

## Заключение

Доказанная теорема позволяет сделать вывод, что требование выполнения интервального условия для кодовых слов кода с КСП гарантирует отсутствие ложного приема при воздействии помех, количество импульсов которых, приходящихся на каждое кодовое слово, ниже некоторого порогового значения. Значит, использование кода с КСП на последней ступени каскадного кодирования позволит улучшить помехоустойчивость информационного канала.

## Литература

1. Peterson, W.W. Error-correcting codes / W. W. Peterson, E. J. Weldon. – Cambridge, Massachusetts and London, England: The MIT Press Publ., 1972. – 593 с.
2. Golomb, S.W. Digital Communications with Space Applications / S. W. Golomb. – New Jersey, Englewood Cliffs: Prentice-Hall Publ., 1964. – 272 с.
3. Светлов, М.С. Принципы обеспечения повышенной надежности дистанционного тестового контроля / М. С. Светлов, А. А. Львов, Д. В. Кленов // Материалы 13 международной конференции "Актуальные проблемы электронного приборостроения". – Саратов, 2016. – С. 403 – 408.
4. Львов, А.А. Повышение информационной надежности цифровых систем с QAM/COFDM модуляцией / А. А. Львов, М. С. Светлов, П. В. Мартынов // Известия Саратовского университета. Новая серия. Серия: "Математика. Механика. Информатика." – 2014. – Т. 14. – № 4. – Ч. 1. – С. 473-482.
5. Юргенсон, Р. И. Помехоустойчивость цифровых систем передачи телемеханической информации / Р. И. Юргенсон. – Л.: Энергия, 1971. – 250 с.
6. L'vov, A. A. Improvement of information reliability of digital systems with QAM/COFDM modulation / A. A. L'vov, M. S. Svetlov and P. V. Martynov // Proceedings of the 20th IMEKO TC4 Symposium, Benevento, Italy, 2014, pp. 479-484.
7. Львов, А. А. Самосинхронизация в информационных каналах с помехами большой интенсивности / А. А. Львов, М. С. Светлов, П. В. Мартынов // Радиотехника. – 2015. – №7. – С. 18-21.
8. Светлов, М. С. Некоторые свойства кода на основе кодового сигнального признака / М. С. Светлов, А. А. Львов, Д. В. Кленов, О. М. Балабан // Материалы 13 международной конференции "Перспективные информационные технологии" (ПИТ 2017). – Самара, 2017. – С. 992 – 995. ISBN: 978-5-93424-784-4.

УДК 004.827, 004.891

## ИНСТРУМЕНТАЛЬНАЯ СРЕДА ДЛЯ ПРЕДСТАВЛЕНИЯ И ОБРАБОТКИ НЕЧЕТКИХ ЗНАНИЙ И ЕЕ ПРИЛОЖЕНИЯ

А. Г. Костанян\*, Т. Г. Демирчян\*\*

Образовательный и исследовательский центр информационных технологий Ереванского государственного университета  
Ереван, Армения

\* armko@ysu.am, \*\* tatedem@gmail.com

**Аннотация** – Рассматривается созданная в образовательном и исследовательском центре информационных технологий ЕГУ инструментальная среда для представления и обработки нечетких знаний, на основе которой могут создаваться экспертные системы в конкретных областях. Система позволяет вводить и обрабатывать нечеткие

понятия с использованием аппарата нечеткой логики. Инструментальная среда реализована на языке логического программирования Prolog, адекватно отражающем суть решаемых задач. Представлены прототипы двух экспертных систем – диагностики заболеваний и обработки результатов обучения, разработанные с использованием возможностей инструментальной среды.

*Ключевые слова* – нечеткая экспертная система; нечеткая логика; база знаний.

## A TOOLKIT FOR REPRESENTATION AND PROCESSING OF FUZZY KNOWLEDGE AND ITS APPLICATIONS

A. H. Kostanyan \*, T. G. Demirchyan \*\*

Educational and Research Center of Information Technologies of  
Yerevan State University  
Yerevan, Armenia

\* armko@ysu.am, \*\* tatedem@gmail.com

**Abstract:** A toolkit for representation and processing of fuzzy knowledge created in the IT Educational and Research Center of YSU, is considered to developing expert systems in specific areas. The system allows enter and process the fuzzy concepts using the methods of fuzzy logic. The toolkit was implemented in logical programming language Prolog, adequately reflecting the essence of the tasks being solved. The prototypes for two expert systems (that is, diagnosis of diseases and processing of learning outcomes) that were developed using the capabilities of the toolkit, are presented.

**Keyword:** fuzzy expert system; fuzzy logic; knowledge base.

### Введение

В различных областях человеческой деятельности приходится принимать решения в условиях неполной информации. В настоящее время широкое распространение получают экспертные системы, которые используя методы нечеткой логики позволяют обрабатывать имеющуюся в наличии неполную информацию и делать определенные заключения. В частности, в работе [1] была предложена математическая структура для представления концепций и моделей с нечетким отношением связи между ними, ориентированная на создание приложений в области систем искусственного интеллекта.

В данной статье рассматривается созданная в образовательном и исследовательском центре информационных технологий ЕГУ инструментальная среда для представления и обработки нечетких знаний, на основе которой могут создаваться экспертные системы в прикладных областях. Система позволяет вводить и обрабатывать нечеткие понятия с использованием методов нечеткой логики. Для адекватного моделирования понятийного аппарата нечеткой логики, ядро инструментальной среды реализовано на базе языка логического программирования Prolog.

Для адаптации инструментальной среды к созданию экспертных систем в конкретных областях к ней должны подключаться специально разработанные модули приложений. В частности, разработаны прототипы мо-

дулей ориентированных на создание экспертных систем диагностики заболеваний и оценки конечных результатов программы обучения.

### Основные понятия и определения

Ниже приводится ряд понятий теории нечетких множеств и приближенных рассуждений [2-4], положенных в основу инструментальной среды.

**Нечеткие множества и отношения.** Пусть  $U$  универсальное множество. Нечеткое подмножество  $A$  множества  $U$  определяется функцией принадлежности  $\mu_A: U \rightarrow [0, 1]$ , сопоставляющей элементу  $x \in U$  степень  $\mu_A(x)$  принадлежности  $x$  множеству  $A$ . Нечеткое подмножество  $A$  множества  $U$  может быть представлено в виде множества пар

$$A = \{ x / \mu_A(x) \mid x \in U \}.$$

Например, можно определить нечеткое подмножество **SMALL** универсального множества  $\{1, 2, 3, 4, 5\}$ , полагая

$$\text{SMALL} = \{1/1, 2/0.75, 3/0.5, 4/0.25, 5/0\}.$$

В теории нечетких множеств основные теоретико-множественные операции (включение, объединение, пересечение, дополнение и т.д.) распространяются на нечеткие множества на основе принципа обобщения. Определяются нечеткие отношения, как нечеткие подмножества соответствующих декартовых произведений и операции, выполняемые над нечеткими отношениями.

**Лингвистическая переменная.** Важным понятием теории нечетких множеств является понятие лингвистической переменной, как переменной, значениями которой являются нечеткие подмножества данного универсального множества. Формально, лингвистическая переменная определяется как пятерка  $(N, T, U, G, M)$ , где

- $N$ —название переменной,
- $T$ —множество термов-значений переменной  $N$ , каждому из которых соответствует нечеткое подмножество универсального множества  $U$ ,
- $G$ —синтаксическое описание (обычно, в виде контекстно-свободной грамматики) множества  $T$ ,
- $M$ —семантическое правило, сопоставляющее элементам множества  $T$  значения в виде нечетких подмножеств универсального множества  $U$ .

Например, можно определить лингвистическую переменную множество термов-значений которой определяется правилом

$$T \rightarrow \text{SMALL} \mid \text{not}(T) \mid \text{very}(T).$$

Для вычисления семантических значений термов необходимо определить действия модификаторов **not** и **very**, что можно сделать, например, полагая

$$\mu_{\text{not}(A)}(x) = 1 - \mu_A(x), \quad \mu_{\text{very}(A)}(x) = (\mu_A(x))^2.$$

В результате, для определенного выше семантического значения базового термина **SMALL**, семантическим значением термина **very(SMALL)** будет нечеткое множество

$$\{1/1, 2/0.5625, 3/0.25, 4/0.0625, 5/0\}.$$

**Композиционное правило вывода и обобщенное правило modus ponens.** Пусть  $U$  и  $V$  два универсальных множества,  $A$  и  $F$  – нечеткие подмножества множеств  $U$  и  $U \times V$ , соответственно. Согласно композиционному правилу вывода, из нечетких множеств  $A$  и  $F$  следует нечеткое подмножество  $B = A \circ F$  универсального множества  $V$  такое, что для всех  $y \in V$

$$\mu_B(y) = \max\{\min\{\mu_A(x), \mu_F(x, y)\} \mid x \in U\}.$$

Пусть, например,  $U = V = \{1, 2, 3, 4, 5\}$ , **ALMOST-EQUAL** есть нечеткое бинарное отношение на множестве  $U \times V$ , определяемое правилом

- $\mu_{\text{ALMOST-EQUAL}}(x, y) = 1$ , если  $x = y$ ,
- $\mu_{\text{ALMOST-EQUAL}}(x, y) = 0.5$ , если  $\text{abs}(x - y) = 1$ ,
- $\mu_{\text{ALMOST-EQUAL}}(x, y) = 0$ , если  $\text{abs}(x - y) > 1$ .

Тогда результатом композиции **SMALL**  $\circ$  **ALMOST-EQUAL** будет нечеткое множество

$$\text{ALMOST-SMALL} = \{1/1, 2/0.75, 3/0.5, 4/0.5, 5/0.25\}.$$

Согласно классическому правилу modus ponens, из истинности высказываний  $A$  и  $A \rightarrow B$  следует истинность высказывания  $B$ . Пусть  $A^*$  некоторое высказывание, близкое к  $A$ . Правило modus ponens можно обобщить, полагая, что из истинности высказываний  $A^*$  и  $A \rightarrow B$  следует истинность некоторого высказывания  $B^*$ , близкого к  $B$ .

Если предположить, что  $A$  и  $B$  нечеткие подмножества универсальных множеств  $U$  и  $V$ , то импликацию  $A \rightarrow B \equiv (\neg A) \vee B$  можно рассматривать как нечеткое бинарное отношение на множестве  $U \times V$ , определяемое правилом

$$\mu_{A \rightarrow B}(x, y) = \max\{1 - \mu_A(x), \mu_B(y)\}.$$

Полагая, что  $A^*$  нечеткое подмножество универсального множества  $U$ , близкое к  $A$ , и рассматривая импликацию  $A \rightarrow B$  как нечеткое бинарное отношение на множестве  $U \times V$ , можно определить обобщенное правило modus ponens, продуцирующее из нечеткого множества  $A^*$  и нечеткого отношения  $A \rightarrow B$  нечеткое множество  $B^* = (A^*) \circ (A \rightarrow B)$ .

**Нечеткие экспертные системы.** Нечеткая экспертная система использует в качестве входных и выходных параметров лингвистические переменные, связанные между собой совокупностью высказываний вида

$$\text{IF } \langle \text{посылка} \rangle \text{ THEN } \langle \text{заключение} \rangle,$$

где  $\langle \text{посылка} \rangle$  определяется как пропозициональная формула, построенная из высказываний вида  $(x \text{ is } A)$ , где  $x$  есть входная лингвистическая переменная, а  $A$  терм-значение переменной  $x$ . Соответственно,  $\langle \text{заключение} \rangle$  определяется как пропозициональная формула, построенная

из высказываний вида ( $y$  is  $B$ ), где  $y$  – выходная лингвистическая переменная, а  $B$  – терм-значение этой переменной.

Для того, чтобы использовать нечеткую экспертную систему с числовыми значениями входных и выходных параметров, достаточно добавить фаззификатор ко входу и дефаззификатор к выходу системы (Рис. 1). Фаззификатор определяет степени достоверности посылок импликаций при заданных значениях входных параметров. С учетом этого, на основе правил нечеткой логики определяются степени достоверности заключений импликаций. Наконец, дефаззификатор определяет возможные значения выходных параметров, при которых обеспечивается нужная степень достоверности заключений импликаций.



Рис. 1. Структура нечеткой экспертной системы

### Инструментальная среда

Инструментальная среда (Рис. 2) состоит из интерфейса пользователя и ядра, включающего в себя совокупность средств, позволяющих, в частности, определять нечеткие множества и отношения, выполнять над ними стандартные операции, описывать лингвистические переменные, моделировать применение нечетких правил вывода. Для создания конкретного приложения (экспертной системы) необходимо разработать специализированный модуль, реализующий основные понятия предметной области приложения с использованием возможностей инструментальной среды. Реализация ядра инструментальной среды выполнена на языке логического программирования Prolog, являющегося удобным средством для разработки систем искусственного интеллекта [5].

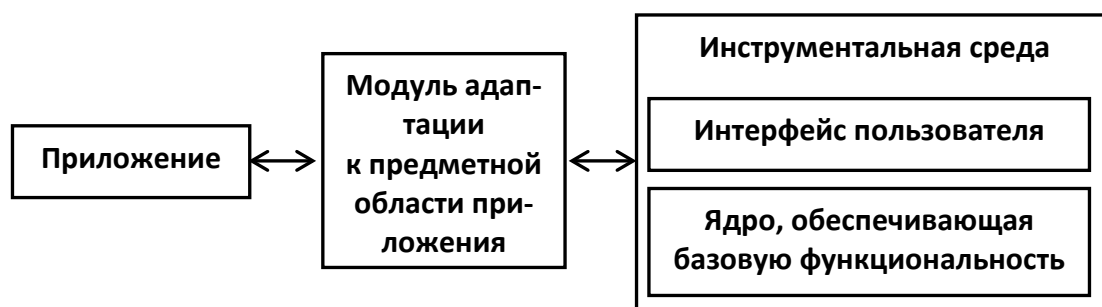


Рис. 2. Структура инструментальной среды

На базе инструментальной среды разработаны прототипы экспертных систем диагностики заболеваний и обработки конечных результатов учебной программы.

### Прототип экспертной системы диагностики заболеваний

В области диагностики заболеваний приходится делать умозаключения относительно состояния болезни на основе наличия определенных симптомов с учетом степени их выраженности. В прототипе экспертной системы для данного приложения каждой болезни сопоставлена выходная лингвистическая переменная, принимающая значения в множестве термов  $T_{out}=\{\text{BEGINNING, DEVELOPMENT, RECOVERY}\}$ , а каждому симптому – входная лингвистическая переменная, принимающая значения в множестве базовых термов  $T_{in}=\{\text{WEAK, MIDDLE, STRONG}\}$ , которые можно модифицировать с использованием модификаторов **not, very, almost** и т. д.

Система логического вывода состоит из продукций следующего вида:

**IF** <посылка> **THEN** <заключение>, где

- <посылка> есть пропозициональная формула, построенная из высказываний вида  $(s \text{ is } M)$ , где  $s$  есть название симптома,  $M$  – модифицированный элемент множества  $T_{in}$ , определяющий степень выраженности симптома  $s$ ,

- <заключение> есть утверждение вида  $(d \text{ is } R)$ , где  $d$  – название болезни, а  $R$  – элемент множества  $T_{out}$ , устанавливающий фазу развития болезни  $d$ .

В качестве примера рассмотрим систему правил, описывающих различные фазы гриппа.

#### Disease Flu.

- **Symptoms:**  
**headache, dizziness, weakness, sneeze, muscular\_pain, temperature, cough.**
- **Inferences:**
  - ✓ [headache is STRONG, temperature is WEAK, sneeze is WEAK, muscular\_pain is STRONG, weakness is STRONG] =>  
**flue is BEGINNING;**
  - ✓ [headache is STRONG, temperature is STRONG, sneeze is WEAK, muscular\_pain is STRONG, weakness is STRONG] =>  
**flue is DEVELOPMENT;**
  - ✓ [headache is WEAK, sneeze is WEAK, muscular\_pain is WEAK] =>  
**flue is RECOVERY;**

Следует отметить, что ряд симптомов (температура, кровяное давление, уровень сахара в крови и т.д.) имеют числовое выражение. Задачей фазификатора является определение степени соответствия числового значения симптома элементам модифицированного множества  $T_{in}$ . Например,

температуру **37.5** можно определить как низкую (**WEAK**) с достоверностью **0.2**, среднюю (**MIDDLE**) – с достоверностью **1**, высокую (**STRONG**) – с достоверностью **0.2**. Заключение системы о состоянии болезни не имеет числового выражения, что устраняет необходимость использования дефаззификатора.

Систему логического вывода можно усовершенствовать путем использования рекурсивных правил вывода, когда одна болезнь может быть симптомом для другой (т.е., соответствующая лингвистическая переменная является входной и выходной одновременно). Кроме того, можно структурировать правила системы путем определения синдромов как групп взаимосвязанных симптомов.

### Прототип экспертной системы обработки результатов обучения

Данное приложение ориентировано на определение степени овладения студентами конечными результатами программы обучения (*learning outcomes*) и их дальнейшую обработку.

Для определения степени овладения студентом *st* учебным результатом *lo* вводится входная лингвистическая переменная  $V[st, lo]$ , принимающая значения в множестве базовых термов  $T_{in}=\{\mathbf{POOR}, \mathbf{PARTLY}, \mathbf{GOOD}, \mathbf{EXCELLENT}\}$ . В качестве универсального множества выбирается диапазон вещественных чисел  $U=[\mathbf{8}, \mathbf{20}]$ , соответствующий шкале положительных оценок в двадцатибалльной системе. Элементы множества  $U$  интерпретируются как усредненные оценки, полученные данным студентом по предметам, соотношенным с данным результатом обучения.

Приложение ориентировано на удовлетворение требований работодателя, в связи с чем вводится выходная лингвистическая переменная  $W[st, emp]$ , принимающая значения в множестве  $T_{out}=\{\mathbf{NOT}, \mathbf{PARTLY}, \mathbf{COMPLETELY}\}$  и определяющая степень соответствия студента *st* требованиям работодателя *emp*.

Система логического вывода состоит из продукций следующего вида:

IF ( $V[X, lo_1]$  is  $M_1$ ), ..., ( $V[X, lo_k]$  is  $M_k$ ) THEN ( $W[X, emp]$  is  $R$ ), где  $X$ –переменная, обозначающая студента;  $lo_1, \dots, lo_k$ –названия результатов обучения;  $M_1, \dots, M_k$ –элементы множества  $T_{in}$ ;  $R$ –элемент множества  $T_{out}$ .

Фаззификатор системы для каждого студента *st* и каждого результата обучения *lo* вычисляет числовое значение параметра  $V[st, lo]$  (т.е., среднюю оценку студента *st* по предметам, соотношенным с результатом обучения *lo*), после чего определяет степень соответствия вычисленного значения каждому из элементов множества  $T_{in}$ . Например, среднюю оценку **16.5** можно считать хорошим уровнем овладения данным результатом обучения (**GOOD**) с достоверностью **1**, и совершенным (**EXCELLENT**) – с достоверностью **0.25**. Заключение системы о степени соответствия студента требованиям работодателя не имеет числового выражения, так что здесь также нет необходимости в использовании дефаззификатора.

## **Заключение**

В работе представлены структура и функциональные возможности реализованной на языке Prolog инструментальной среды для создания нечетких экспертных систем в прикладных областях. Описаны прототипы экспертных систем диагностики заболеваний и обработки результатов обучения.

## **Литература**

1. Kostanyan, L. Soghomonyan. A framework for conceptual knowledge representation // Proc. of 4<sup>th</sup> Annual Scientific Conference of RAU, Yerevan, Armenia, 2010, pp. 171-176.
2. L. A. Zadeh. The Concept of a Linguistic Variable and its Application to Approximate Reasoning // Information Sciences 8, 1975, pp. 199-249.
3. Г. Э. Яхьяева. Нечёткие Множества и нейронные сети, 2-е издание // М.: Интернет-Университет Информационных технологий, 2008, 316 стр.
4. Круглов В.В., Дли М.И. Интеллектуальные информационные системы: компьютерная поддержка систем нечеткой логики и нечеткого вывода // М.: Физматлит, 2002.
5. Ivan Bratko. Prolog Programming for Artificial Intelligence, 4<sup>th</sup> edition // Pearson Education Canada, 2011.

УДК 621.318:004.942

## **ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РАСЧЕТА ГЕОМЕТРИЧЕСКИХ ПАРАМЕТРОВ ПОЛОСКОВЫХ КАТУШЕК ИНДУКТИВНОСТИ**

**В.С. Кожухов\*, А.Г. Улданов\*\*, Е.Р. Кожанова\*\*\***

\* Саратовский государственный технический университет имени Гагарина Ю.А.  
Институт электронной техники и машиностроения  
Саратов, Россия  
kozuhov.slava@yandex.ru

\*\* Саратовский государственный технический университет имени Гагарина Ю.А.  
Институт электронной техники и машиностроения  
Саратов, Россия  
azikuldan@mail.ru

\*\*\* Саратовский государственный технический университет имени Гагарина Ю.А.  
Институт электронной техники и машиностроения  
Саратов, Россия  
Ljubimzh@yandex.ru

**Аннотация** – В статье описано разработанное программное обеспечение для расчета геометрических параметров пяти разновидностей полосковых катушек индуктивности. Может быть использовано для проектирования пленочных и гибридных интегральных микросхем (ПИМС, ГИМС), а в учебном процессе, например: при проведении лабораторной работы по дисциплине «Пассивные микросхемные устройства СВЧ».

**Ключевые слова** – программное обеспечение; интерфейс; полосковая катушка индуктивности; геометрические параметры.