

## **ГИБРИДНЫЕ МОДЕЛИ ПРИНЯТИЯ РЕШЕНИЙ С ИСПОЛЬЗОВАНИЕМ ИММУННОГО ПОДХОДА**

**Кораблев Н., Фомичев А., Соловьев Д., Чуприна А.**

*Современные системы поддержки принятия решений (СППР) характеризуются обработкой больших объемов информации в условиях неопределенности, что требует применения эффективных методов и моделей, использующих различные интеллектуальные технологии параллельной обработки информации. В статье предлагаются гибридные модели принятия решений (МПР), представленные как на основе нейронной сети (НС), так и в виде правил нечеткого вывода, обучение и адаптация которых осуществляются на высокопроизводительных системах с использованием иммунных моделей в зависимости от изменения характеристик окружающей среды и свойств объекта принятия решений (ОПР). Процесс обучения и адаптации НС состоит в настройке и коррекции параметров сети, а также числа нейронов в скрытых слоях и связей между ними. Процесс обучения и адаптации моделей нечеткого вывода (МНВ) состоит в настройке формы и параметров функций принадлежности, а также параметров и структуры модели (числа правил вывода).*

*Ключевые слова. Принятия решений, гибридная модель, обучение, адаптация, нейронная сеть, нечеткий вывод, искусственные иммунные системы.*

### **ВВЕДЕНИЕ**

В настоящее время актуальной задачей является разработка методов, моделей и информационных технологий принятия решений для управления сложными объектами. В условиях неопределенности при принятии решений, а также изменения свойств окружающей среды и состояния ОПР, учесть которые априори невозможно, необходимо использовать интеллектуальные подходы к созданию МПР. Существуют различные подходы к созданию таких моделей, которые используют аппарат теории нечетких множеств, искусственные нейронные сети, генетические алгоритмы, искусственные иммунные системы (ИИС), байесовские сети, экспертные системы и др. [1-9]. Перспективным является разработка гибридных МПР, использующих преимущества каждой из технологий искусственного интеллекта.

Среди МПР особое место занимают модели на основе НС и модели нечеткого вывода (МНВ), которые широко применяются в условиях неопределенности, в случае нелинейности объекта принятия

решений (ОПР), а также в системах, основанных на использовании экспертной информации [10-12]. Основные трудности при использовании таких моделей для решения задачи принятия решений связаны с априорным определением их компонентов. При наличии набора неструктурированных данных определение этих компонентов представляет собой сложную проблему. Кроме того, одной из задач при формировании МПР является устранение их избыточности. Поэтому специфика задачи принятия решений приводит к необходимости устранения этих недостатков.

Как правило, СППР функционируют в условиях априорной и текущей неопределенности. Поэтому необходимо использовать МПР, которые будут эволюционировать в условиях изменения внешней среды и свойств ОПР, т.е. адаптироваться под эти изменения [1]. Параметры адаптивных моделей должны настраиваться путем их оптимизации в соответствии с некоторым критерием, формируемым по данным из обучающей выборки. Решение подобной задачи оптимизации представляет собой трудную задачу по ряду причин. Во-первых, это большая размерность вектора параметров модели. Во-вторых, большая размерность пространства поиска тесно связана с проблемой многоэкстремальности целевой функции. Кроме того, характер целевой функции полностью зависит от данных обучающей выборки, которые меняются с каждым новым моделируемым ОПР.

Для обучения и адаптации МПР может использоваться эволюционный подход. В [10] рассмотрены эволюционные методы формирования нечетких баз правил и их редукции на основе генетического алгоритма. Однако ГА не всегда эффективны при решении задачи мультимодальной оптимизации. В [11] исследованы нейросетевые и гибридные алгоритмы обучения для повышения эффективности МПР в сложных средах. Развитие эволюционных вычислений ведется по пути создания новых методов, использующих возможности динамического изменения параметров алгоритмов оптимизации и различные варианты распараллеливания процессов обработки нечеткой информации. Таким требованиям удовлетворяют ИИС [13]. В [14] рассмотрено применение ИИС в нечетких системах классификации, в частности для генерации базы нечетких правил. В [15] предложена гибридная МПР на основе нейронной сети, обучение и эволюция которой осуществляются с использованием иммунных моделей клонального отбора и иммунной сети. В [16] предложена МПР на основе нечеткой логики, эволюция которой осуществляется на основе иммунного подхода.

В данной работе рассматриваются гибридные МПР, представленные в виде НС и МНВ, обучение и адаптация которых осуществляется на основе использования ИИС. Это позволяет в режиме реального времени корректировать как структуру, так и параметры МПР

в зависимости от изменения характеристик окружающей среды и свойств ОПР.

### ПОСТАНОВКА ЗАДАЧИ

Пусть  $X$  – множество наборов признаков (факторов) описания ОПР,  $Y$  – множество состояний (классов) ОПР, а функционирование МПР описывается как отображение пространства факторов  $X$  на пространство состояний  $Y$  в виде соотношения:

$$F : X \rightarrow Y, X \subset R^m, Y \subset R^n. \quad (1)$$

МПР должна иметь аналитическую зависимость соотношения (1), согласно которой по известным значениям вектора входных факторов  $X^m = \{x_1, x_2, \dots, x_m\} \subset X$  можно оценить значения выходного вектора состояний (классов)  $Y^n = \{y_1, y_2, \dots, y_n\} \subset Y$ .

Для обучения и адаптации МПР строится целевая функция:

$$\Phi(P) = \sqrt{\frac{1}{n \cdot m} \sum_{i=1}^n \sum_{j=1}^m [F_i(x_j, P) - y_i]^2} = \min_P, \quad (2)$$

минимизирующая среднеквадратическое отклонение между фактическими значениями выходной переменной  $y_i$  и полученной на основе МПР оценкой  $F_i(x_j, P)$ . Здесь  $x_j$  – входные переменные,  $P$  – параметры модели, состав которых определяется исходя из выбранного способа описания зависимости (1) и алгоритма обучения.

Ставится задача разработки алгоритмов обучения и адаптации МПР на основе ИИС. С этой целью по данным из обучающей выборки необходимо оптимизировать параметры и структуру МПР в соответствии с критерием (2).

## НЕЙРОСЕТЕВАЯ МОДЕЛЬ ПРИНЯТИЯ РЕШЕНИЙ С ИММУННЫМ ОБУЧЕНИЕМ

### 1. Описание МПР в виде нейронной сети

Для описания связи между входными и выходными переменными в (1) используется МПР в виде многослойной НС. Используется трехслойный персептрон, имеющий входной слой, содержащий число нейронов, равное числу входных признаков, один промежуточный слой и выходной слой, содержащий число нейронов, равное числу классов принимаемых решений. В промежуточном слое НС в качестве функций активации используются сигмоидальные функции:

$$z_m = f(u_m) = \frac{1}{1 + e^{-\lambda_m u_m}}, \quad u_m = \sum_{n=1}^N w_{n,m} x_n + w_{o,m}, \quad (3)$$

где  $z_m$ ,  $m = \overline{1, M}$  – выходной сигнал  $m$ -го нейрона промежуточного

слоя, состоящего из  $M$  нейронов, имеющих  $N$  входов;  $x_n, n = \overline{1, N}$  –  $n$ -я компонента входного вектора признаков;  $w_{n,m}$  – весовой коэффициент  $n$ -го входного признака  $x_n$ , поступающего на вход  $m$ -го нейрона промежуточного слоя;  $w_{o,m}$  – значение смещения;  $\lambda_m$  – коэффициент, определяющий крутизну функции активации  $f(u_m)$ . Нейроны выходного слоя имеют пороговую функцию активации и используются для принятия решения о принадлежности анализируемой СППР к определенному классу решений:

$$y_k = \varphi\left(\sum_{m=1}^M v_{m,k} \cdot z_m + v_{o,k}\right) = \begin{cases} 1, & \text{если } y_k > 0, \\ 0, & \text{если } y_k \leq 0, \end{cases} \quad k = \overline{1, K} \quad (4)$$

где  $v_{m,k}$  – весовые коэффициенты;  $v_{o,k}$  – смещение;  $K$  – число выходов НС.

## 2. Иммунное обучение и адаптация нейросетевой МПР

Существующие методы обучения НС являются трудоемкими. Основной их чертой является отсутствие возможности модификации структуры и параметров НС. Для реализации этого предлагается выполнять обучение НС на основе ИИС. Задача обучения НС рассматривается как требование минимизации целевой функции (2), в которой в векторе параметров  $P$  НС используются параметры  $w_{n,m}, w_{o,m}, v_{m,k}, v_{o,k}$  и  $\lambda_m, n = \overline{1, N}, m = \overline{1, M}, k = \overline{1, K}$ .

Основной идеей обучения НС с использованием ИИС является представление решаемой задачи в виде антигена, а возможные ее решения – в виде антител [13]. Сформируем популяцию антигенов  $Ag_i = \langle x_1^i, x_2^i, \dots, x_n^i, y_1^i, y_2^i, \dots, y_k^i \rangle, i = \overline{1, S}$ , где  $x_1^i, x_2^i, \dots, x_n^i$  – входные переменные, а  $y_1^i, y_2^i, \dots, y_k^i$  – выходные переменные модели для  $i$ -го примера обучающей выборки. В виде популяции антигенов выступает обучающая выборка относительно входных  $x_n, n = \overline{1, N}$  и выходных  $y_k, k = \overline{1, K}$  переменных. В качестве антител используются векторы настраиваемых параметров. В одном антителе кодируются все настраиваемые параметры НС:  $w_{n,m}, w_{o,m}, v_{m,k}, v_{o,k}$  и  $\lambda_m, n = \overline{1, N}, m = \overline{1, M}, k = \overline{1, K}$ .

Для решения задачи используется модель кодирования настраиваемых параметров в виде адаптивного структурированного мультиантитела (рис. 1) [14], состоящего из двух частей, каждая из

которых может обрабатываться независимо друг от друга.

$w_{1,1}, \dots, w_{1,M}, \dots, w_{N,1}, \dots, w_{N,M}; w_{0,1}, \dots, w_{0,M}; \lambda_1, \dots, \lambda_M$	$v_{1,1}$	...	$v_{M,K}$	$v_{0,M}$
$ab_0$	$ab_1$	...	$ab_M$	$ab_{M+1}$
Часть 1	Часть 2			

Рис. 1 – Структура мультиантитела

Популяция мультиантител представлена в виде  $mAb = \{mAb_1, mAb_2, \dots, mAb_N\}$ , где  $mAb_i = \{ab_0, ab_1, ab_2, \dots, ab_K\}$ ,  $i = \overline{1, S}$  –  $i$ -е адаптивное мультиантитело, длина которого  $L$  изменяется в процессе выполнения иммунного алгоритма;  $S$  – размер популяции мультиантител. Каждое мультиантитело популяции характеризуется полным множеством настраиваемых параметров НС. В части 1 мультиантитела закодированы весовые коэффициенты  $w_{n,m}$ , значения смещений  $w_{o,m}$  и коэффициенты функции активации  $\lambda_m$ . В части 2 закодированы коэффициенты  $v_{m,k}$  и смещения  $v_{o,k}$  выходного слоя НС. Вторая часть является адаптивной, т.к. в процессе обучения, помимо оптимизации коэффициентов, изменяется и их общее количество, т.е. выполняется структурный синтез. Это приводит к изменению количества нейронов в скрытом слое НС и связей между ними. Структурированный способ формирования мультиантитела позволяет повысить эффективность иммунного алгоритма за счет отдельного применения иммунных операторов к каждой из частей. Вычисление аффинности выполняется для мультиантитела в целом, оперируя обоими составляющими.

В качестве вычислительной модели ИИС используются принципы клонального отбора и сетевого взаимодействия [13]. При клональном отборе антитело распознает антиген, клонируется, и полученные клоны подвергаются мутации. Если в результате мутации аффинность клона улучшается, то соответствующее антитело заменяется своим клоном, т.е. осуществляется клональный отбор. При использовании модели иммунной сети антитела взаимодействуют не только с антигенами, но и с другими антителами. Это обеспечивает эффект взаимной стимуляции и супрессии, что приводит к сокращению числа нейронов в промежуточном слое НС.

Алгоритм обучения, выполняемый в режиме off-line, представляет собой итерационную процедуру последовательной идентификации наблюдений из обучающей выборки. Антигены представляют примеры обучающей выборки. Каждое антитело кодирует одно из возможных решений, а количество антител в популяции соответствует количеству

примеров в обучающей выборке. Алгоритм обучения НС представляет собой следующую последовательность шагов:

1. Инициализация начальной популяции мультиантител  $mAb$  – выполняется случайным образом.

2. Вычисление аффинности  $Aff_{mAb-Ag}$  для каждого мультиантитела. Необходимо подставить параметры, закодированные в мультиантителе, в НС. На входы НС подаются входные признаки  $x_n$  и вычисляются значения выходных переменных  $y_k$ . Аффинность мультиантитела  $mAb$  с антигеном  $Ag$  вычисляется в виде:

$$Aff_{mAb-Ag} = (1 + d_{mAb-Ag})^{-1}, \quad (5)$$

где  $d_{mAb-Ag}$  – расстояние Хэмминга между полученными значениями выходов НС  $y_k$ ,  $k = \overline{1, K}$  и желаемыми  $y_k^r$  для всех  $S$  антигенов популяции  $Ag$ :

$$d_{mAb-Ag} = \sum_{k=1}^K y_k, \quad \text{где } y_k = \begin{cases} 1, & \text{если } y_k \neq y_k^r, \\ 0, & \text{если } y_k = y_k^r. \end{cases} \quad (6)$$

3. Клонирование мультиантител пропорционально их аффинности и формирование популяции клонов  $Cl$ . Параметрами оператора клонирования являются количество антител для клонирования  $g$  и кратность клонирования мультиантитела  $N_{Cl}$ . Используется фиксированное значение параметра  $g$ . Кратность клонирования мультиантитела  $N_{Cl}$  регулируется в процессе работы иммунного алгоритма в зависимости от аффинности мультиантитела по соотношению:

$$N_{Cl} = \alpha * N_{Cl\_min} + (1 - \alpha) * N_{Cl\_max}, \quad (7)$$

где  $\alpha = \frac{Aff_{best} - Aff_{mAb-Ag}}{Aff_{best}}$ ;  $N_{Cl\_min}$  и  $N_{Cl\_max}$  – минимальная и максимальная кратность клонирования мультиантитела;  $Aff_{best}$  – лучшее значение аффинности в текущем поколении.

4. Мутация клонов обратно пропорционально аффинности мультиантитела и формирование популяции мутированных клонов  $MC$ . Мутация выбранных параметров мультиантитела  $mAb$  выполняется путем добавления гауссовского шума:

$$mAb_{i+1} = mAb_i + N(0, \sigma_i). \quad (8)$$

Для изменения среднеквадратического отклонения случайной величины  $\sigma_i$  используется соотношение:

$$\sigma_{i+1} = \sigma_i \frac{Aff_{best} - Aff_{mAb-Ag}}{Aff_{best} - Aff_{worst}}, \quad (9)$$

где  $Aff_{worst}$  – худшее значение аффинности в текущем поколении.

5. Вычисление аффинности популяции мутированных клонов  $MC$  в соответствии с (5). Если в результате мутации аффинность улучшилась, замена клонами соответствующих мультиантител в популяции  $mAb$ .

6. Вычисление аффинности антител внутри части 2 мультиантитела. Суппрессия антител, аффинность которых больше заданного порога  $\delta_{net}$ . Вычисление аффинности выполняется в соответствии с выражением:

$$mAb_{i+1} = mAb_i + N(0, \sigma_i), \quad (10)$$

где  $d_{ab_i-ab_j}$  – расстояние между  $i$ -м и  $j$ -м антителами части 2 мультиантитела:

$$d_{ab_i-ab_j} = \|ab_i - ab_j\| = \sqrt{\sum_{m=0}^M (v_{m,i} - v_{m,j})^2}, \quad i, j = \overline{1, K}. \quad (11)$$

Выполнение суппрессии путем удаления антител  $ab_i$  с аффинностью, большей заданного порога  $\delta_{net}$ , позволяет уменьшить количество нейронов и связей между ними в скрытом слое и устранить их избыточность.

7. Проверка критерия останова. В качестве критерия останова используется либо достижение заданного порога аффинности, либо достижение заданного количества поколений работы алгоритма.

Результатом работы алгоритма будет мультиантитело с лучшей по популяции аффинностью, определяющее структуру НС и содержащее настроенные ее параметры. Шаги 1-5 алгоритма соответствуют принципу клонального отбора. Алгоритм на данных этапах работает с обеими частями мультиантитела. Шаг 6 соответствует принципу сетевого взаимодействия. Если ранее мультиантитело обрабатывалось как обычное антитело, то на данном шаге работа выполняется только с частью 2 мультиантитела, состоящей из отдельных антител.

В процессе функционирования СППР возможно изменение характеристик окружающей среды и свойств ОПР. В этом случае необходимо адаптировать МПР под эти изменения, т.е. осуществлять ее эволюцию. Предлагается для этого использовать процедуру обучения НС на основе ИИС, но в качестве обучающей выборки на вход будут поступать реальные значения входных признаков МПР, функционирующей в режиме on-line. Используется равномерное распределение нейросетевой обработки данных, основанное на

динамическом перераспределении наборов нейронов НС между вычислителями в зависимости от объема обрабатываемых данных, что позволяет существенно сократить время обучения и функционирования НС, а также уменьшить на порядок вычислительную сложность [17]. Предлагаемый подход позволит корректировать в реальном времени структуру и параметры МПР.

### 3. Экспериментальные исследования нейросетевой МПР

В качестве тестовой рассматривалась задача проектирования и анализа локальной компьютерной сети (ЛКС), которая является примером трудно формализуемой задачи, для решения которой необходим интегрированный подход. При проектировании ЛКС необходимо определить выходные параметры, которым должна удовлетворять сеть, и начальные условия (входные параметры), которые задаются перед процессом проектирования.

Анализ ЛКС позволил выделить входные признаки, наиболее существенные из которых следующие:  $x_1$  – стоимость сети;  $x_2$  – количество и расположение пользователей;  $x_3$  – простота установки и изменения конфигурации сети;  $x_4$  – пропускная способность сети;  $x_5$  – надежность сети;  $x_6$  – безопасность сети;  $x_7$  – возможность расширения сети.

Исходя из требований проектирования ЛКС, имеем следующие основные выходные параметры: 1) сетевая операционная система (ОС); 2) топология сети; 3) сетевая технология.

Варианты выбора ОС: 1) Novell NetWare; 2) Microsoft Windows (NT, Vista, 7, 8, 10); 3) UNIX системы (Solaris, FreeBSD); 4) GNU/Linux системы; 5) IOS; 6) ZyNOS компании ZyXEL.

Варианты выбора топологии сети: 1) звезда; 2) шина; 3) кольцо; 4) дерево; 5) полносвязная; 6) ячеистая; 7) смешанная.

Варианты выбора сетевой технологии: 1) Fast Ethernet; 2) Token Ring; 3) FDDI.

Реализация данной задачи методом простого перебора вариантов неприменима, поскольку комбинаторная емкость перебора составляет несколько порядков. Более того, большая часть комбинаций вариантов никогда не будет реализована. Поэтому реализация была выполнена с применением эволюционных НС с использованием ИИС. Задача условно была разбита на три параллельные задачи, каждая из которых решает проблему по одному из выходных параметров. При этом каждый из выходных параметров может определяться значениями не всех входных, а только нескольких из них.

Для каждой из трех задач была создана своя НС, содержащая

определенное число входов и выходов. Так, для выбора ОС – это 7 и 6, для выбора топологии сети – 6 и 7, для выбора сетевой технологии – 6 и 3 соответственно. Количество скрытых слоев – один, содержащий 15 и более нейронов. Структура спроектированной НС для определения ОС приведена на рис. 2.

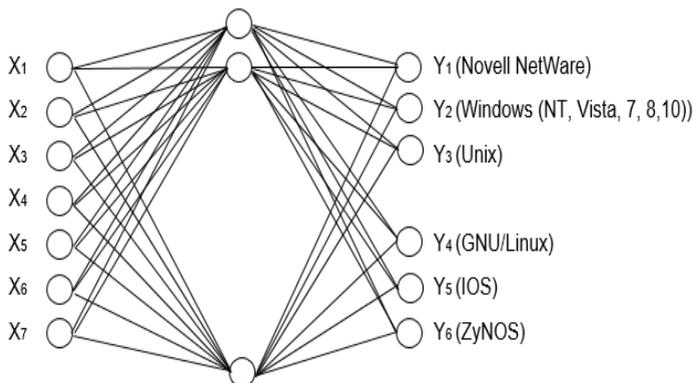


Рис. 2 – Структура нейронной сети для определения ОС ЛКС

Для моделирования полученных результатов использована компьютерная сеть Fast Ethernet с физической топологией «звезда» со скоростью передачи 100 Мбит, которая состоит из 10 четырехядерных вычислителей Intel Core 2 Quad CPU Q8200 @2.33GHz с видеокартой GeForce GTX 460. Распараллеливание выполнено с помощью технологий OpenMP и MS MPI. Для анализа процедуры параллельного и распределенного обучения НС на многопроцессорной системе проведен ряд экспериментов. Анализ показал, что технология MPI позволяет ускорить распределенное обучение НС в 11 раз для топологии передачи данных «звезда». Однако при значительном увеличении количества процессоров в системе растет время на передачу данных между ними. Поэтому, при достижении определенного прироста производительности, дальнейшее увеличение количества процессоров в системе дает обратный эффект – производительность начинает падать. Стандарт OpenMP позволяет ускорить параллельное обучение НС до 7 раз. Ускоренная обработка данных позволяет функционировать интеллектуальной СППР в реальном времени.

Таким образом, проведенные исследования показали, что многослойные модульные нейронные сети персептронного типа с иммунным обучением целесообразно использовать для прогнозирования возможных вариантов решений при выборе параметров ЛКС, при этом наблюдается высокая устойчивость получаемых результатов.

# НЕЧЕТКАЯ МОДЕЛЬ ПРИНЯТИЯ РЕШЕНИЙ С ИММУННЫМ ОБУЧЕНИЕМ

## 1. Описание МПР в виде правил нечеткого вывода

Для описания зависимости (1) будем использовать модель нечеткого вывода Такаги-Сугено первого порядка с правилами вида:

$$R_i : \text{if } x_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } x_m \text{ is } A_{im}, \text{ then} \\ y_i = k_{i0} + k_{i1}x_1 + \dots + k_{im}x_m, \text{ is class}_i, i = \overline{1, n}, \quad (12)$$

где  $A_{ij}, i = \overline{1, n}, j = \overline{1, m}$ , – нечеткие множества и соответствующие им функции принадлежности (ФП), построенные в пространстве значений входных переменных  $x_j$ ,  $k_{ij}$  – коэффициенты аргументов функции, значение выходной переменной  $y_i$  определяет класс системы. В качестве ФП используются гауссовы функции:

$$\mu(x) = \exp\left[-\left(\frac{x-c}{\sigma}\right)^2\right], \quad (13)$$

где  $c$  – центр нечеткого множества;  $\sigma$  отвечает за крутизну функции. Для обучения и адаптации нечеткой МПР (11) на основе ИИС необходимо оптимизировать параметры ФП, а также параметры и структуру этой модели в соответствии с критерием (2).

## 2. Иммунное обучение и адаптация нечеткой МПР

Представим адаптивную систему нечеткого вывода в виде кортежа:

$$AFIS = \{\{X_i, y_i\}, RB, DB, I, G(RB), L(DB), F\}, \quad (14)$$

где  $\{X_i, y_i\}, i = \overline{1, n}$  обучающая выборка;  $RB$  – база правил;  $DB$  – база данных;  $I$  – механизм нечеткого логического вывода;  $G(RB)$  – генерация и оптимизация базы правил;  $L(DB)$  – генерация и оптимизация базы данных;  $F$  – функция, оценивающая эффективность нечеткой модели.

Основная трудность использования данной системы заключается в необходимости структурной и параметрической адаптации – поиске оптимального набора параметров термов лингвистических переменных и коэффициентов нечетких правил, а также структуры множества правил нечеткого вывода [14]. Алгоритм формирования нечеткой базы знаний представляет собой итерационную процедуру последовательной идентификации наблюдений из обучающей выборки с помощью ИИС. Антигены представляют собой примеры обучающей выборки. Каждое антитело кодирует одно правило базы знаний. Количество антител в популяции (правил в базе знаний) соответствует количеству примеров в обучающей выборке. Правила кодируются вещественными числами,

представляющими собой индексы нечетких множеств входных переменных и коэффициенты нечетких правил:

$$Ab_i = \langle IFS_{i1}, IFS_{i2}, \dots, IFS_{im}, k_{i0}, k_{i1}, \dots, k_{im} \rangle, \quad i = \overline{1, n},$$

где  $IFS_{i1}, IFS_{i2}, \dots, IFS_{im}$  – индексы нечетких множеств для  $m$  входных переменных;  $k_{i0}, k_{i1}, \dots, k_{im}$  – коэффициенты нечетких правил (11).

Аффинность антител популяции вычисляется по результатам нечеткого вывода в соответствии с выражением:

$$Af = (1 + \Phi(P))^{-1}, \quad (15)$$

где  $\Phi(P)$  – эвклидово расстояние (2). В результате работы иммунного алгоритма лучшие по популяции антитела формируют популяцию клеток памяти, которая представляет множество нечетких правил вывода вида (11):

Полное множество правил (12) для реальных задач может быть достаточно велико. Для сокращения мощности множества правил (12) необходимо выделить некоторое минимальное подмножество, исключив избыточные и противоречивые правила. Данную задачу в предлагаемом алгоритме выполняет иммунный оператор супрессии [13], который обеспечивает сжатие популяции путем связывания антител в зависимости от значения их аффинности относительно других антител популяции. Антитела с похожей аффинностью удаляются из популяции в соответствии с установленным порогом сжатия сети, что приводит к уменьшению числа правил в базе знаний адаптивной системы нечеткого вывода.

Алгоритм структурной адаптации можно представить в виде следующей последовательности шагов:

1. Генерация начальной популяции антител  $Ab$ .
2. Цикл для каждого антигена  $Ag_i$ :

2.1. В МНВ подставляются параметры ФП и правило вывода с коэффициентами, взятыми из антитела. Вычисляются аффинности антител  $Ab$  к антигену  $Ag_i$ . Выбор  $n$  лучших антител для клонирования  $N$  раз.

2.2. Клонирование выбранных антител, формирование популяции клонов  $C$ .

2.3. Мутация популяции клонов  $C$  и формирование популяции клонированных и мутированных антител  $C^*$ .

2.4. Вычисление аффинности популяции  $C^*$  к антигену  $Ag_i$ . Редактирование популяции антител  $Ab_i$  и заполнение популяции клеток памяти  $M$ . Супрессия антител  $Ab$  внутри популяции.

- 2.5. Замена  $d$  худших антител новыми в популяции антител  $Ab$ .

3. Проверка критерия останова. Если не достигнут – переход к шагу 2, в противном случае – переход к шагу 4.

4. Конец.

Результат – популяция клеток памяти  $M$ , которая представляет собой набор правил вывода с вычисленными коэффициентами нечетких правил.

Параметрическая адаптация включает в себя процедуру оптимизации значений параметров ФП, целью которой является поиск такой модели, с таким набором параметров ФП, для которой качество принятия решений будет наилучшим. В роли антигена выступает обучающая выборка; в роли антител – параметры ФП вида (13) для входных переменных. Антитело представляет собой строку фиксированной длины:

$$Ab = \langle c_{11}, \dots, c_{1m}, \dots, c_{n1}, \dots, c_{nm}, \sigma_{11}, \dots, \sigma_{1m}, \dots, \sigma_{n1}, \dots, \sigma_{nm} \rangle,$$

где  $c_{ij}, \sigma_{ij}$ ,  $i = \overline{1, n}$ ;  $j = \overline{1, m}$  – параметры гауссовых ФП вида (12) для  $n$  входных переменных, каждая из которых имеет  $m$  термов. Каждая из ФП задана на универсуме  $X = [x_{min}, x_{max}]$ , в качестве которого выбран замкнутый интервал действительных чисел. Исходный вектор параметров ФП генерируется случайным образом. Предлагается вещественное кодирование антител, что позволяет повысить точность найденных решений.

На вход адаптивной МНВ подается множество значений входных переменных из обучающей выборки. На основании параметров ФП, взятых из антител, и сгенерированной на этапе структурной адаптации базы правил, выполняется нечеткий вывод. По результатам нечеткого вывода вычисляется аффинность каждого антитела со всеми антигенами популяции в соответствии с (15). Поиск минимума функции (2) ведется в допустимом множестве изменения входных переменных  $x_i$ :  $x_i^{min} \leq x_i \leq x_i^{max}$ ,  $i = \overline{1, m}$ . Антитела с лучшей аффинностью подвергаются клонированию и мутации. Антитела с худшим значением аффинности удаляются из популяции и заменяются новыми. Работа иммунного алгоритма прекращается при достижении критерия останова. База данных адаптивной системы нечеткого вывода формируется по лучшему антителу популяции.

Алгоритм параметрической адаптации можно представить в виде такой же последовательности шагов, как и алгоритм структурной адаптации, только в нем не используется иммунный оператор супрессии антител внутри популяции. Результат – антитело с лучшей аффинностью, содержащее настроенные параметры ФП.

Обобщенный алгоритм обучения адаптивной МНВ представляет собой итерационную процедуру, в которой организовано взаимодействие

процедур параметрической и структурной адаптации. На подготовительном этапе для каждого ОНР формируется обучающая выборка  $(x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)})$ ,  $k = \overline{1, K}$ , где  $x_1^{(k)}, x_2^{(k)}, \dots, x_m^{(k)}, y^{(k)}$  – значения входных переменных  $x_1, x_2, \dots, x_m$  и выходной переменной  $y$  в  $k$ -ом примере;  $K$  – общее количество примеров в обучающей выборке. На основании известных минимальных и максимальных значений каждой входной переменной область определения этих переменных разбивается на отрезки, на каждом из которых задается гауссова ФП. На этапе структурной адаптации формируется  $N$  наборов нечетких правил, в каждом поколении иммунного алгоритма происходит модификация популяции в соответствии с выбранной схемой применения иммунных операторов, вычисление приспособленности каждого антитела, ранжирование антител по степени аффинности в популяции и формирование МНВ, на основе которой выполняется параметрическая адаптация. Иммунный алгоритм параметрической адаптации итерационно повторяется заданное число раз. В конце из полученной популяции выбирается наиболее приспособленное антитело, которое является результатом адаптации МНВ.

### 3. Экспериментальные исследования нечеткой МНР

Для оценки эффективности предложенных адаптивных МНВ были проведены экспериментальные исследования для двух тестовых функций (ТФ):

$$F_1(x_1, x_2) = x_1^2 + x_2^2, \quad (16)$$

$$F_2(x_1, x_2) = 3(1 - x_1)^2 \exp(-x_1^2 - (x_2 + 1)^2) - 10 \left( \frac{x_1}{5} - x_1^3 - x_2^5 \right) \exp(-x_1^2 - x_2^2) - \frac{1}{3} \exp(-(x_1 + 1)^2 - x_2^2), \quad (17)$$

которые исследовались в диапазоне входных переменных  $x_1, x_2 \in [-3, 3]$ . ТФ 1 является унимодальной функцией, а ТФ 2 – мультимодальной. Обучающая выборка сформирована в соответствии с диапазоном входных переменных и составляет 121 пример. По данным из обучающей выборки каждая ТФ восстанавливается с помощью адаптивной МНВ. Начальные значения параметров ФП устанавливаются таким образом, чтобы ФП были равномерно распределены по области определения функций. Эксперименты проводились с использованием разного количества лингвистических термов для каждой из двух входных переменных.

Были выбраны параметры иммунных алгоритмов структурной и параметрической адаптации МНВ. Размер популяции антител для структурной адаптации МНВ соответствовал количеству примеров в

обучающей выборке.

Количество антител для клонирования  $n$  влияет на скорость сходимости алгоритма. Использовался оператор пропорционального клонирования [13], который клонирует  $n$  антител пропорционально их аффинности.

Вероятность мутации влияет на скорость сходимости алгоритма. Большая вероятность мутации поддерживает разнообразие антител в иммунной системе на достаточно высоком уровне и позволяет находить глобальный экстремум. Небольшая вероятность мутации позволяет исследовать локальную область вокруг антитела, делая небольшие шаги в поиске антител с лучшей аффинностью, ведущих к оптимальному значению. Использовался оператор обратно-пропорциональной мутации [13], согласно которому вероятность мутации обратно пропорциональна аффинности антитела.

При редактировании популяции антитела с худшей по популяции аффинностью заменялись новыми случайно сгенерированными антителами, что обеспечивает поддержку разнообразия в популяции.

Оператор супрессии удаляет похожие антитела для того, чтобы избежать избыточности и поддерживать разнообразие в популяции [13]. Оператор супрессии обеспечивает сжатие популяции путем связывания антител в зависимости от значения их аффинностей относительно других антител популяции с установленным порогом сжатия сети, равным 0,001. Критерием останова является достижение заданного количества поколений работы алгоритма.

В табл.1 приведены данные по среднеквадратическому отклонению по результатам 10 запусков алгоритма для тестовых функций (16) и (17).

Таблица 1 – Результаты экспериментов

Количество лингвистических термов	Тестовая функция 1	Тестовая функция 2
3	0.2	0.69
5	0.051	0.082
7	0.012	0.05

Из табл. 1 видно, что использование большего количества ФП для каждой переменной позволяет повысить точность восстановления функции. Восстановленные поверхности тестовых функций (6) и (17) и полученные ФП представлены на рис. 3 и рис. 4.

При адаптации МНВ изменяются не только коэффициенты нечетких правил, но и количество правил нечеткого вывода за счет выполнения супрессии. Количество правил сократилось с 25 до 19 для ТФ (15), и до 23 для ТФ (16).

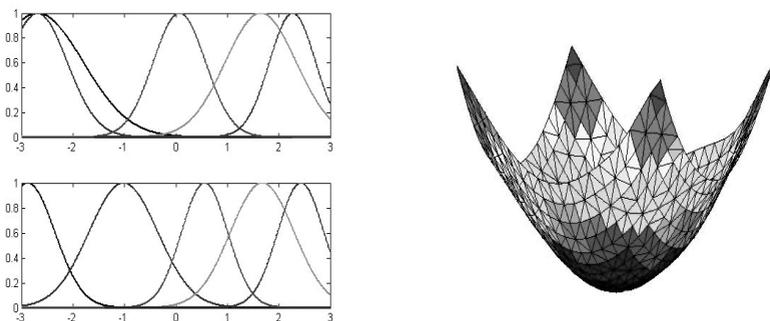


Рис. 3 – Настроенные ФП и восстановленная поверхность для ТФ (15)

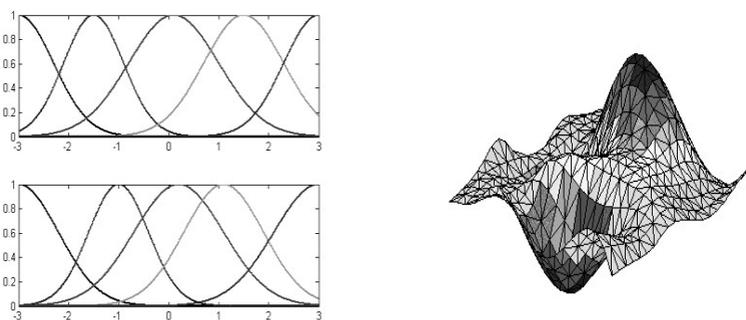


Рис. 4 – Настроенные ФП и восстановленная поверхность для ТФ (16)

## ВЫВОДЫ

В работе рассмотрено решение актуальной задачи повышения качества функционирования интеллектуальных СППР на основе использования гибридных МПР с иммунным обучением. Эффективность решения этой задачи определяется выбором МПР в виде многослойной НС и МНВ Такаги-Сугено первого порядка, а также разработкой методов их параллельного обучения и эволюции на основе иммунного подхода. Эволюция МПР рассматривается как задача их адаптации к условиям изменения внешней среды и свойств ОПР, состоящая из процедур коррекции структуры (числа нейронов в скрытых слоях и связей между ними – для НС; числа правил вывода – для МНВ) и параметров моделей, использующих иммунные модели клонального отбора и иммунной сети.

Осуществлена программная реализация и проведены экспериментальные исследования предложенного подхода с использованием технологий параллельного программирования OpenMP и MS MPI на примере проектирования локальной компьютерной сети и

восстановления тестовых функций, которые подтвердили эффективность предложенных эволюционных МПР, использующих нейросетевой, нечеткий и иммунный подходы.

## **СПИСОК ЛИТЕРАТУРЫ**

- [1] Снитюк В.Е. Эволюционные технологии принятия решений в условиях неопределенности. – К.: «МП Леся», 2015. – 347 с.
- [2] Zohuri B. Neural Network Driven Artificial Intelligence: Decision Making Based On Fuzzy Logic, 2017. – 379 p.
- [3] Ocak H. A medical decision support system based on support vector machines and the genetic algorithm for the evaluation of fetal well-being //Journal of medical systems, 2013. – Т. 37. – №. 2. – P. 99 – 113.
- [4] Мелихова О.А. Методы поддержки принятия решений на основе нейронных сетей // Актуальные проблемы гуманитарных и естественных наук, 2015. – № 9 – 1. – С. 52 – 59.
- [5] Бова В.В. Применение искусственных нейронных сетей для коллективного решения интеллектуальных задач / // Известия Южного федерального университета. Технические науки, 2012. – № 8. – С. 131–136.
- [6] Бухаров О.Е., Боголюбов Д.П. Распараллеленная самообучающаяся система поддержки принятия решений на генетических алгоритмах и нейронных сетях // Системный администратор, 2014. – Вып. № 9 (142). – С. 57 – 69.
- [7] Столярчук А. С. Проектирование модели принятия решения на базе нечёткой логики / Ученые заметки ТОГУ, 2015. – том 6. – № 4. – С. 126 – 130.
- [8] Samigulina G. A. Development of decision support systems based on intellectual technology of artificial immune systems //Automation and Remote Control, 2012. – Т. 73. – №. 2. – С. 397 – 403.
- [9] Самигулина Г. А. Разработка интеллектуальных экспертных систем прогнозирования и управления на основе искусственных иммунных систем // Проблемы информатики. –Новосибирск, 2010. –№.1. –С. 15–22.
- [10] Комарцова Л.Г. Эволюционные методы формирования нечетких баз правил // Материалы международной конференции OSTIS-2011. – С. 181 –184.
- [11] Комарцова Л.Г. Исследование нейросетевых алгоритмов обучения в интеллектуальных эволюционных системах // Материалы IV-й международной конференции OSTIS-2014. – С. 233 – 238.
- [12] Бухаров О. Е., Боголюбов Д. П., Мизикин А. А. Разработка оболочки системы поддержки принятия решений с использованием эволюционных алгоритмов. // «Промышленные АСУ и Контроллеры». – М.: НАУЧТЕХЛИТИЗДАТ, 2013. –№7. – С. 37 – 45.
- [13] Dasgupta D. Recent Advanced in Artificial Immune Systems: Models and

Applications // Applied Soft Computing. Elsevier, 2011. – P. 1574–1587.

[14] Korablev N. Immune Approach for Neuro-Fuzzy Systems Learning Using Multiantibody Model / N. Korablev, I. Sorokina // ICARIS 2011, Springer Lecture Notes in Computer Science, 2011. – Vol. 6825. – P. 395–405.

[15] Korablyov M. Hybrid evolutionary decision-making model based on neural network and immune approaches / M. Korablyov, N. Axak, D. Soloviov // Proceedings of the 13-th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), 2018. – V.1. – P. 378–381.

[16] Korablev N.M. Intelligent Evolution Decision Support System based on Fuzzy Logic and Immune Approach / N.M. Korablev, D.N. Soloviov, R.R. Malyukov // System technologies. №6 (107). – Dnipro, 2016. – P. 98–106.

[17] Axak N. Development of multi-agent system of neural network diagnostics and remote monitoring of patient // Eastern-European Journal of Enterprise Technologies, 2016. – 4/9 ( 82 ) – P.4–11.

## **HYBRID DECISION-MAKING MODELS USING AN IMMUNE APPROACH**

**Korablev N., Fomichev A., Soloviev D., Chuprina A.**

*Modern decision support system (DSS), characterized by processing large amounts of information in an uncertain environment, which requires the use of effective methods and models using a variety of intelligent parallel processing technology. The article offers hybrid decision-making models (DMM), presented both based on the neural network (NN) and in the form of fuzzy inference rules, the training and adaptation of which are carried out on high-performance systems using immune models depending on changes in environmental characteristics and properties decision making object (DMO). The process of learning and adapting NN consists in tuning and correcting network parameters, as well as the number of neurons in hidden layers and the connections between them. The process of training and adapting fuzzy inference models (FIM) consists in adjusting the form and parameters of membership functions, as well as the parameters and structure of the model (the number of inference rules).*

*Keywords. Decision-making, hybrid model, training, adaptation, neural network, fuzzy conclusion, artificial immune systems*