
В. В. Борисов
В. В. Круглов
А. С. Федулов

Нечеткие модели и сети

2-е издание

Москва
Горячая линия – Телеком
2018

ББК 30.17
УДК 681.322
Б82

Р е ц е н з е н т ы : доктор техн. наук, профессор *В. Н. Вагин*,
доктор техн. наук, профессор *А. И. Галушкин*

Борисов В. В., Круглов В. В., Федулов А. С.

Б82 Нечеткие модели и сети. – 2-е изд., стереотип. – М.:Горячая линия–Телеком, 2018. – 284 с.: ил.
ISBN 978-5-9912-0283-1.

Книга посвящена исследованию нечетких моделей, используемых для описания, анализа и моделирования сложных слабо формализуемых систем и процессов, а также вопросам построения, обучения и использования, во-первых, нечетких нейронных сетей, реализующих нечеткие производственные, реляционные и функциональные модели, во-вторых, нейронных нечетких сетей, характеризующихся введением нечеткости в различные компоненты традиционных нейронных сетей. Рассмотрены основные разновидности нечетких моделей систем и процессов, отображаемых структурами на основе графов. Особое внимание уделено анализу способов построения, моделирования и использования нечетких когнитивных карт, реализующих расширенные возможности по анализу и моделированию сложных систем.

Для специалистов в области информатики, вычислительной техники, анализа и моделирования сложных систем и процессов, занимающихся созданием и использованием интеллектуальных систем, а также для аспирантов и студентов соответствующих специальностей.

ББК 30.17

Адрес издательства в Интернет WWW.TECHBOOK.RU

Научное издание

Борисов Вадим Владимирович
Круглов Владимир Васильевич
Федулов Александр Сергеевич

НЕЧЕТКИЕ МОДЕЛИ И СЕТИ

2-е издание, стереотипное

Все права защищены.

Любая часть этого издания не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения правообладателя

© ООО «Научно-техническое издательство «Горячая линия – Телеком»
www.techbook.ru

© В. В. Борисов, В. В. Круглов, Федулов А. С., 2007, 2018

ВВЕДЕНИЕ

Нечеткое моделирование не является альтернативой различным подходам к моделированию сложных систем и процессов, а прежде всего предоставляет эффективные методы и средства для их изучения в следующих основных сферах применения.

Во-первых, недостаточность или неопределенность знаний об исследуемой системе, когда получение требуемой информации является сложной, трудоемкой, дорогостоящей или вовсе невозможной задачей. При этом четкие модели не могут быть получены либо они являются слишком сложными для практического использования. Значимая же часть информации об этих системах доступна в виде экспертных данных или в эвристическом описании процессов функционирования. Эта информация может быть нечеткой и недостаточно определенной для того, чтобы быть выраженной математическими зависимостями. Кроме того, информация о системе может быть разнокачественной, а оценка значений параметров – проводиться с помощью различных шкал. Однако зачастую возможно описать функционирование этих систем в виде эвристических предпочтений, используя конструкции естественного языка в форме нечетких правил или отношений различного типа. Другой аспект неопределенности знаний о системе связан с неясностью или нечеткостью выделения и описания границы системы или отдельных ее состояний, а также входных и выходных воздействий.

Во-вторых, адекватная обработка неопределенной информации, если параметры и входные данные не являются точными и корректно представленными. Часто традиционные методы не только не позволяют адекватно обработать данные, но также не позволяют учесть естественно присущую этим данным неопределенность. Нечеткая логика и теория нечетких множеств являются одним из эффективных подходов к решению данной проблемы.

В-третьих, «прозрачное» моделирование и идентификация реальных систем, которые являются нелинейными в своей основе и не могут быть представлены моделями, использующими существующие методы идентификации. В последнее время серьезное внимание уделяется развитию методов идентификации нелинейных систем на основе экспериментальных данных. Однако, сравнивая нечеткие модели с другими известными методами, например с искусственными нейронными сетями, можно отметить их большую прозрачность, которая возможна благодаря их лингвистической интерпретации в виде нечетких производственных правил.

Логическая структура этих правил способствует пониманию и анализу системы количественно-качественными методами.

В зависимости от назначения постулаты и положения теории нечетких множеств и нечеткой логики в нечетких моделях могут использоваться: непосредственно при описании системы, при задании параметров системы, при задании входов, выходов и состояний системы. Основные трудности при использовании нечетких моделей для решения практических задач связаны, как правило, с априорным определением компонентов этих моделей (нечетких высказываний, функций принадлежности для каждого значения лингвистических переменных, структуры базы нечетких правил и др.). Поскольку эти компоненты зачастую выбираются субъективно, они могут быть не вполне адекватны моделируемой системе или процессу.

Основное же преимущество нейросетевого подхода – возможность выявления закономерностей в данных, их обобщение, т. е. извлечение знаний из данных, а основной недостаток – невозможность непосредственно (в явном виде, а не в виде вектора весовых коэффициентов межнейронных связей) представить функциональную зависимость между входом и выходом исследуемого объекта. Другим недостатком нейросетевого подхода является трудность формирования представительной выборки, большое число циклов обучения и забывание «старого» опыта, сложность определения размера и структуры нейронной сети.

Подходы к исследованию сложных систем на основе нечетких и нейросетевых моделей взаимно дополняют друг друга, поэтому целесообразна их интеграция на основе принципа «мягких» вычислений (Soft Calculation). Основы построения таких моделей сводятся к следующему: терпимость к нечеткости и частичной истинности используемых данных для достижения интерпретируемости, гибкости и низкой стоимости решений.

В части I книги выделено три основных класса нечетких моделей, доминирующих в рамках нечеткого подхода к анализу и моделированию сложных систем, а также различные классы нечетких сетей (и способы их интеграции с искусственными нейронными сетями), используемых для построения и анализа этих моделей.

В части II рассмотрены нечеткие продукционные модели, являющиеся наиболее общим видом нечетких моделей, используемых для описания, анализа и моделирования сложных систем и процессов. Проанализированы и классифицированы основные компоненты, определяющие создание и применение нечетких продукционных моделей. Рассмотрены получившие наибольшее распространение алгоритмы нечеткого вывода.

Представлены нечеткие реляционные модели, а также проиллюстрировано их подобие нечетким производственным моделям.

В части III предложена классификация нечетких нейронных производственных сетей и рассмотрены различные способы и средства интеграции нечетких производственных моделей с нейронными сетями, в которых нейросетевая технология используется в качестве инструмента для реализации компонентов аддитивных нечетких производственных моделей.

В части IV предложена классификация и рассмотрены различные типы нейронных нечетких сетей, характеризующиеся введением нечеткости в различные компоненты традиционных нейронных сетей (в элементы структуры и механизмы обучения и функционирования) на основе положений теории нечетких множеств и нечеткой логики.

В части V книги рассмотрены основные разновидности нечетких моделей систем и процессов, отображаемых некоторыми структурами на основе графов (логико-временными последовательностями, пространственной распределенностью, функциональной взаимосвязанностью, причинно-следственными отношениями): нечеткие автоматы, нечеткие сети Петри, нечеткие ситуационные сети и нечеткие когнитивные карты. Особое внимание уделено анализу способов построения, моделирования и использования нечетких когнитивных карт, реализующих расширенные возможности по анализу и моделированию сложных систем.

Часть I

КЛАССИФИКАЦИЯ СИСТЕМ И НЕЧЕТКИХ МОДЕЛЕЙ

Глава 1. ПОНЯТИЕ СИСТЕМЫ. КЛАССИФИКАЦИЯ И ХАРАКТЕРИСТИКА СИСТЕМ

1.1. Понятие и определения системы

Понятие системы является базовым в различных концепциях теории систем, системного подхода, системологии, прикладных исследований. Разнообразие определений системы обусловливается:

- различными концепциями теории систем и вариантами системного подхода, отличающимися по составу и содержанию используемых понятий и принципов;
- ориентацией как на разные типы, так и на различное назначение систем.

Все многообразие подходов к определению понятия «система» (их известно более 40) можно разделить на следующие группы.

Первую группу составляют определения системы как выбираемой исследователем любой совокупности переменных, свойств или сущностей [1]. Если следовать этим определениям, то системой могут оказаться два любых произвольно выбранных объекта, вовсе не имеющих или имеющих в действительности настолько слабые взаимосвязи, что они могут быть либо не установлены, либо ими можно пренебречь.

Вторая группа определений базируется на понимании системы как множества элементов, связанных между собой. Однако определение системы через понятие множества допускает возможность различных, в том числе произвольных, ее разбиений на подмножества элементов, каждое из которых также является множеством [2].

Третью группу составляют определения системы, связывающие ее с целенаправленной активностью. Здесь система определяется как относительно устойчивая организованная совокупность взаимодействующих и взаимосвязанных элементов, а также комплекс средств достижения об-

щей цели. Развитие и совершенствование такой системы зависит от взаимодействия с окружающей средой [3].

Помимо развития и совершенствования в системах могут происходить процессы деградации и разрушения, которые зависят не только от взаимодействия систем с окружающей средой, но и от внутренних свойств самой системы. Однако ни одна из вышеуказанных групп определений явно не учитывает этого.

Четвертую группу составляют наиболее общие определения системы как комплекса элементов, находящихся во взаимодействии. Однако в этом случае к категории «система» могут быть отнесены любые, даже очень слабо взаимодействующие объекты, которые рассматриваются с системных позиций [4].

Пятая группа определений характеризует систему через указание признаков, которыми должен обладать объект, чтобы его можно было отнести к категории «система». Данные признаки вводятся через понятия совокупности, взаимосвязи и целого [5]. Здесь под *системой* понимается совокупность элементов и процессов (называемых компонентами), находящихся в отношениях и взаимосвязях между собой, образующих единое целое и характеризующихся интегративным, или системным, свойством, отличающим данную совокупность от среды и приобщающим к этому свойству каждый из ее компонентов.

В дальнейшем в данной книге в качестве определения системы будем руководствоваться определением из пятой группы.

1.2. Классификация систем

В настоящее время существует множество классификаций систем по различным признакам, наиболее общим из которых является классификация по происхождению.

По *происхождению* системы делятся на естественные (существующие в объективной действительности: живые; неживые экологические, социальные и др.); концептуальные или идеальные (продукт человеческого мышления: знания, теории, гипотезы и др.); искусственные (созданные человеком: орудия, механизмы, машины, роботы и др.); смешанные, т. е. объединяющие искусственные и естественные подсистемы: эргономические, биотехнические, автоматизированные, организационно-технические (в которых совместно функционируют человеческие коллективы и технические устройства) и др. [6, 7].

С одной стороны, концептуальные системы можно отнести к искусственным, исходя из того, что они созданы человеком. С другой стороны, искусственными могут быть и системы, созданные другими организа-

циями. Следовательно, справедливо разделение систем *по форме их существования*, независимо от того, кто их создал, на идеальные (абстрактные) и реальные.

В зависимости от целевого предназначения системы могут делиться на ценностно-ориентированные и целеориентированные (целенаправленные). Ценностно-ориентированным системам внутренняя цель неприсуща. Функционирование таких систем не подчиняется четкой цели, отсутствует иерархия уровней управления. Целевая функция для этих систем задается извне задачами использования системы, а процесс их функционирования оценивается по некоторым критериям ценностей [8]. В целеориентированных системах, совокупность целей которых связана с иерархией уровней управления, основой функционирования и развития являются факторы целесообразности и целеполагания [9, 10].

Целеориентированные системы могут быть представлены общей схемой управления (рис. 1.1), состоящей из части системы *SYS*, подлежащей управлению *U*, и управляющей части системы, которая это управление вырабатывает. При этом для выработки управления *U* управляющей частью требуется модель системы.

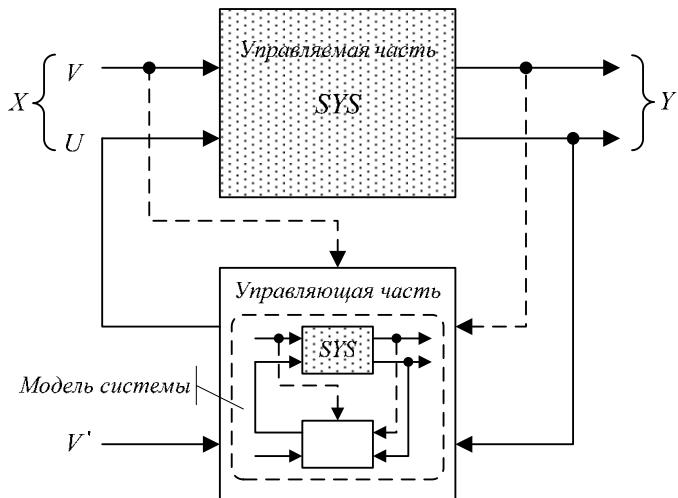


Рис. 1.1. Общая схема управления системой: *U* – управляющие воздействия, *V*, *V'* – неконтролируемые воздействия

В зависимости от модели системы и степени ее соответствия реальной системе можно предложить следующие классификационные призна-

ки систем: по описанию переменных, по типу связи между входными и выходными переменными, по типу управления.

По *типу переменных* системы делятся на системы с количественным, качественным и количественно-качественным их описанием. Системы с количественными переменными допускают их дискретное и непрерывное описание, а также смешанный случай. Системы с качественным описанием переменных допускают представление как средствами естественного языка, так и на основе более глубокой формализации. Для систем с количественно-качественным описанием могут использоваться любые из перечисленных выше представлений. Причем все эти описания могут носить детерминированный, стохастический, нечеткий или смешанный характер.

Классификация по *типу управления* определяется тем, входит ли управляющая часть в систему или является внешней по отношению к ней. В соответствии с этим могут быть выделены, во-первых, системы, управляемые извне, во-вторых, самоуправляемые системы и, в-третьих, системы, управление которыми частично осуществляется извне, а частично – изнутри самой системы.

В соответствии со степенью известности траектории (стратегии), приводящей систему к цели и возможности управляющей части удерживать управляемую часть системы на этой траектории (вне зависимости от того, включена ли управляющая часть в систему или нет), можно выделить *четыре основных способа управления*: без обратной связи (или программное управление), регулирование (или автоматическое регулирование), по параметрам (параметрическая адаптация), по структуре (или структурная адаптация).

Примечание. В скобках приведены названия способов управления для случая самоуправляемых систем.

Первый способ применяется, когда точно известна нужная траектория и, следовательно, известно правильное управление $u_0(t)$.

Второй способ имеет место, когда неконтролируемые воздействия $v_0(t)$ отличаются от ранее предполагаемых и наблюдается разница между текущей $y(t)$ и нужной траекториями $y_0(t)$, которую необходимо устранить.

Третий способ управления характеризуются тем, что траекторию, приводящую систему к цели, принципиально невозможно задать. И управление состоит в такой подстройке параметров системы, обеспечивающей пересечение траекторией целевой области.

Для четвертого способа характерно то, что целевая область не будет достигнута ни при каких возможных комбинациях значений управляемых

параметров. В этом случае необходимо изменять структуру системы в поисках такой, при которой возможно попадание в целевую область. Интересным развитием этого способа является управление (адаптация) по целям для случая принципиальной недостижимости ранее поставленной цели при всевозможных преобразованиях структуры (ресурсов) системы [6, 11].

Как было ранее сказано, нужное управление U системой отыскивается с помощью отбора среди возможных управлений путем сравнения по каким-либо критериям последствий каждого из них. Определить эти последствия и сравнить их можно, если в управляющей части имеется модель системы. Учет этого факта приводит еще к двум классификациям систем.

По ресурсной обеспеченности моделирования системы делятся на малые и большие. Под большими понимаются системы, моделирование которых затруднительно вследствие их размерности, а также наличия компонентов, не поддающихся точному и подробному описанию.

По достаточности информации для моделирования системы можно разделить на простые и сложные. Сложной системой называется система, состоящая из подсистем, являющихся, в свою очередь, простыми системами, и в модели которой не хватает информации для эффективного управления. Степень сложности больше зависит от разнообразия связей и элементов, чем от их количества. При этом сложной является система, обладающая определенным набором свойств [12], например большим числом неоднородных элементов, эмерджентностью, иерархией, агрегированием параметров, многофункциональностью, гибкостью, адаптацией, надежностью, безопасностью, стойкостью, уязвимостью, живучестью.

Глава 2. КЛАССИФИКАЦИЯ МОДЕЛЕЙ СИСТЕМ

Понятие системы может быть уточнено до различной степени формализации в виде поэтапной эволюции и конструктивного развития ее моделей, включающих в нее по мере необходимости дополнительные сведения.

Модель системы, называемая моделью «черного ящика», отражает два ее важных свойства: *целостность* и *обособленность* (но не изолированность) от среды. Данная модель характеризуется заданным набором параметров, среди которых имеются параметры, влияющие на целевое свойство системы, но, какие из них являются определяющими (информационными) и какой математической моделью описываются закономерности их влияния на целевую характеристику, неизвестно. Единственным

источником информации для решения такой задачи служат данные типа «вход-выход» или «стимул-реакция» с описанием входных и выходных параметров наблюдаемой системы. А выбор параметров делается путем проверки разных эмпирических гипотез на материале разнокачественных данных. Причем предполагается, что изучаемый процесс может быть «аппроксимирован». А задача моделирования сводится к постулированию соответствующей структуры аппроксиматора («черного ящика») для того, чтобы корректно отразить динамические свойства и нелинейность системы. Структура данной модели едва ли отражает структуру реальной системы. Проблема идентификации системы в таком случае состоит в оценке параметров модели.

Существенными препятствиями в использовании этих моделей являются следующие. Во-первых, при построении модели системы в виде «черного ящика» существует опасность неполноты перечня входов и выходов, вследствие чего важные из них могут быть сочтены несущественными либо быть неизвестны. Во-вторых, структуру и параметры этих моделей сложно проинтерпретировать. Они не могут быть использованы для анализа поведения системы каким-либо иным путем, кроме как численной имитацией, не могут быть промасштабированы при изменении масштаба исследуемого процесса.

В рамках системного моделирования созданы методы и средства, позволяющие выполнить дальнейшую структуризацию этих наиболее общих моделей системы. Так, при более детальном рассмотрении система может быть разбита на части: неделимые (элементы) и состоящие более чем из одного элемента (подсистемы). В результате описания этих подсистем и элементов может быть сформирована *модель состава системы*. Главная трудность в построении модели состава системы заключается в том, что границы разбиения модели состава системы на подсистемы и элементы определяются целями системы и являются относительными, условными [6].

Следующим шагом в развитии модели системы является *модель структуры системы*, отображающая связи (отношения) между компонентами модели ее состава. Эти отношения могут быть самыми разнообразными (причинности, подобия, сходства, включения, подчиненности и др.)

Все вместе модели «черного ящика», состава и структуры системы образуют еще одну модель системы, которую называют *структурной схемой системы*. В ней указываются все элементы системы, все связи между элементами внутри системы и связи определенных элементов с окружающей средой (входы и выходы системы). Структурная схема системы является наиболее подробной и полной моделью любой систем-

мы. Структурная схема может быть представлена графическим отображением, в виде теоретико-множественных описаний, в виде матриц, графов и с помощью других языков моделирования структур. Данный подход к моделированию, заключающийся в полном понимании системной природы, обычно называется моделированием по методу «белого ящика».

Однако одной структурной информации для ряда исследований недостаточно, и акцент в этом случае делается на рассмотрении конкретных функциональных связей между входными, внутренними и выходными переменными системы [6, 9].

Все рассмотренные выше модели («черного ящика», состава, структуры, структурной схемы) системы являются *статическими моделями*.

Системы, в которых происходят любые изменения со временем, называют динамическими системами, а модели, отражающие эти изменения, – *динамическими моделями* систем. Для разных систем разработано большое число динамических моделей. И развитие этих моделей происходит приблизительно в той же последовательности, в которой это имеет место и для статических моделей.

Так, уже на этапе описания модели системы в виде «черного ящика» различают два типа динамики системы: функционирование и развитие. Под *функционированием* понимают процессы в системе, стабильно реализующей фиксированные цели. *Развитием* называют то, что происходит с системой (изменение структуры, а иногда и состава системы) при изменении ее целей.

Следующий шаг в построении динамических моделей заключается в различии взаимосвязанных этапов процесса (функционирования и/или развития). Для модели «черного ящика» в динамике – это указание последовательности начальных (и конечных) состояний системы; для модели состава в динамике – это перечень этапов в некоторой упорядоченной последовательности действий; динамический вариант структурной схемы системы – это подробное описание происходящего или планируемого процесса (например, сетевые графики).

Рассмотрим те же типы моделей («черного ящика», состава, структуры, структурной схемы) при более глубокой формализации их динамики.

Для модели системы в виде «черного ящика» в динамике выход $y(t)$ системы (может быть вектором) является реакцией на управляемые $u(t)$ и неуправляемые $v(t)$ входы $x(t) = \{u(t), v(t)\}$, выражаемой совокупностью двух процессов:

$$X^T = \{x(t)\} \text{ и } Y^T = \{y(t)\}, t \in T. \quad (2.1)$$

Модель «черного ящика» в динамике предполагает, что преобразование F , при котором $y(t) = F(x(t))$, неизвестно.

При наличии же динамического варианта структурной схемы системы соответствие между входом и выходом можно описать тем или иным способом. Способ же описания зависит от наших знаний и формы их использования.

Задача восстановления неизвестной функции $y(t) = F(x(t))$ может решаться как самостоятельно, так и являться, по существу, задачей перехода от модели «черного ящика» к структурной схеме системы в динамике, в соответствии со следующими этапами:

- модель типа «черного ящика» – связь между входными и выходными параметрами считается вообще неизвестной;
- непараметризованная модель – информация о связи входных и выходных параметров является априорной и настолько общей, что нельзя сделать конкретных выводов о функциональном виде этой зависимости;
- параметризованная модель – в явной форме известна зависимость $y(t) = F(x(t))$ с точностью до конечного числа параметров $F = (F_1, \dots, F_k)$;
- структурная схема модели, в которой параметры системы заданы точно.

И даже в такой постановке задача восстановления неизвестной функции $y(t) = F(x(t))$ совсем не тривиальна, например:

- в параметризованном случае неизвестны параметры функции F ;
- в непараметризованном – вид функции F неизвестен;
- неизвестны сведения о свойствах функции F (непрерывности, гладкости, монотонности, симметричности и др.);
- входы и выходы наблюдаются с помехами или искажениями;
- необходимо учитывать состояние системы в предыдущие моменты;
- система функционирует в условиях неопределенности.

Наибольшая общность определения динамической модели системы достигается введением в ее нотацию понятия *состояния системы*, как некоторой внутренней характеристики системы. То есть существует такое отображение $\eta: S \times T \rightarrow Y$, что

$$y(t) = \eta(t, s(t)), t \in T, \quad (2.2)$$

где $s(t)$ – состояние системы в момент t .

Данная зависимость показывает возможность изменения зависимости выхода системы от ее состояния с течением времени.

Для завершения построения модели нужно ввести семейство отображений $\sigma: S \times X(\bullet) \rightarrow S$, заданных для всех значений параметров $t \in T, \tau \in T, \tau \leq t$. То есть состояние системы в любой момент $t > \tau$ однозначно определяется состоянием s_τ в момент τ и «отрезком реализации» входа $x(\bullet)$ от τ до t :

$$s(t) = \sigma(t, \tau, s_\tau, x(\bullet)). \quad (2.3)$$

Таким образом, наиболее общая динамическая модель системы задает множества входов, состояний и выходов, а также связи между ними:

$$X \xrightarrow{\sigma} S \xrightarrow{\eta} Y. \quad (2.4)$$

Конкретизируя множества X, Y и S и отображения σ и η , можно перейти к примерам моделей различных систем. Например, в зависимости от того, дискретно или непрерывно множество T , модели являются *дискретными* или *непрерывными по времени*. Если же множества X, Y и S – дискретной во времени системы имеют конечное число элементов, то такую модель называют *конечным автоматом*. Если множества X, Y и S – линейные пространства, а σ и η – линейные операторы, то и модель системы называется *линейной*. Если для пространств линейной системы заданы топологические структуры (определенна метрика и сходимость последовательностей), а σ и η непрерывны в этой топологии, то модели являются *гладкими*. Для этого класса моделей систем отображение σ является общим решением дифференциального уравнения

$$\frac{ds}{dt} = f(t, s, x), \quad (2.5)$$

а для дискретных моделей систем – общим решением уравнения

$$s(t_{k+1}) = s(t_k, s, x) = \sigma(t_{k+1}; t_k, s, x(\bullet)), \quad (2.6)$$

где $x(\bullet)$ – «траектория» для моментов времени $t \leq t_k$.

Если свойства систем не меняются со временем, то системы и их модели называют *стационарными*. Стационарность означает независимость функции η от t и инвариантность функции σ к сдвигу во времени:

$$\eta(t, s(t)) = \eta(s(t)), \sigma(t; t_0, s, x(\bullet)) = \sigma(t + \tau; t_0 + \tau, s, \bar{x}(\bullet)), \quad (2.7)$$

где $\bar{x}(\bullet) = x(\bullet)$, сдвинутое на τ [6].

Конкретизацию моделей систем можно продолжить и далее.

Рассмотрим ряд обобщающих определений моделей систем, отличающихся компонентами, их числом и степенью абстракции [13].

- Модель системы представляет собой организованное множество:

$$SYS = (M, OP), \quad (2.8)$$

где M – множество; OP – оператор организации.

- Модель системы есть множество элементов, свойств и отношений [14]:

$$SYS = (\{m\}, \{n\}, \{r\}), \quad (2.9)$$

где m – элементы; n – свойства; r – отношения.

- Модель системы есть множество элементов, образующих структуру и обеспечивающих определенное поведение в условиях окружающей среды:

$$SYS = (\varepsilon, ST, BE, E), \quad (2.10)$$

где ε – элементы; ST – структура; BE – поведение; E – среда.

- Модель системы представляет собой множество входов, множество выходов, множество состояний, характеризуемых функцией переходов и функцией выходов (см. выше):

$$SYS = (X, Y, S, \sigma, \eta), \quad (2.11)$$

где X – входы; Y – выходы; S – состояния; σ – функция переходов; η – функция выходов.

- Модель системы предыдущего типа, дополненная фактором времени и множеством функциональных связей между входами, выходами и состояниями во времени [15]:

$$SYS = (t, X(t), Y(t), S(t), \sigma, \eta), \quad (2.12)$$

где t – время.

- Модель, соответствующая уровню бионических систем:

$$SYS = (GN, KD, MB, EV, FC, RP), \quad (2.13)$$

где параметры учитывают: GN – генетическое (родовое) начало, KD – условия существования, MB – обменные явления, EV – развитие, FC – функционирование, RP – репродукцию.

- Модель, соответствующая уровню организационно-технических систем:

$$SYS = (PL, RO, RI, EX, PR, DT, SV, RD, EF), \quad (2.14)$$

где параметры учитывают: PL – цели и планы, RO – ресурсы внешние, RI – ресурсы внутренние, EX – исполнителей, PR – процесс, DT – помехи, SV – контроль, RD – управление, EF – эффект.

- Проблемно-целевая модель организационно-технической системы:
 $SYS = (PRO, TAR, CR, IR, X, Y, S, CON, RO, RI, ST, T, OSC)$, (2.15)

где параметры учитывают: PRO – проблему, TAR – цели, CR – критерии целей системы, IR – проблемно-целевые информационно-аналитические ресурсы, X и Y – входные и выходные параметры, S – состояние системы, CON – управление, RO – внешние ресурсы управления, RI – внутренние ресурсы управления, ST – стратегию управления, T – время, OSC – решение обратной задачи [16].

Глава 3. ОБЛАСТИ ПРИМЕНЕНИЯ И КЛАССИФИКАЦИЯ НЕЧЕТКИХ МОДЕЛЕЙ

3.1. Области применения нечеткого моделирования

Модели статических и динамических систем, построение, использование и анализ которых базируется на положениях теории нечетких множеств и нечеткой логики, называют *нечеткими моделями*.

Нечеткое моделирование не подменяет собой различные методологии моделирования сложных систем, в которых существенные зависимости выяснены настолько хорошо, что они могут быть выражены в числах или символах, получающих в итоге численные оценки. Нечеткие модели скорее предоставляют необходимый инструмент для исследования как отдельных аспектов, так и всей системы в целом на различных этапах ее анализа в случае доминирования качественных элементов над количественными.

В табл. 3.1 показаны взаимосвязи между описаниями и переменными, характеризующими различные аспекты традиционных четких (*crisp*) и нечетких (*fuzzy*) моделей.

Таблица 3.1. Взаимосвязи между описаниями и переменными четких и нечетких моделей

<i>Описание модели</i>	<i>Входные данные</i>	<i>Результирующие (выходные) данные</i>	<i>Математические методы</i>
Четкое	Четкое	Четкое	Функциональный анализ, линейная алгебра и др.
Четкое	Нечеткое	Нечеткое	Принцип обобщения Заде
Четкое	Нечеткое	Четкое	Нечеткие модели, нечеткие вычисления
Нечеткое	Четкое/нечеткое	Нечеткое	Нечеткие модели, нечеткие вычисления

Нечеткие модели могут быть представлены как обобщение интервально-оцениваемых моделей, которые, в свою очередь, являются обобщением четких моделей. На рис. 3.1 иллюстрируются особенности вычисления функций для четких, интервальных и нечетких данных. Заметим, что функция $f: X \times Y$ может быть задана в виде *отношения* на декартовом произведении $X \times Y$. Вычисление функции для заданного значения входной переменной выполняется за три шага вне зависимости от типа функции и данных (четких, интервальных, нечетких):

- задание значения входной переменной x в пространстве $X \rightarrow Y$ (вертикальные пунктирные линии на рис. 3.1);
- нахождение пересечения с отношением;
- проекция этого пересечения на Y (горизонтальная пунктирная линия).

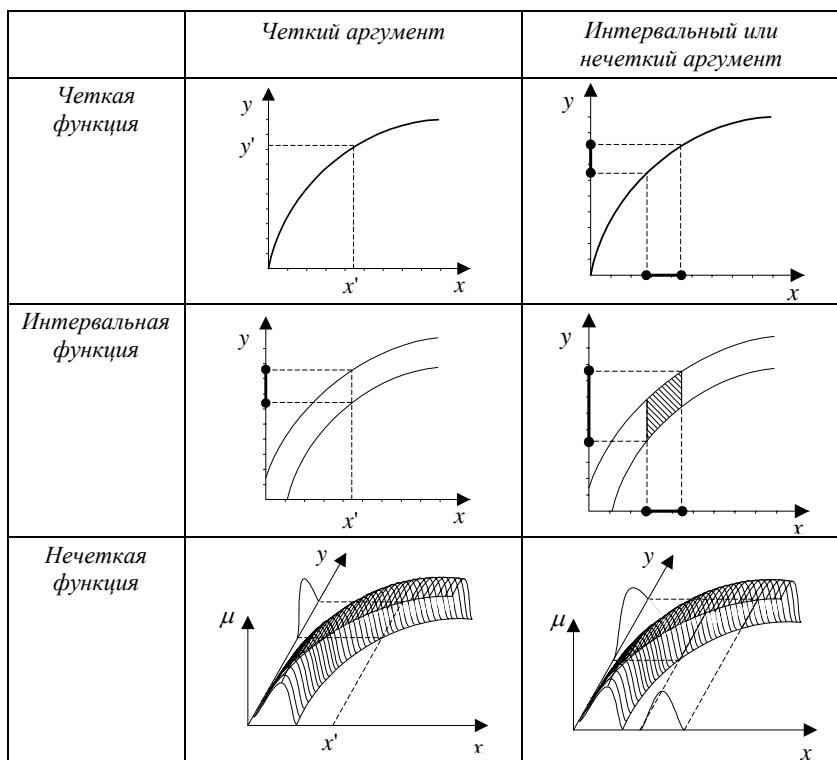


Рис. 3.1. Вычисление значений четких, интервальных и нечетких функций для четких, интервальных и нечетких аргументов

Можно выделить следующие сферы применения нечетких моделей.

Недостаточные или неопределенные знания об исследуемом объекте, системе. Традиционная теория систем использует четкие математические модели в виде алгебраических, дифференциальных или разностных уравнений. Такие модели позволяют адекватно описать системы, закономерности поведения и управления которыми четко определены. Однако для большинства конкретных систем получение приемлемого для построения четкой модели уровня требуемой информации является сложной, трудоемкой, дорогостоящей или вовсе невозможной задачей. Для них четкие математические модели не могут быть получены, или они являются слишком сложными для практического использования. Примеры таких систем могут быть найдены в различных областях: в химической или пищевой индустрии, в сфере биотехнологии, экологии, финансах, социологии [17, 18]. Значимая часть информации об этих системах доступна в виде экспертных данных или в эвристическом описании процессов функционирования. Эта информация может быть нечеткой и неопределенной для того, чтобы быть выраженной математическими зависимостями. Кроме того, информация о системе может быть разнокачественной, а оценка значений параметров проводиться с помощью различных шкал (отношений, интервалов, порядка и наименований). Однако зачастую возможно описать функционирование этих систем в виде эвристических предпочтений, используя конструкции естественного языка в форме правил типа «если-то». Такие нечеткие модели могут быть использованы для создания баз знаний, построенных на основе знаний экспертов в данной предметной области. С этой точки зрения нечеткие модели подобны экспертным системам, достаточно хорошо исследованным в рамках соответствующего направления искусственного интеллекта [19, 20].

Другой аспект неопределенности знаний о системе связан с неясностью или нечеткостью выделения и описания границы системы или отдельных ее состояний, а также входных и выходных воздействий [21].

Адекватная обработка неопределенной информации. Точные вычисления с использованием традиционных математических моделей дают необходимый эффект в случае, если параметры и входные данные являются точными и корректно представленными. Однако часто традиционные математические методы не только не позволяют адекватно обработать данные, но также не позволяют учесть естественно присущую этим данным неопределенность. Вероятностный подход является традиционным путем для учета неопределенности. Однако очевидно, что не все типы неопределенности могут быть учтены с использованием этого подхода. Нечеткая логика и теория нечетких множеств являются одним из эффективных подходов к решению данной проблемы [22].

«Прозрачное» (*gray-box*) моделирование и идентификация. Идентификация динамических систем на основе измерений их входных и выходных параметров является важной задачей в целом ряде практических приложений. Многие реальные системы являются нелинейными в своей основе и не могут быть представлены линейными моделями, использующими существующие методы идентификации [23]. В последнее время серьезное внимание уделяется развитию методов идентификации нелинейных систем на основе измеряемых (экспериментальных) данных. Искусственные нейронные сети и нечеткие модели являются наиболее вос требованными и адекватными для решения этих задач. Нечеткие модели при наличии обучающей выборки позволяют аппроксимировать функции или измеренные данные с любой требуемой точностью. Это свойство нечетких моделей определенного класса позволяет их отнести к универсальным аппроксиматорам [24–26].

Сравнивая нечеткие модели с другими хорошо известными методами аппроксимации, например с искусственными нейронными сетями, можно отметить их большую прозрачность, которая возможна благодаря их лингвистической интерпретации в виде нечетких продукционных правил. Логическая структура этих правил способствует пониманию и анализу системы количественно-качественными методами. В табл. 3.2 сравниваются различные подходы к моделированию систем.

Таблица 3.2. Взаимосвязи между описаниями и переменными четких и нечетких систем

Подход к моделированию	Источник информации	Метод извлечения (приобретения) знаний	Пример	Недостатки
Метод «белого ящика»	Формальные знания или данные	Математический (напр., уравнение Лагранжа)	Дифференциальные уравнения	Не может использовать знания, представленные в «естественному» виде
Метод «черного ящика»	Экспериментальные данные	Оптимизация (обучение)	Регрессия, нейронные сети	Не может вовсе использовать знания
Метод «серого ящика»	Экспертная информация или экспериментальные данные	Основанный на знаниях + +обучение	Нечеткие продукционные и реляционные модели	«Проклятие» размерности

3.2. Классификация нечетких моделей

В зависимости от назначения постулаты и положения теории нечетких множеств и нечеткой логики в нечетких моделях могут использоваться:

- *Непосредственно при описании системы.* В этом случае системы могут быть определены:
 - Нечеткими продукционными моделями (Rule-Based Fuzzy Models/-Systems – нечеткие модели/системы, основанные на правилах). Нечеткие модели данного класса являются наиболее общими для представления различных плохо формализуемых сложных систем.
 - В виде нечетких функциональных моделей (Fuzzy Functional Models/Systems) [27]. Модели этого вида, с одной стороны, можно представить как разновидность нечетких продукционных моделей, в которых данные в области предпосылок представляются в виде нечетких множеств, а зависимости входов-выходов в области заключений задаются в виде функции. С другой стороны, нечеткие функциональные модели можно выделить в отдельный класс в силу различной специфики формирования функциональных зависимостей, а также разнообразия решаемых с их использованием задач.
 - Нечеткими реляционными (в виде нечетких отношений) моделями (Fuzzy Relational Models/Systems) [28, 29]. Эти модели представляют собой альтернативу нечетким продукционным моделям, сохранив их особенности, однако не требуя дополнительных усилий для формирования нечетких продукционных правил.
- *При задании параметров системы.* Система может быть определена в виде алгебраических или дифференциальных уравнений, в которых параметры являются нечеткими числами, например

$$y = \tilde{3}x_1 + \tilde{5}x_2, \quad (3.1)$$

где $\tilde{3}$ и $\tilde{5}$ – нечеткие числа «около трех» и «около пяти», заданные соответствующими функциями принадлежности.

- *При задании входов, выходов и состояний системы [30, 31].*

Кроме того, нечеткие модели могут создаваться на основе совмещения указанных выше подходов.

Примечание. Под термином «нечеткая система» в Rule-Based Fuzzy Systems, Fuzzy Relational Systems и Fuzzy Functional Systems подразумевается совокупность элементов, преобразований и действий, необходимых для формализации, анализа и моделирования конкретного процесса, объекта. Поэтому в дальнейшем в этом контексте будем использовать термин «нечеткая модель».

3.3. Интеграция нечетких и нейронных сетей

Основные трудности при использовании нечетких моделей для решения практических задач связаны, как правило, с априорным определением компонентов этих моделей (нечетких высказываний, функций принадлежности для каждого значения лингвистических переменных, структуры базы нечетких правил и др.).

Нечеткие модели, как правило, удобно представлять в виде *нечетких сетей* – сетевых структур, элементы или совокупности элементов которых реализуют различные компоненты нечетких моделей и этапы нечеткого вывода. К основным достоинствам такого представления следует отнести, во-первых, конструктивность и наглядность интерпретации с их помощью компонентов модели, во-вторых, возможность выделения параметров (прежде всего параметров настройки) модели под соответствующие алгоритмы.

Поскольку компоненты нечетких моделей зачастую выбираются субъективно, они могут быть не вполне адекватны моделируемой системе или процессу, что отражается и на реализующих их нечетких сетях.

Одним из наиболее результативных примеров объединения формализмов модели и сети являются *искусственные нейронные сети*, представляющие собой совокупность нейроподобных элементов, определенным образом соединенных друг с другом и с внешней средой с помощью связей, задаваемых весовыми коэффициентами. Основное преимущество нейросетевого подхода – возможность выявления закономерностей в данных, их обобщение, т. е. извлечение знаний из данных, а основной недостаток – невозможность непосредственно (в явном виде, а не в виде вектора весовых коэффициентов межнейронных связей) представить функциональную зависимость между входом и выходом исследуемого объекта. Другим недостатком нейросетевого подхода является также трудность формирования представительной выборки, большое число циклов обучения и забывание «старого» опыта, сложность определения размера и структуры нейронной сети.

Подходы к исследованию сложных систем на основе нечетких и нейросетевых моделей взаимно дополняют друг друга, поэтому целесообразна их интеграция на основе принципа «мягких» вычислений (Soft Calculation). Основы такой интеграции сводятся к следующему: к терпимости к нечеткости и частичной истинности используемых данных для достижения интерпретируемости; гибкости и низкой стоимости решений. Могут быть предложены различные классификации способов и средств интеграции этих моделей.

В данной книге рассмотрены следующие классы сетей, доминирующие при представлении и анализе соответствующих типов моделей сложных систем и процессов:

- *нечеткие нейронные производственные сети*, в которых нейросетевая технология используется в качестве инструмента для реализации компонентов нечетких производственных моделей, – для анализа и моделирования системы на основе метода «серого ящика» (часть III);
- *нейронные нечеткие сети*, в которых на основе положений теории нечетких множеств осуществляется включение нечеткости в различные компоненты нейронных сетей (элементы структуры и механизмов обучения и функционирования), – для анализа и моделирования системы на основе метода «черного ящика» (часть IV);
- *гибридные нечетко-нейронные сети*, в которых реализуется интеграция нечетких и нейросетевых моделей в единую модель на уровне решаемых задач (часть IV);
- представление нечетких моделей с помощью структур в виде ориентированных графов различных типов, в описание отдельных компонентов которых введена нечеткость, – для анализа и моделирования систем и процессов на основе методов «серого и белого ящика» (часть V).

ВЫВОДЫ

Рассмотрена краткая классификация систем, представлен ряд обобщающих определений моделей систем, отличающихся компонентами описания, их числом и степенью абстракции. Охарактеризованы области применения нечеткого моделирования: недостаточные или неопределенные знания об исследуемой системе, адекватная обработка неопределенной информации, «прозрачное» моделирование и идентификация систем. Представлена классификация нечетких моделей в зависимости от их назначения: для описания системы; задания параметров; задания входов, выходов и состояний системы.

Выделено три основных класса нечетких моделей, доминирующих в рамках рассматриваемого подхода к анализу и моделированию сложных систем, а также различные классы нечетких сетей (и способы их интеграции с искусственными нейронными сетями), используемых для построения и анализа этих моделей.

СПИСОК ЛИТЕРАТУРЫ

1. Советский энциклопедический словарь. 4-е изд., испр. и доп. М., 1980.
2. Мусеев Н. Н. Алгоритмы развития. М., 1987.
3. Гумеров Ш. А. Развитие и организация // Системные концепции развития. Вып. 4. М., 1985.
4. Миротин Л. Б., Ташибаев И. Э. Системный анализ в логистике. М.: Экзамен, 2002.
5. Ерохина Е. А. Теория экономического развития: системно-синергетический подход. М., 1999.
6. Переходов Ф. И., Тарасенко Ф. П. Введение в системный анализ. М.: Высш. шк., 1989.
7. Николаев В. И., Брук В. М. Системотехника: методы и приложения. М.: Машиностроение, 1985.
8. Садовский В. Н. Смена парадигм системного мышления // Общеметодологические проблемы системных исследований. М., 1999.
9. Прангишвили И. В. Системный подход и общесистемные закономерности. М.: СИНТЕГ, 2000.
10. Семенов М. И., Трубилин И. Т., Лойко В. И., Барановская Т. П. Автоматизированные информационные технологии в экономике. М.: Финансы и статистика, 2001.
11. Растигин Л. А. Адаптация сложных систем. Рига: Зинатне, 1981.
12. Жариков О. Н., Королевская В. И., Хохлов С. Н. Системный подход к управлению/ Под ред. В. А. Персианова. М.: ЮНИТИ–ДАНА, 2001.
13. Теория систем и методы системного анализа в управлении и связи/ В. Н. Волкова, В. А. Воронков, А. А. Денисов и др. М.: Радио и связь, 1983.
14. Уемов А. И. Системный подход и общая теория систем. М.: Мысль, 1978.
15. Кухтенко А. И. Об аксиоматическом построении математической теории систем. – В кн. Кибернетика и вычислительная техника. Вып. 31. Киев: Наук. думка, 1976. С. 3–25.
16. Компьютерная поддержка сложных организационно-технических систем/ В. В. Борисов, И. А. Бычков, А. В. Дементьев, А. П. Соловьев, А. С. Федулов. М.: Горячая линия – Телеком, 2002.

17. Pedrycz W. Relevancy of fuzzy models. *Information Sciences.* V. 52. 1990. P. 285–302.
18. Yager R. R., Filev D. P. *Essentials of Fuzzy Modeling and Control.* New York: John Wiley, 1994.
19. Buchanan B. G., Shortliffe E. H. *Rule-Based Expert Systems. Readings (MA), USA: Addison-Wesley,* 1984.
20. Patterson D. W. *Introduction to Artificial Intelligence and Expert Systems.* Englewood Cliffs (NJ), USA: Prentice Hall, 1990.
21. Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БЧВ–Петербург, 2003.
22. Smets P., Mamdani E. H., Dubois D., Prade H. (Eds.). *Non-Standard Logics for Automated Reasoning.* Academic Press. London, 1988.
23. Ljung L. *System Identification, Theory for the User.* New Jersey: Prentice-Hall, 1987.
24. Kosko B. Fuzzy systems as universal approximators. *IEEE Trans. Computers.* V. 43. 1994. P. 1329–1333.
25. Wang L.-X. *Adaptive fuzzy systems and control, design and stability analysis.* New Jersey: Prentice Hall, 1994.
26. Zeng X. J., Singh M. G. Approximation theory of fuzzy systems – MIMO case. *IEEE Trans. Fuzzy Systems.* V. 3, N 2. 1995. P. 219–235.
27. Sugeno M., Tanaka K. Successive identification of a fuzzy model and its applications to prediction of a complex system. *Fuzzy Sets and Systems.* V. 42, 1991. P. 315–334.
28. Higashi M., Klir G. J. Identification of fuzzy relation systems. *IEEE Transactions on Systems, Man, and Cybernetics.* V. 14. 1984. P. 349–355.
29. Pedrycz W. An identification algorithm in fuzzy relational systems. *Fuzzy Sets and Systems.* V. 13. 1984. P. 153–167.
30. Мелихов А. Н., Бернитейн Л. С., Коровин С. Я. *Ситуационные советующие системы с нечеткой логикой.* М.: Наука, 1990.
31. Babuška R. *Fuzzy Systems, Modeling and Identification.* New Jersey. Prentice Hall, 2003.

Часть II

НЕЧЕТКИЕ ПРОДУКЦИОННЫЕ И РЕЛЯЦИОННЫЕ МОДЕЛИ

Глава 4. НЕЧЕТКИЕ ПРОДУКЦИОННЫЕ МОДЕЛИ

4.1. Компоненты нечетких производственных моделей

Нечеткие производственные модели (Rule-Based Fuzzy Models/Systems) являются наиболее общим видом нечетких моделей, используемых для описания, анализа и моделирования сложных слабоформализуемых систем и процессов.

В настоящее время существенно расширился круг изданий, в том числе отечественных авторов, посвященных различным аспектам построения и использования этих моделей. Это обосновывает необходимость уточнения основных терминов и понятий, по-разному трактуемых в этих работах.

Под *нечеткой производственной моделью* будем понимать согласованное множество отдельных нечетких производственных правил вида «**ЕСЛИ A, ТО B**» (где *A* и *B* – предпосылка (антecedent) и заключение (консеквент) данного правила в виде нечетких высказываний), предназначенное для определения степени истинности заключений нечетких производственных правил, на основе предпосылок с известной степенью истинности соответствующих правил.

Для построения нечеткой производственной модели необходимо задать следующие определяющие ее компоненты:

- способ (схему) нечеткого вывода заключений;
- базу нечетких производственных правил;
- процедуру введения нечеткости (fuzzification);
- процедуру агрегирования (aggregation) степени истинности предпосылок по каждому из нечетких производственных правил;
- процедуру активизации (activation) заключений каждого из нечетких производственных правил;

- процедуру аккумулирования (accumulation) активизированных заключений всех нечетких производственных правил для каждой выходной переменной;
- процедуру приведения к четкости (defuzzification) для каждой аккумулированной выходной переменной;
- процедуру параметрической оптимизации конечной базы нечетких правил [1].

В настоящее время предложено множество (более 100) разновидностей нечетких производственных моделей на основе различных комбинаций указанных выше компонентов. Далее более подробно рассмотрим каждый из этих компонентов, а также наиболее известные нечеткие производственные модели.

4.2. Способы нечеткого вывода

В нечетких производственных моделях используются в основном два способа нечеткого вывода заключений: прямой и обратный.

Прямой способ нечеткого вывода, или прямая нечеткая цепочка рассуждений (fuzzy forward-chaining reasoning), базируется на правиле вывода *нечеткий модус поненс* (fuzzy modus ponens), обобщенная схема которого включает в себя следующие этапы.

Этап 1. Задание нечеткой импликации $R: A \rightarrow B$, определяющей нечеткое причинно-следственное отношение между предпосылкой (антecedентом) A и заключением (консеквентом) B , которое представляется в виде нечеткой продукции:

$$\text{ЕСЛИ } x \text{ есть } A, \text{ ТО } y \text{ есть } B, \quad (4.1)$$

где x – входная переменная, $x \in X$, где X – область определения предпосылки нечеткого производственного правила; A – нечеткое множество, определенное на X , с функцией принадлежности $\mu_A(x) \in [0, 1]$; y – выходная переменная, $y \in Y$, где Y – область определения заключения; B – нечеткое множество, определенное на Y , с функцией принадлежности $\mu_B(y) \in [0, 1]$.

Операция нечеткой импликации занимает центральное место в нечетких производственных моделях, определяя причинно-следственное отношение между предпосылками и заключениями правил. В настоящее время существует большое число (несколько десятков) различных вариантов этой операции. Всех их можно разделить на три основных класса:

- *S-импликации*, определяемые как

$$\mu_R(x, y) = S(\text{NOT}(\mu_A(x), \mu_B(y))), \quad (4.2)$$

где S – оператор S -нормы; NOT – оператор нечеткого дополнения на интервале $[0, 1]$.

Основой операций импликации этого класса является булева импликация. Типичными примерами S -импликации могут служить импликации Клине–Даэнса и Лукашевича.

- *R-импликации*, образованные на основе сравнения с T -нормой:

$$\mu_R(x, y) = \sup\{z \in [0, 1] | T(\mu_A(x), z) \leq \mu_B(y)\}. \quad (4.3)$$

Эти операции импликации основаны на формализме интуиционистской логики. Примерами R -импликации являются импликации Геделя и Гейнса.

- *T-импликации*, определяемые как

$$\mu_R(x, y) = T(\mu_A(x), \mu_B(y)), \quad (4.4)$$

где T – оператор T -нормы.

Хотя эти операции импликации не удовлетворяют их общепринятым свойствам, они широко используются в этом качестве в различных приложениях нечеткой логики. Типичными примерами T -импликации являются импликации Мамдани и Ларсена [2].

В работе [3] предложено оценивать операции нечеткой импликации по двум критериям:

- максимальная ошибка, обусловленная операцией импликации, не должна превышать максимальной ошибки лингвистического представления данных;
- нечеткая модель, построенная с использованием выбранной операции нечеткой импликации, должна быть рабастной по отношению к выбору вида функции принадлежности.

На основе проведенного анализа, исходя из вышеуказанных критериев, в ряде работ сделан вывод о наиболее эффективном использовании следующих операций нечеткой импликации [4, 5]:

- классической нечеткой импликации, или импликация Клине–Даэнса (Kleene–Dienes))

$$\mu_R(x, y) = \max\{1 - \mu_A(x), \mu_B(y)\}, \text{ при } \mu_A(x) \geq \mu_B(y); \quad (4.5)$$

- нечеткой импликации Заде (Zadeh)

$$\mu_R(x, y) = \max\{\min\{\mu_A(x), \mu_B(y)\}, 1 - \mu_A(x)\}; \quad (4.6)$$

- нечеткой импликации Мамдани (Mamdani)

$$\mu_R(x, y) = \min\{\mu_A(x), \mu_B(y)\}; \quad (4.7)$$

- нечеткой импликации Ларсена (Larsen)

$$\mu_R(x, y) = \mu_A(x) \cdot \mu_B(y); \quad (4.8)$$

- нечеткой импликации Лукашевича (Lukasiewicz)

$$\mu_R(x, y) = \min\{1, 1 - \mu_A(x) + \mu_B(y)\}, \quad (4.9)$$

или

$$\mu_R(x, y) = \max\{0, \mu_A(x) + \mu_B(y) - 1\}; \quad (4.9)$$

- стандартной четкой (standard strict) импликации

$$\mu_R(x, y) = \begin{cases} 1, & \text{если } \mu_A(x) \leq \mu_B(y), \\ 0, & \text{если } \mu_A(x) > \mu_B(y); \end{cases} \quad (4.10)$$

- нечеткой импликации Геделя (Gödel)

$$\mu_R(x, y) = \begin{cases} 1, & \text{если } \mu_A(x) \leq \mu_B(y), \\ \mu_B(y), & \text{если } \mu_A(x) > \mu_B(y); \end{cases} \quad (4.11)$$

- нечеткой импликации Гейнса (Gaines)

$$\mu_R(x, y) = \begin{cases} 1, & \text{если } \mu_A(x) \leq \mu_B(y), \\ \mu_B(y)/\mu_A(x), & \text{если } \mu_A(x) > \mu_B(y); \end{cases} \quad (4.12)$$

- нечеткой импликации Гогуэна

$$\mu_R(x, y) = \min\{1, \mu_B(y) / \mu_A(x)\}, \text{ при } \mu_A(x) > 0; \quad (4.13)$$

- нечеткой импликации Клине–Даэнса–Лукашевича (Kleene–Dienes–Lukasiewicz):

$$\mu_R(x, y) = 1 - \mu_A(x) + \mu_A(x) \mu_B(y); \quad (4.14)$$

- нечеткой вероятностной импликации

$$\mu_R(x, y) = \min\{1, 1 - \mu_A(x) + \mu_A(x) \mu_B(y)\}; \quad (4.15)$$

- нечеткой импликации с ограниченной суммой

$$\mu_R(x, y) = \min\{1, \mu_A(x) + \mu_B(y)\}; \quad (4.16)$$

- нечеткой импликации Н. Вади

$$\mu_R(x, y) = \max\{\mu_A(x) \mu_B(y), 1 - \mu_A(x)\}; \quad (4.17)$$

- нечеткой импликации Ягера (Yager)

$$\mu_R(x, y) = \mu_A(y)^{\mu_B(x)}. \quad (4.18)$$

Помимо указанных выше критериев, выбор той или иной операции нечеткой импликации осуществляется в зависимости от используемого

базиса нечетких операций и ее наиболее эффективной вычислительной реализации.

Этап 2. Задание нечеткого условия (факта):

$$\langle x' \text{ есть } A' \rangle, \quad (4.19)$$

где x' – фактическое значение переменной x ; A' – нечеткое множество, отражающее значение x' , определенное на X , с функцией принадлежности $\mu_{A'}(x) \in [0, 1]$.

Этап 3. Формирование вывода:

$$\langle y' \text{ есть } B' \rangle, \quad (4.20)$$

где y' – полученное значение переменной y ; B' – нечеткое множество, отражающее значение y' , определенное на Y , с функцией принадлежности $\mu_{B'}(y) \in [0, 1]$.

Так как нечеткое условие (4.19) содержит нечеткое множество A' , которое является «близким» в определенном смысле к нечеткому множеству A , то в результате формируется вывод в виде функции принадлежности нечеткого множества B' , характеризующей степень его «близости» к нечеткому множеству B .

Таким образом, процесс получения результата прямого нечеткого вывода B' с использованием нечеткой импликации $A \rightarrow B$ и нечеткого условия $\langle x' \text{ есть } A' \rangle$ можно представить в виде

$$B' = A' \bullet R = A' \bullet (A \rightarrow B), \quad (4.21)$$

где $\langle \bullet \rangle$ – операция свертки (композиционное правило нечеткого логического вывода).

Нечеткая импликация $A \rightarrow B$ соответствует нечеткому отношению R , которое можно рассматривать как нечеткое подмножество декартова (прямого) произведения $X \times Y$ полного множества предпосылок X и заключений Y с функцией принадлежности $\mu_R(x, y)$. Поэтому функцию принадлежности нечеткого множества B' можно представить в следующем виде [6]:

$$\mu_{B'}(y) = \sup_{x \in X} \{\mu_{A'}(x) T \mu_R(x, y)\}, \quad (4.22)$$

Здесь $\sup_{x \in X} \{\dots\}$ означает верхнюю границу множества элементов $\{\dots\}$, где

x «пробегает» все значения из X ; T – операция Т-нормы.

Если X представляет собой множество с конечным числом элементов, то выражение (4.22) принимает вид

$$\mu_{B'}(y) = \max_{x \in X} \{\mu_{A'}(x) T \mu_R(x, y)\}. \quad (4.23)$$

Само же композиционное правило (4.23) называется (*max-T*)-*композицией* или *макстриангулярной композицией*.

Среди правил *композиции* наибольшее распространение получили следующие операции:

- (*sup-min*)-*композиция*

$$\mu_{B'}(y) = \sup_{x \in X} \{\min [\mu_{A'}(x), \mu_R(x, y)]\}; \quad (4.24)$$

- (*sup-prod*)-*композиция*

$$\mu_{B'}(y) = \sup_{x \in X} \{\mu_{A'}(x) \cdot \mu_R(x, y)\}, \quad (4.25)$$

а также их разновидности, правила *макстриангулярной композиции*, для случаев, если X представляет собой множество с конечным числом элементов:

- (*max-min*)-*композиция*

$$\mu_{B'}(y) = \max_{x \in X} \{\min [\mu_{A'}(x), \mu_R(x, y)]\}; \quad (4.26)$$

- (*max-prod*)-*композиция*

$$\mu_{B'}(y) = \max_{x \in X} \{\mu_{A'}(x) \cdot \mu_R(x, y)\}. \quad (4.27)$$

Широкое применение правила (*max-min*)-*композиции* обусловлено хорошими алгебраическими свойствами этой операции (ассоциативностью и дистрибутивностью относительно *max*), необходимыми в задачах нечеткого моделирования, а правила (*max-prod*)-*композиции* – простотой реализации и большей чувствительностью к изменениям входных переменных в предпосылках нечетких продукционных правил.

В качестве операции *T*-нормы в выражениях (4.22) и (4.23) используется действительная функция от двух переменных $T: [0, 1] \times [0, 1] \rightarrow [0, 1]$, удовлетворяющая следующим свойствам:

$$1) T(a, 0) = 0, T(a, 1) = a \text{ (ограниченность)}; \quad (4.28)$$

$$2) T(a, b) = T(b, a) \text{ (коммутативность)}; \quad (4.29)$$

$$3) T(a, T(b, c)) = T(T(a, b), c) \text{ (ассоциативность)}; \quad (4.30)$$

$$4) T(a, b) \leq T(c, d), \text{ если } a \leq c \text{ и } b \leq d \text{ (монотонность).} \quad (4.31)$$

Каждой *T*-норме соответствует своя *S*-норма, представляющая действительную функцию от двух переменных $S: [0, 1] \times [0, 1] \rightarrow [0, 1]$ и удовлетворяющая следующим свойствам:

$$1) S(a, 0) = a, S(a, 1) = 1 \text{ (ограниченность)}; \quad (4.32)$$

$$2) S(a, b) = S(b, a) \text{ (коммутативность)}; \quad (4.33)$$

$$3) S(a, S(b, c) = S(S(a, b), c) \text{ (ассоциативность);} \quad (4.34)$$

$$4) S(a, b) \leq S(c, d), \text{ если } a \leq c \text{ и } b \leq d \text{ (монотонность).} \quad (4.35)$$

Зависимость между T- и S-нормами выражается следующим равенством:

$$(a \text{ T } b) = 1 - [(1 - a) \text{ S } (1 - b)]. \quad (4.36)$$

«Популярность» использования T- и S-норм обусловлена возможностью настройки параметров этих нечетких операторов, а также их хорошими алгебраическими свойствами, необходимыми для решения целого ряда прикладных задач анализа и моделирования различных систем. В табл. 4.1 представлены наиболее часто используемые T- и S-нормы.

Таблица 4.1. Примеры наиболее часто используемых T- и S-норм

$T(a, b)$	$S(a, b)$	Параметры
$\min\{a, b\}$	$\max\{a, b\}$	
$a \cdot b$	$a + b - a \cdot b$	
$\max\{0, a + b - 1\}$	$\min\{1, a + b\}$	
$\begin{cases} a, & \text{если } b = 1, \\ b, & \text{если } a = 1, \\ 0, & \text{если } a, b < 1 \end{cases}$	$\begin{cases} a, & \text{если } b = 0, \\ b, & \text{если } a = 0, \\ 0, & \text{если } a, b > 0 \end{cases}$	
$\frac{ab}{\max\{a, b, \gamma\}}$	$\frac{(1-a)(1-b)}{\max\{(1-a), (1-b), \gamma\}}$	$\gamma \in [0, 1]$
$\frac{ab}{\gamma + (1-\gamma)(a+b-ab)}$	$\frac{a+b-(2-\gamma)ab}{\gamma - (1-\gamma)ab}$	$\gamma > 0$
$\frac{1}{1 + \left(\left(\frac{1}{a} - 1 \right)^{\gamma} + \left(\frac{1}{b} - 1 \right)^{\gamma} \right)^{\frac{1}{\gamma}}}$	$\frac{1}{1 + \left(\left(\frac{1}{a} - 1 \right)^{-\gamma} + \left(\frac{1}{b} - 1 \right)^{-\gamma} \right)^{\frac{1}{\gamma}}}$	$\gamma > 0$
$\frac{1}{\left(\frac{1}{a^{\gamma}} + \frac{1}{b^{\gamma}} \right)^{\frac{1}{\gamma}} - 1}$	$1 - \frac{1}{\left(\frac{1}{(1-a)^{\gamma}} + \frac{1}{(1-b)^{\gamma}} - 1 \right)^{\frac{1}{\gamma}}}$	$\gamma > 1$
$1 - \left((1-a)^{\gamma} + (1-b)^{\gamma} - (1-a)^{\gamma}(1-b)^{\gamma} \right)^{\frac{1}{\gamma}}$	$(a^{\gamma} + b^{\gamma} - a^{\gamma}b^{\gamma})^{\frac{1}{\gamma}}$	$\gamma > 0$
$\max \left\{ 0, 1 - \left((1-a)^{\gamma} + (1-b)^{\gamma} \right)^{\frac{1}{\gamma}} \right\}$	$\min \{1, (a^{\gamma} + b^{\gamma})^{\frac{1}{\gamma}}\}$	$\gamma \geq 1$

Окончание табл. 4.1

$T(a, b)$	$S(a, b)$	Параметры
$\log_{\gamma} \left(1 + \frac{(\gamma^a - 1)(\gamma^b - 1)}{\gamma - 1} \right)$	$1 - \log_{\gamma} \left(1 + \frac{\gamma^{1-a} + \gamma^{1-b}}{\gamma - 1} \right)$	$\gamma > 0,$ $\gamma \neq 1$
$\max \{0, (\gamma - 1)(a + b) - 1 - \gamma ab\}$	$\min \{1, a + b + \gamma ab\}$	$\gamma > -1$

Обратный способ нечеткого вывода, или обратная нечеткая цепочка рассуждений (fuzzy backward-chaining reasoning), основывается на использовании правила вывода *нечеткий модус толленс* (fuzzy modus tollens). Его схема включает в себя следующие этапы.

Этап 1. Задание нечеткой импликации $R: A \rightarrow B$, определяющей, как и в случае прямого способа вывода, нечеткое причинно-следственное отношение между предпосылкой и заключением, которое представляется в виде нечеткой продукции (4.1).

Этап 2. Задание нечеткого условия « y' есть B' ».

Этап 3. Формирование вывода « x' есть A' ». Целью способа обратного нечеткого вывода является установление истинности предпосылки нечеткой продукции (4.1). А сам процесс получения результата обратного нечеткого вывода A' с использованием нечеткой импликации $A \rightarrow B$ и нечеткого условия « y' есть B' » можно представить в виде

$$A' = R \bullet B' = (A \rightarrow B) \bullet B'. \quad (4.37)$$

Причем вывод по данному способу получается в виде функции принадлежности нечеткого множества предпосылки A' на основе функции принадлежности нечеткой импликации $A \rightarrow B$ и функции принадлежности нечеткого множества заключения B' [6, 7]:

$$\mu_{A'}(x) = \sup_{y \in Y} \{ \mu_R(x, y) T \mu_{B'}(y) \}. \quad (4.38)$$

В дальнейшем, если не указано иное, будем рассматривать модели, реализующие прямой способ нечеткого вывода.

4.3. Создание базы нечетких производных правил

Для задания базы нечетких производных правил необходимо решить следующие вопросы:

- сформировать простые нечеткие высказывания в предпосылках и заключениях правил;
- при необходимости сформировать составные нечеткие высказывания в предпосылках и заключениях правил;

- выбрать тип нечетких производственных правил;
- задать структуру базы нечетких производственных правил;
- оценить и обеспечить полноту (достаточность количества нечетких правил), их непротиворечивость и, по возможности, устраниТЬ корреляции между отдельными нечеткими правилами в базе.

4.3.1. Формирование нечетких высказываний в предпосылках и заключениях правил

При формировании простых нечетких высказываний в предпосылках и заключениях нечетких производственных правил (например, « x есть A », « y есть B » в правилах вида (4.1)) необходимо задать функции принадлежности соответствующих нечетких множеств. Либо эксперту может быть заранее задан набор готовых функций принадлежности, покрывающих по базовым множествам пространства входных и выходных переменных, либо от него будет требоваться построение функций принадлежности.

Сами способы построения функций принадлежности могут быть разделены на *прямые* и *косвенные*. Прямые методы основаны на непосредственном задании экспертом нечетких множеств. Они используются для измеримых понятий или когда выделяются их полярные значения [8]. В случае опроса и согласования мнений группы экспертов, как правило, применяется прямой метод построения функций принадлежности, приближенно равных некоторому нечеткому числу, а также метод приближенных интервальных оценок [9]. Косвенные методы определения значений функций принадлежности используются в случаях, когда нет измеримых свойств. Они употребляются наиболее часто. Рассмотрим краткое описание основных косвенных методов построения функций принадлежности.

Методы парных сравнений

Пусть $X = \{x\}$ – множество из n элементов. Нечеткое множество G множества X есть совокупность пар вида

$$S = \{\mu_G(x)/(x)\}, x \in X, \quad (4.39)$$

где $\mu_G(x)$ – степень принадлежности элемента x ко множеству G . Для всех элементов множества G должно выполняться условие

$$\sum_{i=1}^n \mu_G(x_i) = 1. \quad (4.40)$$

Степень принадлежности элементов ко множеству определяется посредством парных сравнений. При этом, например, используются оценки, приведенные в табл. 4.2.

Таблица 4.2. Оценки парных сравнений элементов

Оценки важности	Качественная оценка	Примечание
1	Однаковая значимость	Одинарный ранг альтернатив
3	Слабое превосходство	Незначительное предпочтение одной альтернативы другой
5	Сильное превосходство	Существенные доказательства превосходства
7	Очевидное превосходство	Убедительные доказательства превосходства
9	Абсолютное превосходство	Комментарии излишни
2, 4, 6, 8	Промежуточные значения	Необходимы при компромиссе

Оценку элемента x_i по сравнению с элементом x_j с точки зрения свойства G обозначим через a_{ij} . Для согласованности этих значений примем $a_{ij} = 1/a_{ji}$. Оценки a_{ij} составляют матрицу $A = \|a_{ij}\|$. Собственный вектор $\mathbf{b} = (b_1, \dots, b_n)$ матрицы A находится из уравнения $A\mathbf{b} = \lambda\mathbf{b}$, где λ – собственное значение матрицы A . Вычисленные значения, составляющие собственный вектор \mathbf{b} , принимаются в качестве степени принадлежности элементов x ко множеству G :

$$\mu_G(x_i) = b_i, i = 1, \dots, n. \quad (4.41)$$

Так как всегда выполняется равенство $A\mathbf{b} = n\mathbf{b}$, то найденные значения тем точнее, чем ближе λ_{\max} к n . Отклонение λ_{\max} от n может служить мерой согласованности суждений экспертов [9].

Методы на основе статистических данных

В качестве степени принадлежности элемента x ко множеству G принимается оценка частоты использования значения терма, задаваемого нечетким множеством, для характеристики элемента. Степень принадлежности некоторого значения вычисляется как отношение числа экспериментов, в которых оно встречалось в определенном интервале шкалы, к максимальному для этого значения числу экспериментов по всем интервалам при условии, что в каждый интервал шкалы попадает одинаковое число экспериментов. Полученные данные затем обрабатываются для уменьшения внесенных экспериментом искажений на основе использования свойств функций принадлежности (один максимум и гладкие, затухающие до нуля фронты), а также правил нормировки, кусочно-линейной аппроксимации [9].

Использование типовых L-R-функций

В качестве функций принадлежности используются типовые L-R-функции (треугольная, трапецидальная, гауссова, колоколообразная, сигмоидальная и др.), конкретный вид которых определяется значениями параметров, входящих в их аналитические представления, и уточняется в соответствии с данными экспериментов [8, 10].

4.3.2. Формирование составных нечетких высказываний в предпосылках и заключениях правил

Как в предпосылках, так и в заключениях нечетких продукционных правил вида (4.1) нечеткие высказывания могут быть как простыми (например: « x есть A », « y есть B »), так и составными, образованными из простых, для комбинирования которых используются нечеткие логические операции конъюнкции «И» и дизъюнкции «ИЛИ».

Возможны следующие случаи:

- во-первых, составные нечеткие высказывания в предпосылке или в заключении, соединяемые нечеткими логическими операциями «И», «ИЛИ», относятся к одной и той же переменной;
- во-вторых, составные нечеткие высказывания в предпосылке, соединяемые нечеткими логическими операциями «И», «ИЛИ», относятся к разным переменным;
- в-третьих, составные нечеткие высказывания в заключении, соединяемые нечеткими логическими операциями «И», «ИЛИ», относятся к разным переменным.

В первом случае, вне зависимости от того, где находятся нечеткие высказывания, относящиеся к одной и той же переменной (в предпосылке или в заключении), из нескольких нечетких высказываний формируется одно новое простое нечеткое высказывание.

Во втором случае, который возможен, например, в ситуации противоречивых либо не вполне согласованных предпосылок, вместо высказывания « x есть A_1 И x есть A_2 » формируется новое высказывание – « x есть A^* », где A^* – нечеткое множество, равное пересечению нечетких множеств A_1 и A_2 с использованием одного из следующих способов:

- *min-конъюнкция* нечетких множеств A_1 и A_2 определяется как нечеткое множество A^* , заданное на X , с функцией принадлежности $\mu_{A^*}(x) \in [0, 1]$:

$$\mu_{A^*}(x) = \min\{\mu_{A_1}(x), \mu_{A_2}(x)\}, \forall x \in X; \quad (4.42)$$

- алгебраическое произведение

$$\mu_{A^*}(x) = \mu_{A_1}(x) \cdot \mu_{A_2}(x), \forall x \in X; \quad (4.43)$$

- граничное произведение

$$\mu_{A^*}(x) = \max\{\mu_{A_1}(x) + \mu_{A_2}(x) - 1, 0\}, \forall x \in X; \quad (4.44)$$

- драстическое произведение

$$\mu_{A^*}(x) = \begin{cases} \mu_{A_2}(x), & \text{если } \mu_{A_1}(x) = 1, \\ \mu_{A_1}(x), & \text{если } \mu_{A_2}(x) = 1, \\ 0, & \text{в остальных случаях,} \end{cases} \quad (4.45)$$

- λ -сумма

$$\mu_{A^*}(x) = \lambda \cdot \mu_{A_1}(x) + (1 - \lambda) \cdot \mu_{A_2}(x), \forall x \in X, \quad (4.46)$$

где $\lambda \in [0, 1]$.

Вместо высказывания « x есть A_1 ИЛИ x есть A_2 » формируется новое высказывание – « x есть A^* », где A^* – нечеткое множество, соответствующее объединению нечетких множеств A_1 и A_2 с использованием одного из следующих способов:

- *так-дизъюнкция* нечетких множеств A_1 и A_2 определяется как нечеткое множество A^* , заданное на X , с функцией принадлежности $\mu_{A^*}(x) \in [0, 1]$:

$$\mu_{A^*}(x) = \max\{\mu_{A_1}(x), \mu_{A_2}(x)\}, \forall x \in X; \quad (4.47)$$

- алгебраическая сумма

$$\mu_{A^*}(x) = \mu_{A_1}(x) + \mu_{A_2}(x) - \mu_{A_1}(x) \cdot \mu_{A_2}(x), \forall x \in X; \quad (4.48)$$

- граничная сумма

$$\mu_{A^*}(x) = \min\{\mu_{A_1}(x) + \mu_{A_2}(x), 1\}, \forall x \in X; \quad (4.49)$$

- драстическая сумма

$$\mu_{A^*}(x) = \begin{cases} \mu_{A_2}(x), & \text{если } \mu_{A_1}(x) = 0, \\ \mu_{A_1}(x), & \text{если } \mu_{A_2}(x) = 0, \\ 1, & \text{в остальных случаях;} \end{cases} \quad (4.50)$$

- λ -сумма (см. выражение (4.46)).

Во втором случае, когда составные нечеткие высказывания в предпосылке, соединяемые нечеткими логическими операциями «И», «ИЛИ», относятся к разным переменным, используется процедура агрегирования степени истинности предпосылок по каждому из нечетких продукционных правил (см. подразд. 4.3).

При этом *конъюнктивная форма предпосылки* (с использованием только нечетких логических операций «И») является наиболее общим видом предпосылки, так как нечеткое продукционное правило, в предпосылке которого нечеткие высказывания соединены нечеткой логической операцией «ИЛИ», может быть представлено несколькими правилами.

Пример. Одно правило:

ЕСЛИ x_1 есть A_1 , **ИЛИ** x_2 есть A_2 , **ТО** y есть B ,

можно представить двумя следующими правилами:

ЕСЛИ x_1 есть A_1 , **ТО** y есть B ,

ЕСЛИ x_2 есть A_2 , **ТО** y есть B .

В третьем случае, когда составные нечеткие высказывания в заключении, соединяемые нечеткими логическими операциями «И», «ИЛИ», относятся к разным переменным, используется процедура аккумулирования заключений нечетких продукционных правил (см. 4.4).

4.3.3. Типы нечетких продукционных правил

Несмотря на то что предпосылки и заключения правил нечетких продукционных моделей могут формироваться по-разному (см. ниже), всех их можно разделить на две основные группы:

- нечеткие лингвистические продукционные правила, в которых как предпосылки, так и заключения правил являются нечеткими высказываниями (fuzzy proposition) [11];
- нечеткие продукционные правила, в которых предпосылки правил являются нечеткими высказываниями, а заключения – четкими значениями (т. е. заданными одноточечными нечеткими множествами) или функциями [12].

Нечеткие лингвистические продукционные правила

Нечеткие лингвистические продукционные правила, используемые при построении базы нечетких правил, можно разделить на следующие типы:

- правила, предпосылки и заключения которых формируются на основе нечетких множеств типа 1 [11];

- правила, предпосылки и заключения которых основаны на нечетких множествах в сочетании с нечеткими отношениями, модифицирующими лингвистические переменные;
- правила, основанные на нечетких множествах в сочетании с групповыми и полугрупповыми расширенными операциями [13];
- правила, основанные на принципе расширения [14];
- правила, основанные на нечетких множествах с адаптацией операций над ними [15].

Рассмотрим более подробно нечеткие лингвистические продукционные правила указанных выше типов.

Правила, предпосылки и заключения которых формируются на основе нечетких множеств типа I, можно представить в виде

$$\begin{aligned} \text{П.1: ЕСЛИ } & x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } & y_1 \text{ есть } B_{i1} \text{ И } \dots \text{ И } y_k \text{ есть } B_{ik} \text{ И } \dots, y_p \text{ есть } B_{ip}, i = 1, \dots, n, \end{aligned} \quad (4.51)$$

где x_j ($j = 1, \dots, m$) – входные переменные, которые могут быть как четкими, так и нечеткими, $x_j \in X_j$, X_j – область определения соответствующей предпосылки; y_k ($k = 1, \dots, p$) – нечеткие выходные переменные, $y_k \in Y_k$, Y_k – область определения соответствующего заключения; A_{ij} , B_{ik} – лингвистические термы (например, {маленький, средний, большой}), представляющие собой нечеткие множества, определенные на X_j и Y_k с функциями принадлежности $\mu_{A_{ij}}(x_j) \in [0, 1]$ и $\mu_{B_{ik}}(x_k) \in [0, 1]$ соответственно.

Эти правила являются наиболее часто используемым, по сравнению с другими, типом нечетких лингвистических продукционных правил, которые применяются как для создания моделей конкретных объектов (систем), так и для построения их оценочных моделей (например, моделей предпочтений лиц, принимающих решения).

Примечание. В соответствии с определением, данным в [16], нечетким множеством типа n называется нечеткое множество, у которого значениями функции принадлежности является нечеткое множество типа $(n-1)$. Например, нечеткое множество типа 1 можно определить как $\mu: X \rightarrow [0, 1]$, а нечеткое множество типа 2 – как $\mu: X \times [0, 1] \rightarrow [0, 1]$ и т. д.

Правила, предпосылки и заключения которых основаны на нечетких множествах в сочетании с нечеткими отношениями, модифицирующими лингвистические переменные [17]. В данных правилах, в отличие от (4.51), нечеткие высказывания типа « x_j есть A_{ij} » и « y_k есть

B_{ik} » заменяются на высказывания соответственно « x_j есть $\Theta_{ij}(A_{ij})$ » и y_k есть $Q_{ik}(B_{ik})$.

$$\begin{aligned} \Pi_i: & \text{ЕСЛИ } x_1 \text{ есть } \Theta_{i1}(A_{i1}) \text{ И } \dots \text{ И } x_j \text{ есть } \Theta_{ij}(A_{ij}) \text{ И } \dots \text{ И } x_m \\ & \quad \text{есть } \Theta_{im}(A_{im}), \\ & \text{ТО } y_1 \text{ есть } Q_{i1}(B_{i1}) \text{ И } \dots \text{ И } y_k \text{ есть } Q_{ik}(B_{ik}) \text{ И } \dots \text{ И } y_p \\ & \quad \text{есть } Q_{ip}(B_{ip}), i = 1, \dots, n, \end{aligned} \quad (4.52)$$

где $\Theta_{ij}(A_{ij})$, $Q_{ik}(B_{ik})$ – лингвистические отношения, задаваемые некоторыми синтаксическими процедурами (с помощью нечетких логических связок «И», «ИЛИ», а также нечетких модификаторов «немного», «НЕ», «очень» и др.) и соответствующими им семантическими процедурами, ставящими в соответствие данным лингвистическим значениям сформированные нечеткие множества.

Показательным примером использования здесь нечетких логических операций «И», «ИЛИ» является замена двух или более нечетких высказываний, относящихся к одной переменной, одним нечетким высказыванием (см. 4.2.1). Например, вместо нечеткого высказывания « x_j есть A_1 **ОП** x_j есть A_2 » формируется нечеткое высказывание « x_j есть A^* », где **ОП** – обозначение логической операции; A^* – нечеткое множество, равное в случае связки «И» пересечению нечетких множеств A_1 и A_2 или равное в случае связки «ИЛИ» объединению этих нечетких множеств. В качестве нечетких логических связок «И» могут быть использованы операции (4.42)–(4.46), а в качестве нечетких логических связок «ИЛИ» – операции (4.46)–(4.50).

В качестве модификаторов нечетких высказываний наиболее часто используют следующие: «ОЧЕНЬ», «ПОЧТИ», «НЕМНОГО БОЛЬШЕ/МЕНЬШЕ», «МНОГО БОЛЬШЕ/МЕНЬШЕ», «МНОГО», а также связку «НЕ». Например, высказывание «давление высокое» заменяется на высказывание «давление немного ниже, чем высокое».

Притом при модификации функций принадлежности соответствующих нечетких множеств, как правило, используются следующие операции:

$$\text{для } A^* = \text{«ОЧЕНЬ } A\text{»}, \mu_{A^*}(x) = \mu_A^2(x), \quad (4.53)$$

$$\text{для } A^* = \text{«ПОЧТИ } A\text{»}, \mu_{A^*}(x) = \mu_A^{1/2}(x), \quad (4.54)$$

$$\text{для } A^* = \text{«НЕ-} A\text{»}, \mu_{A^*}(x) = 1 - \mu_A(x). \quad (4.55)$$

Правила, основанные на нечетких множествах в сочетании с групповыми и полугрупповыми расширенными операциями. Заметим, что возможны различные примеры записи правил данного типа в зависимости от решаемой задачи. Типичным примером таких правил являются следующие:

П_i: ЕСЛИ x_1 есть A_{i1} И ... И x_j есть A_{ij} И ... И x_m есть A_{im} ,
ТО y_1 есть $A_{i1}*B_{i1}$ И ... И y_k есть $A_{ik}*B_{ik}$ И ... И y_m есть $A_{im}*B_{im}$,

$$i = 1, \dots, n,$$
(4.56)

где * – групповая или полугрупповая расширенная операция, например нечеткое умножение, нечеткое сложение и т. д.

Для лингвистического моделирования динамического процесса эти правила можно представить следующим образом [18]:

П_i: ЕСЛИ x_1 есть $x_1(t)$ И ... И x_j есть $x_j(t)$ И ... И x_m есть $x_m(t)$,
ТО y_1 есть $x_1(t)*B_{i1}$ И ... И y_k есть $x_j(t)*B_{ik}$ И ... И y_m есть $x_m(t)*B_{im}$,

$$i = 1, \dots, n.$$
(4.57)

Правила, основанные на принципе обобщения, базируются на введении групповой нечеткой расширенной операции и могут использоваться (как и правила предыдущего типа) при решении задач на графах, при планировании расписаний [18], при аккумулировании влияния нескольких элементов (факторов) на один [19].

Правила, основанные на нечетких множествах с адаптацией операций над ними, где операции над нечеткими высказываниями в заключениях правил выбираются (адаптируются) в зависимости, например, от степени согласованности (значимости) нечетких высказываний в предпосылках [20]:

П_i: ЕСЛИ x_1 есть A_{11} И ... И x_m есть A_{m1} ,
ТО y есть $(A_{11} \theta_1^i A_{12}) \Theta_i^i, \dots, (A_{(m-1)1} \theta_q^i A_{m1}), i = 1, \dots, n,$
(4.58)

где $\theta_1^1, \dots, \theta_q^1, \dots, \theta_1^n, \dots, \theta_q^n$ – операции свертки нечетких множеств, выбираемые в зависимости от уровня совместимости пары соответствующих нечетких высказываний в предпосылке; Θ_i^i – операция комбинирования результатов парных сверток нечетких множеств этих высказываний в предпосылке в i -м правиле ($i = 1, \dots, n$), выбираемая из соответствующей совокупности операций парных сверток $\theta_1^i, \dots, \theta_q^i$ и характеризующая, например, нижний уровень их согласования.

Правила данного типа предназначены для построения оценочных моделей (моделей предпочтения ЛПР) в многоцелевых системах поддержки принятия решений [15].

Нечеткие продукционные правила с заключениями в виде нечетких значений или функций

В нечетких продукционных правилах этой группы антецеденты представляют собой нечеткие высказывания, которые могут быть сформированы так же, как и антецеденты правил одного из рассмотренных выше

типов. А способы формирования консеквентов этих правил являются следующими:

- консеквенты правил представляют собой одноточечные нечеткие множества (singleton fuzzy set) [21];
- консеквенты правил представляют собой четкие функции (crisp function) [12].

Рассмотрим подробнее нечеткие продукционные правила указанных типов на примере MISO-структурь (Multi Inputs – Single Output).

Нечеткие продукционные правила, консеквенты которых представляют собой одноточечные нечеткие множества (singletons – синглетоны), можно представить следующим образом:

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y_i = c_i, i = 1, \dots, n, \quad (4.59)$$

где c_i – действительные числа.

Нечеткие продукционные правила, консеквенты которых представляют собой четкие функции. Существуют различные варианты построения нечетких продукционных правил данного типа, основными из которых являются правила, положенные в основу нечетких продукционных моделей Цукамото (Tsukamoto) и Такаги–Сугэно (Takagi–Sugeno) (см. ниже).

Так, для модели Цукамото в заключениях правил используются монотонные (возрастающие или убывающие) функции f_i . А заключение по каждому из правил формируется путем обратного преобразования этих функций по полученным значениям предпосылок данных правил, например:

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y = f_i^{-1}(\alpha_i), i = 1, \dots, n, \quad (4.60)$$

где α_i – уровень срабатывания антецедента i -го правила.

Для нечеткой продукционной модели Такаги–Сугэно правила можно представить следующим образом:

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y = f_i(x_1, \dots, x_j, \dots, x_m), i = 1, \dots, n, \quad (4.61)$$

где $f_i(x_1, \dots, x_j, \dots, x_m)$ – функция, имеющая одинаковую для всех n правил структуру и различающаяся лишь параметрами для каждого правила.

Наибольшее практическое применение нашел вариант использования линейной функции

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y = c_{i1}x_1 + \dots + c_{ij}x_j + \dots + c_{im}x_m + c_{i0}, i = 1, \dots, n, \quad (4.62)$$

где c_{ij} ($j = 1, \dots, m$) – коэффициенты аргументов функции; c_{i0} – смещение.

Нечеткая производственная модель, основанная на правилах (4.62), называется *аффинной моделью Такаги–Сугэно* [21].

Можно отметить, что при $c_{ij} = 0$ ($j = 1, \dots, n$) данная модель сводится к модели (4.59), другое название которой – модель Такаги–Сугэно 0-го порядка.

4.3.4. Задание структуры базы нечетких производственных правил

В зависимости от количества нечетких высказываний в предпосылках и заключениях база правил нечеткой производственной модели может быть представлена структурой одного из следующих типов:

- SISO-структура (Single Input – Single Output, один вход – один выход);
- MISO-структура (Multi Inputs – Single Output, много входов – один выход);
- MIMO-структура (Multi Inputs – Multi Outputs, много входов – много выходов).

Представленная ниже MIMO-структура является наиболее общим видом структуры базы правил типа (4.51) нечеткой производственной модели:

$$\begin{aligned} \Pi_i: & \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ & \text{ТО } y_1 \text{ есть } B_{i1} \text{ И } \dots \text{ И } y_k \text{ есть } B_{ik} \text{ И } \dots, y_p \text{ есть } B_{ip}, i = 1, \dots, n, \end{aligned} \quad (4.63)$$

где x_j ($j = 1, \dots, m$) – входные переменные, которые могут быть как четкими, так и нечеткими, $x_j \in X_j$, X_j – область определения соответствующей предпосылки; y_k ($k = 1, \dots, p$) – нечеткие выходные переменные, $y_k \in Y_k$, Y_k – область определения соответствующего заключения; A_{ij} , B_{ik} – нечеткие множества, определенные на X_j и Y_k с функциями принадлежности $\mu_{A_{ij}}(x_j) \in [0, 1]$ и $\mu_{B_{ik}}(x_k) \in [0, 1]$ соответственно.

При выполнении гипотезы о взаимной независимости выходных переменных база нечетких производственных правил с MIMO-структурой и *конъюнктивной формой заключения* может быть представлена совокупностью (p) баз правил с MISO-структурой со многими (m) входными переменными и одной выходной переменной. В этом случае правила будут иметь следующий вид:

$$\begin{aligned} \Pi_i: & \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ & \text{ТО } y \text{ есть } B_i, i = 1, \dots, n. \end{aligned} \quad (4.64)$$

Если множество входных переменных x_j ($j = 1, \dots, m$) является независимым, то входное пространство предпосылки может быть разделено по каждой из переменной одномерными функциями принадлежности,

например для переменной $x_j \in X_j$ – функциями принадлежности нечетких множеств $\{A_{1j}, \dots, A_{ij}, \dots, A_{nj}\}$. На рис. 4.1, а показан пример равномерного разбиения входного 2d-пространства предпосылки: функциями принадлежности $\mu_{A_{11}}(x_1), \mu_{A_{12}}(x_1), \mu_{A_{13}}(x_1)$ для входной переменной x_1 ; функциями принадлежности $\mu_{A_{21}}(x_2), \mu_{A_{22}}(x_2), \mu_{A_{23}}(x_2)$ для входной переменной x_2 .

В базе нечетких правил с предпосылками, представленными в конъюнктивной форме, все пространство входных переменных делится на решетку нечетких гиперкубов параллельно осям координат. Каждый из гиперкубов является декартовым произведением соответствующих одномерных нечетких множеств (рис. 4.1, а).

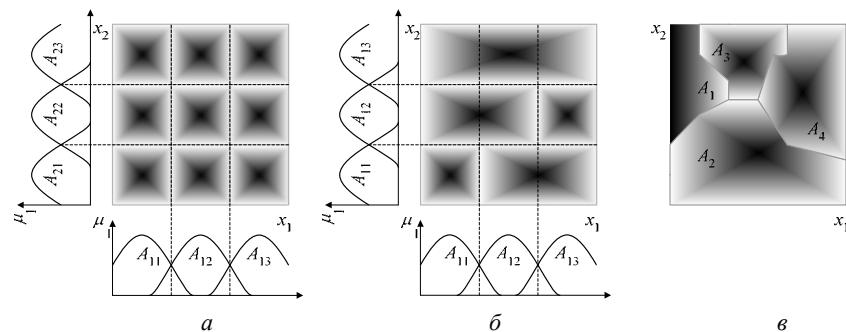


Рис. 4.1. Различные способы деления пространства предпосылок

При этом правила формируются таким образом, чтобы получить все возможные комбинации функций принадлежности всех входных переменных.

Число нечетких правил с конъюнктивной формой предпосылки, необходимых для покрытия пространства входных переменных, определяется следующим образом:

$$n = \prod_{j=1}^m K_j, \quad (4.65)$$

где m – размерность пространства входных признаков; K_j – число лингвистических термов j -й входной переменной.

Так, для приведенного на рис. 4.1, а примера можно записать следующие предпосылки девяти правил:

Π_1 : ЕСЛИ x_1 есть A_{11} И x_2 есть A_{21} , ТО ... ,
 Π_2 : ЕСЛИ x_1 есть A_{11} И x_2 есть A_{22} , ТО ... ,

...
 Π_9 : ЕСЛИ x_1 есть A_{13} И x_2 есть A_{23} , ТО

С целью сокращения числа нечетких правил для покрытия пространства входных переменных могут быть получены различные разбиения пространства предпосылок посредством комбинирования нечетких операций конъюнкции, дизъюнкции и дополнения над одномерными функциями принадлежности (рис. 4.1, б). Но при этом границы такого разбиения ограничены прямоугольной сеткой, заданной нечеткими множествами отдельных входных переменных.

Однако входные переменные не всегда являются взаимно независимыми. Границы между областями, в которых определены лингвистические термы входных переменных, могут быть произвольно изогнуты и не соответствовать осям (рис. 4.1, в). Решением этой проблемы является представление всех входных переменных в виде вектора, а предпосылки – на основе многомерных функций принадлежности

$$\Pi_i: \text{ЕСЛИ } \mathbf{x} \text{ есть } A_i, \text{ ТО } \dots, \quad (4.66)$$

где $\mathbf{x} = [x_1, \dots, x_m]^T$; A_i – нечеткое множество с многомерной функцией принадлежности.

Такое представление предпосылки на основе многомерных функций принадлежности (4.1) является наиболее общим, так как в этом случае не накладывается никаких ограничений на формы областей, в которых определены лингвистические термы входных переменных.

Для сложных моделей данное разбиение может обеспечить наиболее эффективное представление, так как число нечетких множеств, необходимых для покрытия пространства входных переменных, в этом случае может быть существенно уменьшено по сравнению с примерами на рис. 4.1, а, б.

Другим путем уменьшения сложности нечетких моделей с большим количеством входных переменных является их декомпозиция на модели с меньшим числом входных переменных в базе правил с последующим их объединением в однородную или иерархическую (многослойную) структуру. В этом случае выход одной базы правил является входом другой, как это показано на рис. 4.2.

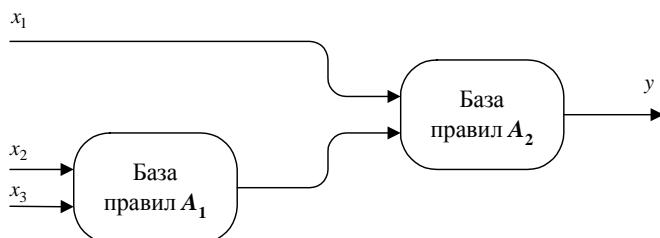


Рис. 4.2. Каскадное соединение двух баз нечетких продукционных правил

Такое каскадное соединение позволяет существенно уменьшить общее количество нечетких правил. В качестве примера предположим, что каждая входная переменная описывается пятью лингвистическими терминами при условии конъюнктивной формы антецедентов. После декомпозиции трехходовой базы правил, которая состоит из $5^3 = 125$ правил, каждая двухходовая база правил будет содержать лишь $5^2 = 25$ правил [21].

4.3.5. Обеспечение полноты и непротиворечивости базы нечетких правил

При создании нечеткой продукционной модели системы могут быть использованы как априорные данные о моделируемой системе, поступающие от экспертов, так и данные, полученные в результате измерений.

В первом случае, если отсутствует необходимость согласования мнений экспертов, предполагается, что задача обеспечения полноты и непротиворечивости базы нечетких правил решена заранее. В случае же, если известны только экспериментальные данные, данную задачу можно отнести к задачам идентификации системы. На практике может иметь место также смешанный случай, когда начальная база нечетких правил строится исходя из эвристических предположений, а ее уточнение проводится с использованием экспериментальных данных. Каждому из этих подходов соответствуют свои алгоритмы.

Рассмотрим основные этапы типичного алгоритма формирования базы нечетких правил заданной структуры в случае, если применяются только экспериментальные данные (обучающая выборка).

Допустим, необходимо создать базу нечетких правил типа (4.64) с MISO-структурой: с двумя входными (x_1, x_2) и одной выходной (y) переменными. При этом задана обучающая выборка, состоящая из множества примеров следующего вида:

$$(x_1^{(k)}, x_2^{(k)}, y^{(k)}), k = 1, \dots, K, \quad (4.67)$$

где $x_1^{(k)}, x_2^{(k)}, y^{(k)}$ – соответственно значения входных переменных x_1, x_2 и выходной переменной y в k -м примере; K – общее число примеров в обучающей выборке.

Этап 1. Разбиение пространств входных и выходных переменных. Пусть известны минимальные и максимальные значения каждой переменной: $x_1 \in [x_1^{(\min)}, x_1^{(\max)}]$, $x_2 \in [x_2^{(\min)}, x_2^{(\max)}]$, $y \in [y^{(\min)}, y^{(\max)}]$.

Разобъем области определений этих переменных на отрезки. Причем число этих отрезков, а также их длина для каждой переменной подбираются индивидуально. На рис. 4.3 показан пример такого разбиения, при усло-

вии, что число отрезков для $[x_1^{(\min)}, x_1^{(\max)}]$ равно трем, для $[x_2^{(\min)}, x_2^{(\max)}]$ – пяти, для $[y^{(\min)}, y^{(\max)}]$ – четырем. На каждом из отрезков задана одна функция принадлежности, например треугольной формы с вершиной в центре отрезка.

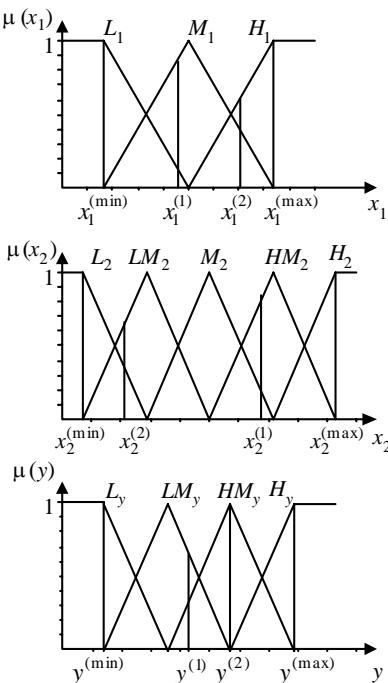


Рис. 4.3. Пример разделения пространств входных переменных x_1, x_2 и выходной переменной y

Для x_1 определены нечеткие множества с лингвистическими значениями $\{L_1\text{ – низкое}, M_1\text{ – среднее}, H_1\text{ – высокое}\}$, для x_2 – $\{L_2, LM_2\text{ – между низким и средним}, M_2, HM_2\text{ – между высоким и средним}, H_2\}$, для y – $\{L_y, LM_y, HM_y, H_y\}$.

Функции принадлежности подбираются таким образом, чтобы они перекрывались (overlap) на уровне 0.5. Следует отметить, что как вид функций принадлежности, так и способ разбиения пространств входных и выходных переменных могут быть иными.

Этап 2. Формирование начальной базы правил. Можно предложить два не противоречащих друг другу подхода к формированию начальной базы правил.

Первый подход основан на генерации множества правил исходя из возможных сочетаний нечетких высказываний в предпосылках и заключениях правил, в соответствии с которым максимальное количество правил в базе определяется следующим соотношением:

$$l = l_1 \cdot l_2 \cdots l_m \cdot l_y, \quad (4.68)$$

где l_1, l_2, \dots, l_m – число функций принадлежности для задания переменных x_1, x_2, \dots, x_m ; l_y – для выходной переменной y .

Таким образом, изначально сформированная для приведенного на рис. 4.3 примера база нечетких правил будет состоять из 60 правил и иметь следующий вид:

Π_1 : **ЕСЛИ** x_1 есть L_1 И ... И x_2 есть L_2 , **ТО** y есть L_y ,

...

Π_q : **ЕСЛИ** x_1 есть M_1 И ... И x_2 есть HM_2 , **ТО** y есть LM_y ,

...

Π_g : **ЕСЛИ** x_1 есть H_1 И ... И x_2 есть LM_2 , **ТО** y есть HM_y ,

...

Π_{60} : **ЕСЛИ** x_1 есть H_1 И ... И x_2 есть H_2 , **ТО** y есть H_y ,

Другой подход к формированию начальной базы правил основан на том, что изначально каждому примеру из выборки ставится в соответствие отдельное правило. Для этого для каждого примера ($x_1^{(k)}, x_2^{(k)}, y^{(k)}$), $k = 1, \dots, K$, определяются степени принадлежности заданных значений переменных к соответствующим нечетким множествам (см. рис. 4.3):

- для $x_1^{(1)}$: $\mu_{L_1}(x_1^{(1)}) = 0.1$; $\mu_{M_1}(x_1^{(1)}) = 0.9$; $\mu_{H_1}(x_1^{(1)}) = 0.1$;
- для $x_2^{(1)}$: $\mu_{L_2}(x_2^{(1)}) = 0$; $\mu_{LM_2}(x_2^{(1)}) = 0$; $\mu_{M_2}(x_2^{(1)}) = 0.2$; $\mu_{HM_2}(x_2^{(1)}) = 0.9$;
 $\mu_{H_2}(x_2^{(1)}) = 0$;
- для $y^{(1)}$: $\mu_{L_y}(y^{(1)}) = 0$; $\mu_{LM_y}(y^{(1)}) = 0.7$; $\mu_{HM_y}(y^{(1)}) = 0.3$; $\mu_{H_y}(y^{(1)}) = 0$.

После чего каждому обучающему примеру ставится в соответствие те нечеткие множества, степени принадлежности к которым у соответствующих значений переменных из этого примера являются максимальными. В рассмотренном выше примере $x_1^{(1)}$ ставится в соответствие нечеткое множество M_1 , $x_2^{(1)} - HM_2$, $y^{(1)} - LM_y$.

Таким образом, примеру ($x_1^{(1)}, x_2^{(1)}, y^{(1)}$) из обучающей выборки ставится в соответствие следующее правило:

ЕСЛИ x_1 есть M_1 И ... И x_2 есть HM_2 , **ТО** y есть LM_y .

Аналогично примеру ($x_1^{(2)}, x_2^{(2)}, y^{(2)}$) ставится в соответствие правило:

ЕСЛИ x_1 есть H_1 И ... И x_2 есть LM_2 , **ТО** y есть HM_y .

Сформированное таким образом множество правил и составляет начальную базу нечетких правил.

Следует отметить, что использование первого подхода при формировании начальной базы правил целесообразно при небольшом числе переменных и функций принадлежности, используемых для задания этих переменных. Второй подход стоит использовать при сравнительно небольшом количестве примеров в обучающей выборке.

Этап 3. Определение рейтингов правил. Поскольку изначально сформированная база правил наверняка является избыточной, а также может содержать противоречие друг другу правила – с одинаковыми предпосылками и разными заключениями, то набор правил необходимо оптимизировать. Сделать это можно как на основе эмпирических гипотез (информации от экспертов), так и путем адаптации к имеющимся экспериментальным данным (обучающей выборке), что приводит к существенному уменьшению количества и к ликвидации противоречивости правил, оставляемых в базе.

Рассмотрим один из простых подходов к сокращению базы нечетких правил на основе экспериментальных данных [22].

Допустим набор экспериментальных данных (обучающая выборка) в полной мере характеризует особенности моделируемой системы.

Все примеры из обучающей выборки ($x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)}$) ($k = 1, \dots, K$) «предъявляются» каждому правилу. В результате для каждого правила определяется его рейтинг:

$$r_i = \sum_{k=1}^K \mu_{A_{i1}}(x_1^{(k)}) \cdot \mu_{A_{i2}}(x_2^{(k)}) \cdots \mu_{A_{im}}(x_m^{(k)}) \cdot \mu_{B_i}(y^{(k)}), i = 1, \dots, n. \quad (4.69)$$

Этап 4. Сокращение числа правил. После подсчета рейтингов правил из базы правил исключаются правила с наименьшими значениями r_i . При этом в первую очередь сокращения проводятся по группам правил, имеющим одинаковые предпосылки и разные заключения, т. е. разные функции принадлежности переменной вывода. Такие правила являются противоречивыми друг другу, и из группы подобных правил необходимо оставлять только одно (с наибольшим рейтингом). Таким образом, решается проблема противоречивых правил, а также существенно уменьшается их общее число. Оставшиеся правила формируют итоговую базу правил.

Этап 5. Адаптация параметров оставленных в базе правил. База правил может считаться окончательно сформированной, если осуществлен этап адаптации оставленных в ней правил. Этот этап сводится к нахождению, в соответствии с имеющимися экспериментальными данными и принятым критерием, оптимальных значений параметров для оставленных в базе правил. Этот этап, по сути, является этапом параметрической оптимизации конечного набора правил. Он заключается в таком изменении параметров функций принадлежности оставшихся правил, при котором обеспечиваются максимальные «степени активности» этих правил по всем примерам обучающей выборки.

Для рассматриваемого на рис. 4.3 случая, когда все функции принадлежности нечетких множеств являются треугольными, настройке подвергается один их параметр – значения мод.

Примечания: 1. Для выполнения этого этапа необходимо синтезировать нечеткую модель. Поэтому вопросы адаптации параметров правил, оставленных в базе, будут рассмотрены далее при описании параметрической оптимизации базы нечетких правил (см. п. 4.8), а также в подразделах, посвященных настройке и обучению нечетких нейронных производственных сетей.

2. Очевидно, что в зависимости от конкретной постановки задачи формирования базы нечетких производственных правил могут быть изменены как перечень, так и содержание решаемых в ходе создания базы нечетких правил этапов.

4.4. Введение нечеткости

Под введением нечеткости (*fuzzification*) в модель с использованием нечетких производственных правил, например типа (4.64), понимается процедура получения значений функций принадлежности нечетких множеств A_{ij} по значениям всех соответствующих входных переменных x_j для всех предпосылок всех нечетких производственных правил.

Примеры. На рис. 4.4 показаны примеры введения нечеткости с использованием четкого x'_j и нечеткого \tilde{x}_j значений входной переменной.

В случае четкого значения входной переменной степень истинности нечеткого высказывания « x'_j есть A_{ij} » определяется значением функции принадлежности $\mu_{A_{ij}}(x'_j)$. При задании же нечеткого значения входной переменной \tilde{x}_j степень истинности соответствующего нечеткого высказывания « \tilde{x}_j есть A_{ij} » в предпосылке определяется на основе одной из операций нечеткой конъюнкции (4.42)–(4.46). Наиболее часто для этого

используется операция min-конъюнкции $\mu_{A'_{ij}}(x_j) = \min\{\mu_{\tilde{x}_j}(x_j), \mu_{A_{ij}}(x_j)\}$, $\forall x_j \in X_j$ (рис. 4.4, *б*) или операция алгебраического произведения $\mu_{A'_{ij}}(x_j) = \mu_{\tilde{x}_j}(x_j) \mu_{A_{ij}}(x_j)$, $\forall x_j \in X_j$ (рис. 4.4, *в*).

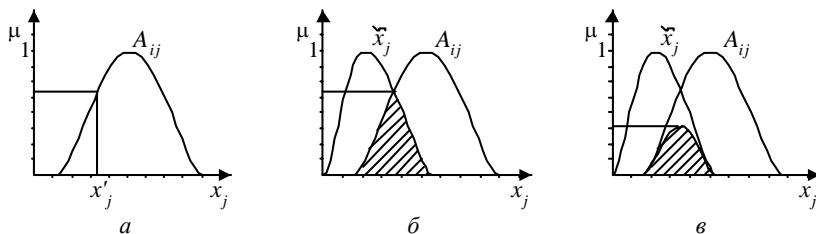


Рис. 4.4. Примеры введения нечеткости с использованием значений входной:
а – переменной четкого x'_j ; *б, в* – нечеткого \tilde{x}_j

Полученное при этом численное значение функции принадлежности [$\mu_{A_{ij}}(x'_j)$ или $\mu_{A'_{ij}}(x_j)$] является результатом выполнения нечеткого вычисления в предпосылке правила и характеризует степень его истинности.

4.5. Агрегирование степени истинности предпосылок правил

В результате данной процедуры определяется агрегированная степень истинности по всем предпосылкам каждого правила: α_i ($i = 1, \dots, n$).

Если в правиле, как, например, в (4.64), имеется несколько предпосылок, то на предыдущем этапе в результате введения нечеткости определяются численные значения функций принадлежности по каждой из этих предпосылок, например для четких значений входных переменных: $\mu_{A_{i1}}(x'_1), \dots, \mu_{A_{ij}}(x'_j), \dots, \mu_{A_{im}}(x'_m)$; либо при задании нечетких значений входных переменных: $\mu_{A'_{i1}}(x_1), \dots, \mu_{A'_{ij}}(x_j), \dots, \mu_{A'_{im}}(x_m)$. Затем эти значения агрегируются в зависимости от используемых в правилах нечетких логических связок между предпосылками.

Так, например, при использовании связки «И» между значениями функций принадлежности предпосылок правил $\mu_{A_{ij}}(x'_j)$ и $\mu_{A_{im}}(x'_m)$ может быть применено одно из следующих выражений:

- *min*-конъюнкция степеней истинности предпосылок правил

$$\mu_{A_{ij} \wedge A_{im}}(x) = \min\{\mu_{A_{ij}}(x'_j), \mu_{A_{im}}(x'_m)\}; \quad (4.70)$$

- алгебраическое произведение степеней истинности предпосылок правил

$$\mu_{A_{ij} \cdot A_{im}}(x) = \mu_{A_{ij}}(x'_j) \cdot \mu_{A_{im}}(x'_m); \quad (4.71)$$

- граничное произведение степеней истинности предпосылок правил

$$\mu_{A_{ij} \cdot A_{im}}(x) = \max\{\mu_{A_{ij}}(x'_j) + \mu_{A_{im}}(x'_m) - 1, 0\}; \quad (4.72)$$

- драстическое произведение степеней истинности предпосылок правил

$$\mu_{A_{ij} \Delta A_{im}}(x) = \begin{cases} \mu_{A_{im}}(x'_m), & \text{если } \mu_{A_{ij}}(x'_j) = 1, \\ \mu_{A_{ij}}(x'_j), & \text{если } \mu_{A_{im}}(x'_m) = 1, \\ 0, & \text{в остальных случаях.} \end{cases} \quad (4.73)$$

В случае же использования связки «ИЛИ» между значениями функций принадлежности предпосылок правил $\mu_{A_{ij}}(x'_j)$ и $\mu_{A_{im}}(x'_m)$ может быть взято одно из выражений:

- max-дизъюнкция степеней истинности предпосылок правил

$$\mu_{A_{ij} \vee A_{im}}(x) = \max\{\mu_{A_{ij}}(x'_j), \mu_{A_{im}}(x'_m)\}; \quad (4.74)$$

- алгебраическая сумма степеней истинности предпосылок правил

$$\mu_{A_{ij} + A_{im}}(x) = \mu_{A_{ij}}(x'_j) + \mu_{A_{im}}(x'_m) - \mu_{A_{ij}}(x'_j) \cdot \mu_{A_{im}}(x'_m); \quad (4.75)$$

- граничная сумма степеней истинности предпосылок правил

$$\mu_{A_{ij} \oplus A_{im}}(x) = \min\{\mu_{A_{ij}}(x'_j) + \mu_{A_{im}}(x'_m), 1\}; \quad (4.76)$$

- драстическая сумма степеней истинности предпосылок правил

$$\mu_{A_{ij} \Delta A_{im}}(x) = \begin{cases} \mu_{A_{im}}(x'_m), & \text{если } \mu_{A_{ij}}(x'_j) = 0, \\ \mu_{A_{ij}}(x'_j), & \text{если } \mu_{A_{im}}(x'_m) = 0, \\ 1, & \text{в остальных случаях.} \end{cases} \quad (4.77)$$

При определении агрегированной степени истинности по всем предпосылкам каждого правила α_i ($i = 1, \dots, n$), с использованием как связок «И», так и связок «ИЛИ», чаще всего используют попарно согласованные операции из одного и того же базиса, например: min-конъюнкция и max-дизъюнкция; алгебраическое произведение и алгебраическая сумма; граничное произведение и граничная сумма; драстическое произведение и драстическая сумма.

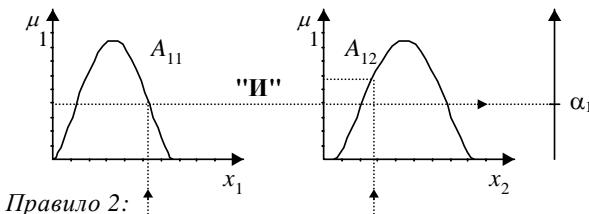
Пример. На рис. 4.5 приведен пример агрегирования степеней истинности предпосылок двух правил:

П1: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} , ТО ... ,

П2: ЕСЛИ x_1 есть A_{21} ИЛИ x_2 есть A_{22} , ТО ...

При этом в качестве связки «И» между предпосылками первого правила использована min-конъюнкция, а в качестве связки «ИЛИ» между предпосылками второго правила – max-дизъюнкция.

Правило 1:



Правило 2:

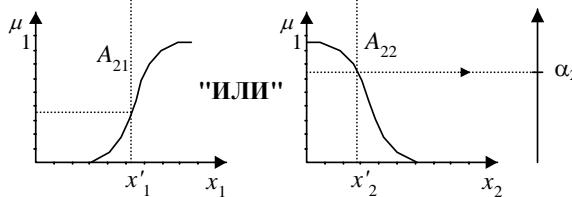


Рис. 4.5. Пример агрегирования степеней истинности предпосылок правил с использованием min-конъюнкции «И» и max-дизъюнкции «ИЛИ»

Агрегированные степени истинности предпосылок этих правил определяются следующим образом:

$$\alpha_1 = \min\{\mu_{A_{11}}(x'_1), \mu_{A_{12}}(x'_2)\},$$

$$\alpha_2 = \max\{\mu_{A_{21}}(x'_1), \mu_{A_{22}}(x'_2)\}.$$

4.6. Активизация заключений правил

Процедура активизации (activation) заключений нечетких продукционных правил состоит в определении модифицированных функций принадлежности этих заключений для каждого i -го ($i = 1, \dots, n$) правила $\mu_{B'_i}(y)$ на основе выполнения композиционной операции, модифицированной для нечеткой продукции, между определенным на предыдущем

этапе агрегированным значением степеней истинности предпосылок этого правила α_i и соответствующей функцией принадлежности его заключения $\mu_{B_i}(y)$.

В качестве такой операции распространение получили следующие модификации соответствующих правил нечеткой композиции [7]:

- *min-активизация*

$$\mu_{B'_i}(y) = \min\{\alpha_i, \mu_{B_i}(y)\}; \quad (4.78)$$

- *prod-активизация*

$$\mu_{B'_i}(y) = \alpha_i \mu_{B_i}(y); \quad (4.79)$$

- *average-активизация*

$$\mu_{B'_i}(y) = 0.5 (\alpha_i + \mu_{B_i}(y)). \quad (4.80)$$

Затем полученные таким образом результаты корректируются путем их алгебраического произведения на весовые коэффициенты соответствующих правил. Если эти коэффициенты не заданы, то предполагается, что они равны единице.

При наличии нескольких (p) заключений в нечетких продукционных правилах [например, в правилах типа (4.51)], весовые коэффициенты могут быть заданы не только для отдельных правил, но и индивидуально для каждого их заключения.

Примечание. Коррекция базы нечетких правил путем умножения на весовые коэффициенты является альтернативой способу параметрической оптимизации этой базы за счет подстройки параметров (формы и расположения) функций принадлежности нечетких правил [23].

Пример. Рассмотрим пример активизации двух заключений для каждого из двух правил на основе операции *min-активизации* в предположении, что весовые коэффициенты правил равны единице (рис. 4.6).

П₁: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} , ТО y_1 есть B_{11} И y_2 есть B_{12} ,

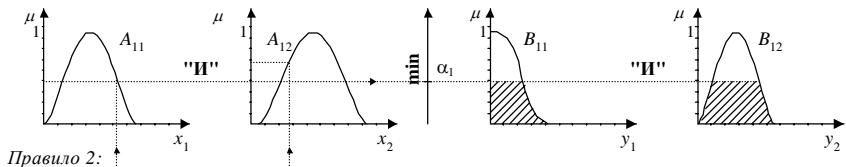
П₂: ЕСЛИ x_1 есть A_{21} ИЛИ x_2 есть A_{22} , ТО y_1 есть B_{21} И y_2 есть B_{22} .

В результате данной процедуры находятся модифицированные функции принадлежности заключений рассматриваемых правил:

$$\mu_{B'_{11}}(y_1) = \min\{\alpha_1, \mu_{B_{11}}(y_1)\}, \quad \mu_{B'_{12}}(y_2) = \min\{\alpha_1, \mu_{B_{12}}(y_2)\},$$

$$\mu_{B'_{21}}(y_1) = \min\{\alpha_2, \mu_{B_{21}}(y_1)\}, \quad \mu_{B'_{22}}(y_2) = \min\{\alpha_2, \mu_{B_{22}}(y_2)\}.$$

Правило 1:



Правило 2:

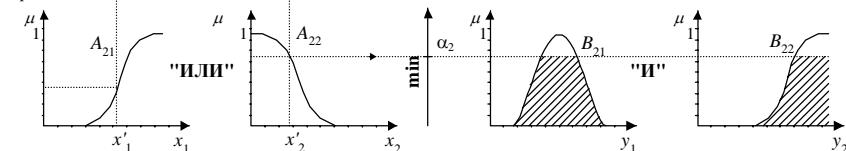


Рис. 4.6. Пример активизации заключений правил на основе операции \min -активизации

4.7. Аккумулирование активизированных заключений правил

После получения активизированных заключений для каждой выходной переменной каждого из нечетких продукционных правил выполняется процедура их аккумулирования (accumulation).

Результат такого аккумулирования для выходных переменных находится путем объединения полученных на предыдущем этапе соответствующих нечетких множеств по одной из формул (4.46)–(4.50).

Пример. На рис. 4.7 показано аккумулирование активизированных заключений правил типа (4.51) в соответствии с примером из п. 4.5 с использованием операции \max -дизъюнкции.

В итоге формируются нечеткие множества для выходных переменных с функциями принадлежности:

$$\mu_{B'_1}(y_1) = \max\{\mu_{B'_{11}}(y_1), \mu_{B'_{12}}(y_1)\},$$

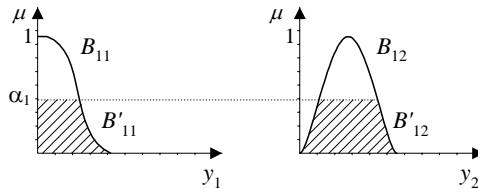
$$\mu_{B'_2}(y_2) = \max\{\mu_{B'_{21}}(y_1), \mu_{B'_{22}}(y_1)\}.$$

4.8. Приведение к четкости

Приведение к четкости (defuzzification) заключается в преобразовании нечетких значений найденных выходных переменных в четкие. При этом все методы получения четкого значения выходной переменной можно разделить на две группы:

- методы дефазификации аккумулированной на предыдущем этапе (из активизированных заключений всех правил базы) выходной переменной;
- методы дефазификации выходной переменной без предварительного аккумулирования активизированных заключений отдельных правил.

Заключения правила 1:



Заключения правила 2:

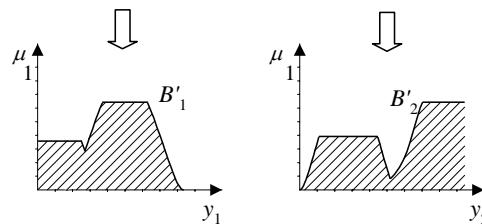
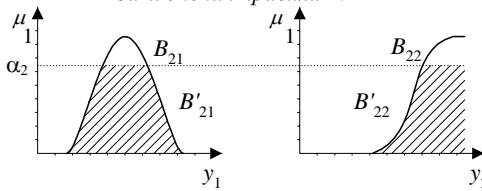


Рис. 4.7. Пример аккумулирования двух заключений правил с использованием тах-дизъюнкции

К первой группе относятся следующие методы дефазификации:

1. Центр тяжести (centre of gravity) (рис. 4.8, а).

Этот метод дефазификации [как, впрочем, и все остальные, кроме метода из п.11] может быть использован только для моделей, основанных на нечетких лингвистических производственных правилах, в которых консеквенты являются нечеткими высказываниями, например типа (4.64).

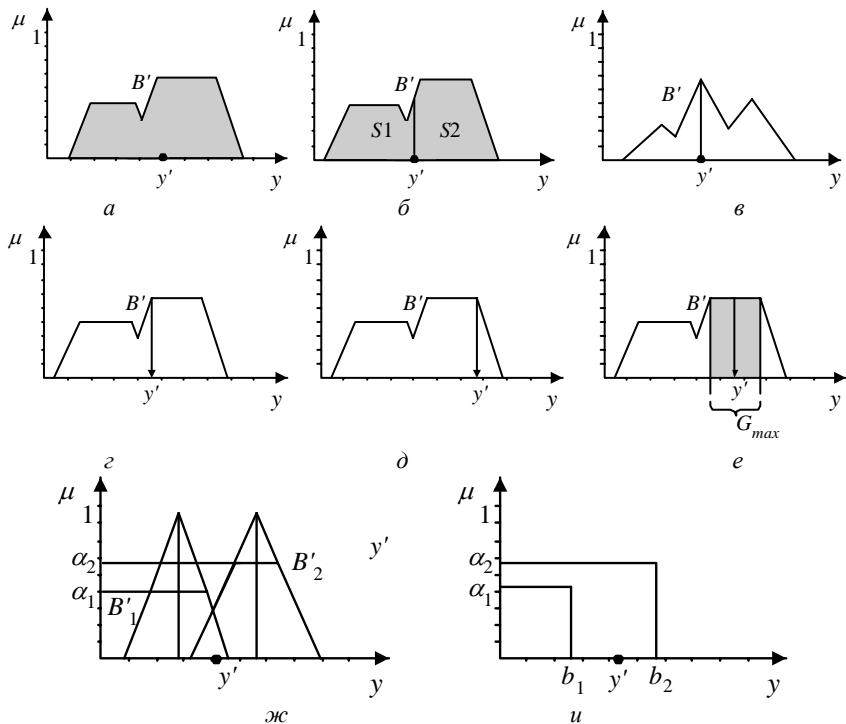


Рис. 4.8. Примеры применения методов дефазификации выходной переменной

Четкое значение y' выходной переменной рассчитывается как центр тяжести функции принадлежности $\mu_{B'}(y)$ и вычисляется по формуле

$$y' = \frac{\int_{Y_{\min}}^{Y_{\max}} y \mu_{B'}(y) dy}{\int_{Y_{\min}}^{Y_{\max}} \mu_{B'}(y) dy}, \quad (4.81)$$

где Y_{\min} , Y_{\max} – границы интервала носителя нечеткого множества выходной переменной y .

Для дискретизированной для вычисления выходной переменной области Y выражение (4.81) примет следующий вид:

$$y' = \frac{\sum_{r=1}^{Y_{\max}} y_r \mu_{B'}(y_r)}{\sum_{r=1}^{Y_{\max}} \mu_{B'}(y_r)}, \quad (4.82)$$

где Y_{\max} – число элементов y_r в дискретизированной для вычисления «центра тяжести» области Y .

2. Центр площади (centre of area) (рис. 4.8, б). Четкое значение выходной переменной y' по этому методу определяется из уравнения

$$\underbrace{\int_{Y_{\min}}^{y'} \mu_{B'}(y) dy}_{S1} = \underbrace{\int_{y'}^{Y_{\max}} \mu_{B'}(y) dy}_{S2}. \quad (4.83)$$

3. Максимум функции принадлежности (рис. 4.8, в). Четкое значение выходной переменной y' рассчитывается по формуле

$$y' = \arg \sup_y \mu_{B_i}(y), \quad (4.84)$$

где $\mu_{B_i}(y)$ – унимодальная функция, форма которой может быть произвольной.

4. Первый максимум (first-of-maxima), называемый также самым левым максимумом (left most maximum) (рис. 4.8, г). Четкое значение y' находится как наименьшее значение, при котором достигается максимум итогового нечеткого множества:

$$y' = \min \{ y_{\max} \mid \mu_{B'}(y_{\max}) = \max_y \mu_{B'}(y) \}. \quad (4.85)$$

5. Самый правый максимум (right most maximum) (рис. 4.8, д). Четкое значение выходной переменной y' находится как наибольшее значение, при котором достигается максимум итогового нечеткого множества:

$$y' = \max \{ y_{\max} \mid \mu_{B'}(y_{\max}) = \max_y \mu_{B'}(y) \}. \quad (4.86)$$

6. Средний максимум (middle-of-maxima) (рис. 4.8, е). Четкое значение выходной переменной y' находится в соответствии с выражением

$$y' = \frac{\int_{G_{\max}}^y dy}{G_{\max}}, \quad (4.87)$$

где G_{\max} – интервал носителя нечеткого множества B' (подмножество его элементов), в котором $\mu_{B'}(y)$ принимает максимальное значение.

Для дискретного варианта $\mu_{B'}(y)$:

$$y' = \frac{1}{G_{\max}} \sum_{G_{\max}} \max_y \mu_{B'}(y), \quad (4.88)$$

где G_{\max} – число элементов носителя нечеткого множества B' , в которых $\mu_{B'}(y)$ принимает максимальное значение.

7. Критерий максимума (max-criterion). Четкое значение выходной переменной y' выбирается произвольно среди множества значений y , для которых значение $\mu_{B'}(y)$ достигает максимума:

$$y' \in \{y_{\max} \mid \mu_{B'}(y_{\max}) = \max_y \mu_{B'}(y)\}. \quad (4.89)$$

8. Высотная дефазификация (height defuzzification). Элементы носителя нечеткого множества B' , для которых значения функции принадлежности меньше чем некоторый заданный уровень λ ($\mu_{B'}(y) < \lambda$), в расчет не принимаются, и дефазифицированное значение выходной переменной y' рассчитывается в соответствии со следующим выражением:

$$y' = \frac{\int y \mu_{B'}(y) dy}{\int_{G_{\lambda}} \mu_{B'}(y) dy}, \quad (4.90)$$

где G_{λ} – интервал носителя нечеткого множества B' , в котором $\mu_{B'}(y) < \lambda$.

Ко второй группе относятся следующие методы дефазификации:

9. Средний центр (centre average defuzzification). Четкое значение выходной переменной y' в нечеткой производной модели, основанной на правилах типа (4.64), рассчитывается по формуле

$$y' = \frac{\sum_{i=1}^n \alpha_i \arg \max_y \mu_{B'_i}(y)}{\sum_{i=1}^n \alpha_i}, \quad (4.91)$$

где $\arg \max_y \mu_{B_i}(y)$ – значение y , при котором $\mu_{B_i}(y)$ принимает максимальное значение, т. е. $\mu_{B_i}(y) = \max_y \mu_{B_i}(y)$.

Данный метод может быть использован в случае унимодальных $\mu_{B_i}(y)$, т. е. если условие $\mu_{B_i}(y) = 1$ справедливо для одной точки универсума Y_i .

На рис. 4.8, в приведен пример реализации этого метода для числа правил $n = 2$.

10. Сумма центров (centre of sums defuzzification).

$$y' = \frac{\int_{Y_{\min}}^{Y_{\max}} y \sum_{i=1}^n \mu_{B_i}(y) dy}{\int_{Y_{\min}}^{Y_{\max}} \sum_{i=1}^n \mu_{B_i}(y) dy}, \quad (4.92)$$

где Y_{\min} , Y_{\max} – границы интервала носителя нечеткого множества выходной переменной y .

11. Нечеткое среднее значение (fuzzy mean). В случае нечеткой производственной модели Такаги–Сугэно 0-го порядка [для нечетких правил типа (4.59)], т. е. если выходные переменные представляют собой одноточечные нечеткие множества, то их значения рассчитываются с использованием модифицированного метода центра тяжести, а именно *метода центра тяжести для одноточечных множеств (centre of gravity for singletons)* (другое название этого метода *fuzzy mean – нечеткое среднее значение*), следующим образом:

$$y' = \frac{\sum_{i=1}^n \alpha_i c_i}{\sum_{i=1}^n \alpha_i}, \quad (4.93)$$

где α_i – агрегированная степень истинности по всем предпосылкам i -го правила; c_i – значение выходной переменной i -го правила ($y_i = c_i$); n – число правил в базе.

Для аффинной модели Такаги–Сугэно, основанной на правилах типа (4.62), выражение (4.93) для вычисления дефазифицированного значения выходной переменной по методу *нечеткого среднего значения* примет следующий вид:

$$y' = \frac{\sum_{i=1}^n \alpha_i \left(\sum_j^m c_{ij} x_j + c_{i0} \right)}{\sum_{i=1}^n \alpha_i}. \quad (4.94)$$

Еще одной разновидностью дефазификации на основе метода центра тяжести является следующее выражение, используемое для вычисления выходной переменной на основе нечеткой производственной модели Цукамото, в заключениях правил которой используются монотонные (возрастающие или убывающие) функции f_i [см. правила типа (4.60)]:

$$y' = \frac{\sum_{i=1}^n \alpha_i f_i^{-1}(\alpha_i)}{\sum_{i=1}^n \alpha_i}, \quad (4.95)$$

где $y'_i = f_i^{-1}(\alpha_i)$ – значение аргумента функции f_i , при котором $\alpha_i = f_i(y'_i)$.

Следует отметить, что метод центра тяжести или его модификации применимы для дефазификации выходных переменных в большинстве нечетких производственных моделей.

4.9. Параметрическая оптимизация конечной базы нечетких правил

Задача параметрической оптимизации конечной базы нечетких правил возникает в случае, если необходимо скорректировать параметры оставленных правил в базе на основе имеющихся экспериментальных данных (обучающей выборки).

Следует отметить, что обычно эта задача решается в рамках представления нечеткой производственной модели в виде так называемой *нечеткой нейронной производственной сети (fuzzy-neural network/system)* при ее обучении. Не останавливаясь пока на вопросах обучения нечетких нейронных производственных сетей (они подробно будут рассмотрены в ч. III), опишем типовой подход к параметрической оптимизации базы правил нечеткой производственной модели.

Допустим, что сформирована окончательная безызбыточная и непротиворечивая база нечетких правил типа (4.64) с MISO-структурой (см. п. 4.2.5).

При этом задана обучающая выборка, состоящая из множества примеров следующего вида:

$$(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)}), k = 1, \dots, K, \quad (4.96)$$

где $x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}$ – значения входных переменных x_1, x_2, \dots, x_m ; $y^{(k)}$ – эталонное значение выходной переменной y в k -м примере; K – общее число примеров в обучающей выборке.

Этап 1. Для каждого примера из обучающей выборки по значениям входных переменных $x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}$ на основе построенной нечеткой продукционной модели (с использованием рассмотренных в п. 4.3–4.7 процедур) определяется значение выходной переменной $y'^{(k)}$.

Этап 2. Вычисляется функция ошибки для всех примеров обучающей выборки:

$$E^{(k)} = \frac{1}{2}(y'^{(k)} - y^{(k)})^2, \quad k = 1, \dots, K. \quad (4.97)$$

Этап 3. Корректировка параметров функций принадлежности нечетких продукционных правил. Введение функции ошибки вида (4.97) позволяет, по аналогии с обычными нейронными сетями, использовать градиентный метод подстройки параметров заданных нечетких продукционных правил.

Так, значения $y'^{(k)}$ можно корректировать, исходя из соотношения

$$y'^{(k)} := y'^{(k)} - \eta \frac{\partial E^{(k)}}{\partial y'^{(k)}}, \quad k = 1, \dots, K, \quad (4.98)$$

где $\eta \in [0, 1]$ – коэффициент, характеризующий скорость коррекции (обучения).

Для изменения $y'^{(k)}$ используются изменения параметров функций принадлежности этих правил с целью минимизации функции ошибки по всем примерам обучающей выборки.

Для рассматриваемого на рис. 4.3 случая, когда все функции принадлежности нечетких множеств являются треугольными, настройке подвергается один их параметр – значения мод.

Процедура параметрической оптимизации (этапы 1–3) итерационно повторяется и считается успешно завершенной в случае, если значение функции ошибки по каждому примеру обучающей выборки не превышает некоторого установленного порога:

$$E^{(k)} < \varepsilon, \quad k = 1, \dots, K. \quad (4.99)$$

Либо для завершения этого процесса можно использовать оценку средней суммарной погрешности нечеткой продукционной модели с учетом всех примеров обучающей выборки:

$$E = \frac{1}{K} \sum_{k=1}^K (y'^{(k)} - y^{(k)})^2 < \varepsilon. \quad (4.100)$$

Параметрическая оптимизация баз нечетких правил различного типа будет рассмотрена ниже при обучении соответствующих им типов нечетких нейронных сетей.

Однако параметрическая оптимизация, основанная на градиентном подходе, зачастую приводит к большим вычислительным затратам, поэтому целесообразным является здесь искать решение на основе генетического подхода.

Глава 5. АЛГОРИТМЫ НЕЧЕТКОГО ВЫВОДА

Компоненты нечетких производственных моделей могут быть реализованы по-разному. Причем выбор различных реализаций одного или нескольких компонентов модели зачастую обосновывает и выбор всех остальных. Совокупность отдельных реализаций описанных выше компонентов нечеткой производственной модели определяет *алгоритм нечеткого вывода*. Рассмотрим алгоритмы нечеткого вывода, получившие в настоящее время наибольшее распространение: Мамдани (Mamdani), Ларсена (Larsen), Цукамото (Tsukamoto), Такаги–Сугэно (Takagi–Sugeno).

5.1. Алгоритм нечеткого вывода Мамдани

При выполнении нечеткого вывода на основе алгоритма Мамдани:

- База правил формируется на основе правил типа (4.64). Предположим, база правил с MISO-структурой (с двумя входами и одним выходом) состоит из двух правил:

П₁: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} , ТО y есть B_1 ,
П₂: ЕСЛИ x_1 есть A_{21} И x_2 есть A_{22} , ТО y есть B_2 .

- Декартово произведение нечетких множеств задано выражением

$$\mu_{A'_{11} \times A'_{12}}(x_1, x_2) = \min\{\mu_{A'_{11}}(x_1), \mu_{A'_{12}}(x_2)\}. \quad (5.1)$$

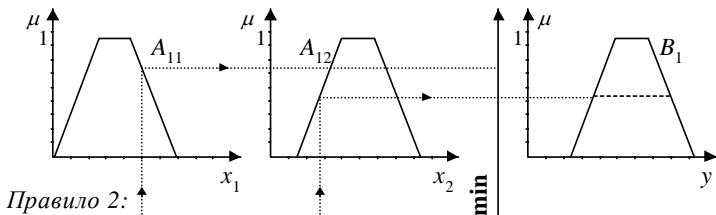
- Нечеткая импликация – операция min-конъюнкция (4.7).
- Т-норма – min-конъюнкция.
- Аккумулирование активизированных заключений правил – максимизация.

Нечеткую модель и механизм нечеткого вывода для приведенного примера можно представить следующим образом:

$$\begin{aligned} \mu_{B'_i}(y) &= \max_{i=1, 2} \sup_{x \in X} \{\min[\mu_{A'_i}(x), \mu_{A_i \rightarrow B_i}(x, y)]\} = \\ &= \max_{i=1, 2} \sup_{x_1, x_2 \in X} \{\min[\mu_{A'_{i1}}(x_1), \mu_{A'_{i2}}(x_2), \mu_{A_{i1}}(x_1), \mu_{A_{i2}}(x_2), \mu_{B_i}(y)]\}. \end{aligned}$$

На рис. 5.1 иллюстрируется выполнение алгоритма нечеткого вывода Мамдани для данного примера.

Правило 1:



Правило 2:

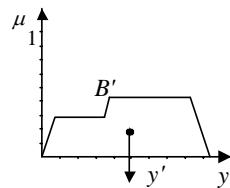
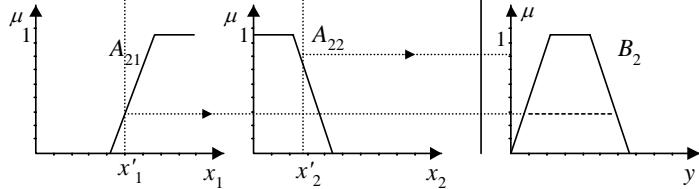


Рис. 5.1. Иллюстрация алгоритма нечеткого вывода Мамдани

Этап 1. Определение степени срабатывания (истинности) каждой предпосылки каждого правила для заданных значений входных переменных $\mu_{A_{ij}}(x'_j)$, ($i, j = 1, 2$). В случае, если входные переменные (x_i) являются четкими или одноточечными нечеткими множествами, данный этап называют этапом введения нечеткости (фаззификацией).

Этап 2. Агрегирование степеней истинности предпосылок по каждому из правил α_i . Для данного примера:

$$\alpha_1 = \min\{\mu_{A_{11}}(x'_1), \mu_{A_{12}}(x'_2)\},$$

$$\alpha_2 = \min\{\mu_{A_{21}}(x'_1), \mu_{A_{22}}(x'_2)\}.$$

В качестве операции агрегирования могут использоваться и другие нечеткие логические операции (4.70)–(4.73).

Этап 3. Активизация (определение степеней истинности) заключений по каждому из правил на основе операции min-активизации:

$$\mu_{B'_1}(y) = \min\{\alpha_1, \mu_{B_1}(y)\},$$

$$\mu_{B'_2}(y) = \min\{\alpha_2, \mu_{B_2}(y)\}.$$

Этап 4. Аккумулирование полученных на предыдущем этапе заключений по всем правилам. Объединение найденных усеченных нечетких множеств проводится с использованием операции максимум (макс-дизъюнкции). В итоге формируется нечеткое множество для выходной переменной с функцией принадлежности

$$\mu_{B'}(y) = \max\{\mu_{B'_1}(y), \mu_{B'_2}(y)\}.$$

Этап 5. Этап приведения к четкости выполняется, если необходимо привести полученное нечеткое множество к четкому виду. В алгоритме нечеткого вывода Мамдани, как правило, используется центроидный метод дефазификации, при котором четкое значение выходной переменной y' определяется как «центр тяжести» (center of gravity) для $\mu_{B'}(y)$:

$$y' = \frac{\int_{Y_{\min}}^{Y_{\max}} y \mu_{B'}(y) dy}{\int_{Y_{\min}}^{Y_{\max}} \mu_{B'}(y) dy},$$

где Y_{\min} , Y_{\max} – границы интервала носителя нечеткого множества выходной переменной y .

Для дискретного варианта:

$$y' = \frac{\sum_{r=1}^{Y_{\max}} y_r \mu_{B'}(y_r)}{\sum_{r=1}^{Y_{\max}} \mu_{B'}(y_r)},$$

где Y_{\max} – число элементов y_r в дискретизированной для вычисления «центра тяжести» области Y .

Из рассмотренных ранее методов дефазификации в алгоритме Мамдани также могут быть применены операции (4.81)–(4.90).

5.2. Алгоритм нечеткого вывода Ларсена

В алгоритме нечеткого вывода Ларсена:

- база правил формируется аналогично алгоритму Мамдани на основе правил типа (4.51);
- декартово произведение нечетких множеств задано формулой (5.1);
- нечеткая импликация – нечеткое умножение (4.8);
- Т-норма – min-конъюнкция;
- аккумулирование активизированных заключений правил – max-дизъюнкция.

В этом случае нечеткую модель и механизм нечеткого вывода для приведенного примера можно представить следующим образом:

$$\begin{aligned}\mu_{B'_i}(y) &= \max_{i=1,2} \sup_{x \in X} \{ \mu_{A'_i}(x) \cdot \mu_{A_i \rightarrow B_i}(x, y) \} = \\ &= \max_{i=1,2} \sup_{x_1, x_2 \in X} \{ \mu_{B_i}(y) \cdot \min[\mu_{A'_{i1}}(x_1), \mu_{A'_{i2}}(x_2), \mu_{A_{i1}}(x_1), \mu_{A_{i2}}(x_2)] \}.\end{aligned}$$

На рис. 5.2 иллюстрируется выполнение алгоритма нечеткого вывода Ларсена в соответствии с этим примером.

Этап 1. Фазификация. Подобно способу Мамдани определяются степени истинности каждой предпосылки каждого правила для заданных значений входных переменных $\mu_{A_{ij}}(x'_j)$, ($i, j = 1, 2$).

Этап 2. Агрегирование результатов срабатывания (степеней истинности) предпосылок по каждому из правил и формирование уровней их отсечения α_i подобно алгоритму Мамдани. В качестве этой операции могут использоваться также операции (4.70)–(4.73).

Этап 3. Активизация заключений каждого из нечетких продукционных правил выполняется на операции prod-активизации:

$$\mu_{B'_1}(y) = \alpha_1 \mu_{B_1}(y),$$

$$\mu_{B'_2}(y) = \alpha_2 \mu_{B_2}(y).$$

Этап 4. Аккумулирование полученных на предыдущем этапе заключений и получение результирующего нечеткого множества выполняется с использованием операции max-дизъюнкции:

$$\mu_{B'}(y) = \mu_{B'_1}(y) \vee \mu_{B'_2}(y)$$

$$(в \text{ случае } n \text{ правил: } \mu_{B'}(y) = \bigvee_{i=1}^n \mu_{B'_i}(y)).$$

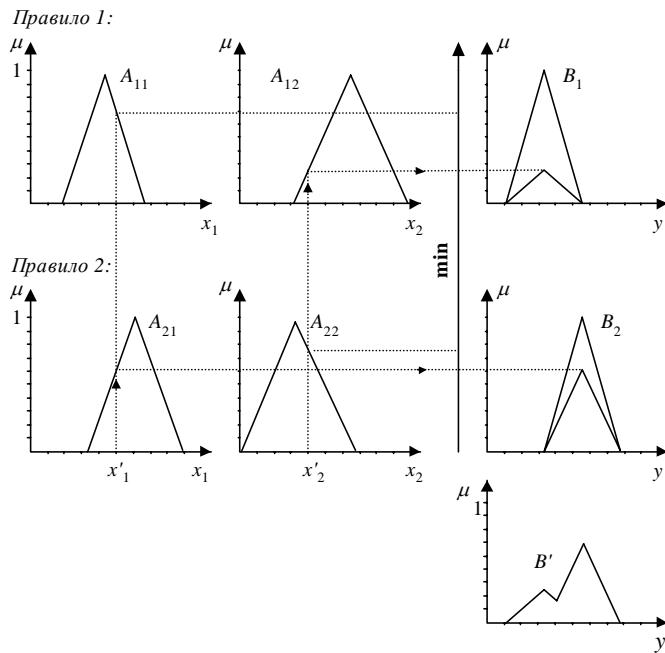


Рис. 5.2. Иллюстрация алгоритма нечеткого вывода Ларсена

Этап 5. Приведение к четкости проводится в случае необходимости на основе одного из рассмотренных выше методов дефазификации (4.81)–(4.90).

5.3. Алгоритм нечеткого вывода Цукамото

При реализации алгоритма нечеткого вывода Цукамото:

- База правил формируется на основе правил типа (4.60). Предположим, база правил с MISO-структурой (с двумя входами и одним выходом) состоит из двух правил:

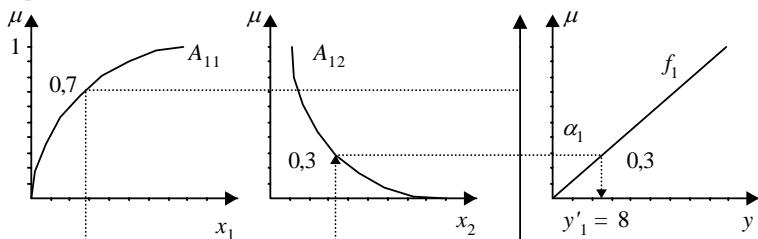
П₁: **ЕСЛИ** x_1 есть A_{11} **И** x_2 есть A_{12} , **ТО** $y = f_1^{-1}(\alpha_1)$,
П₂: **ЕСЛИ** x_1 есть A_{21} **И** x_2 есть A_{22} , **ТО** $y = f_2^{-1}(\alpha_2)$,

где f_1, f_2 – монотонные функции.

- Нечеткая импликация – операция min-конъюнкции.

На рис. 5.3 показан пример выполнения алгоритма Цукамото.

Правило 1:



Правило 2:

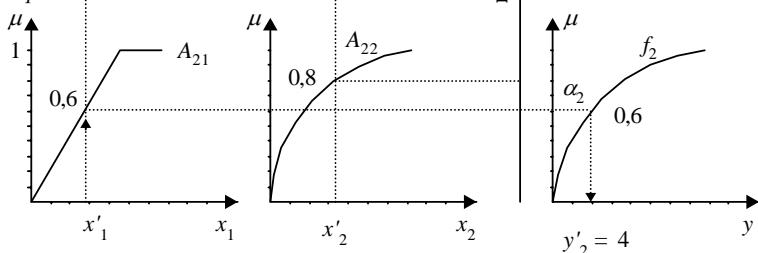


Рис. 5.3. Иллюстрация алгоритма нечеткого вывода Цукамото

Этап 1. Введение нечеткости (как в алгоритме Мамдани).

Этап 2. Агрегирование степеней истинности предпосылок по каждому из правил α_1 и α_2 (как в алгоритме Мамдани).

Этап 3. Активизация заключений по каждому из правил:

$$y'_1 = f_1^{-1}(\alpha_1), \\ y'_2 = f_2^{-1}(\alpha_2).$$

В результате находятся четкие значения выходных переменных в каждом из заключений правил.

Этап 4. Этап аккумулирования активизированных заключений правил в данном алгоритме отсутствует вследствие четких значений выходных переменных.

Этап 5. В качестве метода дефазификации в алгоритме Цукамото используется разновидность метода центра тяжести для одноточечных множеств (4.95), позволяющий осуществить приведение к четкости выходной переменной без предварительного аккумулирования активизированных заключений отдельных правил:

$$y' = \frac{\alpha_1 y'_1 + \alpha_2 y'_2}{\alpha_1 + \alpha_2}.$$

Пример. Пусть заданы $A_{11}(x_1') = 0.7$, $A_{21}(x_1') = 0.6$, $A_{12}(x_2') = 0.3$, $A_{22}(x_2') = 0.8$, соответствующие уровня отсечения:

$$\alpha_1 = \min\{A_{11}(x_1'), A_{12}(x_2')\} = \min\{0.7, 0.3\} = 0.3,$$
$$\alpha_2 = \min\{A_{21}(x_1'), A_{22}(x_2')\} = \min\{0.6, 0.8\} = 0.6.$$

и значения $y_1' = 8$ и $y_2' = 4$, найденные в результате решения уравнений (рис. 5.3):

$$f_1(y_1') = 0.3, f_2(y_2') = 0.6.$$

При этом четкое значение переменной вывода:

$$y' = (8 \cdot 0.3 + 4 \cdot 0.6) / (0.3 + 0.6) = 6.$$

5.4. Упрощенный алгоритм нечеткого вывода

Данный алгоритм также называется алгоритмом нечеткого вывода Сугэно 0-го порядка. В этом алгоритме:

- База правил формируется на основе правил типа (4.59). Предположим, база правил с MISO-структурой (с двумя входами и одним выходом) состоит из двух правил:

П₁: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} , ТО $y = c_1$,

П₂: ЕСЛИ x_1 есть A_{21} И x_2 есть A_{22} , ТО $y = c_2$.

- В качестве нечеткой импликации используется операция min-конъюнкции.

На рис. 5.4 показан пример выполнения данного алгоритма.

Этап 1. Введение нечеткости (как в алгоритме Мамдани).

Этап 2. Агрегирование степеней истинности предпосылок по каждому из правил (как в алгоритме Мамдани).

Для данного примера

$$\alpha_1 = \min\{\mu_{A_{11}}(x_1'), \mu_{A_{12}}(x_2')\},$$

$$\alpha_2 = \min\{\mu_{A_{21}}(x_1'), \mu_{A_{22}}(x_2')\}.$$

В качестве операции агрегирования могут использоваться и другие нечеткие логические операции, см. (4.70)–(4.73).

Этап 3. Активизация заключений по каждому из правил:

$$y_1' = c_1,$$

$$y_2' = c_2.$$

В результате чего находятся четкие значения выходных переменных в каждом из заключений правил.

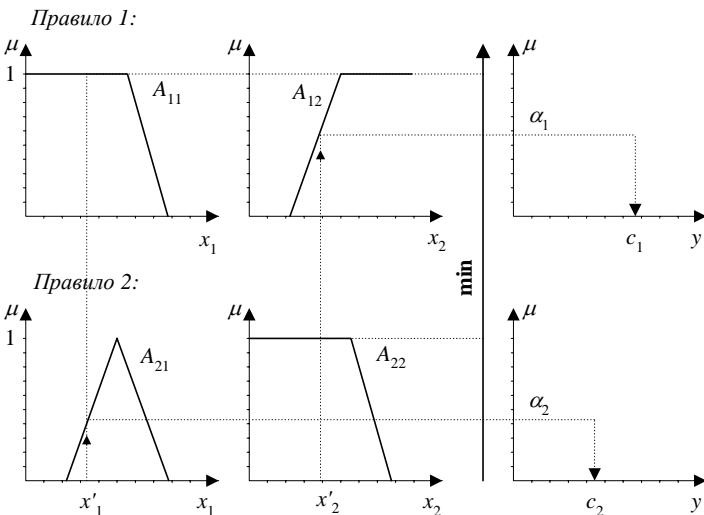


Рис. 5.4. Иллюстрация упрощенного алгоритма нечеткого вывода

Этап 4. Этап аккумулирования заключений нечетких производственных правил в данном алгоритме отсутствует вследствие четких значений выходных переменных.

Этап 5. В качестве метода дефазификации в данном алгоритме используется разновидность метода центра тяжести для одноточечных множеств (4.93), позволяющий осуществить приведение к четкости выходной переменной без предварительного аккумулирования активизированных заключений отдельных правил:

$$y' = \frac{\alpha_1 y'_1 + \alpha_2 y'_2}{\alpha_1 + \alpha_2}.$$

5.5. Алгоритм нечеткого вывода Такаги–Сугэно

В алгоритме нечеткого вывода Такаги–Сугэно:

- База нечетких правил формируется на основе правил типа (4.62). Предположим, база правил с MISO-структурой (с двумя входами и одним выходом) состоит из двух правил:

П1: **ЕСЛИ** \$x_1\$ есть \$A_{11}\$ **И** \$x_2\$ есть \$A_{12}\$, **ТО** \$y = c_{11} x_1 + c_{12} x_2 + c_{10}\$,

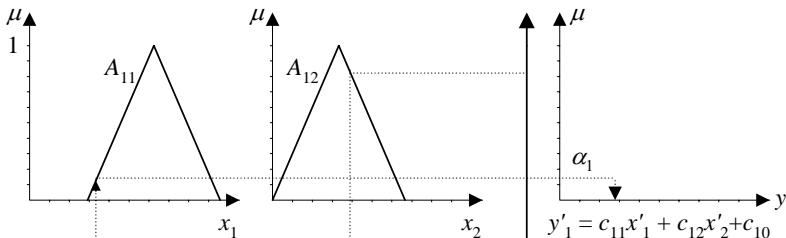
П2: **ЕСЛИ** \$x_1\$ есть \$A_{21}\$ **И** \$x_2\$ есть \$A_{22}\$, **ТО** \$y = c_{21} x_1 + c_{22} x_2 + c_{20}\$,

где \$c_{ij}\$ (\$i, j = 1, 2\$) – коэффициенты компонентов вектора; \$c_{i0}\$ – смещение.

- Нечеткая импликация – операция \$\min\$-конъюнкции.

На рис. 5.5 показан пример выполнения алгоритма Такаги–Сугэно.

Правило 1:



Правило 2:

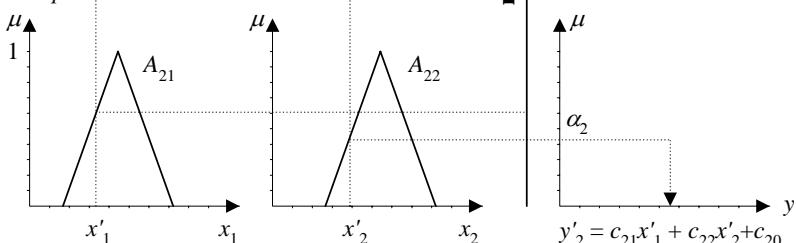


Рис. 5.5. Иллюстрация алгоритма нечеткого вывода Такаги–Сугэно

Этап 1. Введение нечеткости (как в алгоритме Мамдани).

Этап 2. Агрегирование степеней истинности предпосылок по каждому из правил (как в алгоритме Мамдани). Для данного примера

$$\alpha_1 = \min\{\mu_{A_{11}}(x'_1), \mu_{A_{12}}(x'_2)\},$$

$$\alpha_2 = \min\{\mu_{A_{21}}(x'_1), \mu_{A_{22}}(x'_2)\}.$$

В качестве операции агрегирования могут использоваться и другие нечеткие логические операции, см. (4.70)–(4.73).

Этап 3. Активизация заключений по каждому из правил:

$$y'_1 = c_{11} x'_1 + c_{12} x'_2 + c_{10},$$

$$Y'_2 = c_{21} x'_1 + c_{22} x'_2 + c_{20}.$$

В результате находятся четкие значения выходных переменных в каждом из заключений правил.

Этап 4. Этап аккумулирования заключений нечетких продукционных правил в данном алгоритме отсутствует вследствие четких значений выходных переменных.

Этап 5. В качестве метода дефазификации в данном алгоритме используется разновидность метода центра тяжести для одноточечных множеств (4.94), позволяющая осуществить приведение к четкости выходной переменной без предварительного аккумулирования активизированных заключений отдельных правил:

$$y' = \frac{\alpha_1 y'_1 + \alpha_2 y'_2}{\alpha_1 + \alpha_2}.$$

5.6. Алгоритм нечеткого вывода на основе нечеткой производственной модели с адаптацией операций над нечеткими множествами

В данном алгоритме нечеткого вывода база нечетких правил формируется на основе правил типа (4.58), в заключениях которых операции над нечеткими высказываниями выбираются в зависимости, например, от степени согласованности (значимости) нечетких высказываний в предпосылках [15].

Приведем описание этого алгоритма на примере построения модели оценки достижимости общей цели системы с учетом ее согласования со многими неравнозначными частными целями этой системы.

Пример. Пусть в результате анализа сложной системы определена совокупность нечетких целей. Тогда общая цель $G_{об}$ выражается совокупностью частных целей G_i , $i = 1 \dots q$. Каждая частная цель связана с частным критерием, описываемым нечетким множеством, определенным на X_i , причем для $\forall x_i \in X_i$ величина $\mu_{G_i}(x)$ характеризует степень ее достижимости. Общая целевая функция задается нечетким множеством, определенным на базовом множестве $X_1 \dots X_q$.

Рассмотрим основные этапы данного алгоритма нечеткого вывода.

Этап 1. Для каждой из частных целей, а также для общей цели формируется лингвистическая переменная (с одноименным названием), множество лингвистических значений (терм-множество состояний) которой характеризует степени ее достижимости.

Рассмотрим в качестве примера одни и те же значения для всех терм-множеств $\{L - \text{низкая}, M - \text{средняя}, H - \text{высокая}\}$.

Этап 2. Каждая частная цель определяется совокупностью нечетких производственных правил относительно входных переменных. Не будем здесь останавливаться на задаче нечеткого вывода по каждой частной цели, так как эту задачу можно решить известными способами, например

с использованием вышерассмотренного алгоритма нечеткого вывода Мамдани.

Этап 3. Определяются число, вид и лингвистические оценки степени парных согласований общей цели с каждой частной целью. Например, множество лингвистических значений, характеризующих степени согласования общей $G_{\text{об}}$ и i -й частной G_i ($i = 1, \dots, q$) целей можно сформировать следующим образом: {полная совместимость; большая совместимость; средняя совместимость; малая совместимость; несовместимость} (рис. 5.6).

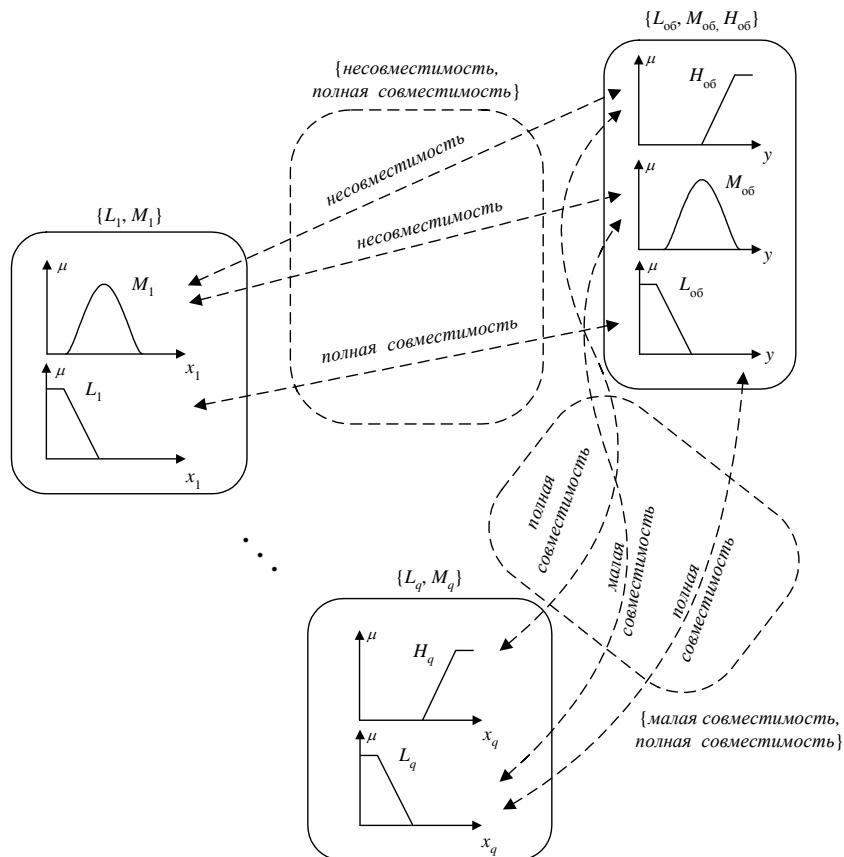


Рис. 5.6. Пример согласованности значений терм-множеств целей

Этап 4. Идентифицируются возможные операции свертки общей целевой функции (для всех значений из терм-множества) с частными в зависимости от степени их согласованности, например:

- $\max(G_{\text{об}}, G_j)$ – полная совместимость целей $G_{\text{об}}$ и G_j ;
- $(G_{\text{об}}, + G_j)/2$ – большая совместимость;
- $\text{med}(G_{\text{об}}, G_j)$ – средняя совместимость;
- $(G_{\text{об}} \times G_j)^{1/2}$ – малая совместимость;
- $\min(G_{\text{об}}, G_j)$ – несовместимость [24].

Для свертки, учитывающей степень согласованности общей и j -й частной целей, целесообразно также использовать параметризированное семейство операций в виде

$$I(G_{\text{об}}, G_j)^\gamma \cdot U(G_{\text{об}}, G_j)^{(1-\gamma)}, \gamma \in [0, 1], \quad (5.2)$$

где I , U – некоторые операции пересечения и объединения соответственно; γ – показатель, характеризующий степень согласованности целей $G_{\text{об}}$ и G_j [25].

Этап 5. Формируется совокупность нечетких правил, каждое из которых определяет стратегию достижения отдельного значения из терм-множества, характеризующего общую целевую функцию $H_{\text{об}}$, $M_{\text{об}}$, $L_{\text{об}}$ (в общем случае число этих правил может быть больше).

Формирование каждого правила включает в себя процедуру идентификации операций свертки для заданного значения общей целевой функции с частными, а также операцию комбинирования результатов этих сверток, характеризующую нижний уровень их согласования:

- Π_1 : **ЕСЛИ** x_1 есть H_1 И ... И x_q есть H_q ,
ТО y есть $(H_1 \theta_1^1 H_{\text{об}}) \varTheta_l^1, \dots, \varTheta_l^1 (H_q \theta_q^1 H_{\text{об}})$,
- Π_2 : **ЕСЛИ** x_1 есть H_1 И ... И x_q есть M_q ,
ТО y есть $(H_1 \theta_1^2 H_{\text{об}}) \varTheta_l^2, \dots, \varTheta_l^2 (M_q \theta_q^2 H_{\text{об}})$,
- ...
- Π_i : **ЕСЛИ** x_1 есть M_1 И ... И x_q есть M_q ,
ТО y есть $(M_1 \theta_1^i M_{\text{об}}) \varTheta_l^i, \dots, \varTheta_l^i (L_q \theta_q^i M_{\text{об}})$,
- ...
- Π_n : **ЕСЛИ** x_1 есть L_1 И ... И x_q есть L_q ,
ТО y есть $(L_1 \theta_1^n L_{\text{об}}) \varTheta_l^n, \dots, \varTheta_l^n (L_q \theta_q^n L_{\text{об}})$,

где x_1, \dots, x_q – входные переменные; y – выходная переменная; $\{L_j, M_j, H_j\}$ – функции принадлежности, характеризующие степень достижимости j -й

частной цели G_j , $j = 1, \dots, q$; $\{L_{\text{об}}, M_{\text{об}}, H_{\text{об}}\}$ – функции принадлежности, характеризующие степень достижимости общей цели $G_{\text{об}}$; $\theta_1^1, \dots, \theta_q^1, \dots, \theta_1^n, \dots, \theta_q^n$ – операции свертки общей целевой функции с частными, выбираемые в зависимости от уровня их совместимости; ϑ_l^i – операция комбинирования результатов сверток общей цели с частными в i -м правиле ($I = 1, \dots, n$), выбираемая из соответствующей совокупности операций парных сверток общей цели с частными $\theta_1^i, \dots, \theta_q^i$ и характеризующая нижний уровень их согласования.

Этап 6. Для каждого из правил Π_i осуществляется парная свертка значений общей и частных целевых функций на основе идентифицированных операций $\theta_1^i, \dots, \theta_q^i$, выбранных в соответствии с уровнем их совместимости, и формируются «модифицированные» функции принадлежности:

$$\begin{aligned}\alpha_{i1} &= (M_1 \theta_1^i M_{\text{об}}), \\ &\dots \\ \alpha_{iq} &= (L_q \theta_q^i M_{\text{об}}).\end{aligned}$$

Этап 7. Находятся степени достижимости всех частных целей, т. е. уровни отсечения для всех предпосылок каждого правила: $L_1(x_1)$, $M_1(x_1)$, $H_1(x_1), \dots, L_q(x_q)$, $M_q(x_q)$, $H_q(x_q)$.

Этап 8. Для каждого правила Π_i формируются «усеченные» функции принадлежности предпосылок каждой парной свертки каждого из правил:

$$\begin{aligned}\alpha_{i1}(x_1) &= (M_1(x_1) \theta_1^i M_{\text{об}}), \\ &\dots \\ \alpha_{iq}(x_q) &= (L_q(x_q) \theta_q^i M_{\text{об}}).\end{aligned}$$

Этап 9. Формируются частные заключения по каждому из правил на основе выбранной операции комбинирования результатов сверток общей целевой функции с частными:

$$\begin{aligned}\beta_1 &= \alpha_{i1}(x_1) \vartheta_1^1 \dots \vartheta_l^1 \alpha_{1q}(x_q), \\ &\dots \\ \beta_i &= \alpha_{i1}(x_1) \vartheta_1^i \dots \vartheta_l^i \alpha_{iq}(x_q), \\ &\dots \\ \beta_n &= \alpha_{n1}(x_1) \vartheta_1^n \dots \vartheta_l^n \alpha_{nq}(x_1).\end{aligned}$$

Этап 10. Осуществляется объединение (композиция) частных заключений правил. При подобном объединении обычно используется операция максимума или суммирования. Для задания наиболее точной оценочной модели (с учетом рассмотренного механизма согласования целей) в качестве такой операции объединения целесообразно использовать операцию из множества Θ_i^i ($i = 1, \dots, n$), характеризующую наибольшую степень согласования целей по всем правилам.

В результате получается комбинированное нечеткое подмножество, описываемое обобщенной функцией принадлежности, и соответствующее степени достижимости общей цели системы.

Этап 11. При необходимости находится четкое значение выходной переменной с использованием одного из методов (например, центроидного) приведения к четкости.

На рис. 5.7 приведена иллюстрация рассмотренного алгоритма нечеткого вывода для построения оценочной модели достижимости общей цели в соответствии с данным примером.

5.7. Аппроксимационные свойства нечетких производственных моделей

Возможность использования нечетких моделей для решения задач распознавания образов, аппроксимации, оптимизации, векторного распознавания, классификации базируется на следующих результатах:

1. В 1992 г. Коско (Kosko) доказал теорему (Fuzzy Approximation Theorem), согласно которой любая функциональная зависимость, заданная на компактном множестве, может быть аппроксимирована *аддитивной нечеткой моделью* (*additive fuzzy system/model*), структура и параметры которой подобны модели Мамдани, но отличающейся от нее тем, что на этапе аккумулирования активизированных заключений правил используется операция их взвешенного суммирования с последующей дефазификацией результата по методу центра тяжести [26]. Конструктивным результатом доказательства этой теоремы является оценка необходимого числа правил модели для заданной точности аппроксимации, которое определяется с помощью минимального расстояния между центроидами двух смежных нечетких множеств, представляющих консеквенты правил, обозначаемых как y_i и y_{i+1} :

$$|y_i - y_{i+1}| < \frac{\varepsilon}{2g-1}, \quad (5.3)$$

где ε – точность аппроксимации; g – максимальное число перекрытий (overlapping) антецедентов по X (для одномерного входа $m = 2$).

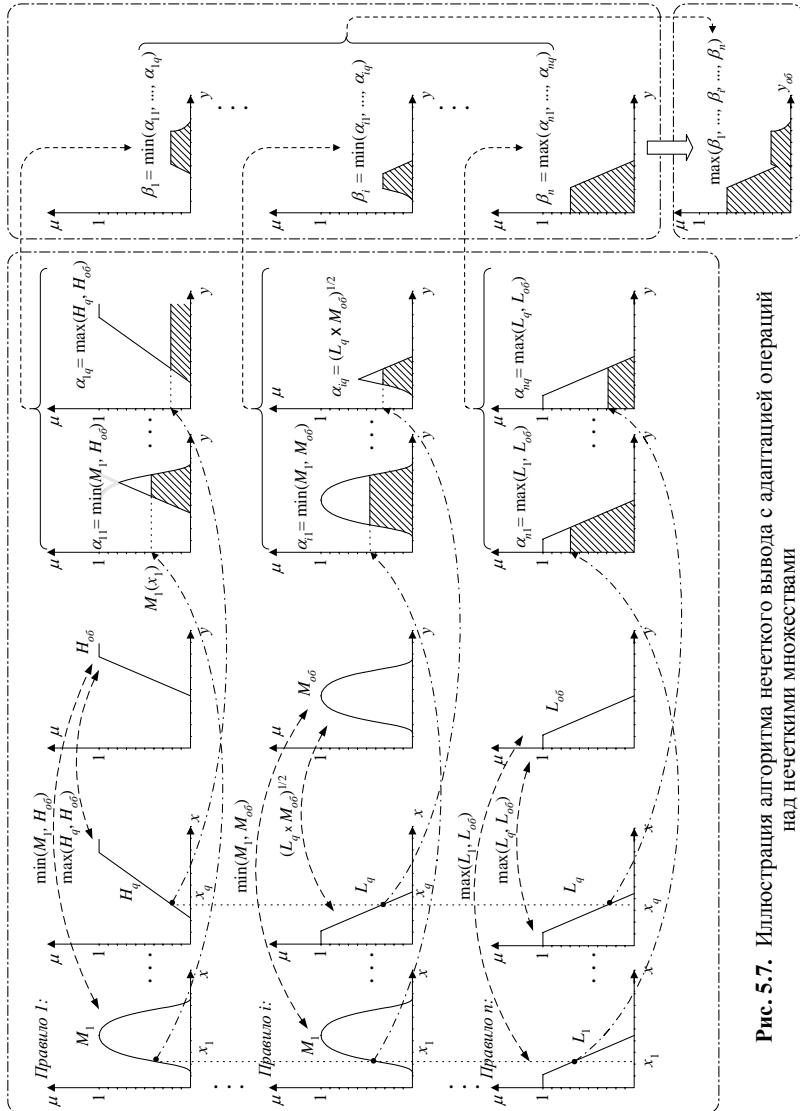


Рис. 5.7. Иллюстрация алгоритма нечеткого вывода с адаптацией операций над нечеткими множествами

Для одномерной входной переменной необходимое число правил определяется следующим образом:

$$n \geq \frac{|X|}{\varepsilon}. \quad (5.4)$$

Очевидно, что для произвольно хорошей аппроксимации число правил неограниченно, однако для заданного ε число правил может быть оценено с использованием выражения (5.3).

2. В 1992 г. Ванг (Wang) в [27] показал, что нечеткая производственная модель является универсальным аппроксиматором, т. е. может аппроксимировать любую непрерывную функцию на компакте X , с произвольной точностью, если использует набор правил n ($n \rightarrow \infty$) типа

П_i: ЕСЛИ x_1 есть A_{i1} И ... И x_j есть A_{ij} И ... И x_m есть A_{im} ,
ТО y есть B_i , $i = 1, \dots, n$

при выполнении следующих условий:

- гауссовых функциях принадлежности:

$$\mu_{A_{ij}}(x_j) = \exp\left[-\frac{1}{2}\left(\frac{x_j - a_{ij}}{b_{ij}}\right)^2\right], \quad \mu_{B_i}(y) = \exp\left[-\frac{1}{2}\left(\frac{y - c_i}{d_i}\right)^2\right],$$

где a_{ij}, b_{ij}, c_i, d_i – параметры функций принадлежности;

- агрегирования степени истинности предпосылок правил в виде произведения:

$$\alpha_i = \mu_{A_{i1}}(x'_1) \cdots \mu_{A_{ij}}(x'_{j'}) \cdots \mu_{A_{im}}(x'_{m'});$$

- активизации заключений правил (импликации) в виде произведения (импликация Ларсена):

$$\mu_{B'_i}(y) = \alpha_i \mu_{B_i}(y);$$

- методе «центра тяжести» при приведении к четкости:

$$y' = \frac{\sum_{r=1}^{Y_{\max}} y_r \mu_{B'_i}(y_r)}{\sum_{r=1}^{Y_{\max}} \mu_{B'_i}(y_r)},$$

где Y_{\max} – число элементов y_r в дискретизированной для вычисления «центра тяжести» области Y .

Ванг (Wang) доказал теорему, что для каждой вещественной непрерывной функции $f(x)$, заданной на компакте X , и для произвольного $\varepsilon > 0$ существует нечеткая продукционная модель, формирующая выходную функцию $y(x)$, такую, что

$$\sup_{x \in X} \|f(x) - y(x)\| \leq \varepsilon,$$

где $\| \dots \|$ – символ принятого расстояния между функциями.

Так же как и в случае теоремы Кооско, результаты Ванга подтверждают трудности, возникающие в случае точной аппроксимации: число правил модели неограниченно.

3. В дальнейшем многие исследователи показали, что свойство универсальной аппроксимации сохраняется для различных типов нечетких продукционных моделей. Например, в 1995 г. Кастро (Castro) обосновал, что нечеткая продукционная модель с правилами типа (4.64) и алгоритмом нечеткого вывода Мамдани также является универсальным аппроксиматором при выполнении следующих условий:

- симметричных треугольных функциях принадлежности:

$$\mu_{A_{ij}}(x_j) = \begin{cases} 1 - |a_{ij} - x_j| / b_{ij}, & \text{если } |a_{ij} - x_j| \leq b_{ij}, \\ 0, & \text{если } |a_{ij} - x_j| > b_{ij}, \end{cases}$$

$$\mu_{B_i}(y) = \begin{cases} 1 - |c_i - x_j| / d_i, & \text{если } |c_i - x_j| \leq d_i, \\ 0, & \text{если } |c_i - x_j| > d_i; \end{cases}$$

- агрегировании степени истинности предпосылок правил в виде минимума:

$$\alpha_i = \min\{\mu_{A_{i1}}(x'_1), \dots, \mu_{A_{ij}}(x'_{j}), \dots, \mu_{A_{im}}(x'_{m})\};$$

- активизации заключений правил (импликации) в виде минимума:

$$\mu_{B'_i}(y) = \min\{\alpha_i, \mu_{B_i}(y)\};$$

- методе «центра тяжести» при приведении к четкости:

$$y' = \frac{\sum_{r=1}^{Y_{\max}} y_r \mu_{B'}(y_r)}{\sum_{r=1}^{Y_{\max}} \mu_{B'}(y_r)}.$$

Помимо этого, в своей наиболее общей работе – [28], посвященной проблеме аппроксимации, Кастро привел ряд доказательств свойств универсальной аппроксимации для широкого спектра нечетких моделей, различающихся классами нечеткой логики, типами функций принадлежности, методами дефазификации.

Однако все вышеуказанные результаты не дают ответа на вопросы, какую конкретно модель необходимо выбрать и сколько должно быть правил для аппроксимации заданной функции.

Можно отметить, что в нечетких производственных моделях имеет место экспоненциальное возрастание числа нечетких правил при стремлении к нулю ошибки аппроксимации.

С практической точки зрения достаточно иметь приемлемо хорошую точность аппроксимации. В этом случае задача состоит в поиске возможного компромисса между указанной точностью и числом правил модели, которые позволяют осуществить аппроксимацию с заданной точностью за приемлемое время.

4. Следует отметить, что если первые работы по универсальной аппроксимации были посвящены нечетким моделям Мамдани, то начиная с середины 90-х гг. аналогичные результаты были получены и для нечетких моделей Такаги–Сугэно. Так, например, в работе [29] было доказано, что нечеткая модель Такаги–Сугэно является универсальным аппроксиматором при следующих условиях:

- с функциями принадлежности антецедентов, полностью перекрывающими (full-overlapped) пространство входных переменных;
- с линейными функциями консеквентов;
- со взвешенным суммированием при дефазификации.

В этой работе также приведены конструктивные результаты проектирования нечеткой модели Такаги–Сугэно, заключающиеся в определении вида функций принадлежности антецедентов и необходимого числа правил при выполнении следующих достаточных условий (sufficient conditions) аппроксимации: заданной непрерывной функции и точности аппроксимации.

Для этого используется следующая двухэтапная процедура аппроксимации.

Этап 1. Определяется полином $P_q(x)$ конечной степени q со многими переменными $x = (x_1, x_2, \dots, x_m)$, который, в соответствии с теоремой Вейерштрасса, на компактном домене равномерно аппроксимирует любую непрерывную функцию с любой заданной точностью ε_1 .

Этап 2. Проектируется нечеткая модель Такаги–Сугэно в соответствии с вышеуказанными условиями для аппроксимации этого полинома $P_q(\mathbf{x})$ с любой точностью $(\varepsilon - \varepsilon_1)$.

При этом для нечеткой модели Сугэно 0-го порядка число правил определяется в соответствии со следующим выражением:

$$n \geq \frac{1}{\varepsilon} \sum_{d_1=0}^{D_1} \sum_{d_2=0}^{D_2} \cdots \sum_{d_m=0}^{D_m} \left(\beta_{d_1 d_2 \dots d_m} x_1^{d_1} x_2^{d_2} \dots x_m^{d_m} \right), \quad (5.5)$$

а для модели Такаги–Сугэно с двумя входами и одним выходом

$$n \geq \frac{\left| \beta_{1,0} \right| + \left| \beta_{0,1} \right| + \sum_{d_1=0}^{m_1} \sum_{d_2=0}^{m_2} \left| \beta_{d_1 d_2} \right| (2^{d_1+d_2} - 1)}{\varepsilon - \varepsilon_1}. \quad (5.6)$$

В работе [30] авторы, также используя рассмотренную выше двухэтапную процедуру аппроксимации, установили другие условия по определению числа правил в модели Такаги–Сугэно для заданной точности аппроксимации. Так, модели Сугэно 0-го порядка достаточное число правил по каждому измерению (для каждой входной переменной x_j) $j \in (1, \dots, m)$ определяется в соответствии с выражением

$$n > \frac{1}{\varepsilon - \varepsilon_1} \sum_{j=1}^m \left\| \frac{\partial P_q}{\partial x_j} \right\|_\infty - 1, \quad (5.7)$$

а для нечеткой модели Такаги–Сугэно – следующим образом:

$$n > \sqrt{\frac{1}{2(\varepsilon - \varepsilon_1)} \sum_{j=1}^m \sum_{k=1}^m \left\| \frac{\partial^2 P_q}{\partial x_j \partial x_k} \right\|_\infty} - 1, \quad (5.8)$$

где P_q – полином конечной степени q со многими переменными:

$$P_q(\mathbf{x}) = \sum_{d_1=0}^{D_1} \sum_{d_2=0}^{D_2} \cdots \sum_{d_m=0}^{D_m} \left(\beta_{d_1 d_2 \dots d_m} x_1^{d_1} x_2^{d_2} \dots x_m^{d_m} \right), \quad (5.9)$$

где $\left\| \frac{\partial P_q}{\partial x_j} \right\|_\infty$ – экстремум частной производной полинома.

Сравнивая (5.5) и (5.7), можно заключить, что (5.7) предпочтительнее, когда размерность пространства входных переменных относительно невелика, а (5.5) предпочтительнее в случае пространства высокой раз-

мерности, так как (5.7) использует экстремум частной производной полинома. Подобные рассуждения имеют место и при сравнении (5.6) и (5.8).

Зачастую выбор той или иной модели обосновывается эвристическими предпочтениями либо простотой реализации при обеспечении требуемого критерия эффективности.

Пример. Сравним эффективность использования алгоритмов нечеткого вывода, базирующихся на нечетких производственных моделях Мамдани и Сугэно 0-го порядка при решении задачи аппроксимации непрерывной функции одной переменной по критериям обеспечения точности аппроксимации и вычислительных затрат на реализацию [31].

Сравнение описанных алгоритмов выполнялось при таких условиях:

- аппроксимация функции проводилась на отрезке $[-1, 1]$;
- аппроксимируемая функция задавалась набором значений $(x^{(k)}, y^{(k)})$, $k = 1, \dots, K$, при этом точки $x^{(k)}$ располагались эквидистантно;
- функции принадлежности имели вид функций Гаусса, т. е.

$$\mu_{A_i}(x) = \exp\left[-\frac{1}{2}\left(\frac{x - x^{(k)}}{a}\right)^2\right], \quad \mu_{B_i}(y) = \exp\left[-\frac{1}{2}\left(\frac{y - y^{(k)}}{b}\right)^2\right];$$

- количество правил n задано заранее;
- варьировались значения параметров a и b для получения наибольшей точности аппроксимации при заданном n .

В табл. 5.1 и 5.2 приведены результаты сравнительной оценки аппроксимации функций соответственно $f(x) = x^2$ и $f(x) = x^3$ при $n = 9$, $a = 0.1$, $b = 0.3$.

Таблица 5.1. Результаты аппроксимации функции $f(x) = x^2$

Номер примера (κ)	1	2	3	4	5	6	7	8	9
$x^{(k)}$	-1	-0.75	-0.5	-0.25	0	0.25	0.5	0.75	1
$y^{(k)}$	1.000	0.563	0.250	0.063	0	0.063	0.25	0.563	1.000
<i>Оценка по алгоритму Мамдани</i>	0.973	0.570	0.257	0.070	0	0.070	0.257	0.570	0.973
<i>Оценка по алгоритму Такаги–Сугэно</i>									
	0.982	0.568	0.255	0.068	0	0.068	0.255	0.568	0.982

Таблица 5.2. Результаты аппроксимации функции $f(x) = x^3$

Номер примера (k)	1	2	3	4	5	6	7	8	9
$x^{(k)}$	-1	-0.75	-0.5	-0.25	0	0.25	0.5	0.75	1
$y^{(k)}$	-1.000	-0.422	-0.125	-0.016	0	0.016	0.125	0.422	1.000
Оценка по алгоритму Мамдани	-0.962	-0.441	-0.136	-0.021	0	0.021	0.136	0.441	0.962
Оценка по алгоритму Такаги–Сугэно	-0.976	-0.433	-0.133	-0.019	0	0.019	0.133	0.433	0.976

Эти, а также другие полученные результаты вычислительных экспериментов позволяют сделать следующие выводы:

- при прочих равных условиях и при оптимальных параметрах a и b погрешность аппроксимации с применением алгоритма Такаги–Сугэно несколько меньше, чем с применением алгоритма Мамдани;
- алгоритм Такаги–Сугэно с вычислительной точки зрения реализуется значительно проще, чем алгоритм Мамдани, а время вычислений для него меньше, чем для алгоритма Мамдани в 50–100 раз.

Таким образом, если нет каких-либо особых доводов в пользу алгоритма Мамдани, то лучше использовать не его, а алгоритм Такаги–Сугэно.

5. В последнее время получен ряд интересных результатов, которые позволяют отчасти по-иному взглянуть на свойства нечетких производственных моделей, обеспечивающих их аппроксимационные возможности.

В работе [32] решается вопрос существенного сокращения числа правил за счет «трансформации» нелинейности аппроксимируемой функции в функции принадлежности антецедентов этих правил. Однако в этом случае функции принадлежности антецедентов правил должны быть построены путем прямого преобразования функции, что часто невозможно, если параметры функции неизвестны.

Другой пример – нечеткие модели, базирующиеся на использовании многозначных импликаций, также являются универсальными аппроксиматорами [33, 34].

В [35] предложены иерархические нечеткие модели: аддитивные m -входовые иерархические нечеткие модели, включающие в себя ($m - 1$) входовых нечетких производственных моделей. Было доказано, что такие модели также являются универсальными аппроксиматорами [36].

Полученные же в [37] результаты по аппроксимации на основе нечетких моделей с разреженными базами правил (sparse rule bases), показывают, что аппроксимационные способности нечетких моделей (т. е. свойство универсальной аппроксимации) остаются неизменными, даже если ослабляется условие того, что множество функций принадлежности полностью покрывает пространство входных переменных в базе правил.

Глава 6. НЕЧЕТКИЕ РЕЛЯЦИОННЫЕ МОДЕЛИ И РЕЛЯЦИОННОЕ ПРЕДСТАВЛЕНИЕ НЕЧЕТКИХ ПРОДУКЦИОННЫХ МОДЕЛЕЙ

6.1. Нечеткие реляционные модели

В нечетких реляционных моделях входные нечеткие переменные отображаются на выходные нечеткие переменные с помощью нечетких отношений, определяемых, например, реляционными матрицами. Эти модели позволяют предварительно зафиксировать лингвистические термы входных и выходных нечетких переменных и затем настроить нечеткие отображения, реализуемые моделями, посредством изменения элементов нечетких отношений [23].

Нечеткие реляционные модели получили распространение в задачах кластеризации и классификации [38], идентификации и управления нелинейными динамическими объектами и системами [39], а также в ряде других задач, требующих решения систем нечетких уравнений [40].

Рассмотрим нечеткую модель MISO-типа. Введем обозначения:

$\mathbf{x} = (x_1, x_2, \dots, x_m)$ – вектор нечетких входных переменных, $\mathbf{x} \in X$;

y – нечеткая выходная переменная, $y \in Y$;

$A = \{A_1, A_2, \dots, A_m\}$ – множество лингвистических термов, определенных на X с функциями принадлежности $\mu_{A_j}(\mathbf{x}) \in [0, 1]$ для $j = 1, \dots, m$;

$B = \{B_1, B_2, \dots, B_p\}$ – множество лингвистических термов, определенных на Y с функциями принадлежности $\mu_{B_k}(y) \in [0, 1]$ для $k = 1, \dots, p$.

Результат нечеткого вывода B' можно представить в виде

$$B' = A' \bullet R = A' \bullet (A \rightarrow B),$$

где $B' = \{B'_1, B'_2, \dots, B'_p\}$ – нечеткие множества, отражающие вектор значений принадлежностей $\mu_{B'_k}(y)$ ($k = 1, \dots, p$); $A' = \{A'_1, A'_2, \dots, A'_m\}$ – нечеткие множества, отражающие вектор четких значений \mathbf{x}' , с вектором значений принадлежностей $\mu_{A'_j}(\mathbf{x})$ ($j = 1, \dots, m$); « \bullet » – композиционное правило нечеткого вывода.

Нечеткое отношение R , определяющее отображение $A \rightarrow B$, может быть задано реляционной матрицей $\mathbf{R}_{(m \times p)}$:

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1p} \\ r_{21} & r_{22} & \cdots & r_{2p} \\ \cdots & \cdots & r_{jk} & \cdots \\ r_{m1} & r_{m2} & \cdots & r_{mp} \end{bmatrix}. \quad (6.1)$$

где $r_{jk} \in [0, 1]$, $j = 1, \dots, m$, $k = 1, \dots, p$.

Тогда при заданном векторе значений принадлежностей $\mu_{A'_j}(\mathbf{x})$ ($j = 1, \dots, m$) результатом нечеткого вывода, полученного с помощью поэлементно реализованного обобщенного правила S–T-композиции, будет являться вектор значений принадлежностей $\mu_{B'_k}(y)$ ($k = 1, \dots, p$):

$$\mu_{B'_k}(y) = \sum_{j=1}^m (\mu_{A'_j}(\mathbf{x}) \text{ T } r_{jk}), \quad k = 1, \dots, p. \quad (6.2)$$

Дефазифицированный в соответствии с методом центра тяжести выход нечеткой реляционной модели вычисляется следующим образом:

$$y' = \frac{\sum_{k=1}^p [y'_k \sum_{j=1}^m (\mu_{A'_j}(\mathbf{x}) \text{ T } r_{jk})]}{\sum_{k=1}^p \sum_{j=1}^m (\mu_{A'_j}(\mathbf{x}) \text{ T } r_{jk})}, \quad (6.3)$$

где y'_k – центры тяжести (центроиды) нечетких множеств B'_k , $k = 1, \dots, p$.

В композиционном правиле (6.2) могут быть использованы различные S- и T-нормы (см. табл. 4.1).

Отметим, что одним из достоинств нечетких реляционных моделей является то, что в процессе их настройки имеется возможность изменения элементов (в диапазоне $[0, 1]$) реляционной матрицы при том, что лингвистические термы антецедентов и консеквентов остаются неизменными.

6.2. Реляционное представление нечеткого вывода с использованием отдельных правил

Нечеткое множество B'_i , характеризующее результат нечеткого вывода по отдельному i -му правилу из базы типа (4.64):

Π_i : ЕСЛИ x_1 есть A_{i1} И ... И x_j есть A_{ij} И ... И x_m есть A_{im} ,
ТО y есть B_i , $i = 1, \dots, n$,

определяется композицией нечеткого множества A' и нечеткого бинарного отношения $R_i: A_i \rightarrow B_i$:

$$B'_i = A'_i \bullet R_i = A'_i \bullet (A_i \rightarrow B_i) = (\times_{j=1}^m A'_{ij}) \bullet ((\times_{j=1}^m A_{ij}) \rightarrow B_i), \quad (6.4)$$

где нечеткое множество $A'_i = A'_{i1} \times A'_{i2} \times \dots \times A'_{im}$ с функцией принадлежности $\mu_{A'_i}(x) = \mu_{A'_{i1} \times A'_{i2} \times \dots \times A'_{im}}(x_1, x_2, \dots, x_m)$ определено на декартовом произведении $X = X_1 \times X_2 \times \dots \times X_m$; а нечеткое отношение $R_i: A_i \rightarrow B_i$ определено на декартовом произведении $X_1 \times X_2 \times \dots \times X_m \times Y$, т. е. R_i – это нечеткое множество с функцией принадлежности $\mu_{R^i}(x, y) = \mu_{A_i \rightarrow B_i}(x, y) = \mu_{A_{i1} \times A_{i2} \times \dots \times A_{im} \rightarrow B_i}(x_1, x_2, \dots, x_m, y)$.

Исходя из определения композиции нечеткого множества и нечеткого отношения, имеем:

$$\mu_{B'_i}(y) = \sup_{x \in X} \{ \mu_{A'_i}(x) T \mu_{R^i}(x, y) \}, \quad (6.5)$$

где $\sup_{x \in X} \{ \dots \}$ – верхняя граница множества элементов $\{ \dots \}$; T – операция Т-нормы.

Если X является множеством с конечным числом элементов, то композиционное правило (6.5) принимает вид макстриангулярной композиции:

$$\mu_{B'_i}(y) = \max_{x \in X} \{ \mu_{A'_i}(x) T \mu_{R^i}(x, y) \}. \quad (6.6)$$

Таким образом, результат нечеткого вывода по отдельному i -му правилу типа (4.64), и соответственно конкретная форма функции принадлежности $\mu_{B'_i}(y)$, зависит от выбора:

- способа определения декартова произведения нечетких множеств $A'_i = A'_{i1} \times A'_{i2} \times \dots \times A'_{im}$;
- операции нечеткой импликации для нахождения функции принадлежности $\mu_{R^i}(x, y)$;
- операции Т-нормы.

Выражение (6.6) является реляционным представлением нечеткого вывода с использованием отдельного правила.

Предварительно отметим, что если на этапе введения нечеткости вместо нечетких значений входных переменных $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_m$ используются четкие значения x'_1, x'_2, \dots, x'_m , то выражение (6.5) примет вид

$$\mu_{B'_i}(y) = \mu_{A_{i1} \times A_{i2} \times \dots \times A_{im} \rightarrow B_i}(x'_1, x'_2, \dots, x'_m, y), \dots, i = 1, \dots, n. \quad (6.7)$$

Декартово произведение нечетких множеств $A'_{i1} \subseteq X_1, A'_{i2} \subseteq X_2, \dots, A'_{im} \subseteq X_m$, обозначаемое как $A'_i = A'_{i1} \times A'_{i2} \times \dots \times A'_{im}$ и заданное на множестве $X = X_1 \times X_2 \times \dots \times X_m$, можно определить следующими различными способами:

$$\mu_{A'_{i1} \times A'_{i2} \times \dots \times A'_{im}}(x_1, x_2, \dots, x_m) = \min\{\mu_{A'_{i1}}(x_1), \mu_{A'_{i2}}(x_2), \dots, \mu_{A'_{im}}(x_m)\} \quad (6.8)$$

или

$$\mu_{A'_{i1} \times A'_{i2} \times \dots \times A'_{im}}(x_1, x_2, \dots, x_m) = \mu_{A'_{i1}}(x_1) \cdot \mu_{A'_{i2}}(x_2) \cdots \mu_{A'_{im}}(x_m). \quad (6.9)$$

Пример (в данном подразделе использованы примеры из [6]). Предположим:

- база правил состоит из двух правил:

П₁: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} , ТО y есть B_1 ,

П₂: ЕСЛИ x_1 есть A_{21} И x_2 есть A_{22} , ТО y есть B_2 ;

- декартово произведение нечетких множеств задано выражением (6.8);
- нечеткая импликация – min-конъюнкция;
- Т-норма – min-конъюнкция.

В этом случае выражение (6.5) примет следующий вид:

$$\begin{aligned} \mu_{B_i}(y) &= \sup_{x \in X} \left\{ \min \left[\mu_{A_i}(x), \mu_{A_i \rightarrow B_i}(x, y) \right] \right\} = \\ &= \sup_{x \in X} \left\{ \min \left[\mu_{A_i}(x), \min \left(\mu_{A_i}(x), \mu_{B_i}(y) \right) \right] \right\} = \\ &= \sup_{x_1, x_2 \in X} \left\{ \min \left[\mu_{A'_{i1} \times A'_{i2}}(x_1, x_2), \min \left(\mu_{A_{i1} \times A_{i2}}(x_1, x_2), \mu_{B_i}(y) \right) \right] \right\} = \\ &= \sup_{x_1, x_2 \in X} \left\{ \min \left[\min \left(\mu_{A'_{i1}}(x_1), \mu_{A'_{i2}}(x_2) \right), \min \left(\min \left(\mu_{A_{i1}}(x_1), \mu_{A_{i2}}(x_2) \right), \mu_{B_i}(y) \right) \right] \right\} = \\ &= \sup_{x_1, x_2 \in X} \left\{ \min \left[\left(\mu_{A'_{i1}}(x_1), \mu_{A'_{i2}}(x_2) \right), \mu_{A_{i1}}(x_1), \mu_{A_{i2}}(x_2), \mu_{B_i}(y) \right] \right\}. \end{aligned}$$

Пример. Предположим для этой же базы правил, что:

- декартово произведение нечетких множеств задано выражением (6.9);
- в качестве нечеткой импликации используется алгебраическое произведение;
- Т-норма – алгебраическое произведение.

Тогда выражение (6.5) запишется следующим образом:

$$\begin{aligned}\mu_{B'_i}(y) &= \sup_{x \in X} \{ \mu_{A'_i}(x) \cdot \mu_{A_i \rightarrow B_i}(x, y) \} = \sup_{x \in X} \{ \mu_{A'_i}(x) \cdot \mu_{A_i}(x) \cdot \mu_{B_i}(y) \} = \\ &= \sup_{x_1, x_2 \in X} \{ \mu_{A'_{i1}}(x_1) \cdot \mu_{A'_{i2}}(x_2) \cdot \mu_{A_{i1}}(x_1) \cdot \mu_{A_{i2}}(x_2) \cdot \mu_{B_i}(y) \}.\end{aligned}$$

Пример. Предположим для этой же базы правил, что:

- декартово произведение нечетких множеств задано выражением (6.9);
- нечеткая импликация – алгебраическое произведение;
- Т-норма – min-конъюнкция.

В этом случае выражение (6.5) запишется следующим образом:

$$\begin{aligned}\mu_{B'_i}(y) &= \sup_{x \in X} \{ \min[\mu_{A'_i}(x), \mu_{A_i \rightarrow B_i}(x, y)] \} = \\ &= \sup_{x \in X} \{ \min[\mu_{A'_i}(x), \mu_{A_i}(x) \cdot \mu_{B_i}(y)] \} = \\ &= \sup_{x_1, x_2 \in X} \{ \min[\mu_{A'_{i1}}(x_1) \cdot \mu_{A'_{i2}}(x_2), \mu_{A_{i1}}(x_1) \cdot \mu_{A_{i2}}(x_2) \cdot \mu_{B_i}(y)] \}.\end{aligned}$$

При наличии нескольких (p) заключений в нечетких продукционных правилах [например, в правилах типа (4.51)] для каждого из заключений определяются значения степеней истинности $\{\mu_{B'_{i1}}(y_1), \dots, \mu_{B'_{ik}}(y_k), \dots, \mu_{B'_{ip}}(y_p)\}$, $i = 1, \dots, n$, $k = 1, \dots, p$.

$$\begin{aligned}\mu_{B'_{ik}}(y) &= \sup_{x \in X} \{ \mu_{A'_i}(x) T \mu_{A_i \rightarrow B_{ik}}(x, y_k) \}, \\ i &= 1, \dots, n, k = 1, \dots, p.\end{aligned}\tag{6.10}$$

Отметим, что как для отдельных правил, так и индивидуально для каждого их заключения могут быть заданы весовые коэффициенты.

6.3. Реляционное представление нечеткого вывода с использованием базы правил

Как отдельное правило, так и вся база нечетких правил может быть задана нечетким отношением. Обобщенное нечеткое отношение, представляющее всю базу правил типа (4.64), формируется объединением нечетких отношений, представляющих отдельные правила:

$$R = \bigcup_{i=1}^n R_i.\tag{6.11}$$

При этом нечеткое множество B' , характеризующее результат нечеткого вывода по всей базе из n нечетких правил, определяется следующим образом:

$$B' = A' \bullet \bigcup_{i=1}^n R_i = \bigcup_{i=1}^n A' \bullet R_i = \bigcup_{i=1}^n \left(\left(\bigtimes_{j=1}^m A'_{ij} \right) \bullet \left(\left(\bigtimes_{j=1}^m A_{ij} \right) \rightarrow B_i \right) \right). \quad (6.12)$$

Исходя из выражения (6.5), а также из определения объединения нечетких множеств, выражение для агрегированной в результате нечеткого вывода функции принадлежности $\mu_{B'}(y)$ на основе, например, связки «максимум» примет вид

$$\mu_{B'}(y) = \sup_{x \in X} \{ \mu_{A'}(x) T \max_{i=1, \dots, n} \mu_{R^i}(x, y) \}. \quad (6.13)$$

Если в качестве Т-нормы и нечеткой импликации используется операция min-конъюнкции либо операция нечеткого произведения (prod), то выражение (6.13) может быть задано в следующем виде:

$$\mu_{B'}(y) = \max_{i=1, \dots, n} \sup_{x \in X} \{ \mu_{A'}(x) T \mu_{R^i}(x, y) \} = \max_{i=1, \dots, n} \mu_{B'_i}(y). \quad (6.14)$$

Выражение (6.14) является реляционным представлением нечеткого вывода с использованием базы нечетких правил.

6.4. Подобие нечетких реляционных и продукционных моделей

Более наглядно подобие между нечеткими реляционными и продукционными моделями можно проиллюстрировать следующим примером.

Допустим, для нечеткой реляционной модели:

- Т-норма – алгебраическое произведение;
- S-норма – среднее арифметическое.

Тогда выражение (6.3) запишется в следующем виде:

$$y' = \frac{\sum_{k=1}^p \left\{ y'_k \frac{1}{m} \sum_{j=1}^m (\mu_{A'_j}(x) T r_{jk}) \right\}}{\sum_{k=1}^p \left\{ \frac{1}{m} \sum_{j=1}^m (\mu_{A'_j}(x) T r_{jk}) \right\}}, \quad (6.15)$$

где y'_k – центры тяжести (центроиды) нечетких множеств B'_k , $k = 1, \dots, p$.

Таким образом, на выходе нечеткой реляционной модели при сделанных допущениях получим:

$$y' = \frac{\sum_{k=1}^p \sum_{j=1}^m \{ y'_k \cdot \mu_{A'_j}(\mathbf{x}) \cdot r_{jk} \}}{\sum_{k=1}^p \sum_{j=1}^m \{ \mu_{A'_j}(\mathbf{x}) \cdot r_{jk} \}}. \quad (6.16)$$

Рассмотрим теперь нечеткую производственную модель типа (4.33) с заданными весами правил w_i :

$$\Pi_i: \text{ЕСЛИ } \mathbf{x} \text{ есть } A_i \text{ ТО } y = B_i(w_i), i = 1, \dots, n, \quad (6.17)$$

где \mathbf{x} – многомерная входная переменная; A_i – многомерные нечеткие множества антецедента; y – выходная переменная; B_i – нечеткие множества консеквента; w_i – вес i -го правила.

Допустим, для нечеткой производственной модели:

- декартово произведение нечетких множеств – алгебраическое произведение;
- нечеткая импликация – алгебраическое произведение;
- метод дефазификации – средний центр (centre average defuzzification) (4.91).

Тогда на выходе нечеткой производственной модели получим

$$y' = \frac{\sum_{i=1}^n \{ y'_i \cdot \mu_{A'_i}(\mathbf{x}) \cdot w_i \}}{\sum_{i=1}^n \{ \mu_{A'_i}(\mathbf{x}) \cdot w_i \}}. \quad (6.18)$$

Если $n = mp$, то каждая группа из p правил определяется одним и тем же многомерным нечетким множеством антецедента, а каждая группа из m правил – одним и тем же значением консеквента, т. е. выражение (6.18) можно записать в виде, подобном (6.16):

$$y' = \frac{\sum_{k=1}^p \sum_{j=1}^m \{ y'_k \cdot \mu_{A'_j}(\mathbf{x}) \cdot w_{jk} \}}{\sum_{k=1}^p \sum_{j=1}^m \{ \mu_{A'_j}(\mathbf{x}) \cdot r_{jk} \}}, \quad (6.19)$$

где w_{jk} – реиндексированный вес w_i .

ВЫВОДЫ

Рассмотрены нечеткие производственные модели, являющиеся наиболее общим видом нечетких моделей, используемых для описания, анализа и моделирования сложных систем и процессов. Проанализированы и классифицированы основные компоненты, определяющие создание и использование нечетких производственных моделей, а именно:

- способ (схема) нечеткого вывода заключений;
- база нечетких производственных правил;
- процедура введения нечеткости (fuzzification);
- процедура агрегирования (aggregation) степени истинности предпосылок по каждому из нечетких производственных правил;
- процедура активизации (activation) заключений каждого из нечетких производственных правил;
- процедура аккумулирования (accumulation) активизированных заключений всех нечетких производственных правил для каждой выходной переменной;
- процедура приведения к четкости (defuzzification) для каждой аккумулированной выходной переменной;
- процедура параметрической оптимизации конечной базы нечетких правил.

Рассмотрены основные алгоритмы нечеткого вывода, получившие в настоящее время наибольшее распространение: Мамдани, Ларсена, Цукамото, Такаги–Сугэно, а также алгоритм вывода на основе нечеткой производственной модели с адаптацией операций над нечеткими множествами.

Обоснована эффективность использования нечетких производственных моделей и представлены результаты (в том числе и конструктивные) доказательств возможности использования нечетких моделей различных типов в качестве универсальных аппроксиматоров.

Представлены нечеткие реляционные модели, а также проиллюстрировано их подобие нечетким производственным моделям.

СПИСОК ЛИТЕРАТУРЫ

1. Борисов В. В., Федулов А. С. Нечеткие производственные модели и сети. Смоленск: Изд-во ВА ВПВО ВС РФ, 2005.
2. Fuller R. Neural Fuzzy Systems. Publishing House: Abo Akademi University, 1995.
3. Cao Z., Kandel A. Applicability of the some fuzzy implication operations // Fuzzy Sets and Systems. 1989. V. 31. P. 151–186.
4. Fukami S., Mizumoto M., Tanaka K. Some considerations of fuzzy conditional inference // Fuzzy Sets and Systems. 1980. V. 4. P. 243–273.
5. Kiszka J. B., Kochanska M. E., Sliwinska D. S. The influence of some fuzzy implication operators on the accuracy of fuzzy model // Fuzzy Sets and Systems. 1985. V. 15. P. 111–128; 223–240.
6. Рутковская Д., Пилинковский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия – Телеком, 2004.
7. Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ–Петербург, 2003.
8. Круглов В. В., Борисов В. В. Гибридные нейронные сети. Смоленск: Русич, 2001.
9. Борисов А. Н., Крумберг О. А., Федоров И. П. Принятие решений на основе нечетких моделей: Примеры использования. Рига: Зинатне, 1990.
10. Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика. М.: Горячая линия – Телеком, 2001.
11. Mamdani E. H. Application of fuzzy logic to approximate reasoning using linguistic systems // Fuzzy Sets and Systems. 1977. V. 26. P. 1182–1191.
12. Takagi T., Sugeno M. Fuzzy identification of systems and its application to modeling and control// IEEE Transactions on Systems, Man and Cybernetics. 1985. V. 15, № 1. P. 116–132.
13. Esoglu A. O., Bellman R. T. Fuzzy dynamic programming and its extensions// Fuzzy Sets and Decis. Anal./ Ed. by H. J. Zimmerman, L. A. Zadeh, B. R. Gains. North-Holand: Elsevier Science Publishers, 1984. P. 11–28.

14. *Foster D. H.* Fuzzy topological groups// *J. Math. Anal. Appl.* 1979. V. 67. P. 549–564.
15. *Борисов В. В., Федулов А. С.* Нечеткие оценочные модели сложных систем с учетом согласования неравнозначных целей// *Нейрокомпьютеры: разработка, применение.* 2003. № 5. С. 3–12.
16. *Заде Л.* Понятие лингвистической переменной и его применение к принятию приближенных решений. М.: Мир, 1976.
17. *Wenstop F.* Deductive verbal models of organizations/ *Int. J. Man-Machine Studies.* 1976. № 8. P. 293–311.
18. *Силов В. Б.* Принятие решений в нечеткой обстановке. М.: ИНПРО-РЕС, 1995.
19. *Борисов В. В., Федулов А. С.* Обобщенные нечеткие когнитивные карты // *Нейрокомпьютеры: разработка, применение.* 2004. № 4. С. 3–20.
20. *Борисов В. В., Федулов А. С., Мнев В. И.* Нечеткий логический вывод в условиях неравнозначности целевых функций/ *Информационный бюллетень Академии военных наук.* Смоленск: Изд-во ВУ ВПВО ВС РФ, 2001.
21. *Babuška R.* Fuzzy Modeling for Control. Boston, USA: Kluwer Academic Publishers, 1998.
22. *Wang L. X., Mendel J. M.* Generating fuzzy rules by learning from examples. *IEEE Transaction on Systems, Man and Cybernetics.* 1992. V. 22. № 6. P. 1414–1427.
23. *Pedrycz W.* Fuzzy Control and Fuzzy Systems. New York: John Wiley and Sons, 1993.
24. *Дюбуа Д., Прад А.* Теория возможностей. Приложения к представлению знаний в информатике. М.: Радио и связь, 1990.
25. *Zimmermann H. J., Zysno P.* Latent Connectives in Human Decision Making// *Fuzzy Sets and Systems.* 1980. V. 4. № 1. P. 37–51.
26. *Kosko B.* Fuzzy systems as universal approximators// In Proc. of the IEEE Int. Conf. on Fuzzy Systems, San Diego, 1992. P. 1153–1162.
27. *Wang L. X.* Fuzzy systems are universal approximators// In Proc. of the IEEE Int. Conf. on Fuzzy Systems, San Diego, 1992. P. 1163–1169.

28. Castro J. L. Fuzzy logic controllers are universal approximators// IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics. 1995. V. 25. P. 629–635.
29. Ying H. Sufficient conditions on uniform approximation of multivariate functions by general Takagi–Sugeno fuzzy systems with linear rule consequents// IEEE Transactions on Systems, Man and Cybernetics. Part A, 1998. V. 28. № 4. P. 515–520.
30. Zeng K., Zhang N.-Y., Xu W.-L. A contrarative study on sufficient conditions for Takagi–Sugeno fuzzy systems as universal approximators// IEEE Trans. on Fuzzy Systems. 2000. V. 8. № 6. P. 773–780.
31. Круглов В. В. Сравнение алгоритмов Мамдани и Сугэно в задаче аппроксимации функции // Нейрокомпьютеры: разработка, применение. 2003. № 5. С. 34–38.
32. Bauer P., Klement E. P., Leikermoser A., Moser B. Modeling of control functions by fuzzy controllers/ In H. Nguyen, M. Sugeno, R. Tong and R. R. Yager, editor/ Theoretical Aspects of Fuzzy Control. NY: Wiley, 1995. P. 91–116.
33. Li Y.-M., Shi Z.-K., Li Z.-H. Approximation theory fuzzy systems based upon genuine many-valued implications – SISO cases // Fuzzy Sets and Systems. 2002. V. 130. P. 147–157.
34. Li Y.-M., Shi Z.-K., Li Z.-H. Approximation theory fuzzy systems based upon genuine many-valued implications – MIMO cases // Fuzzy Sets and Systems. 2002. V. 130. P. 159–174.
35. Raju G. V. S., Zhou J., Kisner R. A. Hierarchical fuzzy control// Int. J. Control. 1991. V. 54. P. 1201–1216.
36. Joo M. G., Lee J. S. Universal approximation by hierarchical fuzzy system with constraints on the fuzzy rule // Fuzzy Sets and Systems. 2002. V. 130. P. 175–188.
37. Tikk D., Baranyi P., Yam Y., Koczy L. T. Stability of a new interpolation method // In Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics (IEEE SMC'99), Tokyo, Japan. 1999. V. 3. P. 7–9.
38. Setness M., Babuška R. Fuzzy relational classifier trained by fuzzy clustering // IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics. 1999. V. 29. №. 5. P. 619–625.

39. Branco P. J. C., Dente J. A. A fuzzy relational identification algorithm and its application to predict the behavior of a motor drive system // Fuzzy Sets and Systems. 2000. V. 109. P. 343–354.
40. Fukumura H., Kamiya K., Miyagi H., Yamashita K. Study on the solution of fuzzy relation equation with interval values // In Proc. of Int. Tech. Conf. on Circuits/Systems, Computers and Communications (ITC-CSCC'97). Okinawa, Japan. V. 1. 1997. P. 495–498.

Часть III

НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ

Несмотря на несомненные достоинства нечетких производственных моделей при решении целого ряда задач, в которых информация о системе, ее параметрах, а также о входах, выходах и состояниях системы является ненадежной и слабоформализованной, заключающиеся прежде всего в описании модели на языке, слишком к естественному, в универсальности, эффективности, вместе с тем для нечетких производственных моделей характерны и определенные недостатки:

- исходный набор нечетких правил формулируется экспертом и может оказаться неполным или противоречивым;
- субъективность в выборе вида и параметров функций принадлежности в нечетких высказываниях правил;
- отсутствует возможность автоматического приобретения знаний.

Формально нечеткие производственные модели и алгоритмы нечеткого вывода на их основе могут быть представлены в виде *нечетких производственных сетей*, по структуре идентичных многослойным нейронным сетям с прямым распространением сигнала (feedforward), элементы каждого слоя (или совокупности слоев) которой реализует отдельный этап нечеткого вывода в нечеткой производственной модели:

- первый слой нейронов выполняет функцию введения нечеткости (fuzzification);
- скрытые слои отображают совокупность нечетких правил и реализуют алгоритм нечеткого вывода;
- последний слой выполняет функцию приведения к четкости (defuzzification) выходной переменной.

Для устранения указанных выше недостатков в ряде работ предложено создавать нечеткие производственные модели адаптивными (с коррекцией в процессе и по результатам их функционирования как состава правил в базе, так и параметров функций принадлежности), а также реализовывать различные компоненты этих моделей на основе нейросетевой технологии.

К настоящему времени предложено и изучено большое количество нейроподобных элементов (нейронов) и искусственных нейронных сетей. Однако, как правило, полагают, что искусственный нейрон состоит из элементов трех типов: умножителей (синапсов), сумматора и нелинейного преобразователя. Синапсы осуществляют связь между нейронами в сети, умножают входной сигнал на число, характеризующее силу связи, (вес синапса). Сумматор выполняет сложение сигналов, поступающих по синаптическим связям от других нейронов, и внешних входных сигналов. Нелинейный преобразователь реализует нелинейную функцию одного аргумента – выхода сумматора, называемую функцией активизации или передаточной функцией нейрона. Сам нейрон в целом реализует скалярную функцию векторного аргумента.

Представление же различных компонентов нечеткой продукционной модели с использованием нейронных сетей предполагает прежде всего использование формализмов, принятых при описании и функционирования нейронов и искусственных нейронных сетей.

Исходя из этого, можно предложить следующую классификацию способов и средств интеграции нечетких продукционных моделей с нейронными сетями:

1. Нечеткие продукционные сети, обладающие возможностью параметрической оптимизации нечетких правил (параметров функций принадлежности, активационных функций, весов связей) на основе алгоритмов обучения (подобных алгоритмам обучения обычных нейронных сетей, например алгоритму обратного распространения ошибки backpropagation) с использованием экспериментальных данных (обучающей выборки).
2. Формирование функций принадлежности предпосылок и заключений нечетких продукционных правил с использованием нейронных сетей.
3. Нечеткие продукционные сети с использованием нейронных сетей для формирования предпосылок нечетких продукционных правил.
4. Нечеткие продукционные сети с использованием нейронных сетей для формирования заключений нечетких продукционных правил.
5. Нечеткие продукционные сети с использованием нечетких нейронных сетей для дефазификации выходной переменной.
6. Нечеткие продукционные сети с использованием нейронных сетей для разбиения пространств входных переменных и формирования для них многомерных функций принадлежности предпосылок.

7. Нечеткие производственные сети с комбинированием указанных выше реализаций компонентов нечетких производственных моделей на основе нейронных сетей.
8. Нечеткие производственные сети с адаптацией структуры под особенности нейронных сетей, построенных на основе традиционных нейронов со взвешенным суммированием входных сигналов и нелинейной функцией активизации.

В контексте изложенного в данной книге для соблюдения единства терминологии, а также для того, чтобы не проводить аналогию с различными типами нейронных нечетких сетей (*neural fuzzy network*), назовем сети перечисленных выше типов *нечеткими нейронными производственными сетями*.

Глава 7. НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ С ПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИЕЙ ПРАВИЛ НА ОСНОВЕ АЛГОРИТМОВ ОБУЧЕНИЯ

7.1. Нечеткие нейронные производственные сети типа ANFIS

Отдельный тип нечетких нейронных производственных сетей, обучаемых на основе классических градиентных алгоритмов, представляют нечеткие сети типа ANFIS (Adaptive Network-based Fuzzy Inference System – адаптивная сетевая нечеткая система вывода), предложенные Чангом (Jang) в 1992 г. [1].

7.1.1. Описание нечеткой нейронной производственной сети ANFIS для реализации алгоритма нечеткого вывода Цукамото

Рассмотрим нечеткую сеть типа ANFIS, реализующую, к примеру, алгоритм нечеткого вывода Цукамото и основанную на следующих правилах:

- П₁: ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} И x_3 есть A_{13} , ТО y есть B_1 ,
П₂: ЕСЛИ x_1 есть A_{21} И x_2 есть A_{22} И x_3 есть A_{23} , ТО y есть B_2 ,
П₃: ЕСЛИ x_1 есть A_{31} И x_2 есть A_{32} И x_3 есть A_{33} , ТО y есть B_3 .

Допустим, нечеткие множества в этих правилах имеют следующие функции принадлежности сигмоидного типа:

$$\mu_{A_{11}}(x_1) = \frac{1}{1 + \exp(b_{11}(x_1 - a_{11}))},$$

$$\mu_{A_{12}}(x_2) = \frac{1}{1 + \exp(b_{12}(x_2 - a_{12}))},$$

$$\mu_{A_{13}}(x_3) = \frac{1}{1 + \exp(b_{13}(x_3 - a_{13}))},$$

$$\mu_{A_{21}}(x_1) = \frac{1}{1 + \exp(-b_{21}(x_1 - a_{21}))},$$

$$\mu_{A_{22}}(x_2) = \frac{1}{1 + \exp(-b_{22}(x_2 - a_{22}))},$$

$$\mu_{A_{23}}(x_3) = \frac{1}{1 + \exp(b_{23}(x_3 - a_{23}))},$$

$$\mu_{A_{31}}(x_1) = \frac{1}{1 + \exp(-b_{31}(x_1 - a_{31}))},$$

$$\mu_{A_{32}}(x_2) = \frac{1}{1 + \exp(-b_{32}(x_2 - a_{32}))},$$

$$\mu_{A_{33}}(x_3) = \frac{1}{1 + \exp(-b_{33}(x_3 - a_{33}))},$$

$$\mu_{B_1}(y) = \frac{1}{1 + \exp(-d_1(y + c_1 - c_2))},$$

$$\mu_{B_2}(y) = \frac{1}{1 + \exp(-d_2(y - c_2))},$$

$$\mu_{B_3}(y) = \frac{1}{1 + \exp(d_3(y - c_3))}.$$

Примечание. Еще раз напомним, что особенностью алгоритма нечеткого вывода Цукамото является то, что в заключениях правил используются монотонные (возрастающие или убывающие) функции.

Нечеткая импликация – операция min-конъюнкции.

На рис. 7.1 приведена иллюстрация алгоритма нечеткого вывода Цукамото в соответствии с рассматриваемым примером.

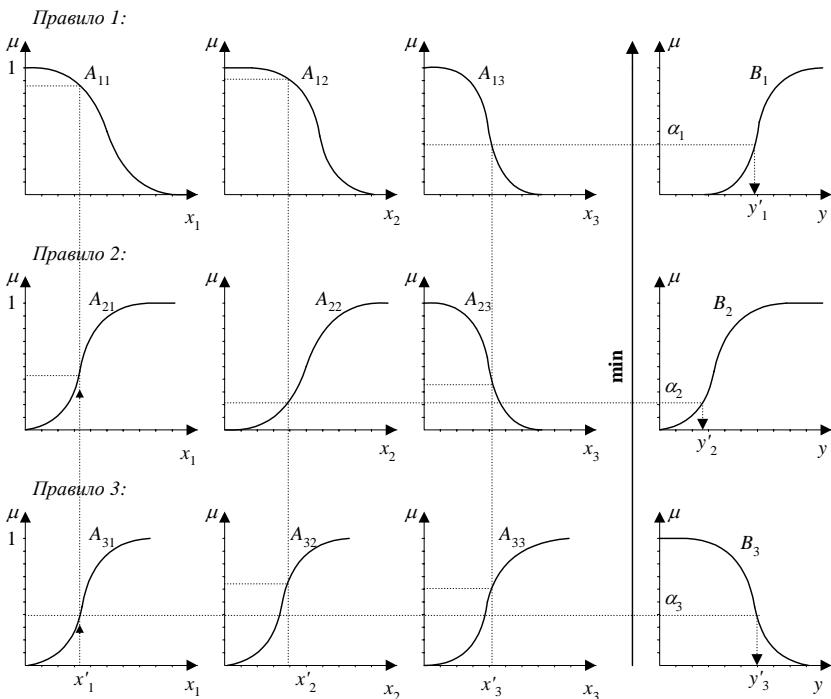


Рис. 7.1. Иллюстрация алгоритма нечеткого вывода Цукамото в нечеткой нейронной производственной сети типа ANFIS

7.1.2. Структура нечеткой производственной сети ANFIS

Структура нечеткой нейронной производственной сети типа ANFIS, реализующая алгоритм нечеткого вывода Цукамото в соответствии с рассматриваемым на рис. 7.1 примером, приведена на рис. 7.2 [2].

Данная сеть может быть описана следующим образом.

Слой 1. Выходы элементов этого слоя представляют собой значения функций принадлежности $\mu_{A_j}(x'_j)$ при конкретных (заданных) значениях входных переменных.

Слой 2. Элементы второго слоя выполняют агрегирование степеней истинности предпосылок каждого правила базы в соответствии с операцией Т-нормы, с качеством которой здесь используется операция \min -конъюнкции, по формулам

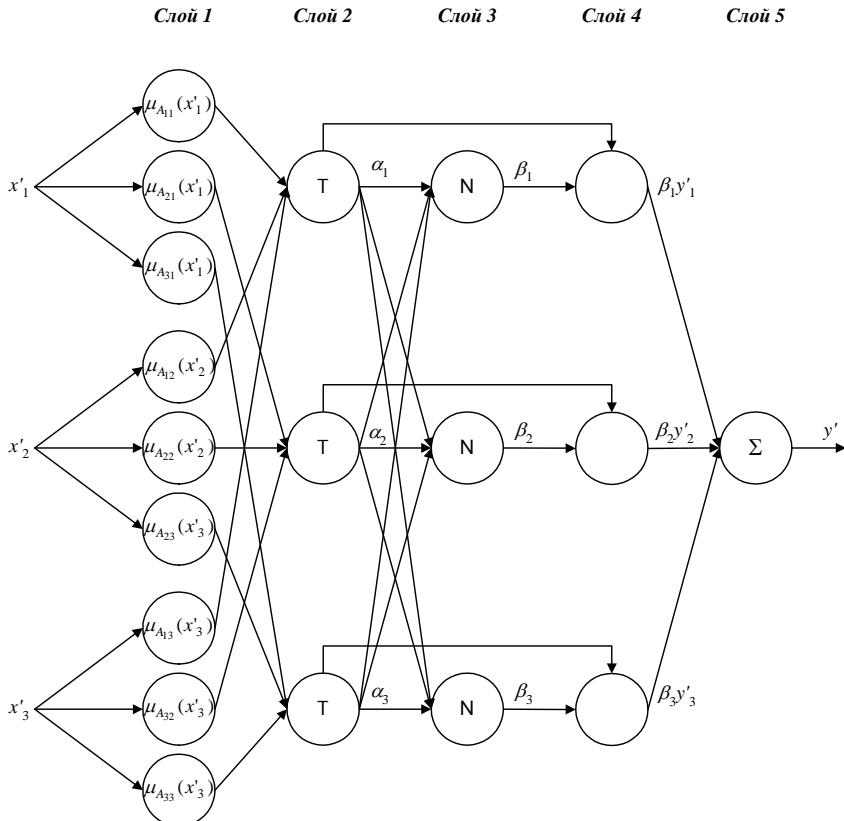


Рис. 7.2. Структура нечеткой нейронной продукционной сети ANFIS, реализующая алгоритм нечеткого вывода Цукамото

$$\alpha_1 = \min\{A_{11}(x_1), A_{12}(x_2), A_{13}(x_3)\},$$

$$\alpha_2 = \min\{A_{21}(x_1), A_{22}(x_2), A_{23}(x_3)\},$$

$$\alpha_3 = \min\{A_{31}(x_1), A_{32}(x_2), A_{33}(x_3)\}.$$

Слой 3. Элементы этого слоя выполняют нормализацию и вычисляют следующие значения:

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3}, \quad \beta_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3}, \quad \beta_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3}.$$

Слой 4. Элементы данного слоя:

- во-первых, вычисляют значения заключений по каждому правилу:

$$y'_1 = B_1^{-1}(\alpha_1) = c_1 + c_2 + \frac{1}{d_1} \ln \frac{1-\alpha_1}{\alpha_1},$$

$$y'_2 = B_2^{-1}(\alpha_2) = c_2 + \frac{1}{d_2} \ln \frac{1-\alpha_2}{\alpha_2},$$

$$y'_3 = B_3^{-1}(\alpha_3) = c_3 + \frac{1}{d_3} \ln \frac{1-\alpha_3}{\alpha_3};$$

- во-вторых, приводят эти результаты к виду, удобному для вычисления выхода нечеткой сети:

$$\beta_1 y'_1 = \beta_1 B^{-1}(\alpha_1), \quad \beta_2 y'_2 = \beta_2 B^{-1}(\alpha_2), \quad \beta_3 y'_3 = \beta_3 B^{-1}(\alpha_3).$$

Слой 5. Единственный элемент этого слоя вычисляет выход сети:

$$y' = \beta_1 y'_1 + \beta_2 y'_2 + \beta_3 y'_3.$$

Параметрическими слоями нечеткой нейронной продукционной сети ANFIS, т. е. слоями, параметры элементов в которых будут настраиваться в ходе обучения, являются первый и четвертый, а настраиваемыми в процессе обучения параметрами служат:

- в первом слое – нелинейные параметры a_{ij} , b_{ij} функций принадлежности $\mu_{A_{ij}}(x_j)$ нечетких множеств предпосылок правил;
- в четвертом слое – нелинейные параметры c_i , d_i функций принадлежности $\mu_{B_i}(y)$ нечетких множеств заключений правил.

Подобно вышерассмотренному примеру с помощью нечеткой нейронной продукционной сети ANFIS могут быть реализованы и другие алгоритмы нечеткого вывода.

7.1.3. Обучение нечеткой продукционной сети ANFIS

Так как нечеткая продукционная сеть ANFIS представляется в виде многослойной структуры с прямым распространением сигнала, а значение выходной переменной можно изменять, корректируя параметры элементов слоев, то для обучения этой сети можно использовать алгоритм обратного распространения ошибки, который относится к классу градиентных алгоритмов.

Пусть задана обучающая выборка, состоящая из множества примеров $(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)})$, где $k = 1, \dots, K$, $x_1^{(k)}, x_2^{(k)}, x_m^{(k)}$ – значения входных переменных x_1, x_2, \dots, x_m , $y^{(k)}$ – эталонное значение выходной переменной y в k -м примере, K – общее число примеров в обучающей выборке.

Еще раз отметим, что для данной сети настраиваемыми параметрами являются параметры a_{ij}, b_{ij} и c_i, d_i ($i = 1, \dots, n; j = 1, \dots, m$).

Этап 1. Для каждого примера из обучающей выборки по значениям входных переменных $x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}$ ($k = 1, \dots, K$) нечеткая сеть рассчитывает значения выходной переменной $y'^{(k)}$.

Этап 2. Вычисляется функция ошибки для всех примеров обучающей выборки:

$$E^{(k)} = \frac{1}{2} (y'^{(k)} - y^{(k)})^2, \quad k = 1, \dots, K. \quad (7.1)$$

В данном случае функция ошибки может быть записана как функция, зависящая от следующих аргументов:

$$E^{(k)} = E^{(k)}(a_{ij}, b_{ij}, c_i, d_i) = \frac{1}{2} (y'^{(k)}(a_{ij}, b_{ij}, c_i, d_i) - y^{(k)})^2, \\ k = 1, \dots, K, \quad (7.2)$$

Этап 3. Корректируются значения a_{ij}, b_{ij} и c_i, d_i ($i = 1, \dots, n; j = 1, \dots, m$) по каждому k -му примеру ($k = 1, \dots, K$) обучающей выборки, исходя из соотношений

$$\begin{aligned} a_{ij}(t+1) &:= a_{ij}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial a_{ij}}, \\ b_{ij}(t+1) &:= b_{ij}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial b_{ij}}, \\ c_i(t+1) &:= c_i(t) - \eta \frac{\partial E^{(k)}(t)}{\partial c_i}, \\ d_i(t+1) &:= d_i(t) - \eta \frac{\partial E^{(k)}(t)}{\partial d_i}, \end{aligned} \quad (7.3)$$

где t – номер итерации обучения; $\eta \in [0, 1]$ – коэффициент, характеризующий скорость обучения.

Так, для рассмотренного выше примера корректировка параметров функций принадлежности $\mu_{B_i}(y)$ осуществляется по формулам

$$c_1 := c_1 - \eta \frac{\partial E^{(k)}}{\partial c_1} = c_1 + \eta(y'^{(k)} - y^{(k)}) \frac{\alpha_1 + \alpha_2 + \alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} = c_1 + \eta(y'^{(k)} - y^{(k)}),$$

$$c_2 := c_2 - \eta \frac{\partial E^{(k)}}{\partial c_2} = c_2 + \eta(y'^{(k)} - y^{(k)}) \frac{\alpha_1 + \alpha_2 + \alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} = c_2 + \eta(y'^{(k)} - y^{(k)}),$$

$$c_3 := c_3 - \eta \frac{\partial E^{(k)}}{\partial c_3} = c_3 + \eta(y'^{(k)} - y^{(k)}),$$

$$d_1 := d_1 - \eta \frac{\partial E^{(k)}}{\partial d_1} = d_1 - \frac{\eta}{d_1^2} (y'^{(k)} - y^{(k)}) \frac{\alpha_1 + \alpha_2 - \alpha_3}{\alpha_1 + \alpha_2 + \alpha_3},$$

$$d_2 := d_2 - \eta \frac{\partial E^{(k)}}{\partial d_2} = d_2 - \frac{\eta}{d_2^2} (y'^{(k)} - y^{(k)}) \frac{\alpha_1 + \alpha_2 - \alpha_3}{\alpha_1 + \alpha_2 + \alpha_3},$$

$$d_3 := d_3 - \eta \frac{\partial E^{(k)}}{\partial d_3} = d_3 - \frac{\eta}{d_3^2} (y'^{(k)} - y^{(k)}) \frac{\alpha_1 + \alpha_2 - \alpha_3}{\alpha_1 + \alpha_2 + \alpha_3}.$$

Соответствующие выражения могут быть получены и для параметров a_{ij} , b_{ij} функций принадлежности $\mu_{A_{ij}}(x_j)$.

Этапы 1–3 итерационно повторяются, и процедура корректировки значений всех параметров считается завершенной в случае, если:

- либо значение функции ошибки по каждому примеру обучающей выборки не превышает некоторого установленного порога:

$$E^{(k)} < \varepsilon, k = 1, \dots, K; \quad (7.4)$$

- либо оценка средней суммарной погрешности нечеткой производственной модели с учетом всех примеров обучающей выборки не превышает некоторого установленного порога:

$$E = \frac{1}{K} \sum_{k=1}^K (y'^{(k)} - y^{(k)})^2 < \varepsilon. \quad (7.5)$$

В этом случае считается, что нечеткая сеть успешно обучилась.

7.1.4. Нечеткая нейронная производственная сеть ANFIS для реализации алгоритма нечеткого вывода Такаги–Сугэно

В качестве еще одного примера нечеткой производственной сети данного типа рассмотрим ANFIS-сеть, реализующую алгоритм нечеткого вывода Такаги–Сугэно [3]. Пусть заданы правила следующего вида:

- Π_1 : ЕСЛИ x_1 есть A_{11} И x_2 есть A_{12} И, ТО $y_1 = a_1 x_1 + b_1 x_2$,
 Π_2 : ЕСЛИ x_1 есть A_{21} И x_2 есть A_{22} И, ТО $y_2 = a_2 x_1 + b_2 x_2$.

Структура нечеткой нейронной производственной сети типа ANFIS, реализующая алгоритм нечеткого вывода Такаги–Сугэно в соответствии с рассматриваемым примером, приведена на рис. 7.3.

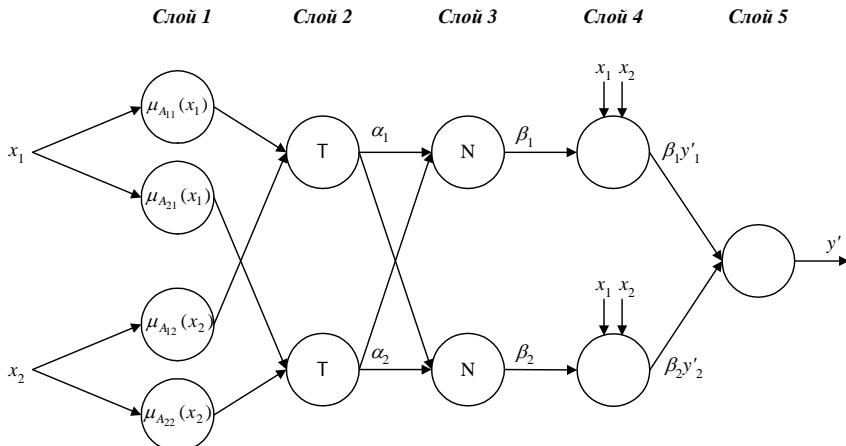


Рис. 7.3. Структура нечеткой нейронной производственной сети ANFIS, реализующая алгоритм нечеткого вывода Такаги–Сугэно

Слой 1. Выходы элементов слоя представляют собой значения функций принадлежности $\mu_{A_{ij}}(x'_j)$ при заданных значениях входных переменных. И например, для колоколообразных функций принадлежности:

$$\mu_{A_{ij}}(x_j) = \exp\left[-\frac{1}{2}\left(\frac{x_j - a_{ij}}{b_{ij}}\right)^2\right].$$

Слой 2. Элементы слоя выполняют агрегирование степеней истинности предпосылок каждого правила в соответствии с операцией Т-нормы, в качестве которой, например, может использоваться операция минимума:

$$\alpha_1 = \min\{A_{11}(x_1), A_{12}(x_2)\},$$

$$\alpha_2 = \min\{A_{21}(x_1), A_{22}(x_2)\}.$$

Слой 3. Элементы данного слоя вычисляют β_i – нормализованные значения α_i :

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2}, \quad \beta_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2}.$$

Слои 4 и 5. Элементы этих слоев позволяют сформировать на выходе сети дефазифицированное значение:

$$y' = \frac{\alpha_1 y_1 + \alpha_2 y_2}{\alpha_1 + \alpha_2} = \beta_1 y'_1 + \beta_2 y'_2,$$

где β_1 и β_2 – нормализованные значения α_1 и α_2 :

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2}, \quad \beta_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2}.$$

Если задано четкое множество примеров обучающей выборки $\{(x^{(k)}, y^{(k)}), k = 1, \dots, K\}$, то параметры сети (a_{ij} и b_{ij}), которые определяют форму функций принадлежности, могут быть настроены с использованием алгоритмов обучения, например на основе градиентных методов.

7.2. Нечеткая нейронная производственная сеть Ванга–Менделя

7.2.1. Описание нечеткой нейронной производственной сети

Нечеткая нейронная производственная сеть Ванга–Менделя (L.–X. Wang and J. M. Mendel) [4, 5] реализует нечеткую производственную модель, основанную на правилах типа (4.33):

П_i: ЕСЛИ x_1 есть A_{i1} И ... И x_j есть A_{ij} И ... И x_m есть A_{im} ,
ТО y есть B_i , $i = 1, \dots, n$,

Кроме того, выполняемый данной нечеткой нейронной производственной сетью алгоритм нечеткого вывода базируется на следующих положениях:

- входные переменные являются четкими;
- функции принадлежности всех нечетких множеств представляются функцией Гаусса;
- декартово произведение нечетких множеств задано выражением (5.6);
- нечеткая импликация – нечеткое произведение;
- Т-норма – нечеткое произведение;
- аккумулирование активизированных заключений правил не проводится;
- метод дефазификации – средний центр (centre average defuzzification) [см. выражение (4.60)].

Таким образом, нечеткую производную модель и механизм нечеткого вывода для данной нечеткой нейронной производной сети можно отобразить следующим выражением:

$$\begin{aligned} \mu_{B'_i}(y) &= \sup_{x \in X} \{ \mu_{A'_i}(x) \cap \mu_{A_i \rightarrow B_i}(x, y) \} = \\ &= \sup_{x \in X} \{ \mu_{A'_i}(x) \cdot \mu_{A_i \rightarrow B_i}(x, y) \} = \sup_{x \in X} \{ \mu_{A'_i}(x) \mu_{A_{ij}}(x) \mu_{B_i}(y) \} = \\ &= \sup_{x_1, \dots, x_m \in X} \{ \mu_{B_i}(y) \prod_{j=1}^m (\mu_{A'_{ij}}(x_j) \mu_{A_{ij}}(x_j)) \}. \end{aligned} \quad (7.6)$$

В данном случае входные переменные x_1, \dots, x_m являются четкими, поэтому выражение (7.6) принимает следующий вид:

$$\mu_{B'_i}(y) = \mu_{B_i}(y) \prod_{j=1}^m \mu_{A_{ij}}(x'_j). \quad (7.7)$$

Так как аккумулирования активизированных заключений правил не проводится, а методом дефазификации является метод среднего центра, то дефазифицированное значение выходной переменной определяется по формуле

$$\begin{aligned} y' &= \frac{\sum_{i=1}^n \left(\arg \max_y \mu_{B'_i}(y) \mu_{B'_i}(y) \right)}{\sum_{i=1}^n \mu_{B'_i}(y)} = \\ &= \frac{\sum_{i=1}^n \left(\arg \max_y \mu_{B'_i}(y) \mu_{B_i}(y) \prod_{j=1}^m \mu_{A_{ij}}(x'_j) \right)}{\sum_{i=1}^n \left(\mu_{B_i}(y) \prod_{j=1}^m \mu_{A_{ij}}(x'_j) \right)}. \end{aligned} \quad (7.8)$$

С учетом того, что максимальное значение, которое $\mu_{B_i}(y)$ может принять в точке $\arg \max_y \mu_{B'_i}(y)$, равно единице, выражение (7.8) приводится к следующему виду:

$$y' = \frac{\sum_{i=1}^n \left(\arg \max_y \mu_{B'_i}(y) \prod_{j=1}^m \mu_{A_{ij}}(x'_j) \right)}{\sum_{i=1}^n \prod_{j=1}^m \mu_{A_{ij}}(x'_j)}. \quad (7.9)$$

В нашем случае функции принадлежности всех нечетких множеств представляются функцией Гаусса, поэтому выражение (7.9) примет вид

$$y' = \frac{\sum_{i=1}^n \left(\arg \max_y \left(\exp \left[-\frac{y - c_i}{d_i} \right] \right) \cdot \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right] \right)}{\sum_{i=1}^n \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right]}, \quad (7.10)$$

где c_i, d_i – соответственно центры и ширина гауссовых функций, представляющих функции принадлежности нечетких множеств B_i заключений правил; a_{ij}, b_{ij} – соответственно центры и ширина гауссовых функций, представляющих функции принадлежности нечетких множеств A_{ij} предпосылок правил.

В окончательном виде

$$y' = \frac{\sum_{i=1}^n \left(c_i \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right] \right)}{\sum_{i=1}^n \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right]}. \quad (7.11)$$

Примечание. Очевидно, что для данного типа нечеткой сети, построенной исходя из вышеуказанных положений, вид функции принадлежности для выходной переменной не имеет значения. И в этом случае реализуемая данной нечеткой сетью модель сводится к нечеткой производственной модели на основе правил типа (4.28) и алгоритма нечеткого вывода Такаги–Сугэно 0-го порядка.

7.2.2. Структура нечеткой нейронной производственной сети

Выражение (7.11) полностью описывает процедуру нечеткого вывода исходя из вышеуказанных положений. На рис. 7.4 представлена многослойная структура нечеткой сети, элементы слоев которой реализуют соответствующие компоненты данного выражения.

Структура данной нечеткой сети состоит из четырех слоев.

Слой 1 выполняет фаззификацию входных переменных x'_j ($j = 1, \dots, n$). Элементы этого слоя вычисляют значения функций принадлежности $\mu_{A_{ij}}(x'_j)$, заданных гауссовыми функциями с параметрами a_{ij} и b_{ij} .

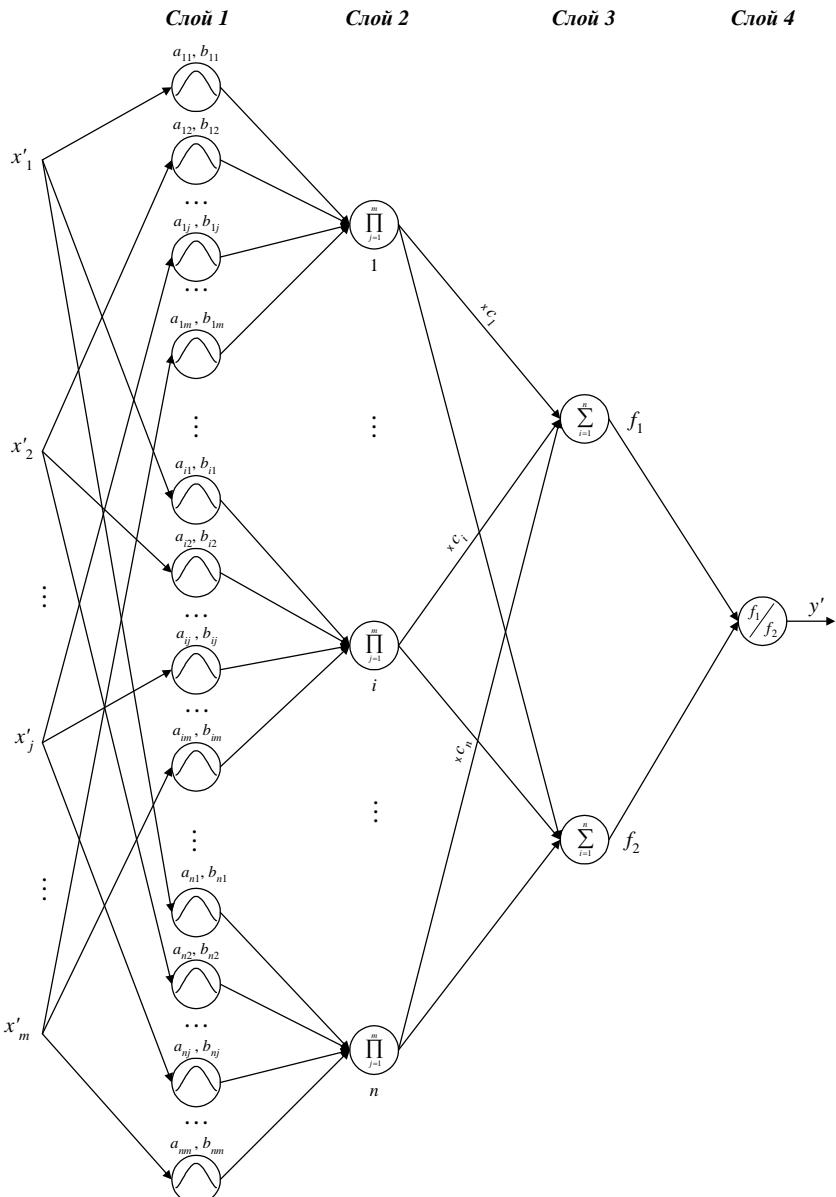


Рис. 7.4. Структура нечеткой нейронной производционной сети Ванга–Менделя

В *слое 2*, число элементов которого равно количеству правил в базе, осуществляется агрегирование степеней истинности предпосылок соответствующих правил.

В *слое 3* первый элемент служит для активизации заключений правил (c_i) в соответствии со значениями агрегированных в предыдущем слое степеней истинности предпосылок правил. Второй элемент слоя проводит вспомогательные вычисления для последующей дефазификации результата.

Слой 4, состоящий из одного элемента, выполняет дефазификацию выходной переменной.

Параметрическими слоями сети являются первый и третий слои, а настраиваемыми параметрами служат соответственно параметры a_{ij} , b_{ij} и c_i .

Структуру рассмотренной выше нечеткой сети можно существенно упростить, если взамен «индивидуальных» нечетких множеств как для каждой входной переменной в каждом правиле, так и для каждой выходной переменной использовать типизированные нечеткие множества [6]. На рис. 7.5 приведен пример модифицированной структуры нечеткой сети.

Замена нескольких близких друг к другу нечетких множеств A_{ij} ($i = 1, \dots, n$), относящихся к одной входной переменной x_j ($j = 1, \dots, m$), заключается в следующем: во-первых, определяется число нечетких множеств для каждой переменной, во-вторых, наиболее близкие (по значениям их центров a_{ij}) между собой нечеткие множества заменяются одним нечетким множеством (с усредненным значением a_{ij}). Общее число этих нечетких множеств будет равно $n^* \times m$, где $n^* < n$.

Аналогично этому может быть выполнено сокращение числа нечетких множеств B_i ($i = 1, \dots, n$) и для выходной переменной у путем их замены на типизированные нечеткие множества, число которых $n^{**} < n$. При этом предпосылки правил с одним и тем же нечетким множеством в заключениях можно объединить оператором нечеткого «ИЛИ», в качестве которого, например, можно использовать операцию максимума. Для чего в структуру модифицированной нечеткой сети вводится дополнительный слой элементов *max* (см. рис. 7.4). Это приводит к существенному упрощению сети, особенно в случае нескольких выходных переменных. Однако следует отметить, что использование недифференцируемой операции максимума в качестве операции нечеткого «ИЛИ» приводит к сложности реализации обучения этой сети на основе алгоритма обратного распространения ошибки, основанного на вычислении производной.

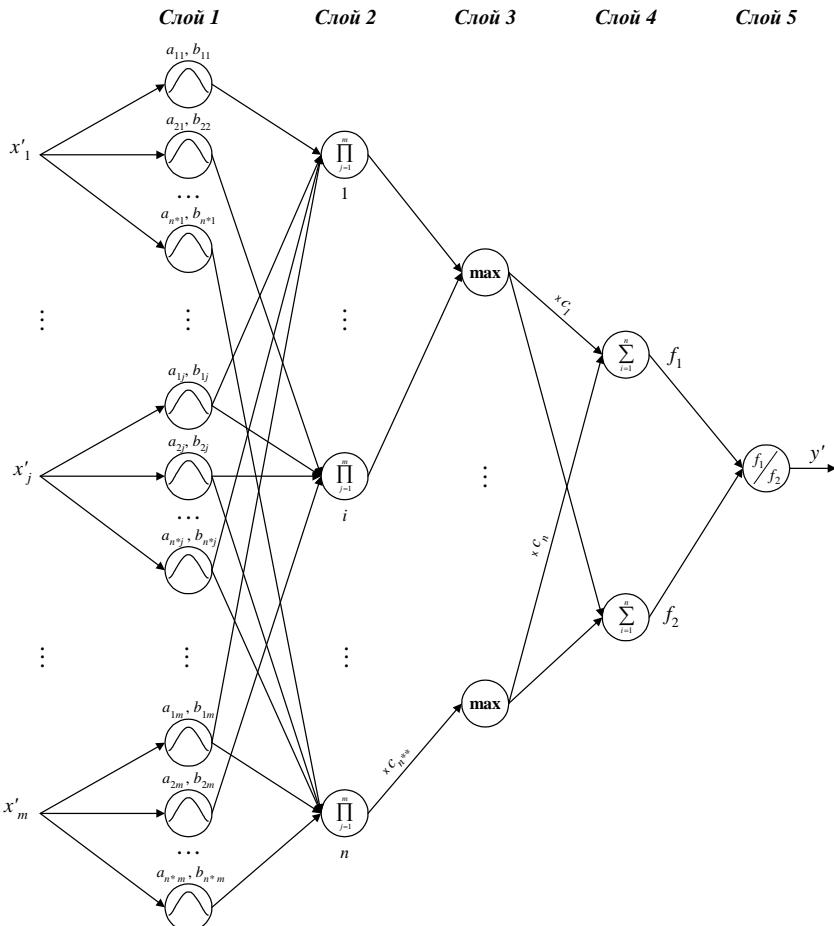


Рис. 7.5. Пример модифицированной структуры нечеткой нейронной производкой сети Ванга–Менделя

Очевидно, что возможны и другие разновидности нечеткой сети данного типа, реализующие, например, алгоритмы Мамдани, Ларсена. Так, для реализации алгоритма нечеткого вывода Мамдани элементы второго слоя вместо операции умножения должны выполнять операцию минимума. Однако так как операция минимума является недифференцируемой, то использование обучения этой сети на основе алгоритма обратного распространения ошибки по вышеуказанной причине также оказывается затруднительным.

Более частое же применение рассмотренных выше вариантов нечеткой сети Ванга–Менделя обусловлено наряду с достаточно простой с вычислительной точки зрения реализацией, также большей чувствительностью к изменениям входных переменных. Поэтому, если нет особых соображений, предпочтение отдается рассмотренным структурам сети.

7.2.3. Обучение нечеткой нейронной производственной сети

Нечеткая нейронная производственная сеть Ванга–Менделя также имеет многослойную структуру с прямым распространением сигнала, значение выхода которой можно изменять, корректируя параметры элементов слоев, что позволяет для обучения этой сети использовать алгоритм обратного распространения ошибки.

В заданной обучающей выборке $(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)}), k = 1, \dots, K$, $x_1^{(k)}, x_2^{(k)}, x_m^{(k)}$ – значения входных переменных x_1, x_2, \dots, x_m , $y^{(k)}$ – эталонное значение выходной переменной y в k -м примере.

Алгоритм обучения разделяется на две процедуры. Сначала настраиваются линейные параметры элементов третьего слоя c_i , а затем – параметры нелинейных функций принадлежности в элементах первого слоя a_{ij} и b_{ij} ($i = 1, \dots, n$; $j = 1, \dots, m$).

Рассмотрим процедуру корректировки значений c_i .

Этап 1. Для каждого примера из обучающей выборки по значениям входных переменных $x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}$ ($k = 1, \dots, K$) рассчитываются значения выходной переменной $y'^{(k)}$.

Этап 2. Вычисляется функция ошибки для всех примеров обучающей выборки:

$$E^{(k)} = \frac{1}{2}(y'^{(k)} - y^{(k)})^2, \quad k = 1, \dots, K.$$

Этап 3. Корректируются значения c_i для каждого i -го правила по каждому k -му примеру обучающей выборки, исходя из соотношения

$$c_i(t+1) := c_i(t) - \eta \frac{\partial E^{(k)}(t)}{\partial c_i(t)}, \quad i = 1, \dots, n, \quad k = 1, \dots, K,$$

где t – номер итерации обучения; $\eta \in [0, 1]$ – коэффициент скорости обучения.

Опуская промежуточные преобразования (см. [6]), получим:

$$c_i(t+1) := c_i(t) - \eta \frac{\left(y'^{(k)} - y^{(k)} \right) \prod_{j=1}^m \exp \left[-\left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]}{\sum_{i=1}^n \prod_{j=1}^m \exp \left[-\left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]} \quad (7.12)$$

Процедура корректировки значений c_i (этапы 1–3) итерационно повторяется и считается завершенной в случае, если:

- либо значение функции ошибки по каждому примеру обучающей выборки не превышает некоторого установленного порога:

$$E^{(k)} < \varepsilon, k = 1, \dots, K;$$

- либо оценка средней суммарной погрешности нечеткой производственной модели с учетом всех примеров обучающей выборки не превышает некоторого установленного порога:

$$E = \frac{1}{K} \sum_{k=1}^K (y'^{(k)} - y^{(k)})^2 < \varepsilon;$$

- либо погрешность стабилизировалась на некотором значении $> \varepsilon$.

В первых двух случаях считается, что нечеткая сеть успешно обучилась, в третьем случае переходят к процедуре настройки параметров нелинейных функций принадлежности a_{ij} и b_{ij} в элементах первого слоя.

При выполнении процедуры корректировки значений a_{ij} и b_{ij} в элементах первого слоя этапы 1 и 2 выполняются аналогично этапам процедуры корректировки значений c_i . На заключительном этапе этой процедуры значения a_{ij} и b_{ij} изменяются в соответствии со следующими выражениями [6]:

$$a_{ij}(t+1) := a_{ij}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial a_i(t)} = a_{ij}(t) - \frac{2(x_j'^{(k)} - a_{ij})(y'^{(k)} - y^{(k)})(c_i - y'^{(k)}) \prod_{j=1}^m \exp \left[-\left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]}{b_{ij}^2 \sum_{i=1}^n \prod_{j=1}^m \exp \left[-\left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]}, \quad (7.13)$$

$$\begin{aligned}
 b_{ij}(t+1) := & b_{ij}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial b_i(t)} = b_{ij}(t) - \\
 & 2 \left(x_j'^{(k)} - a_{ij} \right)^2 \left(y'^{(k)} - y^{(k)} \right) \left(c_i - y'^{(k)} \right) \prod_{j=1}^m \exp \left[- \left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right] \\
 & - \eta \left. \frac{b_{ij}^2 \sum_{i=1}^n \prod_{j=1}^m \exp \left[- \left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]}{b_{ij}^2 \sum_{i=1}^n \prod_{j=1}^m \exp \left[- \left(\frac{x_j'^{(k)} - a_{ij}}{b_{ij}} \right)^2 \right]} \right|_{(t)} \quad (7.14)
 \end{aligned}$$

Условия завершения корректировки значений a_{ij} и b_{ij} подобны c_i . В случае невыполнения первого или второго условия процесс итерационно повторяется начиная с корректировки c_i до тех пор, пока нечеткая сеть не будет корректно обучена.

Существуют различные модификации рассмотренного алгоритма обучения нечеткой сети, одна из которых заключается в том, что параметры c_i , a_{ij} и b_{ij} корректируются в течение одной эпохи обучения, т. е. при однократном предъявлении нечеткой сети всех примеров обучающей выборки.

7.3. Нечеткая нейронная производственная сеть Такаги–Сугэно–Канга

7.3.1. Описание нечеткой нейронной производственной сети

Нечеткая нейронная производственная сеть Такаги–Сугэно–Канга (Takagi–Sugeno–Kang) [7] реализует нечеткую производственную модель, основанную на правилах типа (4.31):

Π_i : ЕСЛИ x_1 есть A_{i1} И ... И x_j есть A_{ij} И ... И x_m есть A_{im} ,

$$\text{ТО } y = c_{i0} + \sum_{j=1}^m c_{ij} x_j, i = 1, \dots, n.$$

Как и в предыдущем случае, данная нечеткая сеть базируется на следующих положениях:

- входные переменные являются четкими;
- функции принадлежности всех нечетких множеств представляются функцией Гаусса;
- декартово произведение нечетких множеств задано выражением (5.6);
- нечеткая импликация – нечеткое произведение;

- Т-норма – нечеткое произведение;
- аккумулирование активизированных заключений правил не проводится;
- метод дефазификации – средний центр (centre average defuzzification).

Исходя из этих положений функциональная зависимость для получения дефазифицированного значения выходной переменной примет следующий вид:

$$y' = \frac{\sum_{i=1}^n \left(\left(c_{i0} + \sum_{j=1}^m c_{ij} x_j \right) \prod_{j=1}^m \mu_{A_{ij}}(x'_j) \right)}{\sum_{i=1}^n \prod_{j=1}^m \mu_{A_{ij}}(x'_j)} = \\ = \frac{\sum_{i=1}^n \left(\left(c_{i0} + \sum_{j=1}^m c_{ij} x_j \right) \prod_{j=1}^m \exp \left[- \left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right] \right)}{\sum_{i=1}^n \prod_{j=1}^m \exp \left[- \left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right]} \quad (7.15)$$

Примечание. Очевидно, что сеть Ванга–Менделя может при этом трактоваться как сеть Такаги–Сугэно–Канга, в которой все параметры c_{ij} (кроме подлежащего уточнению $c_i = c_{i0}$) тождественно равны нулю.

7.3.2. Структура нечеткой нейронной производционной сети

Анализ выражения (7.15), описывающего процедуру нечеткого вывода сетью Такаги–Сугэно–Канга, позволяет отметить ее большое сходство с нечеткой сетью Ванга–Менделя. Компоненты выражений, определяющие предпосылки производционных правил, для обеих сетей идентичны. Различия наблюдаются в представлении заключений. В сети Такаги–Сугэно–Канга результатом является полином $\left(c_{i0} + \sum_{j=1}^m c_{ij} x_j \right)$, тогда как

в сети Ванга–Менделя выходная переменная представляется константой c_i , которую, впрочем, можно рассматривать как полином 0-го порядка. Поэтому сеть Ванга–Менделя можно считать частным случаем нечеткой сети Такаги–Сугэно–Канга.

На рис. 7.6 представлена многослойная структура нечеткой нейронной производственной сети Такаги–Сугэно–Канга.

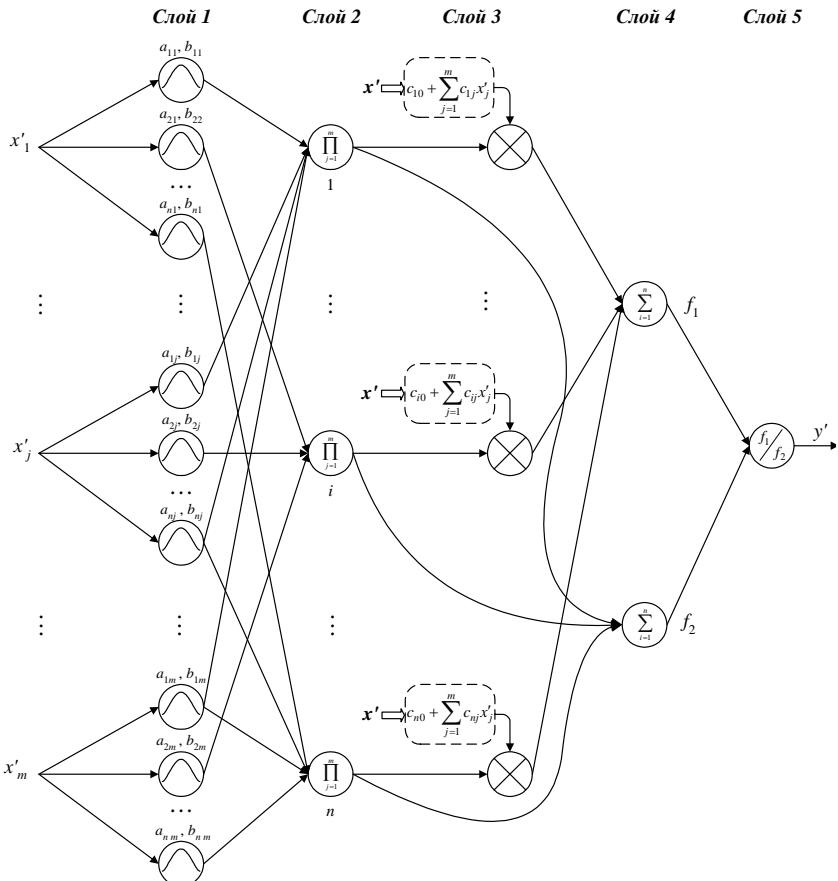


Рис. 7.6. Структура нечеткой нейронной производственной сети Такаги–Сугэно–Канга

Структура нечеткой нейронной производственной сети Такаги–Сугэно–Канга включает в себя пять слоев.

Слой 1 состоит из элементов, которые выполняют фаззификацию входных переменных x'_j ($j = 1, \dots, n$). Элементы этого слоя вычисляют значения функций принадлежности $\mu_{A_{ij}}(x'_j)$, заданных гауссовыми функциями с параметрами a_{ij} и b_{ij} .

Слой 2, число элементов которого равно количеству правил в базе, выполняет агрегирование степеней истинности предпосылок соответствующих правил.

В **слое 3** генерируются значения функций $\left(c_{i0} + \sum_{j=1}^m c_{ij} x'_j \right)$, которые

умножаются на результаты вычислений элементами предыдущего слоя.

В **слое 4** первый элемент (сумматор) служит для активизации заключений правил в соответствии со значениями агрегированных в предыдущем слое степеней истинности предпосылок правил. Второй элемент (сумматор) проводит вспомогательные вычисления для последующей дефазификации результата.

Слой 5, состоящий из одного нормализующего элемента, выполняет дефазификацию выходной переменной.

Из приведенного описания следует, что нечеткая сеть Такаги–Сугэно–Канга содержит два параметрических слоя (Слои 1 и 3). Настраиваемыми в процессе обучения параметрами являются:

- в первом слое – нелинейные параметры a_{ij} , b_{ij} функций принадлежности нечетких множеств предпосылок правил;
- в третьем слое – параметры c_{i0} и c_{ij} линейных функций $\left(c_{i0} + \sum_{j=1}^m c_{ij} x'_j \right)$ из заключений правил.

При наличии n правил и m входных переменных число параметров первого слоя равно $2nm$, а второго – $n(m+1)$. Таким образом, суммарное число настраиваемых параметров равно $n(3m+1)$.

Для уменьшения числа настраиваемых в ходе обучения сети параметров комбинируют функции принадлежности различных входных переменных. Например, если принять, что каждая входная переменная представляется l различными функциями принадлежности, то максимальное число правил, которое можно создать при их комбинировании, составит $n = l^m$. Таким образом, общее число нелинейных параметров в первом слое уменьшится с $2nm$ до $2nm^{1/l}$. Количество параметров в третьем слое при этом остается без изменений.

7.3.3. Обучение нечеткой нейронной производственной сети

Так же как и в алгоритме обучения сети Ванга–Менделя, а алгоритме обучения нечеткой нейронной производственной сети Такаги–Сугэно–Канга подлежащие настройке параметры элементов разделяются на две

группы: параметры c_{i0} и c_{ij} линейных функций, формируемых в третьем слое, а также параметры нелинейных функций принадлежности a_{ij} , b_{ij} , реализуемых элементами первого слоя. Эти параметры настраиваются отдельно друг от друга.

Сначала рассчитываются параметры c_{i0} и c_{ij} линейных функций при фиксированных значениях параметров функций принадлежности a_{ij} , b_{ij} . Они находятся путем решения системы линейных уравнений.

Представим выходную переменную из выражения (7.15) в следующем виде:

$$y' = \sum_{i=1}^n w'_i \left(c_{i0} + \sum_{j=1}^m c_{ij} x_j \right), \quad (7.16)$$

где

$$w'_i = \frac{\prod_{j=1}^m \mu_{A_{ij}}(x'_j)}{\sum_{r=1}^n \prod_{j=1}^m \mu_{A_{rj}}(x'_j)} = \frac{\prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right]}{\sum_{r=1}^n \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{rj}}{b_{rj}} \right)^2 \right]} = \text{const}, \quad i = 1, \dots, n. \quad (7.17)$$

При K обучающих примерах ($x_1^{(k)}$, $x_2^{(k)}$, ..., $x_m^{(k)}$, $y^{(k)}$), где $k = 1, \dots, K$, и замене значений выходных переменных $y'^{(k)}$ эталонными $y^{(k)}$ получим систему из K линейных уравнений вида

$$\begin{bmatrix} w_1'^{(1)} & w_1'^{(1)} x_1^{(1)} & \dots & w_1'^{(1)} x_m^{(1)} & \dots & w_n'^{(1)} & w_n'^{(1)} x_1^{(1)} & \dots & w_n'^{(1)} x_m^{(1)} \\ w_1'^{(2)} & w_1'^{(2)} x_1^{(2)} & \dots & w_1'^{(2)} x_m^{(2)} & \dots & w_n'^{(2)} & w_n'^{(2)} x_1^{(2)} & \dots & w_n'^{(2)} x_m^{(2)} \\ \dots & \dots \\ w_1'^{(K)} & w_1'^{(K)} x_1^{(K)} & \dots & w_1'^{(K)} x_m^{(K)} & \dots & w_m'^{(K)} & w_m'^{(K)} x_1^{(K)} & \dots & w_m'^{(K)} x_m^{(K)} \end{bmatrix} \cdot \begin{bmatrix} c_{10} \\ \dots \\ c_{1m} \\ \dots \\ c_{n0} \\ \dots \\ c_{nm} \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \dots \\ y^{(K)} \end{bmatrix}, \quad (7.18)$$

где $w_i'^{(k)}$ – агрегированная степень истинности предпосылок по i -му правилу при предъявлении k -го входного вектора ($x_1^{(k)}$, $x_2^{(k)}$, ..., $x_m^{(k)}$).

Запишем это выражение в сокращенной матричной форме:

$$\mathbf{W} \cdot \mathbf{c} = \mathbf{y}. \quad (7.19)$$

Размерность матрицы \mathbf{W} равна $K \times (m + 1) \cdot n$, при этом обычно количество строк K значительно больше количества столбцов $(m + 1) \cdot n$. Решение этой системы уравнений можно провести за один шаг при помощи псевдоинверсии матрицы \mathbf{W} :

$$\mathbf{c} = \mathbf{W}^+ \cdot \mathbf{y} = (\mathbf{W}^T \cdot \mathbf{W})^{-1} \cdot \mathbf{W}^T \cdot \mathbf{y}. \quad (7.20)$$

Затем после определения линейных параметров c_{ij} их фиксируют и рассчитывают фактические выходные сигналы сети для всех примеров, для чего используется линейная зависимость

$$\mathbf{y}' = \begin{bmatrix} y'^{(1)} \\ y'^{(2)} \\ \dots \\ y'^{(K)} \end{bmatrix} = \mathbf{W} \cdot \mathbf{c}. \quad (7.21)$$

Вслед за этим определяют вектор ошибки:

$$\mathbf{e} = \mathbf{y}' - \mathbf{y}. \quad (7.22)$$

После чего, например по алгоритму обратного распространения ошибки, уточняются параметры:

$$a_{ij}^{(k)}(t+1) := a_{ij}^{(k)}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial a_{ij}^{(k)}},$$

$$b_{ij}^{(k)}(t+1) := b_{ij}^{(k)}(t) - \eta \frac{\partial E^{(k)}(t)}{\partial b_{ij}^{(k)}},$$

где t – номер очередной итерации.

После уточнения нелинейных параметров процесс адаптации параметров запускается вновь. Этот цикл повторяется вплоть до стабилизации всех параметров процесса. Приведенные выше формулы требуют расчета градиента целевой функции относительно параметров функции принадлежности. Окончательный вид этих формул зависит как от используемого определения функции принадлежности на выходе сети, так и от формы функции принадлежности.

Представленный алгоритм называется *гибридным* и является одним из наиболее эффективных способов обучения нечетких сетей. Его главная черта состоит в разделении процесса обучения на два обособленных во времени этапа. На каждом этапе уточняется только часть параметров сети. Если принять во внимание, что вычислительная сложность каждого

алгоритма пропорциональна (нелинейна) количеству параметров, то уменьшение размерности задачи оптимизации существенным образом сокращает количество математических операций и увеличивает скорость выполнения алгоритма. Благодаря этому гибридный алгоритм значительно более эффективен, чем обычный градиентный алгоритм фронтального типа, согласно которому уточнение всех параметров сети производится параллельно и одновременно.

Примечание. Можно отметить, что рассмотренный гибридный алгоритм обучения применяется к обеим описанным выше нечетким сетям.

Глава 8. НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ С РЕАЛИЗАЦИЕЙ КОМПОНЕНТОВ НЕЧЕТКИХ ПРОДУКЦИОННЫХ МОДЕЛЕЙ НА ОСНОВЕ НЕЙРОСЕТЕВОЙ ТЕХНОЛОГИИ

8.1. Построение функций принадлежности предпосылок и заключений нечетких продукционных правил

Функции принадлежности предпосылок и заключений могут иметь различный вид. Наиболее часто для удобства практической реализации и минимизации вычислительных затрат при осуществлении алгоритмов нечеткого вывода и настройки (обучения) параметров правил используют (*L-R*)-функции: треугольные, трапецидальные, сигмоидальные и гауссовые функции принадлежности.

Как было отмечено выше, реализация различных компонентов нечеткой продукционной модели с использованием нейронных сетей предполагает использование формализмов в виде нейронов и искусственных нейронных сетей, а именно: умножение входных сигналов на весовые коэффициенты, сложение входных сигналов, нелинейное преобразование результатов этого суммирования.

На рис. 8.1 приведен пример нейросетевой реализации гауссовой функции принадлежности [6]:

$$\mu_{A_{ij}}(x_j) = \exp\left[-\left(\frac{x_j - a_{ij}}{b_{ij}}\right)^2\right] \quad (8.1)$$

с использованием двухслойной нейронной сети, два нейрона первого слоя которой имеют сигмоидальные функции активизации:

$$f_{ij}^L(x_j) = \frac{1}{1 + \exp(-b_{ij}^L(x_j + a_{ij}^L))}, \quad (8.2)$$

$$f_{ij}^R(x_j) = -\frac{1}{1 + \exp(-b_{ij}^R(x_j + a_{ij}^R))}, \quad (8.3)$$

а единственный нейрон второго слоя с линейной функцией активизации служит для объединения этих сигмоидальных функций, сдвинутых друг относительно друга и имеющих противоположные знаки:

$$\mu_{A_{ij}}(x_j) = f_{ij}^L(x_j) + f_{ij}^R(x_j). \quad (8.4)$$

Здесь настраиваемыми параметрами являются a_{ij}^L, a_{ij}^R и b_{ij}^L, b_{ij}^R . При этом $a_{ij} = \frac{1}{2}(a_{ij}^L + a_{ij}^R)$, $b_{ij} = b_{ij}^L = b_{ij}^R$.

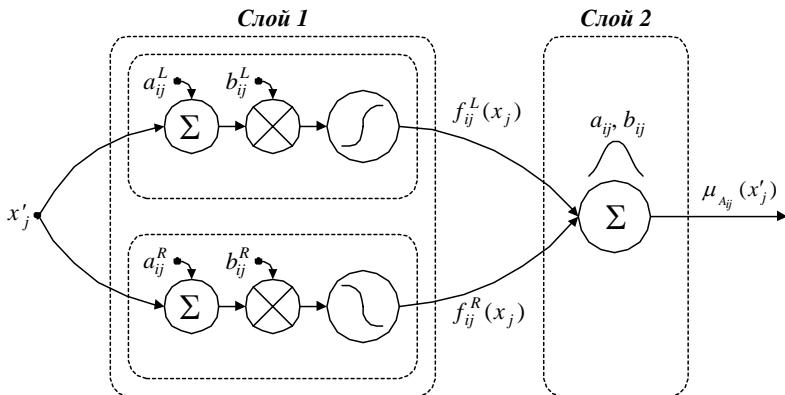


Рис. 8.1. Пример реализации функции принадлежности с использованием нейронной сети

8.2. Формирование предпосылок нечетких производственных правил

Нейронные сети могут использоваться не только для реализации отдельных функций принадлежности, но и для того, чтобы на их основе могли быть осуществлены процедуры нахождения степени истинности

предпосылок по каждому из нечетких производственных правил и определение заключений для выходных переменных.

В различных нечетких производственных моделях формирование предпосылок нечетких производственных правил (т. е. процедуры фазификации и агрегирования степени истинности предпосылок по каждому из нечетких правил) реализуются однотипно. На рис. 8.2 показан фрагмент нечеткой производственной сети, реализующей данную процедуру, пригодную для нечетких производственных моделей различных типов, с использованием нейронных сетей. Отметим, что функции принадлежности предпосылок нечетких правил здесь построены аналогично примеру на рис. 8.1.

Слой 1. Элементы данного слоя вычисляют значения функций принадлежности входных переменных $\mu_{A_j}(x'_j)$ и могут быть реализованы в виде нейронных сетей (см. выше).

Слой 2. Элементы второго слоя данной сети определяют агрегированные степени истинности предпосылок по каждому из нечетких производственных правил, нормированные к общей степени активности всех правил:

$$\hat{\alpha}_i = \frac{1}{\sum_{l=1}^n \alpha_l} \prod_{j=1}^m \mu_{A_{ij}}(x'_j), \quad (8.5)$$

где

$$\alpha_l = \prod_{j=1}^m \mu_{A_{lj}}(x'_j). \quad (8.6)$$

Структура на рис. 8.2 построена исходя из следующих положений.

Все m -мерное пространство независимых входных переменных равномерно разбито: n_j функциями принадлежности для каждой входной переменной x_j ($j = 1, \dots, m$). Очевидно, что число элементов второго слоя определяется числом возможных комбинаций функций принадлежности всех входных переменных и равно $(n_j)^m$ при условии $n_s = n_p$ ($s, p \in \{1, \dots, m\}$) и тем самым обуславливает число возможных правил. На рис. 4.1, а приведен пример разбиения 2d-пространства двух входных переменных x_1, x_2 одномерными функциями принадлежности A_{11}, A_{12}, A_{13} и A_{21}, A_{22}, A_{23} . Вопрос сокращения общего количества функций принадлежности для входных переменных рассмотрен в п. 7.2.2, а вопрос сокращения общего числа правил – в п. 4.2.5. Поэтому обычно окончательное число правил в базе $n < (n_j)^m$.

Первый слой – параметрический. Настраиваемыми являются параметры a_{ij}, b_{ij} . Они задают форму и равномерное разбиение функциями принадлежности пространства входных переменных.

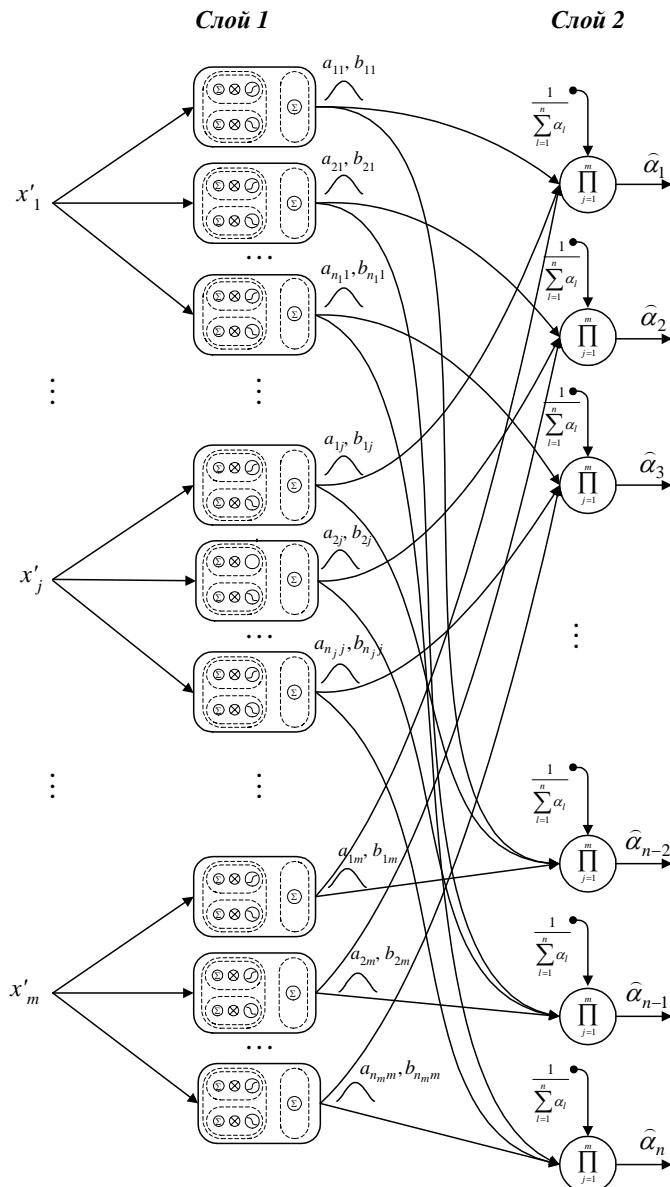


Рис. 8.2. Фрагмент нечеткой нейронной производционной сети для формирования предпосылок правил

8.3. Формирование заключений нечетких производственных правил

В отличие от предпосылок, заключения правил формируются по-разному в зависимости от типа нечеткой производственной модели и алгоритма нечеткого вывода. Поэтому целесообразно рассмотреть различные варианты нейросетевого формирования заключений правил.

Так, для нечеткой производственной модели с базой правил типа

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y_i = c_i, i = 1, \dots, n,$$

с алгоритмом нечеткого вывода Такаги–Сугэно 0-го порядка и с использованием метода дефазификации по среднему центру (centre average defuzzification) значение выходной переменной определяется следующим образом:

$$y' = \sum_{i=1}^n \hat{\alpha}_i c_i, \quad (8.7)$$

где $\hat{\alpha}_i$ – агрегированные степени истинности предпосылок по каждому из нечетких производственных правил, нормированные к общей степени активности всех правил и вычисляемые из выражения (8.5).

На рис. 8.3 показан фрагмент такой нечеткой производственной сети с нейросетевым формированием заключений правил.

Входы элементов второго слоя (см. рис. 8.2) соединяются со входами элементов третьего слоя, которые совместно с единственным элементом четвертого слоя реализуют схему нейрона со взвешенным суммированием сигналов и вычисляют выражение (8.7).

Рассмотрим вариант нейросетевого формирования заключений правил для нечеткой производственной сети, реализующей аффинную модель Такаги–Сугэно:

$$\Pi_i: \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{ij} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ \text{ТО } y = c_{i0} + c_{i1} x_1 + \dots + c_{ij} x_j + \dots + c_{im} x_m, i = 1, \dots, n,$$

и с использованием метода дефазификации по среднему центру.

В этом случае дефазифицированное значение выходной переменной примет вид

$$y' = \sum_{i=1}^n \hat{\alpha}_i (c_{i0} + c_{i1} x_1 + \dots + c_{ij} x_j + \dots + c_{im} x_m). \quad (8.8)$$

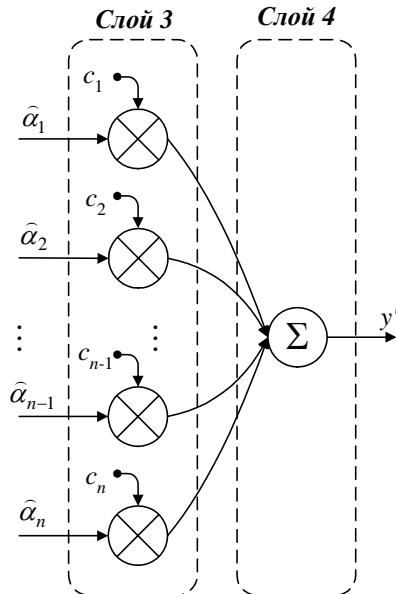


Рис. 8.3. Фрагмент нечеткой производственной сети Такаги–Сугэно 0-го порядка с нейросетевым формированием заключений правил

На рис. 8.4 показан фрагмент нечеткой производственной сети с нейросетевым формированием заключений правил в соответствии с выражением (8.8).

Еще одним условием успешной работы нечеткой производственной сети с рассмотренной нейросетевой реализацией заключений правил является предварительная нормализация значений входных переменных в соответствии с выражением

$$\hat{x}_j = \frac{x_j}{\max_{l=1,\dots,m} |x_l|}, \quad j = 1, \dots, m. \quad (8.9)$$

благодаря чему на входах элементов третьего слоя формируются нормализованные значения входных переменных из диапазона $[-1, 1]$.

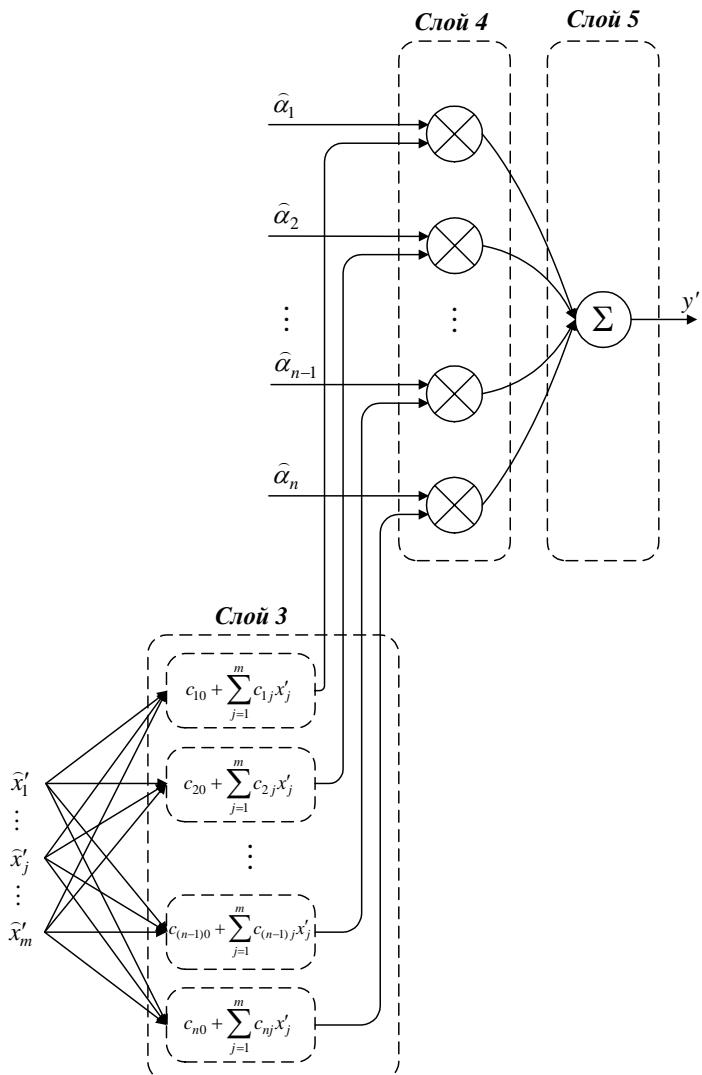


Рис. 8.4. Фрагмент нечеткой аффинной производственной сети Такаги–Сугэно с нейросетевым формированием заключений правил

8.4. Разбиение пространств входных переменных и формирование многомерных функций принадлежности предпосылок

Серьезной проблемой при задании базы нечетких производственных правил является корректное разбиение пространства входных переменных в случаях, если эти переменные являются взаимозависимыми (см. рис. 4.1, в). Как было сказано в п. 4.2.4, решением этой проблемы является представление всех входных переменных в виде вектора, а предпосылок – на основе многомерных функций принадлежности:

$$\Pi_i: \text{ЕСЛИ } \mathbf{x} \text{ есть } A_i, \\ \text{ТО } \dots, i = 1, \dots, n, \quad (8.10)$$

где $\mathbf{x} = [x_1, \dots, x_m]^T$; A_i – нечеткое множество с многомерной функцией принадлежности $\mu_{A_i}(\mathbf{x})$ [8].

Данная задача возникает либо при условии априорного задания экспертом числа классов в пространстве входных переменных и наличия самих экспериментальных данных (обучающей выборки), либо при наличии только экспериментальных данных и заключается в первом случае в разделении входных данных на известное число классов n , а во втором – в их кластеризации. Число этих классов определяет число многомерных функций принадлежности предпосылок и в конечном итоге число правил.

Для решения данной задачи могут быть использованы искусственные нейронные сети: многослойный персепtron – для первого случая, саморганизующаяся сеть Кохонена – для второго случая. Правила построения и настройки нейронных сетей этих типов хорошо известны, и за дополнительной информацией по этим вопросам можно обратиться, например, к [2].

Рассмотрим пример построения нечеткой нейронной производственной сети при условии изначального выделения в двумерном пространстве входных переменных x_1, x_2 четырех классов (т. е. двумерных функций принадлежности) A_1, \dots, A_4 .

На рис. 8.5 приведен пример структуры модифицированной нечеткой нейронной производственной сети Такаги–Сугэно–Канга, иллюстрирующий реализацию разбиения пространств входных переменных и формирование многомерных функций принадлежности предпосылок с использованием нейронной сети (многослойного персептрана).

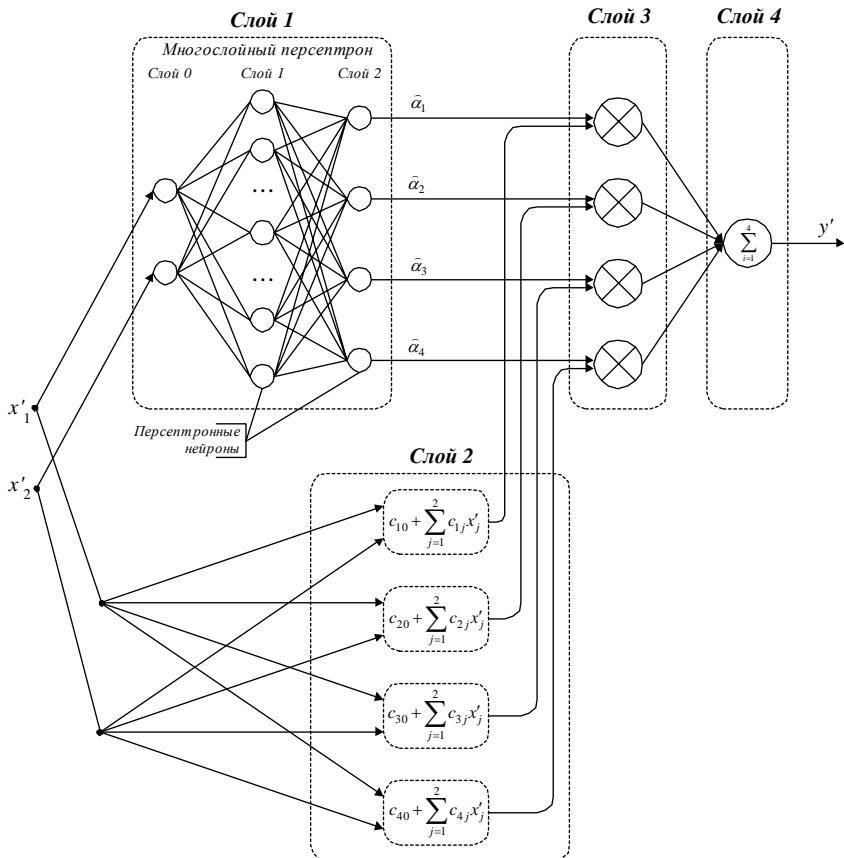


Рис. 8.5. Пример структуры нечеткой производационной сети

Такаги–Сугэно–Канга с использованием нейронной сети для разбиения пространств входных переменных и формирования многомерных функций принадлежности предпосылок

Следует отметить, что данный подход к использованию нейронной сети может быть распространен и на нечеткие производственные сети других типов.

Перед началом работы сеть нейронная сеть (слой 1) должна быть обучена правильной классификации. И фактически ее целью в процессе функционирования всей нечеткой производационной сети является фазификация входных переменных и определение степеней истинности предпосылок по каждому из нечетких производственных правил $\mu_{A_i}(x')$. При

этом исходят из того, что выходы нейронной сети являются нормированными, т. е.

$$\hat{\alpha}_i = \frac{\mu_{A_i}(\mathbf{x}')}{\sum_{l=1}^n \mu_{A_l}(\mathbf{x}')} , \quad (8.11)$$

где

$$\sum_{l=1}^n \mu_{A_l}(\mathbf{x}') = 1 . \quad (8.12)$$

Элементы слоя 2 данной нечеткой производственной сети, реализующие заключения нечетких производственных правил, могут быть построены на основе нейронов со взвешенным суммированием сигналов и линейной функцией активации. Аналогично этому элементы слоев 3 и 4 реализуют схему нейрона со взвешенным суммированием сигналов и вычисляют выражение (8.13), соответствующее алгоритму нечеткого вывода Такаги–Сугэно и методу дефазификации по среднему центру:

$$y' = \sum_{i=1}^n \hat{\alpha}_i (b_{i0} + b_{i1}x_1 + \dots + b_{ij}x_j + \dots + b_{im}x_m) . \quad (8.13)$$

Процесс обучения этой нечеткой нейронной производственной сети разделяется на два этапа.

Этап 1. Обучение нейронной сети, реализующей первый слой, разделению пространства входных переменных по алгоритму обратного распространения ошибок. В данном случае заранее определяется число классов (многомерных функций принадлежности), покрывающих пространство входных переменных, а также подготавливается обучающая выборка, презентативно представляющая примеры для каждого класса и обеспечивающая возможность корректного разбиения этого пространства.

Этап 2. Результаты обучения, полученные на первом этапе, используются для последующего обучения элементов второго слоя (настройки линейных параметров c_{i0} и c_{ij}), реализующих заключения правил. При этом обучение проводится при условии объединения всех слоев нечеткой производственной сети в единую систему с использованием той же самой обучающей выборки. Обучение также проводится по алгоритму обратного распространения ошибки. Особенностью данного этапа обучения является то, что после активизации одного из правил (одного из выходов $\hat{\alpha}_i$ элементов слоя 1) обучению подвергаются параметры только соответствующего элемента слоя 2.

8.5. Нечеткие производственные сети с представлением структуры в виде нейронных сетей

Представление нечеткой производственной модели в виде нейронной сети, как правило, предполагает выполнение следующих положений:

- использование искусственных нейронов, состоящих из элементов трех типов: умножителей, сумматора и нелинейного преобразователя;
- формирование многослойной сетевой структуры с прямым распространением сигнала (feedforward);
- обучение на основе алгоритмов, подобных алгоритмам обучения обычных нейронных сетей.

Рассмотрим пример представления нечеткой производственной модели Мамдани в виде многослойной нейронной сети. Будем исходить из того, что:

- функции принадлежности всех нечетких множеств являются гауссовыми;
- входные переменные – четкие;
- декартово произведение нечетких множеств задано выражением (5.6);
- нечеткая импликация – нечеткое произведение;
- Т-норма – нечеткое произведение;
- аккумулирование активизированных заключений правил не проводится;
- метод дефазификации – средний центр (centre average defuzzification) [см. выражение (4.60)].

Тогда дефазифицированное значение выходной переменной определяется по следующей формуле (см. п. 7.2.1):

$$y' = \frac{\sum_{i=1}^n \left(\arg \max_y \mu_{B_i}(y) \prod_{j=1}^m \mu_{A_{ij}}(x'_j) \right)}{\sum_{i=1}^n \prod_{j=1}^m \mu_{A_{ij}}(x'_j)} = \\ = \frac{\sum_{i=1}^n \left(\arg \max_y \left(\exp \left[-\left(\frac{y - c_i}{d_i} \right)^2 \right] \right) \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right] \right)}{\sum_{i=1}^n \exp \left[-\left(\frac{x'_j - a_{ij}}{b_{ij}} \right)^2 \right]}.$$

Считая для простоты, что ширина всех функций Гаусса для входных переменных одинакова, т. е. $b_{ij} = b_i$, преобразуем

$$\begin{aligned}
 & \prod_{j=1}^m \exp \left[-\left(\frac{x'_j - a_{ij}}{b_i} \right)^2 \right] = \\
 & = \exp \left[\sum_{j=1}^m -\left(\frac{x'_j - a_{ij}}{b_i} \right)^2 \right] = \exp \left[-\frac{\sum_{j=1}^m ((x'_j)^2 + 2x'_j a_{ij} + a_{ij}^2)}{b_i^2} \right] = \quad (8.14) \\
 & = \exp \left[-\frac{\sum_{j=1}^m (x'_j)^2 - \sum_{j=1}^m 2x'_j a_{ij} + \sum_{j=1}^m a_{ij}^2}{b_i^2} \right].
 \end{aligned}$$

Для упрощения последнего выражения выполним нормализацию векторов $\mathbf{x}' = (x'_1, \dots, x'_m)^T$ и $\mathbf{a}_i = (a_{i1}, \dots, a_{im})^T$, т. е. допустим, что

$$\sqrt{\sum_{j=1}^m (x'_j)^2} = 1, \quad \sqrt{\sum_{j=1}^m a_{ij}^2} = 1. \quad (8.15)$$

Тогда итоговое выражение для дефазифицированного значения выходной переменной примет следующий вид:

$$y' = \frac{\sum_{i=1}^n \left(\arg \max_y \left(\exp \left[-\left(\frac{y - c_i}{d_i} \right)^2 \right] \right) \exp \left[\frac{2}{b_i^2} \left(\sum_{j=1}^m x'_j a_{ij} - 1 \right) \right] \right)}{\sum_{i=1}^n \exp \left[\frac{2}{b_i^2} \left(\sum_{j=1}^m x'_j a_{ij} - 1 \right) \right]}. \quad (8.16)$$

На рис. 8.6 представлена структура нейронной сети, реализующая выражение (8.16) и удовлетворяющая указанным выше требованиям.

Можно отметить структурное и функциональное сходство данной сети с вероятностными нейронными сетями [9]. К достоинствам же данной нейронной сети относится возможность интерпретации ее параметров с точки зрения реализуемой нечеткой производционной модели.

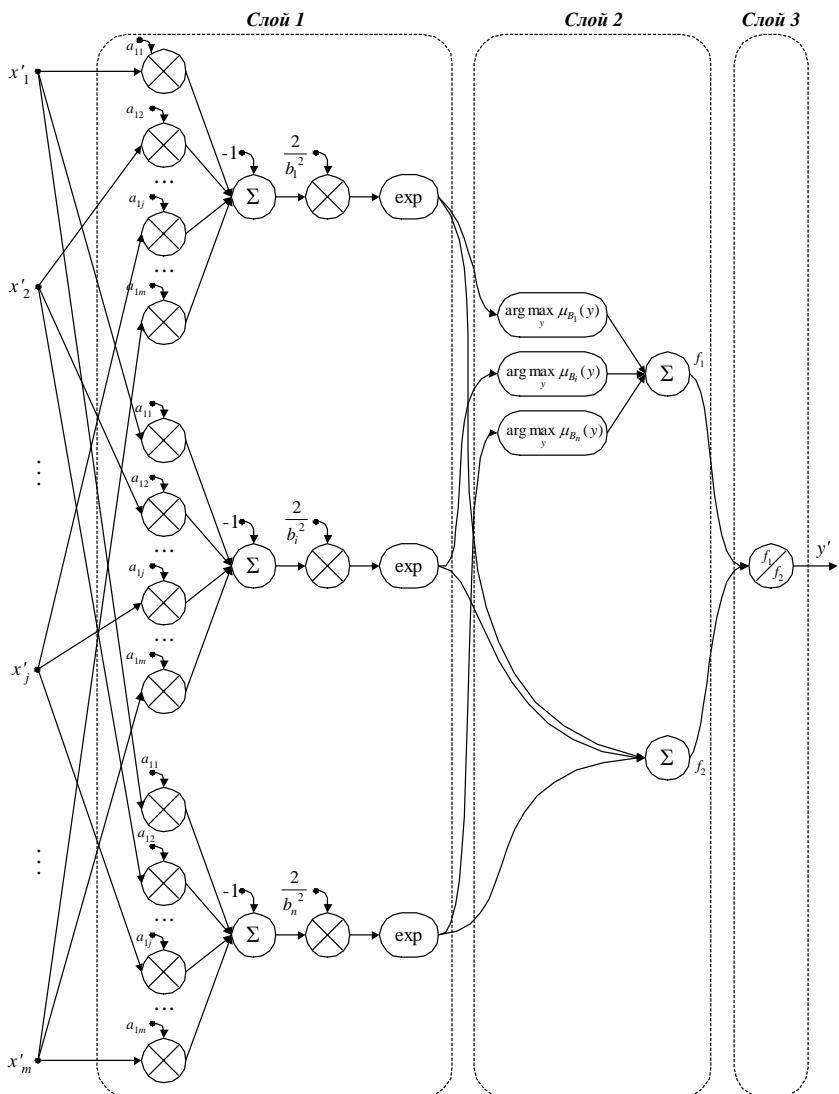


Рис. 8.6. Пример представления нечеткой производственной сети в виде нейронной сети

ВЫВОДЫ

Предложена следующая классификация способов и средств интеграции нечетких продукционных моделей с нейронными сетями:

- нечеткие продукционные сети, обладающие возможностью параметрической оптимизации нечетких правил (параметров функций принадлежности, активационных функций, весов связей) на основе алгоритмов обучения (подобных алгоритмам обучения нейронных сетей, например алгоритму обратного распространения ошибки back propagation) с использованием экспериментальных данных;
- формирование функций принадлежности предпосылок и заключений нечетких продукционных правил с использованием нейронных сетей;
- нечеткие продукционные сети с использованием нейронных сетей для формирования предпосылок нечетких продукционных правил;
- нечеткие продукционные сети с использованием нейронных сетей для формирования заключений нечетких продукционных правил;
- нечеткие продукционные сети с использованием нечетких нейронных сетей для дефазификации выходной переменной;
- нечеткие продукционные сети с использованием нейронных сетей для разбиения пространств входных переменных и формирования для них многомерных функций принадлежности предпосылок;
- нечеткие продукционные сети с комбинированием указанных выше реализаций компонентов нечетких продукционных моделей на основе нейронных сетей;
- нечеткие продукционные сети с адаптацией структуры под особенности нейронных сетей, построенных на основе традиционных нейронов со взвешенным суммированием входных сигналов и нелинейной функцией активации.

Рассмотрены основные типы нечетких нейронных продукционных сетей с параметрической оптимизацией правил на основе алгоритмов обучения: ANFIS (Adaptive Neuro-Fuzzy Inference System – адаптивная нейронечеткая система), Ванга–Менделя, Такаги–Сугэно–Канга. Описаны их структуры, свойства, алгоритмы обучения.

Представлены нечеткие нейронные продукционные сети с реализацией различных компонентов нечетких продукционных моделей на основе нейросетевой технологии.

СПИСОК ЛИТЕРАТУРЫ

1. *Jang J.-S. R.* ANFIS: Adaptive Network-based Fuzzy Inference System // IEEE Trans. on Syst., Man and Cybernetics. 1993. V. 23. № 3. P. 665–685.
2. *Круглов В. В., Борисов В. В.* Искусственные нейронные сети. Теория и практика. М.: Горячая линия–Телеком, 2001.
3. *Jang J.-S. R.* ANFIS: Adaptive-network-based fuzzy inference system// IEEE Trans. on Syst., Man and Cybernetics. 1993. V. 23. P. 665–685.
4. *Wang L.-X., Mendel J. M.* Generating fuzzy rules by learning from examples. IEEE Trans. on Syst., Man and Cybernetics. 1992. V. 22. № 6. P. 1414–1427.
5. *Wang L.-X.* Adaptive fuzzy systems and control. New Jersey. PTR Prentice Hall, Englewood Cliffs, 1994.
6. *Рутковская Д., Пилинковский М., Рутковский Л.* Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия – Телеком, 2004.
7. *Takagi T., Sugeno M.* Fuzzy identification of systems and its application to modeling and control. IEEE Trans. Systems, Man and Cybernetics. 1985. V. 15. №. 1. P. 116–132.
8. *Babuška R.* Fuzzy Modeling for Control. Boston, USA: Kluwer Academic Publishers, 1998.
9. *Rutkowski L, Galkowski T.* On pattern classification and system identification by probabilistic neural networks // Appl. Math. and Comp. Sci. 1994. V. 4. № 3. P. 413–422.

Часть IV

НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ

Нейронные нечеткие сети характеризуются введением нечеткости в различные компоненты традиционных нейронных сетей на основе положений теории нечетких множеств и нечеткой логики.

Предложим и проиллюстрируем в последующих главах такую классификацию основных способов введения нечеткости в компоненты нейронных сетей:

1. Введение нечеткости в структуры нейронных сетей путем приписывания нечеткости значениям обучающих примеров, «размыванием» входных данных и получением выходной информации в терминах теории нечетких множеств [1–3].
2. Наделение нейронов, из которых состоят нейронные сети, нечеткостью. Так, входы, выходы, а также веса таких нейронов могут быть нечеткими. Кроме того, активизация нейронных сетей, состоящих из этих нейронов, является нечетким процессом [4, 5].
3. Изменение базисных характеристик нейронов таким образом, что они выполняют операции, используемые в теории нечетких множеств (подобно нечеткому объединению, пересечению, агрегированию), вместо стандартного умножения и сложения [6–8].
4. Моделирование либо ошибки, либо нестабильности, либо функции энергии нейронной сети с использованием нечетких/неопределенных значений [9]. Так, например, в работе [10] рассматривается фазификация дельта-правила для обучения многослойного персептрона за счет использования нечетких чисел в качестве входов, выходов и/или значений весов. Однако в этом случае возникают проблемы с созданием правила остановки процесса обучения, которые решаются заданием весов в виде треугольных или трапецидальных нечетких чисел (что приводит, однако, к усложнению алгоритма обучения) [11].
5. Использование нечетких производственных сетей в нейронных сетях [12].

Глава 9. НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ С ВВЕДЕНИЕМ НЕЧЕТКОСТИ В СТРУКТУРУ

Впервые особенности нейронных нечетких сетей данного типа были описаны в работе [1]. Келлер (Keller) и Хант (Hunt) предложили концепцию распознавания нечетких образов с использованием персептрона. Возможны различные варианты такой сети, когда модифицируется либо стратегия самого обучения, либо отдельные его параметры, заметно влияющие на качество обучения.

Рассмотрим случай линейной неразделимости классов объектов (рис. 9.1), т. е. когда отдельные классы перекрываются, а образцы (примеры обучающей выборки), лежащие в зоне перекрытия, приводят к неустойчивому поведению алгоритма обучения классического персептрана.

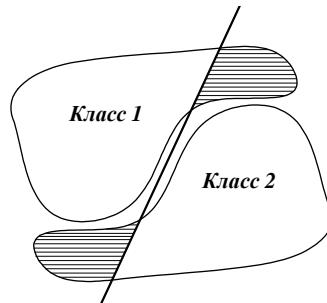


Рис. 9.1. Пример линейной неразделимости двух классов

Основная идея построения нечеткого персептрана и алгоритма его обучения заключается в уменьшении влияния векторов, лежащих в зоне перекрытия, на изменение весовых коэффициентов.

Пусть имеется обучающая выборка, состоящая из множества примеров следующего вида:

$$(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}; \mu_1(x^{(k)}), \mu_2(x^{(k)})), k = 1, \dots, K, \quad (9.1)$$

где $x^{(k)} = \{x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}\}$ – значения вектора входных переменных x_1, x_2, \dots, x_m в k -м примере; $\mu_1(x^{(k)}), \mu_2(x^{(k)})$ – соответствующие функции принадлежности, определяемые как меры нечеткого разбиения пространства входов на классы (в данном примере на два класса) по отношению к этим векторам и показывающие степени принадлежности каждого из этих векторов к каждому из классов; K – общее число примеров в обучающей выборке. При этом должны выполняться следующие соотношения:

$$\sum_{l=1}^2 \mu_l(\mathbf{x}^{(k)}) = 1, \quad 0 < \sum_{k=1}^K \mu_l(\mathbf{x}^{(k)}) < K, \quad \mu_l(\mathbf{x}^{(k)}) \in [0, 1], \quad (9.2)$$

причем для k -го входного вектора, принадлежность к l -му классу которого является менее определенной, соответствующая функция принадлежности $\mu_l(\mathbf{x}^{(k)})$ находится в диапазоне от 0 до 0.5 (рис. 9.2) [13].

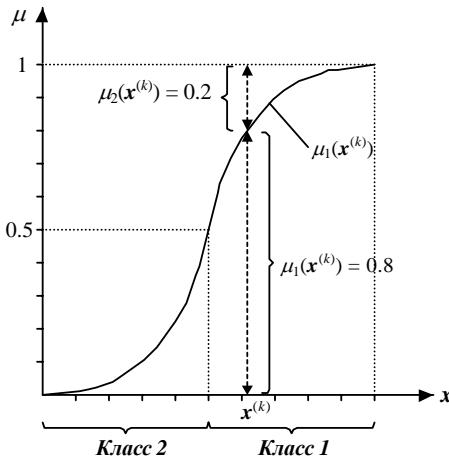


Рис. 9.2. Пример представления функций принадлежности двух пересекающихся классов

Степень влияния входного вектора $\mathbf{x}^{(k)}$ на модификацию весовых коэффициентов персептрона в процессе обучения можно определить как разность $|\mu_1(\mathbf{x}^{(k)}) - \mu_2(\mathbf{x}^{(k)})|^g$, где g – константа, которая в случае неопределенности отнесения к одному из двух классов вектора $\mathbf{x}^{(k)}$ близка к нулю. Учитывая это, можно модифицировать формулу, определяющую коррекцию весовых коэффициентов в алгоритме обратного распространения ошибки для классического персептрона, следующим образом:

$$w_j(t) := w_j(t-1) + h |\mu_1(\mathbf{x}^{(k)}(t-1)) - \mu_2(\mathbf{x}^{(k)}(t-1))|^g \cdot [d^{(k)} - y^{(k)}(t-1)] x_j^{(k)}, \quad 1 \leq j \leq m, \quad (9.3)$$

где $d^{(k)}$, $y^{(k)}(t-1)$ – значения желаемого и полученного в предыдущем такте обучения выходных сигналов при $\mathbf{x}^{(k)}$ входном векторе; t – текущий такт обучения нечеткого персептрона; w_j – весовой коэффициент j -го входа персептронного нейрона; h – скорость обучения.

Из выражения (9.2) следует, что влияние неопределенных векторов на модификацию весовых коэффициентов можно уменьшить за счет изменения константы g . Подобно классическому персептрону, в нечетком персептроне можно найти разделяющую поверхность за конечное число итераций. При этом если выполняется соотношение

$$\left| \mu_1(x^{(k)}(t)) - \mu_2(x^{(k)}(t)) \right|^g = 1, \quad (9.4)$$

то нечеткий персептрон становится четким, т. е. значения функций принадлежности также будут четкими.

На ранних итерациях обучения для входных векторов, значения функций принадлежности которых близки к 0.5, можно установить значение $g > 1$ и на последующих итерациях постепенно его уменьшать.

Для разработки алгоритма обучения нечеткого персептрана предварительно необходимо, во-первых, найти способ определения степеней принадлежности входных векторов к различным классам, во-вторых, определить критерий остановки процесса обучения.

Для определения степеней принадлежности входных векторов к различным классам в соответствии с рассмотренным выше примером можно использовать следующие выражения:

$$\begin{aligned} \mu_1(x^{(k)}) &= 0.5 + \frac{\exp(f(\rho_2 - \rho_1)/\rho_{med}) - \exp(-f)}{2(\exp(f) - \exp(-f))}, \\ \mu_2(x^{(k)}) &= 1 - \mu_1(x^{(k)}), \end{aligned} \quad (9.5)$$

где f – константа, подбираемая опытным путем ($f > 0$); ρ_2 – расстояние от вектора $x^{(k)}$ до центра класса 1; ρ_1 – расстояние от вектора $x^{(k)}$ до центра класса 1 в соответствии с выбранной метрикой (например, евклидовой); ρ_{med} – расстояние между центрами классов 1 и 2.

При выборе критерия остановки процесса обучения нечеткого персептрана необходимо учесть правило, в соответствии с которым входной вектор не может быть однозначно отнесен с определенному классу, если значение его функции принадлежности близко к 0.5. Поэтому для остановки процедуры обучения дополнительно выбирается показатель δ , заданное значение которого показывает близость функции принадлежности к значению 0.5. А именно

$$\mu_1(x^{(k)}) > 0.5 + \delta \text{ или } \mu_1(x^{(k)}) < 0.5 + \delta, \text{ для } \forall k, k = 1, \dots, K. \quad (9.6)$$

Алгоритм обучения нечеткого персептрана состоит из следующих этапов

Этап 1. Начальная инициализация весовых коэффициентов. Установление допустимых значений ошибки обучения, g и δ .

Этап 2. Вычисление степеней функций принадлежности для всех входных векторов обучающей выборки в каждом классе в соответствии с выражениями (9.5).

Этап 3. Предъявление входных векторов $\mathbf{x}^{(k)} = \{x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}\}$, $k = 1, \dots, K$, и вычисление текущих значений выхода:

$$y(t) = \text{sign}\left(\sum_{j=1}^m w_j x_j^{(k)}\right). \quad (9.7)$$

Этап 4. Сравнение текущих и желаемых значений выхода.

Этап 5. Модификация весовых коэффициентов в соответствии с выражением (9.3), если $d^{(k)} = y^{(k)}(t)$.

Этап 6. Просмотр всех примеров из обучающей выборки, проверка критерия ошибки обучения (см. алгоритм обучения классического персептрона) и условия (9.6). Если эти условия не выполняются хотя бы для одного входного вектора, то переход к этапу 3.

Следует отметить, что в процессе итерационного обучения постепенно уменьшают значения h и g .

В работах Хаяши (Hayashi) [14, 15] представлено обобщение рассмотренной выше нейронной сети. Все противоречивые примеры исключаются из обучающей выборки и, таким образом, не могут влиять на процесс обучения. Входной слой состоит из совокупностей нечетких и четких элементов, в то время как выходной слой содержит только совокупности нечетких элементов. Совокупности четких элементов представляются посредством m элементов, принимающих два значения: -1 и 1 . Совокупности нечетких ячеек, с другой стороны, формируют m -мерные векторы, каждый из которых принимает значения из интервала $[-1, 1]$. В каждом высказывании учитываются лингвистические значения, характеризующие их относительную важность, например *очень важный* и *умеренно важный*. Помимо этого, используются лингвистические значения истинности выходных данных, например *абсолютно истинный*, *истинный*, *возможно истинный*, *неизвестный*, *возможно ложный*, *ложный* и *абсолютно ложный*, определенные группой экспертов. При этом допускается применение различных лингвистических значений истинности для моделирования принадлежности отдельного образа к более чем одному классу. Выделение нечетких производных правил возможно

с использованием обхода сети путем анализа активизированных элементов и ассоциированных с ними весов соединений.

Позднее идею введения нечеткости в структуры нейронных сетей развили Пал (Pal) и Митра (Mitra) при создании нечеткой версии многослойного персептрона, используемого для классификации образов [16], обеспечив возможность обработки неопределенной и неточной информации.

На рис. 9.3 приведен пример реализации нечеткого многослойного персептрона. Он состоит из двух частей: нейронов «нечеткого слоя» и собственно многослойного персептрона. Функции активации нейронов «нечеткого слоя» являются, как правило, радиальными базисными функциями (обычно в виде функции Гаусса), моделирующими функции принадлежности. Эти нейроны предназначены для определения степеней принадлежности компонентов входных векторов (которые могут быть и нечеткими). В качестве нейронов данного слоя могут выступать и нечеткие нейроны, выполняющие операции, используемые в теории нечетких множеств (см. гл. 11). На выходах нейронов этого слоя формируются коэффициенты в требуемой для дальнейшего распознавания форме. Выходы нейронов «нечеткого слоя» употребляются в качестве входов традиционного многослойного персептрона.

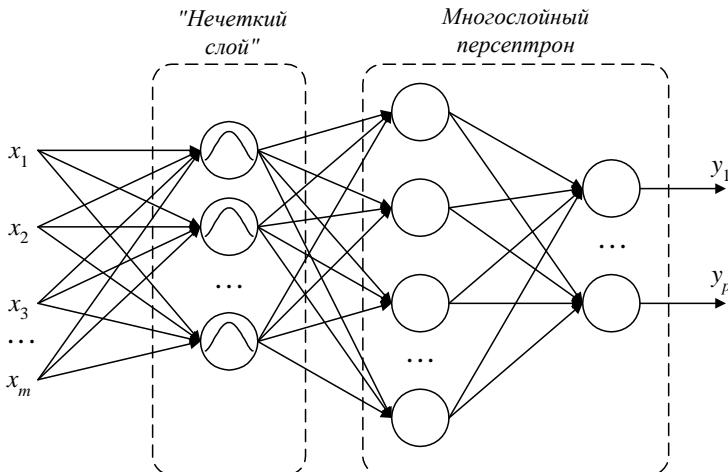


Рис. 9.3. Структура нечеткого многослойного персептрона

Если на вход сети подается вектор $\mathbf{x} = [x_1, x_2, \dots, x_m]^T$, то на выходе «нечеткого слоя» формируется вектор, состоящий из степеней принадлежности \mathbf{x} к конкретным центрам (радиальным базисным функциям):

$\mu(\mathbf{x}) = [\mu_1(\mathbf{x}), \mu_2(\mathbf{x}), \dots, \mu_L(\mathbf{x})]^T$. Конкретные компоненты u_{ij} рассчитываются таким образом, чтобы удовлетворять условию нормализации $\sum_{l=1}^L \mu_l(\mathbf{x}^{(k)}) = 1$ для каждого вектора $\mathbf{x}^{(k)}$, $k = 1, \dots, K$, где K – число векторов в обучающей выборке.

Количество нейронов «нечеткого слоя» определяется числом кластеров. Количество скрытых слоев многослойного персептрона и число нейронов в этих слоях может быть произвольным, хотя обычно для восстановления данных с требуемой точностью достаточно одного скрытого слоя. Размерность выходного слоя многослойного персептрона p зависит от размерности выходного вектора y .

На практике подобная нейронная нечеткая сеть, как правило, более эффективна, чем просто нечеткая сеть или чем обычный многослойный персептрон. Этот вывод следует из факта, что при использовании нейронной нечеткой сети данного типа задача ее обучения разделяется на два этапа, реализуемые независимо друг от друга. Сначала выполняется этап самоорганизации «нечеткого слоя», заключающийся в разделении пространства входных переменных на кластеры, при этом количество кластеров (самоорганизующихся нейронов «нечеткого слоя») может быть произвольным и определяться условиями решаемой задачи. При этом многослойный персептрон приписывает каждой группе кластеров (нейронов «нечеткого слоя») соответствующий ей ожидаемый результат. Например, при решении задачи классификации это может выглядеть как отнесение к одному конкретному классу нескольких кластеров данных. Самоорганизация нейронов «нечеткого слоя» состоит в подборе позиций центров кластеров, представляющих входные данные. Для этого можно применять как алгоритм C-means, так и алгоритм Густафсона–Кесселя. По завершении первого этапа, после стабилизации значений степеней принадлежности (коэффициентов) всех векторов, представляющих входные сигналы для многослойного персептрона, начинается второй этап обучения. На нем значения параметров самоорганизующейся части сети остаются неизменными, а уточняются только веса нейронов многослойного персептрона. Это обычное обучение многослойного персептрона, для которого входом является множество степеней принадлежности (коэффициентов) вектора \mathbf{x} к центрам кластеров самоорганизующегося «нечеткого слоя». В зависимости от типа решаемой задачи выходом сети может быть код класса, к которому принадлежит входной вектор \mathbf{x} , либо ожидаемое значение d выходного вектора, соответствующего вектору \mathbf{x} . По завершении второго этапа обучения веса сеть становится готовой к функционированию в режиме эксплуатации, в котором на нее подаются только входные векторы \mathbf{x} без соответствующих им векторов d .

Существуют и другие варианты введения нечеткости в структуру многослойного персептрона. Например, в [17] для представления нечетких входов и выходов используется интервальное представление компонентов векторов. В алгоритме обучения с обратным распространением ошибки значение оценочной функции определяется посредством α -уровней нечетких множеств текущего и требуемого значений нечеткого выхода с использованием правил интервальной арифметики.

В настоящее время практически для каждой парадигмы нейронной сети предложены различные варианты введения нечеткости. К наиболее известным можно отнести нечеткую самоорганизующуюся сеть Кохонена [3]. В ней, как и в нечетком многослойном персептроне, помимо того что входная информация может быть качественной, лингвистической или в виде нечетких множеств, или в виде комбинации, она также может быть неполной. Например, компоненты входного вектора могут содержать степени принадлежности к нечетким множествам *низкий*, *средний*, *высокий* (для каждого входного параметра). Это обеспечивает возможность использования информации, представленной в лингвистической форме, на фазах обучения и тестирования этих нечетких нейронных сетей и повышает их устойчивость к неточной и неопределенной входной информации. Нечеткая нейронная сеть Кохонена обеспечивает большую по сравнению с традиционной сетью Кохонена эффективность разделения на различные классы множеств сходных входных векторов.

Глава 10. НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ НА ОСНОВЕ НЕЧЕТКИХ НЕЙРОНОВ

Возможны различные варианты наделения нечетких нейронных сетей данного типа нечеткостью. Нечеткими могут быть входы, выходы, а также веса нейронов таких сетей (табл. 10.1) [17].

Таблица 10.1. Варианты наделения нейронных сетей нечеткостью

Тип нечетких нейронов	Входы	Веса	Выходы
Тип 1	Нечеткие	Четкие	Четкие
Тип 2	Нечеткие	Четкие	Нечеткие
Тип 3	Нечеткие	Нечеткие	Нечеткие
Тип 4	Четкие	Нечеткие	Нечеткие
Тип 5	Четкие	Четкие	Нечеткие
Тип 6	Четкие	Нечеткие	Четкие
Тип 7	Нечеткие	Нечеткие	Четкие

Нейронные нечеткие сети *типа 1* используются в основном для классификации нечетких входных векторов на четкие классы [18, 19], а сети *типов 2, 3 и 4*, как правило, применяются для реализации нечетких производственных моделей. Сети остальных типов практически не применяются: так, для сетей *типа 5* выходы всегда будут вещественными числами из-за четкости входов и весов, а для сетей *типов 6 и 7* фаззификация весов не является необходимой, так как выходы являются четкими.

Под *обычной (regular) нейронной нечеткой сетью* понимается нейронная сеть с нечеткими входами и выходами и/или нечеткими весами, с сигмоидальной активационной функцией, все операции в которой определены на основе принципа нечеткого расширения Заде [17].

Рассмотрим пример простой обычной нейронной нечеткой сети (рис. 10.1). Все сигналы и веса являются нечеткими числами. Двухходовые нейроны не изменяют входные сигналы, так что их выходы являются такими же, как и их входы. Входные сигналы x_i комбинируются с весами w_i :

$$r_i = w_i x_i, \quad i = 1, 2. \quad (10.1)$$

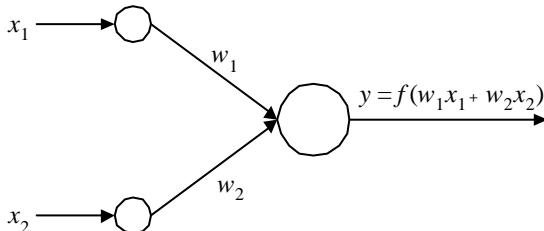


Рис. 10.1. Пример обычной нейронной нечеткой сети (нечеткого нейрона)

При вычислении r_i используется принцип нечеткого расширения Заде. Значения r_i агрегируются с использованием стандартного нечеткого расширенного сложения:

$$net = r_1 + r_2 = w_1 x_1 + w_2 x_2. \quad (10.2)$$

Выходное значение нейрона образуется в результате преобразования значения net сигмоидальной функцией f :

$$y = f(net) = f(w_1 x_1 + w_2 x_2). \quad (10.3)$$

где $f(net) = (1 + \exp(-net))^{-1}$.

Функция принадлежности выхода нейрона к выходному нечеткому множеству вычисляется на основе принципа нечеткого обобщения Заде:

$$\mu(y) = \begin{cases} (w_1 x_1 + w_2 x_2)(f^{-1}(y)), & \text{если } 0 \leq y \leq 1, \\ 0, & \text{иначе.} \end{cases} \quad (10.4)$$

где $f^{-1}(y) = \ln y - \ln(1 - y)$.

Бакли (Buckley) и Хаяши (Hayashi) [5] показали, что обычные (regular) нейронные нечеткие сети являются монотонными, т. е. если $x_1 \subset x'_1$ и $x_2 \subset x'_2$, то

$$y = f(w_1 x_1 + w_2 x_2) \subset y' = f(w_1 x'_1 + w_2 x'_2), \quad (10.5)$$

где f – сигмоидальная активационная функция, а все операторы в этом выражении реализованы на основе принципа нечеткого обобщения Заде.

Это значит, что нейронные нечеткие сети, основанные на принципе нечеткого обобщения, могут быть универсальными аппроксиматорами только для непрерывных монотонных функций. Если функция не является монотонной, то нет никакой уверенности, что она будет аппроксимирована нейронной нечеткой сетью, основанной на принципе нечеткого расширения.

Помимо предложенной выше трактовки нечетких нейронов, обобщающей существующую модель классического нейрона с точки зрения фазификации входов, выходов и весов, к рассматриваемому в данной главе способу введения нечеткости в нейроны можно отнести реализацию учета неопределенности, связанной с оценкой возможности пребывания нейрона в различных состояниях. Эта идея впервые была предложена в работе [4] и развита в работе [20].

В отличие от обычных такие нечеткие нейроны имеют несколько выходов. Входы этих нейронов являются нечеткими, а на выходах формируются степени принадлежности к нечетким множествам. Активизация нечетких нейронов представляет собой нечеткий процесс.

Рассмотрим основные структуры подобных нечетких нейронов, используемых для формирования нейронных нечетких сетей, позволяющих, например, синтезировать нечеткие автоматы и нечеткие сети Петри.

Нечеткий нейрон Квана (Kwan) и Кэи (Cai) (рис. 10.2) [20]. Сигналы x_i комбинируют с весами w_i на основе операции произведения:

$$r_i = w_i x_i, i = 1, \dots, n; \quad (10.6)$$

эти промежуточные результаты r_i агрегируются на основе некоторой функции h :

$$z = h(w_1 x_1, w_2 x_2, \dots, w_m x_m); \quad (10.7)$$

состояние нейрона вычисляется в соответствии с выражением

$$s = f(z - \theta), \quad (10.8)$$

где f – функция активации, а θ – порог срабатывания нейрона.

И значения на выходах этого нейрона вычисляются следующим образом:

$$y_j = g_i(s), \quad j = 1, \dots, p, \quad (10.9)$$

где g_i ($j = 1, \dots, p$) – выходные функции нейрона, которые представляют собой функции принадлежности входных векторов $x = \{x_1, x_2, \dots, x_m\}$ ко всем p нечетким множествам.

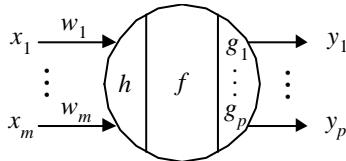


Рис. 10.2. Структура нечеткого нейрона Квана и Кэи

Нечеткий нейрон «MAX» Квана и Кэи (рис. 10.3) [20]. Сигналы x_i комбинируются с весами w_i на основе операции произведения:

$$r_i = w_i x_i, \quad i = 1, \dots, 2; \quad (10.10)$$

эти промежуточные результаты r_i агрегируются на основе S-нормы max:

$$z = \max\{r_1, r_2\} = \max\{w_1 x_1, w_2 x_2\}. \quad (10.11)$$

И на j -м выходе этого нейрона формируется сигнал:

$$\begin{aligned} y_j &= g_i(s) = g_i(f(z - \theta)) = \\ &= g_i(f(\max\{w_1 x_1, w_2 x_2\} - \theta)), \quad j = 1, \dots, p, \end{aligned} \quad (10.12)$$

где g_i ($j = 1, \dots, p$) – выходные функции нейрона, которые представляют собой функции принадлежности входных векторов $x = \{x_1, x_2, \dots, x_m\}$ ко всем p нечетким множествам.

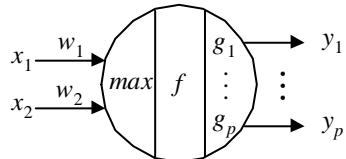


Рис. 10.3. Структура нечеткого нейрона «MAX» Квана и Кэи

Нечеткий нейрон «MIN» Квана и Кэи (рис. 10.4) [20]. Сигналы x_i комбинируются с весами w_i на основе операции произведения:

$$r_i = w_i x_i, \quad i = 1, 2; \quad (10.13)$$

эти промежуточные результаты r_i агрегируются на основе Т-нормы min:

$$z = \min\{r_1, r_2\} = \min\{w_1 x_1, w_2 x_2\}. \quad (10.14)$$

И на j -м выходе нейрона формируется сигнал:

$$\begin{aligned} y_j &= g_j(s) = g_j(f(z - \theta)) = \\ &= g_j(f(\min \{w_1 x_1, w_2 x_2\} - \theta)), \quad j = 1, \dots, p, \end{aligned} \quad (10.15)$$

где g_j ($j = 1, \dots, p$) – выходные функции нейрона, которые представляют собой функции принадлежности входных векторов $x = \{x_1, x_2, \dots, x_m\}$ ко всем p нечетким множествам.

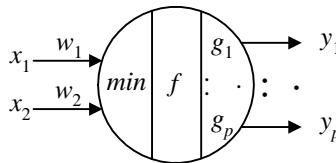


Рис. 10.4. Структура нечеткого нейрона «MIN» Квана и Кэя

Глава 11. ГИБРИДНЫЕ НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ НА ОСНОВЕ НЕЙРОНОВ, РЕАЛИЗУЮЩИХ НЕЧЕТКИЕ ОПЕРАЦИИ

Как было отмечено в гл. 10, нейронные нечеткие сети, основанные на принципе обобщения, могут аппроксимировать только непрерывные монотонные функции. Для построения универсального аппроксиматора необходимо отказаться от принципа нечеткого расширения. В работе [5] Бакли (Buckley) и Хаяши (Hayashi) в качестве универсального аппроксиматора предложили использовать так называемые гибридные нейронные нечеткие сети (hybrid fuzzy neural networks). Эти, а также ряд других авторов показали, что данные сети могут аппроксимировать любую непрерывную функцию, определенную в замкнутой области.

Под *гибридной нейронной нечеткой сетью* будем понимать нейронную сеть, сформированную на основе нечетких нейронов с нечеткими входами и выходами и/или нечеткими весами, а также с четкой активационной функцией. Однако в таких нечетких нейронах в отличие от обычной (regular) нейронной нечеткой сети:

- входные сигналы x_i и веса w_i можно комбинировать с использованием Т-нормы, S-нормы или некоторого другого непрерывного оператора;
- результаты полученных комбинирований r_1 и r_2 могут агрегироваться с использованием Т-нормы, S-нормы или некоторой другой непрерывной функции;
- активационная функция f может быть любой другой (не только сигмоидальной) функцией.

Входы, выходы и веса гибридной нечеткой нейронной сети (обычно представляющие собой степени принадлежности к нечетким множествам) являются вещественными числами из интервала $[0, 1]$.

Примечание. В целом ряде работ под *гибридными нейронными нечеткими сетями* понимают в том числе нечеткие нейронные сети, реализующие нечеткие производственные модели и соответствующие им алгоритмы нечеткого вывода. Однако, по мнению авторов данной книги, указанная разновидность гибридных нейронных нечетких сетей должна быть выделена в отдельный класс *нечетких нейронных производственных сетей* (см. ч. III). Под гибридными нейронными нечеткими сетями, исходя из целей книги, будем подразумевать нейронные нечеткие сети, удовлетворяющие отмеченным в приведенном определении признакам и предназначенные для решения задач, типичных для задач классических нейронных сетей (классификация и распознавание образов, кластеризация/категоризация, аппроксимация, прогнозирование, ассоциативная память).

11.1. Нейроны, реализующие нечеткие операции

Рассмотрим основные разновидности нечетких нейронов, реализующих нечеткие операции и используемых для создания гибридных нейронных нечетких сетей.

Нечеткий нейрон «И» (рис. 11.1) [21, 22]. Сигналы x_i и веса w_i в данном случае комбинируются с помощью операции S-нормы:

$$r_i = S(w_i, x_i), i = 1, 2, \quad (11.1)$$

а выходное значение агрегируется с помощью операции T-нормы:

$$y = T(r_1, r_2) = T(S(w_1, x_1), S(w_2, x_2)). \quad (11.2)$$

Если $T = \min$, $S = \max$, то нечеткий нейрон «И» реализует min-max-композицию:

$$y = \min\{w_1 \vee x_1, w_2 \vee x_2\}. \quad (11.3)$$

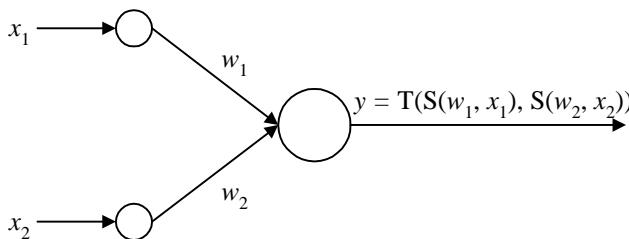


Рис. 11.1. Структура нечеткого нейрона «И»

Нечеткий нейрон «ИЛИ» (рис. 11.2). Сигналы x_i и веса w_i здесь комбинируются с помощью Т-нормы:

$$r_i = T(w_i, x_i), i = 1, 2, \quad (11.4)$$

а выход образуется с применением S-нормы:

$$y = S(r_1, r_2) = S(T(w_1, x_1), T(w_2, x_2)). \quad (11.5)$$

Если принять $T = \min$, $S = \max$, то нечеткий нейрон «ИЛИ» реализует max-min-композицию:

$$y = \max\{w_1 \wedge x_1, w_2 \wedge x_2\}. \quad (11.6)$$

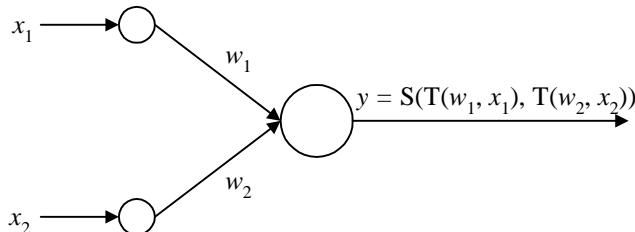


Рис. 11.2. Структура нечеткого нейрона «ИЛИ»

Нечеткие «И»- и «ИЛИ»-нейроны реализуют стандартные логические операции над степенями принадлежности. Причем при формировании нейронных нечетких сетей на основе таких нейронов необходимо различать назначение создаваемых с их помощью соединений между различными слоями. Отметим, что, во-первых, чем больше значение w_i , тем сильнее воздействие от x_i на выходе у нечеткого «ИЛИ»-нейрона, во-вторых, чем меньше значение w_i , тем сильнее воздействие от x_i на выходе у нечеткого «И»-нейрона.

Для нечеткого «И»-нейрона диапазон изменения выхода y в предположении, что все x_i принимают значения от 0 до 1 (с учетом свойства монотонности Т-норм), составляет

$$y \in [T(w_1, w_2), 1], \quad (11.7)$$

а для нечеткого «ИЛИ»-нейрона диапазон изменения выхода y составляет

$$y \in [0, S(w_1, w_2)]. \quad (11.8)$$

Нечеткий нейрон «Импликация–ИЛИ» (рис. 11.3) [23, 24]. Сигналы x_i и веса w_i комбинируются с помощью операции нечеткой импликации I:

$$r_i = I(w_i, x_i) = x_i \rightarrow w_i, i = 1, 2, \quad (11.9)$$

а выходное значение образуется агрегированием значений r_i посредством операции S-нормы:

$$y = S(r_1, r_2) = S(I(w_1, x_1), I(w_2, x_2)). \quad (11.10)$$

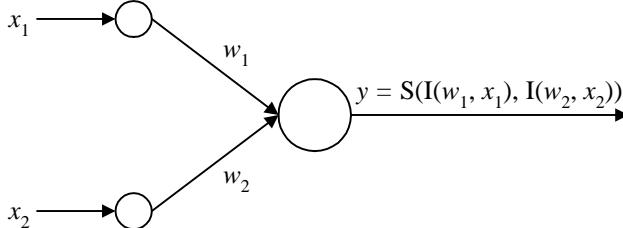


Рис. 11.3. Структура нечеткого нейрона «Импликация–ИЛИ»

Нечеткие нейроны для реализации композиционных правил вывода (рис. 11.4) [5].

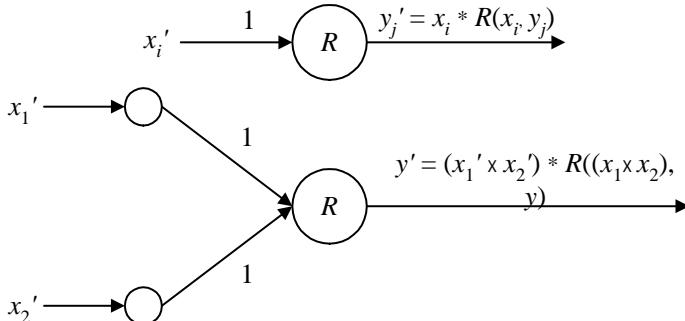


Рис. 11.4. Примеры структур нечетких нейронов для композиционных правил вывода

Нечеткие нейроны данного типа задают различные нечеткие отношения, например $R((x_i \times x_j), y)$ (на основе Т- и S-норм и других нечетких операций), и используются для построения гибридных нейронных нечетких сетей, реализующих нечеткие композиционные правила вывода (см. п. 11.2.3).

11.2. Примеры построения и использования гибридных нейронных нечетких сетей

В качестве типичных примеров построения и использования гибридных нечетких нейронных сетей (на основе представленных выше нечетких нейронов) рассмотрим:

- гибридные нейронечеткие классификаторы [17];
- гибридные нейронные нечеткие сети для реализации деревьев классификации (решений) [25];

- гибридные нейронные нечеткие сети для реализации композиционных правил вывода [17];
- гибридные нейронные нечеткие сети для нечеткого логического вывода [26];
- гибридные нейронные нечеткие сети для извлечения нечетких производственных правил из числовых данных [27, 28].

11.2.1. Гибридные нейроннечеткие классификаторы

Традиционный подход к классификации образов основан на предварительной кластеризации обучающих примеров и отнесении их к заданным классам. Сложность и ограничения при осуществлении этого предварительного этапа в большой степени обусловлены недостаточной эффективностью определения границ между кластерами. Эта проблема становится еще более труднорешаемой, когда число используемых параметров существенно возрастает. В отличие от традиционного подхода нечеткая классификация допускает непрерывность границы между двумя соседними классами с наложением областей, в каждой из которых классифицируемый объект характеризуется своей степенью принадлежности. Данный подход не только подходит для многих приложений, характеризующихся нечеткими границами между классами, но также обеспечивает достаточно простое представление потенциально сложного разделения пространства признаков. Покажем, каким образом данная задача может быть решена с использованием гибридной нечеткой нейронной сети, а именно *гибридного нейроннечеткого классификатора* [17].

Допустим, что задано K образов в виде m -мерных четких векторов $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)})$, $k = 1, \dots, K$, которые относятся к двум классам. Типичные нечеткие правила классификации для $m = 2$ имеют следующий вид:

$$\begin{aligned} &\text{ЕСЛИ } x_1^{(k)} \text{ есть малый И } x_2^{(k)} \text{ есть очень большой,} \\ &\text{ТО } \mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}) \text{ принадлежит к классу } C_1, \end{aligned} \quad (11.11)$$

$$\begin{aligned} &\text{ЕСЛИ } x_1^{(k)} \text{ есть большой И } x_2^{(k)} \text{ есть очень малый,} \\ &\text{ТО } \mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}) \text{ принадлежит к классу } C_2, \end{aligned} \quad (11.12)$$

где $x_1^{(k)}, x_2^{(k)}$ – значения параметров k -го образа; *очень малый*, *малый*, *очень большой* – лингвистические термы, характеризующие соответствующие функции принадлежности.

Уровень активизации (firing level) i -го правила:

$$\begin{aligned} \Pi_i: & \text{ЕСЛИ } x_1^{(k)} \text{ есть } A_{ij} \text{ И } x_2^{(k)} \text{ есть } A_{ji}, \\ \text{ТО } & x^{(k)} = (x_1^{(k)}, x_2^{(k)}) \text{ принадлежит к классу } C_i, i = 1, \dots, n, \end{aligned} \quad (11.13)$$

относительно классифицируемого k -го образа интерпретируется как степень его принадлежности к классу C_i .

Этот уровень активизации i -го правила, обозначаемый как α_i , обычно определяется следующим образом:

$$\alpha_i = T(\mu_{A_{ij}}(x_1^{(k)}), \mu_{A_{ji}}(x_2^{(k)})), \quad (11.14)$$

где T – оператор Т-нормы, моделирующий логический оператор «И».

В результате предъявления образа осуществляется параллельное сравнение его компонентов с антецедентами всех n правил с получением уровней активизации $\alpha_i, i = 1, \dots, n$.

Затем уровни активизации всех правил агрегируются на основе заданной операции. В качестве ее часто используют оператор Т-нормы.

Кроме того, при разработке структуры гибридного нейронечеткого классификатора необходимо таким образом подбирать его характеристики, чтобы максимально учесть возможность обучения в случае изменения данных, гибкость настройки при получении новой информации.

Задача нечеткой классификации заключается в выполнении соответствующего нечеткого разделения признакового пространства. При этом число ошибочно классифицированных образов должно быть очень мало или вовсе равно нулю. Поэтому база нечетких правил должна быть оптимизирована путем исключения неиспользуемых правил.

Рассмотрим пример решения задачи классификации образов на два класса. Предположим, что нечеткое разделение для каждого входного параметра предполагает наличие трех лингвистических термов {малый, средний, большой}, которые представлены гауссовыми функциями принадлежности (рис. 11.5).

Для каждого входного параметра их начальные разделения удовлетворяют условию перекрытия термов на уровне 0.5, и каждый образ $x^{(k)}$ будет отнесен к классу C_i , если существует по крайней мере одно правило для этого класса C_i из базы правил, степень срабатывания которого (определенная посредством Т-нормы в виде операции минимума) относительно $x^{(k)}$ является большей или равной 0.5. Правило для заданного входного образа $x^{(k)}$ формируется так, чтобы в комбинации нечетких множеств каждое из них давало наибольшую степень принадлежности для соответствующего входного признака. Если эта комбинация неидентична антецедентам уже существующего правила, то создается новое правило.

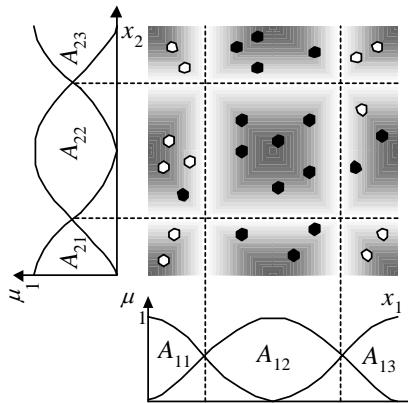


Рис. 11.5. Пример нечеткого разделения пространства входных параметров

Однако если нечеткое разделение выполнено неправильно или если число лингвистических термов для входных признаков не является достаточным, то некоторые образы могут быть классифицированы неверно.

Для начального нечеткого разбиения пространства входных признаков, показанного на рис. 11.5, могут быть сгенерированы следующие 9 правил:

П₁: **ЕСЛИ** $x_1^{(k)}$ есть *малый* И $x_2^{(k)}$ есть *большой*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_1 ,

П₂: **ЕСЛИ** $x_1^{(k)}$ есть *малый* И $x_2^{(k)}$ есть *средний*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_1 ,

П₃: **ЕСЛИ** $x_1^{(k)}$ есть *малый* И $x_2^{(k)}$ есть *малый*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_1 ,

П₄: **ЕСЛИ** $x_1^{(k)}$ есть *большой* И $x_2^{(k)}$ есть *малый*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_1 ,

П₅: **ЕСЛИ** $x_1^{(k)}$ есть *большой* И $x_2^{(k)}$ есть *большой*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_1 ,

П₆: **ЕСЛИ** $x_1^{(k)}$ есть *средний* И $x_2^{(k)}$ есть *малый*,

ТО $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_2 ,

П₇: ЕСЛИ $x_1^{(k)}$ есть средний И $x_2^{(k)}$ есть средний,

ТО $x^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_2 ,

П₈: ЕСЛИ $x_1^{(k)}$ есть средний И $x_2^{(k)}$ есть большой,

ТО $x^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_2 ,

П₉: ЕСЛИ $x_1^{(k)}$ есть большой И $x_2^{(k)}$ есть средний,

ТО $x^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_2 ,

где лингвистический терм *малый* используется для A_{11}, A_{21} ; *средний* – для A_{12}, A_{22} ; *большой* – для A_{13}, A_{23} .

Однако та же самая степень ошибки может быть достигнута при использовании условия «ЕСЛИ $x_1^{(k)}$ есть средний» при отнесении образа $x^{(k)} = (x_1^{(k)}, x_2^{(k)})$ к классу C_2 независимо от значения $x_2^{(k)}$. Поэтому три правила – 6, 7 и 8 могут быть заменены одним:

ЕСЛИ $x_1^{(k)}$ есть средний,

ТО $x^{(k)} = (x_1^{(k)}, x_2^{(k)})$ принадлежит к классу C_2 .

Сан (Sun) и Чанг (Jang) [29] предложили архитектуру для решения задач нечеткой классификации. Одна из возможных структур гибридного нейронечеткого классификатора представлена на рис. 11.6.

Слой 1. На выходе элементов этого слоя формируются степени принадлежности входных переменных к определенным для них нечетким множествам A_{ij} :

$$\mu_{A_{ij}}(x'_j) = \exp\left[-\frac{1}{2}\left(\frac{x'_j - a_{ij}}{b_{ij}}\right)^2\right], \quad (11.15)$$

где a_{ij}, b_{ij} – параметры функции принадлежности колоколообразного типа.

Начальные значения этих параметров установлены таким образом, чтобы функции принадлежности удовлетворяли свойствам полноты, нормальности и выпуклости. Значения данных параметров корректируются в процессе обучения сети (основанном на градиентном методе).

Слой 2. Каждый элемент этого слоя является рассмотренным выше нечетким нейроном «И», выходной сигнал которого представляют «силу» срабатывания нечеткого правила относительно классифицируемого образа.

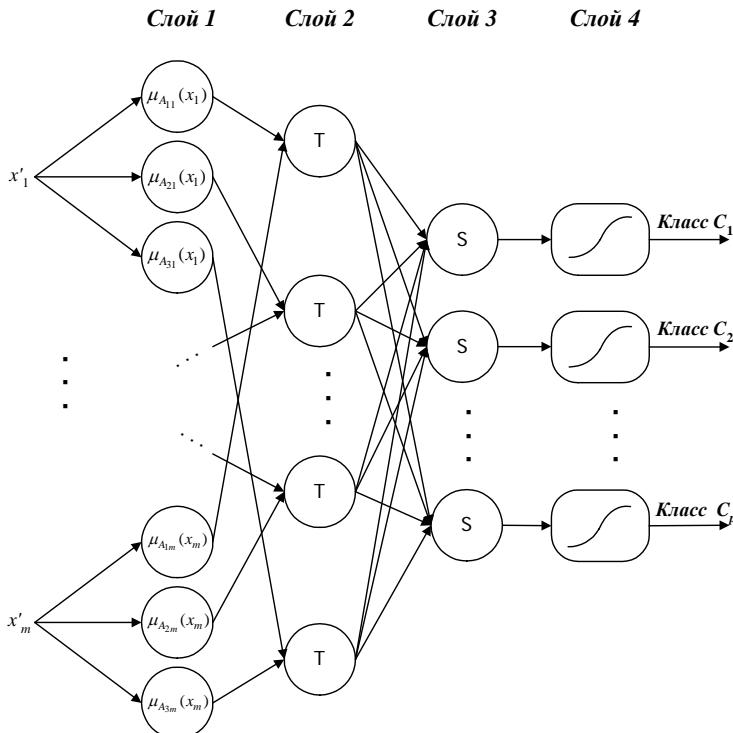


Рис. 11.6. Пример структуры гибридного нейронно-нечеткого классификатора

Слои 3–4. Элементы этих слоев предназначены для взвешенного суммирования значений выходов элементов предыдущего слоя. А значения на выходах элементов слоя 4 формируются с использованием активационных функций сигмоидного типа. Эти выходы трактуются как степени принадлежности предъявленного объекта к соответствующему классу.

Алгоритм обучения данной сети в принципе не отличается от рассмотренных алгоритмов обучения сетей ANFIS-типа.

11.2.2. Гибридные нейронные нечеткие сети для реализации деревьев классификации

Деревьями классификации называется метод, позволяющий предсказывать степень принадлежности фактов или объектов к тому или иному классу категориальной зависимой переменной в зависимости от соответствующих значений одной или нескольких предикторных переменных

(факторов). Графически **бинарное дерево классификации** (или дерево решений) отображается в виде набора узлов, среди которых выделяют внутренние (с двумя дочерними) и терминальные, или окончательные (не имеющие дочерних узлов). Каждый внутренний узел связан с решающей функцией, определяющей, какой узел будет выбран следующим. В задачах классификации каждый окончательный узел ассоциируется с тем или иным классом (образом); в задачах построения регрессии с каждым терминальным узлом связана константа – прогнозируемое значение отклика при предъявлении некоторого входного вектора [25].

На рис. 11.7, *a* приведено типичное бинарное дерево классификации (решений) для задачи регрессии при входах (факторах) x_1 и x_2 и отклике (выходе) y . Очевидно, что дерево решений разбивает пространство входов на 4 неперекрывающиеся прямоугольные области (см. рис. 11.7, *б*), с каждой из которых ассоциирована константа f_i как прогнозируемое значение выхода.

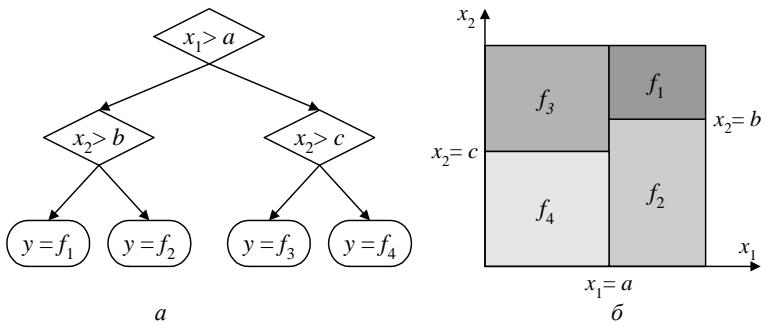


Рис. 11.7. Бинарное дерево: *а* – классификация;
б – разбиение пространства входов

Можно улучшить качество классификации в пределах каждой из показанных на рис. 11.7, *б* подобластей, если с каждым терминальным узлом связать не константу, а, например, линейную функцию вида

$$f_i = q_i + p_i x_1 + r_i x_2, \quad i = 1, \dots, 4. \quad (11.16)$$

где q_i, p_i, r_i – константы.

Подобная модификация приводит к известному алгоритму CART (Classification and Regression Trees – алгоритму деревьев классификации и регрессии), применяемому при анализе сложных зависимостей. Несмотря на достоинства данного подхода, его недостатком является то, что получаемая поверхность отклика имеет разрывы непрерывности на границах выделенных подобластей входных факторов, что не всегда прием-

лемо. Этот недостаток можно преодолеть, если для реализации метода бинарных деревьев решений использовать сети ANFIS-типа.

Для примера дерева решений, показанного на рис. 11.7, можно записать следующий набор четких правил:

$$\begin{cases} \text{ЕСЛИ } x_1 > a \text{ И } x_2 > b, & \text{ТО } y = f_1, \\ \text{ЕСЛИ } x_1 > a \text{ И } x_2 < b, & \text{ТО } y = f_2, \\ \text{ЕСЛИ } x_1 < a \text{ И } x_2 < c, & \text{ТО } y = f_3, \\ \text{ЕСЛИ } x_1 < a \text{ И } x_2 < c, & \text{ТО } y = f_4. \end{cases} \quad (11.17)$$

Для любой пары значений (x_1, x_2) срабатывает только одно из данных правил, что, собственно, и приводит к отмеченным разрывам непрерывности. Для их устранения эти четкие правила необходимо преобразовать в нечеткие. При этом условие « $x > a$ » может быть представлено как нечеткое, характеризуемое, например,

- либо сигмоидальной функцией принадлежности (с параметром δ):

$$\mu_{x>a}(x, \delta) = \frac{1}{1 + \exp(-\delta(x - a))}, \quad (11.18)$$

- либо обобщенной S-образной функцией принадлежности (с параметрами δ и γ):

$$\mu_{x>a}(x, \delta, \gamma) = \begin{cases} 0, & \text{если } x \leq a - \delta, \\ \frac{1}{2} \left[\frac{x - (a - \delta)}{\delta} \right]^{2\gamma}, & \text{если } a - \delta < x \leq a, \\ 1 - \frac{1}{2} \left[\frac{a + \delta - x}{\delta} \right]^{2\gamma}, & \text{если } a < x \leq a + \delta, \\ 1, & \text{если } a + \delta < x. \end{cases} \quad (11.19)$$

Нечеткие правила для данного примера соответствуют гибридной нейронной нечеткой сети, структура которой приведена на рис. 11.8. Элементы «INV» сети реализуют операцию инверсии.

Слой 1. Элементы слоя вычисляют значения функций принадлежности для текущих значений входных переменных.

Слой 2. Элементы этого слоя осуществляют перемножение найденных значений.

Слой 3. Элементы слоя определяют вклад каждого правила (w_i).

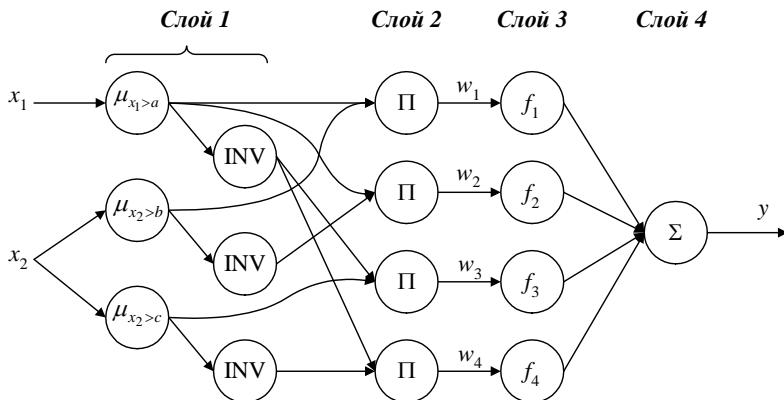


Рис. 11.8. Пример структуры гибридной нейронной нечеткой сети для реализации бинарного дерева классификации

Слой 4. Элемент данного слоя суммирует значения с выходов элементов слоя 3 и вычисляет общий выход сети.

Примечание. Операция нормализации (деление на $\sum_i w_i$) здесь не проводится, что возможно при выполнении условий следующей теоремы.

Теорема. Если в гибридной нейронной нечеткой сети, соответствующей бинарному дереву решений для задачи регрессии:

- во-первых, $\mu_{x>a}(x) + \mu_{x\leq a}(x) = 1$, где x – любая из входных переменных и a – любая из точек разбиения области определения данной переменной на подобласти,
- во-вторых, в качестве Т-нормы используется произведение, т. е. применяется логический вывод,

то сумма значений функций принадлежности (сумма значений истинности предпосылок правил) равна единице, т. е. в данном случае $\sum_i w_i = 1$.

Обучение (настройка параметров) данной гибридной нейронной нечеткой сети проводится по рассмотренным выше алгоритмам.

Следует отметить, что наиболее сложным этапом по созданию сетей подобного типа является этап разбиения исходной области определения входных переменных на подобласти.

11.2.3. Гибридные нейронные нечеткие сети для реализации композиционных правил вывода

Рассмотрим для примера базу нечетких правил:

$$\Pi_k: \text{ЕСЛИ } x \text{ есть } A_k, \text{ ТО } y = B_k, k = 1, \dots, n. \quad (11.20)$$

Для наглядности предположим, что антецедент содержит одну нечеткую переменную. Заметим, что нижеследующие рассуждения можно распространить на случай многих входных переменных.

Обычно в реальных вычислениях используют дискретизацию нечетких множеств по базовым множествам. Пусть $[x^{(\min)}, x^{(\max)}]$ содержит значения из базового множества для всех A_i . Также пусть $[y^{(\min)}, y^{(\max)}]$ содержит значения из базового множества для всех B_i .

На рис. 11.9 показан пример дискретизированного нечеткого отображения $A'(x) = (A'(x^{(1)}), \dots, A'(x^{(p)}))$ на $B'(y) = (B'(y^{(1)}), \dots, B'(y^{(q)}))$, которое нужно реализовать на основе гибридной нейронной нечеткой сети.

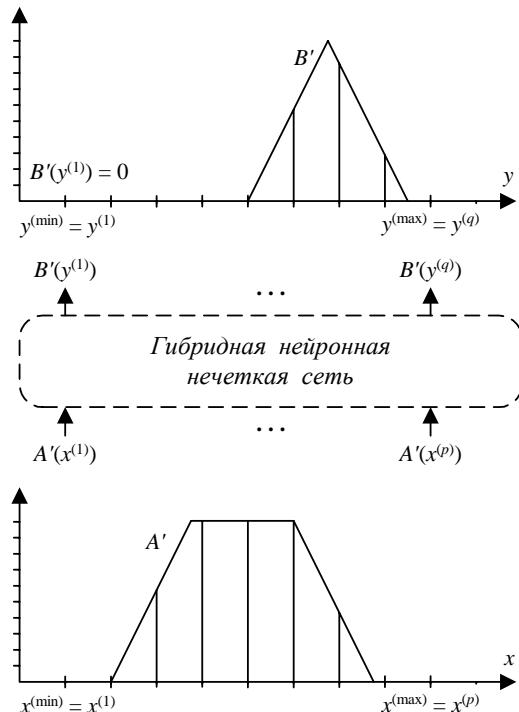


Рис. 11.9. Пример дискретизированного нечеткого отображения

Два основных способа построения гибридных нейронных нечетких сетей на основе нечетких нейронов, реализующих композиционные правила вывода (см. п. 11.1), следующие.

Первый способ основан на комбинировании всех правил в одно, которое используется для получения $B'(y) = (B'(y^{(1)}), \dots, B'(y^{(q)}))$ из $A'(x) = (A'(x^{(1)}), \dots, A'(x^{(p)}))$.

Во-первых, формируются нечеткие отношения $R_k(x^{(i)}, y^{(j)})$, моделирующие операции импликации относительно всей совокупности правил.

Существуют различные варианты осуществления такого моделирования, например:

$$R_k(x^{(i)}, y^{(j)}) = \min\{A_k(x^{(i)}), B_k(y^{(j)})\}. \quad (11.21)$$

Во-вторых, выполняется комбинирование всех нечетких отношений R_k между парами $(x^{(i)}, y^{(j)})$ в одно нечеткое отношение R с использованием операции агрегирования, которое также может быть реализовано различным образом:

$$R(x^{(i)}, y^{(j)}) = \text{Agg}(R_1(x^{(i)}, y^{(j)}), R_2(x^{(i)}, y^{(j)}), \dots, R_n(x^{(i)}, y^{(j)})), \quad (11.22)$$

где Agg – обозначение операции агрегирования.

Способ комбинирования $(B'(y^{(1)}), \dots, B'(y^{(q)}))$ из $(A'(x^{(1)}), \dots, A'(x^{(p)}))$ называется композиционным правилом вывода. Пусть

$$\lambda_{ij} = A'(x^{(i)}) * R(x^{(i)}, y^{(j)}), \quad (11.23)$$

где $*$ – обозначение операции комбинирования (обычно *T-норма*).

В-третьих, формируется множество

$$B'(y) = (B'(y^{(1)}), \dots, B'(y^{(q)})), \quad (11.24)$$

где $B'(y^{(j)}) = \text{Agg}(\lambda_{1j}, \lambda_{2j}, \dots, \lambda_{pj})$.

На рис. 11.10 показан пример структуры гибридной нейронной нечеткой сети, реализующей рассматриваемый способ формирования композиционного правила вывода.

Данная сеть обеспечивает быстрое параллельное вычисление дискретизированного нелинейного отображения. Однако ее реализация требует существенных вычислительных затрат. Например, для $[x^{(\min)}, x^{(\max)}] = [y^{(\min)}, y^{(\max)}] = [-10, 10]$ с шагом дискретизации 0.01 $p = q = 1000$. Тогда число входных и выходных элементов будет равно 2000, а количество элементов, реализующих нечеткие отношения и связывающих входные и выходные элементы, – 2000².

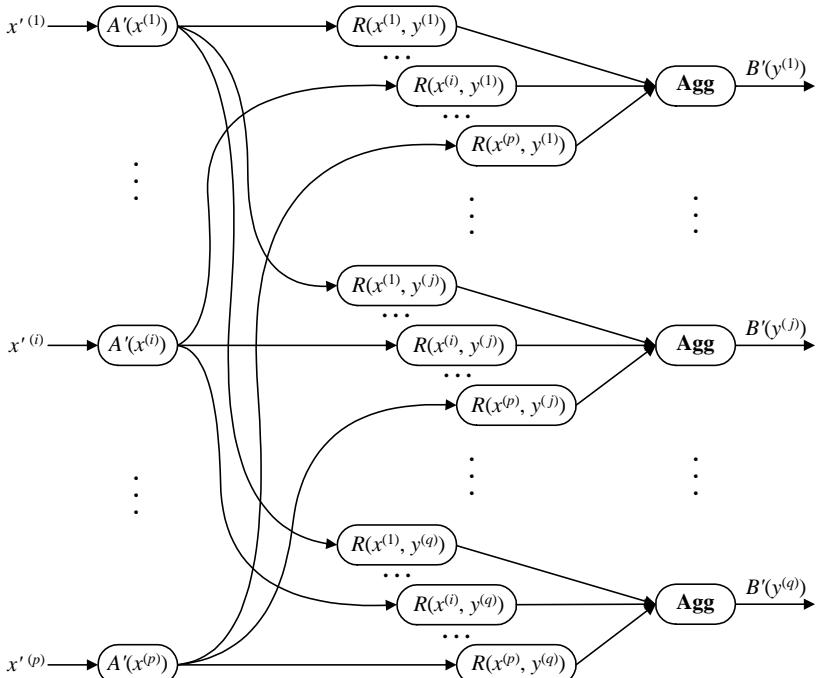


Рис. 11.10. Структура гибридной нейронной нечеткой сети, реализующая первый способ формирования композиционного правила вывода

Второй способ основан на том, что активизация (fire) правил $A'(x^{(1)}), \dots, A'(x^{(p)})$ осуществляется индивидуально, после чего выполняется комбинирование результатов с получением $B'(y) = (B'(y^{(1)}), \dots, B'(y^{(q)}))$. При этом выполняется композиция $A'(x) = (A'(x^{(1)}), \dots, A'(x^{(p)}))$ с каждым нечетким отношением $R_k(x^{(i)}, y^{(j)})$ с получением промежуточных результатов $B'_k(y) = (B'_k(y^{(1)}), \dots, B'_k(y^{(q)}))$, $k = 1, \dots, n$, которые затем комбинируются (агрегируются) в $B'(y) = (B'(y^{(1)}), \dots, B'(y^{(q)}))$.

Во-первых, относительно каждого правила формируется нечеткое отношение $R_k(x^{(i)}, y^{(j)})$ на основе, например, операции \min :

$$R_k(x^{(i)}, y^{(j)}) = \min\{A_k(x^{(i)}), B_k(y^{(j)})\}. \quad (11.25)$$

Во-вторых, вычисляются

$$\lambda_{kij} = A'(x^{(i)}) * R_k(x^{(i)}, y^{(j)}), \text{ и } B'_k(y^{(j)}) = \text{Agg}(\lambda_{k1j}, \lambda_{k2j}, \dots, \lambda_{kpj}). \quad (11.26)$$

В-третьих, выполняется агрегирование компонентов $B'_k(y^{(j)})$:

$$B'(y^{(j)}) = \text{Agg}(B'_{1j}(y^{(j)}), \dots, B'_{nj}(y^{(j)})), \quad j = 1, \dots, q. \quad (11.27)$$

Структура гибридной нейронной нечеткой сети, реализующая данный способ формирования композиционных правил вывода, представлена на рис. 11.11.

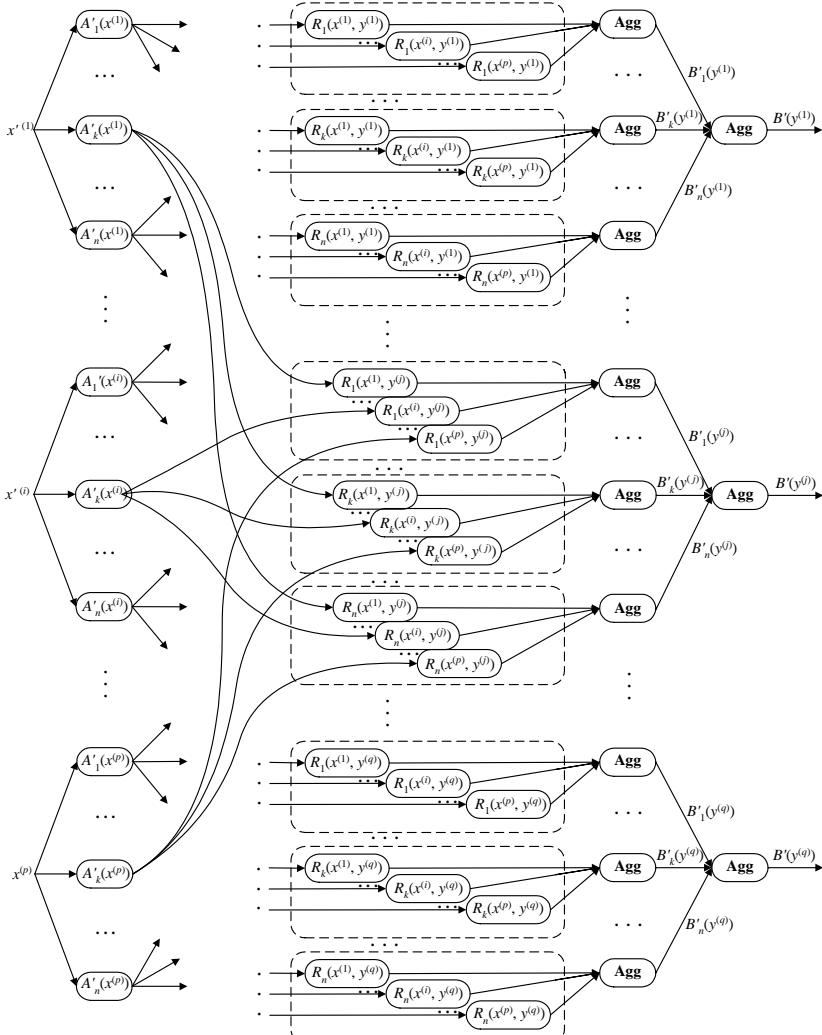


Рис. 11.11. Структура гибридной нейронной нечеткой сети, реализующая второй способ формирования композиционного правила вывода

Как и в предыдущем случае, реализация гибридной нейронной нечеткой сети данного типа требует значительных вычислительных затрат. Предположим, что заданы 10 нечетких правил типа (11.20) и $[x^{(\min)}, x^{(\max)}] = [y^{(\min)}, y^{(\max)}] = [-10, 10]$ с шагом дискретизации 0.01, т. е. $p = q = 1000$. Следовательно, число входных и выходных элементов будет равно 2000; количество элементов, реализующих нечеткие отношения – 10×2000^2 ; 10×2000 элементов, связывающих элементы предпоследнего и выходного слоев.

Очевидно, что использование гибридных нейронных нечетких сетей рассмотренных типов, реализующих композиционные правила вывода, является практически целесообразным в случае ограниченного набора дискрет базовых множеств.

11.2.4. Гибридные нейронные нечеткие сети для нечеткого логического вывода

Интересную разновидность гибридных нейронных нечетких сетей для выполнения нечеткого логического вывода предложили Келлер (Keller), Ягер (Yager) и Тахани (Tahani) [26]. В данной сети каждая базовая структура, представленная на рис. 11.12, реализует отдельное правило из базы нечетких правил вида

ЕСЛИ x_1 есть A_1 И ... И x_m есть A_m , **ТО** y есть B . (11.28)

Пусть $[x^{(\min)}, x^{(\max)}]$ содержит значения из базового множества для всех A_i (в i -м пункте антецедента). Также пусть $[y^{(\min)}, y^{(\max)}]$ содержит значения из базового множества для B .

При этом нечеткие множества $A'_i(x_i)$ представляются в виде совокупности значений функций принадлежности для дискретизированных значений базового множества (аналогично рис. 11.9): $A'_i(x) = (A'_i(x_i^{(1)}), \dots, A'_i(x_i^{(p)}))$. В структуре рассматриваемой на рис. 11.12 сети это реализуется элементами *слоя 1*.

Возможны два варианта активизации нечеткой переменной в антецеденте правила. В обоих случаях полученные значения нечеткой переменной определяют веса элементов *слоя 2* гибридной сети, предназначенных для определения уровней срабатывания входных переменных в правиле.

В первом случае веса w_{ij} являются дополнениями нечетких множеств для входных нечетких переменных, например для i -й нечеткой переменной (i -го пункта антецедента):

$$w_{ij} = 1 - A_i(x_i^{(j)}), i = 1, \dots, m, j = 1, \dots, p. \quad (11.29)$$

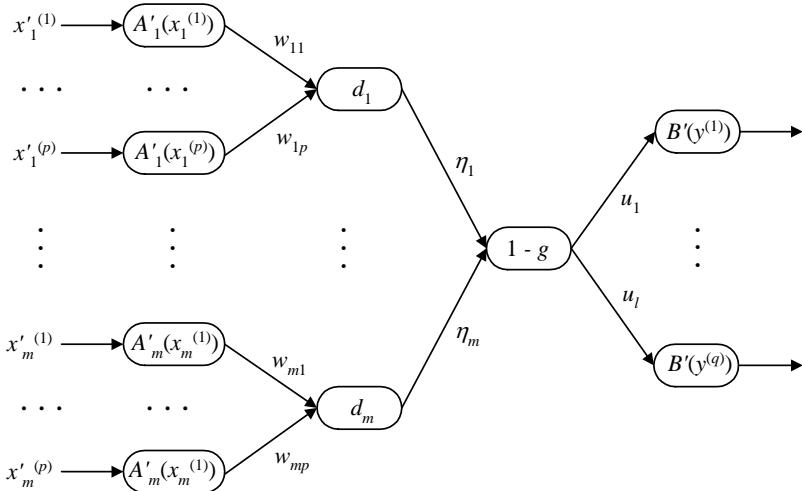
Слой 1 Слой 2 Слой 3 Слой 4


Рис. 11.12. Структура гибридной нейронной нечеткой сети для нечеткого логического вывода

Вычисляемые таким образом веса задают степень различия между распределением заданных входных значений и распределением значений входной переменной для i -го пункта антецедента. Причем чем существеннее эта разница, тем ближе степень различия к единице. Следовательно, если каждый элемент w_{ij} вычисляет степень подобия между входным значением $A'_i(x_i^{(j)})$ и дополнением значения i -го пункта антецедента $(1 - A_i(x_i^{(j)}))$, то таким образом определяется степень «локального» различия. Элементы следующего слоя – *слоя 2* служат объединению этих локальных результатов.

Назначение элементов *слоя 2* заключается в определении степени различия между нечетким множеством i -го пункта антецедента A_i и соответствующими входными данными A'_i :

$$d_k = \max_j \{w_{ij} * A'_i(x^{(j)})\} = \max_j \{(1 - A_i(x^{(j)})) * A'_i(x^{(j)})\}, k = 1, \dots, m, \quad (11.30)$$

где $*$ – обозначение операции умножения или минимума:

$$\begin{aligned} d_k^1 &= \max_j \{(1 - A_i(x^{(j)}))A'_i(x^{(j)})\}, \\ d_k^2 &= \max_j \min \{(1 - A_i(x^{(j)})), A'_i(x^{(j)})\}. \end{aligned} \quad (11.31)$$

Во втором случае элементы *слоя 2* вычисляют

$$d_k^2 = \max_j |A_i(x^{(j)}) - A'_i(x^{(j)})|. \quad (11.32)$$

Функция активации элементов этого слоя устанавливается такой, что выход элемента представляет собой значение, полученное в результате комбинирования входов и весов.

Степени различия от каждого элемента *слоя 2* комбинируются в элементе *слоя 3* с получением общей степени различия между дискретизированными значениями нечетких множеств пункта антецедента и соответствующих входных данных. Полученные степени различия задают сигналы запрещения («торможения») срабатывания (firing) правила. Веса η_k в этих соединениях соответствуют важности различных пунктов антецедента в правиле. Элементы *слоя 3* вычисляют значения

$$1 - g = 1 - \max_k \{\eta_k d_k\}. \quad (11.33)$$

При вычислении этих значений вместо операции *max* можно воспользоваться взвешенным суммированием, т. е.

$$1 - g = 1 - \sum_{k=1}^m \eta_k d_k. \quad (11.34)$$

Веса u_l элементов выходного *Слоя 4* несут информацию о консеквенте правила. Если высказывание «*у* есть *B*» характеризуется дискретным распределением значений $B(y) = (B(y^{(1)}), \dots, B(y^{(q)}))$, то

$$u_l = 1 - B(y^{(l)}), l = 1, \dots, q. \quad (11.35)$$

Каждый *l*-й ($l = 1, \dots, q$) элемент выходного слоя формирует значение

$$\begin{aligned} B'(y^{(l)}) &= 1 - u_l(1 - g) = \\ &= 1 - (1 - B(y^{(l)}))(1 - g) = B(y^{(l)}) + g - B(y^{(l)})g. \end{aligned} \quad (11.36)$$

Из этого выражения следует, что если $g = 0$, то правило срабатывает в точности с консеквентом «*у* есть *B*». В ином случае если общая степень различия равна единице, то заключение срабатывающего правила представляется распределением, составленное из единиц; следовательно, консеквент представляет собой «*у* есть *неизвестен*».

Следует отметить, что более «подробная» дискретизация нечетких множеств как для данной сети, так и для рассмотренных в п. 11.2.3 сетей приводит к существенному увеличению их размеров.

В работе [26] доказан ряд теорем, обосновывающих свойства гибридной нейронной нечеткой сети данного типа.

11.2.5. Гибридные нейронные нечеткие сети для извлечения нечетких правил из данных

Еще одной интересной разновидностью гибридных нейронных нечетких сетей являются сети для извлечения нечетких производственных правил из числовых данных. Это, как правило, осуществляется в рамках конкретной решаемой задачи, например аппроксимации функций, классификации образов.

При решении задачи аппроксимации функций нечеткие нейронные сети более предпочтительны по сравнению с обычными нейронными сетями, так как их поведение может быть описано на основе нечетких правил и, таким образом, настроено в нотации этих правил. Однако вследствие того, что пространство каждой входной переменной, в свою очередь, делится на несколько интервалов, это приводит к ограничению возможного количества входных переменных.

Для преодоления этой проблемы при решении задачи извлечения нечетких правил из числовых данных для нейронных нечетких сетей предложено несколько методов [17].

Изначально полагаем, что нечеткие «ЕСЛИ–ТО»-правила уже представлены в лингвистической форме или в виде четкой кластеризации корректных входных–выходных наборов данных.

11.2.5.1. В работе [30] показано, каким образом можно извлечь нечеткие «ЕСЛИ–ТО»-правила из входных–выходных наборов данных на основе методов нечеткой кластеризации.

11.2.5.2. Входные векторы входных–выходных наборов данных также могут быть представлены в качестве входов для самоорганизующихся карт Кохонена, которые могут быть интерпретированы как термы лингвистических переменных [31]. Основная идея этой интерпретации заключается в том, чтобы не выбирать элемент–победитель до тех пор, пока не будут настроены все веса самоорганизующейся карты. Таким образом, каждый выходной элемент i , являющийся победителем для данного входного вектора x , может быть определен посредством выдачи значения степени принадлежности $\mu_i(x)$, которой удовлетворяет антецедент соот-

ветствующего правила. Окончательно при использовании нечеткого вывода по Сугэно каждому правилу (выходному элементу) будет поставлено в соответствие четкое значение. Исходя из этого определяется следующее выходное значение:

$$\sum_{(x, y) \in S} \sum_{i=1}^n \mu_i(x) y / \sum_{(x, y) \in S} \mu_i(x), \quad (11.37)$$

где S – множество известных входных-выходных наборов данных; i – индексы правил, $i = 1, \dots, n$.

11.2.5.3. Другой подход к реализации нечеткой кластеризации заключается в нечеткой типовой модификации нейронной сети Кохонена [32].

11.2.5.4. Коско [27] использовал иной подход для генерации «ЕСЛИ–ТО»-правил из существующих данных. Он показал, что нечеткие множества могут рассматриваться как точки в многомерном гиперкубе. Это дает возможность применять *нечеткую ассоциативную память* (*fuzzy associative memory – FAM*) для представления нечетких правил.

Примечание. Хотя не совсем верно называть такие ассоциации правилами, так как теоретико-множественный подход предъявляет повышенные требования к определению многомерных нечетких множеств и операций над ними.

Рассмотрим простейшую FAM, кодирующую так называемые FAM-правила типа «ЕСЛИ A_i , ТО B_i », или *нечеткие ассоциации* (A_i, B_i), которые соотносят r -мерное нечеткое множество B_i с n -мерным множеством A_i .

Все правила FAM (A_1, B_1), ..., (A_n, B_n) работают параллельно. При предъявлении на входы FAM нечеткого множества A' каждое правило (A_i, B_i) отображает A' в B'_i , частично активизируя B_i . Чем больше сходство A' с конкретным A_i , тем больше сходство B'_i с B_i . Выходное нечеткое множество B образуется из частично активированных нечетких множеств B'_1, \dots, B'_m путем их взвешенного агрегирования:

$$B' = \sum_{i=1}^n w_i B'_i, \quad (11.38)$$

где w_i – отражает частоту или силу нечеткой ассоциации (A_i, B_i).

Обычно выполняется дефазификация сигнала B , например центроидным методом с получением числового выходного значения y . На рис. 11.13 показан пример структуры такой нечеткой ассоциативной памяти.

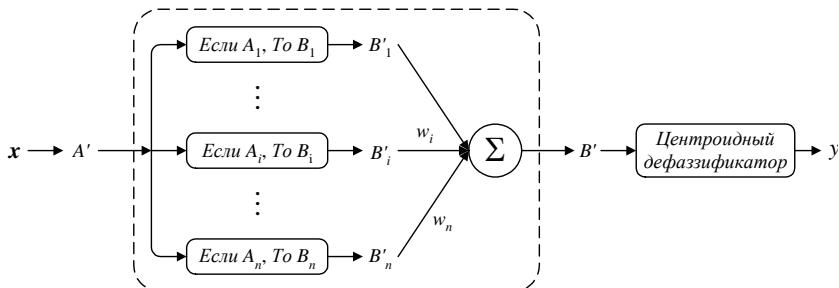


Рис. 11.13. Пример структуры нечеткой ассоциативной памяти Коско

Нечеткая ассоциативная память аппроксимирует функцию, покрывая ее графом в виде перекрывающихся нечетких «пятен» (patches). При аппроксимации уточняется не только увеличение числа нечетких пятен, но и уменьшение их размера. На рис. 11.14 показано, как нечеткие пятна в области входных-выходных значений $X \times Y$ покрывают вещественную функцию $f: X \rightarrow Y$. На рис. 11.14, *a* несколько больших пятен аппроксимируют f , а на рис. 11.14, *б* показаны пятна меньших размеров, которые существенно лучше аппроксимируют f . Ценаю за такое улучшение аппроксимации является увеличение количества небольших пятен (правил) и усложнение самого процесса.

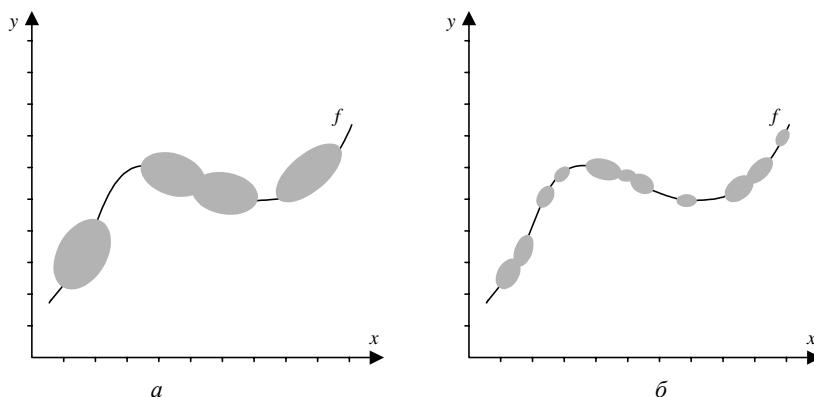


Рис. 11.14. Иллюстрация аппроксимации функции с помощью нечеткой ассоциативной памяти Коско

Каждое FAM-правило определяет нечеткое отношение между входными и выходными значениями (термами) для аппроксимируемой функции f . На рис. 11.15 нечеткое правило «ЕСЛИ X есть ZE , ТО Y есть PS »

показано как декартово произведение $NS \times PS$ нечетких множеств NS (*Zero – нейтральный*) и PS (*Positive Small – положительный малый*). В табл. 11.1 приведено обозначение используемых нечетких множеств.

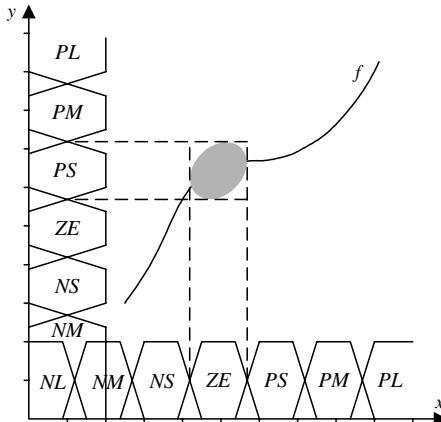


Рис. 11.15. Иллюстрация задания FAM-правила

Таблица 11.1. Обозначение нечетких множеств

Обозначение	Название (англ.)	Название (рус.)
NL	Negative Large	Отрицательный большой
NM	Negative Medium	Отрицательный средний
NS	Negative Small	Отрицательный малый
ZE	Zero	Нейтральный
PS	Positive Small	Положительный малый
PM	Positive Medium	Положительный средний
PL	Positive Large	Положительный большой

В случае же МИМО-структур FAM ($(A^1_i, \dots, A^m_i), (B^1_i, \dots, B^p_i)$) ассоциируют множественный выход B^1_i, \dots, B^p_i с множественным входом A^1_i, \dots, A^m_i . При этом нечеткая ассоциация интерпретируется как FAM-правило следующего вида:

$$\begin{aligned} &\text{ЕСЛИ } X^1 \text{ есть } A^1_i \text{ И } \dots \text{ И } X^m \text{ есть } A^m_i, \\ &\text{ТО } Y^1 \text{ есть } B^1_i \text{ И } \dots \text{ И } Y^p \text{ есть } B^p_i. \end{aligned} \quad (11.39)$$

Одно из основных отличий FAM от нечетких производственных сетей (помимо трактовки FAM в теоретико-множественном смысле) заключается в способе агрегирования консеквентов сработавших правил в нечет-

ком выводе. В отличие от использования общепринятых S-норм, таких, как, например, максимум, Коско предложил использовать алгебраическое сложение соответствующих нечетких множеств. Применение операции суммирования для агрегирования консеквентов позволяет повысить чувствительность нечеткой системы к небольшим изменениям значений входа. Кроме того, как показано в [33], огибающая консеквентов на основе операции максимума имеет тенденцию к равномерному распределению (в пределе – при большом числе наложенных правил), тогда как агрегирование суммированием приводит к симметричному унимодальному распределению выходного значения. Результирующий центроид после дефазификации центроидным методом при нескольких наложенных консеквентах FAM не выходит за пределы крайних центроидов, тогда как в случае агрегирования максимумом это вовсе не так.

11.2.5.5. Разновидность нечеткой ассоциативной памяти – адаптивная нечеткая ассоциативная память (adaptive fuzzy associative memory – AFAM) [27] использует специальные методы адаптивной кластеризации для обучения. К таким можно отнести, например, методы нечеткой самоорганизации C-means, пикового группирования, разностного группирования, нечеткой самоорганизации Густавсона–Кесселя. Используя данные методы, можно оперативно корректировать набор и параметры и веса нечетких правил (нечетких ассоциаций), фактически решая задачу адаптивного извлечения нечетких правил из числовых данных. В некоторых случаях можно использовать методы обучения с учителем, если имеется дополнительная информация для точной оценки ошибки генерации и адаптации правил.

11.2.5.6. В [28] в результате решения задачи классификации из данных извлекаются нечеткие правила с изменяемыми нечеткими областями (hyperboxes). Этот подход принципиально применим для решения задач, характеризующихся большой размерностью пространства входных переменных. Однако должна быть решена проблема перекрытия (overlap) областей различных классов посредством динамического расширения, разделения и сокращения областей. Вместе с тем представленный подход достаточно сложно реализовать для случая нескольких перекрывающихся классов.

11.2.5.7. Абе (Abe) и Лэн (Lan) [34] предложили свой метод извлечения нечетких правил для классификации образов. Нечеткие правила с нечеткими областями определяются посредством областей активизации, которые задают область «существования» данных для соответствующего класса, и области «запрещения», которые запрещают существование дан-

ных для этого класса. Эти данные извлекаются непосредственно из числовых данных посредством рекурсивного уточнения перекрытий между парами классов.

11.2.5.8. Эти же авторы в [35] предложили метод извлечения нечетких правил непосредственно из числовых данных для решения задачи аппроксимации функции. Суть этого метода заключается в следующем.

Пусть неизвестная функция характеризуется одномерным выходом y и m -мерным входным вектором x . Область, на которой определена переменная y , разделяется на n интервалов:

$$[y_0, y_1], (y_1, y_2], \dots, (y_{n-1}, y_n], \quad (11.40)$$

где $y_0 = M_1$, $y_n = M_2$. Назовем i -й интервал $(y_{i-1}, y_i]$ выходным интервалом i .

Используя заданные входные данные, для которых выходы находятся в выходном интервале i , рекурсивно определяется область входных значений, соответствующая выходному интервалу i .

А именно прежде всего находятся области активизации, которые определяют входную область, соответствующую выходному интервалу i , посредством вычисления минимального и максимального значений входных данных для каждого выходного интервала.

Если область активизации для выходного интервала i перекрывается с областью активизации для выходного интервала j , то область перекрытия определяется как область запрещения.

Если входные данные для выходных интервалов i и/или j находятся внутри области запрещения, то определяются одна или две дополнительные области активизации. В дальнейшем если области активизации определены и также перекрываются, то далее определяется дополнительная область запрещения. Данный процесс повторяется до тех пор, пока проблема наложения областей не будет решена. Рис. 11.16 иллюстрирует этот процесс.

Нечеткие правила определяются на основе областей активизации либо на основе областей активизации и соответствующих областей запрещения (если они сгенерированы).

На рис. 11.17 показана обобщенная структура гибридной нейронной нечеткой сети, вычисляющая степени принадлежности входных данных к соответствующим выходным интервалам с последующим приведением к четкому значению. Так, для входного вектора x степень принадлежности к выходным интервалам от 1-го до n -й определяется данной сетью и затем на основе полученных степеней принадлежности вычисляется четкое значение выхода y .

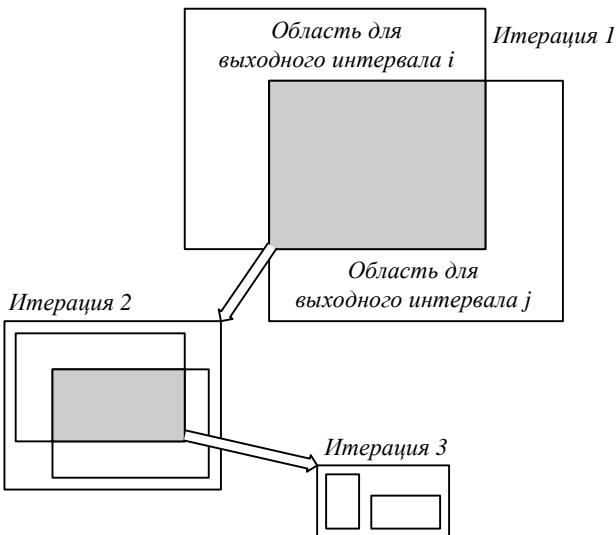


Рис. 11.16. Рекурсивное определение областей активизации и запрещения

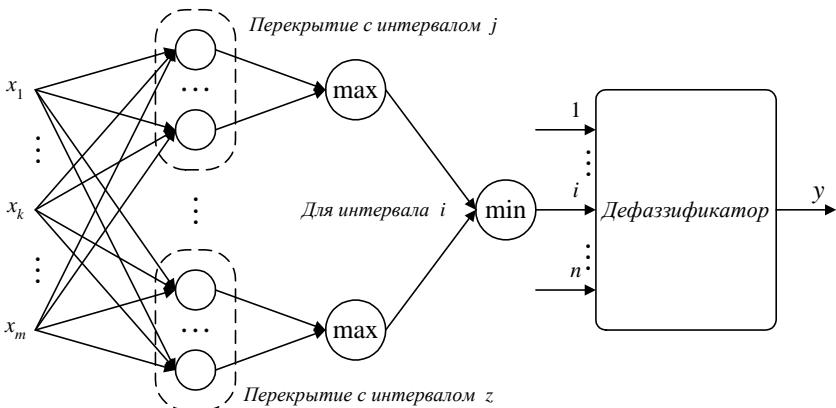


Рис. 11.17. Структура гибридной нейронной нечеткой сети Абе (Abe) и Лэна (Lan) для извлечения нечетких правил из данных при решении задачи аппроксимации функции

Данная сеть состоит из четырех слоев. При этом различные выходные интервалы реализуются разными элементами слоев 2–4. Между элементами этих слоев, реализующими различные выходные интервалы переменной y , отсутствует какая-либо связь.

Слой 2. Элементы этого слоя реализуют нечеткие «ЕСЛИ–ТО»–правила, вычисляющие степени принадлежности переменных входного вектора x .

Слой 3. Элементы слоя вычисляют максимальные значения выходов элементов *слоя 2*, которые представляют собой степени принадлежности, полученные в результате ликвидации перекрытий между парой выходных интервалов. Число элементов *слоя 3* для выходного интервала i определяется по числу выходных интервалов, чьи входные пространства перекрываются со входным пространством данного выходного интервала i . Поэтому, если нет перекрытия между входными пространствами выходного интервала i и какого-либо другого выходного интервала, то сеть для выходного интервала i уменьшается до двух слоев.

Слой 4. Элемент этого слоя для выходного интервала i определяется минимальное значение среди максимальных значений, определенных в предыдущем слое и соответствующих перекрытию между двумя выходными интервалами. Поэтому если выходной интервал i перекрывается только с одним выходным интервалом, то сеть для выходного интервала i уменьшается до трех слоев. Вычисление минимума в *слое 4* позволяет ликвидировать перекрытия среди более чем двух выходных интервалов. Таким образом, в процессе образования входных областей необходимо избавиться от перекрытия между двумя интервалами одновременно.

11.2.5.9. Другой, не менее эффективный по сравнению с выше рассмотренными способом извлечения нечетких правил из данных предложен в книге [36].

Предположим, что исследуемый объект имеет n входов (иначе, векторный вход x) и один выход y и имеет «истинное» (неизвестное) описание:

$$y=f(x) + e, \quad (11.41)$$

где $f(x)$ – функция неизвестного вида; e – случайная аддитивная помеха (отражающая действие неучитываемых системой факторов) с нулевым средним значением и произвольным (неизвестным) распределением на интервале $(-\varepsilon_m, \varepsilon_m)$.

Предположим далее, что на объекте может быть реализован эксперимент, заключающийся в регистрации N пар значений $\langle x^{(k)}, y^{(k)} \rangle$, $k = 1, \dots, K$, при этом величины (векторы) x_k измеряются без ошибок; значение K при необходимости допускает модификацию.

Алгоритм построения системы может быть теперь описан следующим образом.

Этап 1. Из n ($n < K$) произвольных значений $\langle \mathbf{x}_k, y_k \rangle$ составляется начальная база знаний модели, отображаемая матрицей $\mathbf{U}_{n \times (m+1)}$ со строками вида $\langle \mathbf{x}^{(k)}, y^{(k)} \rangle = \langle x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)} \rangle$, $k = 1, \dots, n$.

Такое представление, очевидно, соответствует набору нечетких производственных правил вида

$$\begin{aligned} \Pi_k: \text{ЕСЛИ } x_1 \text{ есть } A_{k1} \text{ И } x_2 \text{ есть } A_{k2} \text{ И } \dots \text{ И } x_m \text{ есть } A_{km}, \\ \text{ТО } y = B_k, k = 1, \dots, n. \end{aligned} \quad (11.42)$$

Этап 2. Для каждой новой экспериментальной точки $\langle \mathbf{x}^{(l)}, y^{(l)} \rangle$ рассчитывается прогнозируемое значение по формуле, соответствующей рассмотренному ранее центроидному методу дефазификации:

$$y' = \frac{\sum_{k=1}^n y^{(k)} \varphi(\|\mathbf{x}^{(l)} - \mathbf{x}^{(k)}\|)}{\sum_{k=1}^n \varphi(\|\mathbf{x}^{(l)} - \mathbf{x}^{(k)}\|)}, \quad (11.43)$$

где $\varphi(\bullet)$ – функция колоколообразной или экспоненциальной формы:

$$\varphi(\|\mathbf{x}^{(l)} - \mathbf{x}^{(k)}\|) = \exp\left(-\lambda \sum_{i=1}^m |x_i^{(l)} - x_i^{(k)}|\right), \quad (11.44)$$

где λ – параметр функции.

Этап 3. Проверяется неравенство

$$|y^{(l)} - y'| > d, \quad (11.45)$$

где d – заданная константа, определяющая погрешность аппроксимации.

При выполнении неравенства база нечетких правил пополняется путем расширения матрицы \mathbf{U} (добавлением строки $\langle \mathbf{x}^{(l)}, y^{(l)} \rangle$). В противном случае матрица \mathbf{U} остается без изменений.

Этап 4. Проверяется правило останова. В данном случае процесс извлечения правил считается завершенным, если в соответствии с этапами 2 и 3 перебраны все K экспериментальных данных (без учета значений начальной базы).

Если не все экспериментальные точки использованы, то осуществляется переход к этапу 2, в противном случае – останов.

В процессе реализации алгоритма матрица \mathbf{U} , а также параметры λ и d считаются априорно заданными.

Очевидно, что база нечетких правил не остается фиксированной, а модернизируется по мере поступления экспериментальных данных. Причем непротиворечивость нового продукционного правила относительно набора правил из базы гарантируется предложенной процедурой ее пополнения.

11.2.5.10. Еще одним подходом к извлечению нечетких правил из данных является *алгоритм постепенно возрастающего разбиения* (Incremental Decomposition Algorithm) [37]. Идея этого подхода заключается в следующем. На первой итерации формируется одно нечеткое продукционное правило, имеющее в качестве области, в которой степень активизации предпосылки нечеткого правила превышает заданное значение, все множество допустимых входных значений (рис. 11.18):

$$\Pi_1: \text{ЕСЛИ } x_1 \text{ есть } A_{11} \text{ И } x_2 \text{ есть } A_{22}, \text{ ТО } y = B_1. \quad (11.46)$$

На итерации 2 данное правило разбивается на два, двумя способами. Проводится обучение и выбирается, какой из способов разбиения дает наименьшую погрешность (данный переход отмечен стрелкой). При этом формируется два правила:

$$\begin{aligned} \Pi_1: & \text{ЕСЛИ } x_1 \text{ есть } A_{11} \text{ И } x_2 \text{ есть } A_{21}, \text{ ТО } y = B_1, \\ \Pi_2: & \text{ЕСЛИ } x_1 \text{ есть } A_{11} \text{ И } x_2 \text{ есть } A_{22}, \text{ ТО } y = B_2. \end{aligned} \quad (11.47)$$

Среди имеющихся правил выбирается то, для которого составляющая погрешности в общей погрешности наибольшая (область его влияния выделена сплошным контуром). Оно и подлежит разбиению на два двумя способами (итерация 3). Описанный процесс продолжается до достижения требуемой точности или пока не будет сгенерировано заданное число продукционных правил.

Глава 12. ОБУЧЕНИЕ НЕЙРОННЫХ НЕЧЕТКИХ СЕТЕЙ

Задачи обучения могут решаться различным образом для нейронных нечетких сетей как одного, так и разных, рассмотренных в предыдущих главах, типов.

Следует отметить, что по характеру обучения на основе экспериментальных данных нейронные нечеткие сети можно классифицировать следующим образом:

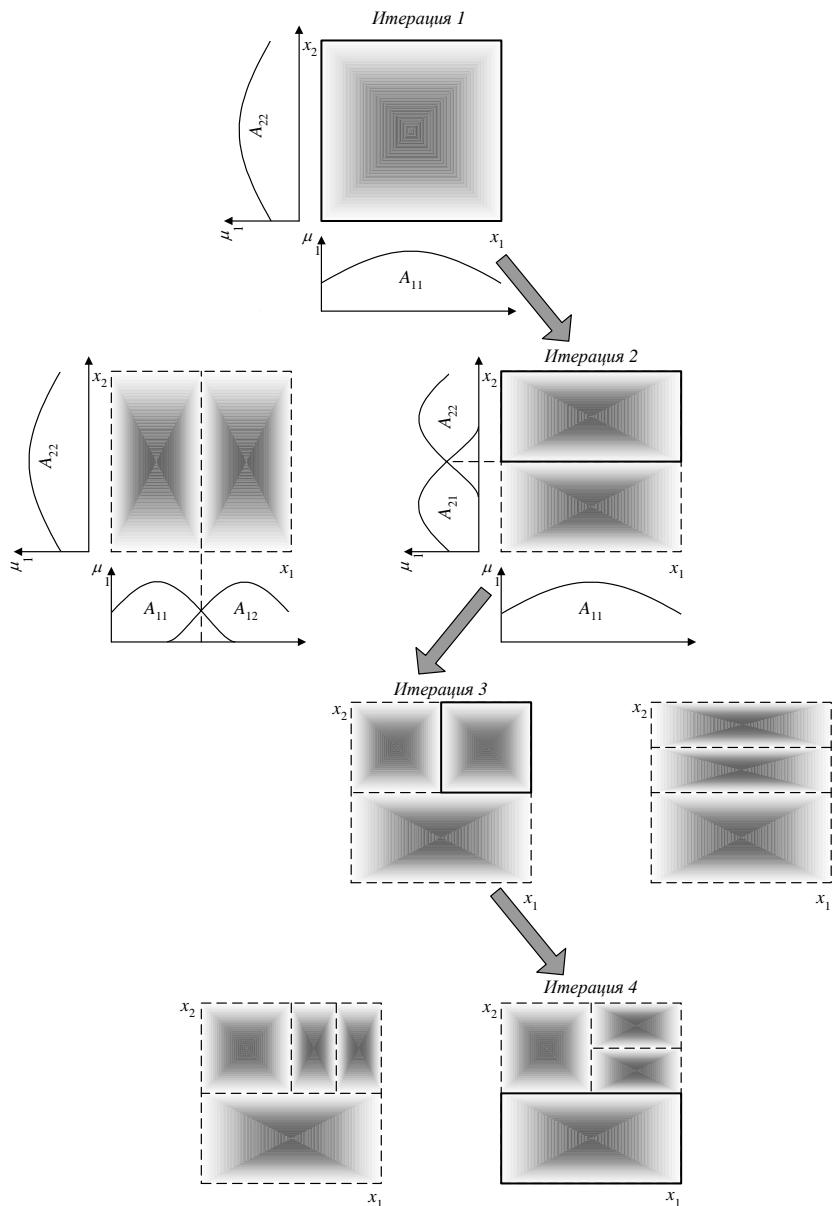


Рис. 11.18. Иллюстрация работы алгоритма Incremental Decomposition Algorithm

- с адаптацией структуры (и параметров) – самонастраивающиеся нейронные нечеткие сети;
- с жесткой структурой и адаптацией параметров сети – адаптивные нейронные нечеткие сети [38–40].

В свою очередь, последние подразделяются:

- по виду используемого алгоритма оптимизации:
 - на использующие детерминированные [41] алгоритмы типа градиентного (как вариант – обратного распространения ошибки) и комбинированные [25, 39];
 - на использующие алгоритмы стохастического типа [41], в том числе генетические алгоритмы [38, 42];
- по типу параметров адаптации:
 - с адаптацией параметров функций принадлежности [42, 43];
 - с адаптацией весов правил [44];
 - с адаптацией параметров оператора агрегирования [45].

Далее рассмотрим наиболее популярные подходы к обучению нейронных нечетких сетей, характеризующихся различными вариантами наделения нечеткости входов, выходов, весов.

12.1. Использование нейронных сетей с обучением с обратным распространением ошибки для реализации нечетких моделей

Следует отметить, что при реализации нечетких производственных моделей на основе традиционных нейронных сетей при решении задачи их обучения применимы подходы, характерные для типичных нейросетевых архитектур.

Пусть задана база нечетких правил типа

$$\Pi_k: \text{ЕСЛИ } x \text{ есть } A_k \text{ ТО } y = B_k, \quad k = 1, \dots, n. \quad (12.1)$$

где A_k и B_k – нечеткие числа.

Каждое правило (12.1) может быть интерпретировано как обучающий пример многослойной нейронной сети, где пункт антецедента правила является входом, а пункт консеквента – требуемым выходом.

Обучающая выборка, получаемая из (12.1), может быть записана в следующем виде:

$$\{(A_1, B_1), \dots, (A_n, B_n)\}. \quad (12.2)$$

Если рассматриваются нечеткие правила с MISO-структурой типа

$$\Pi_k: \text{ЕСЛИ } x_1 \text{ есть } A_{k1} \text{ И } \dots \text{ И } x_j \text{ есть } A_{kj} \text{ И } \dots \text{ И } x_m \text{ есть } A_{km}, \\ \text{ТО } y \text{ есть } B_k, k = 1, \dots, n, \quad (12.3)$$

где A_{k1}, \dots, A_{km} , B_k – нечеткие числа, то обучающая выборка, получаемая из (12.3), записывается в виде

$$\{(A_{11}, \dots, A_{1m}, B_1), \dots, (A_{n1}, \dots, A_{nm}, B_n)\}. \quad (12.4)$$

Для правил с MIMO-структурой типа

$$\Pi_k: \text{ЕСЛИ } x_1 \text{ есть } A_{k1} \text{ И } \dots \text{ И } x_m \text{ есть } A_{km}, \\ \text{ТО } y_1 \text{ есть } B_{k1} \text{ И } \dots \text{ И } y_p \text{ есть } B_{kp}, k = 1, \dots, n, \quad (12.5)$$

где A_{k1}, \dots, A_{km} , B_{k1}, \dots, B_{kp} – нечеткие числа, обучающая выборка представляется в следующем виде:

$$\{(A_{11}, \dots, A_{1m}, B_{11}, \dots, B_{1p}), \dots, (A_{n1}, \dots, A_{nm}, B_{n1}, \dots, B_{np})\}. \quad (12.6)$$

Существует два основных подхода к обучению нейронных сетей, реализующих нечеткие модели, на основе обучения с обратным распространением ошибки [17].

Первый подход, предложенный Умано (Umano) и Езава (Ezawa) [46], основывается на представлении примеров обучающей выборки значениями их функций принадлежности.

Проиллюстрируем этот подход на основе правил типа (12.1). Пусть $[x^{(\min)}, x^{(\max)}]$ содержит значения из базового множества для всех A_k . Также пусть $[y^{(\min)}, y^{(\max)}]$ содержит значения из базового множества для всех B_k , $k = 1, \dots, n$.

Дискретизированная версия обучающей выборки нейронной сети состоит из следующих наборов входных-выходных значений:

$$\{(A_k(x^{(1)}), \dots, A_k(x^{(p)})), B_k(y^{(1)}), \dots, B_k(y^{(q)})\}, k = 1, \dots, n. \quad (12.7)$$

На рис. 12.1 показан пример представления наборов этих входных-выходных значений (значений степеней принадлежности нечетких чисел) относительно нейронной сети.

Число входов и выходов такой нейронной сети зависит от степени дискретизации $[x^{(\min)}, x^{(\max)}]$ и $[y^{(\min)}, y^{(\max)}]$.

Второй подход, предложенный Юхара (Uehara) и Фуджизи Fujise [47], основывается на представлении примеров обучающей выборки множествами α -уровней нечетких чисел.

Пусть $[A_k]^{\alpha_i}$ обозначает множество α_i -уровня нечетких чисел A_k , используемых, например, в правилах типа (12.1), а $[B_k]^{\alpha_i}$ – множество α_i -уровня нечетких чисел B_k , $i = 1, \dots, p$:

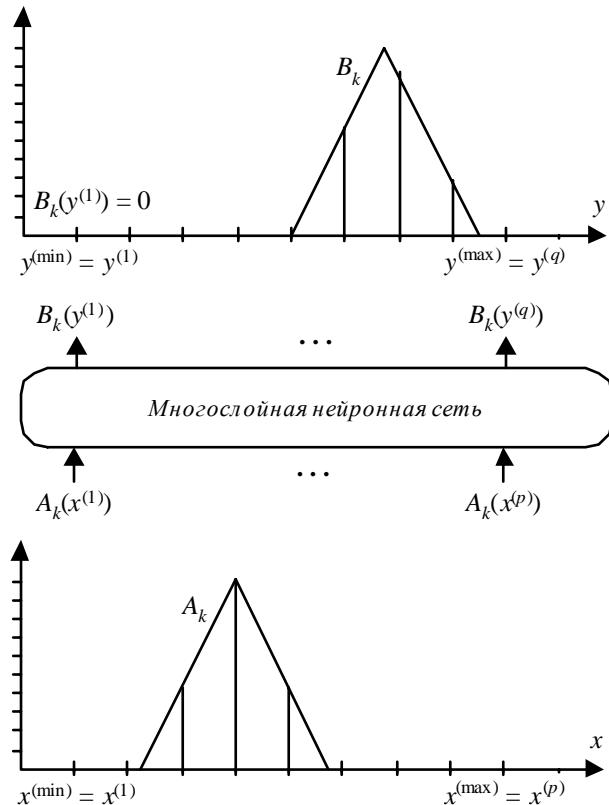


Рис. 12.1. Пример представления значений степеней принадлежности нечетких чисел относительно нейронной сети

$$\begin{aligned} [A_k]^{a_i} &= \{x \mid A_k(x) \geq a_i\} = [a_{ki}^L, a_{ki}^R], \\ [B_k]^{a_i} &= \{y \mid B_k(y) \geq a_i\} = [b_{ki}^L, b_{ki}^R]. \end{aligned} \quad (12.8)$$

Тогда дискретизированная версия обучающей выборки нейронной сети состоит из следующих наборов входных-выходных значений:

$$\{(a_{k1}^L, a_{k1}^R, \dots, a_{kp}^L, a_{kp}^R), (b_{k1}^L, b_{k1}^R, \dots, b_{kp}^L, b_{kp}^R)\}, \quad k = 1, \dots, n. \quad (12.9)$$

На рис. 12.2 показан пример представления наборов этих входных-выходных значений (множествами α -уровней нечетких чисел) относительно нейронной сети.

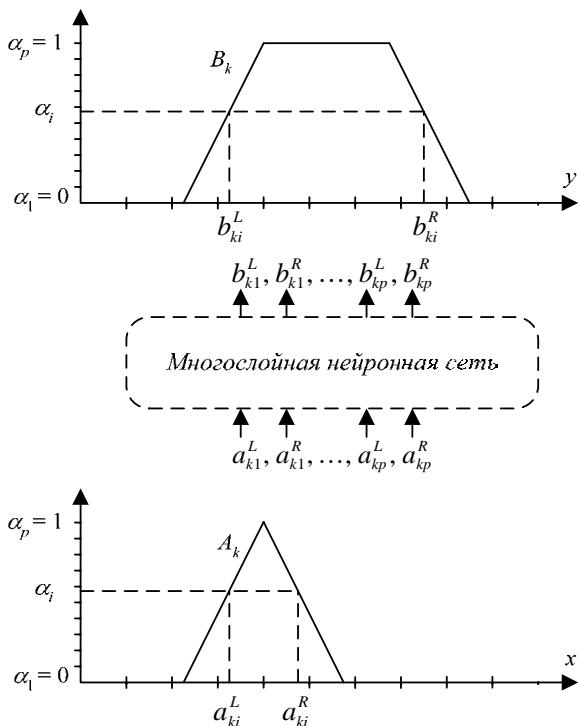


Рис. 12.2. Пример представления значений наборов α -уровней нечетких чисел относительно нейронной сети

Число входов и выходов такой нейронной сети зависит от задаваемого количества множеств α -уровней.

12.2. Обучение нейронных нечетких сетей типа 2 с нечеткими входами и выходами и четкими весами

Ишибучи (Ishibuchi), Квон (Kwon) и Танака (Tanaka) [48] предложили подход к использованию нечетких «ЕСЛИ–ТО»–правил для обучения нейронных сетей *типа 2* (см. табл. 10.1) с нечеткими входами и выходами и четкими весами на основе нечетких обучающих образов.

Пусть заданы нечеткие правила типа (12.3):

П_k: ЕСЛИ x_1 есть A_{k1} И ... И x_j есть A_{kj} И ... И x_m есть A_{km} ,
ТО у есть B_k , $k = 1, \dots, n$,

где $A_{k1}, \dots, A_{km}, B_k$ – нечеткие числа.

Из этих правил могут быть получены следующие обучающие образы:

$$\{(A_1, B_1), \dots, (A_n, B_n)\}, \quad (12.10)$$

где $A_k = (A_{k1}, \dots, A_{km})$ обозначает антецедент, а B_k – консеквент k -го нечеткого правила.

На рис. 12.3 представлен пример структуры нейронной нечеткой сети *типа 2* с нечеткими входами и выходами и четкими весами.

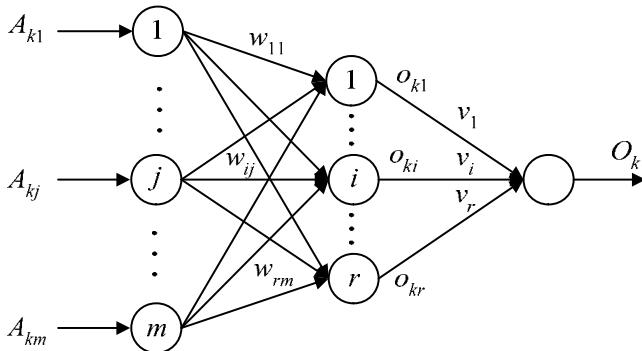


Рис. 12.3. Пример структуры нейронной нечеткой сети *типа 2* с нечеткими входами и выходами и четкими весами

В работе [49] предложен модифицированный алгоритм обучения с обратным распространением ошибки для нейронных нечетких сетей данного типа.

Пусть $A_k = (A_{k1}, \dots, A_{km})$ – k -й обучающий образ. Выход i -го элемента скрытого слоя сети вычисляется следующим образом:

$$o_{ki} = f\left(\sum_{j=1}^m w_{ij} A_{kj}\right). \quad (12.11)$$

Для выходного элемента

$$O_k = f\left(\sum_{i=1}^r v_i o_{ki}\right), \quad (12.12)$$

где $f(z) = 1/(1 + \exp(-z))$ – униполярная функция преобразования.

Следует отметить, что отношение вход-выход каждого элемента (нейрона) такой нейронной нечеткой сети определяется на основе принципа обобщения. На рис. 12.4 показан пример задания такого отношения.

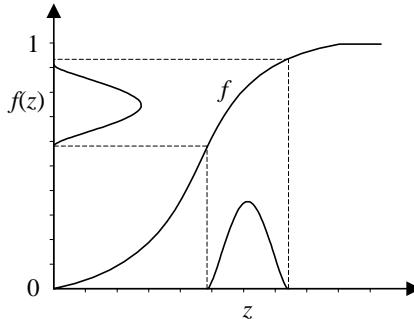


Рис. 12.4. Пример задания отношения вход-выход каждого нейрона нейронной нечеткой сети *типа 2*

Обозначим множества α -уровней вычисляемого выхода O_k следующим образом:

$$[O_k]^\alpha = [O_k^L(\alpha), O_k^R(\alpha)], \quad \alpha \in [0, 1], \quad (12.13)$$

где $O_k^L(\alpha), O_k^R(\alpha)$ – соответственно значения левой и правой границ множеств α -уровней вычисляемого выхода O_k .

Так как функция f является строго монотонно возрастающей, то справедливым является следующее:

$$\begin{aligned} [O_k]^\alpha &= \left[f\left(\sum_{i=1}^r v_i o_{ki}\right) \right]^\alpha = \\ &= \left[f\left(\sum_{i=1}^r [v_i o_{ki}]^L(\alpha)\right), f\left(\sum_{i=1}^r [v_i o_{ki}]^R(\alpha)\right) \right], \end{aligned} \quad (12.14)$$

где

$$\begin{aligned} [o_{ki}]^\alpha &= \left[f\left(\sum_{j=1}^m w_{ij} A_{kj}\right) \right]^\alpha = \\ &= \left[f\left(\sum_{j=1}^m [w_{ij} A_{kj}]^L(\alpha)\right), f\left(\sum_{j=1}^m [w_{ij} A_{kj}]^R(\alpha)\right) \right]. \end{aligned} \quad (12.15)$$

Множества α -уровней целевого выхода B_k обозначаются как

$$[B_k]^\alpha = [B_k^L(\alpha), B_k^R(\alpha)], \quad \alpha \in [0, 1], \quad (12.16)$$

где $B_k^L(\alpha), B_k^R(\alpha)$ – соответственно значения левой и правой границ множеств α -уровней целевого выхода B_k .

Минимизируемая целевая функция (функция ошибки) для каждого множества α -уровня определяется следующим образом:

$$\varepsilon_k(\alpha) := \varepsilon_k^L(\alpha) + \varepsilon_k^R(\alpha), \quad (12.17)$$

где

$$\varepsilon_k^L(\alpha) = \frac{1}{2} \left(B_k^L(\alpha) - O_k^L(\alpha) \right)^2, \quad (12.18)$$

$$\varepsilon_k^R(\alpha) = \frac{1}{2} \left(B_k^R(\alpha) - O_k^R(\alpha) \right)^2, \quad (12.19)$$

т. е. $\varepsilon_k^L(\alpha)$ – ошибка между левыми границами множеств α -уровней целевого и вычисленного значений выхода; $\varepsilon_k^R(\alpha)$ – ошибка между правыми границами множеств α -уровней целевого и вычисленного значений выхода.

Затем формируется функция ошибки для k -го обучающего образа:

$$\varepsilon_k = \sum_{\alpha} \alpha \varepsilon_k(\alpha). \quad (12.20)$$

Теоретически целевая функция удовлетворяет следующему условию:

$$\varepsilon_k \rightarrow 0 \text{ тогда и только тогда, когда } O_k \rightarrow B_k. \quad (12.21)$$

Для данной целевой функции могут быть получены следующие правила для обучения нейронной нечеткой сети:

$$v_i := v_i - \eta \alpha \frac{\partial \varepsilon_k(\alpha)}{\partial v_i}, \quad i = 1, \dots, r. \quad (12.22)$$

$$w_{ij} := w_{ij} - \eta \alpha \frac{\partial \varepsilon_k(\alpha)}{\partial w_{ij}}, \quad i = 1, \dots, r, j = 1, \dots, m, \quad (12.23)$$

где η – показатель, задающий скорость обучения сети, $\eta \in [0, 1]$.

12.3. Обучение нейронных нечетких сетей типа 3 с нечеткими входами, выходами и весами

Рассмотрим вопросы обучения нейронных нечетких сетей *типа 3* (см. табл. 10.1) с нечеткими входами, выходами и весами на примере сети, представленной в работе [11].

Пусть заданы нечеткие правила типа (12.3). Из этих правил могут быть получены обучающие образы типа (12.9).

Выход i -го элемента скрытого слоя сети вычисляется так:

$$o_{ki} = f \left(\sum_{j=1}^m W_{ij} A_{kj} \right); \quad (12.24)$$

элемент выходного слоя формирует значение

$$O_k = f \left(\sum_{i=1}^r V_i o_{ki} \right), \quad (12.25)$$

где $f(z)$ – унипольярная передаточная функция, например вида $f(z) = 1/(1 + \exp(-z))$; W_{ij} , V_i – нечеткие веса элементов скрытого и выходного слоев соответственно; A_{kj} – нечеткий вход.

На рис. 12.5 представлен пример структуры нейронной нечеткой сети *типа 3* с нечеткими входами, выходами и весами.

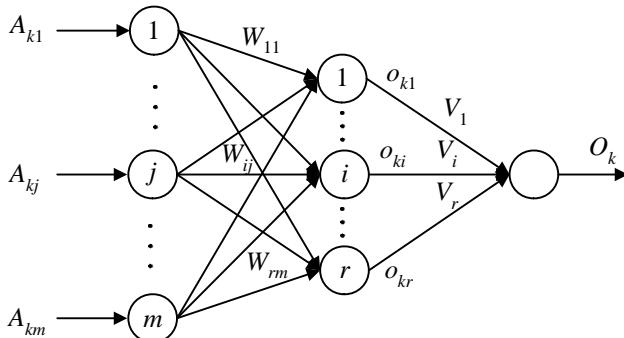


Рис. 12.5. Пример структуры нейронной нечеткой сети *типа 3* с нечеткими входами, выходами и весами

Нечеткие выходы каждого элемента сети вычисляются в виде множеств α -уровней относительно нечетких входов и весов.

Обозначим множества α -уровней вычисляемого выхода O_k , целевого выхода B_k , весов выходного элемента V_i и весов элементов скрытого слоя W_{ij} соответственно как

$$\begin{aligned} [O_k]^\alpha &= [O_k^L(\alpha), O_k^R(\alpha)], \quad [B_k]^\alpha = [B_k^L(\alpha), B_k^R(\alpha)], \\ [V_i]^\alpha &= [V_i^L(\alpha), V_i^R(\alpha)], \quad [W_{ij}]^\alpha = [W_{ij}^L(\alpha), W_{ij}^R(\alpha)], \end{aligned} \quad (12.26)$$

для $\alpha \in [0, 1]$, $i = 1, \dots, r$, $j = 1, \dots, m$.

Так как функция f является строго монотонно возрастающей, то справедливыми являются следующие выражения:

$$\begin{aligned} [O_k]^\alpha &= \left[f\left(\sum_{i=1}^r V_i o_{ki}\right) \right]^\alpha = \\ &= \left[f\left(\sum_{i=1}^r [V_i o_{ki}]^L(\alpha)\right), f\left(\sum_{i=1}^r [V_i o_{ki}]^R(\alpha)\right) \right], \end{aligned} \quad (12.27)$$

где

$$\begin{aligned} [o_{ki}]^\alpha &= \left[f\left(\sum_{j=1}^m W_{ij} A_{kj}\right) \right]^\alpha = \\ &= \left[f\left(\sum_{j=1}^m [W_{ij} A_{kj}]^L(\alpha)\right), f\left(\sum_{j=1}^m [W_{ij} A_{kj}]^R(\alpha)\right) \right]. \end{aligned} \quad (12.28)$$

Минимизируемая целевая функция определяется для каждого множества α -уровня следующим образом:

$$\varepsilon_k(\alpha) := \varepsilon_k^L(\alpha) + \varepsilon_k^R(\alpha), \quad (12.29)$$

где

$$\varepsilon_k^L(\alpha) = \frac{1}{2} (B_k^L(\alpha) - O_k^L(\alpha))^2, \quad (12.30)$$

$$\varepsilon_k^R(\alpha) = \frac{1}{2} (B_k^R(\alpha) - O_k^R(\alpha))^2, \quad (12.31)$$

т. е. $\varepsilon_k^L(\alpha)$ обозначает ошибку между левыми границами множеств α -уровней целевого и вычисленного значений выхода, а $\varepsilon_k^R(\alpha)$ обозначает ошибку между правыми границами множеств α -уровней целевого и вычисленного значений выхода.

Далее формируется функция ошибки для k -го обучающего образа:

$$\varepsilon_k = \sum_\alpha \alpha \varepsilon_k(\alpha). \quad (12.32)$$

Рассмотрим алгоритм обучения сети данного типа для функции ошибки $\varepsilon_k(\alpha)$.

В предположении, что нечеткие веса нейронов скрытого слоя имеют симметричную треугольную форму, они могут быть представлены тремя параметрами: $W_{ij} = (w_{ij}^{(1)}, w_{ij}^{(2)}, w_{ij}^{(3)})$, определяющими соответственно левую границу, центр и правую границу W_{ij} . Аналогично нечеткие веса выходного нейрона могут быть представлены следующими тремя параметрами: $V_i = (v_i^{(1)}, v_i^{(2)}, v_i^{(3)})$, определяющими соответственно левую границу, центр и правую границу V_i .

Из симметричности W_{ij} и V_i следует, что

$$w_{ij}^{(2)} = \frac{w_{ij}^{(1)} + w_{ij}^{(3)}}{2}, v_i^{(2)} = \frac{v_i^{(1)} + v_i^{(3)}}{2}, i = 1, \dots, r, j = 1, \dots, m. \quad (12.33)$$

Изменение весов относительно целевой функции $\varepsilon_k(\alpha)$ формируется следующим образом:

$$\begin{aligned} \Delta w_{ij}^{(1)}(t) &= -\eta \frac{\partial \varepsilon_p(\alpha)}{\partial w_{ij}^{(1)}} + \beta \Delta w_{ij}^{(1)}(t-1), \\ \Delta w_{ij}^{(3)}(t) &= -\eta \frac{\partial \varepsilon_p(\alpha)}{\partial w_{ij}^{(3)}} + \beta \Delta w_{ij}^{(3)}(t-1), \end{aligned} \quad (12.34)$$

где η – коэффициент скорости обучения; β – импульсная константа; t – индекс текущего момента времени; $i = 1, \dots, r, j = 1, \dots, m$.

Аналогично вычисляются изменения весов:

$$\begin{aligned} \Delta w_{ij}^{(1)}(t) &= -\eta \frac{\partial \varepsilon_p(\alpha)}{\partial w_{ij}^{(1)}} + \beta \Delta w_{ij}^{(1)}(t-1), \\ \Delta w_{ij}^{(3)}(t) &= -\eta \frac{\partial \varepsilon_p(\alpha)}{\partial w_{ij}^{(3)}} + \beta \Delta w_{ij}^{(3)}(t-1), \end{aligned} \quad (12.35)$$

где η – коэффициент, задающий скорость обучения; β – импульсная константа; t – индекс, характеризующий текущий момент времени; $j = 1, \dots, m$.

Нечеткие веса $W_{ij} = (w_{ij}^{(1)}, w_{ij}^{(2)}, w_{ij}^{(3)})$ изменяются по правилам

$$\begin{aligned} w_{ij}^{(1)}(t+1) &= w_{ij}^{(1)}(t) + \Delta w_{ij}^{(1)}(t), \\ w_{ij}^{(3)}(t+1) &= w_{ij}^{(3)}(t) + \Delta w_{ij}^{(3)}(t), \\ w_{ij}^{(2)}(t+1) &= \frac{w_{ij}^{(1)}(t+1) + w_{ij}^{(3)}(t+1)}{2}. \end{aligned} \quad (12.36)$$

Нечеткие же веса $V_i = (v_i^{(1)}, v_i^{(2)}, v_i^{(3)})$ изменяются по правилам

$$\begin{aligned} v_i^{(1)}(t+1) &= v_i^{(1)}(t) + \Delta v_i^{(1)}(t), \\ v_i^{(3)}(t+1) &= v_i^{(3)}(t) + \Delta v_i^{(3)}(t), \\ v_i^{(2)}(t+1) &= \frac{v_i^{(1)}(t+1) + v_i^{(3)}(t+1)}{2}. \end{aligned} \quad (12.37)$$

После изменения веса W_{ij} может так случиться, что его левая граница станет больше, чем правая. В этом случае используются следующие простые эвристические правила:

$$\begin{aligned} w_{ij}^{(1)}(t+1) &:= \min\{w_{ij}^{(1)}(t+1), w_{ij}^{(3)}(t+1)\}, \\ w_{ij}^{(3)}(t+1) &:= \max\{w_{ij}^{(1)}(t+1), w_{ij}^{(3)}(t+1)\}. \end{aligned} \quad (12.38)$$

Подобные эвристики используются и для весов V_i :

$$\begin{aligned} v_i^{(1)}(t+1) &:= \min\{v_i^{(1)}(t+1), v_i^{(3)}(t+1)\}, \\ v_i^{(3)}(t+1) &:= \max\{v_i^{(1)}(t+1), v_i^{(3)}(t+1)\}. \end{aligned} \quad (12.39)$$

Вышеперечисленные действия выполняются для всех $k = 1, \dots, n$ обучающих образов:

$$\{(A_1, B_1), \dots, (A_n, B_n)\},$$

где $A_k = (A_{k1}, \dots, A_{km})$.

Полагается также, что для обучения нейронной нечеткой сети используются p значений множеств α -уровней.

Таким образом, алгоритм обучения нейронной нечеткой сети *типа 3* заключается в следующем.

Этап 1. Инициализация нечетких весов элементов сети заданием небольших случайных значений, текущая ошибка E устанавливается равной нулю, и выбирается пороговое значение (критерий обученности сети) $E_{\max} > 0$.

Этап 2. Этап 3 повторяется для $\alpha = \alpha_1, \alpha_2, \dots, \alpha_p$.

Этап 3. Для всех $A_k = (A_{k1}, \dots, A_{km})$, $k = 1, \dots, n$, в процессе распространения по сети вычисляется множество α -уровня нечеткого выходного вектора O_k . Нечеткие веса элементов сети изменяются с использованием функции ошибки $\varepsilon_k(\alpha)$.

Этап 4. Общая ошибка данного цикла обучения вычисляется суммированием текущей ошибки с E .

Этап 5. Выполняется цикл обучения. В случае $E < E_{\max}$ процесс обучения завершается. Если $E > E_{\max}$, то E устанавливается в 0 и инициируется новый цикл обучения с этапа 2 алгоритма.

Глава 13. ИСПОЛЬЗОВАНИЕ НЕЧЕТКИХ ПРОДУКЦИОННЫХ СЕТЕЙ В НЕЙРОННЫХ СЕТИЯХ

Наиболее типичным случаем использования нечетких продукционных сетей в нейронных сетях является оценка и задание с их помощью таких параметров нейронных сетей, как веса между нейронами и значения порогов нейронов. Это позволяет повысить скорость обучения ней-

ронных сетей и уменьшить ошибку моделирования. В качестве примера такого рода интеграции нейронных и нечетких производческих сетей можно привести CoActive Neuro-Fuzzy Inference System (CANFIS) [50].

Примеры структур CANFIS-сетей показаны на рис. 13.1–13.3. Типичная CANFIS-сеть с MISO-структурой включает в себя:

- простую нейронную сеть, обучаемую по алгоритму обратного распространения ошибки;
- ANFIS-сеть, реализующую алгоритм нечеткого вывода Такаги–Сугэно.

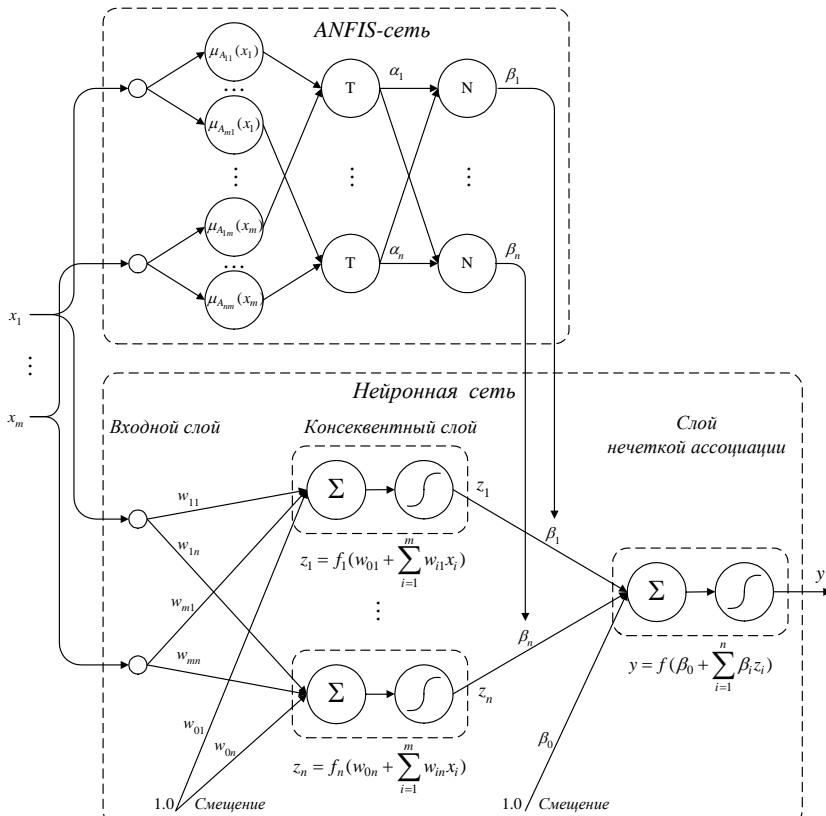


Рис. 13.1. Структура CANFIS-сети с одним выходом

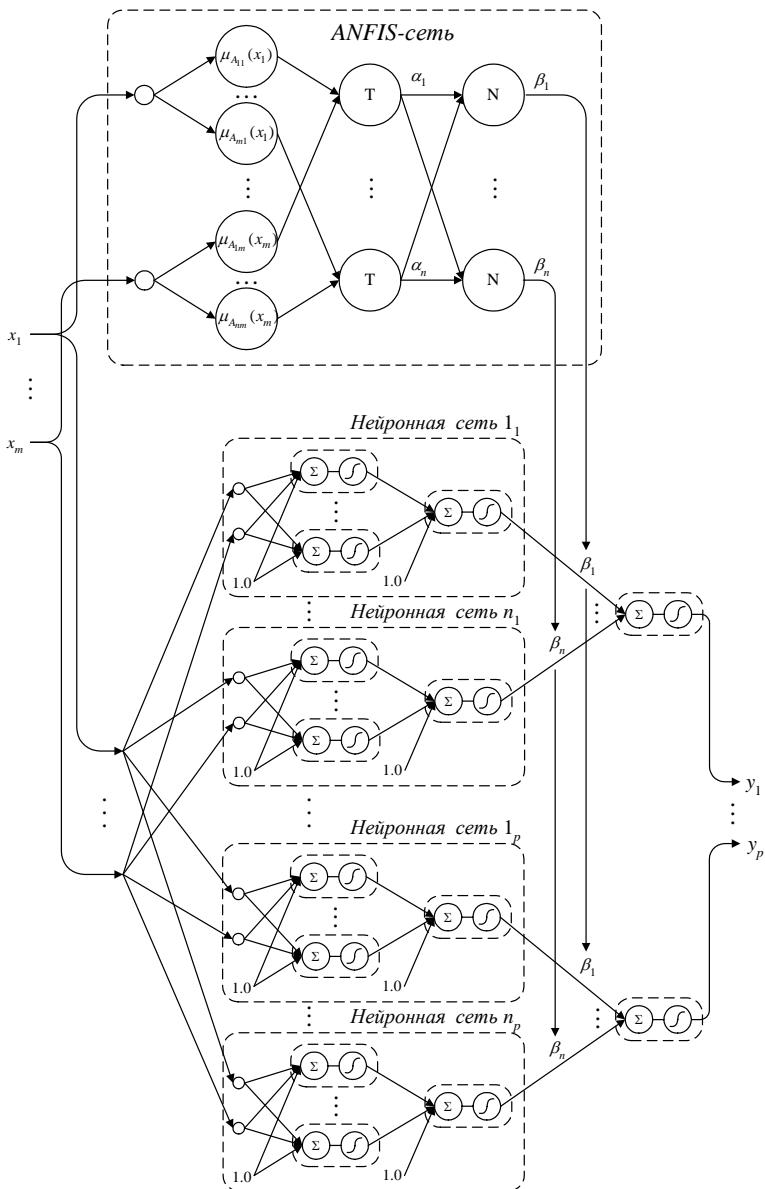


Рис. 13.2. Структура CANFIS-сети со многими выходами

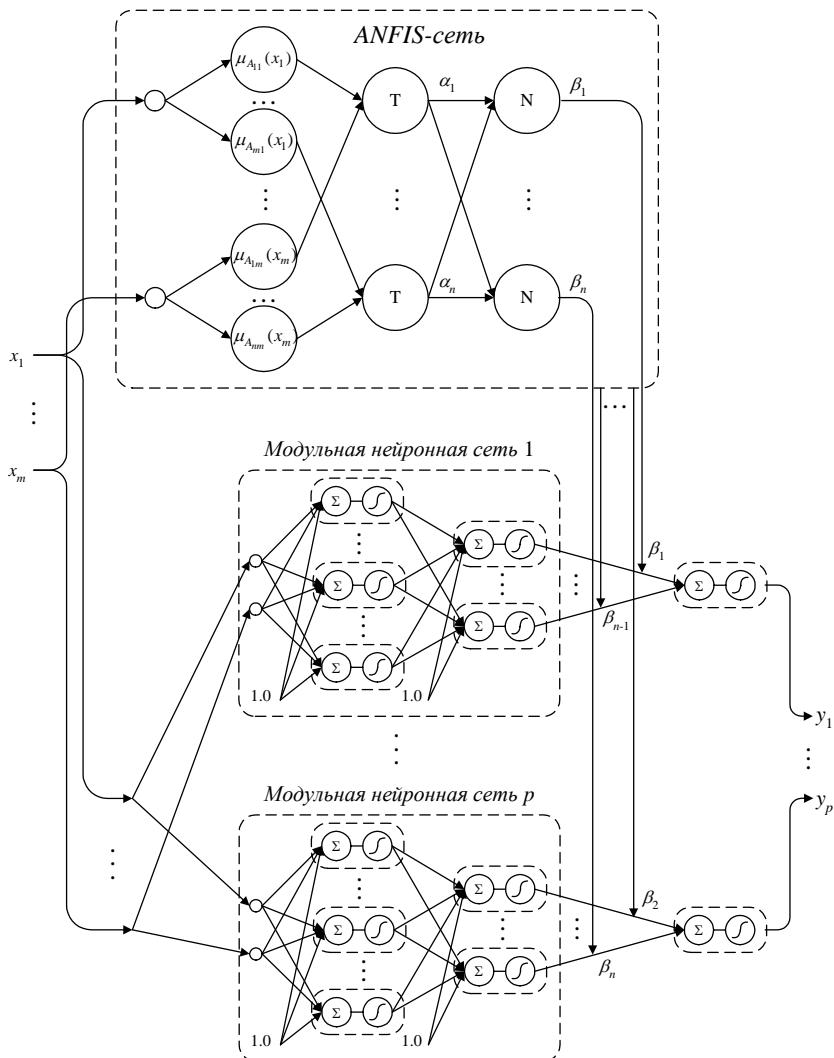


Рис. 13.3. Структура CANFIS-сети (со многими выходами)
с соответствующими модульными нейронными сетями

Нейронная сеть состоит из нейронов входного, скрытого и выходного слоев. Нейроны входного слоя передают входные воздействия на нейроны скрытого *консеквентного слоя* (*consequent layer*) путем изменения активации, определяемой весовыми коэффициентами w_{ij} . Нейроны скрыто-

го и выходного слоев представляют собой нейроподобные элементы, выполняющие преобразования в соответствии с выражениями

$$z_j = f_j \left(w_{0j} + \sum_{i=1}^m w_{ij} x_i \right), \quad y = f \left(\beta_0 + \sum_{j=1}^n \beta_j z_j \right), \quad (13.1)$$

где $f_j(\dots)$, $f(\dots)$ – функции сигмоидального вида.

Весовые коэффициенты β_j нейрона выходного слоя нейронной сети, называемого также *слоем нечеткой ассоциации* (*fuzzy association layer*), формируются ANFIS-сетью, реализующей алгоритм нечеткого вывода Такаги–Сугэно в соответствии с правилами вида

$$\begin{aligned} \Pi_i: & \text{ЕСЛИ } x_1 \text{ есть } A_{i1} \text{ И } \dots \text{ И } x_m \text{ есть } A_{im}, \\ & \text{ТО } z_i = w_{i0} + w_{i1}x_1 + \dots + w_{im}x_m, \quad i = 1, \dots, n. \end{aligned} \quad (13.2)$$

Следует отметить, что назначением ANFIS-сети здесь является формирование весовых коэффициентов β_j для нейрона выходного слоя нейронной сети:

$$\beta_j = \frac{\alpha_j}{\alpha_1 + \dots + \alpha_n}, \quad (13.3)$$

где

$$\alpha_j = \min \{A_{j1}(x_1), \dots, A_{jm}(x_m)\}. \quad (13.4)$$

При этом, в свою очередь, весовые коэффициенты w_{ij} нейронов скрытого слоя нейронной сети являются параметрами для ANFIS-сети.

Такого рода «перекрестное» взаимодействие традиционной нейронной сети и ANFIS-сети позволяет существенно повысить скорость обучения нейронной сети.

С колоколообразными функциями принадлежности CANFIS-сеть функционально подобна нейронной сети с радиальными базисными функциями (Radial Basis Function Network – RBFN) [51]. Как и RBFN-сети, CANFIS-сеть обладает свойством быстрой сходимости и может быстрее обучаться на основе множества примеров по сравнению с нейронными сетями, обучаемыми по алгоритму обратного распространения ошибки.

Недостатком данной нейронной сети является изменение в процессе ее обучения всех весовых коэффициентов w_{ij} , которые, в свою очередь, являются параметрами для ANFIS-сети.

Рассмотренная CANFIS-сеть имеет MISO-структуру со многими входами и одним выходом. Одним из путей получения структуры со многими выходами является использование многих ANFIS-сетей в зависимости

от требуемого числа выходов. В такой MANFIS-сети (Multiple ANFIS) каждая ANFIS-сеть реализует независимое множество нечетких правил. При этом достаточно сложно осуществить корреляции между выходами. Дополнительные сложности здесь заключаются в чрезмерном увеличении количества настраиваемых параметров с ростом числа выходов.

Другим путем формирования сети с MIMO-структурой (со многими входами и многими выходами) является использование CANFIS-сети, представленной на рис. 13.2. Для нее характерно то, что для задания возможных корреляций между выходами совместно используются значения функций принадлежности антецедентов правил ANFIS-сети.

Благодаря корреляциям между консеквентными слоями нейронных сетей эти сети могут быть представлены в виде гораздо меньшего числа модульных нейронных структур в CANFIS-сети (см. рис. 13.3).

ВЫВОДЫ

Предложена классификация способов введения нечеткости в различные компоненты традиционных нейронных сетей на основе положений теории нечетких множеств и нечеткой логики:

- Введение нечеткости в структуры нейронных сетей путем приписывания нечеткости значениям обучающих примеров, «размыванием» входных данных и получением выходной информации в терминах теории нечетких множеств.
- Наделение нейронов, из которых состоят нейронные сети, нечеткостью. Так, входы, выходы, а также веса таких нейронов могут быть нечеткими. Кроме того, активизация нейронных сетей, состоящих из этих нейронов, является нечетким процессом.
- Изменение базисных характеристик нейронов таким образом, что они выполняют операции, используемые в теории нечетких множеств (подобно нечеткому объединению, пересечению, агрегированию), вместо стандартного умножения и сложения.
- Моделирование либо ошибки, либо нестабильности, либо функции энергии нейронной сети с использованием нечетких/неопределенных значений.
- Использование нечетких продукционных сетей в нейронных сетях.

СПИСОК ЛИТЕРАТУРЫ

1. *Keller J. M., Hunt D. J.* Incorporating fuzzy membership function into the perceptron algorithm// IEEE Transactions on Pattern Analysis and Machine Intelligence. 1985. V. 7. №. 6. P. 693–699.
2. *Mitra S., Pal S. K.* Fuzzy multi-layer perceptron, inferencing and rule generation// IEEE Transactions on Neural Networks. 1995. V. 6. P. 51–63.
3. *Mitra S., Pal S. K.* Fuzzy self organization, inferencing and rule generation // IEEE Transactions on Systems, Man and Cybernetics. Part A: Systems and Humans. 1996. V. 26. P. 608–620.
4. *Lee S. C., Lee E. T.* Fuzzy neural networks// Mathematical Biosciences. 1975. V. 23. P. 151–177.
5. *Buckley J. J., Hayashi Y.* Fuzzy neural networks: A survey // Fuzzy Sets and Systems. 1994. V. 66. P. 1–13.
6. *Keller J. M., Krishnapuram R., Rhee F. C.-H.* Evidence aggregation networks for fuzzy logic inference // IEEE Transactions on Neural Networks. 1992. V. 3. P. 761–769.
7. *Mitra S., Pal S. K.* Logical operation based fuzzy MLP for classification and rule generation // Neural Networks. 1994. V. 7. P. 353–373.
8. *Pedrycz W.* A referential scheme of fuzzy decision making and its neural network structure // IEEE Transactions on Systems, Man and Cybernetics. 1991. V. 21. P. 1593–1604.
9. *Ghosh A., Pal N. R., Pal S. K.* Self-organization for object extraction using multilayer neural network and fuzziness measures // IEEE Transactions on Fuzzy Systems. 1993. V. 1. P. 54–68.
10. *Hayashi Y., Buckley J. J., Czogala E.* Fuzzy neural networks with fuzzy signals and weights // Int. J. of Intelligent Systems. 1993. V. 8. №. 4. P. 527–537.
11. *Ishibuchi H., Kwon K., Tanaka H.* A learning algorithm of fuzzy neural networks with triangular fuzzy weights // Fuzzy Set and Systems. 1995. V. 71. P. 277–293.
12. *Nauck D., Klawonn F., Kruse R.* Foundations of neuro-fuzzy systems. Chichester, England: John Wiley & Sons, 1997.
13. *Комарцова Л. Г., Максимов А. В.* Нейрокомпьютеры. М.: Изд-во МГТУ им. Н. Э. Баумана, 2002.
14. *Hayashi Y.* A neural expert systems with automated extraction of fuzzy if-then rules and its application to medical diagnosis // Advanced in Neural

- Information Processing Systems (R. P. Lippmann, J. E. Moody, D. S. Touretzky, eds.), Los Altos: Morgan Kaufmann, 1991. P. 578–584.
- 15. *Hayashi Y.* A neural expert systems using fuzzy teaching input and its application to medical diagnosis // Information Sciences Applications. 1994. V. 1. P. 47–58.
 - 16. *Pal S. K., Mitra S.* Multilayer perceptron, fuzzy sets and classification // IEEE Transactions on Neural Networks. 1992. V. 3. №. 5. P. 683–697.
 - 17. *Fuler R.* Neural fuzzy systems. Abo: Publishing House Abo Akademi University, 1995.
 - 18. *Ishibuchi H., Tanaka H.* Approximate pattern classification using neural networks // in Fuzzy Logic: State of the Art (R. Lowen and M. Roubens eds.). Kluwer, Dordrecht, 1993. P. 225–236.
 - 19. *Lee H.-M., Wang W.-T.* A neural network architecture for classification of fuzzy inputs // Fuzzy Sets and Systems. 1994. V. 63. P. 159–173.
 - 20. *Kwan H. K., Cai Y.* A fuzzy neural network and its application to pattern recognition // IEEE Transactions on Fuzzy Systems. 1994. V. 3. P. 185–193.
 - 21. *Hirota K., Pedrycz W.* Knowledge-based networks in classification problems // Fuzzy Sets and Systems. 1992. V. 51. P. 1–27.
 - 22. *Hirota K., Pedrycz W.* OR/AND neuron in modeling fuzzy set connectives // IEEE Transactions on Fuzzy Systems. 1994. V. 2. P. 151–161.
 - 23. *Eklund P., Virtanen H., Riissanen T.* On the fuzzy logic nature of neural nets // In: Proceedings of Neuro-Nimes, 1991. P. 293–300.
 - 24. *Eklund P., Fogstrom M., Forsstrom J.* A Generic neuro-fuzzy tool for developing medical decision support // In Proceedings MEPP92, International Seminar on Fuzzy Control through Neural Interpretations of Fuzzy Sets (Abo Akademis tryckeri, Abo), 1992. P. 1–27.
 - 25. Круглов В. В., Борисов В. В. Гибридные нейронные сети. Смоленск: Рунич, 2001.
 - 26. *Keller J. M., Yager R. R., Tahani H.* Neural network implementation of fuzzy logic // Fuzzy Sets and Systems. V. 45. 1992. P. 1–12.
 - 27. *Kosko B.* Neural networks and fuzzy systems. Prentice-Hall, Englewood Cliffs, 1992.
 - 28. *Simpson P.* Fuzzy min-max neural networks: 1. Classification // IEEE Transactions on Neural Networks. 1992. V. 3. P. 776–786.
 - 29. *Sun C.-T., Jang J.-S.* A neuro-fuzzy classifier and its applications // In Proc. IEEE Int. Conference on Neural Networks, San Francisco, USA, 1993. P. 94–98.

30. Bezdek J. C., Pal S. K. Fuzzy Models for Pattern Recognition // IEEE Press, New York, 1992.
31. Pedrycz W., Card W. C. Linguistic interpretation of self-organizing maps // In Proc. of the IEEE Int. Conf. on Fuzzy Systems, San Diego, 1992. P. 371–378.
32. Bezdek J. C., Tsao E. C., Pal N. K. Fuzzy Kohonen clustering networks // In Proc. IEEE Int. Conf. on Fuzzy Systems 1992, San Diego, 1992. P. 1035–1043.
33. Kosko B. Fuzzy systems as universal approximators // IEEE Trans. Computers. 1994. V. 43. №. 11. P. 1329–1333.
34. Abe S., Lan M.-S. Fuzzy rules extraction directly from numerical data for function approximation // IEEE Trans. Syst., Man, and Cybernetics. 1995. V. 25. P. 119–129.
35. Abe S., Lan M.-S. A method for fuzzy rule extraction directly from numerical data and its application to pattern classification // IEEE Trans. On Fuzzy Systems. 1995. V. 3. P. 18–28.
36. Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика. М.: Горячая линия – Телеком, 2001.
37. Nelles O., Fink A., Babuška R., Setnes M. Comparison of two construction algorithms for Takagi–Sugeno fuzzy models // International Journal of Applied Mathematics and Computer Science. 2000. V. 4. № 10. P. 835–855.
38. Круглов В. В. Адаптивные системы нечеткого логического вывода// Нейрокомпьютеры: разработка и применение. № 5. 2003. С. 13–16.
39. Осовский С. Нейронные сети для обработки информации. М.: Финансы и статистика, 2002.
40. Song B.G., Marks R.J., Oh S. Adaptive membership function fusion and annihilation in fuzzy if-then rules // In Proc. Second IEEE Int. Conf. Fuzzy Systems, FUZZ-IEEE'93. San Francisco, USA. 1993. V. 2. P. 961–967.
41. Назаров А. В., Лоскутов А. И. Нейросетевые алгоритмы прогнозирования и оптимизации систем. СПб.: Наука и техника, 2003.
42. Ротштейн А. П., Познер М. М., Ракитянская А. Б. Нейронечеткая модель прогнозирования результатов спортивных игр // Сб. трудов 8-й Всероссийской конф. «Нейрокомпьютеры и их применение» НКП–2002, Москва, 2002. С. 251–263.

43. Ротштейн А. П., Митюшкин Ю. И. Нейролингвистическая идентификация нелинейных зависимостей // Кибернетика и системный анализ. 2000. № 3. С. 37–44.
44. Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzzy TECHN. СПб.: БХВ–Петербург, 2003.
45. Рыжов А. П., Федорова М. С. Генетические алгоритмы и задача выбора оператора агрегирования информации в системах информационного мониторинга // Сб. трудов 5-й Всероссийской конф. «Нейрокомпьютеры и их применение» НКП–1999. М., 1999.
46. Umano M., Ezawa Y. Execution of approximate reasoning by neural network // Proceedings of FAN Symposium. 1991. P. 267–273.
47. Uehara K., Fujise M. Learning of fuzzy inference criteria with artificial neural network // In: Proc. 1st Int. Conf. on Fuzzy Logic & Neural Networks, Iizuka, Japan, 1990. P. 193–198.
48. Ishibuchi H., Kwon K., Tanaka H. Implementation of fuzzy IF-THEN rules by fuzzy neural networks with fuzzy weights // In Proc. of EUFIT'93 Conference, Aachen, Germany, 1993. P. 209–215.
49. Ishibuchi H., Fujioka R., Tanaka H. An architecture of neural networks for input vectors of fuzzy numbers // In Proc. IEEE Intern. Conf on Fuzzy Systems, San Diego, USA, 1992. P. 1293–1300.
50. Mizutani E., Jang J.-S. R. Coactive neural fuzzy modeling // In Proc. 1995 IEEE Int. Conf. on Neural Networks, Perth, Western Australia. 1995. V. 2. P. 760–765.
51. Jang J.-S. R., Sun C.-T. Functional equivalence radial basis function networks and fuzzy inference systems // IEEE Trans. Neural Networks. 1993. V. 4. № 1. P. 156–159.

Часть V

НЕЧЕТКИЕ МОДЕЛИ С ПРЕДСТАВЛЕНИЕМ НА ОСНОВЕ ГРАФОВ

Можно выделить отдельный класс нечетких моделей, для представления которых используются структуры в виде ориентированных графов различных типов, в описание отдельных компонентов которых введена нечеткость.

Одной из основных особенностей этих нечетких моделей является ориентированность на анализ и моделирование с их помощью систем и процессов, отображаемых некоторой структурой (в том числе, логико-временными последовательностями, пространственной распределенностью, функциональной взаимосвязанностью, причинно-следственными отношениями и др.) (на основе методов «серого и белого ящика»).

К данным нечетким моделям можно отнести:

- нечеткие автоматы;
- нечеткие алгоритмы;
- нечеткие иерархические сети;
- нечеткие сети Петри;
- нечеткие ситуационные сети;
- нечеткие когнитивные карты;
- нечеткие семантические сети.

В последующих главах рассмотрены наиболее широко распространенные нечеткие модели следующих типов: нечеткие автоматы, нечеткие сети Петри, нечеткие ситуационные сети и нечеткие когнитивные карты.

Глава 14. НЕЧЕТКИЕ АВТОМАТЫ

14.1. Основные понятия, классификация нечетких автоматов

Нечетким автоматом называется упорядоченная шестерка:

$$A = \langle U, X, Y, s_0, \delta, \sigma \rangle, \quad (14.1)$$

где $U = \{a_1, \dots, a_m\}$ – конечное множество входов автомата; $X = \{x_1, \dots, x_n\}$ – конечное множество состояний; $Y = \{y_1, \dots, y_p\}$ – конечное множество его выходов; s_0 – нечеткое начальное состояние автомата $s_0 \in F(X)$ ($s_0: X \rightarrow L$); $\delta: X \times U \times X \rightarrow L$ – функция переходов; $\sigma: X \times Y \rightarrow L$ – функция выходов [1].

Структурно нечеткий автомат представляет собой ориентированный граф, узлами которого являются его состояния, а дуги характеризуют возможные переходы из состояния в состояние.

Функция δ порождает множество нечетких матриц переходов:

$$(T_u)_{u \in U} = \{\delta_{x_i x_j}(u)\}, 1 \leq i, j \leq n, \quad (14.2)$$

функция σ порождает нечеткую матрицу выхода:

$$\sigma = (\sigma_{x_i y_j}), 1 \leq i \leq n, 1 \leq j \leq p. \quad (14.3)$$

Из данного определения нечеткого автомата следует ряд других определений. Так, если $\delta: X \times U \times X \rightarrow \{0, 1\}$, и из некоторых состояний $x_o \in X$ возможен переход в несколько состояний $x_1, \dots, x_r \in X$, то нечеткий автомат является недетерминированным. Если для любого $i \in \{1, \dots, n\}$ существует единственное $j_i \in \{1, \dots, n\}$, такое, что

$$\delta_{x_i x_j}(u) = \begin{cases} 1, & \text{если } j = j_i, \\ 0, & \text{если } j \neq j_i, \end{cases} \quad (14.4)$$

то это нечеткий *детерминированный* автомат.

Допустим U^* – множество возможных входных последовательностей, тогда функция переходов $\delta: X \times U^* \times X \rightarrow L$ вычисляется следующим образом:

$$\begin{aligned} \delta_{xx'}(\lambda) &= \delta_{xx'}(x, \lambda, x') = \begin{cases} 1, & \text{если } x = x', \\ 0, & \text{если } x \neq x', \end{cases} \\ \delta_{xx'}(\theta) &= \delta_{xx'}(x, \theta, x') = \\ &= \vee_{x_1, \dots, x_{k-1} \in X} (\delta(x, u_1, x_1) \wedge \delta(x_1, u_2, x_2) \wedge \dots \wedge \delta(x_{k-1}, u_k, x')), \end{aligned} \quad (14.5)$$

где $\theta = u_1, u_2, \dots, u_k \in U^k$, $\theta \neq \lambda$.

Аналогично определяется матрица переходов:

$$T_\Lambda = \mathbf{I}, T_\theta = T_{u_1} \circ T_{u_2} \circ \dots \circ T_{u_k}. \quad (14.6)$$

Если на вход нечеткого автомата A подается последовательность θ , то выход автомата вычисляется следующим образом:

$$\mu_A(\theta) = s_0 \circ T_\theta \circ \sigma. \quad (14.7)$$

Пример. Рассмотрим пример нечеткого автомата для распознавания нечеткого языка $\Im(V_T)$, где V_T – алфавит, заданный множеством символов, которое является множеством входов U нечеткого автомата.

Нечеткий автомат A , распознающий язык $\Im(V_T)$, строится следующим образом. Допустим,

$$U = V_T = \{a, b\},$$

$Y = \{y\}$ – любое одноэлементное множество,

$$X = \{0, 1, 2, 3, 4\},$$

$$s_0 = \{0\}, \sigma_0 = (1, 0, 0, 0, 0),$$

$$X_F = \{4\}, \sigma = (0, 0, 0, 0, 1),$$

$$T_a = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad T_b = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0 & 0.8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0 & 0.8 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Диаграмма переходов соответствующего нечеткого автомата показана на рис. 14.1. Тогда, если на вход нечеткого автомата подать слово ab , то его выход вычисляется следующим образом:

$$\mu_A(ab) = \sigma_0 \circ T_a \circ T_b \circ \sigma^T = 0.8.$$

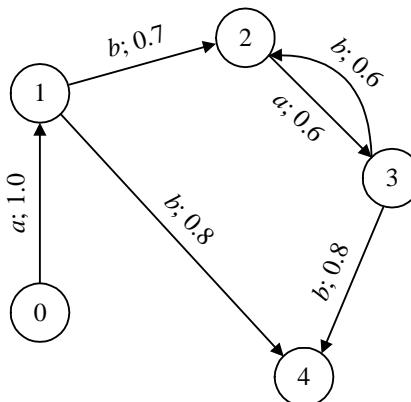


Рис. 14.1. Диаграмма переходов нечеткого автомата

Если на вход автомата подается слово $abab$, то

$$\mu_A(abab) = \sigma_0 \circ T_a \circ T_b \circ T_a \circ T_b \circ \sigma^T = 0.6.$$

Аналогично вычисляется выход для любого слова $(ab)^n$, $n = 3, 4, \dots$. Следовательно, данный нечеткий автомат распознает язык $\Im(V_T)$.

Можно показать, что слово aba , не принадлежащее к языку $\Im(V_T)$, не распознается этим нечетким автоматом:

$$\begin{aligned} \mu_A(aba) &= \sigma_0 \circ T_a \circ T_b \circ T_a \circ \sigma^T = \sigma_0 \circ T_{ab} \circ T_a \circ \sigma^T = \\ &= (1, 0, 0, 0, 0) \circ \begin{bmatrix} 0 & 0 & 1 \wedge 0.7 & 0 & 1 \wedge 0.8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.6 \wedge 0.7 & 0 & 0.6 \wedge 0.8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \circ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \circ \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \\ &= (0, 0, 0, 0.6, 0) \circ \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = 0. \end{aligned}$$

Можно представить еще один специальный тип нечеткого автомата с единственным выходом $Y = \{y_0\}$. В этом случае нечеткий выход определяется вектором $\sigma = (\mu_1, \dots, \mu_k)$, где μ_i – степень получения в состоянии x_i выхода y_0 .

Еще более специальный тип нечеткого автомата можно задать, определяя множество заключительных состояний $X_F \subseteq X$ и функцию выходов:

$$\sigma = (\mu_1, \dots, \mu_k), \quad \mu_i = \begin{cases} 1, & \text{если } x_i \in X_F, \\ 0, & \text{если } x_i \notin X_F. \end{cases} \quad (14.8)$$

Если предположить, что начальное состояние четкое $x_0 \in X$, т. е.

$$\sigma_0 = (i_1, \dots, i_n), \quad i_j = \begin{cases} 1, & \text{если } x_j \in x_0, \\ 0, & \text{если } x_j \neq x_0, \end{cases} \quad (14.9)$$

то значения выхода (значение функции отклика) вычисляется следующим образом:

$$\mu_A(\theta) = \vee_{x_j \in X_F} \delta_{x_0 x_j}(\theta). \quad (14.10)$$

Различные типы нечетких автоматов могут быть определены в зависимости от множества оценок и операций, используемых при вычислении функции переходов (табл. 14.1).

Таблица 14.1. Типы нечетких автоматов

Тип автомата	Используемые операции		Множество оценок
	\vee	\wedge	
Нечеткий автомат [2]	max	min	Полная дистрибутивная решетка на $[0, 1]$
R-нечеткий автомат [1]	+	.	Упорядоченное полукольцо R
Автомат с распределением возможностей [3]	Специально построенные операции		Четырехзначная алгебра Поста
Максимально взвешенный автомат [4]	max	.	Множество неотрицательных действительных чисел
Нечетконечеткий автомат [3]	—	—	Множество нормализованных выпуклых нечетких множеств на $[0, 1]$

Более подробную информацию о свойствах нечетких автоматов, предназначенных для распознавания нечетких языков, можно найти в [1].

Во всех типах четких автоматов (конечных, апериодических, секвенциальных и др.), используемых при решении задач, возникающих при анализе и синтезе дискретных управляющих устройств, явно не описывается временная реализация алгоритмов их функционирования. Реальное время заменено в них набором возможных последовательностей внутренних состояний автомата и правилами выбора в данный тakt работы того или иного перехода на той или иной последовательности. Сами такты работы не связаны с длительностями их реализации в физическом времени.

Для задания автомата в реальном времени необходимы распределения моментов смены сигналов и их значений, а также вероятностные распределения для интервалов смены сигналов и значений сигналов. Однако на практике в большинстве случаев нет априорной информации о необходимых вероятностных распределениях. Поэтому использование нечетких автоматов для описания функционирования дискретных управляющих устройств обосновано, например, необходимостью анализа реальных временных процессов, протекающих в таких устройствах, с заменой вероятностных распределений нечеткими.

Для этого вводится логика времени, дающая возможность пользоваться такими нечеткими понятиями, как «значительно быстрее», «ранее чем», «вскоре после этого» и т. п.

Обозначим через t_i временной интервал. Тогда нечеткое событие на интервале t_i будет обозначаться как $\langle a, t_i, \mu_a(t_i) \rangle$, где a – имя события, t_i – временной интервал, $\mu_a(t_i)$ – функция принадлежности события a к временному интервалу t_i .

Помимо $\mu_a(t_i)$, могут быть использованы специальные нечеткие кванторы типа: \forall^j (например, $j = 1$ – «в подавляющем большинстве случаев», $j = 2$ – «почти всегда», $j = 3$ – «часто», $j = 4$ – «примерно в половине случаев», $j = 5$ – «редко», $j = 6$ – «почти никогда», $j = 7$ – «в исключительных случаях» и т. д.). Таким образом, запись вида $\forall^2 a(t_i)$ будет трактоваться следующим образом: «событие a происходит в интервале t_i почти всегда».

Возможны следующие разновидности нечетких автоматов в зависимости от компонентов, в которые введена нечеткость:

- с нечеткими входами;
- с нечеткими состояниями;
- с нечеткими переходами;
- одновременно с нечеткими входами, состояниями и/или переходами.

Автоматы, нечеткие по входам, целесообразно использовать в случаях, когда проводится анализ процессов, протекающих в сложных схемах, для которых выполнена декомпозиция на блоки. При этом на основании заданной логики работы схемы возможно оценить качественную априорную информацию о возможном появлении или непоявлении сигналов на входах того или иного блока. Временные интервалы в этом случае, как правило, отсчитываются не в абсолютном, а в относительном событийном времени. Функционирование же блоков предполагается детерминированным и задается автоматными таблицами.

Автоматы, нечеткие по состояниям, также являются детерминированными автоматами. Однако в отличие от автоматов, нечетких по входам, в данных автоматах отсутствует точная априорная информация об их начальном состоянии (текущем состоянии в момент анализа их работы).

Автоматы, нечеткие по переходам, не могут быть заданы обычными автоматными таблицами и являются, по существу, недетерминированными. Точная априорная информация о внутреннем состоянии и входных сигналах в таких автоматах не дает возможности однозначно определить новое состояние. Такое возможно не только когда работа анализируемого устройства недетерминированна, но и когда в детерминированном автомате возникают различные сбои.

14.2. Обучение нечетких автоматов

Назначением обучения нечетких автоматов является улучшение качества или эффективности их функционирования за счет модификации структуры или значений параметров. В соответствии с приведенной в предыдущем подразделе классификацией нечетких автоматов различают и алгоритмы их обучения.

Рассмотрим вариант обучения нечеткого автомата с четкими входами и нечеткими состояниями и переходами, используемого для классификации образов [1]. Пусть $i(t)$ – четкий вход с оценкой значения i_l , $\delta(t)$ – нечеткое, зависящее от времени, отношение перехода, $\tilde{s}(t)$ – нечеткое состояние автомата в момент времени t на конечном множестве состояний $S = \{s_1, \dots, s_n\}$. Пусть также состояние нечеткого автомата в момент времени $(t+1)$ определяется тах-мин-композицией:

$$\mu_{\tilde{s}(t+1)}(s_k) = \sup_j \min(\mu_{\tilde{s}(t)}(s_j), \mu_{\delta(t)}(s_x, i_l, s_j)). \quad (14.11)$$

Обучение нечеткого автомата направлено на изменение нечеткой матрицы переходов:

$$\begin{aligned} \mu_{\delta(t)}(s_k, i_l, s_j) &= \mu_{\delta(t-1)}(s_k, i_l, s_k), \quad j \neq k, \\ \mu_{\delta(t)}(s_k, i_l, s_k) &= \alpha_k \mu_{\delta(t-1)}(s_k, i_l, s_k) + (1 - \alpha_k) \lambda_k(t), \end{aligned} \quad (14.12)$$

где α_k – скорость обучения, $0 < \alpha_k < 1$, $0 < \lambda_k(t) \leq 1$, $k = 1, \dots, n$.

Начало обучения нечеткого автомата возможно без априорной информации $\mu_{\tilde{s}(0)}(s_k) = 0$ или 1 , а также при наличии априорной информации $\mu_{\tilde{s}(0)}(s_k) = \lambda_k(0)$. Значение $\lambda_k(0)$ зависит от оценки функционирования автомата. Доказано, что имеет место сходимость автомата вне зависимости от того, есть ли априорная информация (т. е. значение $\mu_{\tilde{s}(0)}(s_j)$ может быть любым из интервала $[0, 1]$).

Заранее выбирается критерий оценки, минимизация или максимизация которого отражает свойства классификации.

Модель обучения включает в себя следующие этапы.

Этап 1. Область определения целевой функции делится на несколько подобластей (форма которых постоянно меняется) и описывается некоторым множеством точек (образцов).

Этап 2. Каждой точке приписывается состояние автомата, причем функция принадлежности в каждом состоянии указывает степень близости к оптимуму (экстремуму целевой функции).

Этап 3. Задается входной образ и выбирается состояние автомата с максимальным значением функции принадлежности (эта точка называется кандидатом), т. е. оценивается целевая функция.

Этап 4. Формируется новая подобласть из точек, окружающих кандидата. При этом размер подобласти увеличивается, если значения целевой функции в точке-кандидате меньше, чем в других точках подобласти, и уменьшается в ином случае.

Этап 5. Когда подобласть пересекается с некоторой другой или две точки-кандидаты находятся в одной подобласти, то подобласти разделяются, если же степень разделения большая, либо подобласти объединяются, если степень разделения малая.

Этап 6. На этапе локального поиска выбираются точки-кандидаты в подобластях. Затем на этапе глобального поиска во всей области среди точек-кандидатов ищется оптимальная точка, характеризующая глобальный экстремум целевой функции. Этапы глобального и локального поиска выполняются поочередно. Следует отметить, что изменение нечеткой матрицы переходов (см. выше) осуществляется на этапах глобального и локального поисков. В случае если целевая функция увеличилась, то

$$\mu_{\delta(t)}(s_k, i_l, s_k) = \alpha_k \mu_{\delta(t-1)}(s_k, i_l, s_k) + (1 - \alpha_k) \lambda_k(t),$$

если же попытка неудачна, то

$$\mu_{\delta(t)}(s_k, i_l, s_j) = \alpha_k \mu_{\delta(t-1)}(s_k, i_l, s_k),$$

$\alpha_k < 1$ гарантирует сходимость.

Глава 15. НЕЧЕТКИЕ СЕТИ ПЕТРИ

Назначением сетей Петри (СП) является адекватное представление и анализ структуры динамических дискретных моделей сложных систем и логико-временных особенностей процессов их функционирования.

В настоящее время известно большое количество разновидностей сетей Петри, предназначенных для анализа различных структурных, функциональных и процессуальных аспектов этих моделей, исследованию которых посвящена обширная библиография. Целью данной главы является рассмотрение основных типов нечетких сетей Петри (НСП), находящих наибольшее практическое применение при решении прикладных задач, характеризующихся учетом различных факторов неопределенности.

15.1. Основные понятия, классификация нечетких сетей Петри

Сеть Петри структурно представляет собой граф специального вида с правилами, которые определяют динамику процесса ее функционирования.

Обобщенная маркованная сеть Петри (далее – сеть Петри) – это упорядоченная пятерка

$$C = (P, T, I, O, \mathbf{m}_0), \quad (15.1)$$

где $P = \{p_1, \dots, p_n\}$ – множество позиций; $T = \{t_1, \dots, t_n\}$ – множество переходов; I – входная функция переходов, являющаяся отображением, $I: P \times T \rightarrow \Lambda_0$ ($\Lambda_0 = \{0, 1, 2, 3, \dots\}$); O – выходная функция переходов, являющаяся отображением, $O: T \times P \rightarrow \Lambda_0^*$; $\mathbf{m}_0 = (m_1^0, \dots, m_n^0)$ – вектор начальной маркировки СП, при этом $m_i^0 \in \Lambda_0$ ($\forall i \in \{1, 2, \dots, n\}$) и m_i^0 – компонент вектора начальной маркировки СП, соответствующий позиции $p_i \in P$.

В определении сети Петри часто выделяют первые 4 компонента, которые задают ее структуру: $N = (P, T, I, O)$. В этом случае сеть Петри может быть записана в виде

$$C = (N, \mathbf{m}_0). \quad (15.2)$$

Динамика изменения маркировок сетей Петри после момента ее запуска подчиняется следующим правилам.

Я₁: *правило определения текущего состояния*. Любое состояние СП определяется некоторой его маркировкой $\mathbf{m} = (m_1, \dots, m_n)$.

Я₂: *правило активности переходов*. Переход $t_j \in T$ называется активным при некоторой маркировке $\mathbf{m}_k = (m_1^k, \dots, m_n^k)$, если выполнено условие

$$m_i^k \geq I(p_i, t_j), \quad \forall p_i \in P. \quad (15.3)$$

То есть переход является активным, если в каждой из его входных позиций имеется такое число маркеров, которое больше или равно количеству дуг, соединяющих соответствующую входную позицию с данным переходом.

Я₃: *правило срабатывания переходов*. Если некоторый переход $t_j \in T$ активен при текущей маркировке $\mathbf{m}_k = (m_1^k, \dots, m_n^k)$, т. е. выполнено правило активности перехода (15.3), то срабатывание данного перехода, осуществляющее мгновенно, приводит к новой маркировке сети $\mathbf{m}_{k+1} = (m_1^{k+1}, \dots, m_n^{k+1})$:

$$m_i^{k+1} = m_i^k + O(t_j, p_i) - I(p_i, t_j), \quad \forall p_i \in P. \quad (15.4)$$

То есть срабатывание j -го активного перехода перемещает маркеры из входных позиций данного перехода в его выходные позиции таким образом, что во всех входных позициях этого перехода исчезает столько маркеров, сколько дуг соединяет эту входную позицию с данным переходом.

Соответственно во всех его выходных позициях появляется столько маркеров, сколько дуг соединяет данный переход с соответствующей выходной позицией. При этом перемещение маркеров осуществляется мгновенно.

Частным случаем рассмотренной обобщенной маркированной сети Петри является *ординарная маркированная сеть Петри*, определяемая той же пятеркой $C = \langle P, T, I, O, m_0 \rangle$, однако обладающая более простой структурой, в которой в качестве входной функции переходов используется отображение $I: P \times T \rightarrow \{0, 1\}$, а в качестве выходной функции переходов – отображение $O: T \times P \rightarrow \{0, 1\}$. Правила, задающие динамику изменения маркировок такой сети, также базируются на соответствующих правилах для обобщенной маркированной сети Петри.

Классом сетей Петри, обобщающим вышерассмотренные сети, используемых для анализа динамических дискретных моделей сложных систем и логико-временных особенностей процессов их функционирования, являются *временные сети Петри* (ВСП), которые определяются следующим образом:

$$\langle C_{PT}, \mathcal{R} \rangle, \quad (15.5)$$

где $C_{PT} = (N, m_0, z, s)$ – базовый формализм временной сети Петри, в котором: $N = (P, T, I, O)$ – структура временной сети Петри, аналогичная структуре обобщенной маркированной сети Петри; $m_0 = (m_1^0, \dots, m_n^0)$ – вектор начальной маркировки, $m_i^0 \in \mathbb{N}_0 (\forall i \in \{1, 2, \dots, n\})$; $z = (z_1, \dots, z_n)$ – вектор временных задержек маркеров в позициях ВСП, $z_i \in \mathbb{N}_0 (\forall i \in \{1, 2, \dots, n\})$; $s = (s_1, \dots, s_n)$ – вектор времен срабатывания разрешенных переходов ВСП, $s_i \in \mathbb{N}_0 (\forall i \in \{1, 2, \dots, n\})$; \mathcal{R} – совокупность правил, определяющих процесс запуска и функционирования ВСП, которые включают в себя условия активности и срабатывания переходов, доступности маркеров в позициях ВСП, изменения начальной и последующих маркировок и др.

Примечание. Приведенные выше определения сетей Петри являются достаточными для уточнения нижеследующих описаний практически используемых для решения прикладных задач нечетких временных сетей Петри, классификация которых получена путем введения нечеткости в различные компоненты временных сетей Петри.

Последовательно вводя нечеткость в описание отдельных компонентов ВСП $\langle C_{PT}, \mathcal{R} \rangle$, можно выделить следующие основные классы нечетких временных сетей Петри [5]:

- $\langle \hat{N}, \mathbf{m}_0, z, s, \mathcal{R} \rangle$ – временные сети Петри с нечеткостью в задании структуры (P, T, I, O) ;
- $\langle N, \hat{\mathbf{m}}_0, z, s, \mathcal{R} \rangle$ – временные сети Петри с нечеткостью задания начальной маркировки \mathbf{m}_0 ;
- $\langle N, \mathbf{m}_0, \hat{z}, \hat{s}, \mathcal{R} \rangle$ – временные сети Петри с нечеткостью задания времен задержки z маркеров в позициях и времен срабатывания s активных переходов;
- $\langle N, \hat{\mathbf{m}}_0, \hat{z}, \hat{s}, \mathcal{R} \rangle$ – временные сети Петри с нечеткостью задания начальной маркировки \mathbf{m}_0 , времен задержки маркеров z в позициях и времен срабатывания s активных переходов;
- $\langle N, \mathbf{m}_0, z, s, \hat{\mathcal{R}} \rangle$ – временные сети Петри с нечеткостью в задании правил \mathcal{R} , определяющих процесс функционирования сети.

Кратко рассмотрим примеры основных практически используемых типов нечетких временных сетей Петри.

15.2. Нечеткие сети Петри с нечеткостью задания начальной маркировки

Для наглядности будем полагать $z = 0, s = 0$. В этом случае нечеткая временная сеть Петри данного типа конкретизируется НСП типа

$$V_f = \langle N, \hat{M}_0, \mathcal{R} \rangle, \quad (15.6)$$

где $N = (P, T, I, O)$ – структура НСП V_f ; $P = \{p_1, \dots, p_n\}$ – множество позиций; $T = \{t_1, \dots, t_u\}$ – множество переходов; I – входная функция переходов, $I: P \times T \rightarrow \mathbb{N}_0$; O – выходная функция переходов, $O: T \times P \rightarrow \mathbb{N}_0$; \hat{M}_0 – матрица начальной маркировки размерности $(n \times (d + 1))$. Каждый ее элемент $m_{ij}^0 \in [0, 1]$ ($\forall i \in \{1, 2, \dots, n\}, \forall j \in J$) равен значению функции принадлежности наличия $(j - 1)$ числа маркеров в позиции p_i на момент начала запуска НСП, $J = \{1, 2, \dots, d, d + 1\} \subset \mathbb{N}_0$. При этом число столбцов матрицы \hat{M}_0 определяется максимальным числом маркеров d в позициях НСП; \mathcal{R} – множество правил, модификация которых отражает специфику введенной нечеткости в начальную маркировку.

На рис. 15.1 представлен пример структуры НСП типа V_f , которая идентична структуре обобщенной маркированной сети Петри и графически изображается ориентированным двудольным мультиграфом.

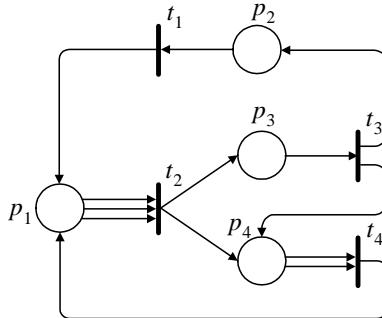


Рис. 15.1. Пример структуры нечеткой сети Петри типа V_f

Динамика изменения маркировок нечеткой сети Петри типа V_f после момента ее запуска подчиняется следующим правилам, отражающим специфику введенной нечеткости в маркировку.

Ж1: правило определения текущего состояния. Любое состояние НСП V_f определяется некоторой его маркировкой, т. е. матрицей \hat{M} размерности $(n \times (d + 1))$, каждый элемент которой $m_{ij} \in [0, 1]$, и интерпретируется как значение функции принадлежности наличия $\{0, 1, 2, \dots, d\}$ маркеров в соответствующих позициях $p_i \in P$. Начальное состояние НСП V_f определяется матрицей начальной маркировки \hat{M}_0 .

Ж2: правило активности переходов. Переход $t_k \in T$ НСП V_f называется активным при некоторой текущей маркировке \hat{M} , если выполнено условие

$$\sigma_i \geq I(p_i, t_k), \quad i = 1, \dots, n, \quad (15.7)$$

где

$$\sigma_i = \max_{(j \in J) \wedge (m_{ij} > 0)} (\{j\} - 1), \quad i = 1, \dots, n. \quad (15.8)$$

То есть σ_i ($i = 1, \dots, n$) равно индексу максимального, отличного от нуля, значения степени принадлежности наличия маркеров в позициях p_i НСП V_f . При этом некоторый переход $t_k \in T$ является активным, если для текущей маркировки \hat{M} во всех его входных позициях имеется нечеткое количество маркеров, большее или равное числу дуг, соединяющих соот-

ветствующие входные позиции с рассматриваемым переходом нечеткой сети Петри.

Яз₃: правило срабатывания переходов. Если некоторый переход $t_k \in T$ НСП V_f активен при некоторой текущей маркировке \hat{M}_l , т. е. выполняется правило (15.7), то срабатывание данного перехода, осуществляемое мгновенно, приводит к новой маркировке сети \hat{M}_{l+1} , элементы которой определяются следующим образом:

- для каждой из входных позиций $p_i \in P$, для которых $I(p_i, t_k) > 0$:

$$\begin{aligned} m_{ii}^{l+1} &= \max_{(j \in \{1, 2, \dots, I(p_i, t_k) + 1\})} \{m_{ij}\}, \\ m_{ij}^{l+1} &= m_{i, j+I(p_i, t_k)}, \quad j = 2, \dots, d+1. \end{aligned} \quad (15.9)$$

- для каждой из выходных позиций $p_i \in P$, для которых $O(t_k, p_i) > 0$:

$$\begin{aligned} m_{ij}^{l+1} &= \min \{m_{ij}, 1 - q_k\}, \quad j = 1, 2, \dots, O(t_k, p_i), \\ m_{ij}^{l+1} &= \max \{\min \{m_{ij}, 1 - q_k\}, \min \{m_{i, j-O(t_k, p_i)}, q_k\}, \\ &\quad (\forall j \in J) \wedge (j > O(t_k)), p_i\}. \end{aligned} \quad (15.10)$$

где q_k – степень принадлежности, характеризующая возможность нечеткого срабатывания перехода t_k , рассчитываемая следующим образом:

$$q_k = \min_{(i \in \{1, 2, \dots, n\})} \{ \max_{(j \in J) \wedge (j > I(p_i, t_k))} \{m_{ij}\} \}, \quad (\forall t_k \in T). \quad (15.11)$$

- если некоторые из позиций $p_i \in P$ являются одновременно входными и выходными для активного перехода t_k , то для них элементы матрицы новой маркировки \hat{M}_{l+1} рассчитываются последовательно, сначала по выражениям (15.9), затем – по выражениям (15.10).

Пример (примеры в данной главе взяты из книги [5]). Допустим, на рис. 15.1 задана структура НСП V_f со множеством позиций $P = \{p_1, p_2, p_3, p_4\}$ и множеством переходов $T = \{t_1, t_2, t_3, t_4\}$.

Определены следующие входная и выходная функции (матрицы) переходов:

$$I = \begin{bmatrix} 0 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}, \quad O = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Также задана начальная маркировка

$$\hat{M}_0 = \begin{bmatrix} 0.1 & 0.2 & 0.6 & 0.8 \\ 0.8 & 0 & 0 & 0 \\ 0.7 & 0 & 0 & 0 \\ 0.6 & 0.5 & 0 & 0 \end{bmatrix}.$$

Элементы строк этой матрицы m_{ij}^0 определяют степени принадлежности нечеткого наличия ($j - 1$) количества маркеров в позиции p_i данной НСП. При этом общее количество маркеров в НСП равно $d = 3$. Например, $m_{11}^0 = 0.1$ означает степень принадлежности нечеткого отсутствия маркеров в позиции p_1 , а $m_{13}^0 = 0.6$ – степень принадлежности наличия двух маркеров в позиции p_1 .

Проиллюстрируем применение описанных правил \mathcal{R} на примере изменения начальной маркировки данной НСП V_f .

В соответствии с правилом \mathcal{R}_2 при данной начальной маркировке \hat{M}_0 активным является единственный переход t_2 , так как только для него выполнено условие (15.7), а именно $(3, 0, 0, 1) \geq (3, 0, 0, 0)$. Срабатывание этого перехода приводит к новой маркировке \hat{M}_1 , элементы которой формируются по правилу \mathcal{R}_3 следующим образом:

- для позиции p_1 , являющейся входной для перехода t_2 со значением входной функции переходов $I(p_1, t_2) = 3$, на основе выражений (15.9):
 $m_{11}^1 = \max\{0.1, 0.2, 0.6, 0.8\} = 0.8$, $m_{12}^1 = m_{13}^1 = m_{14}^1 = 0$;

- для позиций p_3 и p_4 , являющихся выходными для перехода t_2 со значениями $O(t_2, p_3) = 1$ и $O(t_2, p_4) = 1$, с учетом того, что $q_2 = 0.8$, на основе выражений (15.10):

$$m_{31}^1 = \min\{m_{31}^0, 1 - 0.8\} = \min\{0.7, 0.2\} = 0.2;$$

$$m_{32}^1 = \max\{\min\{m_{32}^0, 1 - q_2\}, \min\{m_{31}^0, q_2\}\} = \max\{\min\{0, 0.2\}, \min\{0.7, 0.8\}\} = 0.7;$$

$$m_{33}^1 = m_{34}^1 = 0;$$

$$m_{41}^1 = \min\{m_{41}^0, 1 - 0.8\} = \min\{0.6, 0.2\} = 0.2;$$

$$m_{42}^1 = \max\{\min\{m_{42}^0, 1 - q_2\}, \min\{m_{41}^0, q_2\}\} = \max\{\min\{0.5, 0.2\}, \min\{0.6, 0.8\}\} = 0.6;$$

$$m_{43}^1 = \max\{\min\{m_{43}^0, 1 - q_2\}, \min\{m_{42}^0, q_2\}\} = \max\{\min\{0, 0.2\}, \min\{0.5, 0.8\}\} = 0.5;$$

$$m_{44}^1 = 0;$$

- для позиции p_2 , не являющейся для перехода t_2 ни входной, ни выходной, элементы новой маркировки будут равны элементам начальной маркировки: $m_{21}^1 = 0.8$; $m_{22}^1 = m_{23}^1 = m_{24}^1 = 0$.

Таким образом, после нечеткого срабатывания перехода t_2 начальная маркировка \hat{M}_0 изменится на маркировку \hat{M}_1 :

$$\hat{M}_1 = \begin{bmatrix} 0.8 & 0 & 0 & 0 \\ 0.8 & 0 & 0 & 0 \\ 0.2 & 0.7 & 0 & 0 \\ 0.2 & 0.6 & 0.5 & 0 \end{bmatrix}.$$

Полученная маркировка \hat{M}_1 активизирует переходы t_3 и t_4 и т. д.

15.3. Нечеткие сети Петри с нечеткостью задания начальной маркировки и срабатывания переходов

По-прежнему для наглядности будем полагать $z = 0$, $s = 0$. Однако по сравнению с предыдущим вариантом в нечеткой временной сети Петри данного типа нечеткость вводится в начальную маркировку и в правила срабатывания переходов ординарных СП:

$$C_f = \langle N, \hat{m}_0, \hat{f}, \lambda, \mathcal{R} \rangle, \quad (15.12)$$

где $N = (P, T, I, O)$ – структура НСП V_f : $P = \{p_1, \dots, p_n\}$ – множество позиций; $T = \{t_1, \dots, t_u\}$ – множество переходов; I – входная функция переходов, $I: P \times T \rightarrow \aleph_0$; O – выходная функция переходов, $O: T \times P \rightarrow \aleph_0$; $\hat{m}_0 = (m_1^0, m_2^0, \dots, m_n^0)$ – вектор начальной маркировки, каждый элемент которого определяется значением функции принадлежности нечеткого наличия одного маркера в соответствующей позиции данной НСП типа C_f , $m_i^0 \in [0, 1] (\forall i \in \{1, 2, \dots, n\})$; $\hat{f} = (f_1, f_2, \dots, f_u)$ – вектор значений функции принадлежности нечеткого срабатывания переходов НСП C_f , $f_j \in [0, 1] (\forall j \in \{1, 2, \dots, u\})$; $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_u)$ – вектор значений порога срабатывания переходов НСП C_f , $\lambda_j \in [0, 1] (\forall j \in \{1, 2, \dots, u\})$; \mathcal{R} – множество правил, модификация которых отражает специфику введенной нечеткости в начальную маркировку и в правила срабатывания переходов ординарных СП.

На рис. 15.2 представлен пример структуры НСП типа C_f , которая идентична структуре ординарной маркированной сети Петри и графически изображается ориентированным двудольным графом.

Динамика изменения маркировок нечеткой сети Петри типа C_f после момента ее запуска подчиняется следующим правилам, отражающим специфику введенной нечеткости в маркировку и в правила срабатывания переходов.

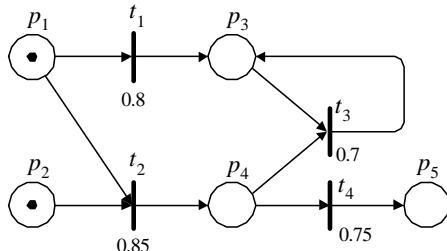


Рис. 15.2. Пример структуры нечеткой сети Петри типа V₁

Я₁: правило определения текущего состояния. Любое состояние НСП C_f определяется вектором $\hat{m} = (m_1, m_2, \dots, m_n)$, элементы которого $m_i \in [0, 1]$, и интерпретируются как значение функции принадлежности нечеткого наличия одного маркера в соответствующих позициях $p_i \in P$. Начальное состояние НСП C_f определяется вектором начальной маркировки \hat{m}_0 .

Я₂: правило активности переходов. Переход $t_k \in T$ НСП C_f называется активным при некоторой текущей маркировке \hat{m} , если выполнено условие

$$\max_{(i \in \{1, 2, \dots, n\}) \wedge (I(p_i, t_k) > 0)} \{m_{ij}\} \geq \lambda_k, \quad (15.13)$$

где λ_k – значение порога срабатывания перехода $t_k \in T$.

То есть переход $t_k \in T$ является активным, если во всех его входных позициях имеются ненулевые значения элементов вектора текущей маркировки \hat{m} , а минимальное из них не меньше порога срабатывания данного перехода.

Я₃: правило срабатывания переходов. Если некоторый переход $t_k \in T$ НСП C_f активен при некоторой текущей маркировке \hat{m}_l [т. е. для него выполнено условие (15.13)], то нечеткое срабатывание данного перехода, осуществляющее мгновенно, приводит к новой маркировке $\hat{m}_{l+1} = (m_1^{l+1}, m_2^{l+1}, \dots, m_n^{l+1})$, элементы вектора которой определяются следующим образом:

- для каждой из входных позиций $p_i \in P$, для которых $I(p_i, t_k) > 0$:

$$m_i^{l+1} = 0, (\forall p_i \in P) \wedge (I(p_i, t_k) > 0); \quad (15.14)$$

- для каждой из выходных позиций $p_i \in P$, для которых $O(t_k, p_i) > 0$:

$$m_j^{l+1} = \max_{(i \in \{1, 2, \dots, n\}) \wedge (I(p_i, t_k) > 0)} \{m_i, f_k\}; \quad (\forall p_j \in P) \wedge O(t_k, p_i) > 0; \quad (15.15)$$

- если некоторые из позиций $p_i \in P$ являются одновременно входными и выходными для активного перехода t_k , то для них элементы вектора новой маркировки \hat{m}_{l+1} рассчитываются последовательно, сначала по выражениям (15.14), затем – по выражениям (15.15).

Пример. Допустим, на рис. 15.2 задана структура НСП C_f со множеством позиций $P = \{p_1, p_2, p_3, p_4, p_5\}$ и множеством переходов $T = \{t_1, t_2, t_3, t_4\}$.

Определены следующие входная и выходная функции (матрицы) переходов:

$$I = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad O = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Также заданы: вектор начальной маркировки $\hat{m}_0 = (0.95, 0.9, 0, 0, 0)$; вектор значений функции принадлежности нечеткого срабатывания переходов $\hat{f} = (0.8, 0.85, 0.7, 0.75)$; вектор значений порога срабатывания переходов $\lambda = (0.6, 0.5, 0.4, 0.5)$.

Рассмотрим применение описанных выше правил \mathcal{R} на примере изменения начальной маркировки данной НСП C_f .

При заданной начальной маркировке НСП активными являются два перехода – t_1 и t_2 , так как для них выполнено условие (15.13), т. е. $0.95 > 0.6$ и $\min\{0.95, 0.9\} > 0.5$ соответственно.

Нечеткое срабатывание перехода t_1 приводит к новой маркировке \hat{m}_1 , элементы вектора которой формируются по правилу \mathcal{R}_3 :

- для позиции p_1 , являющейся входной для перехода t_1 , на основе выражения (15.14) $m_1^1 = 0$;
- для позиции p_3 , являющейся выходной для перехода t_1 , на основе выражения (15.15): $m_3^1 = \max\{0, \min\{0.95, 0.8\}\} = 0.8$;
- для позиций p_2, p_4, p_5 , не являющихся для перехода t_1 ни входными, ни выходными, элементы новой маркировки будут равны элементам начальной маркировки: $m_2^1 = 0.9$; $m_4^1 = m_5^1 = 0$.

Таким образом, нечеткое срабатывание перехода t_1 приводит к новой маркировке: $\hat{m}_1 = (0, 0.9, 0.8, 0, 0)$.

Нечеткое срабатывание перехода t_2 приводит к новой маркировке \hat{m}_2 , элементы вектора которой формируются по правилу \mathcal{R}_3 следующим образом:

- для позиций p_1 и p_2 , являющихся входными для перехода t_2 , на основе выражения (15.14) $m_1^2 = 0, m_2^2 = 0$;
- для позиции p_4 , являющейся выходной для перехода t_2 , на основе выражения (15.15): $m_4^2 = \max\{0, \min\{0.95, 0.9, 0.85\}\} = 0.85$;
- для позиций p_3 и p_5 , не являющихся для перехода t_2 ни входными, ни выходными, элементы новой маркировки останутся неизменными: $m_3^2 = m_5^2 = 0$.

Нечеткое срабатывание перехода t_2 приводит к изменению начальной маркировки на новую маркировку $\hat{\mathbf{m}}_2 = (0, 0, 0, 0.85, 0)$.

Следует отметить, что нечеткое срабатывание перехода t_1 делает неактивным переход t_2 и наоборот. То есть данные переходы являются конфликтными, что приводит к их альтернативному срабатыванию.

При текущей маркировке $\hat{\mathbf{m}}_1$ отсутствуют активные переходы, а при маркировке $\hat{\mathbf{m}}_2$ активным является переход t_4 , так как для него выполнено условие (15.13). Нечеткое срабатывание перехода t_4 изменит маркировку $\hat{\mathbf{m}}_2$ на новую $\hat{\mathbf{m}}_3 = (0, 0, 0, 0, 0.75)$, для которой активных переходов нет.

15.4. Нечеткие сети Петри с нечеткостью задания начальной маркировки, времен задержки маркеров в позициях и времен срабатывания активных переходов

Такие сети описываются следующим образом:

$$C_{PT}^v = \langle N, \hat{\mathbf{M}}_0, \hat{\mathbf{z}}, \hat{\mathbf{s}}, \mathcal{R} \rangle, \quad (15.16)$$

где $N = (P, T, I, O)$ – структура НСП C_{PT}^v : $P = \{p_1, \dots, p_n\}$ – множество позиций; $T = \{t_1, \dots, t_u\}$ – множество переходов; I – входная функция переходов, $I: P \times T \rightarrow \aleph_0$; O – выходная функция переходов, $O: T \times P \rightarrow \aleph_0$; $\hat{\mathbf{m}}_0 = (m_1^0, m_2^0, \dots, m_n^0)$ – вектор начальной маркировки, каждый элемент которого представляет собой некоторую неотрицательную нечеткую величину, определяющую нечеткое наличие одного маркера в соответствующей позиции данной НСП типа C_{PT}^v ; $\hat{\mathbf{z}} = (z_1, z_2, \dots, z_n)$ – вектор временных задержек маркеров в позициях, каждый элемент которого является неотрицательной нечеткой величиной; $\hat{\mathbf{s}} = (s_1, s_2, \dots, s_u)$ – вектор времен срабатывания разрешенных переходов, каждый элемент которого является неотрицательной нечеткой величиной; \mathcal{R} – множество правил,

модификация которых отражает специфику введенной нечеткости в маркировку, во времена задержки маркеров в позициях и во времена срабатывания активных переходов ординарных СП.

Примечание. Нечеткие величины, используемые в качестве элементов векторов \hat{m}_0 , \hat{z} и \hat{s} , могут быть представлены в виде нечетких интервалов и нечетких чисел, например в виде $(L-R)$ -представления. При этом наиболее удобными с вычислительной точки зрения являются трапециевидные нечеткие интервалы и треугольные нечеткие числа.

На рис. 15.3 показан пример структуры НСП типа C_{PT}^V в виде ориентированного двудольного графа. Нечеткие величины, используемые в качестве элементов векторов \hat{m}_0 , \hat{z} и \hat{s} , представлены в виде трапециевидных нечетких интервалов.

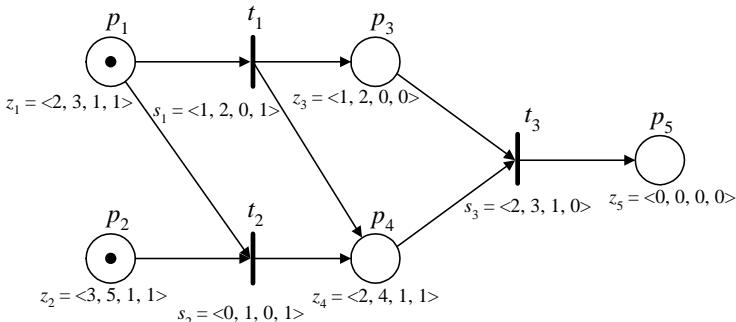


Рис. 15.3. Пример структуры нечеткой сети Петри типа C_{PT}^V с представлением элементов векторов \hat{m}_0 , \hat{z} и \hat{s} в виде трапециевидных нечетких интервалов

Динамика изменения маркировок нечеткой сети Петри типа C_{PT}^V после момента ее запуска подчиняется следующим правилам, отражающим специфику введенной нечеткости в начальную маркировку, во времена задержки маркеров в позициях и во времена срабатывания активных переходов. При этом будем полагать, что НСП данного типа является безопасной в том смысле, что в каждой позиции возможно наличие не более одного маркера.

Я₁: правило определения текущего состояния. Любое состояние НСП C_{PT}^V определяется вектором маркировки $\hat{m} = (m_1, m_2, \dots, m_n)$, элементы которого представляют собой трапециевидные нечеткие интервалы $m_i = <a_i, b_i, \alpha_i, \beta_i>$ ($i = 1, \dots, n$), и интерпретируются как значения функции принадлежности нечеткого наличия одного маркера в соответствую-

ших позициях $p_i \in P$ относительно момента запуска данной сети. Начальное состояние НСП C_{PT}^V определяется вектором начальной маркировки \hat{m}_0 .

Я₂: правило активности переходов. Переход $t_k \in T$ НСП C_{PT}^V называется активным при некоторой текущей маркировке \hat{m} , если выполнено следующее условие:

$$m_i > 0, (\forall p_i \in P) \wedge (I(p_i, t_k) > 0), \quad (15.17)$$

т. е. во всех входных позициях данного перехода t_k в заданный момент времени должны быть доступные маркеры, представленные отличными от нуля $m_i = \langle a_i, b_i, \alpha_i, \beta_i \rangle$. Доступность маркеров во входных позициях определяется в соответствии с правилом Я₄ (см. ниже).

Я₃: правило нечеткого срабатывания переходов. Если некоторый переход $t_k \in T$ НСП C_{PT}^V активен при текущей маркировке \hat{m}_l [т. е. для него выполнено условие (15.17)], то нечеткое срабатывание этого перехода, выполняемое за время s_k , приводит к новой маркировке $\hat{m}_{l+1} = (m_1^{l+1}, m_2^{l+1}, \dots, m_n^{l+1})$, элементы вектора которой определяются следующим образом:

- для каждой из входных позиций $p_i \in P$, для которых $I(p_i, t_k) > 0$:

$$m_i^{l+1} = \langle 0, 0, 0, 0 \rangle (\forall p_i \in P) \wedge (I(p_i, t_k) > 0); \quad (15.18)$$

- для каждой из выходных позиций $p_i \in P$, для которых $O(t_k, p_i) > 0$:

$$m_j^{l+1} = \min \left\{ \max_{(i \in \{1, 2, \dots, n\}) \wedge (I(p_i, t_k) > 0)} (\{m_i\} + s_k, m_j) \right\} \\ (\forall p_j \in P) \wedge O(t_k, p_i) > 0 \wedge (m_j^l \neq \langle 0, 0, 0, 0 \rangle), \quad (15.19)$$

где \max , \min и $+$ – нечеткие операции для трапециевидных нечетких интервалов;

- если некоторые из позиций $p_i \in P$ являются одновременно входными и выходными для активного перехода t_k , то для них элементы вектора новой маркировки \hat{m}_{l+1} вычисляются последовательно, сначала по выражению (15.18), затем – по выражению (15.19);
- для каждой из позиций $p_i \in P$, не являющейся ни входной, ни выходной:

$$m_i^{l+1} = m_i^l (\forall p_i \in P) \wedge (I(p_i, t_k) = 0) \wedge O(t_k, p_i) > 0. \quad (15.20)$$

Я₄: правило нечеткой задержки маркеров в позициях. После нечеткого срабатывания активного, в соответствии с правилом Я₃, маркеры в вы-

ходных позициях для новой маркировки \hat{m}_{l+1} в общем случае являются недоступными. На них начинают действовать временные задержки в соответствующих выходных позициях сработавшего перехода, определяемые вектором \hat{z} . Соответствующие маркеры становятся доступными только после окончания действия временных задержек, которые определяют доступную маркировку \hat{m}'_{l+1} по формуле

$$m_j^{(l+1)'} = m_j^{l+1} + z_j \ (\forall p_j \in P) \wedge O(t_k, p_j) \wedge m_j^{l+1} \neq <0, 0, 0, 0>. \quad (15.21)$$

Пример. Допустим, на рис. 15.3 задана структура НСП C_{PT}^v .

Элементы вектора начальной маркировки \hat{m}_0 представляют собой трапециевидные нечеткие интервалы: $m_1^0 = <1, 2, 1, 1>$, $m_2^0 = <2, 3, 0, 1>$, $m_3^0 = m_4^0 = m_5^0 = <0, 0, 0, 0>$. В соответствии с правилом \mathcal{R}_4 на начальную маркировку начинают действовать временные задержки маркеров в позициях данной НСП. Поэтому, согласно выражению (15.21), доступной начальной маркировкой будет вектор \hat{m}'_0 с элементами: $m_1' = <3, 5, 2, 2>$, $m_2' = <5, 8, 1, 2>$, $m_3' = m_4' = m_5' = <0, 0, 0, 0>$. В соответствии с правилом \mathcal{R}_2 (15.17) для доступной маркировки \hat{m}'_0 активными являются переходы t_1 и t_2 .

Нечеткое срабатывание перехода t_1 приводит к новой маркировке \hat{m}_1 , элементы вектора которой формируются по правилу \mathcal{R}_3 следующим образом:

$m_1^1 = <0, 0, 0, 0>$, так как позиция p_1 является входной для этого перехода;

$m_2^1 = <5, 8, 1, 2>$, так как позиция p_2 не является выходной для перехода t_1 ;

$m_3^1 = <3, 5, 2, 2> + <1, 2, 0, 1> = <4, 7, 2, 3>$, так как позиция p_3 является выходной для данного перехода;

$m_4^1 = <3, 5, 2, 2> + <1, 2, 0, 1> = <4, 7, 2, 3>$, так как позиция p_4 также является выходной для данного перехода;

$m_5^1 = <0, 0, 0, 0>$, так как позиция p_5 не является входной для перехода t_1 .

Нечеткое срабатывание перехода t_2 приводит к новой маркировке \hat{m}_2 , элементы вектора которой формируются по правилу \mathcal{R}_3 следующим образом:

$m_1^2 = m_2^2 = <0, 0, 0, 0>$, так как позиции p_1 и p_2 являются входными для перехода t_2 ;

$m_3^2 = m_5^2 = <0, 0, 0, 0>$ остаются без изменения, так как позиции p_3 и p_5 не являются выходными для перехода t_2 ;

$m_4^2 = \max\{<3, 5, 2, 2>, <5, 8, 1, 2>\} + <0, 1, 0, 1> = <5, 9, 1, 3>$, так как позиция p_4 является выходной для перехода t_2 .

После нечеткого срабатывания переходов t_1 и t_2 маркировки \hat{m}_1 и \hat{m}_2 являются недоступными, так как на маркеры в выходных позициях p_3 и p_4 действуют нечеткие временные задержки. В соответствии с правилом \mathcal{R}_4 доступными будут являться маркировки \hat{m}'_1 и \hat{m}'_2 с элементами для \hat{m}'_1 : $m_1^{1'} = m_5^{1'} = <0, 0, 0, 0>$, $m_2^{1'} = <5, 8, 1, 2>$, $m_3^{1'} = <5, 9, 2, 3>$, $m_4^{1'} = <6, 11, 3, 4>$; для \hat{m}'_2 : $m_1^{2'} = m_2^{2'} = m_3^{2'} = m_5^{2'} = <0, 0, 0, 0>$, $m_4^{2'} = <7, 13, 2, 4>$.

Отметим, что при доступной маркировке \hat{m}'_1 активным будет переход t_1 , а при доступной маркировке \hat{m}'_2 активных переходов нет.

15.5. Задачи анализа свойств нечетких сетей Петри

Исследование дискретных динамических систем с помощью нечетких сетей Петри заключается в интерпретации свойств этих систем через свойства конкретных НСП и в их анализе.

При этом если анализ структурных (статических) свойств нечетких сетей Петри осуществляется с точки зрения теории графов, то динамические характеристики нечетких сетей Петри определяют перечень задач анализа таких их свойств, как:

- *ограниченность* (позиция называется *k-ограниченной*, если количество маркеров в данной позиции не превышает натурального числа k , для всех маркировок, достижимых из начальной маркировки; НСП называется *k-ограниченной*, если все позиции этой сети являются *k-ограниченными*);
- *безопасность* (1-ограниченная позиция называется *безопасной позицией*. 1-ограниченная НСП является *безопасной НСП*);
- *сохраняемость* (НСП называется *сохраняющей*, если существует вектор коэффициентов, сумма покомпонентных произведений которых на соответствующие элементы вектора маркировки остается постоянной для любой достижимой маркировки; в случае равенства всех элементов вектора коэффициентов единице НСП является *строгого сохраняющей*, или *консервативной*);

- *устойчивость* (переход называется *устойчивым*, если он является активным при данной маркировке и никакой другой переход, также активный при этой маркировке, не может, сработав, сделать его неактивным; НСП является *устойчивой*, если все ее переходы являются устойчивыми);
- *непосредственная достижимость маркировок* (одна маркировка *непосредственно достижима* из другой, если она получается в результате срабатывания некоторого активного перехода);
- *достижимость маркировок* (одна маркировка *достижима* из другой, если существует конечная упорядоченная последовательность маркировок и соответствующая ей упорядоченная последовательность переходов, причем такие, что любая пара соседних маркировок удовлетворяет отношению непосредственного следования маркировок).

Трактовка этих свойств нечетких сетей Петри в общем случае различается в зависимости от типа НСП.

К задачам же анализа свойств нечетких сетей Петри относятся следующие задачи:

- *достижимости маркировки* (т. е. достижима ли конкретная маркировка из начальной маркировки);
- *достижимости подмаркировки* (т. е. существует ли достижимая маркировка, заданные элементы вектора которой равны соответствующим элементам выбранной маркировки; остальные элементы вектора могут быть произвольными);
- *покрываемости маркировки* (т. е. существует ли достижимая маркировка, элементы вектора которой не меньше элементов вектора заданной маркировки);
- *достижимости нуля* (т. е. достижима ли маркировка, все элементы вектора которой равны нулю, из начальной маркировки);
- *достижимости нуля в одной или нескольких позициях* (т. е. достижима ли маркировка, заданный элемент – элементы – вектора которой равен нулю, из начальной маркировки);
- *равенства множеств достижимых маркировок* (т. е. равны ли множества достижимых маркировок двух соответствующих НСП одного типа);

- подмножества достижимых маркировок (т. е. является ли одно из множеств достижимых маркировок одной НСП подмножеством допустимых маркировок другой НСП);
- *k-ограниченности позиций и НСП в целом* (т. е. существует ли натуральное число k , для которого выбранные позиции – или НСП в целом – являются k -ограниченными);
- *безопасности позиций и НСП в целом* (т. е. является ли данная позиция – или НСП в целом – безопасной);
- *устойчивости переходов* (т. е. является ли данная позиция – или НСП в целом – устойчивой);
- *активности переходов* (определение для заданного подмножества переходов уровней их активности: 0 – переход не бывает активным; 1 – переход потенциально активный, т. е. существует маркировка, при которой переход является активным; 2 – в последовательности срабатываемых переходов рассматриваемый переход присутствует ограниченное число раз; 3 – в последовательности срабатываемых переходов рассматриваемый переход присутствует неограниченное число раз; 4 – в течение всей последовательности срабатываемых переходов рассматриваемый переход является активным);
- *сохраняемости НСП* (определение, является ли НСП сохраняющей);
- *строгой сохраняемости (консервативности) НСП* (определение, является ли НСП строго сохраняющей, или консервативной);

Более подробную информацию об анализе свойств нечетких сетей Петри различных типов можно найти в [5]

Глава 16. НЕЧЕТКИЕ СИТУАЦИОННЫЕ СЕТИ

Нечеткая ситуационная сеть (НСС) представляет собой нечеткий ориентированный взвешенный граф переходов по нечетким эталонным ситуациям. Набор нечетких управляющих решений, необходимых для вывода решений по текущей ситуации, а также их последовательность для достижения целевой ситуации определяются стратегией управления – нечетким маршрутом в нечеткой ситуационной сети между текущей и целевой ситуациями.

Выбор целевой нечеткой ситуации из множества всех нечетких ситуаций может осуществляться либо на основе некоторой нечеткой про-

дукционной модели, либо непосредственно по НСС. Для построения НСС используются специальные нечеткие отношения, а также нечеткая производственная модель типа «нечеткая ситуация – нечеткое действие», определяющая степени соответствия управляющих решений нечетким ситуациям.

16.1. Формирование нечеткой ситуационной сети

Пусть $P = \{p_1, p_2, \dots, p_m\}$ – множество признаков, описывающих состояния анализируемой системы. Каждый признак p_i ($i \in \{1, 2, \dots, m\}$) описывается соответствующей лингвистической переменной:

$$\langle p_i, T_i, D_i \rangle, \quad (16.1)$$

где $T_i = \{T_1^i, T_2^i, \dots, T_{m_i}^i\}$ – терм-множество лингвистической переменной, m_i – число термов (нечетких значений) признака p_i ; D_i – базовое множество признака p_i . Для описания термов T_j^i ($i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, m_i\}$) используются нечеткие множества A_{ij} , заданные значениями функций принадлежности $\mu_{A_{ij}}(p_i)$ в соответствующих базовых множествах $p_i \in D_i$:

$$A_{ij} = \{(\mu_{A_{ij}}(p_i)/p_i)\}, \quad p_i \in D_i. \quad (16.2)$$

Нечеткой ситуацией \hat{s} называется нечеткое множество второго уровня:

$$\hat{s} = \{(\hat{s}(p_i)/p_i)\}, \quad i = 1, 2, \dots, m, \quad (16.3)$$

где

$$\hat{s}(p_i) = \{(\mu_{A_{ij}}(p_i')/T_j^i)\}, \quad j = 1, 2, \dots, m_i, \quad (16.4)$$

$\mu_{A_{ij}}(p_i')$ – значение функции принадлежности при заданном значении p_i .

Пример. Некоторая нечеткая ситуация, описывающая одно из нечетких состояний устойчивости системы при заданных значениях признаков {живучесть, надежность, помехоустойчивость}, может быть представлена следующим образом:

$$\begin{aligned} \hat{s}_k = & \{((0.1/\text{«низкая»}), (0.7/\text{«средняя»}), (0.3/\text{«высокая»}) / \text{«живучесть»}); \\ & ((0.8/\text{«низкая»}), (0.4/\text{«средняя»}), (0.3/\text{«высокая»}) / \text{«надежность»}); \\ & ((0.2/\text{«низкая»}), (0.9/\text{«высокая»}) / \text{«помехоустойчивость»})\}. \end{aligned}$$

Нечеткой ситуационной сетью называют нечеткий ориентированный взвешенный граф:

$$G = (S, R, \alpha), \quad (16.5)$$

где $\hat{S} = \{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_w\}$ – вершины НСС, соответствующие эталонным нечетким ситуациям; $R = (R_1, \dots, R_f)$ – множество дуг между вершинами, представляющих возможные управляющие решения; $\alpha(\hat{s}_i, R_j)$ – степень предпочтения применения управляющего решения R_j в ситуации \hat{s}_i по сравнению с другими управляющими решениями из множества R [6].

На рис. 16.1 показан фрагмент нечеткой ситуационной сети. Степени предпочтения управляющих решений либо неизменны в каждой ситуации и определяются экспертами, либо зависят от ситуации.

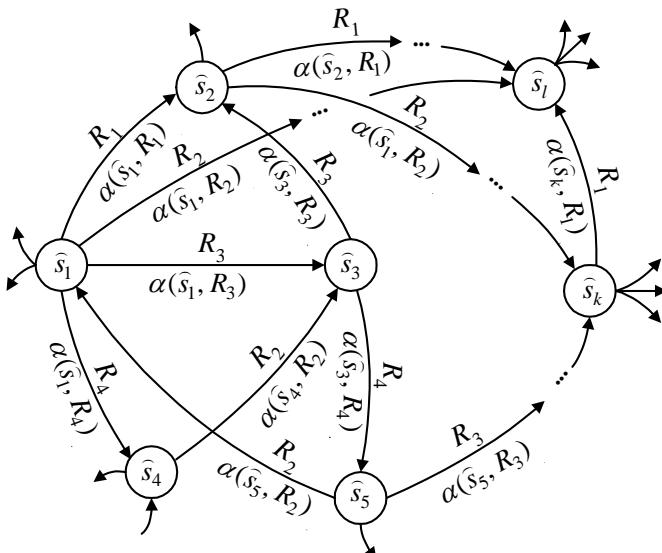


Рис. 16.1. Фрагмент нечеткой ситуационной сети

Стоит отметить, что ограниченный набор эталонных нечетких ситуаций позволяет описать неограниченное число состояний анализируемой системы. Существует два подхода к формированию нечеткой ситуационной сети: прямой и обратный.

Прямой подход к построению НСС заключается в моделировании переходов системы из одной нечеткой ситуации в другую. Для этого предварительно выявляется множество возможных управляющих решений $R = (R_1, R_2, \dots, R_f)$. Эти управляющие решения задаются в виде нечетких отношений определенного типа между значениями (термами) $T_i = \{T_1^i, T_2^i, \dots, T_{m_i}^i\}$ признаков $P = \{p_1, p_2, \dots, p_m\}$.

Для каждой нечеткой ситуации из множества возможных нечетких ситуаций $\hat{s}_k \in \hat{S}$ формируется подмножество ситуаций $\hat{S}^k \subseteq \hat{S}$, в которые может перейти система из ситуации \hat{s}_k под воздействием управляющих решений из множества R . После чего вершина НСС \hat{s}_k соединяется дугами с каждой вершиной из \hat{S}^k . Дуги определяют соответствующие решения и степени их предпочтения в ситуации \hat{s}_k . Данная процедура повторяется для всех нечетких ситуаций из множества \hat{S} . Ситуация же \hat{s}_l , в которую переходит система из ситуации \hat{s}_k , под воздействием управляющего решения $R_y \in R$ определяется в результате композиции \hat{s}_k с нечетким отношением, задающим решение R_g .

Пример. Построим фрагмент НСС, характеризующей изменение состояния системы (ее устойчивости), описываемой тремя признаками $\{p_1 - \text{живучесть}, p_2 - \text{надежность}, p_3 - \text{помехоустойчивость}\}$. Пусть заданы следующие терм-множества этих признаков:

$$T_1 = \{ T_1^1 - \text{«низкая»}, T_2^1 - \text{«средняя»}, T_3^1 - \text{«высокая»} \};$$

$$T_2 = \{ T_1^2 - \text{«низкая»}, T_2^2 - \text{«средняя»}, T_3^2 - \text{«высокая»} \};$$

$$T_3 = \{ T_1^3 - \text{«низкая»}, T_2^3 - \text{«высокая»} \}.$$

Допустим, управляющие решения по признакам заданы следующими терм-множествами:

$$R_1 = \{ R_1^1 - \text{«сильно увеличить»}, R_2^1 - \text{«немного увеличить»}, R_3^1 - \text{«не изменять»}, R_4^1 - \text{«немного уменьшить»}, R_5^1 - \text{«сильно уменьшить»} \};$$

$$R_2 = \{ R_1^2 - \text{«сильно увеличить»}, R_2^2 - \text{«немного увеличить»}, R_3^2 - \text{«не изменять»}, R_4^2 - \text{«немного уменьшить»}, R_2^2 - \text{«сильно уменьшить»} \};$$

$$R_3 = \{ R_1^3 - \text{«увеличить»}, R_2^3 - \text{«не изменять»}, R_3^3 - \text{«уменьшить»} \}.$$

Так как управляющие решения задают некоторые преобразования значений признаков, то каждому j -му терму управляющих решений ставится в соответствие матрица отношений M_j^i , описывающая силу воздействия управляющего решения R_j^i на значения данного признака p_i .

Например, матрицы отношений, описывающих воздействие управляющих решений $R_1 = \{ R_1^1, R_2^1, R_3^1, R_4^1, R_5^1 \}$ на признак p_1 , следующие:

$$\begin{aligned}
 & T_1^1 \quad T_2^1 \quad T_3^1 \quad \quad \quad T_1^1 \quad T_2^1 \quad T_3^1 \\
 M_1^1 = & \begin{matrix} T_1^1 & 0.1 & 0.2 & 1 \\ T_2^1 & 0 & 0.1 & 1 \\ T_3^1 & 0 & 0 & 1 \end{matrix}, \quad M_2^1 = \begin{matrix} T_1^1 & 0.1 & 1 & 0.2 \\ T_2^1 & 0 & 0.1 & 1 \\ T_3^1 & 0 & 0 & 1 \end{matrix}, \\
 & T_1^1 \quad T_2^1 \quad T_3^1 \quad \quad \quad T_1^1 \quad T_2^1 \quad T_3^1 \quad \quad \quad T_1^1 \quad T_2^1 \quad T_3^1 \\
 M_3^1 = & \begin{matrix} T_1^1 & 1 & 0.1 & 0 \\ T_2^1 & 0 & 1 & 0 \\ T_3^1 & 0 & 0.1 & 1 \end{matrix}, \quad M_4^1 = \begin{matrix} T_1^1 & 0.2 & 0 & 0 \\ T_2^1 & 1 & 0.1 & 0 \\ T_3^1 & 0.2 & 1 & 0.1 \end{matrix}, \quad M_5^1 = \begin{matrix} T_1^1 & 0.1 & 0 & 0 \\ T_2^1 & 1 & 0.1 & 0 \\ T_3^1 & 1 & 0.2 & 0.1 \end{matrix}.
 \end{aligned}$$

Матрицы отношений, описывающих воздействие управляющих решений $R_2 = \{R_1^2, R_2^2, R_3^2, R_4^2, R_5^2\}$ на признак p_2 , следующие:

$$\begin{aligned}
 & T_1^2 \quad T_2^2 \quad T_3^2 \quad \quad \quad T_1^2 \quad T_2^2 \quad T_3^2 \\
 M_1^2 = & \begin{matrix} T_1^2 & 0.1 & 0.2 & 1 \\ T_2^2 & 0 & 0.1 & 1 \\ T_3^2 & 0 & 0 & 1 \end{matrix}, \quad M_2^2 = \begin{matrix} T_1^2 & 0.1 & 1 & 0.2 \\ T_2^2 & 0 & 0.1 & 1 \\ T_3^2 & 0 & 0 & 1 \end{matrix}, \\
 & T_1^2 \quad T_2^2 \quad T_3^2 \quad \quad \quad T_1^2 \quad T_2^2 \quad T_3^2 \\
 M_3^2 = & \begin{matrix} T_1^2 & 1 & 0.1 & 0 \\ T_2^2 & 0 & 1 & 0 \\ T_3^2 & 0 & 0.1 & 1 \end{matrix}, \quad M_4^2 = \begin{matrix} T_1^2 & 0.2 & 0 & 0 \\ T_2^2 & 1 & 0.1 & 0 \\ T_3^2 & 0.2 & 1 & 0.1 \end{matrix}, \\
 & T_1^2 \quad T_2^2 \quad T_3^2 \\
 M_5^2 = & \begin{matrix} T_1^2 & 0.1 & 0 & 0 \\ T_2^2 & 1 & 0.1 & 0 \\ T_3^2 & 1 & 0.2 & 0.1 \end{matrix}.
 \end{aligned}$$

Матрицы отношений, описывающих воздействие управляющих решений $R_3 = \{R_1^3, R_2^3, R_3^3\}$ на признак p_3 , следующие:

$$M_1^3 = \begin{matrix} T_1^3 & T_2^3 \\ T_1^3 & 0.2 & 1 \\ T_2^3 & 0 & 1 \end{matrix}, \quad M_2^3 = \begin{matrix} T_1^3 & T_2^3 \\ T_1^3 & 1 & 0.1 \\ T_2^3 & 0.1 & 1 \end{matrix}, \quad M_3^3 = \begin{matrix} T_1^3 & T_2^3 \\ T_1^3 & 0.2 & 0 \\ T_2^3 & 1 & 0.2 & 0.2 \end{matrix}.$$

Покажем, каким образом определяется результат воздействия управляющего решения на нечеткое значение признака. Пусть задано нечеткое значение признака: $p_2 = \{(0.1/\text{«низкая»}), (0.7/\text{«средняя»}), (0.3/\text{«высокая»})\}$.

Определим новое нечеткое значение признака p_2 после применения управляющего решения R_1^1 – «сильно увеличить». Для этого выполняется min-max-композиция вектора $(0.1, 0.7, 0.1)$ и матрицы M_1^1 . В результате формируется вектор $(0.1, 0.1, 0.7)$, задающий новое нечеткое значение признака: p_2 , который задает новую нечеткую ситуацию \hat{s}_l :

$$\begin{aligned}\hat{s}_l = \{ & ((0.1/\text{«низкая»}), (0.7/\text{«средняя»}), (0.3/\text{«высокая»}) / \text{«живучесть»}); \\ & (0.1/\text{«низкая»}), (0.1/\text{«средняя»}), (0.7/\text{«высокая»}) / \text{«надежность»}); \\ & (0.2/\text{«низкая»}), (0.9/\text{«высокая»}) / \text{«помехоустойчивость»} \}.\end{aligned}$$

Данная ситуация включается в формируемую НСС, если выполнены дополнительные условия по заданной степени нечеткого различия (ближности) образуемых нечетких ситуаций.

Кроме того, необходимо учитывать дополнительные проблемы, возникающие при определении воздействий управляющих решений, в случаях, если необходимо учитывать их взаимосвязь. Они заключаются в том, что применение одного управляющего решения по отдельному признаку может вызвать изменение значений других признаков, что в конечном итоге соответствует применению управляющих решений по этим признакам.

Описание путей решения этих проблем выходит за рамки тематики данной книги. Более подробно они рассмотрены в [6].

Обратный подход к построению НСС заключается в том, что на множестве эталонных нечетких ситуаций задается некоторое отношение, граф которого отражает возможные переходы из ситуации в ситуацию. И нечеткая ситуационная сеть формируется посредством задания необходимых для этих переходов управляющих решений и степеней их предпочтения.

Данный подход к формированию НСС достаточно разработан только для ситуаций, характеризующихся взаимной независимостью значений признаков. В этом случае для определения множества эталонных нечетких ситуаций используется отношение общности ситуаций.

Введем необходимые для иллюстрации данного подхода определения нечеткого равенства и нечеткой общности ситуаций [6].

Степень нечеткого равенства $\mu(\hat{s}_k, \hat{s}_l)$ ситуаций \hat{s}_k и \hat{s}_l , обобщенная на случай недостаточной определенности ситуаций, задается выражением

$$\mu(\hat{s}_k, \hat{s}_l) = \bigwedge_{p_i \in P} \mu(\mu_{s_k}(p_i), \mu_{s_l}(p_i)), \quad (16.6)$$

где

$$\mu(\mu_{s_k}(p_i), \mu_{s_l}(p_i)) = \bigwedge_{T_j^i \in T_l} C((\mu_{s_k}(p_i)/T_j^i), (\mu_{s_l}(p_i)/T_j^i)), \quad (16.7)$$

$$C((\mu_{s_k}(p_i)/T_j^i), (\mu_{s_l}(p_i)/T_j^i)) = \begin{cases} (\mu_{s_k}(p_i)/T_j^i) \leftrightarrow (\mu_{s_l}(p_i)/T_j^i), \\ \text{если } (\mu_{s_k}(p_i)/T_j^i) \notin (1-t, t) \text{ И} \\ \quad (\mu_{s_l}(p_i)/T_j^i) \notin (1-t, t); \\ 1, \text{ если } (\mu_{s_k}(p_i)/T_j^i) \in (1-t, t) \text{ ИЛИ} \\ \quad (\mu_{s_l}(p_i)/T_j^i) \in (1-t, t). \end{cases} \quad (16.8)$$

$$\begin{aligned} & (\mu_{s_k}(p_i)/T_j^i) \leftrightarrow (\mu_{s_l}(p_i)/T_j^i) = \\ & = \min((\max(1 - (\mu_{s_k}(p_i)/T_j^i)), (\mu_{s_l}(p_i)/T_j^i))), \\ & \quad (\max(1 - (\mu_{s_l}(p_i)/T_j^i)), (\mu_{s_k}(p_i)/T_j^i))). \end{aligned} \quad (16.9)$$

Нечеткой $(g - q)$ -общностью ситуации называется такое сходство ситуаций, при котором нечеткие значения всех признаков в ситуациях нечетко равны, кроме нечетких значений не более чем q признаков. Если ситуации \hat{s}_k и \hat{s}_l описываются p признаками, то для их $(g - q)$ -общности достаточно нечеткого равенства $(g - q)$ признаков из множества P .

Степень нечеткой $(g - q)$ -общности $k_{g-q}(\hat{s}_k, \hat{s}_l)$ ситуаций \hat{s}_k и \hat{s}_l определяется выражением

$$k_{g-q}(\hat{s}_k, \hat{s}_l) = \bigwedge_{p_i \in P \setminus P_q} \mu(\mu_{s_k}(p_i), \mu_{s_l}(p_i)), \quad (16.10)$$

где нечеткие ситуации \hat{s}_k и \hat{s}_l заданы следующим образом:

$$\hat{s}_k = \{(\mu_{s_k}(p_i)/p_i)\}, \hat{s}_l = \{(\mu_{s_l}(p_i)/p_i)\}, i = 1, 2, \dots, m, p_i \in P, \quad (16.11)$$

при этом $|P_q| \leq q$ и признак $p_i \in P_q$, если $\mu(\mu_{s_k}(p_i), \mu_{s_l}(p_i)) < t$.

То есть при определении $(g - q)$ -общности ситуаций \hat{s}_k и \hat{s}_l не учитываются признаки, имеющие нечетконеравные значения в \hat{s}_k и \hat{s}_l , если число этих признаков не превышает q . Следовательно, при $P_q = \emptyset$ ситуации \hat{s}_k и \hat{s}_l нечеткоравны.

Будем считать ситуации \hat{s}_k и \hat{s}_l имеющими $(g - q)$ -общность, если $k_{g-q}(\hat{s}_k, \hat{s}_l) \geq t$.

Пример. Пусть $P = \{p_1, p_2, p_3\}$, $T_1 = \{T_1^1, T_2^1, T_3^1\}$, $T_2 = \{T_1^2, T_2^2, T_3^2\}$, $T_3 = \{T_1^3, T_2^3\}$. Ситуации:

$$\begin{aligned}\hat{s}_k &= \{((0.1/T_1^1), (0.7/T_2^1), (0.3/T_3^1)/T_1); \\ &((0.8/T_1^2), (0.4/T_2^2), (0.3/T_3^2)/T_2); ((0.2/T_1^3), (0.9/T_2^3)/T_3)\}; \\ \hat{s}_l &= \{((0.3/T_1^1), (0.9/T_2^1), (0.1/T_3^1)/T_1); \\ &((0.2/T_1^2), (0.3/T_2^2), (0.8/T_3^2)/T_2); ((0.1/T_1^3), (0.8/T_2^3)/T_3)\}.\end{aligned}$$

Пусть задан порог $t = 0.7$.

Нечеткие ситуации имеют 2-общность ($p = 3, g = 1$) при $t = 0.7$, так как

$$\begin{aligned}\mu(\mu_{s_k}(p_1), \mu_{s_l}(p_1)) &= 0.7; \quad \mu(\mu_{s_k}(p_2), \mu_{s_l}(p_2)) = 0.2; \\ \mu(\mu_{s_k}(p_3), \mu_{s_l}(p_3)) &= 0.8.\end{aligned}$$

Следовательно, $p_2 \in P_q$, $|P_q| = 1$, $k_2(\hat{s}_k, \hat{s}_l) = 0.7$.

Исходя из определения, ситуации \hat{s}_k и \hat{s}_l имеют $(g - q)$ -общность, если нечеткие значения всех признаков, кроме q признаков, в этих ситуациях нечеткоравны. Следовательно, применяя не более q управляющих решений, действующих на нечеткое значение только одного признака, можно перейти из ситуации \hat{s}_k в ситуацию \hat{s}_l и наоборот.

Вначале формируется граф, вершинами которого являются все эталонные нечеткие ситуации, обладающие $(g - 1)$ -общностью при заданном пороге нечеткого равенства ($t \geq 0.6 \div 1$). В графе две ситуации \hat{s}_k и \hat{s}_l соединяются дугой от \hat{s}_k к \hat{s}_l , если эти ситуации имеют $(g - 1)$ -общность по признакам. Затем для окончательного формирования НСС дуги переходов из состояния в состояние нагружаются соответствующими управляющими решениями и степенями их предпочтения.

Для определения нечеткого управляющего решения, применение которого при заданном значении p_i^k признака p_i приводит к известному нечеткому значению этого признака p_i^l , необходимо задать матрицу нечеткого отношения, определяющее это управляющее решение.

Пример. Допустим значение признака p_i в ситуации \hat{s}_k :

$$p_i^k = \{(0.8/\text{«низкая»}), (0.4/\text{«средняя»}), (0.2/\text{«высокая»})\},$$

при переходе в ситуацию \hat{s}_l под воздействием неизвестного управляющего решения изменяется следующим образом:

$$p_i^l = \{(0.2/\text{«низкая»}), (0.8/\text{«средняя»}), (0.4/\text{«высокая»})\}.$$

Для определения неизвестного управляющего решения формируется нечеткое отношение R_h , композиция которого с p_i^k приводит к p_i^k . Для чего вычисляется декартово произведение $p_i^k \times p_i^k$ и формируется матрица M_h отношения R_h :

	низкая	средняя	высокая
низкая	0.2	0.8	0.4
средняя	0.2	0.4	0.4
высокая	0.2	0.2	0.2

Данные действия повторяются для каждой пары нечетких ситуаций, обладающих $(g - 1)$ -общностью при заданном пороге нечеткого равенства.

Примечание. При построении НСС, отражающей переходы из состояния в состояние реальной системы, необходимо дополнительно решать вопросы разложения управляющих решений по задаваемому базису типовых управляющих решений и выбора решения(ий).

16.2. Задачи моделирования процессов управления на основе нечетких ситуационных сетей

Моделирование процессов управления на основе нечеткой ситуационной сети с использованием отношений специального типа (нечеткого включения, равенства, общности) включает в себя решение следующих основных, связанных между собой задач:

- формирование нечеткой ситуационной сети;
- идентификация и оценка текущей ситуации, т. е. поиск наиболее близкой к ней эталонной ситуации;
- постановка целевых ситуаций;
- формирование стратегии управления, т. е. оптимального в некотором смысле маршрута в НСС между текущей и целевой ситуациями.

Задача идентификации текущей ситуации решается следующим образом. Имеется набор эталонных описаний состояний системы в виде набора нечетких ситуаций – нечетких множеств второго уровня на множестве признаков системы. Текущее состояние системы, также представляющее собой нечеткую ситуацию, сопоставляется с эталонными нечеткими ситуациями на основе специальных мер близости (нечеткого включения, равенства, общности) и выбирается ситуация, наиболее близкая к текущей.

Задача определения целевой ситуации существенно влияет на поиск оптимального маршрута в НСС. Сложность решения этой задачи заключается в том, что трудно предвидеть достижимость целевой ситуации из произвольной входной ситуации. Степени предпочтения управляющих решений в НСС могут быть определены таким образом, что ситуация, наиболее подходящая в качестве целевой для выбранной исходной ситуации, часто недостижима из этой входной ситуации.

Набор решений, необходимых для перевода системы из текущей ситуации в целевую, определяется *задачей формирования стратегии управления*, т. е. заданием оптимального в некотором смысле маршрута в НСС между текущей (входной) и целевой ситуациями. Критерием оптимальности может быть, например:

- средний вес маршрута – отношение суммы предпочтения входящих в него дуг к количеству этих дуг;
- прочность маршрута – минимум степеней предпочтения дуг маршрута;
- длина маршрута – количество дуг маршрута.

Оптимальным будет маршрут, имеющий либо максимальный средний вес, либо максимальную прочность, либо минимальную длину маршрута.

В дополнение к вышеуказанным задачам моделирования процессов управления на основе нечеткой ситуационной сети также могут быть решены следующие задачи:

- последовательное уточнение управляющих решений (при использовании отношений нечеткого включения ситуаций);
- разбиение множества типовых ситуаций на классы нечеткой эквивалентности (с отношениями нечеткого равенства ситуаций);
- сокращение числа эталонных ситуаций, т. е. сжатие НСС при поиске стратегий управления (при использовании отношений нечеткого равенства ситуаций);
- объединение эталонных ситуаций в классы нечеткой общности (при использовании отношений нечеткой общности ситуаций).

Кратко рассмотрим те задачи из приведенного перечня, которые решаются зачастую по-разному в зависимости от типа нечетких отношений и реализация которых требует некоторого пояснения.

Отношение нечеткого включения ситуаций

Степень нечеткого включения ситуации \hat{s}_k в ситуацию \hat{s}_l из множества $\hat{S} = \{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_w\}$ определяется выражением

$$v(\hat{s}_k, \hat{s}_l) = \bigwedge_{p_i \in P} v(\mu_{\hat{s}_k}(p_i), \mu_{\hat{s}_l}(p_i)), \quad (16.12)$$

где

$$v(\mu_{\hat{s}_k}(p_i), \mu_{\hat{s}_l}(p_i)) = \bigwedge_{T_j^i \in T_i} (\mu_{s_k}(p_i)/T_j^i) \rightarrow (\mu_{s_l}(p_i)/T_j^i), \quad (16.13)$$

$$\begin{aligned} & (\mu_{s_k}(p_i)/T_j^i) \rightarrow (\mu_{s_l}(p_i)/T_j^i) = \\ & = \max(1 - (\mu_{s_k}(p_i)/T_j^i), (\mu_{s_l}(p_i)/T_j^i)). \end{aligned} \quad (16.14)$$

Ситуация \hat{s}_k нечетко включается в ситуацию \hat{s}_l ($\hat{s}_k \subset \hat{s}_l$), если $v(\hat{s}_k, \hat{s}_l) \geq \lambda$, где λ – некоторый порог включения. Обычно принимают $\lambda \in [0.6, 1]$.

Отношение нечеткого включения при хорошей определенности ситуаций из множества \hat{S} является отношением нечеткого нестрогого порядка [6]. Это делает возможным организацию на множестве \hat{S} иерархии ситуаций.

Для организации иерархии на множестве \hat{S} строится диаграмма Хасце [7], для чего в нечеткой ситуационной сети, описывающей систему, убираются транзитивно замыкающие дуги, а сами типовые ситуации разносятся по уровням. На верхнем уровне располагаются вершины, из которых не выходит ни одна дуга; а на каждом последующем уровне располагаются вершины, из которых выходят дуги, инцидентные вершинам на предыдущем уровне.

Задача идентификации и оценки текущей ситуации (поиск наиболее сходной к ней типовой ситуации). Поиск наиболее сходной с \hat{s}_0 ситуации \hat{s}_k , такой, что $(\hat{s}_0 \subset \hat{s}_k)$ и $(\forall \hat{s}_l \in \hat{S}) (\hat{s}_0 \subset \hat{s}_k) \rightarrow (\hat{s}_0 \subset \hat{s}_l)$, начинается с верхнего уровня иерархии в соответствии со следующим алгоритмом.

Этап 1. Находится ситуация \hat{s}_r , в которую нечетко включается \hat{s}_0 , т. е. $(\hat{s}_0 \subset \hat{s}_r)$.

Этап 2. Ситуация \hat{s}_k полагается равной \hat{s}_r .

Этап 3. Осуществляется переход к ситуациям на нижних, смежных с \hat{s}_r уровнях иерархии и т. д. Поиск заканчивается, если либо на нижнем уровне иерархии в ситуацию \hat{s}_k не включается ни одна ситуация из множества S , либо для любой ситуации \hat{s}_l , включающейся в ситуацию \hat{s}_k , выполняется условие $(\hat{s}_0 \subset \hat{s}_l)$.

$$v(\hat{s}_k, \hat{s}_l) = \bigwedge_{p_i \in P} v(\mu_{\hat{s}_k}(p_i), \mu_{\hat{s}_l}(p_i)), \quad (16.12)$$

где

$$v(\mu_{\hat{s}_k}(p_i), \mu_{\hat{s}_l}(p_i)) = \bigwedge_{T_j^i \in T_i} (\mu_{s_k}(p_i)/T_j^i) \rightarrow (\mu_{s_l}(p_i)/T_j^i), \quad (16.13)$$

$$\begin{aligned} & (\mu_{s_k}(p_i)/T_j^i) \rightarrow (\mu_{s_l}(p_i)/T_j^i) = \\ & = \max(1 - (\mu_{s_k}(p_i)/T_j^i), (\mu_{s_l}(p_i)/T_j^i)). \end{aligned} \quad (16.14)$$

Ситуация \hat{s}_k нечетко включается в ситуацию \hat{s}_l ($\hat{s}_k \subset \hat{s}_l$), если $v(\hat{s}_k, \hat{s}_l) \geq \lambda$, где λ – некоторый порог включения. Обычно принимают $\lambda \in [0.6, 1]$.

Отношение нечеткого включения при хорошей определенности ситуаций из множества \hat{S} является отношением нечеткого нестрогого порядка [6]. Это делает возможным организацию на множестве \hat{S} иерархии ситуаций.

Для организации иерархии на множестве \hat{S} строится диаграмма Хасце [7], для чего в нечеткой ситуационной сети, описывающей систему, убираются транзитивно замыкающие дуги, а сами типовые ситуации разносятся по уровням. На верхнем уровне располагаются вершины, из которых не выходит ни одна дуга; а на каждом последующем уровне располагаются вершины, из которых выходят дуги, инцидентные вершинам на предыдущем уровне.

Задача идентификации и оценки текущей ситуации (поиск наиболее сходной к ней типовой ситуации). Поиск наиболее сходной с \hat{s}_0 ситуации \hat{s}_k , такой, что $(\hat{s}_0 \subset \hat{s}_k)$ и $(\forall \hat{s}_l \in \hat{S}) (\hat{s}_0 \subset \hat{s}_k) \rightarrow (\hat{s}_0 \subset \hat{s}_l)$, начинается с верхнего уровня иерархии в соответствии со следующим алгоритмом.

Этап 1. Находится ситуация \hat{s}_r , в которую нечетко включается \hat{s}_0 , т. е. $(\hat{s}_0 \subset \hat{s}_r)$.

Этап 2. Ситуация \hat{s}_k полагается равной \hat{s}_r .

Этап 3. Осуществляется переход к ситуациям на нижних, смежных с \hat{s}_r уровнях иерархии и т. д. Поиск заканчивается, если либо на нижнем уровне иерархии в ситуацию \hat{s}_k не включается ни одна ситуация из множества S , либо для любой ситуации \hat{s}_l , включающейся в ситуацию \hat{s}_k , выполняется условие $(\hat{s}_0 \subset \hat{s}_l)$.

Задача последовательного уточнения управляющих решений. Данная задача решается аналогично вышерассмотренной. Поиск наиболее сходной с \hat{s}_0 ситуации \hat{s}_k выполняется аналогично, но при этом вместе с направленным поиском ситуации выдаются управляющие решения, поставленные в соответствие типовым ситуациям, по которым осуществляется перемещение между уровнями иерархии, вплоть до ситуации \hat{s}_k . Допустим, для определения \hat{s}_k строится цепочка ситуаций $\hat{s}_r, \hat{s}_g, \hat{s}_h, \dots, \hat{s}_q, \hat{s}_k$, где \hat{s}_r расположена на первом, а \hat{s}_k – на z -м уровне иерархии. По мере выстраивания цепочки ситуаций должна вырабатываться последовательность управляющих решений $R_r, R_g, R_h, \dots, R_q, R_k$ (R_i соответствует ситуации \hat{s}_i). Таким образом, по мере уточнения ситуации последовательно уточняются и управляющие решения.

Отношение нечеткого равенства ситуаций

Степень нечеткого равенства типовых нечетких ситуаций \hat{s}_k и \hat{s}_l из множества $S = \{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_w\}$ определяется выражением

$$\mu(\hat{s}_k, \hat{s}_l) = v(\hat{s}_k, \hat{s}_l) \wedge v(\hat{s}_l, \hat{s}_k). \quad (16.15)$$

То есть степень равенства типовых нечетких ситуаций равна минимальной из степеней их взаимного нечеткого включения.

Считается, что ситуации \hat{s}_k и \hat{s}_l нечеткоравны ($\hat{s}_k \approx \hat{s}_l$), если $\mu(\hat{s}_k, \hat{s}_l) \geq \lambda$, где λ – порог нечеткого равенства ситуаций, $\lambda \in [0.6, 1]$.

Задача разбиения множества типовых ситуаций \hat{S} на классы нечеткой эквивалентности. При разбиении на классы нечеткой эквивалентности в один класс будут входить нечеткоравные между собой ситуации из множества \hat{S} . При заданном пороге нечеткого равенства λ все ситуации одного класса эквивалентности можно считать одной ситуацией, которая получается нечетким объединением ситуаций, принадлежащих к данному классу.

Задача сокращения числа типовых ситуаций. Число новых типовых ситуаций соответствует числу выделенных в соответствии с порогом λ классов нечеткой эквивалентности.

Задача идентификации входной ситуации посредством ее сравнения эталонными ситуациями на нечеткое равенство. Условием решения данной задачи является то, что на входные (текущие) ситуации \hat{s}_0

хорошо определены относительно порога λ . В отличие от решения задачи идентификации входной ситуации на основе отношений нечеткого включения в данном случае входная ситуация \hat{s}_0 сравнивается на нечеткое равенство с ситуациями, полученными в результате разбиения множества типовых ситуаций \hat{S} на классы нечеткой эквивалентности.

Отношение нечеткой общности ситуаций

Задача объединения типовых ситуаций в классы нечеткой общности. Данная задача решается с целью сжатия нечеткой ситуационной сети в условиях недоопределенности ситуаций относительно порога нечеткого равенства (более подробно см. в [6]).

Глава 17. НЕЧЕТКИЕ КОГНИТИВНЫЕ КАРТЫ

Когнитивные карты представляют собой разновидность математических моделей для формализации проблемы, сложной системы в виде множества концептов, отображающих ее системные факторы (переменные), и выявления причинно-следственных отношений (отношений влияния) между ними с учетом воздействия на эти факторы или изменения характера отношений.

Они предназначены для моделирования и анализа в системах принятия решений [8–11], в системах управления [12], в теории игр [13], в мультиагентных технологиях [14, 15], в технологиях разработки геоинформационных систем [16], при анализе электрических схем [17], в технологиях виртуальных миров и мультимедийных приложений [18], для моделирования экономических и демографических проблем [19].

Эффективность использования существующих методов построения и использования когнитивных карт в качестве моделей слабоструктурированных проблем и сложных систем обусловлена:

- возможностью наглядного представления анализируемой проблемы или системы;
- отсутствием необходимости предварительной спецификации концептов и отношений влияния;
- конструктивностью, наглядностью и относительной легкостью интерпретации с их помощью причинно-следственных связей (отношений) между концептами;
- интегрированностью с методами оценок результатов анализа [20].

В настоящее время предприняты усилия по приданию методам когнитивного моделирования дополнительных возможностей, позволяющих интегрировать различные методы и технологии решения разнотипных эвристических и аналитических задач, возникающих при исследовании слабоструктурированных проблем и сложных систем [21].

17.1. Задачи построения и анализа когнитивных карт

Процесс формирования и использования когнитивных карт можно представить в виде последовательности следующих процедур:

- определение списка концептов (согласованного списка концептов в случае опроса группы экспертов);
- определение отношений причинности (влияния) между каждой парой концептов (согласованных отношений причинности);
- построение когнитивной карты;
- динамическое моделирование;
- анализ системных характеристик когнитивной карты;
- анализ устойчивости;
- обучение когнитивной карты.

Обобщая задачи построения и анализа когнитивных карт, можно отметить, что их формализация предполагает решение следующих вопросов:

- определение способов и средств описания состояний или значений концептов;
- определение способов и средств задания взаимовлияния концептов друг на друга, а также задание в том или ином виде причинно-следственных отношений между концептами;
- задание механизма аккумулирования влияния нескольких входных концептов на один выходной;
- определение способов передачи влияния (опосредованного влияния) по когнитивной карте;
- определение системных характеристик когнитивной карты;
- построение моделей системной динамики;
- задание алгоритма обучения когнитивной карты.

Примечание. Перечень и характер решаемых задач из вышеупомянутого набора для каждого конкретного типа когнитивной карты определяется как ее свойствами, так и необходимостью учета ее адекватности моделируемой системе.

17.2. Обзор способов построения и анализа когнитивных карт

17.2.1. Традиционные знаковые когнитивные карты

Впервые термин «когнитивные карты» (Cognitive Maps) предложил Толмен [22]. Аксельрод [23] описал когнитивные карты в виде направленного «знакового» (взвешенного двухуровневыми значениями) графа, применив эту модель в теории принятия решений в политике и в экономике.

Традиционные знаковые когнитивные карты задаются в виде ориентированного графа и представляют моделируемую систему в виде множества концептов, отображающих системные переменные, связанные между собой отношениями влияния. Эти отношения могут быть положительными, отрицательными или нейтральными, характеризующими соответствующее влияние концептов друг на друга [24].

Описание состояний или значений концептов

Существующие варианты построения когнитивных карт различают по используемым формам представления концептов в зависимости от особенностей решаемых задач при моделировании предметной области.

Под концептом в когнитивном моделировании могут пониматься переменные, факторы, действия, состояния, индикаторы и другие понятия, относящиеся к моделируемой предметной области. Концепты когнитивной карты могут рассматриваться в качестве системных переменных, значение которых и определяет состояние системы в целом.

В знаковых когнитивных картах концепты представляются лишь качественно, характеризуя, например, возрастание, убывание либо неизменность значений «концептуального» параметра.

Задание взаимовлияния концептов

Базовым элементом когнитивной карты является связь, описывающая влияние одного концепта K_i («входного» концепта/концепта-источника/концепта-причины) на другой концепт K_j («выходной» концепт/концепт-приемник, концепт-следствие). Направленность этой связи w_{ij} означает, что концепт-источник *влияет* на концепт-приемник, т. е. изменение значений (состояний) концепта-источника приводит к изменению значений (состояний) концепта-приемника.

В знаковых когнитивных картах этот «механизм» в простейшем случае представлен двумя типами связи: положительной и отрицательной. Передача влияния рассматривается качественно: при положительной связи и увеличении концепта K_i концепт K_j увеличивается, при уменьшении $K_i - K_j$ уменьшается. При отрицательном значении связи увеличение значения K_i вызовет уменьшение K_j (и наоборот).

Анализ устойчивости

Анализ устойчивости когнитивных карт в широком смысле заключается в определении сбалансированных, согласованных, устойчивых когнитивных структур и в концептуальном плане базируется на основных положениях теорий социальной психологии: когнитивного диссонанса Фестингера, структурного баланса Хайдера, коммуникационных актов Ньюкома. Данные исследования, а также результаты работ по многомерному шкалированию и иерархическому кластерному анализу [25, 26] позволяют использовать для анализа устойчивости когнитивных структур свойства отношений (прежде всего свойство транзитивности) между концептами карты.

В [27] предложена конструктивная процедура вычисления системных характеристик когнитивной карты: совместного консонанса, диссонанса и влияния, основанная на сравнении контуров, образованных из концептов карты по критерию соответствия, баланса и силы влияния.

При поддержке принятия решения знаковые когнитивные карты используются в основном для прогноза и оценки взаимовлияния концептов друг на друга. По ней можно определить, что произойдет с системой, если изменится состояние любого или любых концептов, не станут ли некоторые концепты неприемлемыми в контексте реальных объектов предметной области.

Во многих случаях причинный концепт (из которого выходит дуга) является одновременно и концептом результата, т. е. в него входит дуга выходящего из другого причинного концепта. Путь в карте начинается с концепта и через несколько вершин может вернуться к этому же концепту. Устойчивость когнитивной карты определяется характером его обратных связей. Значение начальной переменной концепта при проходе по дугам карты только со знаком «+» может быть сильно увеличено за счет образованной положительной обратной связи. Если в когнитивной карте есть несколько таких положительных циклов, это может привести к нестабильности системы даже при незначительном начальном изменении. Наличие в карте отрицательных обратных связей часто ведет к стабилизации модели. Однако это также может вызвать все большее колебание модели и, как следствие, другую нестабильную ситуацию.

Общие правила определения путей в когнитивной карте следующие:

- в цикле существует положительная обратная связь, если в ней четное число знаков «-» или их нет, и отрицательная, если число дуг со знаком «-» нечетное;
- общая обратная связь между двумя концептами есть сумма обратных связей в циклах на всем пути между ними (даже если это один и тот же концепт).

Когнитивная карта может быть использована для качественной оценки влияния отдельных концептов на устойчивость анализируемой системы в целом, например в случае, если будет оказано достаточно сильное влияние на один или несколько концептов или изменится знак дуги, характеризующий характер отношения причинности между концептами.

Система, описываемая данной когнитивной картой, является устойчивой, если значения всех тех переменных концептов, на которые не оказывают влияния внешние воздействия, не изменяются под влиянием любых изменений значений одной или нескольких переменных, вызванных воздействием внешних причин.

Переменная является стабильной начальной переменной, если резкое увеличение ее значения не приведет к увеличению значений других переменных системы. Как правило, стабильными оказываются те начальные переменные, которые не влияют на другие переменные либо непосредственно влияют на те, которые не влияют на другие.

Одно из преимуществ традиционного когнитивного моделирования заключается в возможности моделирования и оценке применения различных качественных стратегий при принятии решений. Когнитивные карты позволяют оценить возможные последствия решений и определить, могут ли эти решения destabilизировать систему, и какие качественные изменения произойдут в концептах после того, как эти решения будут выполнены.

В основе анализа устойчивости (определения совместного консонанса) традиционных когнитивных карт, по сути, лежит известная методология анализа обычных линейных систем, основанная на сравнении различных контуров, образованных из концептов такой карты. Однако подобный анализ не позволяет выявить особенности взаимовлияния концептов, определить и ранжировать по степени влияния положительные и отрицательные тенденции, особенно в условиях многоцелевого задания критериев.

Кроме того, ограниченность алфавита знаковых когнитивных карт для представления отношений между концептами позволяет исследовать характер протекающих процессов лишь качественно, на основе анализа тенденций изменения системных переменных.

17.2.2. Развитие знаковых когнитивных карт

Недостатки традиционных знаковых когнитивных карт привели к возникновению целого ряда их разновидностей, расширяющих возможности и позволяющих проводить количественный анализ сложных систем.

Описание состояний или значений концептов

Необходимо отметить, что во всех модификациях знаковых когнитивных карт состояние или значение концептов количественно не определено. Можно лишь говорить о качественных тенденциях: концепт увеличивается, уменьшается, изменяется, остается без изменений.

Задание взаимовлияния концептов

В ряде работ предложено развитие знаковых когнитивных карт с точки зрения введения дополнительных видов связей между концептами, например неотрицательной, неположительной, неизвестной, амбивалентной. При этом данные связи характеризуются значениями из некоторого конечного набора [28, 29].

Аккумулирование влияния нескольких концептов на один концепт

В работах [28, 29] для знаковых когнитивных карт разработана специальная алгебра, позволяющая осуществить аккумулирование влияний нескольких концептов на один концепт, а также распространить данное влияние по когнитивной карте. Однако оценка этого влияния все равно остается качественной, что не позволяет проводить количественный (конструктивный) анализ поведения сложной системы.

Анализ устойчивости

Анализ устойчивости когнитивной карты во многих случаях является очень полезным, однако совершенно недостаточным для принятия решения по возникшей проблеме. Поэтому в настоящее время предприняты усилия по дополнению метода знаковых когнитивных карт новыми конструктивными механизмами, направленными на углубленный анализ распространения влияния за счет расширения типов отношений между концептами при сохранении конечного числа значений (уровней) этих отношений [28, 29].

17.2.3. «Нечеткие» когнитивные карты Коско

В работе [30] Коско предложил новый тип когнитивных карт, получивших название «нечеткие когнитивные карты» (Fuzzy Cognitive Maps).

Описание состояний или значений концептов

Концепты в нечеткой когнитивной карте (НКК) могут принимать значения из диапазона действительных чисел $[0, 1]$.

Термин «нечеткие» обозначает только то, что причинные связи (связи взаимовлияния) могут принимать не только значение, равное 0 или 1, а лежат в диапазоне действительных чисел, отражающих «силу» влияния одного концепта на другой. Подход, основанный на теории нечетких

множеств Заде, по крайней мере в вычислительном аспекте, в модели Коско не применяется.

В работах [31, 32] предложен подход, расширяющий диапазон изменения значений концептов нечеткой когнитивной карты до $[-1, 1]$.

Задание взаимовлияния концептов

В работе [30] проблема учета отрицательных весов влияния (их представления в диапазоне $[-1, 1]$) решается с помощью «расщепления» концепта на концепт и дисконцепт. При этом отрицательные влияния преобразуются в положительные с таким же абсолютным значением, но влияющие на дисконцепт. Таким образом, концепт учитывает все положительные влияния, а дисконцепт – отрицательные.

Несмотря на то что когнитивные карты этого типа названы нечеткими, тем не менее их нельзя считать таковыми в полной мере, так как обычные нечеткие множества определены в системе: базовое множество – степень принадлежности. В нечетких же когнитивных картах Коско для описания влияния используется только одна из этих «координат», что обедняет возможности моделирующих средств при исследовании свойств реальных сложных систем.

Аккумулирование влияния нескольких концептов на один концепт

Структура влияния нескольких входных концептов на выходной в картах данного типа соответствует структуре однослоиного персептрона [33]. В работе [30] предложен способ аккумулирования отдельных влияний, аналогичный взвешенному суммированию компонентов входного вектора искусственным нейроном с последующим нелинейным преобразованием результатов этого суммирования. Отдельные четкие влияния входных концептов суммируются, и для предотвращения выхода за пределы диапазона выходного концепта используется специальная нелинейная функция f

$$K_j = f \left[\sum_{i=1}^N w_{ij} K_i \right], \quad (17.1)$$

где K_i и K_j – значения входного и выходного концептов; N – число концептов, непосредственно влияющих на концепт j ; w_{ij} – вес влияния концепта i на концепт j .

Функция f [...] используется для нормализации результата вычислений в диапазоне $[0, 1]$. Обычно применяется сигмоидальная функция

$$f(x) = \frac{1}{1 + e^{-\lambda x}}, \quad \lambda > 0. \quad (17.2)$$

Несмотря на то что в вычислительном аспекте нечеткие когнитивные карты Коско похожи на искусственную нейронную сеть, однако между этими двумя моделями существуют различия. Нечеткие когнитивные карты могут носить чисто экспертный характер (хотя могут и обучаться) и соответствуют модели типа «белого ящика», тогда как искусственная нейронная сеть принципиально ориентирована на обучение (модель типа «черного ящика»).

Для учета распространения отрицательных влияний концептов друг на друга (так как $w_{ij} \in [-1, 1]$) используется понятие дисконцепта. При этом отрицательные влияния преобразуются в положительные с тем же абсолютным значением, но влияющие на дисконцепты: $K_i \xrightarrow{-} K_j$ заменяется на $K_i \xrightarrow{+} \bar{K}_j$.

Данное решение хотя и удваивает количество концептов, однако позволяет корректно и независимо обрабатывать влияния разного знака.

Моделирование динамики и анализ устойчивости

$$K_j(t+1) = f \left[\sum_{i=1}^N w_{ij} K_i(t) \right]. \quad (17.3)$$

Выражение (17.3) позволяет рассмотреть функционирование нечеткой когнитивной карты в дискретном времени.

Состояние системы описывается вектором состояний концептов в пространстве $[0, 1]^N$ или $[-1, 1]^N$, N – число концептов НКК. Умножение этого вектора на квадратную матрицу смежности для графа НКК, элементы которой – веса влияний, описывают изменения в системе (ее фазовый портрет). Как показано в [18], асимптотически эти изменения НКК могут приводить к устойчивому состоянию, предельному циклу и странному аттрактору.

17.2.4. Развитие «нечетких» когнитивных карт Коско

За исторически первой статьей Коско [30] появился целый ряд работ, в которых предложены различные модификации «нечетких» когнитивных карт для моделирования сложных систем.

Описание состояний или значений концептов

В работах [34–36] концепты когнитивных карт представляются на основе шкалы действительных чисел, ограниченных в некотором диапазоне, например $[-1, 1]$.

Задание взаимовлияния концептов

В работах [35, 36] в качестве диапазона значений силы связи w_{ij} используется интервал действительных чисел от -1 (самая сильная отрицательная) до $+1$ (самая сильная положительная). Чем сильнее связь, тем сильнее положительное или отрицательное влияние входного концепта на выходной, и наоборот. Таким образом, в данной модели сила связи между концептами представляет собой действительное значение из интервала $[-1, 1]$. И при условии, что «значения» концептов также могут быть приведены к безразмерной шкале действительных чисел в диапазоне $[-1, 1]$ (см. выше), передача влияния концепта i на концепт j может выражаться количественно:

$$K_j = w_{ij} K_i. \quad (17.4)$$

Аккумулирование влияния нескольких концептов на один концепт

В четких когнитивных картах [35, 36] процедура аккумулирования влияния нескольких концептов на один реализована на основе обычного алгебраического сложения отдельных влияний. Они имеют следующий вид:

$$K_j = \sum_{i=1}^N w_{ij} K_i, \quad (17.5)$$

где K_i и K_j – значения входного и выходного концептов; N – число концептов, непосредственно влияющих на концепт j ; w_{ij} – вес влияния концепта i на концепт j .

Особенностью когнитивных карт в работах [35, 36] является отсутствие нормализации результирующего значения концепта K_j . Хотя это и позволяет более корректно (количественно) учитывать свойства системных переменных, однако приводит к проблеме выхода результата за пределы диапазона $[-1, 1]$.

Модели системной динамики

В работах [37, 38] рассмотрен подход, позволяющий учитывать наличие петель в графе нечеткой когнитивной карты. При этом выражение для определения передачи влияния между концептами и моделирования динамики принимает следующий вид:

$$K_j(t+1) = f \left[\sum_{i=1}^N w_{ij} K_i(t) + K_j(t) \right]. \quad (17.6)$$

Выражение (17.6) предполагает, что такая нечеткая когнитивная карта может обладать памятью состояний, что, помимо прочего, позволяет «сглаживать» изменения значений концептов в процессе моделирования.

В этих же работах для описания причинно-следственных соотношений между каждой парой концептов строится нелинейное функциональное отображение на основе способа нечеткого логического вывода Цукамото. Такой подход позволяет представить причинное отношение между входным и выходным концептами в виде некоторого отображения g_{ij} , которое собственно и представляет вес влияния w_{ij} . Формула (17.6) приобретает следующий вид:

$$K_j(t+1) = f \left[\sum_{i=1}^N g_{ij} K_i(t) + K_j(t) \right]. \quad (17.7)$$

Однако значения концептов при этом остаются четкими числами, что ограничивает возможности данной модели.

17.2.5. Нечеткие когнитивные карты Силова

В работе [39] для моделирования нечеткоцелевых динамических систем предложено использовать нечеткие когнитивные карты, характеризующиеся следующими особенностями.

Задание взаимовлияния концептов

Отношения между концептами нечеткой когнитивной карты данного типа представлены, как и в модели Коско, в виде весов $w_{ij} \in [-1, 1]$, но они рассматриваются как элементы *нечеткой* матрицы смежности для графа НКК. Так как веса могут быть отрицательными, а операции над нечеткими множествами определены для функции принадлежности на $[0, 1]$, то существует проблема обработки отрицательных влияний, аналогичная модели Коско. В [39] она решается подобным образом за счет удвоения мощности множества концептов и раздельной обработки положительных и отрицательных влияний.

Нечеткие значения выходного концепта получаются с использованием характерных для нечеткой логики операций Т-норм (см., например, [40]) над нечеткими значениями входных концептов и весов влияния. Наиболее употребительными разновидностями Т-норм являются операции минимума (*min*) и алгебраического произведения (*prod*). Так, при использовании связки *min* передача непосредственного влияния концепта K_i на концепт K_j осуществляется следующим образом:

$$\tilde{K}_j = \min(\tilde{K}_i, w_{ij}). \quad (17.8)$$

Аккумулирование влияния нескольких концептов на один концепт

В работе [39] для решения задач аккумулирования влияния нескольких концептов на один концепт нечеткой когнитивной карты и определения опосредованного влияния концептов предложена нечеткая матричная регулярная алгебра, названная *каузальной алгеброй*, задаваемая четверкой $M_n(R) = \langle FM, \vee, \circ, * \rangle$, где FM – множество нечетких матриц, \vee – операция \max , \circ – макстриангулярная композиция, $*$ – унарный оператор замыкания.

В нечетких когнитивных картах [39] отдельные нечеткие влияния входных концептов, непосредственно воздействующие на выходной концепт, объединяются на основе S-норм [40–33]. Наиболее распространенной S-нормой, соответствующей операции объединения нечетких множеств, является операция максимума.

А значение выходного концепта K_j , смежного с N входными концептами K_i , определяется следующим образом:

$$K_j = \sum_{i=1}^N (K_i \text{ T } w_{ij}). \quad (17.9)$$

где Т – операция Т-нормы, а S – операция S-нормы. В [39] в качестве Т-нормы используется операция умножения, а S-нормы – операция максимума.

Обзор наиболее употребительных Т- и S-норм, используемых в нечеткой логике, приведен, например, в [39, 40].

Опосредованное влияние концептов

Задача нахождения опосредованного влияния концепта K_i на концепт K_y заключается в определении совокупного причинного эффекта, или «веса», того пути между этими концептами, который характеризуется максимальным совокупным значением влияния w_{iy} .

Определим l -й путь между рассматриваемыми концептами K_i и K_y нечеткой когнитивной карты следующим образом: $K_i \xrightarrow{l} K_y$, $P_l = = (i, z_1^l, z_2^l, \dots, z_n^l, y)$, $l = 1, \dots, m$, где m – возможное число путей между концептами K_i и K_y . Тогда влияние w_{iy} концепта K_i на концепт K_y , определяется в соответствии с выражением

$$w_{iy} = \sum_{l=1}^m \prod_{p \in P_l} w_{p, p+1}, \quad (17.10)$$

где в качестве Т-нормы берется операция минимума или произведения, а в качестве S-нормы – операция максимума.

Для определения опосредованного влияния всех концептов карты друг на друга в [39] используется операция транзитивного замыкания квадратной нечеткой матрицы весов:

$$\hat{W} = W \vee W^2 \vee W^3 \vee \dots, \quad (17.11)$$

где степени нечетких матриц вычисляются на основе операции макстри-ангюлярной композиции:

$$W^k = W^{k-1} \circ W. \quad (17.12)$$

Определение системных характеристик

Для определения взаимовлияния концептов от исходной нечеткой когнитивной карты с положительно-отрицательными нечеткими связями переходят к нечеткой матрице положительных связей R размера $2n \times 2n$, элементы которой определяются из матрицы $W = \|w_{ij}\|_{n \times n}$ путем замены:

$$\begin{aligned} \text{ЕСЛИ } w_{ij} > 0, \text{ ТО } r_{2i-1, 2j-1} = w_{ij}, r_{2i, 2j} = w_{ij}, \\ \text{ЕСЛИ } w_{ij} < 0, \text{ ТО } r_{2i-1, 2j-1} = -w_{ij}, r_{2i, 2j} = -w_{ij}. \end{aligned} \quad (17.13)$$

Остальные элементы принимают нулевые значения.

В случае амбивалентности исходной когнитивной карты положительно-отрицательная пара влияния преобразуется по аналогичному алгоритму, только вместо нулей на диагоналях ставятся определенные значения. Если необходимо, то при учете мнений нескольких экспертов результирующие связи в исходной карте получаются усреднением отдельно положительных и отрицательных связей по оценкам всех экспертов с учетом значимости каждого эксперта. Кроме того, для снижения диссонанса при формировании отношений взаимовлияния концептов в исходной карте могут быть использованы проверенные психометрические меры [41], на которых основаны свертки в моделях предпочтений.

Согласованные отношения взаимовлияния концептов определяются в результате транзитивного замыкания R :

$$\hat{R} = R \vee R^2 \vee R^3 \vee \dots \quad (17.14)$$

Представим полученный результат в виде матрицы, состоящей из положительно-отрицательных пар элементов (v_{ij}, \bar{v}_{ij}) , образованных по правилу

$$\begin{aligned} v_{ij} &= \max (r_{2i-1, 2j-1}, r_{2i, 2j}), \\ \bar{v}_{ij} &= -\max (r_{2i-1, 2j}, r_{2i, 2j-1}). \end{aligned} \quad (17.15)$$

Элементы матрицы $V = \|(v_{ij}, \bar{v}_{ij})\|$ могут использоваться в качестве показателей, характеризующих динамику достижения одной или нескольких целей.

Полученная таким образом матрица взаимовлияния позволяет также решать задачи проблемно-целевого анализа сложных систем и процессов [21].

Так, на основе этой матрицы могут быть рассчитаны описанные ниже основные системные показатели нечеткой когнитивной карты, по которым определяется *взаимный консонанс*, *диссонанс*, *положительное и отрицательное влияние* концептов друг на друга и на систему в целом и другие системные и интегральные показатели, используемые для исследования свойств анализируемой системы [39]:

- консонанс влияния концепта K_i на концепт K_j

$$c_{ij} = \frac{|v_{ij} + \bar{v}_{ij}|}{|v_{ij}| + |\bar{v}_{ij}|}; \quad (17.16)$$

- диссонанс влияния концепта K_i на концепт K_j

$$d_{ij} = 1 - c_{ij}; \quad (17.17)$$

- взаимный консонанс влияния концептов K_i и K_j

$$\tilde{c}_{ij} = \frac{|(v_{ij} + v_{ji}) + (\bar{v}_{ij} + \bar{v}_{ji})|}{|v_{ij} + v_{ji}| + |\bar{v}_{ij} + \bar{v}_{ji}|}; \quad (17.18)$$

- взаимный диссонанс влияния концептов K_i и K_j

$$\tilde{d}_{ij} = 1 - \tilde{c}_{ij}; \quad (17.19)$$

- воздействие (влияние) концепта K_i на концепт K_j

$$p_{ij} = \text{sign}(v_{ij} + \bar{v}_{ij}) \max(|v_{ij}|, |\bar{v}_{ij}|) \text{ для } v_{ij} \neq -\bar{v}_{ij}; \quad (17.20)$$

- взаимное положительное влияние концептов K_i и K_j

$$\tilde{p}_{ij} = \tilde{p}_{ji} = (v_{ij} S v_{ji}), \quad (17.21)$$

где S – соответствующая S -норма;

- взаимное отрицательное влияние концептов K_i и K_j

$$\tilde{n}_{ij} = \tilde{n}_{ji} = -|\bar{v}_{ij}| S |\bar{v}_{ji}|; \quad (17.22)$$

- консонанс влияния i -го концепта на систему (карту)

$$\tilde{C}_i = \frac{1}{n} \sum_{j=1}^n c_{ij}; \quad (17.23)$$

- диссонанс влияния i -го концепта на систему

$$\vec{D}_i = \frac{1}{n} \sum_{j=1}^n d_{ij}; \quad (17.24)$$

- консонанс влияния системы на j -й концепт

$$\bar{C}_j = \frac{1}{n} \sum_{i=1}^n c_{ij}; \quad (17.25)$$

- диссонанс влияния системы на j -й концепт

$$\vec{D}_j = \frac{1}{n} \sum_{i=1}^n d_{ij}; \quad (17.26)$$

- воздействие (влияние) i -го концепта на систему

$$\vec{P}_i = \frac{1}{n} \sum_{j=1}^n p_{ij}; \quad (17.27)$$

- воздействие (влияние) системы на j -й концепт

$$\bar{P}_j = \frac{1}{n} \sum_{i=1}^n d_{ij}; \quad (17.28)$$

- взаимный консонанс i -го концепта и системы

$$I_i^{SC} = (\vec{C}_i \text{ S } \bar{C}_i); \quad (17.29)$$

- взаимный диссонанс i -го концепта и системы

$$J_i^{SC} = (\vec{D}_i \text{ S } \bar{D}_i). \quad (17.30)$$

Показатели $\vec{C}_i, \vec{D}_i, \bar{C}_j, \bar{D}_j, \vec{P}_i, \bar{P}_j$ характеризуют каждый концепт относительно системы.

Выбирая соответствующий тип отношений и задавая α -уровень их значений, можно получить бинарную матрицу и, следовательно, выделить классы взаимосвязанных концептов, характеризуемых этим уровнем относительно соответствующего, выбранного для анализа свойства (взаимного консонанса, диссонанса, положительного и отрицательного влияния).

Пример. Построим и проанализируем нечеткую когнитивную карту, характеризующую проблему управления обучением по охране труда уровня предприятия, которая позволяет:

- уяснить структуру этой проблемы;
- определить факторы, влияющие на проблему (концепты);
- определить связи и уяснить их характер между выделенными фактами (отношения причинности или взаимовлияния);
- реализовать возможности оценки влияния отдельных концептов на устойчивость системы и определения результатов различных «стратегий» принимаемых решений и оценки их последствий с точки зрения стабильности системы.

С целью построения проблемно-целевой модели управления обучением по охране труда уровня предприятия был проведен экспертный опрос, в ходе которого был уточнен перечень концептов, а также определены значения отношений влияния концептов когнитивной карты друг на друга.

Рассмотрим этапы построения и анализа нечеткой когнитивной карты системы управления обучением по охране труда уровня предприятия [21].

Этап 1. Задание согласованного списка концептов нечеткой когнитивной карты системы управления обучением по охране труда уровня предприятия. В результате опроса и согласования мнений группы экспертов был сформирован следующий перечень концептов нечеткой когнитивной карты:

- (K_1) Проведение обучения по охране труда.
- (K_2) Уровень знаний руководителем предприятия законодательства по труду и по охране труда.
- (K_3) Уровень знаний отдельными категориями застрахованных требований по охране труда.
- (K_4) Обеспечение прав работника на безопасный труд.
- (K_5) Решение руководителя предприятия о проведении мероприятий по охране труда.
- (K_6) Финансовый баланс предприятия.
- (K_7) Штрафные санкции (в том числе приостановка работы предприятия).
- (K_8) Затраты на охрану труда (мероприятия по охране труда).
- (K_9) Прямые и косвенные потери от несчастных случаев и профзаболеваний.

Этап 2. Определение согласованных отношений причинности между каждой парой концептов карты. Эти отношения также были получены и уточнены в результате экспертного опроса и согласования экспертной информации.

Этап 3. Определение согласованных значений отношений причинности между каждой парой концептов карты. В результате обработки данных экспертного опроса на основе статистических методов анализа экспертной информации и последующего их согласования получены уточненные значения отношения причинности (влияния) между каждой парой концептов карты:

$$\begin{aligned} w_{12} &= 0.8; w_{13} = 0.6; w_{25} = 0.4; w_{35} = 0.6; w_{47} = -0.4; \\ w_{49} &= -0.6; w_{58} = 0.8; w_{65} = 0.8; w_{75} = 0.8; w_{76} = -0.4; \\ w_{84} &= 1; w_{86} = -0.1; w_{96} = -0.4. \end{aligned}$$

Этап 4. Построение нечеткой когнитивной карты. На рис. 17.1 представлена разработанная нечеткая когнитивная карта, предназначенная для анализа проблемы управления обучением по охране труда (ОТ) уровня предприятия.

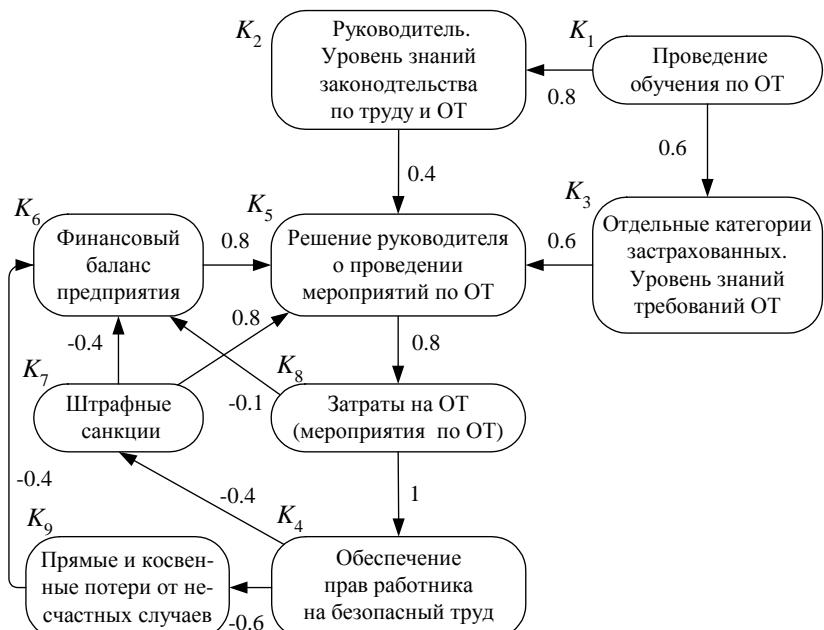


Рис. 17.1. Нечеткая когнитивная карта управления обучением по охране труда уровня предприятия

Этап 5. Формирование когнитивной матрицы взаимовлияний и согласование отношений взаимовлияния концептов. На рис. 17.2 показана сформированная исходная нечеткая когнитивная матрица $W = \{w_{ij}\}_{n \times n}$, из которой формируется нечеткая матрица положительных связей R ($2n \times 2n$).

Расчет параметров когнитивной карты - Когнитивная матрица									
Файл	Вычисление	Вид	Отчеты						
Концепт 1	Концепт 2	Концепт 3	Концепт 4	Концепт 5	Концепт 6	Концепт 7	Концепт 8	Концепт 9	
Концепт 1	8	6							
Концепт 2		4							
Концепт 3		6							
Концепт 4			-4						
Концепт 5				8					
Концепт 6					8				
Концепт 7						4			
Концепт 8							3		
Концепт 9								-8	

Рис. 17.2. Исходная когнитивная матрица

В результате расчетов формируется транзитивно замкнутая когнитивная матрица взаимовлияний (рис. 17.3).

Расчет параметров когнитивной карты - Транзитивно Замкнутая когнитивная матрица									
Файл	Вычисление	Вид	Отчеты						
Концепт 1	Концепт 2	Концепт 3	Концепт 4	Концепт 5	Концепт 6	Концепт 7	Концепт 8	Концепт 9	
Концепт 1	0.000; 0.000	0.000; 0.000	0.600; 0.000	0.288; -0.056	0.360; -0.069	0.069; -0.032	0.022; -0.115	0.288; -0.056	0.033; -0.173
Концепт 2	0.000; 0.000	0.000; 0.000	0.320; -0.061	0.400; -0.077	0.077; -0.032	0.025; -0.128	0.320; -0.061	0.037; -0.182	
Концепт 3	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.480; -0.056	0.600; -0.115	0.115; -0.048	0.037; -0.132	0.480; -0.056	0.095; -0.288
Концепт 4	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.154; -0.182	0.192; -0.240	0.240; -0.046	0.077; -0.400	0.154; -0.182	0.115; -0.600
Концепт 5	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.000; 0.154	0.154; -0.182	0.182; -0.080	0.061; -0.322	0.000; 0.154	0.052; -0.480
Концепт 6	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.640; -0.123	0.800; -0.154	0.154; -0.064	0.049; -0.256	0.640; -0.123	0.074; -0.384
Концепт 7	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.480; -0.256	0.600; -0.320	0.115; -0.400	0.102; -0.132	0.480; -0.256	0.154; -0.288
Концепт 8	0.000; 0.000	0.000; 0.000	0.000; 0.000	1.000; -0.182	0.192; -0.240	0.240; -0.100	0.077; -0.400	0.154; -0.182	0.115; -0.600
Концепт 9	0.000; 0.000	0.000; 0.000	0.000; 0.000	0.549; -0.256	0.061; -0.322	0.036; -0.400	0.102; -0.030	0.049; -0.256	0.154; -0.289

Рис. 17.3. Транзитивно замкнутая когнитивная матрица взаимовлияний

Этап 6. Расчет системных показателей карты. В соответствии с выражениями (17.15)–(17.29) можно определить системные, а также интегральные показатели влияния концептов на систему (рис. 17.4) и системы на концепты (рис. 17.5).

Этап 7. Анализ нечеткой когнитивной карты. Основными выводами, полученными на основе проведенного анализа, являются следующие.

Концепт K_1 (Проведение обучения по охране труда) в наибольшей степени по сравнению с другими концептами усиливает систему, а система, в свою очередь, совершенно не влияет на этот концепт.

Следующими по степени положительного усиления системы являются следующие концепты: K_6 (Финансовый баланс предприятия), K_3 (Уровень знаний требований по охране труда отдельными категориями застрахованных).

	Консонанс	Диссонанс	Влияние
Концепт 1	0.645	0.355	0.235
Концепт 2	0.422	0.578	0.089
Концепт 3	0.422	0.578	0.133
Концепт 4	0.263	0.737	-0.154
Концепт 5	0.359	0.641	0.089
Концепт 6	0.422	0.578	0.177
Концепт 7	0.231	0.769	0.076
Концепт 8	0.296	0.704	-0.021
Концепт 9	0.474	0.526	-0.108

Рис. 17.4. Интегральные показатели влияния концептов на систему

	Консонанс	Диссонанс	Влияние
Концепт 1	0.000	1.000	0.000
Концепт 2	0.111	0.889	0.089
Концепт 3	0.111	0.889	0.067
Концепт 4	0.573	0.427	0.396
Концепт 5	0.447	0.553	0.196
Концепт 6	0.509	0.491	0.032
Концепт 7	0.636	0.364	-0.211
Концепт 8	0.510	0.490	0.263
Концепт 9	0.636	0.364	-0.317

Рис. 17.5. Интегральные показатели влияния системы на концепты

Сама же система в наибольшей степени усиливает концепт K_4 (Обеспечение прав работника на безопасный труд) и в наибольшей степени, по сравнению с другими концептами, ослабляет концепт K_9 (Прямые и косвенные потери от несчастных случаев и профзаболеваний).

Из двух концептов, характеризующих уровень знаний требований по охране труда руководителем предприятия (K_2) и отдельных категорий застрахованных (K_3), концепт K_3 оказывает большее воздействие на систему, чем K_2 . Что позволяет сделать вывод о необходимости обращать большее внимание именно на организацию обучения отдельных категорий застрахованных. В свою очередь, система более существенно воздействует на концепт K_2 , чем на концепт K_3 . Что можно интерпретировать как большую независимость концепта K_3 от системы по сравнению с концептом K_2 .

Анализ нечеткой когнитивной карты можно продолжить, используя матрицы взаимного консонанса, диссонанса, положительного и отрицательного влияния с учетом задания соответствующего множества α -уровня. Что позволяет выделить классы связанных концептов, характеризуемых задаваемым уровнем относительно соответствующего свойства (взаимного консонанса, диссонанса, положительного и отрицательного влияния).

Кроме того, информация в положительно-отрицательной транзитивно замкнутой когнитивной матрице позволяет осуществить анализ стратегий, которые привели к той или иной зависимости между концептами и определить максимально положительный/отрицательный путь на нечетком ориентированном графе карты или максимальный консонанс/диссонанс между двумя концептами.

Когнитивные карты Силова, в отличие от карт Коско, могут быть названы действительно нечеткими. Однако их анализ позволяет сделать следующие выводы.

Во-первых, использование связки *max* при агрегировании отдельных влияний на выходном концепте в выражении (17.11) и при транзитивном замыкании (17.12) (т. е. выбор в качестве результата максимального из отдельных влияний) полностью исключает учет меньших по значению влияний, что снижает чувствительность результата ко входным воздействиям и плохо согласуется с физическим смыслом используемых отношений между концептами.

Во-вторых, в рассмотренной модели все же для описания влияния не в полной мере реализованы постулаты нечеткого подхода, рассматривавшего нечеткие множества, по двум «координатам»: базовое множество значений и принадлежность к базовому множеству. Как и в модели Коско, в нечетких когнитивных картах Силова одна «координата» не учитывается.

17.2.6. Нечеткие производственные когнитивные карты

Стремление максимально использовать постулаты теории нечетких множеств привело к появлению нечетких когнитивных карт, для описания влияний между концептами в которых используются нечеткие производственные правила (Rules Based Fuzzy Cognitive Maps – нечеткие когнитивные карты, основанные на правилах) [42–44].

Описание состояний или значений концептов

В этих картах концепты представлены в виде нечетких множеств, определяемых функциями принадлежности к базовому множеству.

Задание взаимовлияния концептов

Причинно-следственные отношения между двумя концептами выражены в виде нечеткого производственного правила со структурой «один вход – один выход» (Single Input – Single Output: SISO) относительно нечетких приращений концептов. Правила имеют такой вид:

ЕСЛИ «приращение концепта K_i малое»,
ТО «приращение концепта K_j среднее». (17.31)

Передача влияния между концептами осуществляется на основе способа нечеткого логического вывода Мамдани.

Аккумулирование влияния нескольких концептов на один концепт

Для объединения отдельных влияний на выходном концепте применяется введенная в [42–44] специальная операция «нечеткого аккумулирования с переносом» (Fuzzy Carry Accumulation). Она используется при агрегировании двух влияний от входных концептов, представленных

в виде нечетких множеств (функций принадлежности) выходного концепта. Функция принадлежности, расположенная в области меньших значений базового множества, «перемещается» в сторону больших значений до совмещения максимумов. Далее совмещенные функции принадлежности дискретизируются по базовому множеству и полученные одноточечные значения принадлежностей, расположенные в одних точках базового множества, суммируются арифметически начиная с самого малого значения базового множества. Превышение над единицей добавляется к следующей точке, аналогично поразрядному сложению с переносом.

Этот вид аккумулирования, единственный из всех, рассмотренных выше, позволяет обрабатывать нечеткие числа по двум «координатам» (принадлежности и базовому множеству) и учитывать аддитивный характер влияний отдельных концептов. Однако его механизм выглядит достаточно произвольным.

17.3. Обучение нечетких когнитивных карт

Рассматриваемые только с позиций экспериментного подхода нечеткие когнитивные карты не требуют обучения и полностью «готовы к употреблению». Однако если существует экспериментальная информация о функционировании во времени моделируемой системы, то естественно ее использовать для коррекции значений влияния концептов друг на друга. Структурное подобие нечетких когнитивных карт и искусственных нейронных сетей позволяет использовать результаты теории обучения нейронных сетей для обучения нечетких когнитивных карт. Однако существенным различием двух моделей с точки зрения возможности обучения является необходимость в случае когнитивной карты иметь информацию о поведении всех концептов, в то время как в процессе обучения искусственных нейронных сетей внутренняя структура отношений между нейронами является скрытой.

В работе [45] предложено обучать когнитивную карту методом обучения без учителя, в котором правила подстройки весов базируются на методе градиентного спуска. В [46] для обучения нечетких когнитивных карт предложено использовать дифференциальный алгоритм обучения Хэбба. В [47] предложена модификация этого алгоритма, получившая название *балансового дифференциального алгоритма обучения Хэбба*. А в работе [48] для этих целей предложено использовать генетические алгоритмы.

Обучение когнитивных карт позволяет повысить их адекватность исследуемым системам за счет учета доступной экспериментальной информации о поведении систем.

17.4. Динамическое моделирование с помощью когнитивных карт

Задачи, решаемые с помощью когнитивных карт, весьма разнообразны. Анализ когнитивной карты позволяет выявить структуру проблемы (системы), найти наиболее значимые факторы, влияющие на проблему (систему), оценить влияние факторов (концептов) друг на друга. С помощью когнитивных моделей решаются задачи исследования поведения сложной системы. Если в когнитивной карте выделены целевые концепты и концепты, на которые можно воздействовать, то круг решаемых задач включает задачи оценки достижимости цели, разработки сценариев и стратегий управления, поиска управлеченческих решений.

Часть отмеченных задач можно решить, исследуя статические системные характеристики когнитивных карт, связанные с различными типами влияний в карте, а также с исследованием свойств структур графа (циклов, путей, обратных связей).

Большое число работ, посвященных нечетким системам, рассматривают статические модели, в которых нечеткие системы аппроксимируют статическую функцию. При этом значения входов, как правило, однозначно определяют значения выходов.

Реальные физические системы, кроме статических свойств, характеризуются и динамическими свойствами, описывающими поведение системы во времени. Что определяет насущную потребность создания конструктивных механизмов их динамической имитации когнитивными картами. Для этого необходимо формализовать представление времени, преобразование вход-выход с учетом времени, а также учесть инерционные свойства системы. Кроме того, характеристики динамических моделей, в том числе и нечетких, зависят от типа моделируемой системы, перечня решаемых задач и используемых средств моделирования.

Основные принципы моделирования когнитивных карт заключаются в следующем. Модельное время дискретно и представлено в безразмерной шкале значений моментов времени 0, 1, 2, ... Предполагается некоторое соответствие шкал модельного и физического времени. Обычно используются простые модели инерционности передачи влияния между концептами, аналогичные транспортным задержкам. Часто задержка распространения влияния принимается одинаковой для всей когнитивной карты и определяется интервалом времени между двумя моментами дискретного времени. Что приводит к синхронному распространению влияния по карте. Моделирование может проводиться из некоторой начальной ситуации, определяемой начальными значениями концептов без влияния на модель извне, что соответствует саморазвитию начальной си-

туации. Если значения некоторых входных концептов могут быть подвержены внешним изменениям (или некоторые концепты соотнесены с этими воздействиями), то моделируемая система является управляемой. Моделирование может закончиться либо по достижении заданного момента дискретного времени, либо по достижении некоторой устойчивой ситуации.

Специфика когнитивного моделирования предполагает одинаковое описание поведения во времени для всех концептов, на которые оказывается влияние другими концептами. При формализации этого описания целесообразно разделять абсолютные уровни значений концептов, определяющие их *состояния* (уровни), и *приращения* (изменения этих уровней). В течение одного такта модельного времени состояния остаются неизменными.

Основные типы динамических моделей, предложенные Робертсоном в [34], для когнитивных карт могут быть интерпретированы таким образом:

$$\Delta K_j(t+1) = \sum_{i=1}^N w_{ij} \Delta K_i(t), \quad (17.32)$$

$$K_j(t+1) = \sum_{i=1}^N w_{ij} K_i(t), \quad (17.33)$$

$$\Delta K_j(t+1) = \sum_{i=1}^N w_{ij} K_i(t), \quad (17.34)$$

где $t, t+1$ – дискретные моменты времени; $j = 1, 2, \dots, P$ – номер выходного концепта; P – общее число концептов карты; $K_j, \Delta K_j$ – состояние (значение) и приращение значения j -го выходного концепта j ; $K_i, \Delta K_i$ – состояние (значение) и приращение значения i -го входного концепта; w_{ij} – вес влияния концепта i на концепт j ; N – число входных концептов для j -го выходного концепта.

Эти модели динамики (17.32)–(17.34) наиболее часто используются для когнитивного моделирования сложных социально-экономических систем [35, 36].

Примечание. По аналогии с выражениями (17.32)–(17.34) можно предложить еще один вариант описания динамики в когнитивных картах:

$$K_j(t+1) = \sum_{i=1}^N w_{ij} \Delta K_i(t). \quad (17.35)$$

Выражения (17.31)–(17.34) связывают изменения значений уровней и приращений концептов в моменты времени t и $t+1$. При моделировании они применяются сразу ко всем концептам когнитивной карты. Инерционность передачи влияния моделируется транспортной задержкой на один отсчет системного времени. На основе этих выражений решаются прямая и обратная задачи управления, задачи анализа устойчивости, прогноза развития ситуаций, анализа системных характеристик.

Примечание. Следует отметить, что в теории динамических дискретных систем (в том числе и нечетких) используются и более сложные модели системной динамики, например нелинейная авторегрессионная модель с внешними входами (*Nonlinear AutoRegressive with eXogenous input: NARX*) [49], учитывающие некоторые аспекты нелинейного поведения систем и их инерционных свойств. Возможность использования подобных моделей в когнитивных картах требует дополнительных исследований. Анализ практических задач когнитивного моделирования позволяет сделать вывод о том, что подход, основанный на моделях, аналогичных представленным в выражениях (17.31)–(17.34), позволяет решать основные задачи исследования системной динамики. Поэтому в работе рассмотрены модели именно этих видов.

К недостаткам рассмотренных моделей можно отнести следующее. Не учтены совместные воздействия состояний (абсолютного уровня значений) концептов и их приращений. Кроме того, эти модели являются линейными и не учитывают в полной мере динамические свойства реальных систем, которые могут быть существенно нелинейными и нестационарными.

Ниже рассмотрена предложенная в [50] модель динамики нечетких когнитивных карт, позволяющая учитывать нелинейные аспекты поведения моделируемых систем.

17.5. Требования к обобщенному представлению и анализу нечетких когнитивных карт

Исходя из рассмотренных в п. 17.1 задач, сформулируем требования к созданию и моделированию нечетких когнитивных карт, обобщающих свойства рассмотренных в пп. 17.2–17.4 типов когнитивных карт и реализующих расширенные возможности по анализу и моделированию сложных систем.

1. Необходимость реализации полностью нечеткого подхода в построении и анализе когнитивных карт, предполагающего нечеткость всех компонентов и механизмов когнитивной карты: концептов, способа передачи влияния, аккумулирования влияния нескольких кон-

цептов на один концепт, обучения, моделирования динамики. Конкретизируем это требование относительно задач построения и анализа когнитивных карт.

2. При решении задачи нечеткого представления концептов необходимо учитывать возможность их количественной интерпретации, или хотя бы сравнения. При этом концепты могут быть представлены в виде как нечетких множеств, так и четких значений либо одноточечных принадлежностей.
3. Механизм нечеткого влияния между концептами должен иметь характер нечеткого отображения нечеткого множества значений входного концепта на нечеткое множество значений выходного концепта. При этом должна быть учтена возможность использования в качестве механизма передачи влияния и четких отображений (функций).
4. Должна быть решена проблема учета отрицательных весов влияния между концептами и аккумулирования влияний разных знаков. При этом желательно предложить механизм совместного учета как положительных, так и отрицательных влияний концептов друг на друга не только в виде четких значений или одноточечных принадлежностей, но и в виде функций принадлежности.
5. Модель динамики должна учитывать существенную нелинейность поведения системы за счет совместного учета состояний и приращений при влиянии входных концептов на выходные.
6. Процедура аккумулирования отдельных влияний входных концептов на выходном должна иметь аддитивный, накопительный характер, с возможностью учета вклада самых незначительных по значению влияний.
7. Данная процедура аккумулирования не должна зависеть от порядка учета отдельных факторов (концептов), следовательно, должна обладать свойствами коммутативности и ассоциативности.
8. При аккумулировании влияний должен учитываться сдвиг функций принадлежности по координате базового множества.

17.6. Обобщенные нечеткие продукционные когнитивные карты

В работе [51] предложены обобщенные нечеткие продукционные когнитивные карты (Generalized Rule-Based Fuzzy Cognitive Maps), построенные в соответствии с вышеуказанными требованиями.

Обобщенной нечеткой производственной когнитивной картой (ОНПКК) называют нечеткую причинно-следственную сеть (сеть взаимовлияния) вида

$$G = (\mathbf{K}, \mathbf{W}), \quad (17.35)$$

где $\mathbf{K} = \{K_1, K_2, \dots, K_p\}$ – множество концептов; $\mathbf{W} = \{w_{ij}\}$ – множество связей между концептами, описание состояний и правила задания взаимовлияния между концептами которой приведены ниже в пп. 17.6.1 и 17.6.2.

17.6.1. Описание состояний или значений концептов обобщенной нечеткой производственной когнитивной карты

Каждый концепт обобщенной нечеткой производственной когнитивной карты K_i ($i \in I = \{1, 2, \dots, p\}$) описывается соответствующей лингвистической переменной $\langle \tilde{K}_i, T_i, D_i \rangle$, где $T_i = \{T_1^i, T_2^i, \dots, T_{m_i}^i\}$ – терм-множество лингвистической переменной (набор лингвистических значений концепта, характеризующих его типовые состояния); m_i – число типовых состояний данного концепта; D_i – базовое множество \tilde{K}_i . Для описания термов T_z^i ($z \in Z = \{1, 2, \dots, m_i\}$), соответствующих типовым состояниям (значениям) концепта K_i , используются нечеткие переменные $\langle T_z^i, D_i, \tilde{C}_z^i \rangle$, т. е. значение T_z^i описывается нечетким множеством \tilde{C}_z^i в базовом множестве D_i :

$$\tilde{C}_z^i = \{\langle \mu_{C_z^i}(d)/d \rangle\}, d \in D_i. \quad (17.36)$$

17.6.2. Задание взаимовлияния между концептами обобщенной нечеткой производственной когнитивной карты

Веса влияния w_{ij} ($i, j \in I = \{1, 2, \dots, p\}$) между типовыми состояниями каждой пары концептов задаются одним из значений терм-множества лингвистической переменной $\langle \tilde{W}_{ij}, T_{w_{ij}}, D_{w_{ij}} \rangle$, где $T_{w_{ij}} = \{T_{11}^{w_{ij}}, \dots, T_{zl}^{w_{ij}}\}$ – терм-множество лингвистической переменной \tilde{W}_{ij} ; $z \times l$ – число значений $T_{w_{ij}}$; $D_{w_{ij}}$ – базовое множество \tilde{W}_{ij} .

Веса влияния между типовыми состояниями каждой пары концептов обобщенной нечеткой производственной когнитивной карты задаются нечеткими переменными $\langle T_{zl}^{w_{ij}}, D_{w_{ij}}, \tilde{H}_{zl}^{w_{ij}} \rangle$, которые описываются нечетки-

ми множествами $\tilde{H}_{zl}^{w_{ij}}$ в базовом множестве $D_{w_{ij}}$ ($K_i, K_j \in \mathbf{K}, z \in Z = \{1, 2, \dots, m_i\}, l \in L = \{1, 2, \dots, m_j\}$) следующего вида:

$$\tilde{H}_{zl}^{w_{ij}} = \{\langle \mu_{\tilde{H}_{zl}^{w_{ij}}} (d)/d \rangle\}, d \in D_{w_{ij}}. \quad (17.37)$$

На рис. 17.6 показан пример структуры обобщенной нечеткой производционной когнитивной карты.

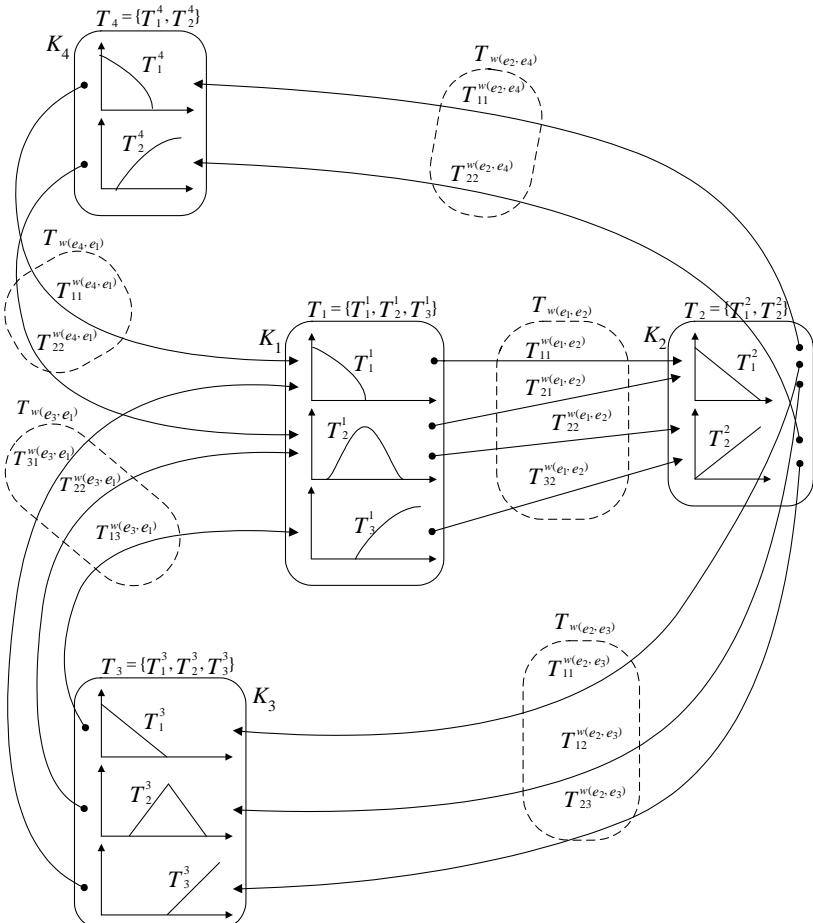


Рис. 17.6. Пример структуры обобщенной нечеткой производционной когнитивной карты

Примечание. Проблема совместного учета положительного и отрицательного влияния концептов и аккумулирования влияний в обобщенной нечеткой производственной когнитивной карте решается путем расширения базовых множеств состояний концептов, их приращений и весов влияния на область отрицательных значений и использованием для задания нечетких отображений нечетких производственных систем типа Мамдани и нечетких алгебраических операций на всем диапазоне базовых множеств.

17.6.3. Аккумулирование влияния концептов и модель динамики для обобщенной нечеткой производственной когнитивной карты

Учет совместного влияния уровней и приращений концептов обобщенной нечеткой производственной когнитивной карты позволяет предложить, по аналогии с моделями Робертса, следующую модель динамики:

$$\begin{cases} K_j(t+1) = K_j(t) + \sum_{i=1}^N \Delta K_{ij}(t+1), \\ \Delta K_{ij}(t+1) = f_{ij}[K_i(t), K_j(t), \Delta K_i(t)], \end{cases} \quad (17.38)$$

где $f_{ij} [\bullet]$ – нелинейный оператор, представляющий передачу влияния от входного концепта i к выходному концепту j .

Зависимость ΔK_{ij} не только от ΔK_i , но и от состояний K_i , K_j позволяет учесть особенности нелинейного поведения системы.

Построим модель динамики для обобщенной нечеткой производственной когнитивной карты. Приняв гипотезу о независимости и аддитивности влияний отдельных входных концептов на выходной, можно свести нечеткую структуру с N входами к N нечетким структурам с одним входом.

Для корректного аккумулирования нескольких нечетких значений выходов одновходовых структур $\tilde{\Delta K}_{ij}(t+1)$ используется операция \oplus нечеткого алгебраического сложения [40, 52], позволяющая вести обработку по двум «координатам» – принадлежностей и базового множества – и допускающая *сдвиг по базовому множеству*. В этом случае на основе модели (17.38) может быть построена следующая модель динамики обобщенной нечеткой производственной когнитивной карты:

$$\begin{cases} \tilde{K}_j(t+1) = \tilde{K}_j(t) \oplus [\bigoplus_{i=1, 2, \dots, N} \tilde{\Delta K}_{ij}(t+1)], \\ \tilde{\Delta K}_{ij}(t+1) = \tilde{f}_{ij}[\tilde{K}_i(t), \tilde{K}_j(t), \Delta \tilde{K}_i(t)], \end{cases} \quad (17.39)$$

или

$$\tilde{K}_j(t+1) = \tilde{K}_j(t) \oplus \{ \bigoplus_{i=1, 2, \dots, N} \tilde{f}_{ij}[\tilde{K}_i(t), \tilde{K}_j(t), \Delta \tilde{K}_i(t)] \}, \quad (17.40)$$

где \tilde{K}_i , $\Delta\tilde{K}_{ij}$ – нечеткие множества, представляющие значения уровня i -го концепта и приращения этого концепта, смежного с выходным концептом j ; \tilde{K}_j , $\Delta\tilde{K}_j$ – нечеткие множества, представляющие значения уровня и приращения выходного концепта j ; \oplus – операция нечеткого алгебраического сложения; \tilde{f}_{ij} – нечеткий оператор, задающий нечеткое отображение типа «много входов – один выход» (Multi Input – Single Output: MISO) с тремя входами и одним выходом.

Трехходовая нечеткая структура, представляющая оператор \tilde{f}_{ij} , может быть сведена к каскадному соединению двух двухходовых структур: первая из которых представляет собой нечеткий оператор \tilde{w}_{ij} , характеризующий силу влияния концептов по их абсолютным значениям (типовым состояниям), а вторая – собственно нечеткий оператор \tilde{f}_{ij} . Тогда выражение (17.40) записывается следующим образом:

$$\tilde{K}_j(t+1) = \tilde{K}_j(t) \oplus \left\{ \bigoplus_{i=1, 2, \dots, N} \tilde{f}_{ij}[\tilde{\varphi}_{ij}(t), \Delta\tilde{K}_i(t)] \right\}, \quad (17.41)$$

где

$$\tilde{\varphi}_{ij}(t) = \tilde{w}_{ij}[\tilde{K}_i(t), \tilde{K}_j(t)]. \quad (17.42)$$

На рис. 17.7 представлена каскадная схема этой модели динамики.

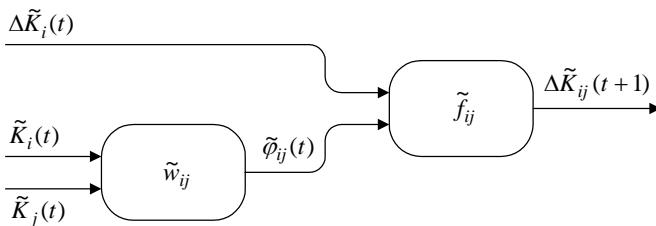


Рис. 17.7. Каскадная схема модели динамики для обобщенной нечеткой производной когнитивной карты

17.6.4. Реализация модели динамики для обобщенной нечеткой производной когнитивной карты

Перейдем к интерпретации этой динамической модели в терминах обобщенной нечеткой производной когнитивной карты и к ее конкретной реализации. Проведенный в работе [53] анализ позволяет сделать вывод о целесообразности использования для представления нечетких

операторов \tilde{w}_{ij} и \tilde{f}_{ij} нечетких производственных моделей типа Мамдани, позволяющих представить антецеденты и консеквенты нечетких правил в виде нечетких множеств.

В этом случае для каждой пары «входной концепт – выходной концепт» нечеткие множества \tilde{K}_i , \tilde{K}_j , $\Delta\tilde{K}_i$ задаются с помощью терм-множеств лингвистических значений (термов), каждому из которых соответствует функция принадлежности к базовому множеству.

На рис. 17.8 проиллюстрирован пример взаимодействия входного концепта \tilde{K}_i , описываемого лингвистической переменной $\langle \tilde{K}_i, T_i, D_i \rangle$, где $T_i = \{T_1^i, T_2^i, T_3^i\}$, и выходного концепта \tilde{K}_j , описываемого лингвистической переменной $\langle \tilde{K}_j, T_j, D_j \rangle$, где $T_j = \{T_1^j, T_2^j\}$.

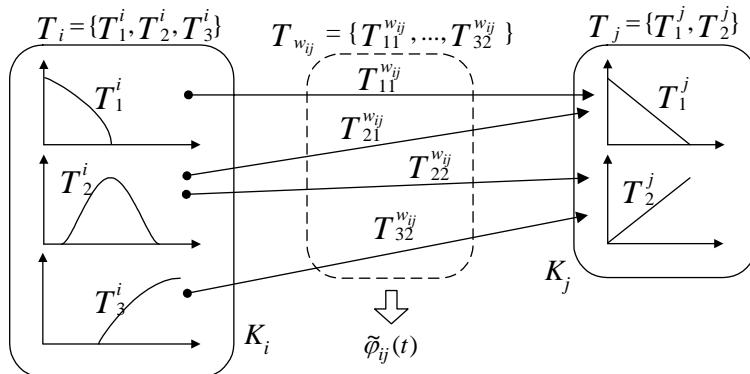


Рис. 17.8. Пример взаимодействия входного и выходного концептов обобщенной нечеткой производственной когнитивной карты

Рассмотрим первую двухвходовую нечеткую производственную модель, реализующую нечеткий оператор \tilde{w}_{ij} . Влияние между типовыми состояниями каждой пары «входной концепт – выходной концепт» карты задается значениями терм-множеств лингвистической переменной $\langle \tilde{w}_{ij}, T_{w_{ij}}, D_{w_{ij}} \rangle$, где $T_{w_{ij}} = \{T_{11}^{w_{ij}}, T_{21}^{w_{ij}}, T_{22}^{w_{ij}}, T_{32}^{w_{ij}}\}$. И таким образом, для представления нечеткого оператора \tilde{w}_{ij} , определяемого в выражении (17.42) силой влияния концептов по их абсолютным значениям (типовым состояниям), нечеткие правила будут иметь следующий вид:

ЕСЛИ $\tilde{K}_i = T_1^i$ **И** $\tilde{K}_j = T_1^j$, **ТО** $\tilde{\phi}_{ij}(t) = T_{11}^{w_{ij}}$,
ЕСЛИ $\tilde{K}_i = T_2^i$ **И** $\tilde{K}_j = T_1^j$, **ТО** $\tilde{\phi}_{ij}(t) = T_{21}^{w_{ij}}$,
ЕСЛИ $\tilde{K}_i = T_2^i$ **И** $\tilde{K}_j = T_2^j$, **ТО** $\tilde{\phi}_{ij}(t) = T_{22}^{w_{ij}}$,
ЕСЛИ $\tilde{K}_i = T_3^i$ **И** $\tilde{K}_j = T_2^j$, **ТО** $\tilde{\phi}_{ij}(t) = T_{32}^{w_{ij}}$. (17.43)

Вторая двухвходовая нечеткая производственная модель, реализующая оператор \tilde{f}_{ij} , будет иметь, например, следующий вид:

ЕСЛИ $\tilde{\phi}_{ij}(t) = T_{11}^{w_{ij}}$ **И** $\Delta\tilde{K}_i(t) = \text{«среднее отрицательное»}$,
ТО $\Delta\tilde{K}_{ij}(t+1) = T_{11}^{w_{ij}}$,
ЕСЛИ $\tilde{\phi}_{ij}(t) = T_{21}^{w_{ij}}$ **И** $\Delta\tilde{K}_i(t) = \text{«малое отрицательное»}$,
ТО $\Delta\tilde{K}_{ij}(t+1) = T_{11}^{w_{ij}}$,
ЕСЛИ $\tilde{\phi}_{ij}(t) = T_{11}^{w_{ij}}$ **И** $\Delta\tilde{K}_i(t) = \text{«среднее положительное»}$,
ТО $\Delta\tilde{K}_{ij}(t+1) = T_{21}^{w_{ij}}$,
 \dots
ЕСЛИ $\tilde{\phi}_{ij}(t) = T_{22}^{w_{ij}}$ **И** $\Delta\tilde{K}_i(t) = \text{«большое положительное»}$,
ТО $\Delta\tilde{K}_{ij}(t+1) = T_{32}^{w_{ij}}$. (17.44)

Так как число термов $T_{w_{ij}}$ достаточно мало, то это позволяет существенно сократить количество правил при построении двухкаскадной нечеткой производственной модели по сравнению с однокаскадной трехвходовой нечеткой производственной моделью для реализации оператора нечеткого отображения \tilde{f}_{ij} .

Результирующее значение приращения $\Delta\tilde{K}_{ij \text{ res}}(t+1)$ (итоговое нечеткое множество) для j -го выходного концепта (от i -го входного концепта) формируется на основе алгоритма нечеткого вывода Мамдани путем композиции (объединения) найденных по каждому правилу «усеченных» функций принадлежности (частных заключений) с использованием операции max.

Данная процедура должна быть проведена для всех входных концептов $i = 1, 2, \dots, N$ по отношению к рассматриваемому выходному концепту.

Далее все полученные итоговые нечеткие подмножества $\Delta\tilde{K}_{ij \text{ res}}(t+1)$ ($i = 1, 2, \dots, N$) аккумулируются между собой с использованием операции \oplus нечеткого алгебраического сложения:

$$\Delta\tilde{K}_{j \text{ res}}(t+1) = \bigoplus_{i=1, 2, \dots, N} \Delta\tilde{K}_{ij \text{ res}}(t+1). \quad (17.45)$$

И на последнем шаге после предварительной идентификации типового состояния (определения абсолютного значения) выходного концепта j , проводимой на каждом такте моделирования, выполняется операция нечеткого сложения над нечеткими множествами $\tilde{K}_j(t)$ и $\Delta\tilde{K}_{j \text{ res}}(t+1)$ и определяется следующее модельное значение $\tilde{K}_j(t+1)$ выходного концепта:

$$\tilde{K}_j(t+1) = \tilde{K}_j(t) \oplus \Delta\tilde{K}_{j \text{ res}}(t+1) \quad (17.46)$$

Примечание. Операция нечеткого алгебраического сложения, используемая в выражениях (17.45) и (17.46), в общем случае нехарактерна для нечетких продукционных моделей. Так, в алгоритме нечеткого вывода Мамдани для агрегирования на выходе используется операция объединения нечетких множеств на основе связки \max ; в аддитивной модели Кооско – обычное сложение принадлежностей. При этом объединение проводится только по координате принадлежностей нечетких множеств. Это можно объяснить тем, что сдвиг по координате базового множества в продукционных моделях с MISO-структурой должен быть учтен экспертом при составлении базы нечетких правил и выражается в расположении функций принадлежности консеквентов относительно базового множества. В обобщенных же нечетких продукционных когнитивных картах эксперт определяет лишь отдельные нечеткие веса причинных отношений между концептами. Поэтому для корректного аккумулирования нескольких нечетких влияний используется операция нечеткого алгебраического сложения, позволяющая вести обработку по двум координатам – принадлежностей и базового множества.

Единственным ограничением использования операции нечеткого алгебраического сложения в предложенной для ОНПКК модели динамики является расширение интервала неопределенности состояний (нечетких значений) концептов. Может быть предложено два способа решения этой проблемы.

Во-первых, полученное из выражения (17.47) значение в дальнейшем используется для идентификации состояния концепта K_j в момент времени $(t+1)$ на основе, например, определения максимальной степени нечеткого равенства с одним из типовых нечетких состояний данного концепта

$T_j = \{T_1^j, T_2^j, \dots, T_{m_j}^j\}$ [14, 46]. Данная процедура идентификации, проводимая после каждого шага моделирования, гарантирует от расширения интервала неопределенности состояний концептов.

Во-вторых, в случае необходимости можно обойтись и без идентификации типового значения концепта, воспользовавшись приведением полученного результата к четкому значению на основе одной из известных процедур дефазификации (например, центроидным методом).

При моделировании все вышеуказанные действия выполняются для всех концептов $K = \{K_1, K_2, \dots, K_p\}$ ОНПКК в моменты времени $t = 1, 2, \dots$ вне зависимости от реализуемого режима (саморазвития либо управления).

Представленная реализация предложенной модели динамики для обобщенных нечетких продукциинных когнитивных карт позволяет существенно сократить размерность модели за счет следующих решений:

- Во-первых, благодаря принятой гипотезе о независимости и аддитивности влияний отдельных входных нечетких концептов на выходной удается свести нечеткую систему с N входами к N нечетким системам с одним входом. И поэтому аккумулирование нечетких значений выходов одновходовых моделей $\tilde{K}_{ij}(t+1)$ с использованием операции нечеткого алгебраического сложения в выражениях (17.39)–(17.41) позволяет исключить экспоненциальный рост числа правил при увеличении числа входов нечеткой модели.
- Во-вторых, представление трехходовой нечеткой модели, реализующей оператор \tilde{f}_{ij} , каскадным соединением двух двухходовых моделей позволяет уменьшить количество правил при построении результирующей нечеткой продукциинной модели (17.44).

17.7. Нечеткие реляционные когнитивные карты

С целью обеспечения максимально возможной общности и гибкости построения и анализа моделей слабоформализуемых систем и проблем, обусловливаемых нечетким подходом, в работе [54] предложено обобщение и развитие методов нечеткого когнитивного моделирования за счет реляционного представления нечетких соотношений влияния между концептами.

17.7.1. Определение нечеткой реляционной когнитивной карты

Пусть значение концепта K_1 может быть представлено в виде нечеткого подмножества \tilde{A} множества $X \subset \Re$ (\Re – множество действительных чисел) с функцией принадлежности $\mu_{\tilde{A}}(x)$, $x \in X$, а значение концепта

K_2 – в виде нечеткого подмножества \tilde{B} множества $Y \subset \mathcal{R}$ с функцией принадлежности $\mu_{\tilde{B}}(x), y \in Y$.

Введем понятие *нечеткого влияния* концепта K_1 на концепт K_2 , устанавливающего некоторое нечеткое отображение подмножества \tilde{A} на подмножество \tilde{B} .

В наиболее общем виде нечеткое влияние концептов может быть представлено в виде бинарного нечеткого отношения $\tilde{R}(x, y)$, которое назовем *нечетким причинно-следственным отношением* или *нечетким отношением влияния*. Функцию принадлежности этого отношения обозначим как $\mu_{\tilde{R}}(x, y)$.

Передача влияния между двумя концептами когнитивной карты осуществляется на основе нечеткой композиции

$$\tilde{B} = \tilde{A} \circ \tilde{R}, \quad (17.47)$$

где \circ – операция нечеткой композиции.

Выражение (17.47) позволяет найти нечеткое множество \tilde{B} по известным нечеткому множеству \tilde{A} и нечеткому отношению $\tilde{R}(x, y)$. С использованием функций принадлежностей (17.47) можно представить в следующем виде:

$$\mu_{\tilde{B}}(y) = \vee_{x \in X} \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{R}}(x, y), \quad (17.48)$$

где \wedge – операция Т-нормы; \vee – операция S-нормы.

На основе Т-норм реализуется операция пересечения $\mu_{\tilde{A}}(x)$ и $\mu_{\tilde{R}}(x, y)$ при фиксированном y , и на основе S-норм агрегируются полученные множества принадлежностей для получения значения $\mu_{\tilde{B}}(y)$.

Под *нечеткой реляционной когнитивной картой* (НРКК) будем понимать когнитивную карту, взаимосвязи между концептами которой определены на основе нечетких отношений влияния, а передача влияния между концептами осуществляется на основе нечеткой композиции.

С учетом нечеткого реляционного описания связи концепт–концепт всю когнитивную карту можно описать с помощью *матрицы нечетких отношений* \tilde{M} , элементами которой являются нечеткие отношения \tilde{R}_{ij} , описывающие нечеткую силу влияния концепта K_i на концепт K_j :

$$\tilde{M} = \begin{vmatrix} 0 & \tilde{R}_{12} & \dots & \tilde{R}_{1N} \\ \tilde{R}_{12} & 0 & \dots & \tilde{R}_{2N} \\ \dots & \dots & \dots & \dots \\ \tilde{R}_{N1} & \tilde{R}_{N2} & \dots & 0 \end{vmatrix}, \quad (17.49)$$

где N – общее число концептов когнитивной карты; нули на главной диагонали предполагают, что в графе карты отсутствуют петли.

С учетом (17.49) нечеткую реляционную карту можно рассматривать как *нечеткую реляционную сеть* $\tilde{E} = (\tilde{K}, \tilde{R})$, где \tilde{K} – множество нечетких концептов, соответствующих узлам сети, а \tilde{R} – множество нечетких отношений, взвешивающих дуги сети [54].

17.7.2. Способы интерпретации и построения нечетких отношений влияния

Подход к описанию когнитивных карт на основе нечетких отношений влияния и нечеткой композиции является обобщенным и предполагает использование разных способов интерпретации и описания отношений между концептами, представления концептов и передачи влияния между ними. Ниже рассмотрена интерпретация нечетких отношений в аспекте нечетких отображений (функций). Показано, что нечеткие отношения в НРКК могут быть построены на основе нечетких продуктов, используемых в нечеткой производственной и нечеткой реляционной моделях.

Интерпретация нечетких отношений между концептами как нечетких отображений (функций)

В соответствии с принципом обобщения Заде [55] нечеткие отношения могут играть роль функций в исчислении нечетких множеств, являясь расширением четких отношений, представляющих отображения множества X в множество $Y: X \rightarrow Y$.

Четкое функциональное отображение $y = f(x)$ в нотации отношений может быть представлено в виде следующего четкого отношения между четкими переменными x и y :

$$\mu(x, y) = \begin{cases} 1, & \forall x, y : y = f(x), \\ 0, & \forall x, y : y \neq f(x). \end{cases} \quad (17.50)$$

Расширяя область значений $\mu(x, y)$ до интервала $[0, 1]$, получим нечеткое отношение. Отображение же входной нечеткой переменной на выходную переменную производится на основе нечеткой композиции [1]. При этом значения переменных могут быть как «четкими» (точнее, одно-

точечными нечеткими множествами), так и произвольными нечеткими множествами. Любое четкое отношение можно рассматривать как частный случай нечеткого.

Подобный подход допускает интерпретацию нечетких реляционных когнитивных карт как *нечеткого расширения* существующих типов четких когнитивных карт. Четкие линейные или нелинейные функции, используемые в когнитивных картах для представления взаимовлияния концептов, могут рассматриваться как множества α -уровня нечеткого отношения при $\alpha = 1$, т. е. пересечение нечеткого отношения с плоскостью $\mu = 1$. Этот подход позволяет наглядно сравнивать результаты вычислений в нечеткой и четкой моделях.

Нечеткое отношение может быть задано аналитически, в виде функции принадлежности, зависящей от двух четких переменных базовых множеств входа и выхода. При таком способе задания нечеткое отношение в общем случае представляется непрерывной поверхностью в трехмерном пространстве координат « X » базового множества входа, « Y » базового множества выхода и принадлежностей « μ ». При практической реализации удобнее использовать дискретизированное по координатам « X » и « Y » нечеткое отношение, которое в этом случае может быть представлено в виде матрицы принадлежностей [7].

В подходе, основанном на теории нечетких множеств, важно применение способов построения нечетких отношений, базирующихся на экспертных знаниях о предметной области, так как точные законы поведения сложной системы часто неизвестны. Экспертное построение матрицы нечеткого отношения путем задания функции принадлежности каждой точке декартова произведения пространства входной и выходной переменных часто бывает трудноразрешимой задачей. Поэтому в качестве альтернативы можно указать на возможность применения экспертного способа задания нечеткого отношения между концептами, в рамках функциональной интерпретации, на основе понятия *нечеткого пучка функций* [52]:

$$\tilde{f} = \{(f_i, \mu_{\tilde{f}}(f_i))\}, \\ f_i : X \rightarrow Y, i = 1, 2, \dots, N, f_i = f(x), \forall x \in X, \quad (17.51)$$

где f_i – четкие функции, характеризуемые некоторой степенью принадлежности $\mu_{\tilde{f}}(f_i)$ к нечеткому пучку функций \tilde{f} .

Нечеткий пучок функций может быть интерпретирован как частный случай нечеткого отношения. При этом также возможно аналитическое задание нечеткого отношения, которое будет представлено не непрерыв-

ной поверхностью, а семейством линий, расположенных в плоскостях $\mu_{\tilde{f}}(f_i)$. Задание конечного числа функций и их принадлежностей к пучку может выражать степень уверенности эксперта в том, что та или иная функция адекватно отражает закон взаимозависимости между концептами когнитивной карты.

При экспертном построении нечетких отношений влияния в НРКК могут быть использованы подходы к проектированию нечетких продукционных и нечетких реляционных моделей.

Построение нечетких отношений между концептами на основе нечетких продукции

Нечеткие отношения «концепт–концепт» в нечеткой реляционной когнитивной карте могут быть построены на основе лингвистической нечеткой продукционной модели типа SISO, включающей в себя нечеткие правила следующего типа:

$$\text{П.: ЕСЛИ } \tilde{X} \text{ есть } \tilde{A}_i, \text{ ТО } \tilde{Y} \text{ есть } \tilde{B}_i, \quad i = 1, 2, \dots, L, \quad (17.52)$$

где \tilde{X} – входная переменная. \tilde{A}_i , \tilde{B}_i – лингвистические термы (константы); \tilde{Y} – выходная переменная; \tilde{A}_i , \tilde{B}_i , \tilde{X} и \tilde{Y} в общем случае – нечеткие множества, определенные на базовом множестве четких переменных $x \in X$, $y \in Y$; и заданные соответствующими функциями принадлежности.

Правила (17.52) проектируются обычно экспертом. Каждое правило может рассматриваться как нечеткое отношение $\tilde{R} = \tilde{A}_i \times \tilde{B}_i$ (\times – декартово произведение), построенное с помощью Т-норм (например, операции \min):

$$\mu_{\tilde{R}_i}(x, y) = \mu_{\tilde{A}_i}(x) \wedge \mu_{\tilde{B}_i}(y). \quad (17.53)$$

В свою очередь, вся совокупность нечетких правил может быть задана нечетким отношением. Нечеткое отношение, представляющее всю базу правил, формируется объединением нечетких отношений, представляющих отдельные правила $\tilde{R} = \bigcup_{i=1}^L \tilde{R}_i$, например, на основе связки \max :

$$\mu_{\tilde{R}}(x, y) = \bigvee_{i=1, 2, \dots, L} (\mu_{\tilde{A}_i}(x) \wedge \mu_{\tilde{B}_i}(y)). \quad (17.54)$$

Нечеткое значение выходной переменной определяется с помощью нечеткой композиции входной переменной с полученным нечетким отношением:

$$\tilde{Y} = \tilde{X} \circ \tilde{R}. \quad (17.55)$$

Таким образом, формирование нечеткого отношения между концептами может быть проведено на основе базы нечетких продукциий, т. е. на основе лингвистической таблицы значений, связывающей входной и выходной концепты. Далее путем, рассмотренным выше, эта база сводится к одному нечеткому отношению. Передача влияния между концептами производится на основе нечеткой композиции.

В нечетких реляционных системах используются нечеткие отношения несколько другого рода, которые также основаны на нечетком производном представлении знаний о предметной области.

Нечеткая реляционная модель типа SISO [56, 57] может рассматриваться как расширение нечеткой производной модели типа SISO.

Пусть $A = \{ \tilde{A}_h \}, h = 1, 2, \dots, H$ и $B = \{ \tilde{B}_h \}, h = 1, 2, \dots, H$ – множества лингвистических термов переменных антецедента и консеквента в модели.

База правил (17.52) может быть представлена в виде четкого отношения R между лингвистическими термами переменных антецедента и консеквента:

$$R: A \times B \rightarrow \{0, 1\}, \quad (17.56)$$

где \times – декартово произведение терм-множеств антецедентов и консеквентов.

Нечеткая реляционная модель формируется с помощью обобщения четкого отношения (17.56) до нечеткого отношения:

$$R: A \times B \rightarrow [0, 1]. \quad (17.57)$$

В нечеткой реляционной модели каждому правилу соответствует свой весовой коэффициент, определяемый соответствующим элементом нечеткого отношения (17.57).

Необходимо отметить, что отношения (17.57) и (17.55) различны. Каждый элемент отношения (17.55) представляет степень принадлежности декартова произведения четких элементов базовых множеств переменных антецедента и консеквента. В нечеткой реляционной модели нечеткое отношение (17.57) представляет степени принадлежности декартова произведения лингвистических термов. В нечетких реляционных моделях в процедуре нечеткого логического вывода используется нечеткая композиция. При этом эти модели предполагают представление произвольного нечеткого множества на базе множества лингвистических термов.

Очевидно, что рассмотренный подход, используемый в нечетких реляционных моделях типа SISO, также может быть использован для представления нечеткого отношения концепт–концепт и передачи влияния между двумя концептами в нечеткой реляционной когнитивной карте.

17.7.3. Модель динамики нечеткой реляционной когнитивной карты

Следует отметить, что при моделировании динамики в когнитивной карте целесообразно учитывать как приращения значений концептов, так и их абсолютные уровни [50]. Кроме того, при этом необходим гибкий подход, позволяющий использовать представление значений концептов на основе различных шкал, определяемых свойствами моделируемой системы.

Рассмотрим модель динамики четкой когнитивной карты Робертса относительно приращений значений концептов. Представим ее в следующем виде:

$$K_j(t+1) = K_j(t) + \sum_{i=1}^N w_{ij} \Delta K_i(t). \quad (17.58)$$

Будем исследовать нечеткое расширение этой модели для нечеткой реляционной когнитивной карты. Представим значения концептов нечеткими множествами, вес влияния – нечетким отношением, а четкие арифметические операции – операциями над нечеткими множествами. При этом приращение значений концептов $\Delta K_i(t)$ будем представлять на основе операции нечеткого приращения, умножение на вес влияния w_{ij} – нечеткой композицией, суммирование отдельных влияний – операцией нечеткого агрегирования.

Нечеткое расширение выражения (17.58) примет следующий вид:

$$\tilde{K}_j(t+1) = \tilde{K}_j(t) \oplus \left(\bigoplus_{i=1}^N (\tilde{K}_i(t) \simeq \tilde{K}_i(t-1)) \circ \tilde{R}_{ij} \right), \quad (17.59)$$

где $\tilde{K}_j(t+1)$, $\tilde{K}_j(t)$, $\tilde{K}_i(t)$, $\tilde{K}_i(t-1)$ – нечеткие значения концептов в соответствующие моменты времени; N – число входных концептов; \tilde{R}_{ij} – нечеткое отношение между концептами; операция \oplus – операция нечеткого агрегирования совокупных влияний и предыдущего значения выходного концепта; $\bigoplus_{i=1}^N$ – операция агрегирования отдельных влияний; \simeq – операция приращения нечетких значений концептов; \circ – операция нечеткой композиции.

Выражение (17.59) представляет собой модель динамики нечеткой дискретной системы с обратными связями. Обратные связи являются «двууровневыми». Во-первых, значение концепта в момент времени ($t+1$) зависит от значения концепта в момент времени t , что приводит к итерационным вычислениям при моделировании. Во-вторых, могут

иметь место обратные связи, связанные с распространением влияния по нечеткой сети.

Нечеткие отношения в (17.59) могут быть нечетким расширением как линейных, так и нелинейных функций, а значит, (17.59) позволяет моделировать широкий спектр самых разнообразных реальных систем, а также особенностей их поведения во времени.

Следует отметить также, что логика моделирования на основе когнитивных карт накладывает некоторые алгебраические ограничения, прежде всего на операцию агрегирования влияний. Результат операции не должен зависеть от порядка «вхождения» отдельных нечетких операндов. Кроме того, одним из методов анализа когнитивной карты, например для поиска решений, для ответа на вопрос: «*что ..., если ...?*», являются ее структурные преобразования, например удаление узлов и дуг в графе когнитивной карты или добавление новых. При этом соответствующие изменения в операции агрегирования не должны влиять на результат «постоянной части» операции. Эти соображения приводят к необходимости выполнения по крайней мере свойств *коммутативности* и *ассоциативности* для операции агрегирования.

Рассматривая проблему агрегирования влияний, можно отметить, что наиболее естественна гипотеза об *аддитивном* характере накопления отдельных влияний. Эта гипотеза наиболее распространена в практике когнитивного моделирования. Она характеризуется количественно выраженным свойствами концептов и независимостью отдельных влияний при агрегировании. Например, пусть концепт K_1 на рис. 17.9 характеризует «Число заводов», K_2 – «Численность населения», а K_3 – «Потребление энергии». Отношения и \tilde{R}_{23} \tilde{R}_{13} представляют положительное влияние. Даже если входные концепты «Число заводов» и «Численность населения» связаны между собой посредством нечетких отношений, очевидно, что агрегирование их влияний на выходной концепт «Потребление энергии» носит аддитивный характер.

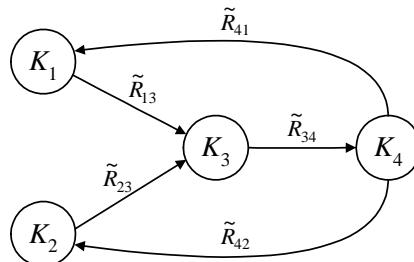


Рис. 17.9. Пример структуры когнитивной карты

Если принять гипотезу аддитивности при агрегировании, то (17.59) можно реализовать на основе операций обычной нечеткой арифметики [1, 40, 52].

Нечеткое приращение значений концептов в моменты $(t-1)$ и t может быть осуществлено на основе операции нечеткого вычитания, а агрегирование отдельных влияний и агрегирование полученного результата со значением выходного концепта в момент времени t – на основе операции нечеткого сложения. Как показано, например, в [40], нечеткое сложение и вычитание обладают свойствами коммутативности и ассоциативности.

Реализация предложенной модели динамики на основе обычной нечеткой арифметики требует решения проблемы расширения неопределенности результатов при нечетких вычислениях. Эта проблема становится особенно существенной при итерационных вычислениях, характерных для моделирования систем с обратными связями. Как показано, например, в [58], функция принадлежности в этом случае может стремиться к равномерному распределению, если в качестве S-нормы используется связка «максимум».

Проиллюстрируем этот эффект при реализации модели динамики (17.59) нечеткой реляционной когнитивной карты, структура которой показана на рис. 17.9. Будем сравнивать результаты этого моделирования с моделью динамики когнитивной карты, основанной на выражении (17.58). Пусть нечеткие значения концептов представлены гауссовыми функциями принадлежности вида $\mu(x) = \exp(-(x-b)^2 / \sigma^2)$, где b – среднее значение, σ – параметр функций принадлежности, влияющий на «степень размытости» нечеткого отношения. Под входным сигналом будем понимать импульс изменения значения концептов в начальный момент времени. Нечеткие отношения являются нечетким расширением веса взаимовлияния концептов в четкой модели. Нечеткая реляционная модель реализована на основе min-max-нечеткой композиции. Начальные значения концептов задаются с помощью функций принадлежности со средним значением, равным соответствующему начальному состоянию концепта в четкой модели: $\mu_0(x) = \exp(-(x - K_i(0))^2 / \sigma^2)$. В процессе моделирования нечеткие состояния концептов изменяются. Изменение состояния, дающее нечеткое приращение, определяется с помощью операции нечеткого вычитания. Агрегирование влияний – на основе нечеткого сложения. На рис. 17.10 представлены результаты моделирования в системе MATLAB при следующих значениях весов влияния и начальных значениях концептов четкой модели: $w_{13} = -0.1$, $w_{23} = -0.2$, $w_{34} = 0.1$, $w_{41} = -0.5$, $w_{42} = 0.6$. $K_1(0) = 0$, $K_2(0) = 0$, $K_3(0) = 0$, $K_4(0) = 0$, $\Delta K_1(0) = 0.1$, $\Delta K_2(0) = 0.2$, $\Delta K_3(0) = -0.2$, $\Delta K_4(0) = -0.1$, $\sigma = 0.002$. Сплошные линии – дефазифицированные (центроидным методом) нечеткие значения концептов,

символами обозначены значения концептов соответствующей четкой модели (K_1 – «о», K_2 – «+», K_3 – «х», K_4 – «*»).

Из рис. 17.10 видно, что в течение нескольких тактов модельного времени обе модели (четкая и нечеткая) дают близкие результаты. Затем (по истечении приблизительно 15 эпох) увеличение степени неопределенности значений концептов (значения функций принадлежности приближаются к единице на всем базовом множестве, а дефазифицированные значения концептов стремятся к нулю) приводит к неработоспособности нечеткой модели в целом.

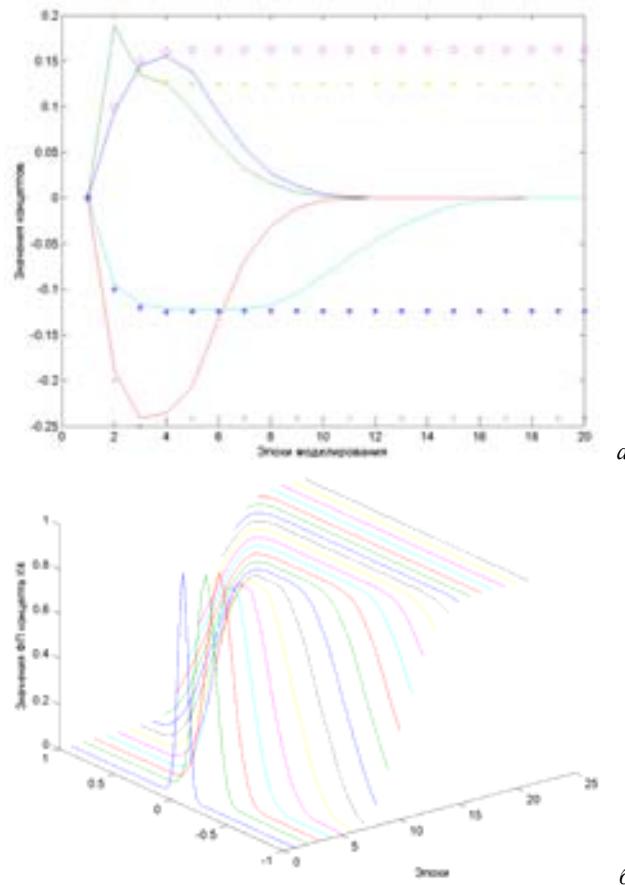


Рис. 17.10. Эффект расширения неопределенности при использовании обычной нечеткой арифметики в модели динамики нечеткой реляционной когнитивной карты: *а* – значения концептов; *б* – функция принадлежности концепта K_4

Одним из путей решения этой проблемы является дефазификация. Поставим задачу максимального сохранения «нечеткости» при реализации (17.59). В этом случае целесообразно проводить дефазификацию отдельных нечетких влияний входных концептов на выходной. Результат дефазификации будем представлять в виде одноточечного нечеткого множества с принадлежностью, равной единице, к значению базового множества, полученному при дефазификации.

При таком подходе в (17.59) можно сохранить все значения концептов нечеткими, а все операции реализовывать на основе обычной нечеткой арифметики. Расширение неопределенности при реализации операций агрегирования отдельных влияний устраниется при дефазификации, а нечеткое сложение с одноточечным нечетким множеством не изменяет степени неопределенности.

На рис. 17.11 проиллюстрирована работоспособность нечеткой реляционной когнитивной карты, использующая предложенное решение. Исходные данные аналогичны данным, примененным на рис. 17.10.

Как видно из рис. 17.11, при итерационных вычислениях не происходит расширения неопределенности и нечеткие значения концептов стабилизированы.

Примечание. Дефазифицированные значения концептов НРКК близки к значениям концептов соответствующей четкой когнитивной карты. Различия связаны с конечным шагом дискретизации базового множества при определении нечетких отношений и функций принадлежности. Значение этого шага ограничивает точность вычислений нечеткой модели.

Рассмотрим решение другой проблемы, возникающей при моделировании системной динамики с использованием НРКК, – возможного выхода нечетких значений концептов за диапазон базового множества (даже для устойчивой системы). Эта проблема существует для моделей динамики как четких, так и нечетких когнитивных карт. В случае нечетких арифметических вычислений она обостряется, так как при нечетком сложении и вычитании диапазон базового множества, на котором определены нечеткие операнды, удваивается. После выполнения операции при ограничении результата до диапазона, используемого при моделировании, функция принадлежности, имеющая ненулевую ширину, может быть «обрезана» по краю диапазона, даже если среднее значение лежит внутри диапазона базового множества. Это смещает центроид и вносит ошибку при последующей дефазификации.

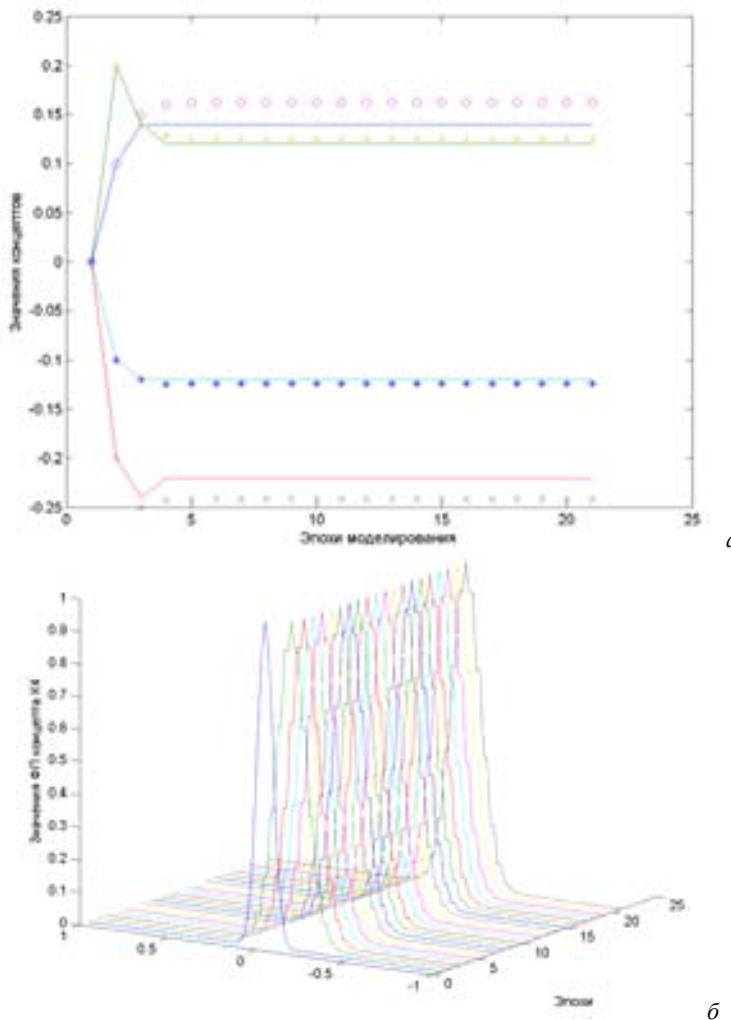


Рис. 17.11. Использование промежуточной дефазификации значений концептов при моделировании динамики в нечеткой реляционной когнитивной карте:
а – значения концептов; б – функция принадлежности концепта K_4

Кардинальный способ решения этой проблемы для нечетких когнитивных карт и их модификаций был предложен Коско. Он заключается в использовании нормирующих функций, например сигмоидальной (аналогично функциям активации искусственных нейронов). Нелинейные

нормирующие функции решают проблему выхода за диапазон значений, однако приводят к новым проблемам. В частности, введение нелинейности усложняет динамику нечеткой модели. Так, в [18] отмечено, что динамическое моделирование «нечетких» когнитивных карт Кооско может приводить к поведению типа странного аттрактора и предельного цикла.

Использование нелинейных функций при распространении влияния по нечеткой реляционной когнитивной карте также требует обоснования с учетом свойств используемых отношений и интерпретации получаемых результатов. Целесообразен гибкий подход, основанный на возможном применении одной из нескольких ограничивающих функций. Так, для диапазона значений $[-1, 1]$ можно использовать нелинейные функции типа $y = th(x)$ или кусочно-линейную функцию y :

$$y = \begin{cases} x, & |x| \leq 1, \\ 1, & x > 1, \\ -1, & x < -1. \end{cases} \quad (17.60)$$

Прямое использование функций ограничения в модели динамики (17.59) приводит к выражению

$$K_j(t+1) = f[K_j(t) \oplus \{\bigoplus_{i=1}^N [K_i(t) \simeq K_i(t-1)] \circ R_{ij}\}]. \quad (17.61)$$

Нелинейное преобразование в выражении (17.61) позволяет ограничить носители концептов в диапазоне $[-1, 1]$ базового множества.

Пример. Когнитивная карта на рис. 17.1 иллюстрирует пример формализованного представления проблемы управления обучением по охране труда уровня предприятия, полученного на основе анализа результатов экспертного опроса [21]. В данной когнитивной карте отношения влияния между концептами представлены в виде весов из диапазона $[-1, 1]$. На основе этой когнитивной карты строится НРКК в соответствии в подходом, изложенным в п. 17.7.3. В полученной НРКК значения концептов представляются в виде нечетких множеств, а отношения влияния между ними – в виде нечетких отношений.

Следует отметить, что концепты K_1-K_3 являются входными (управляющими). Остальные концепты могут быть как входными, так и целевыми (выходными, результирующими), внутренними, а также могут рассматриваться в качестве ограничений или индикаторов. Функциональная типизация концептов зависит от задачи моделирования и аспектов рассматриваемой проблемы.

В качестве примера анализа данной когнитивной карты рассмотрим задачу оценки качественного влияния концептов друг на друга на примере влияния концепта K_5 на остальные концепты с использованием методики, изложенной в п. 17.7.3.

Пусть начальные состояния всех концептов карты характеризуются средними значениями. При этом концепт K_5 может получать как положительное, так и отрицательное приращение. Результаты проведенного моделирования представлены на рис. 17.12. Сплошные линии показывают дефазифицированные нечеткие значения концептов НРКК, символами обозначены значения концептов исходной когнитивной карты (K_4 – «..», K_5 – «□», K_6 – «◊», K_7 – «○», K_8 – «+», K_9 – «×»).

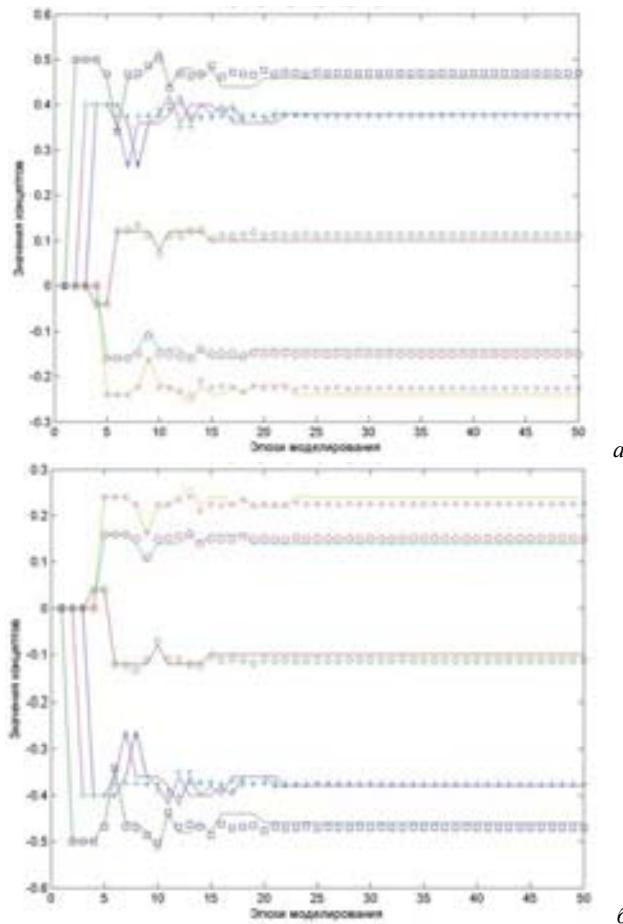


Рис. 17.12. Влияние концепта K_5 на концепты когнитивной карты системы управления обучением по охране труда уровня предприятия: *а* – положительное приращение K_5 ($\Delta K_5=0.5$); *б* – отрицательное приращение K_5 ($\Delta K_5=-0.5$)

Как видно из рис. 17.12, в установившемся режиме концепт K_5 наиболее сильное положительное влияние оказывает на концепты K_4 и K_8 , а наиболее сильное отрицательное – на концепт K_9 . Менее сильное положительное влияние концепт K_5 оказывает на концепт K_6 , а на концепт K_7 – небольшое отрицательное.

ВЫВОДЫ

Рассмотрены следующие основные разновидности нечетких моделей систем и процессов, отображаемых некоторыми структурами на основе графов (логико-временными последовательностями, пространственной распределенностью, функциональной взаимосвязанностью, причинно-следственными отношениями): нечеткие автоматы, нечеткие сети Петри, нечеткие ситуационные сети и нечеткие когнитивные карты.

Особое внимание уделено анализу способов построения, моделирования и использования различных типов когнитивных карт: знаковых, нечетких, нечетких продукционных, а также предложенных авторами обобщенных нечетких продукционных когнитивных карт и нечетких реационных когнитивных карт, обобщающих свойства существующих разновидностей когнитивных карт и реализующих расширенные возможности по анализу и моделированию сложных систем.

СПИСОК ЛИТЕРАТУРЫ

1. Нечеткие множества в системах управления и искусственного интеллекта/ Под ред. Д. А. Постепова. М.: Наука. Гл. ред. физ.-мат. лит., 1986.
2. Шапиро Д. И. Принятие решений в системах организационного управления: использование расплывчатых категорий. М.: Энергоатомиздат, 1983.
3. Мелихов А. Н., Бернштейн Л. С. Конечные четкие и расплывчатые множества. Ч. 2. Таганрог: ТРТИ, 1981.
4. Bellman R., Giertz M. On the analytic formalism on the theory of fuzzy sets // Information Sciences. 1974. V. 5. P. 149–157.
5. Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzyTECH. СПб.: БХВ-Петербург, 2003.
6. Мелихов А. Н., Бернштейн Л. С., Коровин С. Я. Ситуационные совещающие системы с нечеткой логикой. М.: Наука, 1990.

7. Хофман А. Введение в теорию нечетких множеств. М.: Радио и связь, 1982.
8. Brown S. M. Cognitive mapping and repertory grids for qualitative survey research: some comparative observations // Journal of Management Studies. 1992. V. 29. P. 287–307.
9. Carley K., Palmquist M. Extracting, representing, and analyzing mental models // Social Forces. 1992. V. 70. P. 601–636.
10. Montazemi A. R., Conrath D. W. The use of cognitive mapping for information requirements analysis // MIS Quarterly. 1986. V. 10. P. 45–55.
11. Nakamura K., Iwai S., Sawaragi T. Decision support using causation knowledge base // IEEE Transactions on Systems, Man, and Cybernetics SMC. 1982. V. 12. P. 765–777.
12. Kosko B. Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence. Prentice-Hall, Englewood Cliffs, New Jersey, 1992.
13. Klein J. H., Cooper D. F. Cognitive maps of decision-makers in a complex game // Journal of the Operational Research Society. 1982. V. 33. P. 63–71.
14. Zhang W. R., Chen S. S., Wang W., King R. S. A cognitive-map-based approach to the coordination of distributed cooperative agents // IEEE Trans. Systems Man Cybernet. 1992. V. 22. P. 103–114.
15. Kosko B., Mitaim S. Neural fuzzy agents for profile learning and adaptive object matching // Presence. 1998. V. 7. №. 6. P. 617–637.
16. Satur R., Liu Z. Q. A contextual fuzzy cognitive map framework for geographic information systems // IEEE Trans. Fuzzy Syst. 1999. V. 7. P. 481–494.
17. Styblinski M. A., Meyer B. D. Fuzzy cognitive maps, signal flow graphs, and qualitative circuit analysis // In Proc. of the 2nd IEEE International Conference on Neural Networks (ICNN-87), San Diego, California, 1988. P. 549–556.
18. Dickerson J. A., Kosko B. Virtual worlds as fuzzy cognitive maps // Presence. 1994. V. 3. P. 173–189.
19. Schneider M., Shnaider E., Kandel A., Chew G. Automatic construction of FCMs // Fuzzy Sets and Systems. 1998. V. 93. P. 161–172.
20. Максимов В. И. Структурно-целевой анализ развития социально-экономических ситуаций: Автореф. дис. ... д-ра М.: ИПУ РАН, 2002.

21. Борисов В. В., Бычков И. А., Федулов А. С. и др. Компьютерная поддержка сложных организационно-технических систем. М.: Горячая линия – Телеком, 2002.
22. Tolman E. C. Cognitive maps in rats and men // Psychological Review. 1948. V. 42. №. 55. P. 189–208.
23. Axelrod R. Structure of Decision: the cognitive maps of political elites. Princeton Univ, Press, N. Y.: 1976.
24. Трахтенгерц Э. А. Компьютерная поддержка принятия решений. М.: СИНТЕГ, 1998.
25. Классификация и кластер / Под ред. Дж. Вэн Райзина. М.: Мир, 1980.
26. Загоруйко Н. Г. Прикладные методы анализа данных и знаний. Новосибирск: Изд-во Ин-та мат., 1999.
27. Swkagi T., Iwai S., Katai O. An integration on qualitative causal knowledge for user-oriented decision support // Contr. Theory and Advanced Technology. 1986. V. 2. № 3. P. 451–482.
28. Wellman M. P. Inference in cognitive maps // Mathematics and Computers in Simulation. 1994. V. 36. P. 137–148.
29. Chaib-draa B., Decharnais J. A relational model of cognitive maps // Int. J. Human-Computer Studies. 1998. V. 49. P. 181–200.
30. Kosko B. Fuzzy cognitive maps // Int. Journal of Man-Machine Studies. 1986. V. 24. P. 65–75.
31. Zhang W. R., Chen S. S. A logical architecture for cognitive maps // In Proc. of the 2nd IEEE International Conference on Neural Networks (ICNN-88). 1988. V. 1. P. 231–238.
32. Zhang W. R., Chen S., King R. S. A cognitive map based approach to the coordination of distributed cooperative agents // IEEE Transactions on Systems, Man and Cybernetics 1992. . V. 22. P. 103–113.
33. Reinmann S. On the design of artificial auto-associative neural networks // Neural Networks. 1998. V. 11. P. 611–621.
34. Робертс Ф. С. Дискретные модели с приложениями к социальным, биологическим и экологическим задачам. М.: Наука, 1986.
35. Максимов В. И. Когнитивные технологии – от незнания к пониманию // Материалы I Междунар. конф. «Когнитивный анализ и управление развитием ситуаций (CASC`2001)»: В 3 т. М.: ИПУ РАН, 2001. Т. 1. С. 4–41.

36. Максимов В. И. Когнитивный анализ и управление развитием ситуаций // Материалы I Междунар. конф. «Когнитивный анализ и управление развитием ситуаций (CASC'2001)»: В 3 т. М.: ИПУ РАН, 2001. Т. 2. С. 10–21.
37. Stylios C. D., Groumpos P. P. Application of fuzzy cognitive maps in large manufacturing systems // In Proc. of the IFAC LSS'98, Rio, Patras, Greece. V. 1. 1998. P. 531–536.
38. Stylios C. D., Groumpos P. P. Fuzzy cognitive maps in modeling supervisory control systems // Journal of Intelligent & Fuzzy Systems. 2000. V. 8. №. 2. P. 83–98.
39. Силов В. Б. Принятие стратегических решений в нечеткой обстановке. М.: ИНПРО–PEC, 1995.
40. Дюбуа Д., Прад А. Теория возможностей. Приложения к представлению знаний в информатике. М.: Радио и связь, 1990.
41. Козелецкий Ю. Психологическая теория решений. М.: Прогресс, 1979.
42. Carvalho J. P., Tomé J. A. Rule-based fuzzy cognitive maps and fuzzy cognitive maps – a comparative study // In Proc. of the 18th International Conference of the North American Fuzzy Information Processing Society. NAFIPS'99, New York, 1999. P. 115 – 119.
43. Carvalho J. P., Tomé J. A. Automatic implementation and simulation of qualitative cognitive maps // In Proc. of the International Conference on Computational Science, San Francisco, California, USA, 2001. P. 217–221.
44. Carvalho J. P., Tomé J. A. Rule-based fuzzy cognitive maps – expressing time in qualitative system dynamics // In Proc. of the FUZZ-IEEE'2001, Melbourne, Australia. P. 280–283.
45. Carlsson C., Fullér R. Adaptive fuzzy cognitive maps for hyperknowledge representation in strategy formation process // In Proc. of the International Panel Conference on Soft and Intelligent Computing, Technical University of Budapest, 1996. P. 43–50.
46. Kosko B. Differential hebbian learning // AIP Conference Proceedings. V. 151. 1986. P. 265–270.
47. Huerga A. A balanced differential learning algorithm in fuzzy cognitive maps // In Proc. of the Sixteenth International Workshop on Qualitative Reasoning, QR'2002, Spain, 2002. P. 210–214.

48. Комарцова Л. Г. Исследование нейросетевых и гибридных методов и технологий в интеллектуальных системах поддержки принятия решений: Автoref. дис. ... д-ра М.: МЭИ (ТУ), 2003.
49. Chen S., Billings S. Neural Networks for Nonlinear Dynamic System Modeling and Identification // Int. J. Control. 1992. V. 56. P. 319–329.
50. Борисов В. В., Федулов А. С. Обобщенные нечеткие когнитивные карты // Нейрокомпьютеры: разработка, применение. 2004. № 4. С. 3–21.
51. Борисов В. В., Федулов А. С., Мнев В. И. Моделирование сложных нечетко-целевых систем на основе обобщенных нечетких когнитивных карт // Информ. бюл. Акад. воен. наук. Смоленск: Изд-во ВУ ВПВО ВС РФ, 2001. С. 22–25.
52. Борисов А. Н., Алексеев А. В., Меркульева Г. В. и др. Обработка нечеткой информации в системах принятия решений. М.: Радио и связь, 1989.
53. Борисов В. В., Федулов А. С. Нечеткие оценочные модели сложных систем с учетом согласования неравнозначных целей // Нейрокомпьютеры: разработка, применение. 2003. № 5. С. 3–12.
54. Федулов А. С. Нечеткие реляционные когнитивные карты // Изв. РАН. Теория и системы управления. 2005. № 1. С. 120–133.
55. Заде Л. А. Понятие лингвистической переменной и его применение к принятию приближенных решений. М.: Мир, 1976.
56. Pedrycz W. An identification algorithm in fuzzy relational systems // Fuzzy Sets and Systems. 1984. V. 13.
57. Pedrycz W. Fuzzy Control and Fuzzy Systems. N. Y.: John Wiley and Sons, 1993.
58. Kosko B. Fuzzy systems as universal approximators // IEEE Trans. Computers. 1994. V. 43.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ЧАСТЬ I. КЛАССИФИКАЦИЯ СИСТЕМ И НЕЧЕТКИХ МОДЕЛЕЙ	6
Глава 1. ПОНЯТИЕ СИСТЕМЫ. КЛАССИФИКАЦИЯ И ХАРАКТЕРИСТИКА СИСТЕМ.....	6
1.1. Понятие и определения системы.....	6
1.2. Классификация систем	7
Глава 2. КЛАССИФИКАЦИЯ МОДЕЛЕЙ СИСТЕМ	10
Глава 3. ОБЛАСТИ ПРИМЕНЕНИЯ И КЛАССИФИКАЦИЯ НЕЧЕТКИХ МОДЕЛЕЙ	16
3.1. Области применения нечеткого моделирования	16
3.2. Классификация нечетких моделей	20
3.3. Интеграция нечетких и нейронных сетей.....	21
ВЫВОДЫ	22
СПИСОК ЛИТЕРАТУРЫ	23
ЧАСТЬ II. НЕЧЕТКИЕ ПРОДУКЦИОННЫЕ И РЕЛЯЦИОННЫЕ МОДЕЛИ.....	25
Глава 4. НЕЧЕТКИЕ ПРОДУКЦИОННЫЕ МОДЕЛИ	25
4.1. Компоненты нечетких продукционных моделей.....	25
4.2. Способы нечеткого вывода.....	26
4.3. Создание базы нечетких продукционных правил.....	32
4.4. Введение нечеткости	49
4.5. Агрегирование степени истинности предпосылок правил	50
4.6. Активизация заключений правил	52
4.7. Аккумулирование активизированных заключений правил	54
4.8. Приведение к четкости.....	54
4.9. Параметрическая оптимизация конечной базы нечетких правил.....	60
Глава 5. АЛГОРИТМЫ НЕЧЕТКОГО ВЫВОДА	62
5.1. Алгоритм нечеткого вывода Мамдани	62
5.2. Алгоритм нечеткого вывода Ларсена	65
5.3. Алгоритм нечеткого вывода Цукамото	66
5.4. Упрощенный алгоритм нечеткого вывода.....	68

5.5. Алгоритм нечеткого вывода Такаги–Сугэно.....	69
5.6. Алгоритм нечеткого вывода на основе нечеткой продукционной модели с адаптацией операций над нечеткими множествами	71
5.7. Аппроксимационные свойства нечетких продукционных моделей	75
Глава 6. НЕЧЕТКИЕ РЕЛЯЦИОННЫЕ МОДЕЛИ И РЕЛЯЦИОННОЕ ПРЕДСТАВЛЕНИЕ НЕЧЕТКИХ ПРОДУКЦИОННЫХ МОДЕЛЕЙ	83
6.1. Нечеткие реляционные модели	83
6.2. Реляционное представление нечеткого вывода с использованием отдельных правил.....	84
6.3. Реляционное представление нечеткого вывода с использованием базы правил.....	87
6.4. Подобие нечетких реляционных и продукционных моделей.....	88
ВЫВОДЫ	90
СПИСОК ЛИТЕРАТУРЫ.....	91
ЧАСТЬ III. НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ.....	95
Глава 7. НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ С ПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИЕЙ ПРАВИЛ НА ОСНОВЕ АЛГОРИТМОВ ОБУЧЕНИЯ	97
7.1. Нечеткие нейронные продукционные сети типа ANFIS	97
7.2. Нечеткая нейронная продукционная сеть Ванга–Менделя.....	105
7.3. Нечеткая нейронная продукционная сеть Такаги–Сугэно–Канга	113
Глава 8. НЕЧЕТКИЕ НЕЙРОННЫЕ ПРОДУКЦИОННЫЕ СЕТИ С РЕАЛИЗАЦИЕЙ КОМПОНЕНТОВ НЕЧЕТКИХ ПРОДУКЦИОННЫХ МОДЕЛЕЙ НА ОСНОВЕ НЕЙРОСЕТЕВОЙ ТЕХНОЛОГИИ	119
8.1. Построение функций принадлежности предпосылок и заключений нечетких продукционных правил	119
8.2. Формирование предпосылок нечетких продукционных правил	120
8.3. Формирование заключений нечетких продукционных правил	123
8.4. Разбиение пространств входных переменных и формирование многомерных функций принадлежности предпосылок	126

8.5. Нечеткие производственные сети с представлением структуры в виде нейронных сетей	129
ВЫВОДЫ	131
СПИСОК ЛИТЕРАТУРЫ	133
 ЧАСТЬ IV. НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ	134
Глава 9. НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ С ВВЕДЕНИЕМ НЕЧЕТКОСТИ В СТРУКТУРУ	135
Глава 10. НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ НА ОСНОВЕ НЕЧЕТКИХ НЕЙРОНОВ	141
Глава 11. ГИБРИДНЫЕ НЕЙРОННЫЕ НЕЧЕТКИЕ СЕТИ НА ОСНОВЕ НЕЙРОНОВ, РЕАЛИЗУЮЩИХ НЕЧЕТКИЕ ОПЕРАЦИИ	145
11.1. Нейроны, реализующие нечеткие операции	146
11.2. Примеры построения и использования гибридных нейронных нечетких сетей.....	148
Глава 12. ОБУЧЕНИЕ НЕЙРОННЫХ НЕЧЕТКИХ СЕТЕЙ	173
12.1. Использование нейронных сетей с обучением с обратным распространением ошибки для реализации нечетких моделей.....	175
12.2. Обучение нейронных нечетких сетей типа 2 с нечеткими входами и выходами и четкими весами.....	178
12.3. Обучение нейронных нечетких сетей типа 3 с нечеткими входами, выходами и весами	181
Глава 13. ИСПОЛЬЗОВАНИЕ НЕЧЕТКИХ ПРОДУКЦИОННЫХ СЕТЕЙ В НЕЙРОННЫХ СЕТЯХ	185
ВЫВОДЫ	190
СПИСОК ЛИТЕРАТУРЫ	191
 ЧАСТЬ V. НЕЧЕТКИЕ МОДЕЛИ С ПРЕДСТАВЛЕНИЕМ НА ОСНОВЕ ГРАФОВ	195
Глава 14. НЕЧЕТКИЕ АВТОМАТЫ	195
14.1. Основные понятия, классификация нечетких автоматов	195
14.2. Обучение нечетких автоматов	201
Глава 15. НЕЧЕТКИЕ СЕТИ ПЕТРИ	202
15.1. Основные понятия, классификация нечетких сетей Петри	203
15.2. Нечеткие сети Петри с нечеткостью задания начальной маркировки	205

15.3. Нечеткие сети Петри с нечеткостью задания начальной маркировки и срабатывания переходов	209
15.4. Нечеткие сети Петри с нечеткостью задания начальной маркировки, времен задержки маркеров в позициях и времен срабатывания активных переходов	212
15.5. Задачи анализа свойств нечетких сетей Петри	216
Глава 16. НЕЧЕТКИЕ СИТУАЦИОННЫЕ СЕТИ	218
16.1. Формирование нечеткой ситуационной сети	219
16.2. Задачи моделирования процессов управления на основе нечетких ситуационных сетей	226
Глава 17. НЕЧЕТКИЕ КОГНИТИВНЫЕ КАРТЫ	230
17.1. Задачи построения и анализа когнитивных карт	231
17.2. Обзор способов построения и анализа когнитивных карт	232
17.3. Обучение нечетких когнитивных карт	249
17.4. Динамическое моделирование с помощью когнитивных карт	250
17.5. Требования к обобщенному представлению и анализу нечетких когнитивных карт	252
17.6. Обобщенные нечеткие продукционные когнитивные карты ..	253
17.7. Нечеткие реляционные когнитивные карты	261
ВЫВОДЫ	275
СПИСОК ЛИТЕРАТУРЫ	275