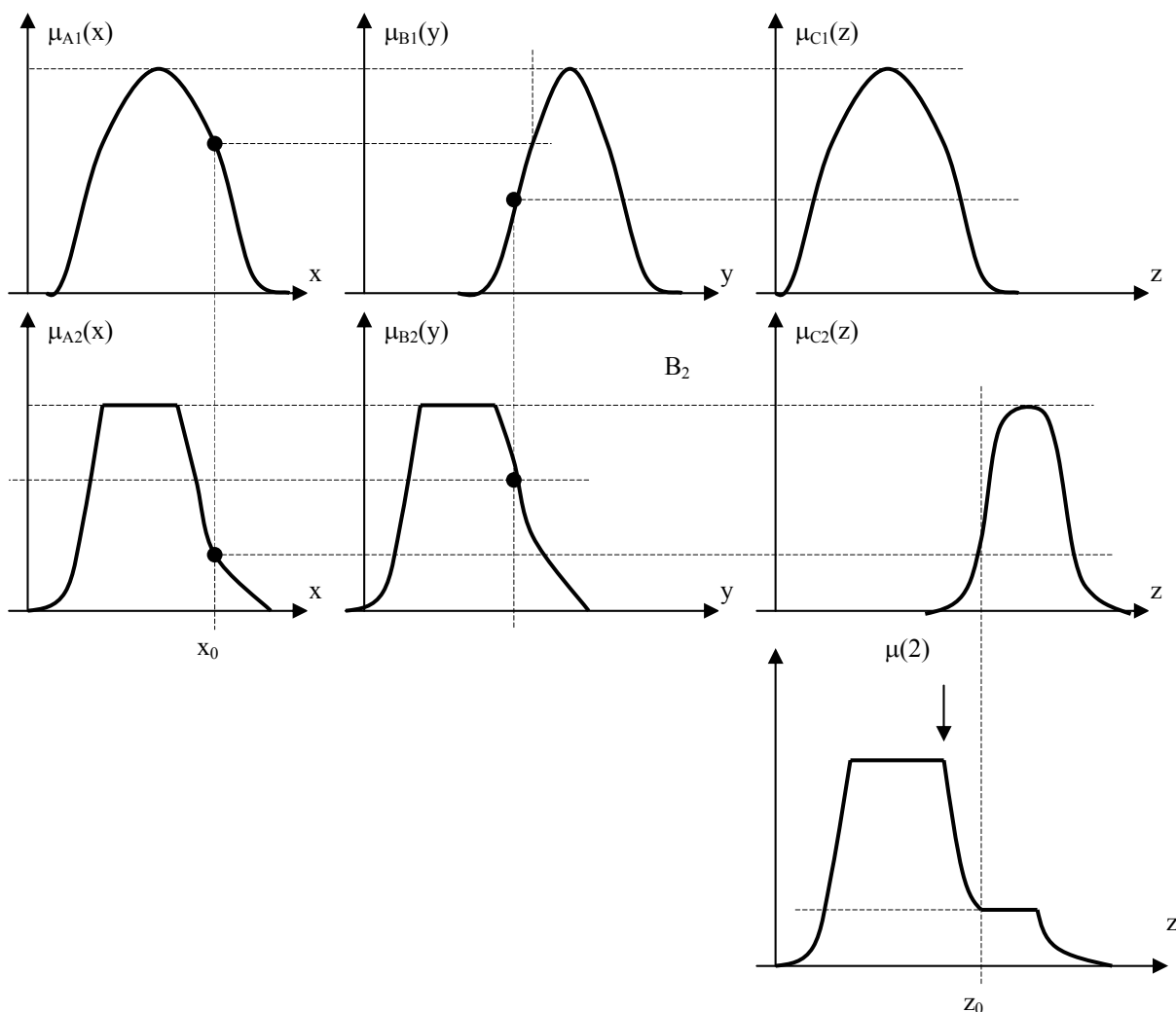


Рис. 3.4

$$\mu(z) = \mu'_{C1}(z) \cup \mu'_{C2}(z).$$

5. Дефазификация производится, например, вычислением центра тяжести кривой $\mu(z)$

$$z_0 = \frac{\int z \mu(z) dz}{\int \mu(z) dz}.$$

Нечеткое управление в системе Трейс Моуд

Пакет Трейс Моуд (Trace Mode) фирмы «АдАстра» предназначен для разработки автоматизированных систем управления технологическими процессами помимо средств разработки проектов систем, использующих традиционные алгоритмы управления, предоставляет возможности для реализации нечеткого управления.

Нечеткий регулятор FZCTR в системе Трейс Моуд реализован в качестве функционального блока библиотеки регулирования (рис. 3.5) [33]. Регулятор имеет три функциональных входа и два выхода. На вход INP надо подавать регулируемое значение; второй вход предназначен для ввода значения уставки.

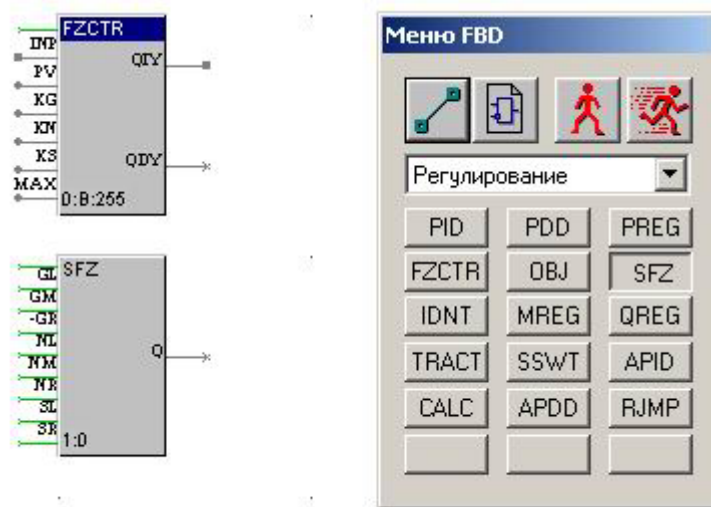


Рис. 3.5

Для выбора типа объекта управления используется вход IC, который может устанавливаться в состояние:

- "0", когда характеристики системы регулирования настраиваются с помощью функциональных блоков ГД, НД, SD;

- "1", выполняется регулирование малоинерционных параметров, для которых значения постоянных времени лежат в диапазоне от 2 до 5 с;

- "2", выполняется регулирование параметров средней инерционности с постоянными времени в пределах от 5 до 60 с;

- "3", для регулирования сильноинерционных параметров, значения постоянных времени которых могут быть в пределах от 1 до 30 мин.

На выходе Q формируется величина управляющего воздействия, выход dQ предназначен для формирования величины приращения управляющего воздействия на текущем шаге управления.

Алгоритм формирования выходных сигналов Q и dQ использует соотношения

$$Q = Q_{t-1} + dQ,$$
$$dQ = k_g dQ_g + k_n dQ_n + k_s dQ_s,$$

где Q_t – управляющее воздействие на момент t ;

Q_{t-1} – управляющее воздействие на предыдущем шаге $t-1$;

dQ – приращение управляющего воздействия;

k_g – принадлежность текущего рассогласования к категории "большее";

k_n – принадлежность текущего рассогласования к категории "среднее";

k_s – принадлежность текущего рассогласования к категории "малое";

dQ_g – приращение управляющего воздействия по условию "большое отклонение";

dQ_n – приращение управляющего воздействия по условию "среднее отклонение";

dQ_s – приращение управляющего воздействия по условию "малое отклонение".

Приращение по каждой из категорий отклонения рассчитывается по формуле

$$dQ_j = (k_{1j} (Pv - INP) + k_{2j} \text{SIGN} (Pv - INP) + k_{3j} dQ_{t-1} \text{SIGN} (Pv - INP))/5,$$

где Pv – значение уставки;

INP – значение регулируемой величины;

dQ_{t-1} – приращение управляющего воздействия на предыдущем такте пересчета;

j – признак категории рассогласования;

g – сильное рассогласование;

n – среднее рассогласование;

s – слабое рассогласование;

k_1 – коэффициент при рассогласовании;

k_2 – коэффициент нечеткой составляющей;

k_3 – коэффициент учета предыдущего изменения управления.

Настройка коэффициентов k_1 , k_2 , k_3 , а также границ диапазонов категорий рассогласования задается с помощью соответствующих блоков ГД, НД и СД. Эти блоки передают заданные в них настройки всем присутствующим в данном узле нечетким логическим регуляторам. Поэтому перед каждым из регуляторов, имеющих индивидуальные настройки, следует предусмотреть набор этих блоков, которые при этом надо также разместить в программе, чтобы они выполнялись перед соответствующим блоком нечеткого логического регулятора.

Блок настройки «FZCTR – сильное отклонение» (ГД) предназначен для настройки диапазонов, которые воспринимаются как сильное отклонение регулируемой величины от уставки, и задания коэффициентов для расчета управляющего воздействия в этом случае.

Блок настройки «FZCTR – среднее отклонение» (НД) предназначен для настройки диапазонов, которые воспринимаются как среднее отклонение регулируемой величины от уставки, и задания коэффициентов для расчета управляющего воздействия в этом случае.

Блок настройки «FZCTR – слабое отклонение» (СД) предназначен для настройки диапазонов, которые воспринимаются как слабые отклонения регулируемой величины от уставки, и задания коэффициентов для расчета управляющего воздействия в этом случае.

Fuzzy-библиотека пакета Con Sept

Пакет *Con Sept*, предназначенный для синтеза контроллеров Modicon TSX Quantum, содержит fuzzy-библиотеку, позволяющую вести также синтез и нечетких регуляторов. Она содержит набор функциональных модулей, который можно разделить на несколько групп:

функциональные модули для фазификации:

FUZ_ATERN – всех термов лингвистических переменных;

FUZ_STERM – отдельного термина лингвистической переменной по четырем точкам;

FUZ_ATERN_STI – данных целого типа;

FUZ_ATERN_STR – данных реального типа.

функциональные модули для дефазификации:

DEFUZ_INT – для получения результата дефазификации лингвистической переменной в целочисленном формате;

DEFUZ_REAL – для получения результата дефазификации лингвистической переменной в вещественном формате.

функциональные модули для обработки правил нечеткого вывода:

FUZ_MIN – для выполнения операции пересечения (\min) над функциями принадлежности;

FUZ_MAX – для выполнения операции объединения (\max) над функциями принадлежности;

FUZ_PROD – для формирования нечеткого произведения функций принадлежности;

FUZ_SUM – для вычисления ограниченной суммы функций принадлежности.

Fuzzy-библиотека пакета Con Sept поддерживает только упрощенный вариант функций принадлежности, при этом возможно до девяти лингвистических значений (рис. 3.6) Дефазификация выполняется по правилу максимума, а конечный результат вычисляется как среднее по максимуму.

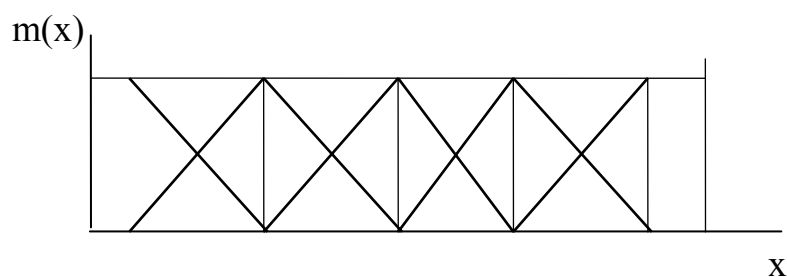


Рис. 3.6

Для проектирования нечетких регуляторов в пакете Con Sept используется язык функциональных схем.

Рассмотрим пример построения такой схемы на примере следующих правил.

Объект имеет две входные переменные X , Y и одну выходную Z .

Для переменной X определены лингвистические значения $\{L_{x_1}, L_{x_2}, L_{x_3}\}$ и соответствующие функции принадлежности.

Для переменной Y заданы лингвистические значения $\{L_{y_1}, L_{y_2}, L_{y_3}, L_{y_4}\}$ и соответствующие функции принадлежности. Выходная переменная Z имеет набор лингвистических значений $\{L_{z_1}, L_{z_2}, L_{z_3}, L_{z_4}\}$.

Определены следующие правила:

P1: Если $\langle X = L_{x_2} \rangle$ и $\langle Y = L_{y_3} \rangle$, то $\langle Z = L_{z_1} \rangle$;

P2: Если $\langle X = L_{x_3} \rangle$ или $\langle Y = L_{y_3} \rangle$, то $\langle Z = L_{z_1} \rangle$;

P3: Если $\langle X = L_{x_2} \rangle$ и $\langle Y = L_{y_2} \rangle$, то $\langle Z = L_{z_3} \rangle$;

P4: Если $\langle X = L_{x_1} \rangle$ или $\langle Y = L_{y_1} \rangle$, то $\langle Z = L_{z_4} \rangle$.

Соответствующая этим правилам схема приведена на рис. 3.7.

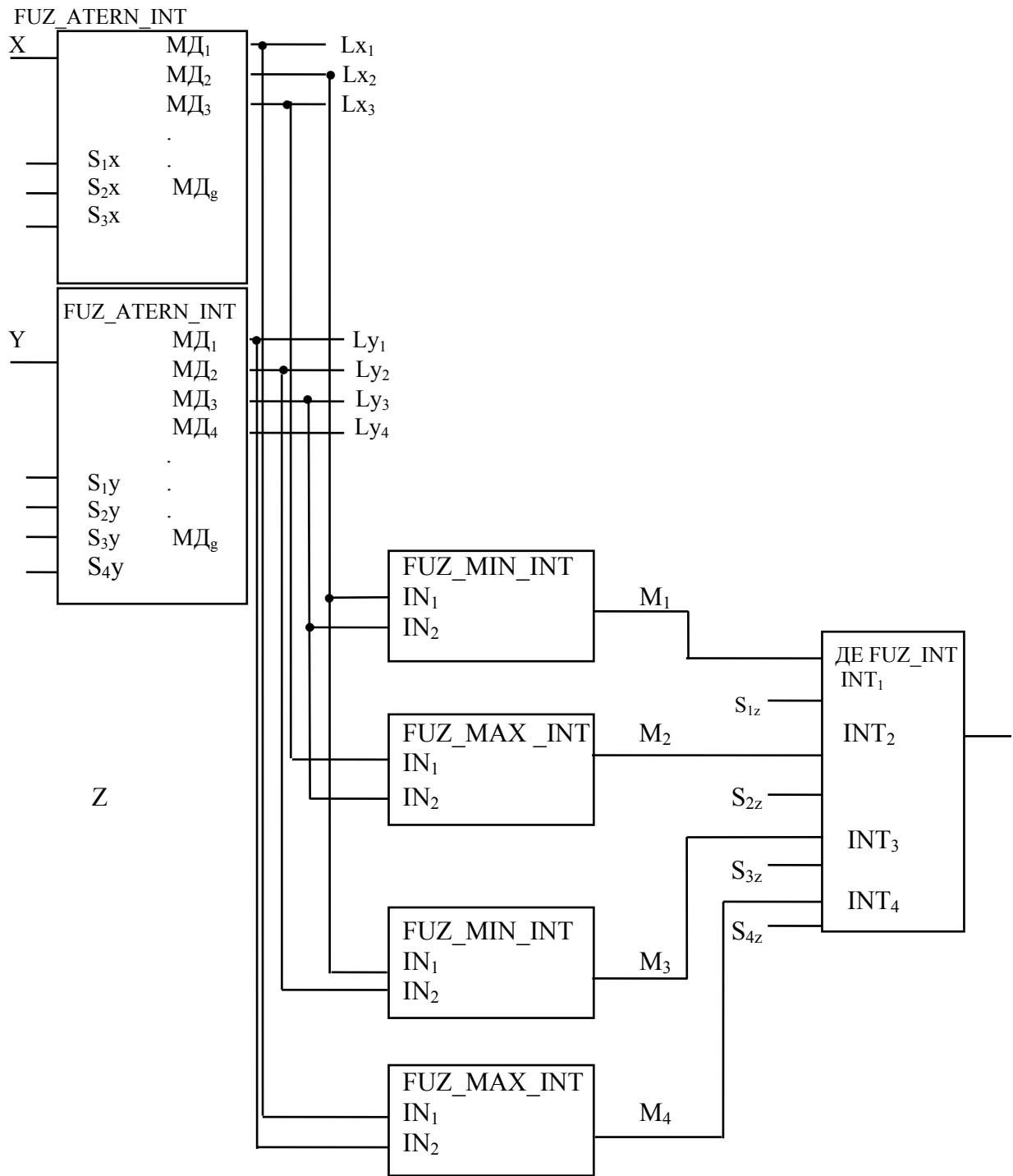


Рис. 3.7

На рис. 3.7 обозначены: S_{1x} , S_{2x} , S_{3x} – центральные значения базовых множеств, соответствующих лингвистическим значениям Lx_1 , Lx_2 , Lx_3 ; S_{1y} , S_{2y} , S_{3y} , S_{4y} – центральные значения базовых множеств, соответствующих лингвистическим значениям Ly_1 , Ly_2 , Ly_3 , Ly_4 ;

$S_{1z}, S_{2z}, S_{3z}, S_{4z}$ – центральные значения базовых множеств, соответствующих лингвистическим значениям Lz_1, Lz_2, Lz_3, Lz_4 ;

M_1, M_2, M_3, M_4 – результирующие функции принадлежности после обработки условной части правил P_1, \dots, P_4 ;

$\sum M_i(\max) S_i$, $M_i(\max)$ – максимальное значение;

$Z = \sum M_i(\max)$ – функции принадлежности M_i , $i = 1, 2, 3, 4$.

Пакет проектирования нечетких систем Fuzzy Tech

Пакет *Fuzzy Tech*, разработанный фирмой «Inform Software Corporation», предназначен для проектирования нечетких систем. Конечным продуктом при разработке системы является генерируемый при помощи пакета программный модуль.

В пакете имеются два редактора:

- для создания и работы с лингвистическими переменными;
- для работы с базой нечетких правил.

Работа с редактором переменных. Каждая лингвистическая переменная соответствует определенной исходной переменной, которая называется базовой. Для каждой базовой переменной вводятся диапазон изменения, а также значения, которые ей присваиваются. В случае, если они не определены для входной переменной или не вычислены для выходной, то обозначаются словом «Default» и используются, например, в случае, если есть ошибки или пропуски во входных данных.

Для определения лингвистической переменной задается ее имя и терм-множество возможных значений (Term name). Редактор для работы с переменными позволяет графически определить для каждой из возможных лингвистических переменных функцию принадлежности (сокращенное обозначение MBF).

В пакете *Fuzzy Tech* функции принадлежности определяются заданием координат точек определения (Definition point), которые соединяются

линейными или нелинейными функциями. Наиболее часто используют четыре стандартные функции принадлежности (рис. 3.8).

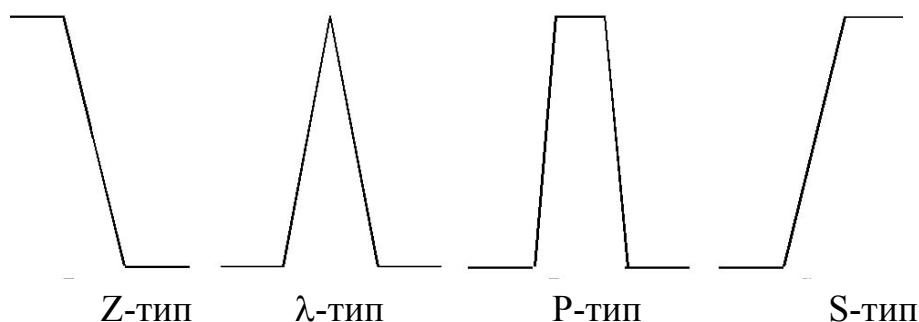


Рис. 3.8

Используются нормальные функции принадлежности

$$\max \{\mu(x)\} = 1, \min \{\mu(x)\} = 0.$$

При построении функций принадлежности для соседних лингвистических значений $L_i, L_i + 1$ координата максимума $\mu_{L_i}(x)$ совпадает с координатой минимума $\mu_{L_i+1}(x)$.

Значению $\mu(x) = 1$ соответствует наиболее типичное, наиболее ожидаемое значение аргумента.

Функции S-типа и Z-типа используются для крайнего правого и крайнего левого лингвистических значений. Нелинейные функции принадлежности в пакете Fuzzy Tech представляются кубическими сплайнами. На использование этого типа функций принадлежности указывает параметр Shape.

Дефазификация выполняется одним из трех методов.

СоМ – метод центра максимума, который используется, когда результатом нечеткого логического вывода может быть несколько термов выходной переменной. Данный метод является наиболее компромиссным;

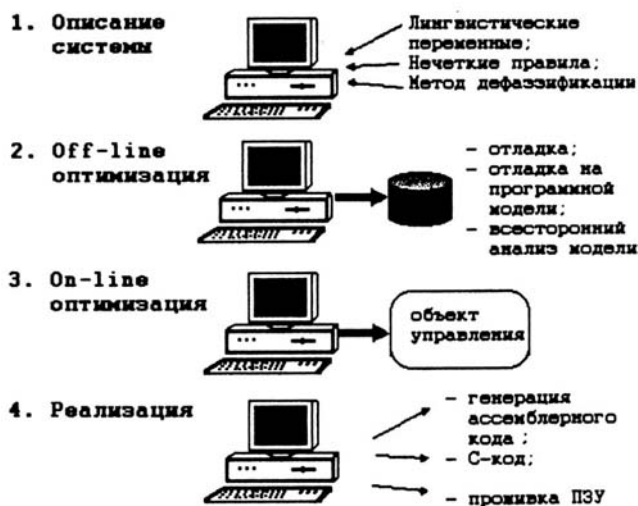
МоМ – метод максимума;

СоА – метод центра тяжести.

Редактор для работ с базой правил. Блоки правил используются для реализации стратегий управления. Каждый блок правил содержит правила для определенного решения. Основные операции, используемые для обработки правил, – это операции \max, \min .

Разработка нечеткой системы управления выполняется с помощью CAD-системы Fuzzy Tech и включает следующие этапы, схематично показанные на рис. 3.9.

1. Описание системы. На этом этапе при помощи средств, доступных



в пакете, задача формализуется. Здесь описываются лингвистические переменные, которые будут использоваться, и соответствующие функции принадлежности. Стратегия управления описывается нечеткими правилами, которые объединяются в базу знаний системы.

На этом этапе используются CASE-технологии, на основе которых построен пакет.

Рис. 3.9

2. Off-line-оптимизация. Выполняется проверка работоспособности созданной системы. Для этой цели вместо реального объекта можно использовать его программную модель. Для связи системы управления с моделью существует специально разработанный протокол связи fTlink, в основу которого положена концепция обмена сообщениями Windows.



Рис. 3.10

3. On-line-оптимизация. На этом шаге разрабатываемая система управления и реальный объект управления объединяются в единую систему (рис. 3.10). В этом случае в реальных условиях наблюдается поведение системы, а также, в случае необходимости, вносятся изменения в систему управления. После отладки создается окончательный вариант кода для конкретного

микроконтроллера. Основу программного кода, генерируемого пакетом,

составляет аппаратно-ориентированное на конкретный тип контроллера ядро. Генерируемый код состоит из трех основных частей:

- кода библиотечных функций;
- сегмента базы правил и функций принадлежности;
- функций нечеткой системы.

Контрольные вопросы

1. Какие алгоритмы обработки правил условного вывода реализованы в пакете MathLab?
2. В чем заключаются различия между алгоритмами Сугэно нулевого и первого порядка?
3. Какие методы дефазификации используют при построении нечетких контроллеров?
4. Какие типы функций принадлежности используются в пакете Fuzzy Tech?
5. Из каких этапов состоит процесс разработки нечеткой системы управления с использованием пакета Fuzzy Tech?

Глава 4

ПРИМЕРЫ ПРАКТИЧЕСКОЙ РЕАЛИЗАЦИИ НЕЧЕТКИХ АЛГОРИТМОВ СИСТЕМ УПРАВЛЕНИЯ

4.1. Нечеткая система управления доменной печью

Диапазон возможных применений нечетких алгоритмов и систем управления, реализующих эти алгоритмы, необычайно широк. Известно, что NASA рассматривает возможность применения, а возможно, уже применяет, нечеткие системы для управления процессами стыковки космических аппаратов. Можно также отметить сокращение публикаций по аппаратным средствам нечетких систем, связанное с интересами Министерства обороны США в этой области. На другом конце этого диапазона находятся часто рекламируемые стиральные машины с Fuzzy Logic. Однако более основательное применение нечеткие системы все-таки находят в традиционных областях управления промышленными объектами, некоторые примеры которых рассмотрены в нижеследующих разделах.

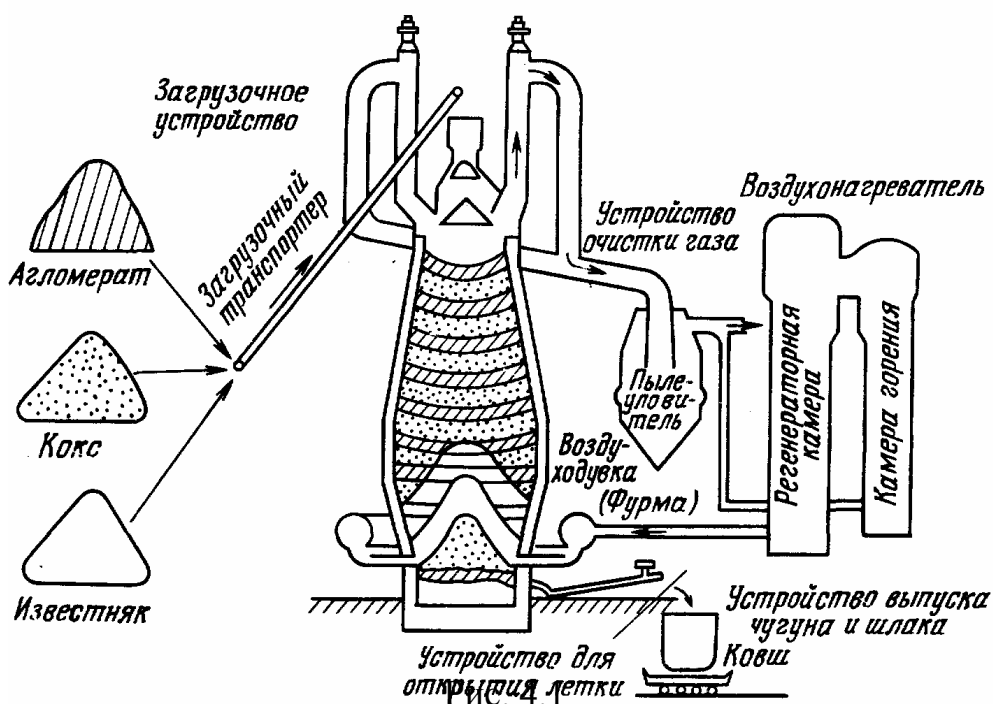
Одним из самых внушительных применений нечеткого управления как по объему и сложности задачи, так и по полученным результатам является нечеткая система управления доменной печью, описанная в [31]. Учитывая малую доступность этого источника, а также полноту изложения задачи, приведем описание этой системы, полностью заимствовав его из работы [31].

Устройство и работа печи

Доменная печь состоит из корпуса, загрузочного устройства, воздухонагревателей, воздуходувки для подачи горячего воздуха, устройства очистки газа, образующегося в печи, и лётки для чугуна (рис. 4.1).

В доменную печь через колошник попеременно загружаются агломерат-продукт спекания мелкой железной руды и кокс.

Через несколько десятков расположенных по окружности фурм непрерывно вдувается нагретое дутье с температурой 1100 °С, в результате чего кокс сжигается. Агломерат постепенно опускается из колошника в нижнюю часть (~6 м/ч) и за счет восстановительного газа, образующегося при сгорании кокса, непрерывно восстанавливается и плавится.



Доменная печь имеет цилиндрическую форму с диаметром нижней части около 15 м, диаметр колошника – 10 м. Расстояние от уровня засыпки шихты до оси лётки для чугуна составляет до 40 м. Внутренний объем печи превышает 5000 м³, а масса выплавляемого за сутки чугуна достигает 10000 т. Характерные особенности такой крупной доменной печи состоят в следующем.

1. За счет атмосферы с высокой температурой и высоким давлением в печи одновременно идут реакции трех фаз – газообразной, твердой и жидкой. Эти реакции чрезвычайно сложны и близки к «черному ящику».

2. Нагретый воздух, превращаясь в восстановительный газ, проходит расстояние от нижней части печи до колошника за несколько секунд. Время пребывания шихты в печи составляет 6 – 7 ч. Таким образом, процесс имеет большое время запаздывания.

3. После задувки печь должна работать непрерывно до остановки на ремонт (около 10 лет).

Доменная печь должна быстро реагировать на изменение производственных планов и рабочего режима и обеспечивать стабильное производство высококачественного чугуна. В связи с этим обязательным условием является надлежащее поддержание нагрева, для чего необходимо постоянное и точное управление нагревом. Для того чтобы поддерживать стабильный нагрев, следует избегать охлаждения печи из-за ошибок в работе и возникающих при этом аномальных явлений типа осадки и канального хода*.

Нагрев доменной печи

Сырье, загружаемое через колошник, за счет действия восстановительного газа, образующегося при сжигании кокса, восстанавливается и плавится; жидкий чугун и шлак опускаются и скапливаются в нижней части печи. Под нагревом печи подразумевают температуру жидкого чугуна и газа в ее нижней части. В общем случае при повышенном нагреве возрастает количество примесей в чугуне, и качество его снижается. Если нагрев слишком мал, плавление агломерата запаздывает и снижается объем производства чугуна. Таким образом, нагрев является важным индикатором рабочего состояния печи. Вместе с тем метод непосредственного измерения нагрева до сих пор не разработан, поэтому в качестве индикатора используют температуру чугуна на выходе из печи.

Нагревом печи можно управлять, изменяя массу подачи, порядок загрузки материалов, объем, температуру и давление горячего дутья и влажность дутья. Для этого специалисты на основе ежедневных результатов работы, показаний большого числа датчиков, работающих в реальном времени, информации, получаемой по статистической модели, и практических знаний о работе прогнозируют нагрев и состояние печи (при которых возникают осадка и канальный ход) и предусматривают оптимизацию загрузки сырья и нагрева дутья.

* Осадка – явление резкого оседания сырья за счет соскальзывания из состояния подвешивания (приостановка опускания загруженного сырья). В состояниях подвешивания и осадки изменяется объем сырья, подлежащего восстановлению и расплавлению, и нагрев становится неустойчивым. Канальный ход-явление, при котором сила давления восстановительного газа превышает вес загружаемого материала и газ резко устремляется к колошнику через каналы. Восстановительный газ при этом практически не участвует в восстановлении сырья, что вызывает охлаждение печи.

Система измерения и управления доменной печью

Состояние печи непрерывно меняется, и при нарушении равновесия внутри печи между загружаемыми материалами и давлением газа возникают аномальные явления типа осадки и канального хода. В этом случае существенно изменяется и нагрев. В связи с этим для управления необходимы датчики, а также модель для прогнозирования состояния печи и ее нагрева.

На рис. 4.2 показано размещение датчиков. В качестве датчиков используются в основном датчики температуры и давления. Их число в каждой печи достигает 1000. Показания датчиков обрабатываются на компьютере и оперативно в виде диаграмм изменений и распределений предоставляются оператору. Кроме того, было разработано большое число моделей управления нагревом, которые испытывали в реальных условиях. Однако, поскольку в печи протекают одновременно три сложные взаимосвязанные реакции газовой, твердой и жидкостной фаз, модели для практического применения пока еще нет. Поэтому для управления доменной печью широко использовали знания высококвалифицированных операторов.

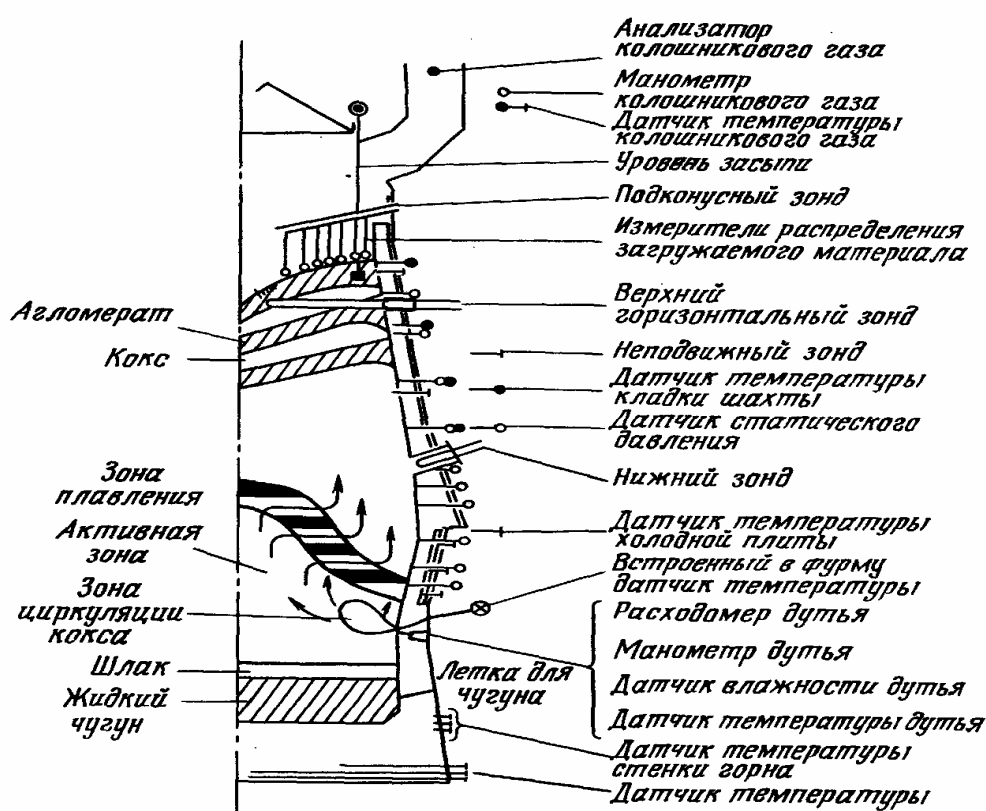


Рис. 4.2

Температура жидкого чугуна как показатель нагрева

Жидкий чугун и шлак, скопившиеся в нижней части печи, периодически выпускают из нее через четыре летки. В скиммерном устройстве главного желоба за счет разности удельных весов происходит разделение чугуна и шлака. Чугун сливают в ковш (емкостью 200 – 300 т) и транспортируют в сталеплавильный цех. Шлак сливают в шлаковую яму.

Сразу же после разделения чугуна и шлака измеряют температуру чугуна в скиммерном устройстве (рис. 4.3). Время между выпусками чугуна определяется с учетом износа

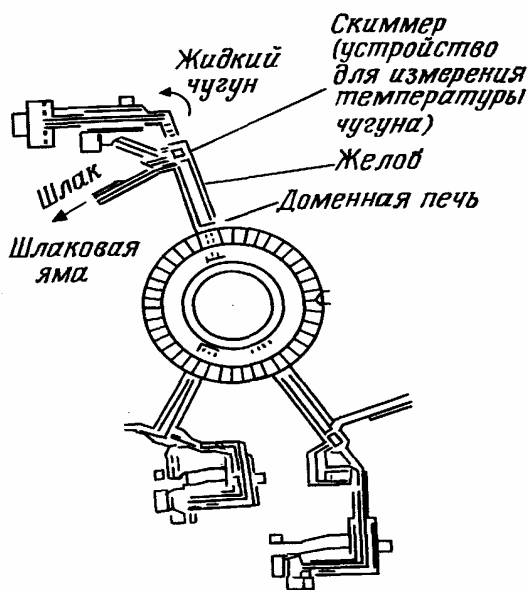


Рис. 4.3

футировки леток и желоба и составляет обычно 2 – 6 ч. Таким образом, из-за периодичности выпуска чугуна его температура (рис. 4.4) содержит свойственную процессу неопределенность.

В общем случае даже при устойчивом нагреве печи из-за длительного пребывания чугуна в ней происходит остывание горна, кроме того, из-за падения теплосодержания чугуна в желобе, который служит для него руслом, температура оказывается заниженной. При выпуске

время нахождения чугуна в печи уменьшается и лещадь остывает в меньшей степени. Температура желоба повышается, и температура чугуна начинает достаточно точно описывать нагрев.

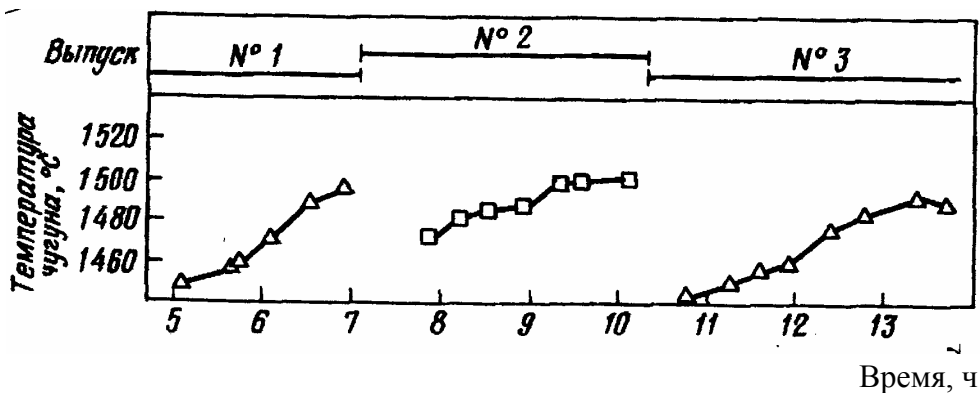


Рис. 4.4

Таким образом, в течение некоторого времени после выпуска нельзя получить информацию, хорошо описывающую нагрев. Более того, изменение температуры чугуна во время выпуска непостоянно и зависит от условий работы печи. Температура чугуна, используемая в качестве информации о нагреве, всегда содержит неопределенность. Из-за этого при оценке нагрева печи по температуре чугуна следует учитывать время от начала выпуска и условия работы, а также неопределенность, свойственную процессу.

Экспертная система управления нагревом печи

Структура системы

Система (рис. 4.5) состоит из управляющего компьютера (который осуществляет сбор данных от датчиков и их предварительную обработку для представления в виде, удобном для логических выводов), процессора искусственного интеллекта (ИИ) (который, используя базу знаний, делает выводы о нагреве печи) и цифровой контрольно-измерительной аппаратуры, управляющей нагревом по результатам выводов.

Процессор ИИ содержит программные средства поддержки экспертной системы на базе языка Лисп (средства ИИ) и машину выводов, интерпретирующую знания. Управляющий компьютер кроме функций предварительной обработки данных от датчиков оснащен функциями анализа данных и диагностики нагрева по математической модели, разработанными и используемыми ранее. На основе информации от датчиков и базы знаний в системе делается вывод о текущем уровне нагрева с периодом 20 мин и об увеличении или уменьшении нагрева; по результатам вывода с помощью правил управления определяются объем дутья и другие параметры управления.

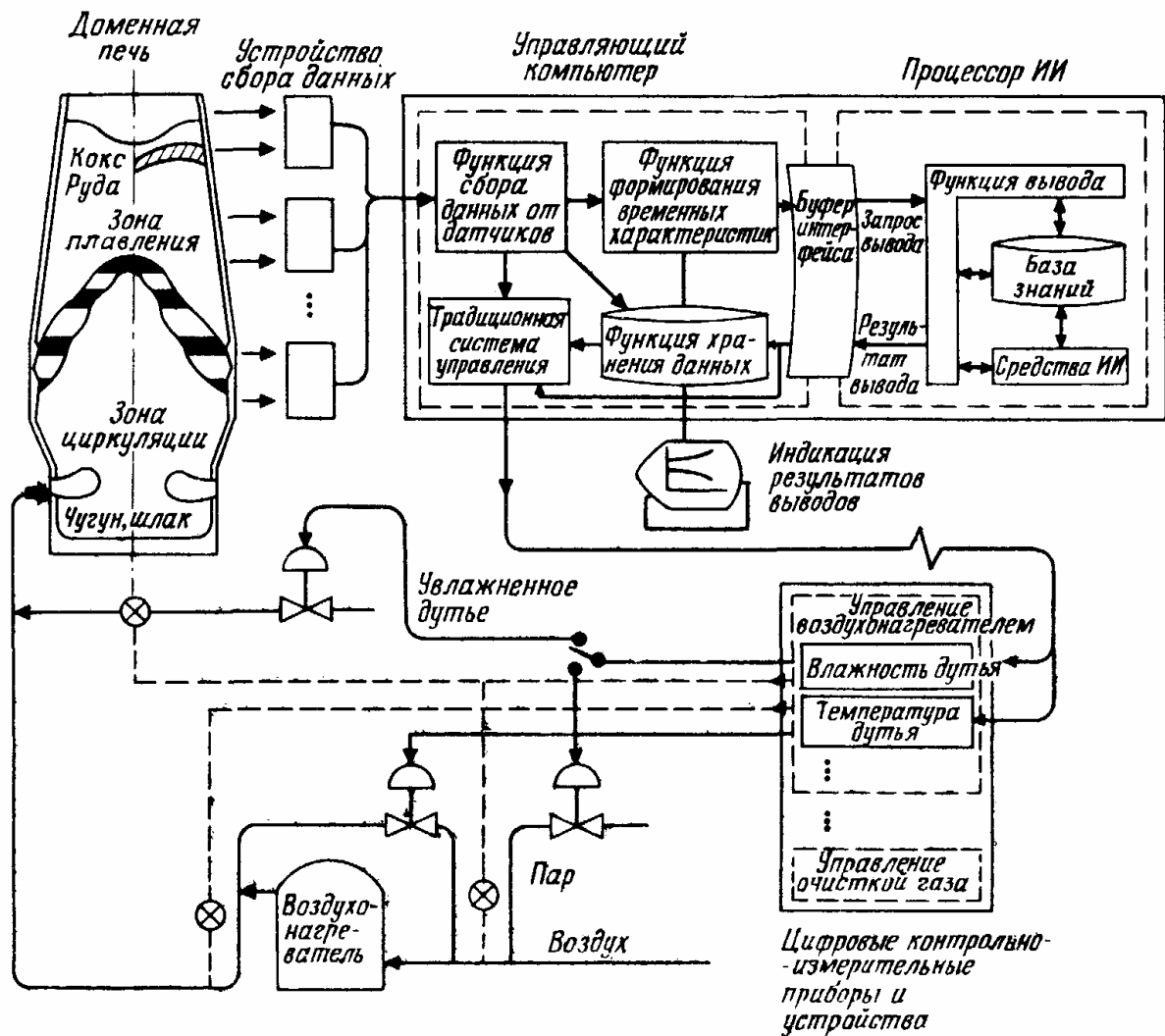


Рис. 4.5

Механизм выводов

Механизм выводов представлен на рис. 4.6. Источники знаний (ИЗ) объединяются в блоки для каждого функционального элемента системы, включают атрибуты датчиков и правила и размещаются в процессоре ИИ. Данные от датчиков, собранные компьютером, преобразуются в фактические данные для вывода и записываются на доску объявлений* (ДО). Процесс вывода о нагреве начинается с запуска машины выводов с помощью механизма планирования в реальном времени, затем машина выводов запускает ИЗ об управлении выводом. Например, если из некоторого ИЗ о датчиках выводится промежуточное заключение о нагреве, то это заклю-

* Область памяти, общедоступная для всех модулей системы.

чение и информация, относящаяся к ИЗ о нагреве, заносятся на ДО. Управление выводом всегда осуществляется по информации на ДО с последующим определением ИЗ, который следует запустить.

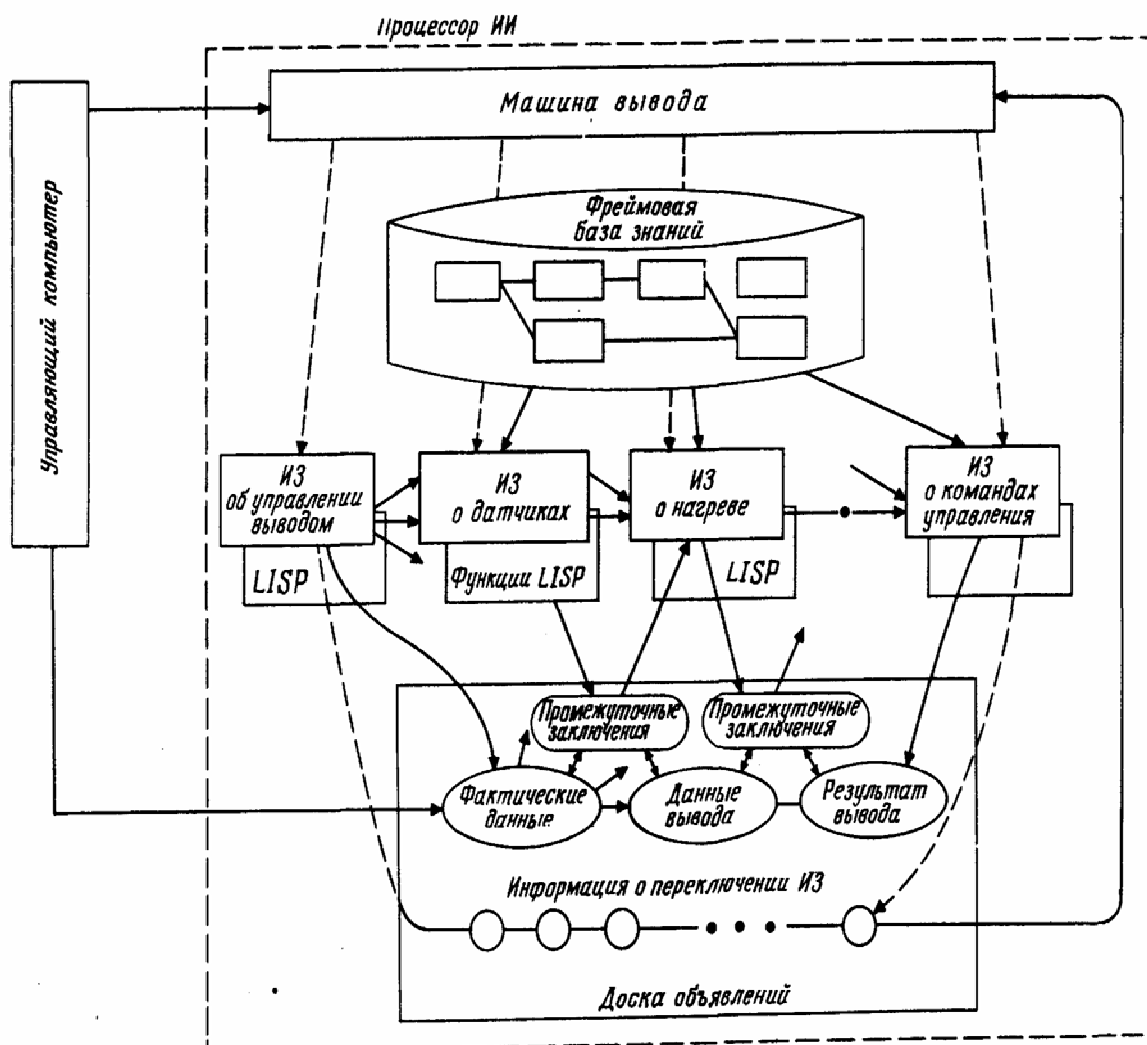


Рис. 4.6

Технологические знания об управлении нагревом

На рис. 4.7 в рамках инженерии знаний показано развитие технологических знаний, используемых до настоящего времени для управления нагревом. Среди них имеются диагностические знания о нагреве, фундаментальные знания о доменной печи, обычные знания (здравый смысл) оператора печи и др. Диагностические знания содержат сведения, позволяющие диагностировать нагрев по наблюдениям оператора (условия сго-

рания кокса перед фурмами, цвет шлака на выходе и т.п.), и знания для получения и проверки гипотез о нагреве по характеру специальной информации. Оператор, мгновенно сопоставляя эти знания, оценивает нагрев и производит необходимые действия, применяя знания об управлении нагревом, руководство по эксплуатации и другую информацию.

Фундаментальные знания о доменной печи содержат специальные знания, получаемые при теоретическом анализе в рамках теории металлургических реакций, сведения, получаемые с помощью замораживания*, и знания по технике эксплуатации, подтвержденную многолетним опытом. Кроме этих знаний существуют обычные знания, используемые только специалистами по доменной печи. Например, зона плавления – это зона вблизи 1250 °С, в которой происходит размягчение и плавление руды; подвисяние – это явление временной приостановки опускания сырья в печи и т.п.

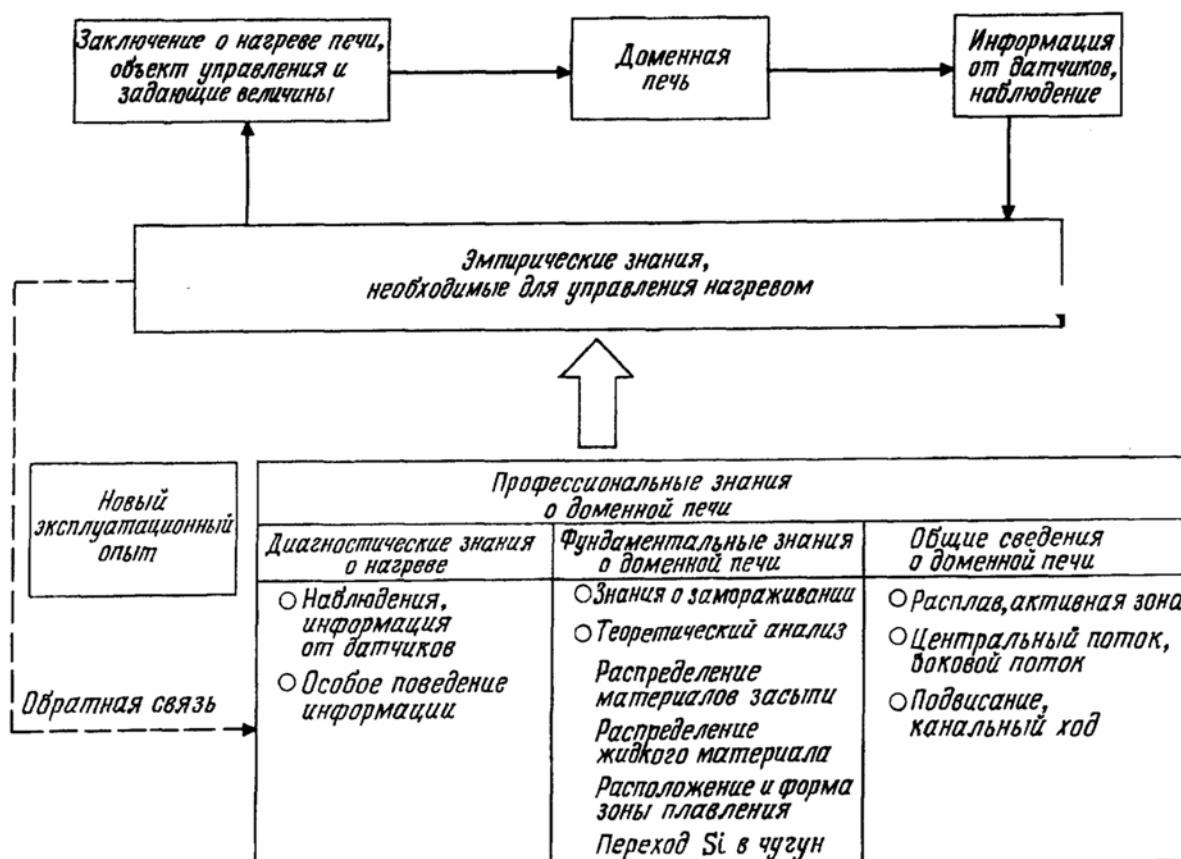


Рис. 4.7

* Исследования условий восстановления и плавления агломерата и кокса в печи проводятся путем прекращения дутья в действующую печь, заливки водой и извлечения остатков через колошник.

Представление знаний и база знаний

В данной системе для описания специальных знаний, используемых для вывода, введены три формы представления – порождающие правила, фреймы и функции Лиспа. Основная часть знаний представляется в виде продукционных правил, распределяется по функциональным элементам со своими правилами и атрибутами датчиков и объединяется в блоки, образуя иерархическую структуру.

На рис. 4.8 показаны пример правил и структура базы знаний. Например, группа ИЗ о нагреве состоит из четырех ИЗ. ИЗ о температуре жидкого чугуна по измеренной его температуре делает вывод о нагреве и содержит около десяти правил.

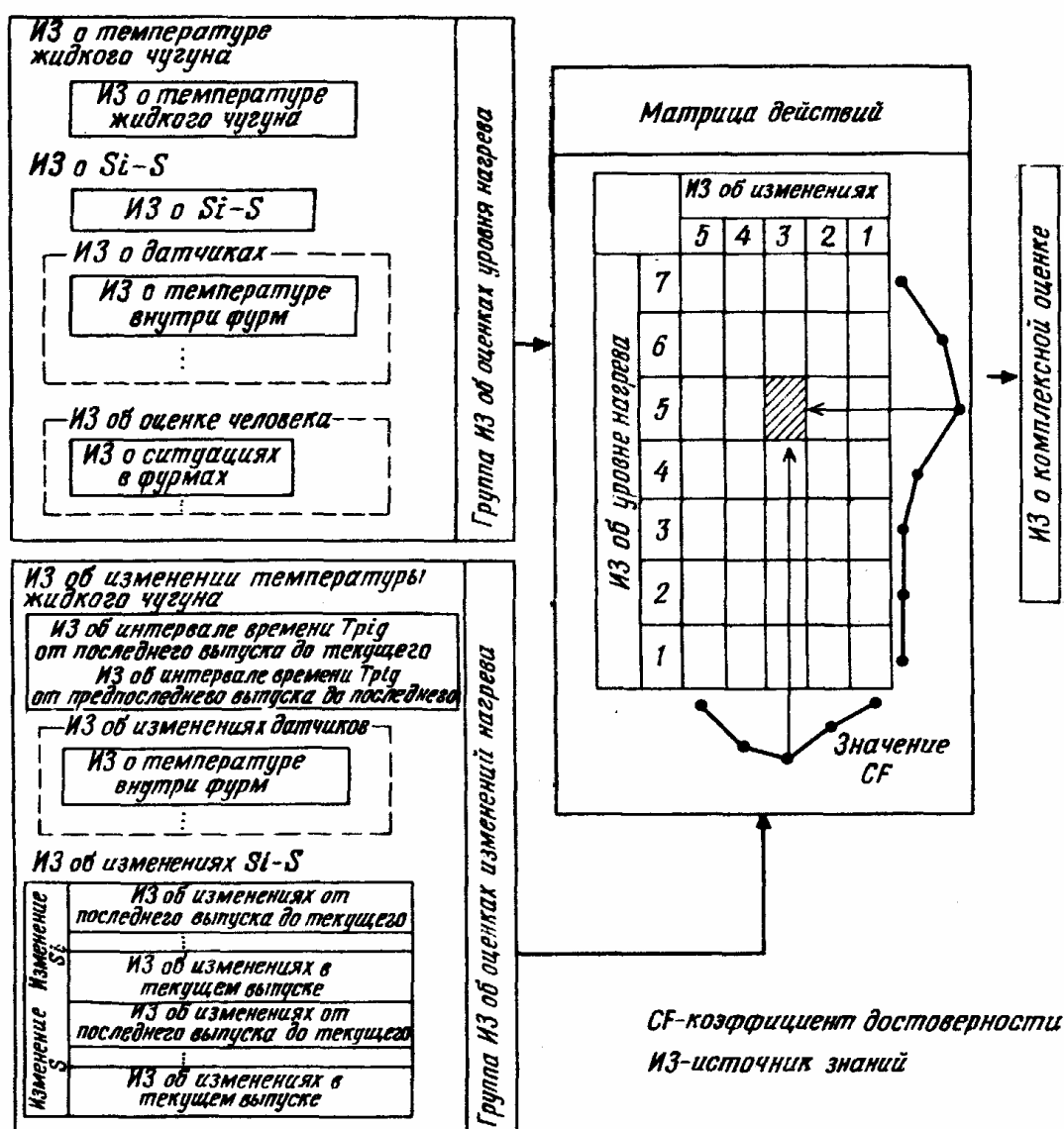


Рис. 4.8

Вывод о температуре чугуна делают с учетом времени, прошедшего с начала выпуска, и числа ковшей, используя приведенное ниже понятие нечеткого множества. Затем делают вывод о нагреве с использованием ИЗ о содержании в чугуне кремния (Si), серы (S) и других элементов. После окончания этой обработки запускаются ИЗ о датчиках и ИЗ об оценке человека, и по результатам выводов в каждой группе ИЗ делается заключительная оценка уровня нагрева.

Фреймы содержат постоянные величины, которые служат исходными данными для вывода: влажность и температура дутья для управления нагревом, содержание кокса, задающие величины. Во фреймы часто включают знания общего характера, поэтому стараются повысить эффективность представления знаний, используя их преемственность.

С помощью функций Лиспа описываются знания процедурного типа для вычисления поправок к параметрам рабочего воздействия по степени достоверности, представляющей четкость в знаниях, и по предыстории изменения задающих величин.

Представление нечеткостей

Одна из проблем данной экспертной системы, содержащей эмпирические правила, – представление нечеткостей в знаниях. Для ее решения обычно используют степени достоверности СФ-вывода для каждого правила либо нечеткие множества. При нечетком управлении, основанном на теории нечетких множеств, с помощью функций принадлежности, позволяющих в естественном виде представить субъективные нечеткие понятия, свойственные человеку, описываются профессиональные знания квалифицированного оператора и реализуется управление, аналогичное тому, которое он может выполнять. В производстве чугуна такое управление уже разрабатывается и внедряется для оперативного управления процессом агломерации в реальном времени.

Вместе с тем доменный процесс является очень сложным процессом, в котором одновременно протекают реакции трех фаз – газообразной, твердой и жидкой. Поэтому профессиональные знания квалифицированного оператора трудно представить только функциями принадлежности и так же трудно реализовать управление в виде единой системы.

В связи с этим в рассматриваемой системе в качестве способа представления обширных профессиональных знаний использовали порожд-

дающие правила, а в качестве средства представления нечеткостей ввели понятия теории нечетких множеств. Наряду с простотой представления знаний это позволило избежать увеличения числа правил и сократить время вывода. В результате появилась возможность оперативного управления в реальном времени на базе экспертной системы.

Введение функции принадлежности

Когда состояние нагрева прогнозируется по измеренным значениям температуры чугуна, оператор не сразу определяет уровень нагрева. Поэтому в данной системе нагрев, который фактически является непрерывной функцией, был разделен на семь уровней с учетом опыта оператора (табл. 4.1).

Таблица 4.1. Уровни и состояние нагрева

Уровень	Состояние нагрева
7-й	Сильный перегрев
6-й	Перегрев
5-й	Легкий перегрев
4-й	Нормальный нагрев
3-й	Легкий недогрев
2-й	Недогрев
1-й	Сильный недогрев

Обычно оператор по одному замеру вряд ли скажет, что нагрев сейчас на седьмом уровне; скорее всего он сделает вывод, что текущий уровень – шестой или седьмой. В данной системе не дается однозначного заключения по фактическим данным (температуре чугуна), а вводится нечеткость, которая для одного замера выражается показателем достоверности по каждому уровню. Предложен способ представления показателя достоверности на каждом уровне с учетом удобства эксплуатации с помощью трехмерной обобщенной функции принадлежности, состоящей из трех элементов: фактические данные (например температура чугуна), заключение (например уровень нагрева) и показатель достоверности.

На рис. 4.9 показана обобщенная функция принадлежности для температуры чугуна, уровня нагрева и показателя достоверности для N-го ковша в выпуске: X – температура жидкого чугуна, °С; Y – уровень нагрева; Z – показатель достоверности. Если, например, измеренное значение температуры чугуна составляет X_0 , то сечение Y - Z при $X=X_0$ дает показатели достоверности Z_j для каждого уровня нагрева ($j' = 1, \dots, 7$).

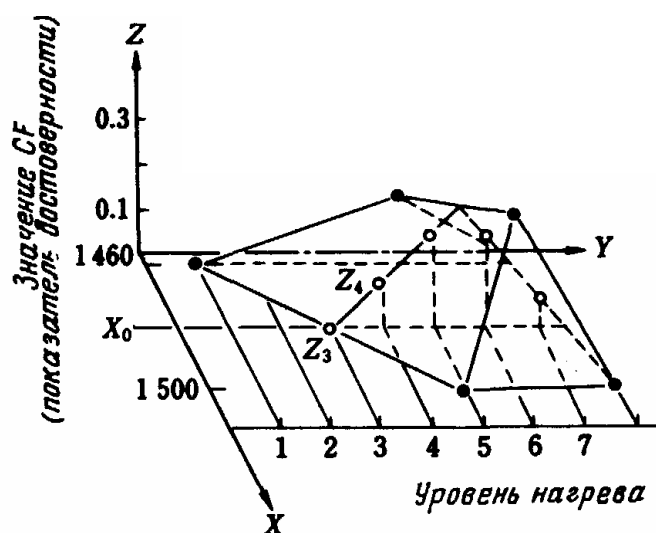


Рис. 4.9

Поскольку смысл информации и ее нечеткость меняются со временем от начала выпуска, в качестве параметра для температуры чугуна в данной системе выбрано число ковшей в выпуске. Кроме того, обобщение функции принадлежности введено не только для температуры чугуна, но и для другой информации от датчиков, связанных с нагревом.

Способ формирования функций принадлежности

Ниже описан способ формирования обобщенных функций принадлежности, используемых в данной системе (рис. 4.10). Построена зависимость между измеренными значениями температуры чугуна (ось X) для N-го ковша (см. рис. 4.10, а) и максимальной температурой чугуна во время выпуска (ось Y), наилучшим образом описывающей нагрев. На рис. 4.10, в эта зависимость представлена в виде функций принадлежности, при этом выделены определенные температурные области, а по оси Z показана частота появления максимальных температур в каждой области. На рис. 4.10, б

показана обобщенная функция принадлежности, которая получена путем аппроксимации всех функций принадлежности в виде треугольников и соединением соответствующих вершин. Поскольку температура жидкого чугуна зависит от времени, прошедшего после начала выпуска, и условий работы печи, были подготовлены более 30 обобщенных функций принадлежности, с тем чтобы их можно было выбирать автоматически в зависимости от условий работы. Формы этих функций описываются в виде фреймов.

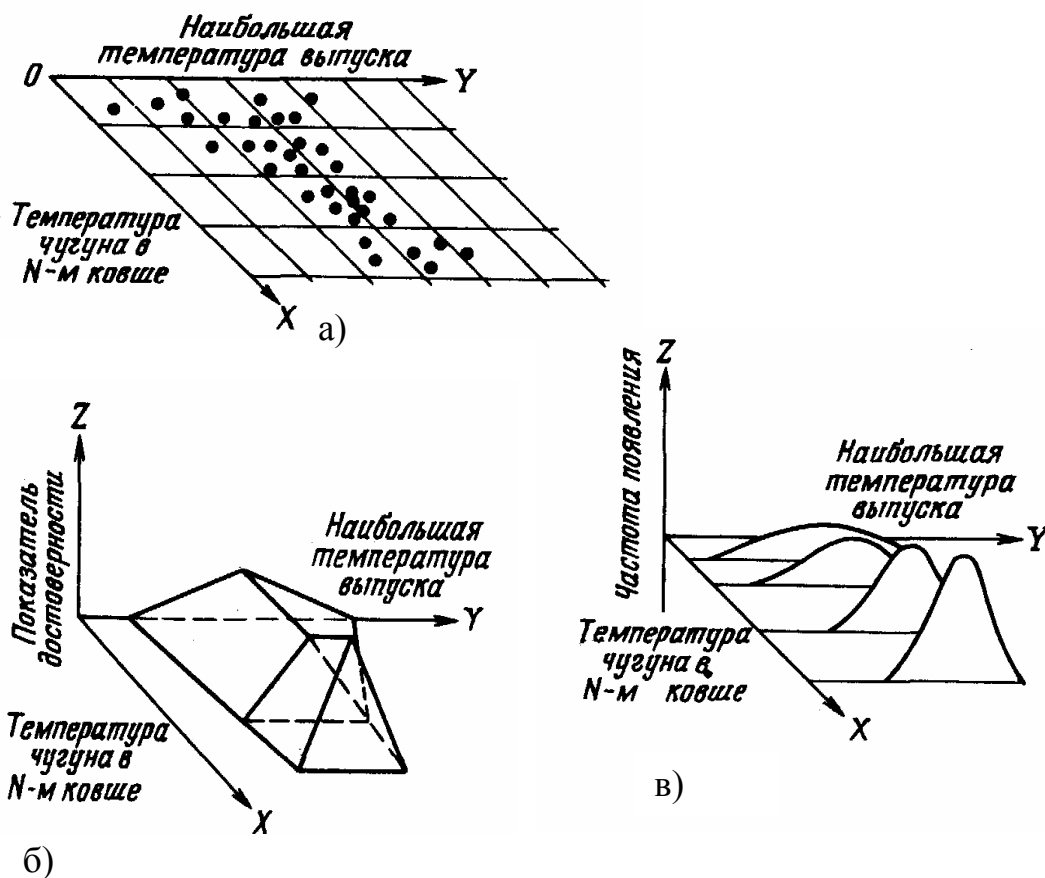


Рис. 4.10

Обучение обобщенной функции принадлежности

Если управление процессом осуществляется на основе профессиональных знаний операторов, как в данной системе, то эффективность системы будет зависеть от того, насколько функции принадлежности, выражающие нечеткости, согласуются с профессиональными навыками оператора. Более того, эти навыки постоянно изменяются при смене оборудования и условий работы печи.

По этим причинам срок службы системы в значительной степени зависит от способности функции принадлежности отслеживать изменения в оборудовании и условиях работы. В данной системе с учетом этих обстоятельств обобщенные функции принадлежности наделены способностью к обучению по методу, показанному на рис. 4.11. В компьютере накапливаются данные от датчиков и данные о работе печи за несколько последних месяцев; время от времени эти данные извлекаются, из них удаляются аномальные данные, такие, как данные, полученные до и после остановки дутья или во время аномальных явлений (осадки и канального хода), данные для случаев, когда имеются резкие расхождения максимальных температур чугуна во время выпуска до и после перекрытия летки, и другие данные. Затем выделяются четыре случая (табл. 4.2) и проводится анализ причин. В случае 2 функция принадлежности формируется заново методом, изложенным в предыдущем пункте.

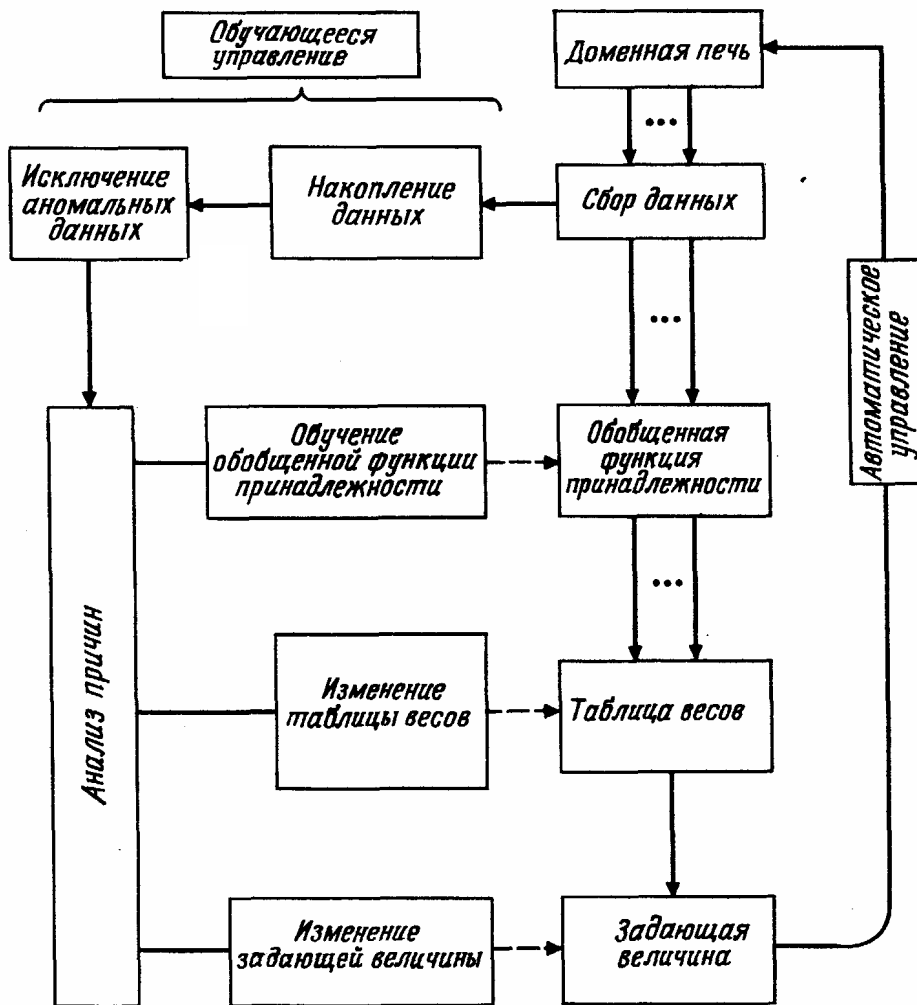


Рис. 4.11

Таблица 4.2. Анализ причин и обработка данных

Случай	Анализ причины	Обработка данных
1	Отклонение температуры чугуна от целевого значения и изменения малы	Сохраняется текущее состояние
2	Отклонение температуры чугуна от целевого значения и изменения велики	Обучение функции принадлежности
3	Условия работы печи меняются; управляемость ухудшается	Изменение таблицы весов
4	Температура чугуна стабилизируется на высоком или низком уровне (при малых значениях задающих величин)	Изменение весов

Результаты внедрения нечеткого управления и обучения

Результаты внедрения нечеткого управления. В данной системе в качестве средства представления нечеткостей, содержащихся в профессиональных знаниях операторов, были введены понятия теории нечетких множеств. В результате по сравнению с описанием знаний с помощью только продукционных правил это введение позволило упростить представление знаний, уменьшить число правил и повысить эффективность выводов, а также появилась возможность оперативного управления в реальном времени. При этом существенно повысились удобства технического обслуживания.

Результаты внедрения обучающегося управления. Способность функций принадлежности к обучению позволяет легко адаптироваться к различиям в технологических процессах и к изменениям режима работы печи. Можно ожидать повышения управляемости нагревом, увеличения срока службы системы, расширения области применения системы для других доменных печей и т.д.

Рассмотренная система начала применяться в январе 1987 г. и успешно работает до настоящего времени; коэффициент использования автоматического управления превышает 95 %.

В дальнейшем возможности для применения принципов нечеткого управления в системах управления технологическими процессами будут расширяться. Методы применения могут быть различными в зависимости от характера процесса. Метод, изложенный в данном пункте, может быть эффективным в случае его применения в качестве части экспертной системы для таких сложных процессов, как доменный.

4.2. Использование нечетких алгоритмов управления в автомобильных системах

Электронные контроллеры для управления различными процессами в автомобиле получили в последнее время очень широкое распространение. В то же время это связано с решением весьма сложных задач, которые необходимо решать в реальном масштабе времени с периодом цикла управления в единицы миллисекунд при необходимости обработки больших по объему таблиц данных. Кроме этого обычно разработка традиционных автомобильных контроллеров выполняется в предположении, что удовлетворяются три условия:

- идентичность объектов;
- стационарность объекта;
- незначительность возмущений, действующих на объект.

В реальных ситуациях эти три условия выполняются лишь приблизительно или же совсем не выполняются. Кроме этого практически невозможно точно описать все ситуации, в которых должны работать автомобильные контроллеры. Все это создает значительные перспективы для применения нечетких контроллеров в автомобильных системах.

Первыми на нечеткую логику обратили внимание японские автомобилестроители. В 1981 году компания «Nissan» впервые применила компоненты нечеткой логики в системе управления пятиступенчатой коробкой передач, годом позже появилась аналогичная коробка на автомобилях «Honda», затем компания «Mitsubishi Motors» представила свою «разумную» коробку передач, компания «Renault» совместно с «Siemens» разработала проактивную коробку переключения передач, предназначенную для автомобиля «Renault Megane». К сожалению, открытая информация о применяемых в автомобильных системах контроллерах с нечеткой логикой на сегодня просто отсутствует. Есть только фрагментарная информа-

ция об идеях, заложенных в эти системы. Прежде всего в системах нечеткого управления выполняется анализ манеры вождения автомобиля, в основу которого положены идеи нечеткой логики.

В память компьютера закладывается лингвистическое описание степени нажатия педали газа. Допустим понятие «степень нажатия педали газа велика». Если педаль нажата на 60 %, то это соответствует понятию сильного нажатия педали на 40 % (рис. 4.12). Исходя из этого выполняется ограничение по уровню 40 % функции принадлежности, определяющей лингвистическое описание манеры вождения. Для заштрихованной области вычисляется центр тяжести, координата которого определяет, что в данный момент спортивная манера вождения имеет место на 50 %. Соответственно контроллер определяет, на каких оборотах будет выполняться переключение передач. Для спокойной манеры вождения (см. рис. 4.12, а) переключение выполняется при меньших оборотах, при спортивной (см. рис. 4.12, б) – при больших. Использование нечеткой логики позволяет нечеткому контроллеру определять моменты, когда при сбросе газа нужно включить низшую передачу, а когда нет, при этом при анализе используется не один параметр (частота торможения), а шесть.

Типичный пример системы, хорошо поддающейся реализации с помощью нечеткой логики, – АБС – антиблокировочная тормозная система. Реализаций АБС существует достаточно много, но в общем случае управление выполняется по двум входным параметрам: проскальзыванию колеса

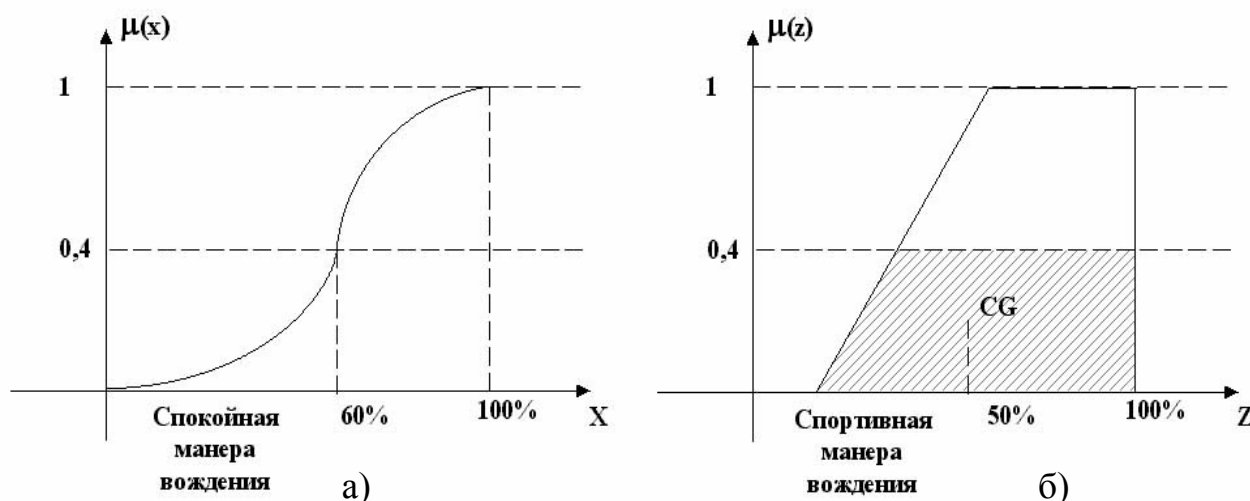


Рис. 4.12

(отношение скорости автомобиля к мгновенной линейной скорости точки на внешнем радиусе колеса относительно его центра) и реальному ускорению колеса. В нечетких АБС оба параметра представляются в виде логических переменных с набором из 5 – 8 термов, на основании которых кон-

троллер, используя набор правил, количество которых равно произведению количества термов входных переменных, получает значение давления в тормозном цилиндре, стремясь к поддержанию оптимального проскальзывания.

Одним из самых сложных объектов для систем управления является двигатель внутреннего сгорания (ДВС). Прежде всего это объясняется тем, что полная математическая модель ДВС слишком сложна и до сих пор не создана. Из-за этого большинство систем управления ДВС используют табличную модель, полученную экспериментальным путем на испытаниях и с учетом опыта экспертов. Серьезные недостатки такого подхода – сложность создания многомерных таблиц и большой объем памяти, требуемый для их хранения. Сегодняшние табличные системы используют в основном регулирование по двум параметрам. Например, для управления углом опережения зажигания (УОЗ) базовая характеристика представляет собой примерную функцию, отражающую оптимальную для установившегося режима и прогретого двигателя взаимосвязь между УОЗ, частотой вращения и нагрузкой, определяемой давлением во впускном трубопроводе, расходом воздуха или положением дроссельной заслонки. Нечеткая логика позволяет заменить таблицы правилами и реализовать управление УОЗ. Нечеткая система управления УОЗ входит в комплексную нечеткую систему управления ДВС (фирма «Nissan»), представленную на рис. 4.13.

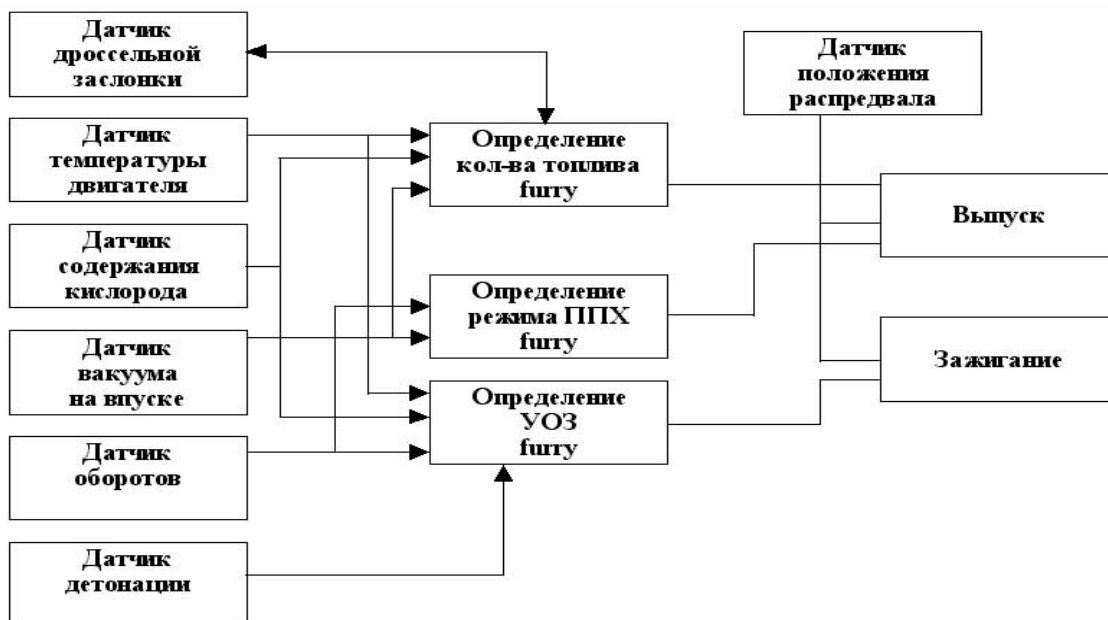


Рис. 4. 13

Основная идея состоит в том, что определяется состояние двигателя, описываемого лингвистической переменной, для которой заданы несколько

130

ко термов: «Запуск», «Холостой ход», «Низкая нагрузка», «Большая нагрузка», «Торможение», «Разгон». Выбор терма определяется сигналами с датчиков, и соответствующие нечеткие модули вырабатывают сигналы управления для подсистем впрыска, зажигания и принудительного холостого хода.

Автоматические трансмиссии – еще один объект управления, где нечеткая логика имеет большие перспективы. Задачи, решаемые здесь, не описываются математически, но вполне допускают словесное описание.

В автомобильных системах очень важно, что одним из источников входных сигналов является человек, точнее, его действия в процессе движения, которые позволяют с некоторым приближением понять характер дороги или желания водителя.

Все это позволяет говорить как о реальном направлении развития автомобильных систем, так и о создании «интеллектуальных» систем управления.

4.3. Нечеткое управление концентрационным режимом алюминиевых электролизеров

Процесс электролиза алюминия характеризуется, с одной стороны, достаточной сложностью и многообразием протекающих в электролизной ванне физико-химических процессов, с другой – отсутствием возможности непрерывного оперативного контроля за рядом важнейших технологических параметров (температурой, уровнем электролита и др.), характеризующих эти процессы, что создает значительные трудности при разработке эффективных алгоритмов автоматического управления.

Одной из важнейших задач при управлении технологическим процессом электролиза является обеспечение оптимального концентрационного режима электролизера. Наилучшие технико-экономические показатели процесса обеспечиваются при поддержании концентрации глинозема (Cr) на уровне 1,5 – 3,5 %. Чрезмерное снижение этой концентрации ведет к истощению ванны, т.е. ухудшению ее работоспособности, увеличение – к выпадению глиноземных остатков, т.е. к неэффективному использованию сырья.

На электролизерах, которые загружаются один раз за период $T = 1,5 - 3$ ч, стабилизация концентрации в узкой области практически невозможна. Концентрация глинозема циклически меняется от $C_{max} = 5 - 6$ %, достигаемой непосредственно после обработки, до $C_{min} = 1 - 2$ % перед началом следующего цикла.

Задача управления электролизной ванной заключается в обеспечении сбалансированной концентрации сырья, непосредственное измерение которой в условиях существующего производства невозможно. Поэтому на первый план выходят методы косвенной оценки концентрации сырья.

Однако многие из этих методов достаточно сложны, что ограничивает область их применения и, кроме этого, не обеспечивают необходимую точность расчета C_r .

Уровень концентрации глинозема можно оценивать по изменению электрических параметров процесса.

Падение напряжения U на электролизной ванне определяется соотношением

$$U=IR+E,$$

где I – ток серии, А;

R – активное сопротивление электролизера, Ом;

E – электродвижущая сила, В.

Зависимость этих параметров от концентрации C_r представлена на рис. 4.14.

В современных алгоритмах управления процессом электролиза алюминия, чтобы исключить влияние колебания тока, напряжение $U_{пр}$ на электролизной ванне рассчитывается как приведенное к номинальному току серии I_n

$$U_{пр} = (U - E) I_n / I + E.$$

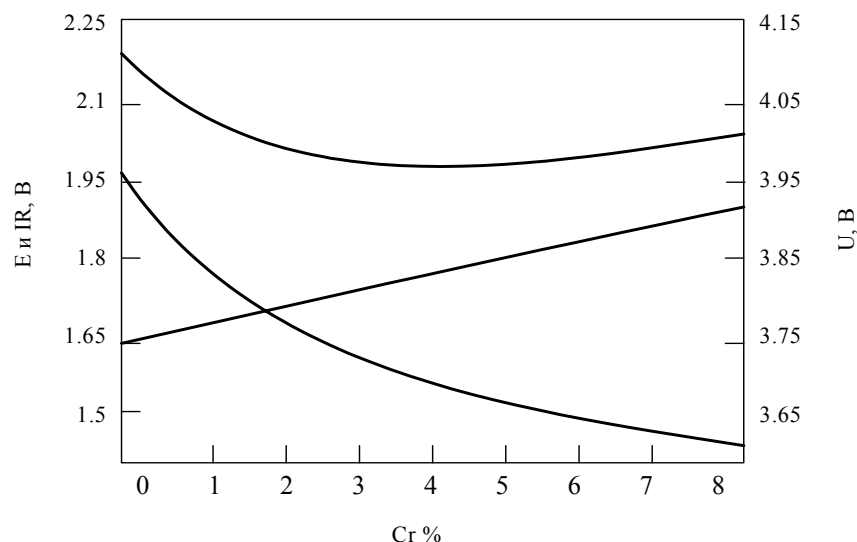


Рис. 4.14

Типичная зависимость напряжения $U_{пр}$ от времени t в интервале между загрузками глинозема приведена на рис. 4.15.

По характеру деформации кривой $U_{пр(t)}$ в сравнении с некоторой эталонной можно судить о концентрационном состоянии электролизной ванны. Существует ряд алгоритмов ситуационного управления концентрационным режимом электролизеров, питание которых осуществляется с помощью передвижных механизмов, основанных на анализе вида зависимости $U(t)$ в интервале между подачами глинозема. В частности, алгоритм управления, представленный в работе [31], базируется на анализе длительности временного промежутка t между загрузкой глинозема и началом роста напряжения (см. рис. 4.15).

В упомянутых алгоритмах используют сведения в неформализованном виде с применением для их описания лингвистических переменных со значениями "Много", "Мало" и т.д., т.е. предпринимается попытка имитировать действия эксперта, принимающего решение (ЛПР) [37].

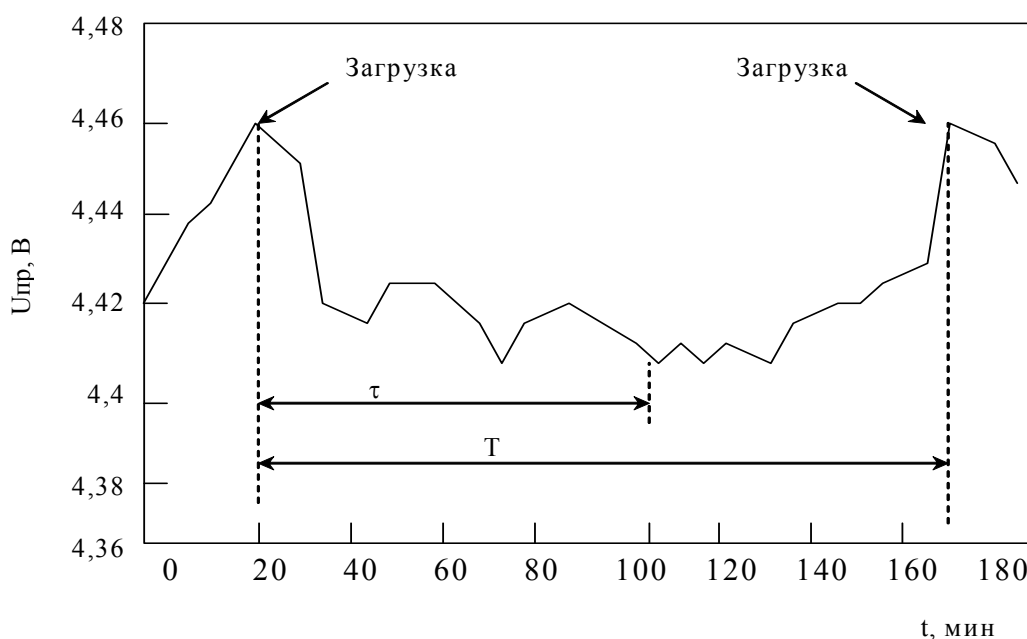


Рис. 4.15

Алюминиевый электролизер является объектом, в котором сложность и многообразие протекающих физико-химических превращений сочетаются со слабой информативностью процесса и в некоторых случаях значительную роль играют скорее не количественные показатели, а качественные оценки. Хотя электролизная ванна и обрабатывается по регла-

менту, решения по коррекции количества загружаемого глинозема принимаются оператором-технологом на основе личного опыта и носят, по сути, интуитивный характер. Как следствие этого, процесс управления концентрационным режимом плохо поддается автоматизации в рамках четкой математической логики. Применение же аппарата нечеткой логики позволит моделировать механизм принятия решения оператором, используя опыт ЛПР в полной мере, и, следовательно, более эффективно автоматизировать процесс.

Благодаря тому что существует нелинейная зависимость между скоростью изменения напряжения U и концентрацией C_r , представляется возможным суждение о концентрационном состоянии электролизной ванны основывать на анализе скорости изменения напряжения dU/dt в интервале 20 – 30 мин непосредственно перед загрузкой очередной порции глинозема. При этом характер изменения dU/dt за период между обработками зависит от начального значения C_r , достигаемого после растворения глинозема. Тогда в области низких концентраций, когда начальное значение C_r меньше некоторого оптимального, величина dU/dt может быть положительна с самого начала и к концу периода T резко возрастает. В области же высоких концентраций величина U может уменьшаться в течение всего периода времени между обработками и достигнуть в конце периода минимального значения. Таким образом, большая скорость приращения напряжения в конце периода T должна свидетельствовать о сильном истощении ванны, незначительный рост или даже снижение названной скорости – о возможной перепитке ванны. Для устранения влияния различных помех (например колебания тока серии) в алгоритме должны быть предусмотрены необходимые процедуры обработки зависимости $U(t)$, включающие в себя усреднение, расчет приведенного значения напряжения $U_{пр}$, фильтрацию, а также коррекцию $U_{пр}$ в случае перемещения анода.

Концентрационное состояние электролизной ванны характеризуется лингвистической переменной «Концентрационный режим», которая в зависимости от скорости изменения напряжения U может принимать три вербальных значения (терма): «Недопитка» (НП), «Нормальный режим» (Н), «Перепитка» (ПП).

Как показала практика, этого качества лингвистических значений вполне достаточно для управления. Непрерывные кусочно-линейные функции принадлежности нечетких множеств "НП" $\tilde{C}_{НП}$ "Н" $\tilde{C}_Н$ и "ПП" $\tilde{C}_{ПП}$, построенные на интервале 0 – 0,04 мВ/с, представлены на рис. 4.16.

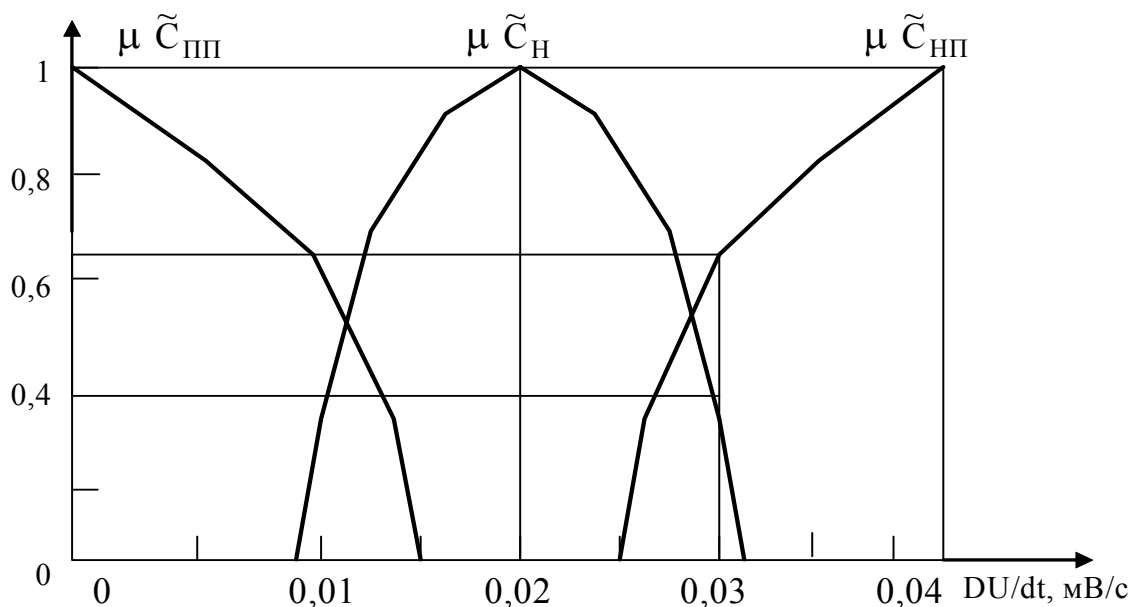


Рис. 4.16

Возможные управляющие решения по количеству загружаемого глинозема ограничены в зависимости от конкретной ситуации тремя управляющими воздействиями из терм-множества R лингвистической переменной «Подача глинозема»: «Уменьшить подачу» (УМ), «Не изменять» (НИ), «Увеличить подачу» (УВ). Непрерывные кусочно-линейные аппроксимированные функции принадлежности нечетких множеств, задающих на абсолютной оси значения управляющих решений, изображены на рис. 4.17. Диапазон изменения количества подаваемого глинозема ограничен интервалом 75 – 375 кг. Максимальное значение функции принадлежности нечеткого множества «НИ» достигается на отметке 225 кг, что приблизительно соответствует количеству глинозема, загружаемого при обработке двух бортов промышленных электролизеров со средней

производительностью 1 т/сут и расходе глинозема 1,8 т на 1 т алюминия (из расчета восьми обработок за сутки).

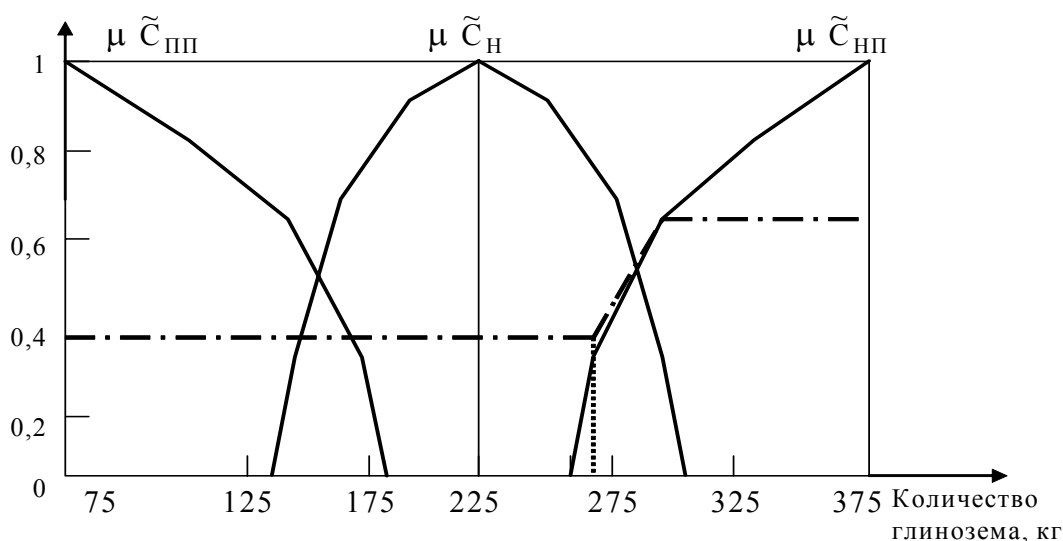


Рис. 4.17

Для каждого терма "УМ", "НИ", "УВ" лингвистической переменной "Подача глинозема" соответственно определены матрицы A_{UM} , M_{NI} и M_{UV} , описывающие силу воздействий соответствующих управляющих решений (рис. 4.18). Например, на рис. 4.18, в изображена матрица M_{UV} , характеризующая управляющее решение "УВ". Из анализа матрицы следует, что если объект управления имел значение "НП", то в результате управляющего воздействия он со степенью уверенности 0,2 будет иметь прежнее значение, со степенью 0,8 – значение "Н" и со степенью уверенности 0,4 – "ПП".

$$M_{UM} = \begin{array}{c|ccc} & \text{НП} & \text{Н} & \text{ПП} \\ \hline \text{НП} & 1 & 0 & 0 \\ \text{Н} & 0,8 & 0,3 & 0 \\ \text{ПП} & 0,8 & 1 & 0 \end{array}$$

а)

$$M_{NI} = \begin{array}{c|ccc} & \text{НП} & \text{Н} & \text{ПП} \\ \hline \text{НП} & 1 & 0 & 0 \\ \text{Н} & 0 & 1 & 0 \\ \text{ПП} & 0 & 0 & 1 \end{array}$$

б)

$$M_{UV} = \begin{array}{c|ccc} & \text{НП} & \text{Н} & \text{ПП} \\ \hline \text{НП} & 0,2 & 0,8 & 0,4 \\ \text{Н} & 0 & 0,3 & 0,8 \\ & 0 & 0 & 1 \end{array}$$

в)

Рис. 4.18

Ситуация, в которую желательно перевести объект, т.е. целевая ситуация, определяется исходя из анализа степеней предпочтения управляющих решений. Степени предпочтения последних зависят от вида конкретной ситуации и задаются как продукционная система "ситуация - предпочтение решений", которая содержит следующие продукции:

- 1) если лингвистическая переменная имеет значение "НП", то степени предпочтения управляющих решений составляют: $\alpha(\text{УМ}) = 0$; $\alpha(\text{НИ}) = 0,5$; $\alpha(\text{УВ}) = 1$;
- 2) если значение лингвистической переменной "Н", то $\alpha(\text{УМ}) = 0,4$; $\alpha(\text{НИ}) = 1$; $\alpha(\text{УВ}) = 0,4$;
- 3) если лингвистическая переменная имеет значение "ПП", то $\alpha(\text{УМ}) = 1$; $\alpha(\text{НИ}) = 0,2$; $\alpha(\text{УВ}) = 0$.

При этом степени предпочтения применения значений управляющих решений в конкретной ситуации определяются как конъюнкция степеней применимости соответствующих продукций, в качестве которых допускается использовать степени принадлежности значений признаков нечеткой ситуации, и заданных в них степеней предпочтения значений управляющих решений. Результирующие степени предпочтения применения значений управляющих решений принимаются равными максимальным среди соответствующих степеней предпочтения по каждому значению признаков "УМ", "НИ", "УВ".

Рассмотрим процедуру поиска управляющего решения на основе алгоритмов нечеткого ситуационного вывода, представленных в работе [25].

1. По рис. 4.16 идентифицируем текущую ситуацию, в которой находится объект. Например, имеет место ситуация $\tilde{s}_0 = \{ \langle 0,6/\text{Н} \rangle, \langle 0,4/\text{Н} \rangle \}$

2. Определяем степени предпочтения каждого управляющего решения "УМ", "НИ", "УВ". Они соответственно равны $\alpha(R_1, \tilde{s}_0) = 0,4$; $\alpha(R_2, \tilde{s}_0) = 0,5$; $\alpha(R_3, \tilde{s}_0) = 0,6$.

Выбираем управляющее решение R_3 "УВ" как имеющее наибольшую степень предпочтения.

3. Моделируем принятие выбранного управляющего решения R_3 . Для этого выполняем композицию нечетких значений признаков в ситуации \tilde{s}_0 и нечетких отношений, задающих силу воздействия управления R_3 (см. рис. 4.18, в). В результате получаем ситуацию $\tilde{s}_{03} = \{ \langle 0,2/\text{ПП} \rangle, \langle 0,6/\text{Н} \rangle, \langle 0,4/\text{ПП} \rangle \}$.

4. Находим величину требуемого управляющего воздействия. Сначала определяем нечеткое отношение M , композиция которого с множеством \tilde{s}_0 приводит к получению множества \tilde{s}_{03} . Для этого вычисляем декартово произведение $\tilde{s}_0 \times \tilde{s}_{03}$ и строим матрицу нечетких правяющих решений, которая представлена на рис. 4.19.

5. Раскладываем полученное управляющее решение в базисе $\{\text{УМ}, \text{НИ}, \text{УВ}\}$. Для этого необходимо определить нечеткое множество $R^J = \{ \langle \mu_{\tilde{R}^J}(\text{УМ})/\text{УМ} \rangle, \langle \mu_{\tilde{R}^J}(\text{НИ})/\text{НИ} \rangle, \langle \mu_{\tilde{R}^J}(\text{УВ})/\text{УВ} \rangle \}$. Для определения коэффициентов $\mu_{\tilde{R}^J}(\text{УМ})$, $\mu_{\tilde{R}^J}(\text{НИ})$, $\mu_{\tilde{R}^J}(\text{УВ})$ следует вычислить степени включения отношения ЛУ в отношения $M_{\text{УМ}}$, $M_{\text{НИ}}$, $M_{\text{УВ}}$. Управляющее решение в нечетком виде выглядит следующим образом: $R^J = \{ \langle 0,4/\text{УМ} \rangle, \langle 0,4/\text{НИ} \rangle, \langle 0,6/\text{УВ} \rangle \}$.

6. Определяем количественное значение управляющего решения. Для этого необходимо построить объединение конъюнкций нечетких множеств, задающих термы "УМ", "НИ", "УВ", со степенями принадлежности этих термов нечеткому множеству R^J и найти центр площади полученной фигуры (см. рис. 4.17).

$M' =$		НП	Н	ПП
	НП	0,2	0,8	0,4
	Н	0	0,3	0,8
		0	0	1

Рис. 4.19

Верхняя огибающая, соответствующая объединению полученных множеств, показана штрихпунктирной линией. Перпендикуляр, построенный в точке 260 кг, делит площадь фигуры пополам. Это означает, что значение 260 кг является количественным выражением нечеткого управляющего решения.

Следовательно, при возникновении рассмотренной в примере нечеткой ситуации надо увеличить количество загружаемого глинозема примерно до 260 кг. Реализация требуемого управляющего воздействия на электролизерах, для загрузки которых используются напольно-рельсовые машины, может быть осуществлена увеличением скорости подачи глинозема, например путем соответствующего изменения положения шиберной

заслонки бункера. Таким образом, вычисляемые при помощи данного алгоритма управляющие воздействия позволяют обеспечить более точное дозирование, ограниченное лишь принятым интервалом дискретности по загрузке, и, как следствие, плавное регулирование подачи глинозема в электролизную ванну в зависимости от конкретной ситуации в отличие от существующего сейчас достаточно грубого управления с двумя воздействиями ("Осуществить загрузку" или "Пропустить обработку").

4.4. Принципы адаптации в нечетком управлении

Сама идея нечеткого управления, воспроизводящая в определенной степени методы интеллектуального управления, может быть в неявном виде, предполагает возможность адаптации.

Уровень реализации этой возможности зависит и от сложности решения задач и от ресурсов, которые можно использовать.

Первый, наиболее простой вариант адаптации заключается в том, что в нечетком контроллере более конкретно в базе знаний, предусмотрен счетчик интенсивности использования правил нечеткого вывода. После некоторого количества циклов управления возможны перегруппировка правил в базе знаний, создание своего рода кэш-памяти, когда наиболее интенсивно используемые правила размещаются в области с меньшим временем доступа. Правила, интенсивность использования которых минимальна, наоборот, помещаются в область с большим временем доступа. Возможно также и удаление правил из базы знаний.

Второй вариант адаптации заключается в том, что в случае возникновения ситуации, для которой отсутствует напрямую связанное с ней правило нечеткого вывода, система делает попытку либо найти ближайшее правило или комбинацию правил и на их основе выполнить вывод, либо сформулировать новое правило, ранее отсутствующее в системе.

Одним из вариантов решения этой задачи может быть следующий.

Пусть ситуация по переменным, характеризующим систему в некоторый момент времени, определяется нечеткими множествами

$$\tilde{A} = \{\mu_A(x)/x\}, \tilde{B} = \{\mu_B(y)/y\}, \dots, W = \{\mu_W(z)/z\}.$$

В базе знаний системы нет правила, в условной части которого находится этот набор полностью. Однако находятся правила, содержащие отдельные фрагменты этого набора. Естественным выбором является то правило, которое содержит фрагмент, наиболее близкий к рассматриваемому.

мой комбинации. Степень близости можно, например, оценить по величине взвешенной мощности свертки нечетких множеств.

Возможен и другой подход. В системе нечеткого управления для переменной, определяющей управляющее воздействие, определено множество ее лингвистических значений L_y и соответствующие нечеткие множества:

$$L_y = \{ \mu_{L1}(\xi)/\xi, \mu_{L2}(\xi)/\xi, \dots \}.$$

Для неизвестного набора вычисляется свертка соответствующих нечетких множеств и определяется импликация:

$$\tilde{S} \longrightarrow L_i \quad i = 1, 2, \dots,$$

где \tilde{S} – свертка нечетких множеств, входящих в неизвестный набор.

Импликация, в свою очередь, является нечетким множеством, и в качестве наиболее соответствующей данной ситуации выбирается та, для которой максимально значение взвешенной мощности.

К недостаткам рассмотренных методов следует отнести то, что их реализация занимает большое время, т.е. не может быть обеспечена в реальном масштабе времени.

В ряде случаев, например при управлении электроприводом, может быть использована адаптация с эталонной моделью (рис. 4.20).



Рис. 4.20

На рис. 4.20 обозначены:

$U = g + Z$ – управляющее содействие;

X_1, X_2 – выходные координаты объекта управления;

g – задающее воздействие;

Z – выходная скалярная переменная нечеткого контроллера;

X_{1M}, X_{2M} – выходные координаты эталонной модели;

$e_1 = X_1 - X_{1M}, e_2 = X_2 - X_{2M}$ – сигналы ошибки;

E_1, E_2 – выходные координаты фазификатора.

Нечеткий контроллер вырабатывает сигнал адаптации в зависимости от величины расхождения между выходными координатами объекта управления и эталонной модели.

В зависимости от особенностей объекта управления могут использоваться эталонная модель традиционного типа либо нечеткая модель.

Адаптация нечеткого контроллера может заключаться в изменении правых частей тех правил базы правил, которые приводят к низкому качеству управления.

Общая идея адаптивных систем заключается во введении в систему управления дополнительного контура адаптации, который обеспечивает коррекцию закона управления. Адаптация с коррекцией правил заключается в следующем. Адаптивный нечеткий регулятор (рис. 4.21) оценивает информацию об изменении параметров объекта управления по значениям ошибки управления и ее производной dE/dt . Контур адаптации формирует корректирующее воздействие, изменяя правые части правил нечеткого

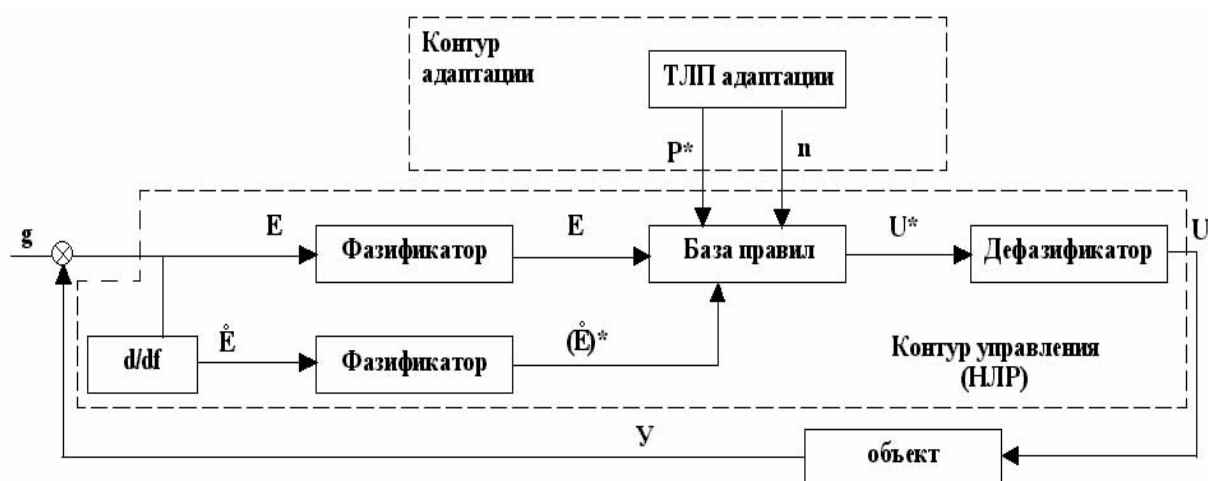


Рис. 4.21

управления. Таблица лингвистических правил (ТЛП) адаптации использует информацию о желаемом отклике системы в виде таблицы соответствий. Некоторые ситуации (E^* , dE^*/dt) воспринимаются как соответствующие нормальному протеканию переходного процесса, другие воспринимаются как требующие коррекции. Коррекция правил заключается в изменении не текущего правила, а предыдущего правила, которое и создало неудовлетворительную текущую ситуацию.

Адаптация с коррекцией правил имеет ряд недостатков. Отдельное правило нечеткого управления описывает некоторую локальную область управления. Если некоторое правило скорректировано, то работа нечеткого регулятора не обязательно должна улучшаться, поскольку может потребоваться коррекция других правил. Если же корректируются несколько правил одно за другим, то сложно оценить результат коррекции, особенно если для дефазификации используется метод центра тяжести. Это говорит о том, что структура, представленная на рис. 4.21, полезна, скорее всего, при первоначальном обучении нечеткого регулятора и для достаточно простых объектов управления, так как зависимость ошибки от сигнала управления может носить сложный и неоднозначный характер. К тому же поведение системы при такой структуре может быть улучшено лишь в будущем при повторном возникновении неудовлетворительной ситуации.

Адаптация с выбором баз знаний предполагает использование различных наборов правил нечеткого управления для различных ситуаций.

Отдельная база правил задает трехмерную (E^* , dE^*/dt , U^*) поверхность управления. Меняя параметры управления управляемого объекта и обучая каждый раз нечеткий регулятор, можно построить несколько таких поверхностей, которые могут быть близки между собой или сильно отличаться. Пересечения различных поверхностей могут соответствовать инвариантным правилам управления. При выполнении гипотезы о квазистационарности объекта управления (обычной при постановке задач адаптивного управления) нужно осуществить выбор между некоторым количеством баз знаний, соответствующих существенно различным параметрам объекта. Практически в такой системе используется несколько нечетких регуляторов со своими базами знаний, которые адаптированы для существенно различных параметров объектов. В процессе управления заранее неизвестно, какая база знаний является наиболее подходящей, поэтому входная ситуация (E^* , dE^*/dt) подается одновременно на все нечеткие ре-

142

гуляторы. Затем формируется интегральный управляющий сигнал, при этом предусматривается необходимость подавления сигнала с нечеткого регулятора, база знаний которого не соответствует текущим параметрам объекта.

Контрольные вопросы

1. Как организован механизм вывода в нечеткой системе управления доменной печью?
2. Как организуется база знаний в нечеткой системе управления доменной печью?
3. Что характеризует обобщенная функция принадлежности?
4. Чем можно объяснить перспективность нечетких систем управления в автомобилях?
5. Какие основные варианты адаптации используются в нечетких системах управления?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. «Разумная» коробка передач // Авто. – 1996. – № 5 – 6. – С. 44.
2. *Алиев Р.А., Ульянов С.В.* Нечеткие алгоритмы и системы управления. – М.: Знание, 1990. – 45 с.
3. *Алиев Р.А., Церковный А.Э., Мамедова Г.А.* Управление производством при нечеткой исходной информации. – М.: Энергоатомиздат, 1991. – 240 с.
4. *Анисимов В.Ю., Борисов Э.В.* Методы и устройства преобразования нечетко определенных параметров при проектировании радиотехнических систем // Изв. вузов. Радиоэлектроника. – 1985. – № 4. – С. 30.
5. *Белкин А.Р.* Нечеткая классификация на основе лингвистических переменных и задачи дифференциальной диагностики // Вопросы кибернетики. Принятие решений и анализ экспертной информации. – М.: Наука, 1989. – С. 129 – 133.
6. *Берг А.Р., Бирюков Б.В., Геллер Е.С.* и др. Управление, информация, интеллект. – М.: Мысль, 1976. – 383 с.
7. *Борисов А.Н., Крумберг О.А., Федоров И.П.* Принятие решений на основе нечетких моделей. – Рига: Зинатне, 1980. – 184 с.
8. ConCept for Quantum User Manual.-DOK-E-No.424704 980. Block Library Fuzzy Control. AEG Schneider Automation, 1995. – 305 с.
9. *Бураков М.В.* Механизм адаптации нечеткого регулятора // Известия АН. Теория и системы управления. – 1998. – № 1. – С. 84 – 87.
10. *Борцов Ю.А., Поляков П.Д., Соколов П.В.* Синтез адаптивного нечеткого регулятора электропривода // Приборы и системы управления. – 1993. – № 3. – С. 27 – 29.
11. *Глотов В.А., Павельев В.В.* Экспертные методы определения весовых коэффициентов // АиТ. – 1976. – № 12. – С. 95 – 108.
12. *Гриняев С.* Нечеткая логика в системах управления // Компьютера, #38(415). – 2001. – С. 20 – 26.
13. *Голованов Л., Сорокин К.* Последний аргумент " автомата" // Авторе-
вью. – 1997. – № 24. – С. 24 – 25.

14. *Деменков Н.П.* Использование пакета ConСерт для нечеткого управления работой парового котла // Промышленные АСУ и контроллеры. – 1999. – № 7. – С. 20 – 22.
15. *Деменков Н.П.* Нечеткое управление в системе Трейс Моуд // Промышленные АСУ и контроллеры. – 1999. – № 5. – С. 26 – 28.
16. *Заде Л.А.* Основы нового подхода к анализу сложных систем и процессов принятия решений // Математика сегодня. – М.: Знание, 1974. – С. 5 – 48.
17. *Захаров В.Н., Ульянов С.В.* Нечеткие модели интеллектуальных промышленных регуляторов и системы управления // Известия АН СССР. Техническая кибернетика. – 1993. – № 4. – С. 189 – 205.
18. *Борисов Э.В., Анисимов В.Ю.* Алгоритмы выбора порога наблюдения при нечетких уровнях сигнала и шума // Изв. вузов. Радиотехника. – 1984. – № 1. – С. 82.
19. *Кафаров В.В., Дорохов И.Н., Марков Е.П.* Системный анализ процессов химической технологии. Применение метода нечетких множеств. – М.: Наука, 1985. – 531 с.
20. *Кофман А.* Введение в теорию нечетких множеств. – М.: Радио и связь, 1983. – 432 с.
21. *Кузьмин В.Б., Травкин С.И.* Теория нечетких множеств в задачах управления и принципах устройства нечетких процессоров // Изв. АН СССР. Техническая кибернетика. – 1992. – № 5. – С. 171 – 197.
22. *Леонов С.* Третьего не дано // Компьютера #38(415). – 2001. – С. 18 – 19.
23. *Матлаб 5.0.* Руководство пользователя. – М.: ДМК, 2001. – 405 с.
24. *Mamdani E.H.* Applications of fuzzy logic to approximate reasoning using linguistic systems // IEEE Trans.Comput. C-26. – 1977. – P. 1182 – 1191.
25. *Мелихов А.Н., Бернштейн Л.С., Коровкин С.Я.* Ситуационные советующие системы с нечеткой логикой. – М.: Наука, 1990. – 272 с.
26. *Нечеткие множества в моделях управления и искусственного интеллекта* /Под ред. Д.А. Поспелова. – М.: Наука, 1986. – 312 с.
27. *Нечеткие множества и теория возможностей. Последние достижения: Пер. с англ./ Под ред. Р.Р. Ягера.* – М.: Радио и связь, 1986. – 408 с.

28. *Никольский С.* Нечетко едешь – дальше будешь // *Компьютера* #38(415). – 2001. – С. 27 – 28.
29. *Обработка нечеткой информации в системах принятия решений / А.Н. Борисов и др.* – М.: Радио и связь, 1989. – 304 с.
30. *Петров Б.Н., Гольденблат И.И., Уланов Г.М. и др.* Теория моделей в процессе управления: Термодинамические аспекты. – М.: Наука, 1978. – 130 с.
31. *Прикладные нечеткие системы: Пер. с япон. / К. Асаи, Д. Вамада, С. Иваи и др.; Под ред. К. Тэрано, К. Асаи, М. Сугэно.* – М.: Мир, 1993. – 368 с.
32. Пат. 1654381 РФ. Способ управления серией алюминиевых электролизеров / Шапировский М.Р. и др. // *Изобретения.* – 1991. – № 21.
33. *Трейс М.* Графическая инструментальная система для разработки АСУ. Версия 5.0: Руководство пользователя. – М.: Бином, 1998. – 300 с.
34. *Ульянов С.В.* Нечеткие модели интеллектуальных систем управления: теоретические и прикладные аспекты (обзор) // *Изв. АН СССР. Техническая кибернетика.* – 1991. – № 3. С. 3 – 28.
35. *Чернов В.Г.* Применение теории нечетких множеств для управления углом опережения зажигания ДВС // *Системный анализ, моделирование и управление сложными процессами на базе ЭВМ, НТК.* – Ташкент: ТПИ, 1993. – С. 47 – 50.
36. *Чернов В.Г., Какушкин М.Ю.* Методы управления процессами подачи топлива и его сгорания, базирующиеся на теории нечетких множеств // *Микроэлектроника и информатика.* – М.: МИЭТ, 1998. – С. 59 – 60.
37. *Чернов В.Г.* Нечеткие множества в задачах управления и принятия решений: Текст лекций / Владим. гос. ун-т. – Владимир, 1999. – 88 с.
38. *Шапировский М.Р., Меликянц Р.В., Сириченко А.В.* Применение нечеткой математической логики в алгоритмах ситуационного управления концентрационным режимом алюминиевых электролизеров // *Приборы и системы, управление контроль, диагностика.* – 2000. – № 2. – С. 20 – 23.

Учебное издание

ЧЕРНОВ Владимир Георгиевич

НЕЧЕТКИЕ КОНТРОЛЛЕРЫ. ОСНОВЫ ТЕОРИИ И ПОСТРОЕНИЯ

Учебное пособие по курсу
«Интеллектуальные системы управления»

Редактор Р.С. Кузина
Корректор В.В. Гурова
Компьютерная верстка Е.Г. Радченко

ЛР № 020275. Подписано в печать 27.06.03.
Формат 60x84/16. Бумага для множит. техники. Гарнитура Таймс.
Печать офсетная. Усл. печ. л. . Уч.-изд. л. . Тираж 70 экз.

Заказ

Редакционно-издательский комплекс
Владимирского государственного университета.
600000, Владимир, ул. Горького, 87.