

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО АТОМНОЙ ЭНЕРГИИ  
РОССИЙСКАЯ АКАДЕМИЯ НАУК  
РОССИЙСКАЯ АССОЦИАЦИЯ НЕЙРОИНФОРМАТИКИ  
МОСКОВСКИЙ ИНЖЕНЕРНО-ФИЗИЧЕСКИЙ ИНСТИТУТ  
(ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ)

---

**НАУЧНАЯ СЕССИЯ МИФИ–2005**

**НЕЙРОИНФОРМАТИКА–2005**

**VII ВСЕРОССИЙСКАЯ  
НАУЧНО-ТЕХНИЧЕСКАЯ  
КОНФЕРЕНЦИЯ**

**ЛЕКЦИИ  
ПО НЕЙРОИНФОРМАТИКЕ**

По материалам Школы-семинара  
«Современные проблемы нейроинформатики»

Москва 2005

УДК 001 (06)+004.032.26 (06)

ББК 72я5+32.818я5

М82

**НАУЧНАЯ СЕССИЯ МИФИ–2005. VII ВСЕРОССИЙСКАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ «НЕЙРОИНФОРМАТИКА–2005»: ЛЕКЦИИ ПО НЕЙРОИНФОРМАТИКЕ.** – М.: МИФИ, 2005. – 214 с.

В книге публикуются тексты лекций, прочитанных на Школе-семинаре «Современные проблемы нейроинформатики», проходившей 26–28 января 2005 года в МИФИ в рамках VII Всероссийской конференции «Нейроинформатика–2005».

Материалы лекций связаны с рядом проблем, актуальных для современного этапа развития нейроинформатики, включая ее взаимодействие с другими научно-техническими областями.

Ответственный редактор

*Ю. В. Тюменцев*, кандидат технических наук

ISBN 5–7262–0526–X

© *Московский инженерно-физический институт  
(государственный университет), 2005*

## Содержание

|   |            |
|---|------------|
| <b>Н. Г. Ярушкина. Нечеткие нейронные сети в когнитивном моделировании и традиционных задачах искусственного интеллекта</b> | <b>166</b> |
| Введение . . . . .  | 167        |
| Нечеткие нейронные сети (ННС) . . . . .   | 170        |
| Определение нечеткого множества . . . . .   | 170        |
| Определение нечеткой системы . . . . .  | 170        |
| Системы нечеткого вывода . . . . .  | 171        |
| Определения <i>И</i> - и <i>ИЛИ</i> -нейронов . . . . .   | 172        |
| Определение нечеткой нейронной сети (ННС) . . . . .   | 174        |
| ANFIS — адаптивная НС, основанная на системе нечеткого вывода . . . . .   | 174        |
| Задачи когнитивного моделирования: Возможности ННС . . . . .  | 176        |
| Цели и задачи когнитивной науки . . . . .   | 176        |
| Задача моделирования человеческого восприятия . . . . .   | 177        |
| Синергетический компьютер . . . . .   | 177        |
| Задача исследования баланса право- и левополушарных методов . . . . .   | 179        |
| ННС — сочетание образного восприятия и логической обработки . . . . .   | 179        |
| Задача самоорганизации знаний . . . . .   | 180        |
| Нечеткая кластеризация . . . . .  | 180        |
| Задача исследования соотношения осознанного и подсознательного . . . . .  | 182        |
| ННС и возможности генерации правил поведения . . . . .  | 182        |
| Искусственный интеллект, нейроинтеллект, естественный интеллект . . . . .   | 183        |
| Задачи искусственного интеллекта: Возможности ННС . . . . .   | 184        |
| Научное направление «искусственный интеллект» (ИИ) . . . . .  | 184        |
| Состояние ИИ . . . . .  | 185        |
| Искусственный и вычислительный интеллект . . . . .  | 186        |
| Традиционная задача ИИ — инженерия знаний и экспертные системы . . . . .  | 186        |
| Мягкие экспертные системы . . . . .   | 188        |
| Сравнение нечеткой и мягкой экспертной систем . . . . .   | 188        |

|  |     |
|--|-----|
| Инженерия знаний и мягкие экспертные системы . . .   | 189 |
| Представление знаний в мягкой экспертной системе.<br>Содержание баз знаний и данных мягкой<br>экспертной системы . . . . .               | 189 |
| Пример мягкой ЭС для проектирования радиоэлектронных<br>изделий . . . . .  | 190 |
| Описание задачи конструирования РЭА . . . . .  | 190 |
| Нейросетевая реализация МЭС конструирования стен-<br>дов контроля РЭА . . . . .  | 191 |
| Задачи прикладного ИИ: Возможности ННС . . . . .   | 193 |
| Интеллектуальный анализ данных . . . . .   | 193 |
| ННС как Data Miner для нечетких реляционных баз данных<br>Нечеткая реляционная модель данных. Способ пред-<br>ставления данных . . . . . | 193 |
| Кластеризация реляционных кортежей . . . . .   | 194 |
| Решения задачи нечеткой кластеризации на основе<br>сети Кохонена . . . . .   | 196 |
| Обучение сети . . . . .  | 197 |
| Режимы работы сети . . . . .   | 199 |
| Извлечение правил из реляционных кортежей с помощью ННС  | 199 |
| Прогнозирование нечетких временных рядов . . . . .   | 199 |
| Определение нечеткого временного ряда . . . . .  | 200 |
| Определение нечеткой тенденции . . . . .   | 200 |
| Распознавание нечеткой тенденции с помощью ННС   | 204 |
| Нейросетевое решение задачи прикладного ИИ — развитие интел-<br>лектуальных информационных хранилищ . . . . .                            | 206 |
| Определение информационного хранилища (ИХ) . . . . .   | 206 |
| Основные функции системы управления ИХ . . . . .   | 206 |
| Нейросетевая система управления ИХ . . . . .   | 208 |
| Нечеткая кластеризация информационных ресурсов с помо-<br>щью сети Кохонена. Морфологический анализ тек-<br>ста. Стемминг . . . . .      | 209 |
| Классификация информационного ресурса на основе ННС .  | 210 |
| Заключение . . . . .   | 212 |
| Литература . . . . .   | 212 |

**Н. Г. ЯРУШКИНА**

Ульяновский государственный технический университет  
УлГТУ, г. Ульяновск,  
**E-mail: jng@ulstu.ru**

**НЕЧЕТКИЕ НЕЙРОННЫЕ СЕТИ В КОГНИТИВНОМ  
МОДЕЛИРОВАНИИ И ТРАДИЦИОННЫХ ЗАДАЧАХ  
ИСКУССТВЕННОГО ИНТЕЛЛЕКТА**

**Аннотация**

Нечеткие нейронные сети (ННС) представляют собой реализацию систем нечеткого логического вывода методами нейронных сетей. Рассмотрен когнитивный потенциал ННС для широкого спектра задач: соотношения логического и образного в мышлении, осознанного и неосознанного, обобщенного и конкретного. Задачи использования когнитивных моделей в искусственных системах решает традиционный искусственный интеллект. В лекции сопоставлены задачи инженерии знаний и возможности нейросетевых технологий. Приведены примеры использования ННС для создания прикладных интеллектуальных систем.

**N. G. YARUSHKINA**

Ulyanovsk State Technical University,  
UISTU, Ulyanovsk,  
**E-mail: jng@ulstu.ru**

**FUZZY NEURON NETWORKS IN COGNITIVE MODELLING AND  
ARTIFICIAL INTELLIGENCE**

**Abstract**

Fuzzy neuron networks (FNN) with genetic optimization are fuzzy logic inference systems. The FNNs have cognitive possibilities for development of different human brain functions models. A basic result of Artificial Intelligence is Knowledge Engineering. In this Lecture knowledge engineering and possibilities of the FNN are compared. An example is considered for some kind of applied intelligence system.

## Введение

В современной нейроинформатике соотношение фундаментальных и технологических задач складывается в пользу технологических. Отметим, что фундаментальные задачи — моделирование ощущений, восприятий, распознавание образов, обучение и запоминание паттернов являются основными задачами когнитивной науки. Технологические задачи связаны с повышением эффективности использования нейронных сетей в решении задач кластеризации, классификации образов и сцен, аппроксимации функций для процессов технической, экономической, биологической природы. Достижения нейроинформатики в решении технологических задач привели к созданию целого спектра разнообразных нейронных сетей: сетей с прямым распространением сигнала, рекуррентных сетей, радиально базисных сетей. Созданы развитые архитектуры гибридных систем: нечетких нейронных сетей, нечетких нейронных с генетической настройкой параметров.

Ситуация в нейроинформатике требует вернуться к фундаментальным когнитивным задачам и выяснить, позволяют ли новые классы нейронных и гибридных систем решать новые когнитивные задачи. Далее рассматриваются в основном нечеткие нейронные сети с генетической оптимизацией параметров, так как они обладают комплексной структурой и сложными алгоритмами обучения, интегрирующими структурные части различных нейронных сетей. Следовательно, именно гибридные системы, как можно предположить, должны обладать сильным когнитивным потенциалом. Далее будем называть именно их гибридными системами. Гибридные системы включают в себя слои радиально базисных нейронов, логических нейронов, традиционных пороговых суммирующих нейронов. Алгоритм их обучения обычно комбинирует соревновательное обучение (по алгоритму победителя), генетическую оптимизацию параметров и классический метод обратного распространения ошибки.

Рассмотрим также некоторые новые применения гибридных интеллектуальных систем, характеризующие их прикладной потенциал, в том числе для традиционных задач искусственного интеллекта. Выберем направления, сочетающие распознавание и логическую обработку. На наш взгляд место для успешных применений гибридных технологий может быть найдено в направлении интеллектуализации информационных систем: баз данных и корпоративных хранилищ ресурсов.

### **Интеллектуализация баз данных**

Технология баз данных является важнейшей из прикладных технологий информационных систем, так как на ней строятся экономические информационные системы любого класса. Потребность в интеллектуализации баз данных связана с несколькими причинами.

Базы данных состоят из десятков тысяч рабочих таблиц, индексов, ограничений, доменов, представлений. Простого словаря, регистрирующего объекты, недостаточно для управления данными. Обработка транзакций в корпоративных системах (более 1000 автоматизированных рабочих мест) требует встраивания в сервер данных логики событий. Состоявшаяся в свое время в искусственном интеллекте широкая дискуссия о том, чем данные отличаются от знаний, остановилась на мнении, что «знания, это данные и их интерпретация», то есть знания — это еще и метазнания. Ограничения целостности, характерные для серверов данных, это всегда знания о законах проблемной области. Показательно, что крупные производители систем баз данных (Oracle Inc.) включают в состав программного продукта систему описания и интерпретации бизнес-правил в форме продукций ЕСЛИ–ТО и систему логического вывода действий. Любое действие, которое выводится из бизнес-правил, реализуется в системе как транзакция обработки данных.

### **Новые возможностные типы данных**

Взаимодействие баз данных с системами обработки знаний расширяет базовые типы данных реляционной модели данных. Если система знаний представлена системой нечетких продукций, то такой системе становится необходимой среда хранения и обработки нечетких атрибутов. В настоящее время построена нечеткая реляционная алгебра, выполняющая операции соединения, селекции, проекции, объединения и пересечения. Нечеткий реляционный сервер данных — это репозиторий, который предназначен для хранения функций принадлежности, лингвистических меток.

### **Интеллектуальный анализ данных Data Mining**

Анализ данных (Data Mining) — это научное направление, изучающее методы извлечения законов проблемной области из баз данных. Такие «законы»,

разумеется, не имеют строгой аналитической формы, в лучшем случае аппроксимируются некоторым набором правил ЕСЛИ–ТО. Следовательно, Data Miner имеет на входе базу данных (обучающую выборку), а на выходе базу правил (базу знаний). Но именно такие входы и выходы имеет нечеткая нейронная сеть, которая в таком контексте становится средством класса Data Miner.

### **Интеллектуализация Intranet и Internet-технологий**

Корпоративные хранилища информационных ресурсов строятся на основе технологии менеджеров файлов и внутреннего WWW-сервиса. Под ресурсом понимают совокупность файлов, объединенных общей семантикой. Хранение и использование таких ресурсов требует построения индексов хранения, которые не просто обеспечивают быстрый поиск ресурса, а становятся метаописанием ресурсов, т. е. знаниями.

### **Интеллектуальная поисковая машина ресурсов**

Чтобы обеспечить хранение и поиск ресурсов, необходимо формировать индексы (каталоги) в соответствии с систематическими рубриками. Рубрики представляют собой классы объектов проблемной области, причем трудно вручную составить исходную достаточно полную классификацию ресурсов. Следовательно, необходимо решить задачу кластеризации — порождение системы нечетких классов для ресурсов. Затем, каждый новый ресурс необходимо относить к нечетким классам. Очевидно, что такую задачу можно решить только на основе аннотаций текстов, которые должны присутствовать в каждом ресурсе. Текст можно отнести к нечеткому классу на основе встречаемости ключевых слов. Для решения задачи кластеризации можно применить нейронную сеть Кохонена, обучаемую по алгоритму победителя, задача классификации ресурсов решается с помощью нечеткой нейронной сети, например, класса ANFIS.

## Нечеткие нейронные сети

### Определение нечеткого множества

Нечетким множеством  $f$  называют множество базовых элементов  $b$ , степень принадлежности которых выражается с помощью функции принадлежности, обозначаемой  $\mu$ . Функция  $\mu(b, f)$  отображает базовое значение  $b$  и нечеткое значение  $f$  в интервал  $[0, 1]$ . По определению,  $0 < \mu(b, f) < 1$  для  $b$  и  $f$ . На рис. 1 приведены нечеткие значения, характерные для переменной *скорость автомобиля*.



Рис. 1. Пример определения нечетких значений скорости автомобиля

### Определение нечеткой системы

Системы нечетких продукций строятся на основе понятия лингвистической переменной, которой называют пятерку объектов:

$$\langle x, T(x), U, G, M \rangle,$$

где  $x$  — собственное имя переменной;  $T(x)$  — терминальное множество, то есть набор значений переменной (нечетких меток);  $U$  — множество объектов (или универсум);  $G$  — синтаксические правила употребления;  $M$  — семантические правила употребления.

Обобщив нечеткую зависимость переменных  $x$  и  $y$ , получим продукцию: **если**  $A(x)$ , **то**  $B(y)$ . Цель нечеткого логического вывода — определить значения выходной переменной  $B'(y)$  в зависимости от значений входной переменной  $C(x')$ .

Основными правилами умозаключений являются два силлогизма: *modus ponens* (MP) и *modus tollens* (MT). Рассмотрим изменение основных правил вывода в случае, если предпосылка и заключение выражаются в терминах нечетких множеств:

Нечеткий *modus ponens* (Fuzzy MP):  
**если**  $A(x)$  **то**  $B(x)$ ;  $A'(x)$ , **то**  $B'(x)$ .

Нечеткий *Modus tollens* (FMT):  
**если**  $A(x)$  **то**  $B(x)$ ,  $\neg B'(x)$ , **то**  $\neg A'(x)$ .

#### Системы нечеткого вывода

Схема вывода на базе нечетких правил сводится к решению следующей задачи:

|             |  |
|-------------|--|
| База правил | $R_1$ : <b>если</b> $(X - A_1)$ <b>И</b> $(Y - B_1)$ <b>тогда</b> $(Z - C_1)$<br><b>в противном случае</b> |
|             | $R_2$ : <b>если</b> $(X - A_2)$ <b>И</b> $(Y - B_2)$ <b>тогда</b> $(Z - C_2)$<br><b>в противном случае</b> |
|             | ...  |
|             | $R_n$ : <b>если</b> $(X - A_n)$ <b>И</b> $(Y - B_n)$ <b>тогда</b> $(Z - C_n)$                              |
| Факт:       | $x = x_0, y = y_0$ .   |
| Следствие:  | $z = ?$  |

Интеллектуальная система осуществляет логический вывод по базе нечетких правил по этапам.

- Фаззификация фактических данных, т. е. точное значение  $x_0$  интерпретируется как нечеткая точка.
- Композиция входной переменной и условной части правила:  
 $x_0 \circ A_i, y_0 \circ B_i$ , т. е. вычисляется уровень пригодности правила к

ситуации. Если факт задан нечеткой точкой, то композиция сводится к выявлению соответствующей степени принадлежности.

- Вычисление нечеткой импликации

$$((x_0 \circ A_i) \cap (y_0 \circ B_i)) \rightarrow C_i$$

для  $\forall R$ . Результатом выполнения для всех правил являются  $N$  нечетких значений для выхода  $Z$ .

- Агрегация среднего значения, т. е. построение нечеткого значения выхода по результатам предыдущих этапов:

$$C = \bigcup_{i=1}^n C_i.$$

- Дефаззификация, т. е. выбор представляющего элемента по агрегированному нечеткому понятию.

### Определения И- и ИЛИ-нейронов

Гибридные системы, которые в качестве базовой используют нейронные сети, интерпретируемые как системы нечеткого вывода, называются нечеткими нейронными сетями. Рассмотрим далее нечеткие нейронные сети, как пример успешной технологии вычислительного интеллекта. Глубинная интеграция нечетких систем и нейросетей, связана с разработкой новой архитектуры элементов нейросети. Для интеграции двух технологий — нечетких систем и нейрокомпьютинга, необходимо предложить способ четкого дискретного представления непрерывных функций принадлежности, для чего выберем максимально большой интервал  $[x_1, x_2]$ , в котором представлены все нечеткие множества условных частей правил. Если разбить интервал с равным шагом, то любое нечеткое значение представляется четким вектором. Другой способ представления нечеткого понятия в виде четких данных состоит в представлении нечеткого множества в виде совокупности  $\alpha$ -срезов (рис. 2).

При использовании  $\alpha$ -срезов каждое  $\alpha_j$ -подмножество представляется двумя числами — левой и правой границами:  $\alpha - L_{ij}$ ,  $\alpha - R_{ij}$ , где  $j$  — номер  $\alpha$ -среза, а  $i$  — номера точек на его левой и правой границах, т. е.  $\alpha$ -срезы четко представляют непрерывную функцию принадлежности. Модификация модели нейрона для адаптации к нечетким системам касается выбора

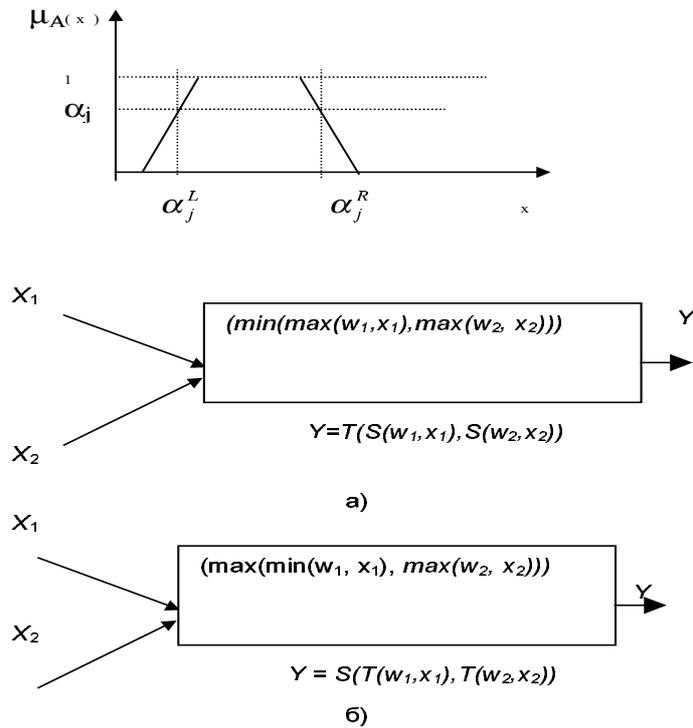


Рис. 2.  $\alpha$ -срез, И-ИЛИ нейроны

функции активации, реализации операций сложения и умножения, так как в нечеткой логике сложение моделируется любой треугольной конормой (например,  $\max, a + b - a * b, \dots$ ), а операция умножения – треугольной нормой ( $\min, a * b, \dots$ ).

**И-нейроном** называется нейрон, в котором умножение веса  $w$  на вход  $x$  моделируется конормой  $S(w, x)$ , а сложение – нормой  $T(w, x)$ .

Для двухвходового И-нейрона справедлива формула:

$$Y = T[S(w_1, x_1), S(w_2, x_2)].$$

**ИЛИ-нейроном** называется нейрон, в котором умножение веса  $w$  и

входа  $x$  моделируется нормой  $T(w, x)$ , а сложение взвешенных весов — конормой  $S(w, y)$ . Для двухвходового *ИЛИ*-нейрона справедлива формула:

$$Y = S[T(w_1, x_1), T(w_2, x_2)].$$

Если выбрать в качестве  $T$ -нормы  $\min$ , а  $\max$  — в качестве  $S$ -нормы, то формула преобразования *ИЛИ*-нейрона уточняется следующим образом:

$$\max[\min(w_1, x_1), \min(w_2, x_2)].$$

В качестве функции активации обычно используют радиальную базисную функцию  $F(x) = \exp[-b * (x^2 - a)]$ .

### Определение нечеткой нейронной сети

*Нечеткой нейронной сетью* (ННС) [5] обычно называют четкую нейросеть, которая построена на основе многослойной архитектуры с использованием *И*-, *ИЛИ*-нейронов. Нечеткая нейросеть функционирует стандартным образом на основе четких действительных чисел. Нечеткой является только интерпретация результатов. При создании гибридной технологии кроме объединения систем по данным можно использовать нейрокомпьютинг для решения частной подзадачи нечетких экспертных систем, а именно настройки параметров функции принадлежности. Функции принадлежности можно сформировать двумя способами: методом экспертной оценки; на основе статистики. Гибридные технологии предлагают третий способ: в качестве функции принадлежности выбирается параметризованная функция формы (например, гауссова кривая с параметрами  $b, a$ ), параметры которой настраиваются с помощью нейросетей. Настройка параметров может быть получена в рамках алгоритма обратного распространения ошибки. Таким образом, нечеткая нейронная сеть наряду с классическими нейронами, являющимися пороговыми суммирующими элементами, должна включать в себя *И*-, *ИЛИ*-нейроны. В качестве примера рассмотрим архитектуру нечеткой нейронной сети класса ANFIS, включающую адаптивный слой функций принадлежности и логические *И*-, *ИЛИ*-нейроны, «логические» нейроны, моделирующие логические связи.

### ANFIS — адаптивная НС, основанная на системе нечеткого вывода

Приведем на рис. 3 структуру ANFIS (Adaptive Network based Fuzzy Inference System) — адаптивной НС для двух правил:

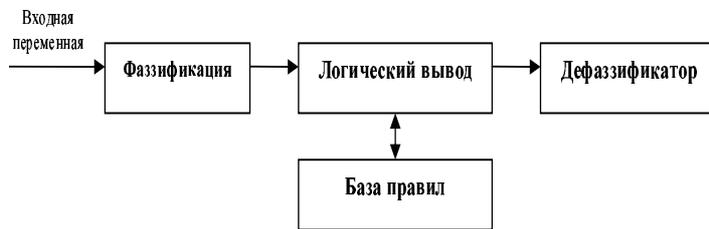
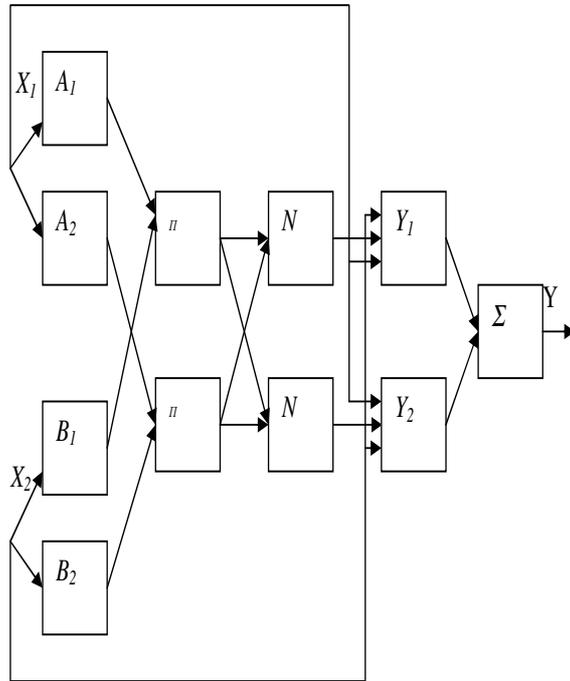


Рис. 3. Структуры ANFIS и нечеткой экспертной системы

**если**  $x_1 = A_1$  **И**  $x_2 = B_1$  **то**  $y_1 = c_{11} * x_1 + c_{12} * x_2$

**если**  $x_1 = A_2$  **И**  $x_2 = B_2$  **то**  $y_2 = c_{21} * x_1 + c_{22} * x_2$

Выход ННС формируется по формуле:

$$y = \frac{(w_1 * y_1 + w_2 * y_2)}{(w_1 + w_2)}.$$

Слой ННС ANFIS выполняют следующие функции.

Слой 1 представлен радиальными базисными нейронами и моделирует функции принадлежности.

Слой 2 — это слой И-нейронов, которые моделируют логическую связку И произведением  $w_i = \mu_{A_i}(x_1) * \mu_{B_i}(x_2)$ .

Слой 3 вычисляет нормированную силу правила:

$$w_i = \frac{w_i}{(w_1 + w_2)}.$$

Слой 4 формирует значение выходной переменной:

$$y(x_1, x_2) = w_i y_i = w_i (c_{i1} * x_1 + c_{i2} * x_2).$$

Слой 5 выполняет дефаззификацию:

$$y = w_1 * y_1 + w_2 * y_2.$$

Гибридная сеть архитектуры ANFIS обучается с помощью алгоритма обратного распространения ошибки.

## **Задачи когнитивного моделирования: Возможности ННС**

### **Цели и задачи когнитивной науки**

Исследование известных архитектур гибридных систем и алгоритмов их обучения позволяет, по нашему мнению, считать их моделями, адекватно отражающими соотношение восприятия и логического умозаключения при когнитивной деятельности человека. При реальном решении задачи человек комбинирует процессы мышления (компьютерный аналог — логический вывод), вспоминания (сознательное «внутреннее» возбуждение образа), восприятия окружающих предметов (распознавание) и, возможно, движения (управление телом). Обученная нечеткая нейронная сеть хранит

паттерны и выполняет логические операции, так как содержит *И-ИЛИ*-нейроны; она переключается с распознавания на логический вывод, так как обычно включает в себя несколько отдельно обученных сетей с разными функциями. Таким образом, несмотря на то, что нечеткие нейронные сети созданы как нейро-нечеткие контроллеры для управления процессами, они могут служить когнитивной моделью для изучения взаимодействия процессов восприятия и мышления при когнитивной деятельности.

### **Задача моделирования человеческого восприятия**

Особенность человеческого восприятия — это способность понимать искаженную информацию (ассоциативная память). Глядя на образ, мы выделяем примитивы, которые порождают параметр порядка. В результате процесса самоорганизации и при подчинении параметру порядка человеческий мозг воссоздает целый образ и распознает объект. На основе теории синергетики (нелинейной динамики) создан синергетический компьютер в форме алгоритма [1]. Синергетический компьютер — это программа распознавания человеческих лиц, основанная на принципах самоорганизации. Такой компьютер проверен на искажениях, амбивалентных изображениях.

### **Синергетический компьютер**

Детерминированный хаос — стереотип поведения многих синергетических систем, например, игровые автоматы часто используют запланированный хаос. Как хаос связан с синергетикой? Синергетическая система может управлять не одним а сразу несколькими параметрами порядка. Параметры сотрудничают, конкурируют, один из них может доминировать, но смена доминанты может быть хаотической. Следовательно, хаотическими являются те процессы, которые при малейшем изменении условий полностью изменяются. Хаос обладает двумя особенностями:

- чувствительность к исходным условиям;
- самоподобие.

Аттрактор — недостижимая точка притяжения состояния системы. Почему аттракторные НС имеют перспективы в решении интеллектуальных задач? Из детерминированного и микроскопического хаоса возникает порядок. Предъявленный сенсорам образ, организует НС и позволяет извлечь нужный паттерн.

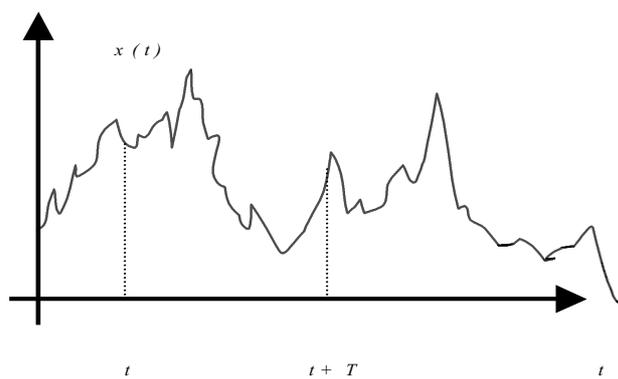
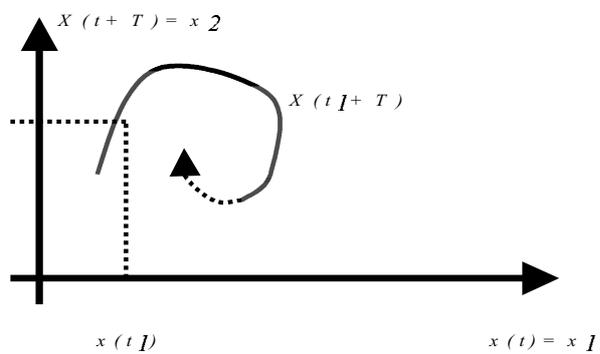


Рис. 4. Применение аттракторной НС для прогнозирования временного ряда

Например, рассмотрим применение аттракторной нейронной сети в анализе временных рядов (рис. 4). Необходима реконструкция аттракторов из временного ряда. В некоторый момент времени  $t_1$  определено значение  $x$ , второе значение определяется в момент, смещенный относительно первого на  $T$ . Таким образом, обрабатываются все точки ряда, строится траектория на плоскости  $X(t_1), X(t_1 + T)$ . По траектории можно проследить расположение аттрактора, т. е. выполнить экстраполяцию. В случае хаотических аттракторов для его реконструкции требуется, по меньшей мере, трехмерная система координат (а то и большей размерности). Соответствующие новые координаты получают при этом смещением временной оси не только  $T$ , но и  $2T$  и т. д. Проблема выбора смещения  $T$  не решена и определяется эмпирически.

#### **Задача исследования баланса право- и левополушарных методов**

В литературе встречаются схемы гибридизации нейроинформатики и ИИ, построенные по следующему принципу: правое полушарие — нейрокомпьютер; левое полушарие — основанная на знаниях система, а вопрос лишь в их взаимодействии или балансе право- и лево-полушарности. Причем гибридизация является именно глубинной, так как одна нейронная сеть содержит и логические (дискретные) *И-ИЛИ*-нейроны, и непрерывные традиционные пороговые суммирующие нейроны. Таким образом происходит создание структуры, объединяющей обработку сигнала и оценку такой обработки, в том числе логическую.

#### **ННС — сочетание образного восприятия и логической обработки**

В реальном поведении человека невозможно разделить восприятие и логическую обработку, поэтому более успешной представляется схема глубинной интеграции. Примерами таких глубинных успешных гибридов может служить создание нечетких нейронных сетей, объединяющих методы обучения нейронных сетей, управление неопределенностью нечеткой логики и логический вывод дедуктивных систем.

### Задача самоорганизации знаний

Интеграция НС и вероятностных вычислений привела к созданию и широкому использованию радиальных базисных сетей, сетей регрессии, вероятностных сетей, байесовских сетей. Первой особенностью таких сетей является параллельное формирование не только решения, но и вероятностной оценки такого решения. Например, сети регрессии, которые используют в основном для классификации, не только соотносят с классом входной вектор, но и формируют вероятность такого классифицирования. Таким образом, вероятностные нейронные сети сочетают собственно распознавание образа с формированием его оценки и могут служить для моделирования восприятия. Второй особенностью вероятностных НС служит сложная послойная структура, так как разные слои содержат разные типы нейронов. Такая структура приводит к сложным многошаговым алгоритмам обучения, которые часто включают в себя соревновательное обучение по Хохену, обучение на основе штрафов, по алгоритму обратного распространения ошибки. Такие алгоритмы обучения далеки от минимизации функции ошибки с помощью метода градиентного спуска, они комбинируют кластеризацию образцов, поиск зависимостей, то есть моделируют обобщение, порождение понятий в ходе обучения. Рассмотрим далее базовый механизм обобщения — кластеризацию, в том числе нечеткую. Приведем постановку задачи нечеткой кластеризации.

### Нечеткая кластеризация

Алгоритм нечеткой кластеризации называют FCM-алгоритмом (Fuzzy Classifier Means, Fuzzy C-Means). Целью FCM-алгоритма кластеризации является автоматическая классификация множества объектов, которые задаются векторами признаков в пространстве признаков. Другими словами, такой алгоритм определяет кластеры и, соответственно, классифицирует объекты. Кластеры представляются нечеткими множествами и, кроме того, границы между кластерами также являются нечеткими. FCM-алгоритм кластеризации предполагает, что объекты принадлежат всем кластерам с определенной функцией принадлежности (ФП). Степень принадлежности определяется расстоянием от объекта

Для заданного множества  $K$  входных векторов  $x_k$  и  $N$  выделяемых кластеров  $c_j$  предполагается, что любой  $x_k$  принадлежит любому  $c_j$  со степенью принадлежности  $\mu_{jk}$ , где  $j$  — номер кластера, а  $k$  — номер входного

вектора. Принимаются во внимание следующие условия нормирования для  $\mu_{jk}$ :

$$\sum_{j=1}^N \mu_{jk} = 1, \text{ для всех } k = 1, \dots, K,$$

$$0 \leq \sum_{j=1}^N \mu_{jk} \leq K, \text{ для всех } j = 1, \dots, N.$$

Цель алгоритма заключается в минимизации суммы всех взвешенных расстояний  $\|x_k - c_j\|$

$$\sum_{j=1}^N \sum_{k=1}^K (\mu_{jk})^q \|x_k - c_j\| \rightarrow \min,$$

где  $q$  — фиксированный параметр, задаваемый перед итерациями. Для достижения вышеуказанной цели необходимо решить следующую систему уравнений:

$$\frac{\partial \sum_{j=1}^N \sum_{k=1}^K (\mu_{jk})^q \|x_k - c_j\|}{\partial \mu_{jk}} = 0,$$

$$\frac{\partial \sum_{j=1}^N \sum_{k=1}^K (\mu_{jk})^q \|x_k - c_j\|}{\partial c_j} = 0.$$

Совместно с условиями нормирования  $\mu_{jk}$  данная система дифференциальных уравнений имеет следующее решение:

$$c_j = \frac{\sum_{k=1}^K (\mu_{jk})^q x_k}{\sum_{k=1}^K (\mu_{jk})^q}$$

(взвешенный центр гравитации) и

$$\mu_{jk} = \frac{\frac{1}{\|x_k - c_j\|^{\frac{1}{(q-1)}}}}{\sum_{j=1}^N \frac{1}{\|x_k - c_j\|^{\frac{1}{(q-1)}}}}.$$

**Задача исследования соотношения осознанного и подсознательного**

Особенностью гибридных систем является их принципиальная интерпретируемость, то есть как и всякая система логического вывода, гибридная система объясняет свой результат с помощью обратного просмотра протокола применяемых вербализованных правил. Любая нечеткая нейронная сеть работает как система нечеткого логического вывода, но строится не с помощью инженерии знаний, а с помощью обучения по образцам. В результате матрица весов отражает силу связи входных и выходных переменных. Результатом обучения нечеткой нейронной сети служит не только матрица весов, но совокупность правил и оценок их достоверности. Следовательно, любая гибридная система является как минимум двухуровневой, включающей систему и метасистему, отражающую систему первого уровня. Исходя из сказанного, можно предложить нечеткие нейронные сети в качестве когнитивной модели соотношения сознательного (логического) и бессознательного (аналогового, вычислительного) процессов в решении интеллектуальной задачи. Таким образом, в нечеткой нейронной сети возможно сочетать манипулирование образами, заданными количественными параметрами, с преобразованием символов (слов). Такая возможность заложена в самой базовой конструкции теории нечетких множеств. Каждое нечеткое множество связывает слово, имя, с порядковой или метрической шкалой с помощью функции принадлежности, т. е. количественно моделирует новое качество — смысл.

**ННС и возможности генерации правил поведения**

Эффективность аппарата нейросетей определяется их аппроксимирующей способностью, причем НС являются универсальными функциональными аппроксиматорами [2]. С помощью НС можно выразить любую непрерывную функциональную зависимость на основе обучения НС, без предварительной аналитической работы по выявлению правил зависимости выхода от входа. Недостатком нейросетей является невозможность объяснить выходной результат, так как значения распределены по нейронам в виде значений коэффициентов весов. Основной трудностью в применении нечетких экспертных систем служит необходимость явно сформулировать правила проблемной области в форме продукции. В нечетких экспертных системах легко построить объяснение результата в форме протокола рассуждений. Поэтому в настоящее время создаются гибридные технологии,

сочетающие преимущества нечетких систем и нейронных сетей. Примером гибридной технологии служит реализация системы нечетких правил на основе нейросети. База нечетких правил для двух входных и одной выходной переменных имеет следующую структуру:

$R_i$  : **если**  $(x_{1i} - A_{1i})$  **И**  $(x_{2i} - A_{2i})$  **то**  $(z_i - C_i)$ .

Для реализации базы нечетких правил будем интерпретировать ее как таблицу определения некоторой функции, то есть базу правил можно представить обучающей выборкой:  $((A_{1i}, A_{2i}), C_i)$ . Например,  $((\text{малое}, \text{большое}), \text{около нуля})$ .

### **Искусственный интеллект, нейроинтеллект, естественный интеллект**

Естественный интеллект сформировался в ходе многомиллиардной эволюции. Искусственный интеллект создается в научном сообществе, довольно несоординированном, последние 40 лет. Если учесть, что человечество не знает ответа на вопрос, как работает человеческий мозг, то силы явно не равны. Не существует общепризнанного определения естественного интеллекта. Разброс мнений колеблется от «нечто, что измеряется в интеллектуальных тестах», до «способности решать задачи с неизвестным алгоритмом решения». Для настоящего момента развития ИИ характерно внимание к эволюционному процессу развития жизни, который и создал естественный интеллект [3]. Создаются системы, моделирующие эволюцию, но не живых организмов, а машин. Однако такой способ создания искусственного разума потребует геологических эпох, несмотря на быстрое действие компьютеров. От примитивной лабиринтной модели мышления ИИ быстро перешел к моделированию свойств естественного интеллекта, проявляемых в ходе фило- и этногенеза интеллекта. Филогенез естественного интеллекта представляет собой сложное взаимодействие развития движений и пространственных моделей окружающего мира. Телевосприятие и теледействие потребовали развития моторики, а органы движения живого представляли собой пространственную модель мира. Осуществление все более сложных движений привело к развитию систем управления, то есть различных оценочных функций, безусловных и условных рефлексов. Усложнение движений и поведения в целом привело к формированию сложных врожденных программ движения и поведения — инстинктов. Наука в современном состоянии не может ответить на вопрос, когда и как появилась память, но ее появление позволило накапливать и использовать прижизненный опыт. Рано появившаяся асимметрия мозга привела к раз-

делению функций, к формированию оценки (осознания) интеллектуальной деятельности и в конце концов к формированию психики и сознания. Данные онтогенеза объясняют явление интроспекции, которое заключается в том, что при освоении какой-то деятельности ребенок помогает себе движениями, которые все менее активно используются при дальнейшем росте. Например, решая лабиринтную задачу, ребенок водит пальцем, а взрослый человек «следит глазами». Приведенные данные говорят о связи движения, поведения, жизни с появлением интеллекта. Иначе говоря, чтобы машины получили интеллект в ходе эволюции, они должны начать жить.

## Задачи искусственного интеллекта: Возможности ННС

### Научное направление «искусственный интеллект» (ИИ)

В XIII веке средневековый испанский философ, математик, поэт *Раймонд Луллий* пытался создать механическую машину для решения различных задач на основе всеобщей классификации понятий. В XVIII веке *Лейбниц* и *Декарт* независимо предложили универсальные языки классификации всех наук. Эти работы можно считать первыми работами по теории искусственного интеллекта [4]. В 40-е годы XX века *Н. Винер* создал основополагающие работы по кибернетике. Термин «искусственный интеллект» предложен в 1956 году на семинаре в Дартмутском колледже (США). В английском языке существуют два термина: *intelligence* (умение рассуждать разумно) и *intellect* (собственно семантический аналог русского слова «интеллект»). Таким образом, английский термин изначально был лишен того фантастического семантического оттенка, которым обладает русскоязычный термин. Но и в английском языке у нового термина были свои проблемы, в частности общий корень слов *artificial* (искусственный) и *artifice* (искусственный прием, трюк), в результате ряд критиков нового направления переводил наименование «кафедра искусственного интеллекта» на вычислительном факультете как «кафедра удивительных трюков». Первые сторонники и творцы направления и сами не отказывались от слова «трюк», уточняя, что введение термина — это трюк в стиле “Lifemanship”, в англоязычных странах так называют всякую настойчивую инициативу, успех которой полностью зависит от демонстрации абсолютной уверенности ее организаторов.

В истории искусственного интеллекта можно выделить термины-пред-

шественники: кибернетика, информатика, вычислительная техника; термины-попутчики: бионика, психоника, распознавание образов, нейроинформатика и термины-потомки: инженерия знаний, вычислительный интеллект, мягкие вычисления. Почему научные направления, моделирующие интеллектуальную деятельность человека с разных точек зрения, такие как нейроматематика, теория роботов, распознавание образов, бионика, искусственный интеллект входят в информатику как отдельные направления, отличающиеся языком, методами и кадровым потенциалом? Разделение исследователей произошло из-за разных ответов на вопрос: что такое интеллект и как можно построить разумную машину? Часть исследователей отвечали на вопрос таким образом: интеллект — это способность человека к формально-логической обработке символической информации. А разумная машина должна эффективно перерабатывать символы (*Минский, Маккарти, Робинсон, Кольмеро*). Таким образом, важен знак, символ, слово, понятие. Со временем именно данное направление стали называть «искусственный интеллект». Часть исследователей отвечали на вопрос так: интеллект — это способность человеческого мозга к целенаправленной деятельности, сложная система рефлексов, инстинктов, логической обработки информации. Думающая машина может быть только подобна естественному интеллекту (*Розенблатт, МакКаллок, Хебб*). Таким образом, важен сигнал, ощущение, восприятие. Сторонники направления работают в направлениях — нейроинформатика, распознавание образов, бионика. В ходе формирования науки искусственного интеллекта, определения ее содержания, структуры, определялся и специфический предмет исследования. Задача была и остается сложной, так как наука обладает принципиальной междисциплинарностью. Сложность ее истории в значительной степени определяется мировоззренческим потенциалом искусственного интеллекта.

### Состояние ИИ

К настоящему времени предмет искусственного интеллекта остался по-прежнему неопределенным, что не мешает конкретным исследованиям быть полезными. Перечислим только основные достижения ИИ:

- создание систем, основанных на знаниях (knowledge based system), решение задач представления, обработки и извлечения знаний, разработка многочисленных экспертных систем;
- разработка естественно-языковых интерфейсов и развитие машинного перевода. Хотя ни одна из этих задач и не решена окончательно,

но продвижение в указанных направлениях существенное;

- развитие математической логики — от метода резолюций *Робинсона* до развития нетрадиционных логик (немонотонных, правдоподобного вывода и т. д.);

Значительное влияние ИИ оказал на развитие индустрии программирования. В частности, язык ЛИСП был предложен *Маккарти* для решения задач именно ИИ, а сейчас используется более широко. Парадигма объектно-ориентированного программирования сложилась в ходе проведения работ по ИИ. Анализ показывает сближение представителей разных направлений ИИ: традиционная обработка знаний и естественные языки; нейронные сети; распознавание и синтез речи; многоагентные системы. Сближение связано не только с естественными циклами развития любой науки, но и с тенденцией глубинной интеграции. На наш взгляд важная тенденция в развитии ИИ ближайшего будущего — это интеграция автономных достижений, причем глубинная интеграция, осуществляемая на основе создания гибридных систем. Трудно сказать под каким именем будет происходить такая широкая гибридизация, возможна регенерация термина «гибридные системы» или развитие термина «синергетический интеллект». В любом случае впереди гибридизация достижений инженерии знаний, распознавания образов, нейроинформатики, роботики, которая должна обеспечить синергизм сигнала, числа, слова и понятия. В самом общем виде такую тенденцию можно выразить как переход от задачи создания думающей машины к задаче создания интеллектуального существа.

### **Искусственный и вычислительный интеллект**

*К. Циммерман* (Аахен, Германия) предложил в начале 90-х годов термин «вычислительный интеллект». Синонимичный «зонтичный» термин предложил в 1994 году на семинаре в Беркли *Л. Заде* — «мягкие вычисления», включающий нечеткую логику, нейронные вычисления и эволюционные вычисления. Рождение новых направлений, а, значит, и развитие науки продолжается.

### **Традиционная задача ИИ — инженерия знаний и экспертные системы**

Экспертные системы (ЭС) — это прикладные интеллектуальные системы, в которых база знаний представляет собой формализованные эмпирические

знания высококвалифицированных специалистов (экспертов) в какой-либо узкой предметной области.

Экспертная система состоит из следующих компонент:

- *База знаний* предназначена для хранения экспертных знаний о предметной области, используемых при решении задач экспертной системой. База знаний состоит из набора фреймов и правил-продукций.
  - Фреймы используются в базе знаний для описания объектов, событий, ситуаций, прочих понятий и взаимосвязей между ними. Фрейм — это структура данных, состоящая из слотов (полей).
  - Правила используются в базе знаний для описания отношений между объектами, событиями, ситуациями и прочими понятиями. На основе отношений, задаваемых в правилах, выполняется логический вывод. В условиях и заключениях правил присутствуют ссылки на фреймы и их слоты.
- *База данных* предназначена для временного хранения фактов или гипотез, являющихся промежуточными решениями или результатом общения системы с внешней средой, в качестве которой обычно выступает человек, ведущий диалог.
- *Машина логического вывода* — механизм рассуждений, оперирующий знаниями и данными с целью получения новых данных из знаний и других данных, имеющихся в рабочей памяти. Для этого обычно используется программно реализованный механизм дедуктивного логического вывода (какая-либо его разновидность) или механизм поиска решения в сети фреймов или семантической сети. Машина логического вывода может реализовывать рассуждения в виде:
  - дедуктивного вывода (прямого, обратного, смешанного);
  - нечеткого вывода;
  - вероятностного вывода;
  - унификации (подобно тому, как это реализовано в Прологе);
  - поиска решения с разбиением на последовательность подзадач;
  - поиска решения с использованием стратегии разбиения пространства поиска с учетом уровней абстрагирования решения или понятий, с ними связанных;
  - монотонного или немонотонного рассуждения,
  - рассуждений с использованием механизма аргументации;
  - ассоциативного поиска с использованием нейронных сетей;
  - вывода с использованием механизма лингвистической переменной.

- *Подсистема общения* служит для ведения диалога с пользователем, в ходе которого ЭС запрашивает у пользователя необходимые факты для процесса рассуждения, а также, дающая возможность пользователю в какой-то степени контролировать и корректировать ход рассуждений экспертной системы.
- *Подсистема объяснений* необходима для того, чтобы дать возможность пользователю контролировать ход рассуждений и, может быть, учиться у экспертной системы.
- *Подсистема приобретения знаний* служит для корректировки и пополнения базы знаний. В простейшем случае это — интеллектуальный редактор базы знаний, в более сложных экспертных системах — средства для извлечения знаний из баз данных, неструктурированного текста, графической информации и т. д.

Нечеткая ЭС — экспертная система, которая для вывода решения использует совокупность нечетких функций принадлежности и правил. Функция принадлежности выражает степень соответствия элемента некоторому понятию. Реальным источником ее формирования являются расспросы эксперта и обобщение статистических данных.

#### Мягкие экспертные системы

Рассмотрим архитектуру и основные структурно-функциональные решения мягкой экспертной системы (МЭС). Для определения МЭС, сопоставим понятия нечеткой и мягкой экспертных систем. Для описания архитектуры МЭС будем использовать три признака:

- способ извлечения знаний;
- представление знаний;
- обработка знаний.

Перечисленные признаки создают общую «координатную сетку» описания.

**Сравнение нечеткой и мягкой экспертной систем.** Нечеткие экспертные системы используют представление знаний в форме нечетких продукций и лингвистических переменных. Основу представления лингвистической переменной составляет терм с функцией принадлежности. Способ

обработки знаний в нечетких ЭС — это логический вывод по нечетким продукциям. Особенностью нечеткой ЭС является способ извлечения функций принадлежности, который сводится либо к статистическим методам построения, либо к методу экспертных оценок. Мягкой ЭС будем называть нечеткую ЭС, которая обладает следующими особенностями:

- Мягкая ЭС для извлечения знаний использует статистические данные, которые интерпретирует как обучающие выборки для нечетких нейронных сетей.
- Мягкая ЭС представляет знания в виде лингвистических переменных (функций принадлежности), нечетких продукций и обученных нейронных сетей. Редукция множества нечетких продукций, настройка ФП и базы правил выполняется с помощью ГА.

Мягкими называют вычисления, сочетающие теорию нечетких систем, нейронные сети, вероятностные рассуждения и генетические алгоритмы и обладающие синергическим эффектом, следовательно, мягкой ЭС называют ЭС, сочетающую перечисленные теории ради того же эффекта взаимного усиления.

**Инженерия знаний и мягкие экспертные системы.** Мягкая ЭС извлекает знания из статистических выборок, интерпретируя их как обучающие выборки для обучения нейронной сети. В дальнейшем особенности мягкого извлечения знаний продемонстрируем на двух примерах:

- извлечение нечетких продукций для ЭС анализа тенденций развития предприятия;
- извлечение нечетких продукций для задачи конструирования стенов изделий.

**Представление знаний в мягкой экспертной системе. Содержание баз знаний и данных мягкой экспертной системы.** Если использовать ННС на этапе извлечения знаний, то кроме функций принадлежности и нечетких продукций, порождается совокупность обученных НС, которые входят в базу знаний МЭС. Оптимизация (редукция) множества извлеченных правил выполняется на основе генетического алгоритма.

База знаний МЭС должна содержать следующие части:

- функции принадлежности;
- нечеткие продукции;

- обученные нечеткие нейронные сети;
- процедуры интерпретации хромосом генетических алгоритмов;
- функции оптимальности.

Рассмотрим проблему представления перечисленных составных частей в компьютерных интеллектуальных системах. Если функция принадлежности характеризуется такими математическими свойствами, как непрерывность, выпуклость (униmodalность), то функция принадлежности может быть представлена параметризованной функцией формы. Наибольшее распространение получили три вида функций формы: треугольная, трапециевидная и колоколообразная (сигмоидальная); которые определяются тройкой, четверкой и двойкой параметров соответственно.

### **Пример мягкой ЭС для проектирования радиоэлектронных изделий**

**Описание задачи конструирования РЭА.** Для производственной деятельности конструкторских бюро характерно изготовление опытных образцов, малых серий изделий, модифицированных блоков серийных изделий. Задачей МЭС служит разработка стендов для испытаний или контроля и диагностики блоков. Большие конструкторские бюро, имеющие длительную историю создания таких стендов обладают значительными архивами разнообразных стендов. Архивы проектов в настоящее время представляют собой хранилища бумажной технической документации. Информация, хранимая таким образом, редко используется повторно и достаточно оперативно. Другим результатом продолжительной производственной деятельности конструкторских бюро является наличие опытных специалистов-конструкторов. Такие специалисты, эксперты, покидая предприятие часто уносят уникальный опыт проектирования. Поэтому в качестве решения проблемы сохранения накопленного опыта проектирования необходимо комплексное решение. Недостаточно использовать даже самую современную распределенную информационную систему доступа к архиву проектов, так как методика конструирования не отражается в конечном продукте — технической документации на стенд. Информационная система должна включать в себя экспертную систему, содержащую знания экспертов, которые в настоящее время никак не документируются на предприятиях. Для создания в качестве интеллектуальной компоненты именно экспертной системы имеются серьезные предпосылки: наличие экспертов; наличие архива проектов из которого можно извлекать знания, возможность участво-

Таблица 1. Средства решения задач конструирования изделий радио-электронной аппаратуры (РЭА)

| Задача               | Средство решения         |
|----------------------|--------------------------|
| Выбор корпуса        | Система нечеткого вывода |
| Выбор типа корпуса   | Система принятия решений |
| Размещение элементов | Генетический алгоритм    |

вать в конструировании какого-либо стенда, разрабатываемого в проектной организации. Для конструирования стендов выберем средства решения, результаты выбора приведены в табл. 1.

**Нейросетевая реализация МЭС конструирования стендов контроля РЭА.** Учитывая значительный объем реальных архивов проектных организаций, особенно привлекательным представляется извлечение знаний из данных для организации процесса извлечения знаний необходимо выделить факторы влияния в качестве базовых переменных и составить таблицы значений этих переменных для проектов, хранимых в архиве. Выходная переменная — это переменная, обозначающая тип и размер корпуса. В качестве базовых переменных были выделены следующие:

- наличие элементов на передней панели с рабочим положением «Вертикальное»;
- наличие индикаторов на передней панели с небольшим углом обзора;
- количество регулируемых элементов;
- количество ячеек;
- суммарная площадь ячеек;
- количество элементов, устанавливаемых внутри блока;
- суммарный объем элементов, устанавливаемых внутри блока;
- суммарная установочная площадь внутри блока;
- наличие элементов внутри блока с креплением «шасси»;
- количество тумблеров на передней панели;
- количество элементов со шкалами;
- количество стрелочных индикаторов;
- количество галетных переключателей;
- суммарная установочная площадь на передней панели.

Для извлечения правил необходима нейронная нечеткая сеть, вербализующая правила. Таким инструментом может служить ННС с редукцией полученных правил генетическим алгоритмом FUNGEN 1.1. Обучающая выборка должна представлять собой множество строк, каждая строка содержит сведения об одном стенде. Структура строки содержит две части: входные переменные, определяющие выбор корпуса или передней панели и сделанный выбор — тип и габариты корпуса или размеры передней панели. В результате обучения ННС не только отбираются правила вывода, но и настраиваются параметры функций принадлежности. Для извлечения правил необходима нейронная нечеткая сеть, вербализующая правила. Таким инструментом может служить ННС с редукцией полученных правил генетическим алгоритмом FUNGEN 1.1. Обучающая выборка должна представлять собой множество строк, каждая строка содержит сведения об одном стенде. Структура строки содержит две части: входные переменные, определяющие выбор корпуса или передней панели и сделанный выбор — тип и габариты корпуса или размеры передней панели. В результате обучения ННС не только отбираются правила вывода, но и настраиваются параметры функций принадлежности. Уточненные функции принадлежности являются результатами эксперимента. Для каждой входной и выходной переменных разработчик МЭС составляет первоначальные функции принадлежности нечетких термов. Правила, полученные в результате обучения ННС выражают аналогию между стендами, т.е. для стендов с похожими структурно-функциональными элементами рекомендуются одинаковые корпуса, а ННС оценивает их близость. Исходные лингвистические переменные и их нечеткие метки определены на основе экспертных данных и анализа истории проектирования стендов изделий. На основе выделенных с помощью классификаций базовых переменных и задания их семантики с помощью функций принадлежности можно сформулировать правило вывода проектных решений. Правила конструирования стендов определяют степень возможности выбора типа и типоразмера блока с точки зрения элементной базы и составляют основную часть экспертной системы. Левые части правил определяют нечеткие значения значимых лингвистических переменных, а правые определяют степень возможности выбора каждого из допустимых типов корпуса.

## Задачи прикладного ИИ: Возможности ННС

### Интеллектуальный анализ данных

В настоящее время накоплены огромные объемы информации, описывающие различные сложные системы. Анализ этих данных, содержащих свойства объектов, позволяет описать реальность и построить модель для дальнейшего изучения и прогнозирования поведения системы. Результатом анализа является нахождение взаимосвязей, динамики развития свойств объектов, т.е. знание о предметной области. Знание может выражаться функциональными зависимостями между свойствами, логическими связями. Исследования данных и их методов анализа оформились в виде отдельного направления называемого Data Mining (DM). Современные комплексы DM позволяют «добывать» знания о системах в виде простых или сложных функций, множества продукции {если... то...} и представляют собой инструменты интеллектуального анализа данных. Каждый метод DM имеет свои достоинства и недостатки, что обусловлено характером данных и их взаимосвязей. Один из видов извлекаемых знаний — это знания о динамике развития объекта. Существует статистическое решение задачи выделения тренда — постоянной составляющей временного ряда. Аналитическое выражение тренда можно найти методами регрессионного анализа, сезонной декомпозиции, анализа Фурье, различными методами сглаживания. К их общим недостаткам следует отнести необходимость задания явной параметрической модели тренда и требования стационарности остатка. Подход с точки зрения вычислительного интеллекта (мягких вычислений) позволяет использовать для распознавания тенденций аппарат нейронных сетей. Социологические, экономические, экологические базы данных (БД) содержат лингвистические оценки и являются нечеткими реляционными БД. Поэтому извлечение знаний (Data Mining) предполагает решение двух задач:

- кластеризации;
- поиска зависимостей.

### ННС как Data Miner для нечетких реляционных баз данных

**Нечеткая реляционная модель данных. Способ представления данных.** Нечетким отношением будем называть конечное множество отображений, содержащих нечеткие данные. Нечеткие данные могут присутство-

вать только на определенных атрибутах — нечетких атрибутах. Нечеткий атрибут определим как атрибут, на домене которого можно определить нечеткое множество. Модель рассчитана на представление нечетких чисел, соответственно доменом атрибута нечеткого числа является множество действительных чисел. Нечеткое число может быть задано в виде функции принадлежности (лингвистической оценки). Под лингвистической оценкой будем понимать одно из возможных значений лингвистической переменной, которое определяется соответствующим термом. Функция принадлежности, задающая нечеткое число удовлетворяет следующим свойствам:

- ограниченности (по определению  $\max(\mu(x)) \leq 1, \min(\mu(x)) \geq 0$ );
- однозначности (каждое множество  $\mu(x) = Z, x \in X$ , состоит только из одного элемента);
- непрерывности (в каждой точке существует предел функции).

Если для некоторого аргумента значение функции принадлежности неопределенно, то предполагается, что это значение равно нулю. Для представления точных (четких) значений используется вырожденный вид функции принадлежности, возвращающей 1 для представляемого точного значения и 0 для всех остальных значений.

В модели принят табличный способ задания функция принадлежности. Количество пар  $\mu(x_i)/x_i$ , задающих функцию принадлежности, не ограничено. Предполагается, что множество таких пар задает ломаную линию, которая и является графиком функции принадлежности [6].

**Кластеризация реляционных кортежей.** Задача кластеризации заключается в разбиении объектов на кластеры, когда основой разбиения служит вектор параметров объекта. Объекты в пределах одного класса считаются эквивалентными с точки зрения критерия разбиения. Сами кластеры часто бывают неизвестны заранее, а формируются динамически. Кластеры зависят от предъявляемых объектов, и поэтому добавление нового объекта требует корректирования системы кластеров.

Будем характеризовать объекты, подлежащие кластеризации, вектором  $x^p \in X$ , имеющим  $N$  компонент, компоненты обозначим нижним индексом:  $x^p = (x_1^p, \dots, x_N^p)^T$ . Вектор параметров — единственная характеристика объектов при их кластеризации. Введем множество кластеров  $C^1, \dots, C^M = C^m$  в пространстве кластеров  $C$ :

$$(C^1 \cup C^2 \dots \cup C^M) \subset C.$$

Пространство кластеров может не совпадать с пространством объектов  $X$  и иметь другую размерность. В простейшем случае, когда пространства кластеров и объектов совпадают, кластеры представляют собой области пространства  $X$ , и объект  $x^p$  будет отнесен к одному из классов  $m_0$ , если  $x^p \in C$ . В общем случае  $X$  и  $C$  различны.

Определим ядра кластеров  $C = c^1, \dots, c^m$  в пространстве классов  $C$ , как объекты, типичские для своего класса. Очевидно, что близость объекта к ядру необходимо оценивать численно. Введем меру близости  $d(x^p, c^m)$  — скалярную функцию от объекта и ядра кластера, которая тем меньше, чем больше объект похож на ядро кластера. Могут потребоваться вспомогательные меры близости, определенные для двух объектов,  $d(x^p, c^m)$ , и для двух ядер классов,  $d(c^{m_1}, c^{m_2})$ . Чаще всего применяется евклидова мера:

$$d(x, y) = \sum_i (x_i - y_i)^2$$

или

$$d(x, y) = \sum_i \|(x_i - y_i)\|.$$

Задавшись числом кластеров, можно поставить задачу кластеризации: найти  $M$  ядер кластеров  $C^m$  и разбить объекты  $X^M$  на кластеры  $C^m$ , то есть построить функцию  $m(p)$  таким образом, чтобы минимизировать сумму мер близости:

$$m(D) = \sum_p (d(x^p, c^{m(p)})).$$

Функция  $m(p)$ , определяющая номер кластера по индексу  $p$  множества объектов  $X^p$ , задает разбиение на кластеры и является решением задачи кластеризации. В простейшем случае  $X = C$ , пространство объектов  $X$  разбивается на области  $C^m$  и если  $x_0^p \in C_0^m$ , то  $(p_0) = m_0$ , тогда объект относят к кластеру  $m_0$ . Количество кластеров может динамически меняться. При этом часто возникают ситуации, когда объекты распределены по кластерам неравномерно. Необходимо контролировать равномерность плотности ядер  $c^m$  в пространстве  $C$  и долю объектов, относящихся к каждому кластеру  $m_0$ . Рассмотрим общий алгоритм с фиксированным количеством ядер  $M$ . Количество ядер выбирается заранее, исходя из конкретной задачи. Начальные значения ядер могут выбираться случайными  $c^1, \dots, c^m$ , одинаковыми или по другим эвристическим правилам. Каждая итерация алгоритма состоит из двух этапов:

- При неизменных ядрах  $C^m = \text{const}$  ищем такое разбиение  $m(p)$  объектов  $X^p$  на классы, чтобы минимизировать суммарную меру близости между объектами и ядрами их кластеров:

$$\min \left[ \sum_p (d(x^p, c^{m(p)})) \right].$$

Результат этапа — создание функции  $m(p)$ , разбивающей объекты на кластеры.

- При измененном разбиении  $m(p)$  настраиваем ядра  $C^m$  так, чтобы в пределах каждого кластера  $m_0$  суммарная мера близости ядра этого кластера и объектов, ему принадлежащих, была минимальной:  $\sum_p d[x^p, c^{m(p)}]$  для всех  $m_0 = 1, \dots, M$ .

Результат этого этапа — новый набор ядер  $C^m$ .

### Решение задачи нечеткой кластеризации на основе сети Кохонена.

Выберем в качестве входных данных вектор параметров единственного объекта. Результатом работы сети будет код кластера, к которому принадлежит предъявленный на входе объект. Сеть будет иметь  $M$  выходов, по числу кластеров, и чем большее значение принимает выход номер  $m_0$ , тем больше «уверенность» сети в том, что входной объект принадлежит к кластеру  $m_0$ . Если применить функцию активации SOFTMAX, тогда сумма выходов всегда будет равна единице. Каждый выход можно будет трактовать как степень уверенности того, что объект принадлежит данному кластеру. Выберем евклидову меру близости. В этом случае ядро кластера, минимизирующее сумму мер близости для объектов этого кластера, совпадает с центром тяжести объектов:

$$c^{m_0} = \frac{1}{N(m_0)} \sum_{p:m(p)=m_0} x^p,$$

где  $N(m_0)$  — число объектов  $x^p$  в кластере  $m_0$ . При разбиении на кластеры должна быть минимизирована суммарная мера близости для всего множества  $X^p$  входных объектов:

$$D = \sum_p \sum_i [x_i^p - c_i^{m(p)}]^2 = \sum_p [(x^p, x^p) - 2(x^p, c^{m(p)}) + (c^{m(p)}, c^{m(p)})]$$

расписано скалярное произведение. В этой сумме два слагаемых не зависят от способа разбиения и постоянны:

$$\sum_p (c^{m(p)}, c^{m(p)}) = \text{const},$$

$$\sum_p (x^p, x^p) = \text{const}.$$

Поэтому задача поиска минимума  $D$  эквивалентна поиску максимума выражения:

$$\min D \rightarrow \max \sum_p \sum_i x_i^p c_i^{m(p)}.$$

Вариант алгоритма кластеризации для поиска максимума этой функции:

---

**Алгоритм 1:** Алгоритм кластеризации

---

```

for all Для каждого вектора  $x^p$  do
  for all Для каждого  $m$  do
    Рассчитать  $\sum_i x_i^p c_i^m = D^{m,p}$ 
  end for
  Найти  $m_0$ , для которого  $m_0 : \max(D^{m,p})$ 
  Отнести объект к кластеру  $m_0$ 
end for

```

---

Данный алгоритм легко реализуется в виде нейронной сети. Для этого требуется  $M$  сумматоров, находящих все  $D^{m,p}$ , а также интерпретатор, находящий сумматор с максимальным выходом. Входные векторы сети чаще всего нормируются:

$$\frac{x^p}{\|x^p\|}; \quad \frac{x^p}{\sum_p \|x^p\|^2} \rightarrow x^p; \quad \frac{x}{\sqrt{\sum_p \|x^p\|^2}} \rightarrow x^p.$$

Возможны и другие способы нормировки.

**Обучение сети.** Задача обучения — научить сеть активировать один и тот же нейрон для похожих векторов  $x^p$  на входе. Не важно, какой конкретно нейрон будет активирован. Обычно начальные значения в нейронных

сетях выбираются малыми случайными числами. В случае, если ядра кластеров нормированы, то и начальные значения необходимо нормировать. Если веса инициализируются случайными значениями с равномерным распределением, то возникает проблема. Когда ядра распределяются равномерно, то в областях пространства, где мало входных векторов, ядра будут использоваться редко, так как мало будет похожих векторов. В тех областях, где входных векторов много, плотность ядер окажется недостаточной, и непохожие объекты будут активировать один и тот же нейрон, так как более похожего ядра не найдется. Для устранения проблемы можно выделять ядра в соответствии с плотностью входных векторов. Но распределение входных векторов часто бывает заранее неизвестно. В этом случае возможно использование метода выпуклой комбинации. Если число входных векторов равно числу ядер, то обучение не нужно. Достаточно присвоить ядрам значения входных векторов, и каждый вектор будет активировать свой нейрон Кохонена. Но чаще всего количество кластеров меньше числа входных векторов. В этом случае веса сети настраиваются итеративным алгоритмом.

Алгоритм аналогичен исходному алгоритму классификации, но коррекции весов проводятся после предъявления каждого входного вектора, а не после предъявления всех, как требует исходный алгоритм. Сходимость при этом сохраняется.

1. Присваиваем начальные значения весовым коэффициентам.
2. Подаем на вход один из векторов  $x^p$ .
3. Рассчитываем выход слоя Кохонена,  $D^{m,p}$ , после чего определяем номер выигравшего нейрона  $m_0$ , выход которого максимален,  $m_0 : \max D^{m,p}$ .
4. Корректируем веса только выигравшего нейрона  $m_0$ :

$$\omega_m := \omega_m + \alpha(x^p - \omega_m);$$

коррекция записана в виде векторного выражения (вектор весов  $\omega_{m_0}$  нейрона  $m_0$  имеет столько компонент, сколько их у входного вектора  $x^p$ ). Здесь  $\alpha$  — скорость обучения, малая положительная величина. Часто используют расписание с обучением, когда  $\alpha = \alpha(t)$  монотонно убывает.

Веса корректируются так, что вектор весов приближается к текущему входному вектору. Скорость обучения управляет быстротой приближения ядра

кластера (вектора весов) ко входному вектору  $x^p$ . Алгоритм выполняется до тех пор, пока веса не перестанут меняться.

**Режимы работы сети.** Обычная сеть Кохонена работает в режиме аккредитации. Это означает, что активируется единственный нейрон Кохонена с максимальным значением выхода. Можно не затормаживать остальные нейроны слоя Кохонена, а пронормировать выходные сигналы, например, функцией SOFTMAX:

$$OUT_j = \frac{e^{NET_j}}{\sum_i e^{NET_i}}.$$

Тогда сумма всех выходов слоя будет равна единице и можно трактовать выходы как степень принадлежности объекта к каждому из классов. Такой режим работы сети, когда активируется несколько нейронов одновременно, называется режимом интерполяции. Название режима объясняется тем, что если входной вектор  $x^p = \omega^{m_1}$  плавно меняется от одного вектора весов  $x^p = \omega^{m_1}$  к другому вектору весов  $x^p = \omega^{m_2}$ , то выход сети в режиме интерполяции (если применена функция SOFTMAX) будет плавно меняться от  $m_1$  к  $m_2$ , то есть классификация оказывается непрерывной. Если же сеть работает в режиме аккредитации, выход изменяется от  $m_1$  к  $m_2$  скачкообразно.

#### Извлечение правил из реляционных кортежей с помощью ННС

**Прогнозирование нечетких временных рядов.** Подход с точки зрения мягких вычислений позволит использовать для распознавания тенденций аппарат нейронных сетей. Исходные данные могут быть представлены в нечетком выражении, что может быть обусловлено экспертным оцениванием состояния системы, а так же несовершенными способами измерения параметров. Для описания изменений этих нечетких величин вводится понятие нечеткого ряда: упорядоченная последовательность нечетких меток, другими словами, нечеткая зависимость от аргумента. Наиболее часто изменения параметра рассматриваются во времени, в этом случае говорится о нечетком временном ряде, где время есть свойство системы. Тенденция изменений так же выражается нечеткостью, являясь «нечеткостью второго порядка». Носителем нечеткого множества тенденций является множество всевозможных функций.

В простейшем случае тенденция может описываться лингвистическими терминами «рост», «стабилизация», «падение», оценивающие знак производной функции. Но под нечеткой тенденцией можно понимать образы любых функции, не только «рост», «падение», «стабилизация», но и экспертные оценки тенденций: «колебания», «хаос». Таким образом, текущие состояние системы можно описать нечеткими тенденциями ее параметров.

**Определение нечеткого временного ряда** Нечетким временным рядом (НВР) будем называть упорядоченную последовательность наблюдений над некоторым явлением, характер которого изменяется во времени, если значения, которые принимает некоторая величина в момент времени, выражена с помощью нечеткой метки. Нечеткая метка представлена лингвистическим термом и функцией принадлежности.

**Определение нечеткой тенденции.** Тенденцией нечеткого временного ряда (НТ) будем называть нечеткую метку, выражающую динамику (систематическое движение) НВР. К значимым наименованиям тенденций относятся следующие нечеткие метки:

- рост;
- падение;
- стабилизация;
- колебания;
- хаос.

Разумеется, для нечетких термов, обозначающих тенденцию, возможно применение модификаторов «очень», «более-менее» и т. д. Подход с точки зрения мягких вычислений позволяет использовать для распознавания тенденций аппарат нейронных сетей. Временные ряды представляются своими графиками. Распознавание тенденции каждого графика можно считать задачей распознавания изображений. В результате подход мягких вычислений позволяет сформулировать следующее утверждение.

Результативным средством распознавания тенденций НВР является нейронная сеть. Кроме распознавания тенденций НВР рядов интерес представляет возможность извлечения правил распознавания тенденций методом нейрокомпьютинга. Нечеткое понятие «тенденция» имеет лингвистические термины «рост-развитие», «снижение-деградация», «стабилизация», «хаос». Например, для оценки темпов роста оборота предприятия используется

следующие базовые переменные: «Оборот предприятия — растет, падает, стабилизация, хаос; темпы роста — высокие, средние, низкие».

Анализ тенденций — это удобная задача для извлечения продукций, так как можно разработать генераторы тестов различных тенденций. При разработке таких генераторов необходимо учитывать диапазоны значений конкретной проблемной области, типичные значения таких переменных.

Сравним эффективность нечеткой нейронной сети архитектуры FUN с классическими НС при решении данной задачи. Для успешного обучения важна методика обучения тенденциям, которая состоит в том, что более качественные результаты распознавания дает нейронная сеть, обученная отличать один вид тенденции от всех других. Результативной является методика обучения отдельной ННС одной тенденции. Сначала — тенденции «рост», затем — «падение», «стабилизация» и другим. На заключительном этапе методика требует синтезировать нейро-нечеткую ЭС из обученных отдельных ННС, рассматриваемых как элементы. Архитектура FUN удобна для извлечения знаний, так как она позволяет выполнить настройку функций принадлежности, представленных параметризованным семейством сигмоидальных функций.

Синергические средства, такие как нечеткие нейросети с генетической редукцией правил позволяют сформулировать систему продукций для анализа тенденций на основе обучения. В качестве нейроимитатора можно выбрать нейроимитатор FUNEGEN 1.1, свободно доступный для исследователей в Internet. FUNEGEN 1.1 позволяет сформулировать систему продукций для анализа тенденций на основе обучения. Интерес представляет сравнение качества распознавания тенденций системой продукций, сформулированной экспертом, с системой продукций, сгенерированной нечетким нейроимитатором. Краткое описание нейроимитатора позволяет сделать вывод о наличии эффекта синергизма при использовании этого инструмента.

Ниже приведены сгенерированные правила распознавания.

|        |                                |                    |
|--------|--------------------------------|--------------------|
| 3.892  | <b>IF</b> (i3 mf0) <b>THEN</b> | output0            |
| 2.555  | <b>IF</b> (i3 mf2) <b>THEN</b> | output1            |
| 1.985  | <b>IF</b> (i0 mf3)             |                    |
|        | <b>OR</b> (i1 mf2) <b>THEN</b> | output2            |
| –4.889 | <b>IF</b> (i2 mf3) <b>THEN</b> | <i>NOT</i> output0 |
| –4.565 | <b>IF</b> (i3 mf0) <b>THEN</b> | <i>NOT</i> output1 |
| –1.530 | <b>IF</b> (i2 mf0) <b>THEN</b> | <i>NOT</i> output2 |
| 1.316  | <b>IF</b> (i0 mf1)             |                    |
|        | <b>OR</b> (i3 mf2) <b>THEN</b> | output1            |
| –0.909 | <b>IF</b> (i3 mf2) <b>THEN</b> | <i>NOT</i> output2 |
| 0.532  | <b>IF</b> (i0 mf0) <b>THEN</b> | output0            |
| –0.511 | <b>IF</b> (i0 mf1) <b>THEN</b> | <i>NOT</i> output2 |

Многослойная НС (11 входов и 3 выхода) представляет собой трехслойную сеть с семью нейронами в скрытом слое, 110 входами и тремя выходами и сигмоидальной функцией активации. Потребовалось 2830 итераций для уменьшения ошибки НС до уровня  $10^{-4}$ , что обеспечивает стабильное распознавание образов на уровне 94%. Эксперимент, проводимый на сети Кохонена, проводился на той же самой обучающей выборке, что и эксперимент, проводимый на многослойной сети. Сеть Кохонена представляла собой двухслойную сеть со 110 входами и тремя выходами. В качестве функции активации нейронов использовалась униполярная сигмоида. Потребовалось 10000 итераций для стабилизации выходов с точностью в  $10^{-6}$  на всех образцах, что обеспечивает стабильное распознавание тенденций на уровне 89%.

Результаты сравнения результативности многослойной нейронной сети и сети Кохонена для решения задачи распознавания тенденций приведены в табл. 2.

Переменные означают:  $X_1$  — количество распознаваемых тенденций;  $X_2$  — количество итераций для многослойной НС;  $X_3$  — количество итераций для сети Кохонена;  $X_4$  — процент корректного распознавания для многослойной НС;  $X_5$  — процент корректного распознавания для сети Кохонена. Из табл. 2 видно, что сеть Кохонена обучается быстрее только при двух выходах, а при большем количестве выходов уступает в скорости обучения многослойной нейронной сети. По качеству обучения многослойная нейронная сеть предпочтительнее. Сеть Кохонена не только обучается указанным тенденциям, но выделяет классы тенденций, так как работает в режиме классификатора.

ТАБЛИЦА 2. Количество итераций обучения и процент корректного распознавания

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ |
|-------|-------|-------|-------|-------|
| 2     | 2500  | 3300  | 99    | 99    |
| 3     | 3300  | 5000  | 94    | 89    |
| 4     | 9000  | 10000 | 94    | 88    |
| 5     | 15000 | 17500 | 94    | 87    |
| 6     | 20000 | 30000 | 93    | 86    |
| 7     | 23300 | 32000 | 92    | 86    |
| 8     | 25000 | 35000 | 92    | 85    |
| 9     | 30000 | 40000 | 91    | 85    |
| 10    | 35000 | 50000 | 89    | 85    |

Для этапа интегральной оценки состояния объекта можно использовать нейросеть многослойной архитектуры. Система таких нейросетей, которые называют макронейроном, способна выполнить прогнозирование в широком смысле. Ранее рассматривались аспекты анализа отдельных базовых переменных объекта, но объект экспертизы является многомерным и, разобравшись с оценками и тенденциями отдельных величин, необходимо разобраться с состоянием объекта в целом. Задача распадается на две подзадачи: выделения множества базовых переменных, зависимых переменных и зависимостей между ними, определяющими правила поведения объектов; генерации интегральной оценки состояния объекта. Первая из подзадач решается традиционными методами регрессионного анализа, вторая представляется задачей, для которой результативно использование мягких вычислений (МВ).

Прогнозирование будущего функционирования системы основано на тенденциях базовых показателей и методах аппроксимации. Для прогнозирования значений нечетких ВР можно использовать НС в режиме предиктора. НС обучается распознавать тенденции с приемлемым качеством для одного или нескольких ВР. Для получения интегральной оценки необходима совокупность НС, обученных поведению различных базовых переменных. Экспертную систему, сочетающую традиционный механизм продукционного вывода с использованием обученных НС будем называть нейроэкспертной системой. Задача прогнозирования в широком смысле, как

прогнозирование интегрального состояния объекта проектирования может быть выполнена нейроэкспертной системой.

**Распознавание нечеткой тенденции (НТ) с помощью ННС.** Нахождение НТ — основополагающая задача в построении системы анализа данных, решение которой позволит приступить к выявлению связей между параметрами исследуемого объекта. НТ должна обнаруживаться независимо от выбранного периода, что позволит для каждого вида тенденции разбивать исходные данные на периоды наблюдения этой НТ. Например, для текущей точки (момента времени) определяются следующие тенденции: период  $\delta t_3$  — падение параметра,  $\delta t_2$  — периодические колебания, а  $\delta t_1$  — рост (рис. 5).

Исходные данные преобразуются в множество, которое можно представить в виде плоскости в трехмерном пространстве  $(t - x - w)$ , где  $t$  — время,  $x$  — исследуемый параметр,  $w$  — степень уверенности (рис. 5).

Срез по оси  $t$  есть нечеткая метка со своей функцией принадлежности. В качестве инструмента определения НТ может выступать многослойная нейронная сеть, входом которой является множество нечетких меток, а выходом значения тенденций (0 или 1). Количество выходов может варьироваться в зависимости от выбранного варианта: одна нейронная сеть (НС) — один тип НТ, одна НС — на все типы. Задача эксперта не только описать тенденцию, но также и обучить НС. Для обучения НС пользователем подготавливаются выборка данных, представляющая собой ряды нечетких меток и соответствующие им тенденции. Каждая нечеткая метка выражается вероятностью лингвистических термов: «низкое», «среднее», «высокое», т. е. тройкой чисел от 0 до 1. Например, тенденцию «период» можно представить следующим рядом чисел: «высокое» (0, 0, 1), «среднее» (0, 1, 0), «низкое» (1, 0, 0), «среднее» (0, 1, 0), «высокое» (0, 0, 1), «среднее» (0, 1, 0) и т. д. Следует отметить такую характеристику тенденции, как период ее протекания, выражающий глубину просмотра данных для текущей точки. Рассмотрим результат анализа нечетких тенденций временного ряда. Временной ряд представляет выборку недельных доходов в течение четырех лет. Для анализа определены нечеткие тенденции: рост доходов, падение доходов, стабильность, периодические колебания. Обучение нейронных сетей проводилось на основе следующих описаний эталонных тенденций: лингвистические термы нечетких меток — *низкий*, *средний*, *высокий*; период обучающей тенденции равен 7. В качестве размерности матрицы оце-

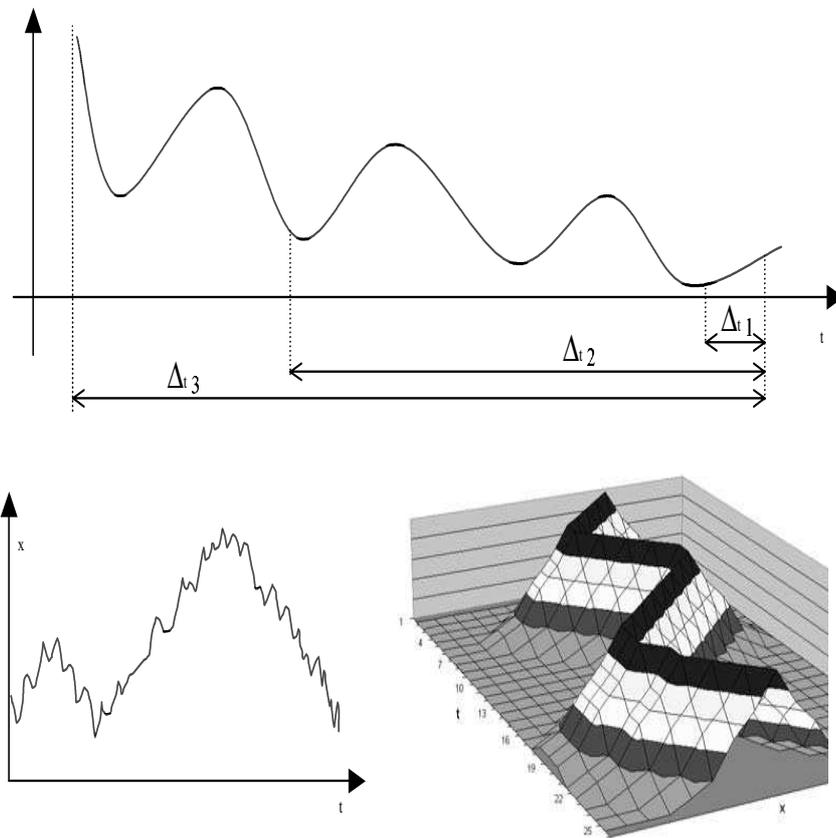


Рис. 5. Исходные данные и срез по оси  $t$

нок выбраны следующие значения: временной ряд на моментах от 7 до 217, интервалы нечетких тенденций от 7 до 60, что соответствует поиску тенденций на интервалах от 1.5 месяца до 1.5 года. Рассмотрим результат анализа полученной матрицы оценок тенденции роста.

На рис. 6 представлены исходные данные и матрица оценки нечеткой тенденции роста доходов. Выявлены основные группы динамики роста: долгосрочный рост в интервале 9 месяцев – 1.5 года, и краткосрочный рост доходов – 2 месяца.

## **Нейросетевое решение задачи прикладного ИИ – развитие информационных хранилищ**

### **Определение информационного хранилища (ИХ)**

Быстрый рост объемов информации требует особых средств для ее хранения и обработки и способствует развитию специального класса программных комплексов – информационных хранилищ (репозиторий). Единицей обработки и хранения в таких комплексах является информационный ресурс. Информационный ресурс – это файл или совокупность файлов, объединенных общим смыслом и имеющих текстовую аннотацию. В частном случае, информационный ресурс – это один или несколько текстовых файлов. Текст аннотации (или текст самого ресурса) однозначно отражает смысловое содержание данного ресурса.

### **Основные функции системы управления ИХ**

Основными функциями информационного хранилища являются:

- хранение информационных ресурсов;
- обеспечение доступа к ресурсам с рабочих мест;
- обеспечение возможности публикации новых ресурсов;
- возможность поиска интересующих ресурсов по определенным критериям.

Самым простым примером может являться использование корпоративного файл-сервера в виде хранилища различной электронной документации. В этом случае, хранение ресурса осуществляется на файловой системе сервера. Доступ с рабочих мест организован через доступ по ftp-, http-

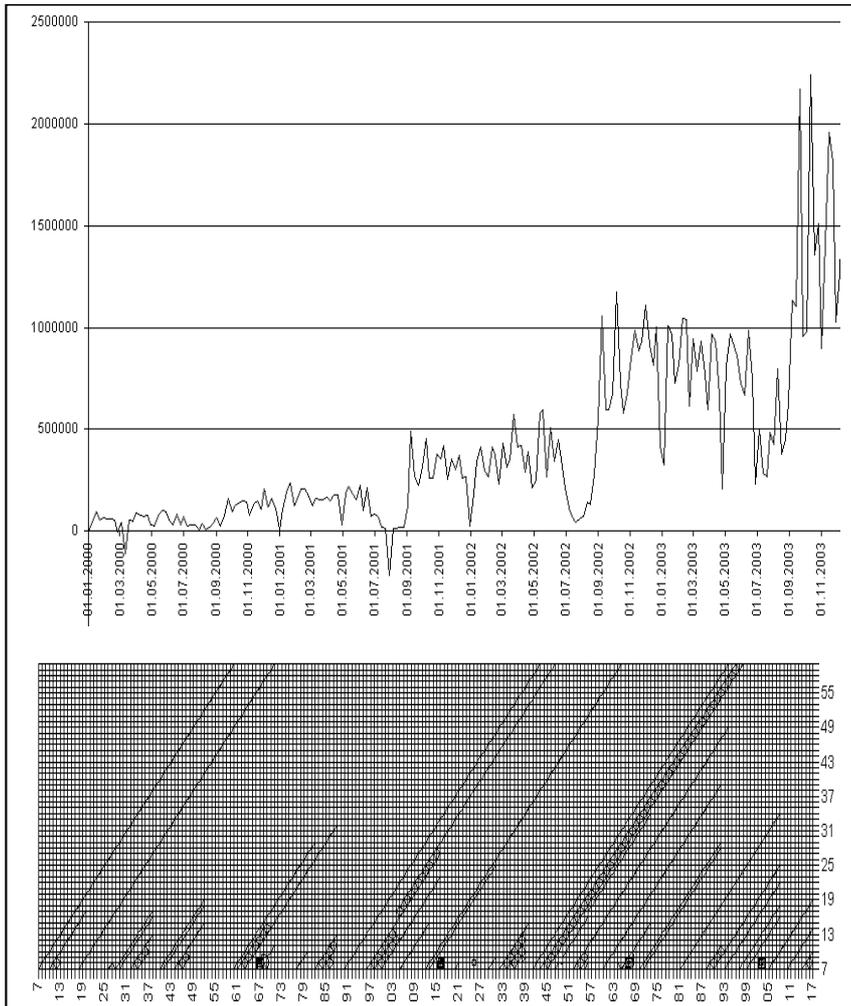


Рис. 6. Результат анализа тенденции

или smb-протоколам в локальной сети. Публикация производится путем загрузки файла/файлов на сервер по тем же протоколам. Поиск нужного ресурса происходит в виде обычного путешествия по каталогам файл-сервера.

### **Нейросетевая система управления ИХ**

Рассмотрим проблему подробнее. По сути структура каталогов отражает структуру категорий хранимой информации. А значит, задача размещения ресурса по каталогам, есть задача классификации ресурса. При классификации ресурса должны быть решены следующие вопросы. Во-первых, должно быть составлено дерево категорий. В нашем случае, это дерево каталогов на файл-сервере, составленное группой публикаторов. Во-вторых, требуется определить, какой (каким) из этих категорий, ресурс более всего соответствует. От решения этих вопросов напрямую зависит результат классификации, а значит и последующего поиска ресурса. Но результат решения этих вопросов сильно зависит от субъективных представлений эксперта о предметной области. В случае независимой публикации ресурсов из одной предметной области разными людьми, они могут оказаться в разных категориях. Кроме того, даже у одного человека эти представления могут меняться с течением времени. Это неминуемо скажется на качестве и скорости поиска нужного ресурса конечным пользователем. Таким образом, мы не можем полагаться на субъективность человека, когда речь идет о крупном информационном хранилище с большими объемами поступающих ресурсов. Нужно какое-то унифицированное представление о соответствии ресурсов дереву категорий. Предлагаемым решением проблемы построения дерева категорий является машинная кластеризация ресурсов. При кластеризации мы полагаемся на гипотезу о том, что смысловое содержание текста можно извлечь из статистического распределения слов. То есть, по частотному распределению слов, составляющих текст ресурса (или аннотации), мы можем определить его семантику. Согласно этому предположению кластеризация основывается на данных частотного анализа текста. Процесс частотного анализа связан с морфологическим анализом слов, составляющих текст. В этом процессе все словоформы одного слова должны быть учтены как одно слово, чтобы снизить шум на входах сети. Для этого применяется механизм стемминга.

**Нечеткая кластеризация информационных ресурсов с помощью сети Кохонена. Морфологический анализ текста. Стемминг**

Стемминг — это формальное выделение основы слова, стабильной, графически неизменной при склонении и спряжении части слова. Задача стемминга — это задача морфологического анализа языка. Словоформа представляется в виде:

*префикс + корень + суффикс + окончание + постфикс*

Основу слова составляют префикс и корень. Все словоформы в процессе стемминга приводятся к их основам, и уже основы участвуют в подсчете относительных частот. По результатам подсчета частот выделяются наиболее часто встречаемые словоформы, так называемые «ключевые слова». Ключевыми словами считаются основы, которые чаще всего встретились в тексте и являющиеся семантически значащими. Например, ключевыми словами не могут считаться предлоги, частицы, союзные слова, местоимения. Это так называемые «стоп-слова», которые отбрасываются при частотном анализе. Основываясь на результаты частотного анализа, строится нейронная сеть Кохонена, которая, собственно, и проводит кластеризацию ресурсов. На входы сети подаются относительные частоты встречаемости ключевых слов по каждому ресурсу. Сеть обучается по алгоритму «победитель получает все». Выходы сети соответствуют списку категорий. После окончания обработки всех ресурсов, сеть можно считать обученной. Для завершения построения дерева категорий, эксперту необходимо назвать образованные нейронной сетью кластеры. При этом он может основываться на наборы ключевых слов и ресурсов, отнесенных сетью к данной категории. После кластеризации задача публикации ресурсов решается этой же сетью. Она переводится в режим классификатора и выбирается наиболее подходящая для задачи функция активации. Поступающие ресурсы проходят стемминг и частотный анализ. Результаты частотного анализа подаются на входы сети. По состоянию выходов сети можно принимать решение о принадлежности ресурса к категориям. Таким образом мы избавляем публикаторов и экспертов от работы по созданию дерева категорий и принятия решений о публикации ресурса. Вопрос «куда публиковать ресурс» решен.

### Классификация информационного ресурса на основе ННС

Рост объемов проектной информации хранимой в электронном виде требует развития средств поиска нужных ресурсов. Обычно используются следующие виды поиска:

- обычный текстовый поиск по подстроке с некоторыми шаблонами;
- поиск, учитывающий морфологию языка;
- нечеткий поиск;
- браунинг по категориям-каталогам.

При больших объемах и семантической разнородности хранимой информации поиска по подстроке с шаблонами и морфологического поиска уже становится недостаточно. Четкий поиск с заданными критериями сменяется так называемым «нечетким поиском», позволяющим указывать неточные значения критериев поиска. Но зачастую нет возможности точно сформулировать такие критерии. Известно только, что ресурс может принадлежать определенным (достаточно общим) категориям. В таком случае поиск по категориям в виде простого выбора категории из дерева с просмотром всех ресурсов, наполняющих ее, бывает значительно удобнее. Этот вид поиска хоть и является исторически самым старым, но до сих пор не имеет отработанных стандартов на автоматизированные средства для построения индекса. Автоматизированная индексация информационных ресурсов для последующего поиска в виде браунинга по категориям требует создания особого вида индекса. Индекс должен отражать структуру категорий и их взаимосвязь с информационными ресурсами. Для этого предложен следующий формат:  $IR = (FR, NN)$ , где  $IR$  — индекс ресурса;  $FR$  — относительные частоты слов текста, составляющего ресурс;  $NN$  — обученная нейронная сеть; Здесь структура категорий и их взаимосвязь с информационными ресурсами будет неявно заключена в матрице весов нейронной сети. Относительные частоты ( $FR$ ) вычисляются по результатам стемминга и частотного анализа текста, составляющего ресурс. Нейронная сеть в индексе ( $NN$ ) — это нейронная сеть Кохонена, обученная в процессе кластеризации ресурсов. То есть, процесс кластеризации ресурсов также дает нам и уникальный по возможностям индекс. На основе этого индекса мы можем осуществлять все вышеописанные виды поиска, включая браунинг, а также комбинировать их. Например, можно дать возможность осуществлять полнотекстовый нечеткий поиск только в определенных категориях. В тени осталась задача управления хранилищем. При последующем росте

объемов информации, поступающей в хранилище, возможно возникновение ситуации, когда текущее дерево категорий не будет более покрывать всех поступающих ресурсов. Симптомами такой ситуации будет являться появление ресурсов, не принадлежащих ни одной категории, или наоборот, принадлежащих сразу большому количеству категорий. В этом случае требуется перестройка дерева категорий с учетом вновь поступивших ресурсов, то есть повторная кластеризация всего объема ресурсов с последующим повторным именованием полученных кластеров. Итак, что нам дало наделение информационного хранилища искусственным интеллектом в виде нейронной сети?

- Информационный репозиторий сам может построить дерево категорий для хранящихся в нем ресурсов и разнести ресурсы по дереву. Теперь время экспертов требуется только для того, чтобы поименовать полученные кластеры на основе списков ключевых слов и ресурсов, отнесенных к данной категории.
- Репозиторий сам решает, к какой категории отнести тот или иной ресурс. Это позволяет публикаторам не задумываться над тем, в каких категориях опубликовать ресурс. А пользователи не становятся заложниками субъективности экспертов и публикаторов.
- Автоматизация кластеризации порождает индекс, позволяющий реализовать различные виды поиска, включая браузеринг по категориям. Это ускоряет поиск со стороны пользователей хранилища.
- Хранилище само следит за актуальностью дерева категорий. При несоответствии дерева категорий поступающим ресурсам, то есть при невозможности правильно классифицировать ресурс, хранилище должно принять решение о перекластеризации.

Таким образом от понятия «информационное хранилище» мы переходим к понятию «интеллектуальное информационное хранилище». Набор его функций стал более широким и теперь включает в дополнение к предыдущим функциям:

- импорт уже имеющихся ресурсов;
- кластеризация имеющихся ресурсов с целью составления дерева категорий;
- обработка поступающих ресурсов, то есть выполнение классификации поступающих ресурсов и публикация их в соответствующих категориях;

- возможность полнотекстового и нечеткого поиска по текстам ресурсов, возможность поиска по дереву категорий;
- слежение за тем, чтобы дерево категорий всегда покрывало поступающие ресурсы (чтобы не было ресурсов, не относящихся ни к одной категории, или наоборот, относящихся ко всем категориям).

### Заключение

Перспективой развития гибридных систем считают разработку синергетического компьютера, построенного как взаимодействующая система. Вместе с тем, ННС успешно решают важные задачи в информационных системах. Для успеха применения важно разработать конкретные методики формирования обучающих выборок для ННС, выбрать архитектуры ННС, получить приемлемые настройки параметров.

### Литература

1. *Хакен Г., Хакен-Крель М.* Тайны восприятия. – М.: Институт компьютерных исследований, 2002. – 272 с.
2. *Ярушклина Н. Г.* Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.
3. *Редько В. Г.* Эволюционная кибернетика. – М.: Наука, 2001. – 156 с.
4. *Гаврилова Т. А., Хорошевский В. Ф.* Базы знаний интеллектуальных систем. – СПб: Питер, 2000. – 384 с.
5. *Ярушклина Н. Г.* Нечеткие нейронные сети с генетической настройкой // Лекция научной школы «Нейроинформатика–2004». – М.: МИФИ, 2004.
6. *Ярушклина Н. Г.* Нечеткие и гибридные системы: Обзор итогов и тенденций развития // Новости искусственного интеллекта, № 5, 2003, с. 5–12.
7. *Наместников А. М., Ярушклина Н. Г.* Эффективность генетических алгоритмов для задач автоматизированного проектирования // Известия РАН. Теория и системы управления. – № 2, 2002.
8. *Ярушклина Н. Г.* Гибридные системы, основанные на мягких вычислениях: Определение, архитектура, возможности // Программные продукты и системы. – № 3, 2002.

**Надежда Глебовна ЯРУШКИНА**, доктор технических наук, заведующая кафедрой «Информационные системы» Ульяновского государственного технического университета. Область научных интересов — теория нечетких множеств и систем, искусственные нейронные сети, генетические алгоритмы, гибридные системы. Автор 2 монографий и более 160 научных публикаций.