

Горькавый М.А., Соловьев В.А.

05.13.06

M.A.Gorkavy, V.S.Solovyev

**АВТОМАТИЗАЦИЯ СИНТЕЗА НЕЧЕТКИХ ПОДСИСТЕМ ЭКСПЕРТНОЙ СИСТЕМЫ
ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ**
**AUTOMATION OF FUZZY SUB-SYSTEMS SYNTHESIS WITHIN AN EXPERT SYSTEM
(ES) OF AN INDUSTRIAL ENTERPRISE**



Горькавый Михаил Александрович — аспирант кафедры «Электропривод и автоматизация промышленных установок» Комсомольского-на-Амуре государственного технического университета (г. Комсомольск-на-Амуре).

E-mail: idpo@knastu.ru

Mikhail A. Gorkavy — PhD candidate at the Department of Electro drive Engineering and Industrial Automation, Komsomolsk-on-Amur State Technical University (Komsomolsk-on-Amur). E-mail: idpo@knastu.ru



Соловьев Вячеслав Алексеевич — доктор технических наук, профессор, заведующий кафедрой «Электропривод и автоматизация промышленных установок» Комсомольского-на-Амуре государственного

технического университета (г. Комсомольск-на-Амуре). E-mail: kepapu@knastu.ru

Vyacheslav A. Solovyev — Doctor in Engineering, Professor, Head of the Department of Electro drive Engineering and Industrial Automation, Komsomolsk-on-Amur State Technical University (Komsomolsk-on-Amur). E-mail: kepapu@knastu.ru

Аннотация: В статье рассматривается проектирование экспертной системы (ЭС) для оценки и формирования компетентности технического персонала промышленного предприятия. В качестве структуры ЭС предлагается использование нечеткой модели компетенций. Раскрыты вопросы автоматизации и разработана программная реализация синтеза нечетких подсистем ЭС. Приведены примеры.

Summary: The paper presents a method for expert system (ES) design for the purpose of estimation and build-up of the competence of technical staff at an industrial enterprise. A fuzzy model of competencies is suggested to be used as the structure of the ES. Problems of automation are dealt with and a program realization of the ES fuzzy subsystems synthesis is developed. Examples of the above are given.

Ключевые слова: Экспертная система, компетентность, инновации, автоматизация, оптимизация, нечеткая логика, методика синтеза, лингвистические переменные.

Keywords: expert system, competence, innovations, automation, optimization, fuzzy logic, synthesis technique, linguistic variables

Введение

Авторами в настоящей работе решается задача проектирования экспертной системы, стратегической целью которой является формирование компетентности технического персонала промышленного предприятия.

Необходимость создания такой системы обосновано. Из-за высокого темпа развития научно-технического прогресса и высокой конкуренции практически в любом производстве отдельное предприятие, для того чтобы поддерживать необходимый уровень конкурентоспособности, вынуждено осуществлять политику инновационного развития. Носителями и агентами реализации инновационных решений служат люди, а в конкретном случае — персонал предприятия. Способность синтеза и реализации инновационных решений напрямую зависит от знания, умения, навыков, жизненного опыта, интуиции и т.д. конкретного сотрудника, т.е. его компетентности. Поэтому для успешной разработки и реализации инновационной стратегии предприятия в нем должна быть сформирована система, обладающая знаниями о компетентности каждого отдельно взятого сотрудника предприятия, иными словами, система, облада-

Поступила в редколлегию 20.01.2010.

ющая метазнаниями. С помощью метазнаний в любой момент можно определить сотрудника предприятия, обладающего необходимой для решения какой-либо конкретной задачи компетентностью, а также провести отбор и оценку персонала, сформировать необходимую компетентность и многое другое. Для синтеза экспертной системы и формирования метазнаний о компетентности персонала промышленного предприятия металлургического комплекса ОАО «Амур-металл» авторами в работе [1] была разработана методика построения (в двух вариантах), и предложены способы использования нечетких моделей компетенций в качестве базовой структуры экспертной системы. В качестве математического аппарата предлагается использовать теорию нечетких множеств, изложенную в работе [2]. Использование теории нечетких множеств и нечеткой логики при построении экспертной системы видится авторам целесообразным, поскольку эталонным поведением экспертной системы является именно поведение человека (эксперта), и в качестве языка «мышления» экспертной системы необходимо использовать язык, максимально приближенный к языку мышления человека.

Автоматизация процесса построения нечеткой модели компетенций

Первый вариант методики, подробно изложенный и проиллюстрированный примером (работа [1]), включает в себя формирование нечеткой модели компетенций на основе определения экспертной комиссией списка имен лингвистических переменных (названий компетенций), содержания лингвистических переменных (универсальное множество, количество и семантику лингвистических термов и т.д.), типа лингвистических переменных (входная или выходная), идентификации нелинейных зависимостей нечеткими базами знаний, а также обучения и тестирования полученных нечетких систем. Трудоемкость первого варианта высока, т.к. большое количество операций по синтезу нечеткой системы необходимо выполнять человеку. Второй вариант методики [1] представляет собой автоматизированное построение нечеткой модели компетенций на основе исходной статистической экспертной информации и является предметом обсуждения в первой части текущей работы. Все операции, выполняющиеся в первом варианте, также выполняются и во втором, но в автоматизированном или автоматическом режиме. В качестве среды разработки нечеткой системы авторами была выбрана среда Matlab и встроенные пакеты Fuzzy Logic Toolbox и Optimization Toolbox, в отдельных частях работы также используются авторские разработки, расширяющие вышеприведенные пакеты.

Исходной информацией для синтеза нечеткой модели в автоматизированном режиме служит таблица размерности $n+1 \times m$, содержащая экспертную информацию (см. табл. 1).

Таблица 1

Исходная экспертная информация

№	In ¹	In ²	...	In _i	...	In _n	Out ₁
1	in ₁₁	in ₂₁		ini ₁		inn ₁	out ₁₁
2	in ₁₂	in ₂₂		ini ₂		inn ₂	out ₁₂
m	in _{1m}	in _{2m}		in _{im}		inn _m	out _{1m}

В табл. 1 n —количество входных лингвистических переменных, m —объем экспертной информации, In_i—название i -й входной лингвистической переменной, Out₁—название первой (и в нашей системе единственной) выходной лингвистической переменной, in_{ij}— j -е значение i -й входной переменной, Out_{1j}— j -е предполагаемое значение первой выходной переменной, при соответствующих входных переменных. Необходимо отметить, что чем больше объем экспертной информации, тем имеется более полное представление для синтеза нечеткой системы.

В таблице 1 содержится исчерпывающая информация, необходимая для синтеза системы. Действительно, имеются имена входных и выходной лингвистических переменных, известно универсальное множество для каждой входной переменной, определяемое как $U_i = [\min(In_i), \max(In_i)]$, известно универсальное множество для выходной переменной, определяемое аналогично. Содержание лингвистических переменных, а именно количество термов и их семантику, можно определить двумя способами:

1) эксперт указывает количество термов для каждой лингвистической переменной, а также указывает их семантику путем задания типа функций принадлежности, которые, на его взгляд, лучше всего отражают содержание соответствующих лингвистических переменных;

2) количество термов и соответствующие им функции принадлежности определяются на основе проведения кластеризации исходных данных, например, алгоритмом горной кластеризации или нечетких s -средних [3].

Необходимо отметить, что второй способ целесообразнее использовать для построения классификаторов, при реализации второго способа для синтеза «классических» нечетких систем, в которых наименьшему и наибольшему значению из универсального множества соответствуют точки максимума функций принадлежности крайних термов, теряется смысловая нагрузка термов лингвистических переменных, т.е. прозрачность системы.

После того, как подготовлена исходная информация и определено содержание лингвистических переменных, необходимо идентифицировать нелинейные зависимости нечеткими базами знаний. В работе рассматриваются базы знаний типа Mamdani и Sugeno.

Идентифицировать нелинейную зависимость в автоматическом режиме базой знаний типа Sugeno предлагается с использованием Anfis-редактора. В основе технологии работы Anfis-редактора лежит идея пред-

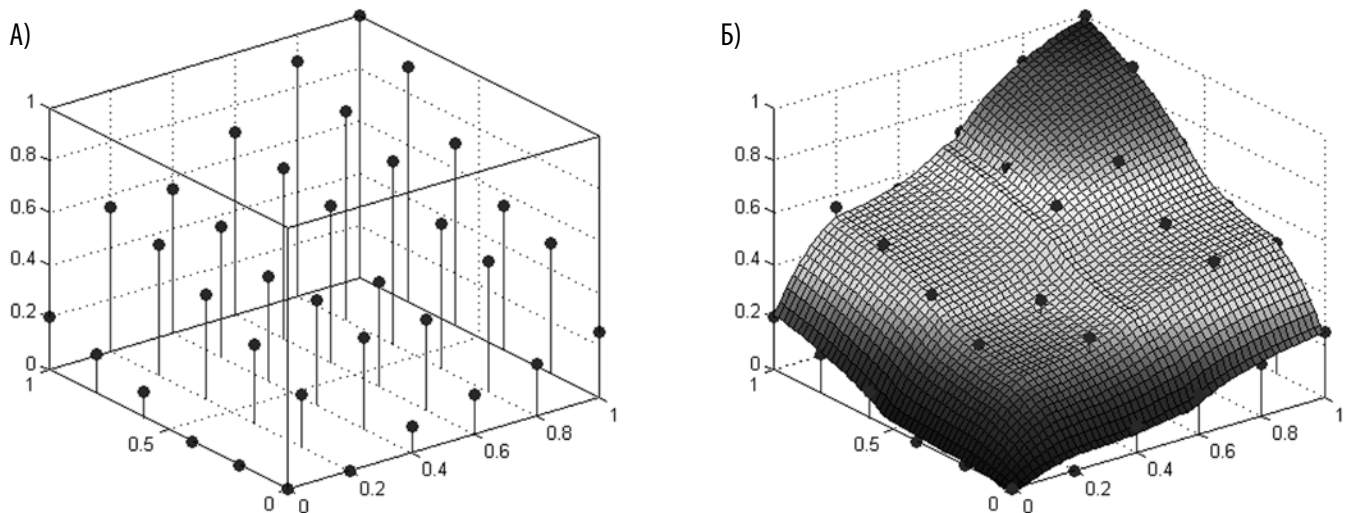


Рис. 1. Результаты моделирования: а — исходная экспертная информация; б — синтезированная нечеткая система типа Sugeno

ставления нечеткой системы в виде нейро-нечеткой сети — нейронной сети прямого распространения сигнала особого типа [3]. Архитектура нейро-нечеткой сети изоморфна нечеткой базе знаний. Использование в нейро-нечетких сетях дифференцируемых реализаций треугольных норм, а также гладких функций принадлежности, позволяет применять для настройки нейро-нечетких сетей быстрые алгоритмы обучения, основанные на методе обратного распространения ошибки.

Алгоритм работы Anfis-редактора следующий: на начальном этапе, на основе таблицы исходной экспертной информации и в зависимости от способа определения содержания лингвистических переменных, редактор вызовом функций genfis1 или genfis2 генерирует из данных исходную нечеткую систему типа Sugeno без использования кластеризации или с использованием субтрактивной кластеризации соответственно. Способы импликации, дефазификации, агрегации и т. д. устанавливаются по умолчанию, подробно синтаксис функций изложен в прим. 1. После того, как сформирована исходная система и структура нейро-нечеткой сети, редактор производит обучение системы, результатом которого является нахождение оптимальных значений параметров нечеткой системы согласно критерию оптимизации — минимизации невязки на обучающей выборке:

$$RMSE = \sqrt{\frac{1}{M} \sum_{r=1}^M (y_r - F(P, W, X_r))^2} \rightarrow \min, \quad (1)$$

где: $(X_r, y_r), r = \overline{1, M}$ — обучающая выборка из M пар экспериментальных данных, связывающих входы $X_r = (x_{r,1}, x_{r,2}, \dots, x_{r,n})$ с выходом y в r -й паре обучающей выборки;

P — вектор параметров функций принадлежности термов входных и выходной переменных;

W — вектор весовых коэффициентов правил базы знаний;

$F(P, W, X)$ — результат вывода по нечеткой базе знаний с параметрами (P, W) при значении входов X_r .

Обучение осуществляется посредством вызова функции anfis (подробное описание функции, а также список входных и выходных аргументов представлен в прим. 1). В функции anfis предусмотрено два варианта оптимизации параметров исходной нечеткой системы: метод обратного распространения ошибки и гибридный алгоритм. Выбор того или иного алгоритма оптимизации зависит от степени адекватности его использования в конкретной ситуации. Результатом работы функции anfis является обученная нечеткая система (система с установленными оптимальными параметрами). В редакторе предусмотрено тестирование обученной нечеткой системы на тестовой выборке. Тестирование необходимо для проверки адекватности результатов нечеткой системы на значениях входных лингвистических переменных из универсальных множеств, не попавших в обучающую выборку, а также для исключения процесса переобучения системы.

Пример 1. Рассмотрим автоматический синтез нечеткой системы (нечеткой модели компетенций) и автоматическую идентификацию нелинейной зависимости базой знаний типа Sugeno для оценки компетентности сотрудника сортопрокатного цеха ОАО «Амур-металл», осуществляющего установку и наладку микропроцессорного оборудования фирмы Siemens.

В таблице 2 представлена исходная экспертная информация, где Out1 (имя выходной лингвистической переменной) — «наладчик микропроцессорного оборудования фирмы Siemens технологических участков сортопрокатного цеха», in¹ (имя первой входной лингвистической переменной) — «теоретические аспекты функционирования микроконтроллеров фирмы Siemens», in² (имя второй входной лингвистической переменной) — «практические навыки установки и настройки микроконтроллеров фирмы Siemens», m (объем экспертной информации) = 36, единицы измерения универсальных множеств для всех переменных — доли от максимально возможного уровня соответствующей компетенции.

Таблица 2

Экспертная информация для построения нечеткой модели компетенций для наладчика микропроцессорного оборудования

№	In1	In ²	Out1	№	In1	In ²	Out1	№	In1	In ²	Out1	№	In1	In ²	Out1
1	0	0	0	10	0.6	0.2	0.35	19	0	0.6	0.1	28	0.6	0.8	0.65
2	0.2	0	0	11	0.8	0.2	0.5	20	0.2	0.6	0.4	29	0.8	0.8	0.8
3	0.4	0	0.1	12	1	0.2	0.5	21	0.4	0.6	0.4	30	1	0.8	0.9
4	0.6	0	0.15	13	0	0.4	0	22	0.6	0.6	0.6	31	0	1	0.2
5	0.8	0	0.2	14	0.2	0.4	0.3	23	0.8	0.6	0.7	32	0.2	1	0.55
6	1	0	0.25	15	0.4	0.4	0.4	24	1	0.6	0.7	33	0.4	1	0.55
7	0	0.2	0	16	0.6	0.4	0.4	25	0	0.8	0.15	34	0.6	1	0.7
8	0.2	0.2	0.2	17	0.8	0.4	0.55	26	0.2	0.8	0.5	35	0.8	1	0.9
9	0.4	0.2	0.35	18	1	0.4	0.55	27	0.4	0.8	0.5	36	1	1	1

Графическое изображение исходной экспертной информации представлено на рисунке 1 а. Обработка экспертной информации, согласно вышеизложенному алгоритму с использованием Anfis-редактора, выполнялась со следующими параметрами синтеза системы:

- исходная система синтезирована методом решетчатого разбиения без проведения кластеризации исходных данных (функция *genfis1*);
- количество лингвистических термов для входных переменных — 3;
- тип функций принадлежности лингвистических термов — «trimf» (треугольная);
- обучение исходной нечеткой системы проводилась с использованием гибридного метода оптимизации;
- количество эпох обучения — 50.

В результате была синтезирована нечеткая система, результаты моделирования которой представлены на рисунке 1 б, оптимальные значения параметров системы, согласно критерию (1), полученные в результате обучения, представлены в таблице 3. При этом невязка на обучающей выборке составила 0,033.

Таблица 3

Оптимальные параметры обученной нечеткой системы

Имя терма	Параметры функции принадлежности терма	Имя терма	Параметры функции принадлежности терма	База знаний в индексном формате
In1mf1	[-0.5-0.1306 0.5353 0]	Out1mf1	-0.02622	11, 1 (1):1
In1mf ²	[0.002525 0.2256 1.006 0]	Out1mf ²	0.03755	12, 2 (1):1
In1mf ³	[0.5851 0.9795 1.5 0]	Out1mf ³	0.2159	13, 3 (1):1
In ² mf1	[-0.5-0.1009 0.5369 0]	Out1mf4	0.1218	21, 4 (1):1
In ² mf ²	[-9.35e- 005 0.3315 1.007 0]	Out1mf5	0.5359	22, 5 (1):1
In ² mf ³	[0.5268 0.9906 1.5 0]	Out1mf6	0.6791	23, 6 (1):1
	Out1mf7	0.2656	31, 7 (1):1	
	Out1mf8	0.6614	32, 8 (1):1	
	Out1mf9	1.009	33, 9 (1):1	

В результате автоматизации процесса построения нечетких систем значительно сокращается время, необходимое для синтеза нечеткой модели компетенций и формирования метазнаний.

Необходимо отметить, что одним из недостатков нечетких баз знаний типа Sugeno является сложность их интерпретации для человека, поэтому авторами была разработана функция *genfis4*, расширяющая пакет Fuzzy Logic Toolbox, позволяющая производить экстракцию из данных нечеткой системы типа Mamdani в автоматическом режиме. Кроме того, на основе использования нечетких систем типа Mamdani для построения нечетких моделей компетенций, возможна организация вывода в виде лингвистических термов, что позволяет приблизить язык «мышления» экспертной системы к языку мышления эксперта.

На первом этапе функция *genfis4* генерирует из данных систему нечеткого вывода типа Mamdani по методу решетчатого разбиения. Функции принадлежности входных и выходной переменной выбираются таким образом, чтобы равномерно покрыть диапазоны изменения данных. Объем базы знаний определяется как произведение мощностей терм-множеств входных и выходной переменных, следовательно, генерируются всевозможные правила. Веса правил принимаются равными нулю. Полученная система нечеткого вывода не отражает представленной данными закономерности. Она является исходной системой для обучения (на втором этапе работы функции *genfis4*) посредством нелинейной оптимизации, в результате которой идентифицируются заложенные в данные закономерности.

Нелинейная оптимизация проводится с помощью встроенной функции *fmincon* пакета Optimization Toolbox, решающей задачу поиска минимума нелинейной задачи с ограничениями:

$$\min_x f(x) \quad (2)$$

при условии что:

$c(x) < 0, ceq(x) = 0, A \cdot x \leq b, Aeq \cdot x = beq, lb < x < ub$
где x, b, beq, lb, ub — векторы; A, Aeq — матрицы; $c(x), ceq(x)$ — есть функции; $f(x)$ — функция, возвращающая скаляр. Полное описание функции представлено в (прим. 1)

Синтаксис функции *genfis4* включает до четырех входных аргументов, *fis=genfis4 (data, numMfs, mftype, options)*, где:

data — матрица исходных данных, каждая строка которой является парой «входы-выход»;

numMfs — необязательный аргумент, задающий количество термов для оценки входных и выходной переменных. Значение по умолчанию — 3;

mftype — необязательный аргумент, задающий типы функций принадлежности нечетких термов входных и выходной переменной. Допустимые значения: 'trimf' — треугольная, 'gaussmf' — Гауссова. Значение по умолчанию — 'trimf';

options — структура параметров настройки для нелинейной оптимизации функцией *fmincon*.

Результаты автоматической идентификации нели-

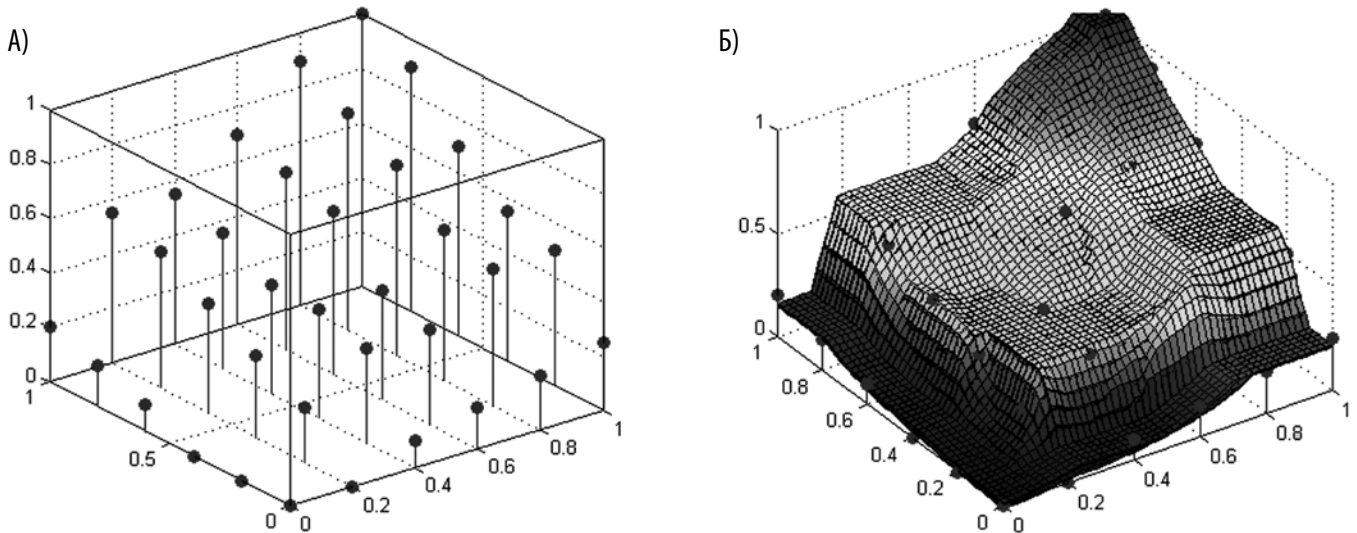


Рис. 2. Результаты моделирования: а — исходная экспертная информация; б — синтезированная нечеткая система типа Mamdani

нейной зависимости из примера 1 (см. табл. 1) функцией *genfis4* представлены на рисунке 2. Оптимальные параметры, согласно критерию (1), синтезированной системы представлены в таблице 4.

При этом невязка на обучающей выборке, согласно критерию (1), составила 0,027.

Автоматическая идентификация проводилась со следующими параметрами функции *genfis4*:

- data — согласно таблице 1;
- numMfs — 3;
- mftype — 'trimf';
- options = [];
- options = optimset ('Display', 'iter');
- options. DiffMinChange = 0.0001;
- options. DiffMaxChange = 0.2;
- options. LargeScale = 'off';
- options. MaxIter = 50;

Таблица 4

Оптимальные параметры обученной нечеткой системы

Имя терма	Параметры функции принадлежности терма	База знаний в индексном формате			
		№	Правило	№	Правило
In1mf1	[-0.40 0.252 0]	1	11, 1 (1):1	10	22, 2 (0.035):1
In1mf²	[0.1 0.4247 0.9 0]	2	12, 1 (0.26382):1	11	23, 2 (0.05784):1
In1mf³	[0.29451 1.4 0]	3	13, 1 (0.08081):1	12	31, 2 (0.02124):1
In²mf1	[-0.40 0.4487 0]	4	21, 1 (0.26419):1	13	32, 2 (0.08019):1
In²mf²	[0.1 0.3051 0.9 0]	5	22, 1 (0.26585):1	14	22, 3 (0.21133):1
In²mf³	[0.27711 1.4 0]	6	23, 1 (0.09663):1	15	23, 3 (0.224):1

Out1mf1	[-0.58-0.1 0.1696 0]	7	31, 1 (0.11763):1	16	32, 3 (0.26223):1
Out1mf²	[0.02 0.498 0.998 0]	8	32, 1 (0.11168):1	17	33, 3 (1):1
Out1mf³	[0.9947 1.1 1.58 0]	9	13, 2 (0.00813):1		

Разработанная авторами методика автоматической идентификации нелинейных зависимостей (представленных экспертными данными) нечеткими системами для построения моделей компетенций позволяет значительно сократить время формирования метазнаний и улучшить качество синтезируемых систем, согласно критерию (1), посредством проведения процесса обучения.

В ходе проведения исследования авторами были установлены некоторые особенности автоматической идентификации нечеткими системами Mamdani и Sugeno, позволяющие сделать выводы о целесообразности использования того или иного типа базы знаний в конкретной ситуации.

При условии, что исходная экспертная информация репрезентативна и представлена с достаточным шагом дискретизации, нечеткие системы типа Sugeno легче поддаются обучению, т. е. качественнее согласно критерию (1). Это обусловлено меньшей размерностью задачи оптимизации, т. к. выходная лингвистическая переменная в нечеткой базе типа Sugeno представлена в виде констант или линейных выражений. Нечеткие системы типа Mamdani лучше поддаются интерпретации для человека, т. к. выходная лингвистическая переменная представлена в виде лингвистических термов, семантика которых определяется соответствующими функциями принадлежности. Легкость в интерпретации позволяет организовать процесс производства мотивированного заключения.

Распределенная нечеткая модель компетенций с классификацией надежности решения
Нечеткая система, построенная на основе экспертной информации, представленная мнением только одного эксперта (таблица 1), является в той или иной мере

субъективной. Для повышения объективности системы необходимо в качестве исходных данных использовать мнения нескольких экспертов. Исходными данными для построения объективной нечеткой системы является таблица 5. Все обозначения таблицы 5 аналогичны обозначениям таблицы 1, индекс r — число экспертов.

Таблица 5

Исходная экспертная информация для объективной системы

№	In1	...	Inn	Out11	...	Out1r
1	in11	...	inn1	out111	...	out11r
2	in12	...	inn ²	out121	...	out12r
		
m	in1m	...	innm	out1m1	...	out1mr

Число экспертов определяется исходя из конкретных условий задачи и критериев, предъявляемых к решению.

Подготовительным этапом построения системы является определение консолидированных значений выходной переменной. Самым простым таким способом является определение среднего значения:

$$\overline{out_{1i}} = \frac{out_{1i}^1 + out_{1i}^2 + \dots + out_{1i}^r}{r} \quad (3)$$

Либо среднего взвешенного:

$$\overline{out_{1i}} = \frac{\omega_1 out_{1i}^1 + \omega_2 out_{1i}^2 + \dots + \omega_r out_{1i}^r}{\omega_1 + \omega_2 + \dots + \omega_r} \quad (4)$$

где ω_1 — вес мнения i -го эксперта.

В итоге формируются m усредненных значений выходной переменной, по которым совместно со значениями входных переменных строится нечеткая система путем проведения идентификации нелинейной зависимости нечеткой базой знаний Sugeno или Mamdani, согласно вышеприведенным методикам автоматического синтеза.

Несомненно, построенная таким образом система будет более объективна, чем система, построенная на основании мнения только одного эксперта. Но процедура усреднения экспертных данных на этапе формирования входной матрицы для синтезируемой нечеткой системы оставляет без внимания заложенную в экспертные данные информацию, а именно, степень согласованности мнений экспертов. Другими словами, не учитывается рассеивание предполагаемых r экспертами значений выходной переменной по каждой строке таблицы 5. Действительно, информация о степени согласованности решения очень важна, т. к. опасно опираться на решение, основанное на слишком разных мнениях экспертов.

Для построения объективной системы и учета степени согласованности экспертов авторами была разработана методика синтеза распределенной системы нечеткого вывода с классификацией по степени надежности. Степень надежности или класс надежности — определяется трехмерной системой нечеткого дискретного вывода нечетким классификатором.

Входными переменными классификатора являются количество экспертов r и степень согласованности их решения.

Методика синтеза распределенной системы с классификацией надежности решения:

Проводится автоматическая идентификация нелинейных зависимостей r подсистемами нечеткого вывода, где r — количество экспертов, т. е. мнению каждого эксперта соответствует своя нечеткая подсистема. При построении нечетких подсистем определяется RMSE и проверяется на соответствие допустимой невязки на обучающей выборке. Если необходимо, то определяются веса мнений экспертов.

Синтезируется нечеткий классификатор. Классификатор имеет две входные переменные: In1 — количество экспертов, In² — отношение стандартного (среднеквадратического) отклонения к размаху универсального множества выходной переменной, т. е. нормированное среднеквадратичное отклонение. In² — является мерой определения степени согласованности мнений экспертов. Значение нормированного среднеквадратического отклонения в точке i определяется по формуле

$$S_{norm\ i} = \frac{\sqrt{\frac{1}{r} \sum_{k=1}^r (out_{1i}^k - \overline{out_{1i}})^2}}{\max(out_{1i}) - \min(out_{1i})} \quad (5)$$

Необходимо отметить, что выход нечеткого классификатора является дискретным, с количеством дискрет, равным количеству классов надежности. Нечеткий классификатор представляет собой систему нечеткого вывода типа Sugeno, с тем основным отличием, что процедура определения четкого выходного значения — процедура дефазификации — заменена процедурой отыскания выходного лингвистического термина в правиле с максимальной степенью выполнения. Решение выдается в виде имени найденного лингвистического термина.

Для синтеза распределенной нечеткой системы с классификацией решения по степени надежности авторами была разработана функция genfisD. Синтаксис функции genfisD включает до 7 входных аргументов, [sub_fis, fis_class] = genfisD (exp_number, data, numMfs, mftype, trnopt, optmethod, class_opt), где:

exp_number — количество экспертов;

data — матрица исходных данных, в формате таблицы 5;

numMfs — необязательный аргумент, задающий количество термов для оценки входных и выходной переменных. Значение по умолчанию — 3;

mftype — необязательный аргумент, задающий типы функций принадлежности нечетких термов входных и выходной переменной. Допустимые значения: 'trimf' — треугольная, 'gaussmf' — Гауссова. Значение по умолчанию — 'trimf';

trnopt — вектор параметров настройки, структура и значения вектора аналогичны одноименному вектору в функции anfis (см. прим. 1);

optmethod — метод оптимизации; 0 — гибридный, 1 — метод обратного распространения ошибки;

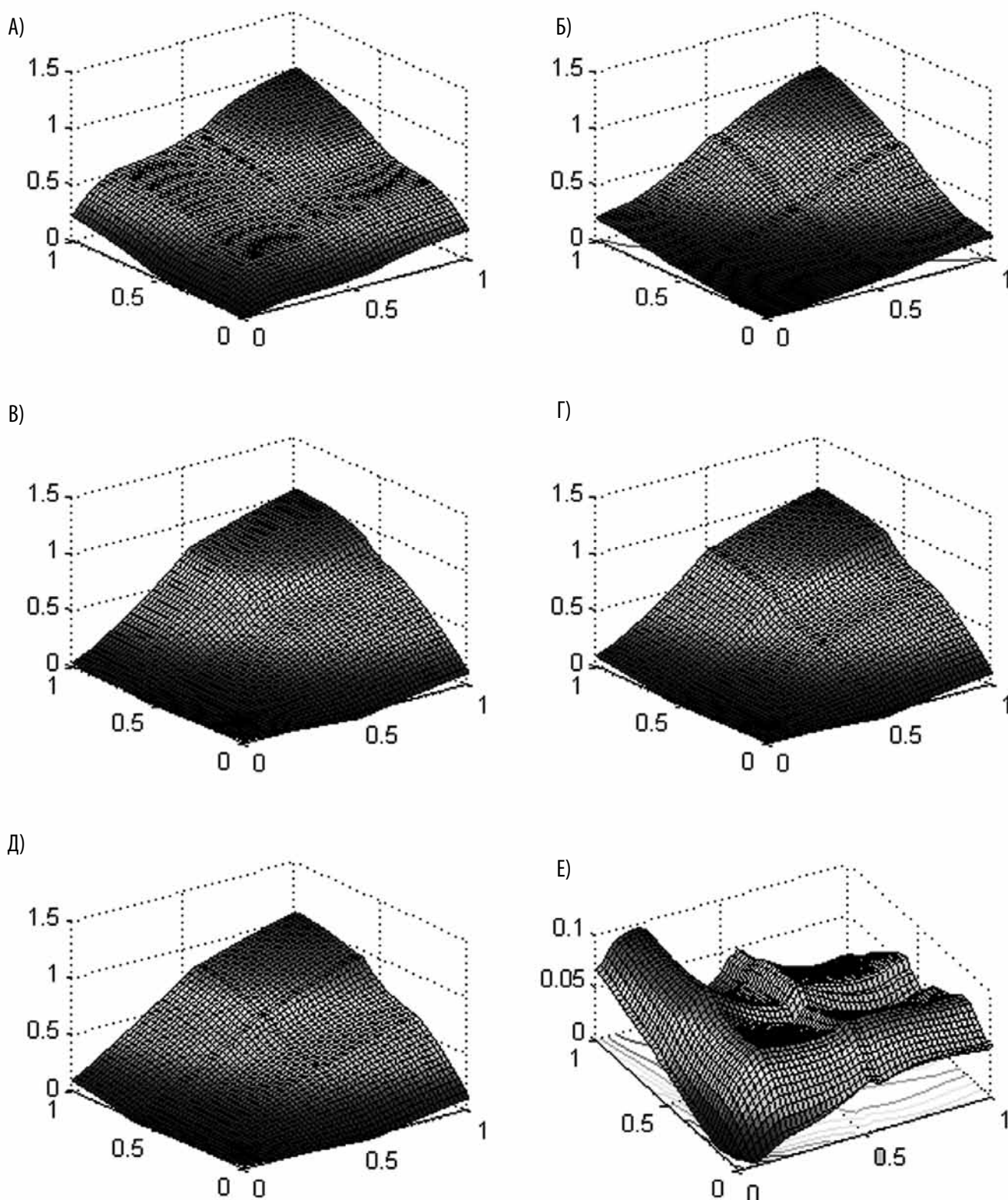


Рис. 3. Результаты моделирования распределенной системы
 а-д — подсистемы № 1-5 соответственно; е — график распределения стандартного отклонения

class_opt—вектор параметров настройки классификатора:
 class_opt (1)—количество классов;
 class_opt (2)—минимальное допустимое количество экспертов;
 class_opt (3)—максимальное допустимое количество

экспертов;
 class_opt (4)—минимальное допустимое s_norm системы;
 class_opt (5)—максимальное допустимое s_norm системы;
 Функция genfisD может вернуть два выходных

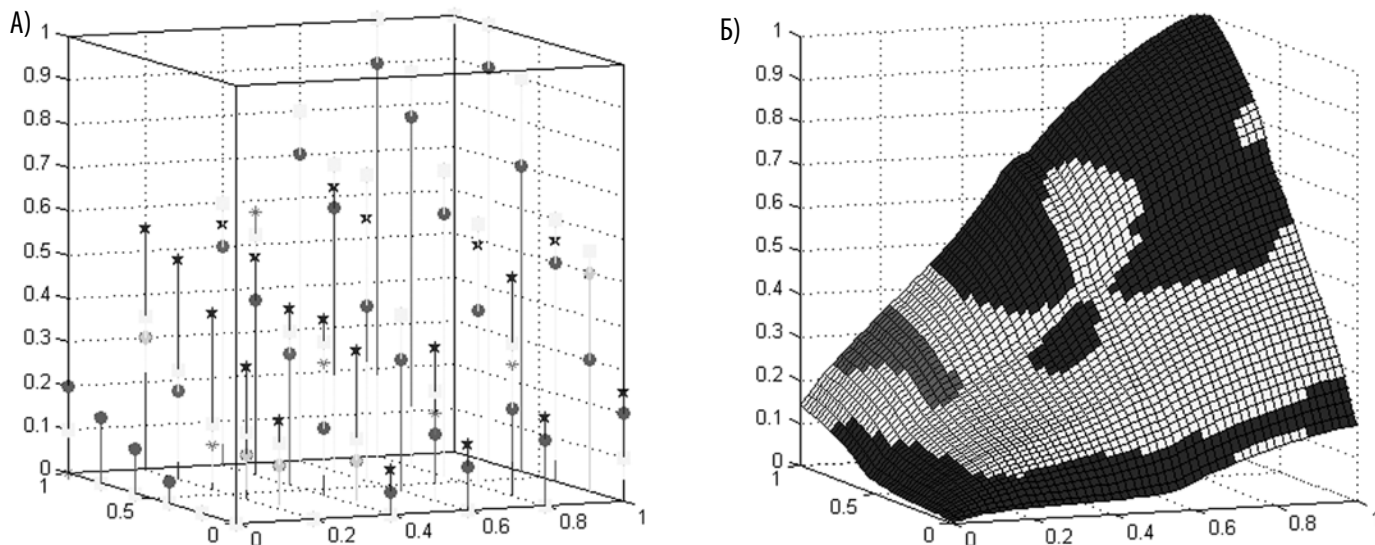


Рис. 4. Результаты моделирования распределенной системы:

а — исходная экспертная информация;

б — поверхность распределения интегрированного среднего значения с классификацией по зонам надежности (1 класс — чёрный цвет; 2 класс — белый цвет; 3 класс — серый цвет)

аргумента:

1) `sub_fis` — $1 \times r$ вектор структур, каждая из которых представляет собой настроенную подсистему нечеткого вывода, построенную на основании мнения одного эксперта.

2) `fis_class` — настроенная система дискретного нечеткого вывода — нечеткий классификатор.

Кроме того функция `genfisD` автоматически производит моделирование полученной распределенной системы. Результаты моделирования включают в себя графики вывода по всем нечетким подсистемам, график изменения стандартного отклонения, а также график интегрированного вывода распределенной системы в виде поверхности средних значений с классификацией по степени надежности. Графически классификация изображается в виде s — цветов, где s — количество классов.

Для осуществления логического вывода по нечеткой распределенной системе разработана функция `evalfisD`, которая представляет собой модифицированную стандартную функцию `evalfis` (см. прим. 1). Отличие заключается в том, что `evalfisD` осуществляет нечеткий логический вывод одновременно по r нечетким подсистемам, т. е. по всем выходным аргументам функции `genfisD`, осуществляет интегрированный вывод в виде среднего значения и производит классификацию значений.

Синтаксис функции: `[output, class, out_sub_fis] = evalfisD (input, fis_class, sub_fis)`, где:

`output` — матрица усредненных значений выходной переменной по всем нечетким подсистемам, полученная на основе входных данных `input`;

`class` — матрица значений выходной переменной нечеткой системы `fis_class`, получаемая в результате нечеткого вывода для данных `input`;

`out_sub_fis` — матрица значений выходных переменных по каждой подсистеме, размерность матри-

цы $m \times r$, где m — количество строк входных данных, r — количество экспертов. Каждый столбец соответствует i -й нечеткой подсистеме;

`input` — исходные данные, по которым необходимо произвести нечеткий вывод;

`fis_class`, `sub_fis` — аргументы, аналогичные одноименным аргументам функции `genfisD`.

Иллюстрация работы функций `genfisD` и `evalfisD` представлена в примере 2.

Пример 2. Первоначальные условия и задачи те же, что и в примере 1. Исходная экспертная информация пяти экспертов представлена в таблице 6.

Таблица 6

Исходная экспертная информация для примера 2

№	In1	In2	Out11	Out12	Out13	Out14	Out15
1	0	0	0	0	0	0	0
2	0.2000	0	0	0	0	0	0
3	0.4000	0	0.1000	0.0500	0	0	0
4	0.6000	0	0.1500	0.1000	0	0	0
5	0.8000	0	0.2000	0.1500	0	0	0
6	1.0000	0	0.2500	0.2000	0.1000	0.1000	0.1000
7	0	0.2000	0	0	0	0	0
8	0.2000	0.2000	0.2000	0.1000	0.1000	0.1000	0.1500
9	0.4000	0.2000	0.3500	0.1000	0.1000	0.1000	0.1500
10	0.6000	0.2000	0.3500	0.1500	0.2000	0.2500	0.2500
11	0.8000	0.2000	0.5000	0.2000	0.3000	0.3500	0.3500
12	1.0000	0.2000	0.5000	0.3000	0.5000	0.5000	0.5500
13	0	0.4000	0	0.0500	0	0	0
14	0.2000	0.4000	0.3000	0.1000	0.1000	0.1000	0.1500

15	0.4000	0.4000	0.4000	0.1500	0.3000	0.3500	0.3500
16	0.6000	0.4000	0.4000	0.3000	0.4000	0.4000	0.4000
17	0.8000	0.4000	0.5500	0.4000	0.6000	0.6000	0.6000
18	1.0000	0.4000	0.5500	0.5000	0.6000	0.6000	0.6000
19	0	0.6000	0.1000	0.1000	0	0	0
20	0.2000	0.6000	0.4000	0.1500	0.1000	0.1500	0.1500
21	0.4000	0.6000	0.4000	0.3000	0.3000	0.3500	0.3500
22	0.6000	0.6000	0.6000	0.4000	0.7000	0.7000	0.7000
23	0.8000	0.6000	0.7000	0.6000	0.7000	0.7000	0.7000
24	1.0000	0.6000	0.7000	0.7000	0.9000	0.9000	0.9000
25	0	0.8000	0.1500	0.1500	0	0	0
26	0.2000	0.8000	0.5000	0.2000	0.2000	0.2500	0.2500
27	0.4000	0.8000	0.5000	0.4000	0.6000	0.5500	0.5500
28	0.6000	0.8000	0.6500	0.6000	0.7000	0.7000	0.7000
29	0.8000	0.8000	0.8000	0.8000	0.9000	0.9000	0.9000
30	1.0000	0.8000	0.9000	0.9000	1.0000	1.0000	1.0000
31	0	1.0000	0.2000	0.2000	0	0.1000	0.1000
32	0.2000	1.0000	0.5500	0.3000	0.3000	0.3000	0.3500
33	0.4000	1.0000	0.5500	0.5000	0.5000	0.6000	0.6000
34	0.6000	1.0000	0.7000	0.7000	0.8000	0.8000	0.8000
35	0.8000	1.0000	0.9000	0.9000	1.0000	1.0000	1.0000
36	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Функция `genfisD`, вызванная с параметрами:

```
class_opt
class_opt (1) =3;
class_opt (2) =1;
class_opt (3) =10;
class_opt (4) =0;
class_opt (5) =0,1;
optmethod = 1;
```

все остальные параметры — по умолчанию; сгенерировала распределенную нечеткую систему, выдав результаты моделирования, представленные на рисунке 3 и 4. Графики вывода по пяти нечетким подсистемам представлены на рисунках 3 а—д. Невязка на обучающей выборке для каждой из пяти нечетких подсистем находится в пределах допустимых значений 0,025-0,03. Поверхность распределения нормированного среднеквадратичного отклонения представлена на рисунке 3 е. Максимальное значение нормированного среднеквадратичного отклонения составляет 0,1.

На основании анализа поверхностей вывода нечетких подсистем и нормированного среднеквадратичного отклонения специалист, проектирующий распределенную нечеткую систему, принял решение принять распределенную систему, т. е. вывод по распределенной нечеткой системе осуществлять согласно полученной зависимости, график которой представлен на рисунке 4 б. Исходная экспертная информация представлена на рисунке 4 а. Таким образом, синте-

зирована распределенная нечеткая система с классификацией решения по степени надежности, позволяющая определить значение выходной переменной на множестве значений входных переменных и классифицировать надежность полученного решения. В данном примере наивысший класс надежности — 1 (синий), низший — 3 (красный). Но тем не менее все три класса удовлетворяют требованиям качества синтезируемой системы.

Все элементы распределенной системы являются открытыми, т. е. на любом этапе проектирования и работы распределенную систему можно корректировать с помощью стандартных средств пакетов `Fuzzy Logic Toolbox` и `Optimization Toolbox`.

Пример вывода функции `evalfisD` для полученной распределенной системы на входных данных [0.25 0.53; 0.15 1] представлен ниже:

```
>> [output, class, out_sub_fis] =evalfisD ( [0.25 0.53;
0.15 1],fis_class, sub_fis)
```

```
output =
0.2132
0.2806
class =
2
3
out_sub_fis =
0.3658 0.1511 0.1649 0.1791 0.2050
0.4696 0.2503 0.1820 0.2333 0.2678
```

Заключение

Разработанные алгоритмы и способы автоматизации синтеза нечетких подсистем, представленные в работе, позволяют значительно сократить время и упростить процесс проектирования и отладки экспертной системы на базе нечеткой модели компетенций. Использование экспертной системы для оценки и формирования компетентности технического персонала промышленного предприятия в условиях инновационных изменений позволит повысить оперативность и объективность управленческих решений, а также создаст условия для эффективного прогнозирования инновационного развития предприятия.

ЛИТЕРАТУРА

1. Горькавый, М. А. Синтез нечеткой модели компетенций технического персонала промышленного предприятия/М. А. Горькавый, В. А. Соловьев // Информатика и системы управления. — 2009. — № 1 (25).
2. Заде, Л. А. Понятие лингвистической переменной и его применение к принятию приближенных решений/Л. А. Заде. — М.: Мир, 1976.
3. Штовба, С. Д. Проектирование нечетких систем средствами MATLAB/С. Д. Штовба. — М.: Горячая линия—Телеком, 2007.

ПРИМЕЧАНИЕ

1. Documentation for MathWorks Products, R2009b. Access: <http://www.mathworks.com/access/helpdesk/help/helpdesk.html>