

УДК 681.5

*А.В. Козлов, Ю.П. Кондратенко*

## ПОИСК ОПТИМАЛЬНЫХ ФУНКЦИЙ ПРИНАДЛЕЖНОСТИ НЕЧЕТКИХ СИСТЕМ НА ОСНОВЕ БИОИНСПИРИРОВАННЫХ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ. Часть 1. ПОШАГОВЫЙ МЕТОД

**Ключевые слова:** нечеткая система, функции принадлежности, метод структурной оптимизации, биоинспирированные эволюционные алгоритмы, генетический алгоритм, алгоритм искусственных иммунных систем, биогеографический алгоритм.

### Введение

Теория нечетких множеств и нечеткая логика в настоящее время широко внедрены в практику научных исследований и проектирования разного класса систем [1–3]. Математический аппарат нечеткой логики после теоретического обоснования в ряде фундаментальных работ Л.А. Заде, И. Мамдани, Б. Коско и др. [4–6] успешно применяется для решения различных задач в условиях неполной информации и неопределенности. Наиболее целесообразным является использование нечеткой логики при построении интеллектуальных систем управления и принятия решений для автоматизации сложных нестационарных технических объектов, в частности, таких как морские суда и плавучие сооружения, промышленные и мобильные роботы, необитаемые подводные и летательные аппараты, химические реакторы и т.д. [7–9].

Современные исследования в области создания и развития нечетких систем (НС) управления и принятия решений ведутся преимущественно в направлении разработки высокоэффективных методов их синтеза и структурно-параметрической оптимизации [10–12]. В связи с интенсивным развитием и увеличением мощностей вычислительной техники достаточно перспективными для решения задач синтеза и оптимизации нечетких систем являются биоинспирированные интеллектуальные методы глобального поиска, к которым относятся эволюционные и мультиагентные методы, моделирующие природные процессы естественного отбора и коллективного поведения различных социальных групп животных, насекомых и микроорганизмов [13, 14]. Данные методы являются стохастическими методами глобальной оптимизации и имеют ряд преимуществ по сравнению с классическими методами поиска:

- а) позволяют эффективно оптимизировать нечеткие системы различной конфигурации и большой размерности;
- б) не накладывают дополнительные ограничения на целевые функции;
- в) дают возможность находить глобальные экстремумы больших негладких и мультимодальных пространств поиска [15, 16].

Данная публикация посвящена разработке и исследованию одного из методов структурной оптимизации нечетких систем, в частности метода поиска оптималь-

© А.В. КОЗЛОВ, Ю.П. КОНДРАТЕНКО, 2021

*Международный научно-технический журнал  
«Проблемы управления и информатики», 2021, № 1*

ных функций принадлежности лингвистических термов (ФПЛТ), на основе использования комбинации нескольких биоинспирированных интеллектуальных алгоритмов и состоит из двух частей. В первой части детально изложены теоретические основы предложенного пошагового метода и приведены различные варианты реализации его поисковых процедур. Во второй части приведен анализ исследования эффективности разработанного метода (на конкретном примере нечеткой системы управления движением многоцелевого гусеничного мобильного робота) с детальным обсуждением результатов компьютерного моделирования.

### 1. Функциональная структура обобщенной нечеткой системы и постановка задачи

Функциональная структура обобщенной системы нечеткого логического вывода представлена на рис. 1, где приняты следующие обозначения: SPO Mechanism — механизм структурно-параметрической оптимизации (structural-parametric optimization); FU — блок фаззификации (fuzzification unit); FIE — блок нечеткого логического вывода (fuzzy inference engine); AGG — блок агрегации (aggregation unit); ACT — блок активации (activation unit); ACC — блок аккумуляции (accumulation unit); DFU — блок дефаззификации (defuzzification unit);  $x_1, x_2, \dots, x_i, \dots, x_n$  — входные переменные нечеткой системы (Fuzzy System);  $y_1, y_2, \dots, y_j, \dots, y_m$  — выходные переменные НС; **S** — вектор вариантов структуры нечеткой системы; **P** — вектор параметров НС; **Q** — вектор выходных переменных объекта управления/рабочей среды (Controlled Object/Operating Environment), используемых для структурно-параметрической оптимизации нечеткой системы.

Обобщенная нечеткая ММО-система, приведенная на рис. 1, реализует нелинейную зависимость  $f_{FS}$  [17]

$$\mathbf{Y} = f_{FS}(\mathbf{X}), \quad (1)$$

$$\mathbf{Y} = (y_1, y_2, \dots, y_j, \dots, y_m), \mathbf{X} = (x_1, x_2, \dots, x_i, \dots, x_n),$$

где **X** — вектор  $n$  входных переменных  $x_1, x_2, \dots, x_i, \dots, x_n$ ; **Y** — вектор  $m$  выходных переменных  $y_1, y_2, \dots, y_j, \dots, y_m$  нечеткой системы.

Блок фаззификации FU определяет степени принадлежности численных значений всех  $n$  входных переменных вектора  $\mathbf{X}^*$  к соответствующим нечетким входным лингвистическим термам НС [18]. Блок нечеткого логического вывода FIE, в свою очередь, на основе фаззифицированных сигналов и полученных данных из БП — базы правил (Rule Base) последовательно выполняет операции агрегации AGG, активации ACT и аккумуляции ACC [19].

База правил включает в себя набор правил, составленных из определенных антецедентов (Antecedents) и консеквентов (Consequents). Так, для реализации нечеткой системой функциональной зависимости (1) одно из правил БП нечеткой ММО-системы, построенной на основе нечеткого логического вывода типа Мамдани с использованием соответствующих четырех лингвистических термов для любой переменной  $x_i$ , ( $i = 1 \dots n$ ) или  $y_j$ , ( $j = 1 \dots m$ ), может быть представлено выражением (2) [17, 20]:

$$\text{IF } "x_1 = A_1" \text{ AND } "x_2 = A_2" \text{ AND } \dots \text{ AND } "x_i = A_3" \dots \text{ AND } \dots \text{ AND } "x_n = A_4" \\ \text{THEN } "y_1 = B_1" \text{ AND } "y_2 = B_2" \text{ AND } \dots \text{ AND } "y_j = B_3" \dots \text{ AND } \dots \text{ AND } "y_m = B_4", \quad (2)$$

где  $A_1, A_2, A_3, A_4, B_1, B_2, B_3, B_4$  — соответствующие лингвистические термы входных и выходных переменных НС.

Блок дефаззификации DFU осуществляет преобразование консолидированного нечеткого логического вывода в четкий численный сигнал для каждой  $j$ -й ( $j = 1, 2, \dots, m$ ) выходной переменной НС [18].

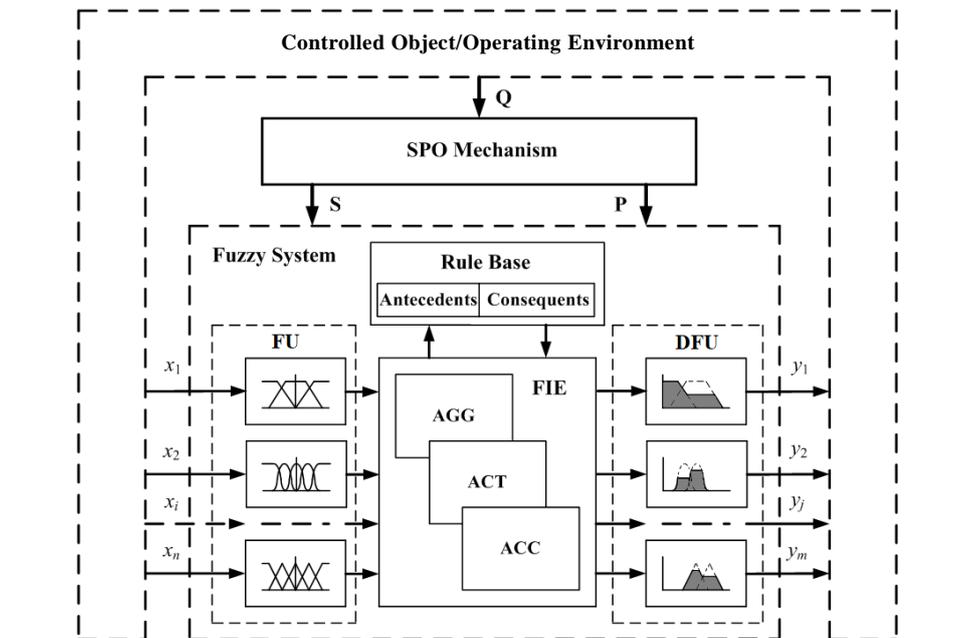


Рис. 1

Проектирование представленной нечеткой системы (рис. 1) целесообразно осуществлять в автоматизированном режиме с помощью встроенного механизма структурно-параметрической оптимизации (SPO Mechanism). При этом SPO Mechanism должен определять такие векторы вариантов структуры  $S$  и параметров  $P$ , при которых будет обеспечиваться достаточно эффективное применение НС для решения той или иной задачи (например, задачи автоматического управления, принятия решений и др.). При этом эффективность функционирования нечеткой системы может быть оценена с помощью определенной целевой функции  $J$ , которая рассчитывается на основе полученных значений вектора измеряемых выходных переменных объекта управления либо рабочей среды  $Q$  [11, 21].

Качество и продуктивность процесса проектирования нечеткой системы непосредственно зависят от применяемых методов оптимизации ее структуры и параметров, реализуемых в блоке SPO Mechanism [17]. Причем очень важна роль методов структурной оптимизации, так как полученный в процессе оптимизации наилучший вариант структуры НС непосредственно влияет не только на значение целевой функции  $J$ , но и на вычислительные затраты при формировании/реализации базы правил и параметрической оптимизации НС, а также на сложность ее последующей программно-аппаратной реализации. В дальнейшем под оптимальной структурой НС подразумевается вариант структуры, при котором обеспечивается:

а) достижение оптимального значения целевой функции  $J$ ;

б) допустимые вычислительные затраты на параметрическую оптимизацию и формирование базы правил;

в) приемлемая сложность программно-аппаратной реализации НС [10, 13, 21].

Исследования, направленные на разработку и усовершенствование методов поиска оптимальной структуры НС, безусловно, являются актуальными и важными для современной теории нечетких систем управления и принятия решений.

Так как методы поиска оптимальной структуры должны успешно решать сложную компромиссную задачу минимизации целевой функции, вычислительных затрат при параметрической оптимизации и степени сложности программно-аппаратной реализации НС, то их разработку целесообразно осуществлять на основе высокоэффективных стохастических алгоритмов глобальной оптимизации, в число которых входят биоинспирированные интеллектуальные алгоритмы [22–24]. Результаты современных исследований показывают, что эволюционные биоинспири-

рованные методы и алгоритмы поиска позволяют с высокой эффективностью проводить оптимизацию нечетких систем различных типов и назначений [25, 26]. Во многих работах приведены примеры успешного применения генетических алгоритмов для структурной и параметрической оптимизации НС [27, 28], в частности, для нечетких систем автоматического управления (САУ) теплоэнергетическими объектами [29], промышленными и мобильными роботами [30, 31], морскими и воздушными транспортными средствами [32, 33]. В свою очередь, в работах [34, 35] приведены результаты оптимизации различных нечетких систем управления и принятия решений на основе методов, моделирующих иммунные системы живых организмов. Также для синтеза и структурной оптимизации НС достаточно перспективным может быть использование методов дифференциальной эволюции [36], эволюционных стратегий [37], биогеографических поисковых алгоритмов [38] и др. [39].

Для более детального исследования влияния различных компонентов структуры проектируемой НС на ее эффективность целесообразно задачи поиска (например, оптимальных функций принадлежности, оптимального количества лингвистических термов, наиболее рациональных процедур агрегации, активации, аккумуляции и дефазификации) решать отдельно и поочередно с помощью специально разработанных методов. При этом одной из важнейших задач для оптимизации структурных компонентов НС является поиск и выбор оптимальных функций принадлежности лингвистических термов для оценивания входных и выходных сигналов НС.

Цель данной публикации — разработка метода поиска оптимальных функций принадлежности нечетких систем на основе комбинации биоинспирированных эволюционных алгоритмов глобальной оптимизации, а также исследование эффективности реальной нечеткой системы управления с использованием синтезированных структурных компонентов.

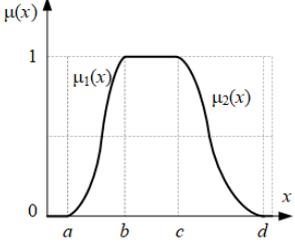
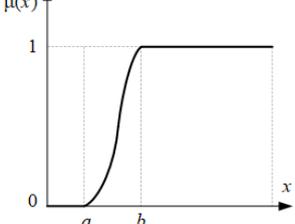
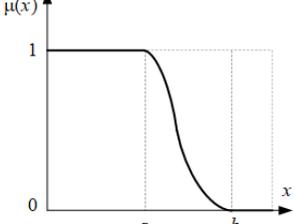
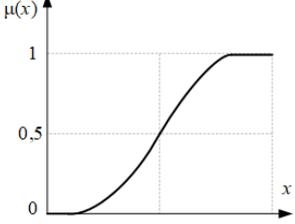
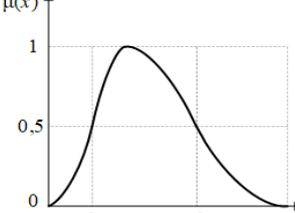
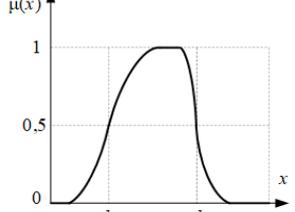
## **2. Метод поиска оптимальных функций принадлежности нечетких систем на основе биоинспирированных эволюционных алгоритмов глобальной оптимизации**

Предложенный авторами метод поиска оптимальных функций принадлежности НС на основе комбинации биоинспирированных эволюционных алгоритмов глобальной оптимизации состоит из следующих последовательных этапов.

**Этап 1. Установление рабочих диапазонов изменения входных и выходных переменных разрабатываемой нечеткой системы.** На данном этапе для каждой  $i$ -й ( $i = 1, 2, \dots, n$ ) входной и  $j$ -й ( $j = 1, 2, \dots, m$ ) выходной переменных НС устанавливаются рабочие диапазоны, в пределах которых могут изменяться данные переменные. Например, если входные переменные поступают на вход НС в относительных единицах от их максимального значения, то их рабочие диапазоны целесообразно устанавливать от  $-1$  до  $1$ . Если некоторые переменные не могут иметь отрицательных значений (например, «расход газа», «мощность электрического нагрева» и др.), то их диапазоны изменения могут быть представлены интервалом  $[0, 1]$ .

**Этап 2. Формирование множества функций принадлежности, используемых в процессе оптимизации.** На данном этапе создается множество альтернативных ФПЛТ  $S_{MF}$ , на котором будет осуществляться поиск оптимальных функций принадлежности для всех лингвистических термов каждой  $i$ -й входной и  $j$ -й выходной переменных разрабатываемой НС. В данное множество целесообразно включать наиболее часто используемые функции принадлежности, к которым относятся: треугольная  $TrFN$ , трапецеидальная  $TrpFN$ , колоколообразная  $GbFN$ , гауссова 1-го  $Gs1FN$  и 2-го  $Gs2FN$  рода,  $\pi$ -образная  $PiFN$ , S-образная  $SFN$ , Z-образная  $ZFN$ , сигмоидная  $SgFN$ , двухсторонняя сигмоидная функция разности  $DsgFN$ , а также двухсторонняя сигмоидная функция произведения  $PsgFN$  [40, 41]. Количество подстраиваемых параметров  $k_{MF}$ , математические модели и графическое изображение [8, 42] данных функций принадлежности лингвистических термов приведены в таблице, где  $a, b, c, d$  — настраиваемые параметры ФПЛТ.

$S_{MF}$	$k_{MF}$	Математическое описание	Графическое изображение
$TrFN$	3	$\mu(x) = \begin{cases} 0 & \text{при } x \leq a \text{ или } x \geq c; \\ \frac{x-a}{b-a} & \text{при } a < x \leq b; \\ \frac{c-x}{c-b} & \text{при } b < x < c; \end{cases}$ $a \leq b \leq c.$	
$TrpFN$	4	$\mu(x) = \begin{cases} 0, & \text{при } x \leq a \text{ или } x \geq d; \\ \frac{x-a}{b-a} & \text{при } a < x \leq b; \\ 1 & \text{при } b < x \leq c; \\ \frac{d-x}{d-c} & \text{при } c < x < d; \end{cases}$ $a \leq b \leq c \leq d.$	
$GbFN$	3	$\mu(x) = \frac{1}{1 + \left  \frac{x-c}{a} \right ^{2b}};$ $a \in (0; +\infty); b \in (-\infty; +\infty);$ $c \in (-\infty; +\infty).$	
$Gs1FN$	2	$\mu(x) = e^{-\frac{(x-b)^2}{2a^2}}.$	
$Gs2FN$	4	<p>если <math>b &lt; d</math>, то</p> $\mu(x) = \begin{cases} e^{-\frac{(x-b)^2}{2a^2}} & \text{при } x < b; \\ 1 & \text{при } b \leq x \leq d; \\ e^{-\frac{(x-d)^2}{2c^2}} & \text{при } x > d; \end{cases}$ <p>если <math>b &gt; d</math>, то</p> $\mu(x) = \begin{cases} e^{-\frac{(x-b)^2}{2a^2}} & \text{при } x < d; \\ e^{-\frac{(x-b)^2}{2a^2}} \cdot e^{-\frac{(x-d)^2}{2c^2}} & \text{при } d \leq x \leq b; \\ e^{-\frac{(x-d)^2}{2c^2}} & \text{при } x > b. \end{cases}$	

<p><i>PiFN</i></p>	<p>4</p>	$\mu(x) = \min \{ \mu_1(x), \mu_2(x) \};$ $\mu_1(x) = \begin{cases} 0, & \text{при } x \leq a; \\ 2 \left( \frac{x-a}{b-a} \right)^2 & \text{при } a < x \leq \frac{a+b}{2}; \\ 1 - 2 \left( \frac{b-x}{b-a} \right)^2 & \text{при } \frac{a+b}{2} < x \leq b; \\ 1 & \text{при } x > b; \end{cases}$ $\mu_2(x) = \begin{cases} 1 & \text{при } x \leq c; \\ 1 - 2 \left( \frac{x-c}{d-c} \right)^2 & \text{при } c < x \leq \frac{c+d}{2}; \\ 2 \left( \frac{d-x}{d-c} \right)^2 & \text{при } \frac{c+d}{2} < x \leq d; \\ 0 & \text{при } x > d. \end{cases}$	
<p><i>SFN</i></p>	<p>2</p>	$\mu(x) = \begin{cases} 0 & \text{при } x \leq a; \\ 2 \left( \frac{x-a}{b-a} \right)^2 & \text{при } a < x \leq \frac{a+b}{2}; \\ 1 - 2 \left( \frac{b-x}{b-a} \right)^2 & \text{при } \frac{a+b}{2} < x \leq b; \\ 1 & \text{при } x > b. \end{cases}$	
<p><i>ZFN</i></p>	<p>2</p>	$\mu(x) = \begin{cases} 1 & \text{при } x \leq a; \\ 1 - 2 \left( \frac{x-a}{b-a} \right)^2 & \text{при } a < x \leq \frac{a+b}{2}; \\ 2 \left( \frac{b-x}{b-a} \right)^2 & \text{при } \frac{a+b}{2} < x \leq b; \\ 0 & \text{при } x > b. \end{cases}$	
<p><i>SgFN</i></p>	<p>2</p>	$\mu(x) = \frac{1}{1 + e^{-a(x-c)}}.$	
<p><i>DsgFN</i></p>	<p>4</p>	$\mu(x) = \frac{1}{1 + e^{-a(x-b)}} \cdot \frac{1}{1 + e^{-c(x-d)}}.$	
<p><i>PsgFN</i></p>	<p>4</p>	$\mu(x) = \frac{1}{1 + e^{-a(x-b)}} \cdot \frac{1}{1 + e^{-c(x-d)}}.$	

**Этап 3. Выбор количества лингвистических термов для входных и выходных переменных нечеткой системы.** На данном этапе выбираются количества лингвистических термов  $\tau_i$  ( $i = 1, 2, \dots, n$ ) и  $\tau_j$  ( $j = 1, 2, \dots, m$ ) для каждой  $i$ -й входной и  $j$ -й выходной переменных НС. В реальных НС количество лингвистических термов для входных переменных целесообразно устанавливать в диапазоне от 2 до 7 ( $\tau_i = 2 \dots 7$ ) [8], а для выходных переменных — от 3 до 9 ( $\tau_j = 3 \dots 9$ ) [40].

**Этап 4. Установление начальных значений параметров функций принадлежности лингвистических термов разрабатываемой нечеткой системы.** На данном этапе для всех ФПЛТ, входящих в сформированное на этапе 2 множество (см. табл. 1), предварительно устанавливаются значения их параметров ( $a, b, c, d$ ). В большинстве случаев в начале поиска параметры функций принадлежности целесообразно выбирать таким образом, чтобы лингвистические термы для всех входных и выходных переменных в зависимости от их количества  $\tau_i$  и  $\tau_j$ , выбранного на этапе 3, были равномерно распределены по их рабочим диапазонам, предварительно сформированным на этапе 1.

**Этап 5. Формирование множества альтернативных функций принадлежности лингвистических термов и структуры вектора  $\mathbf{S}$ , определяющего набор ФПЛТ для разрабатываемой НС.** Искомый вектор  $\mathbf{S}$  ФПЛТ для систем с нечетким логическим выводом Мамдани [5] или Такаги-Сугено [7] соответственно может быть сформирован в виде (3) или (4) на основе множества альтернативных ФПЛТ (см. табл. 1):

$$\mathbf{S} = \{S_{in}^i(q), S_{out}^j(k)\}, q = (1, 2, \dots, \tau_i), k = (1, 2, \dots, \tau_j), i = (1, 2, \dots, n), j = (1, 2, \dots, m),$$

$$S_{in}^i(q) \in$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, SFN, ZFN, SgFN, DsgFN, PsgFN\}, (3)$$

$$S_{out}^j(k) \in$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, SFN, ZFN, SgFN, DsgFN, PsgFN\},$$

$$\mathbf{S} = \{S_{in}^i(q)\}, q = (1, 2, \dots, \tau_i), i = (1, 2, \dots, n),$$

$$S_{in}^i(q) \in (4)$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, SFN, ZFN, SgFN, DsgFN, PsgFN\},$$

где  $S_{in}^i(q)$ ,  $S_{out}^j(k)$  — переменные, определяющие типы функций принадлежности  $q$ -го лингвистического терма  $i$ -й входной переменной и  $k$ -го лингвистического терма  $j$ -й выходной переменной соответственно.

Вектор  $\mathbf{S}$  может иметь определенные ограничения. Например,  $S$ -образная  $SFN$  и сигмоидная  $SgFN$  функции принадлежности (см. табл. 1) могут использоваться только для крайних правых лингвистических термов ( $q = \tau_i$ ,  $k = \tau_j$ ), а, в свою очередь,  $Z$ -образная  $ZFN$  — только для крайних левых термов ( $q = 1$ ,  $k = 1$ ) для всех переменных НС:

$$S_{in}^i(q), S_{out}^j(k) \in$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, ZFN, DsgFN, PsgFN\}$$

$$\text{при } q = 1, k = 1,$$

$$S_{in}^i(q), S_{out}^j(k) \in$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, DsgFN, PsgFN\} (5)$$

$$\text{при } q = (2, \dots, \tau_i - 1), k = (2, \dots, \tau_j - 1),$$

$$S_{in}^i(q), S_{out}^j(k) \in$$

$$\in \{TrFN, TrpFN, GbFN, Gs1FN, Gs2FN, PiFN, SFN, SgFN, DsgFN, PsgFN\}$$

$$\text{при } q = \tau_i, k = \tau_j.$$

**Этап 6. Выбор исходной структуры (начальной гипотезы) вектора  $\mathbf{S}$ , определяющего функции принадлежности лингвистических термов разрабатываемой нечеткой системы.** На данном этапе в соответствии с выбранным на этапе 3 количеством термов  $\tau_i$  ( $i = 1, 2, \dots, n$ ) и  $\tau_j$  ( $j = 1, 2, \dots, m$ ) выбираются начальные значения компонентов вектора  $\mathbf{S}$ , определяющего ФПЛТ системы. Данные начальные значения могут быть выбраны, в частности, на основе экспертного подхода. Например, если в начале поиска в НС типа Мамдани (с параметрами  $n = 3$ ,  $\tau_i = \{4, 3, 2\}$ ,  $m = 1$ ,  $\tau_j = \{5\}$ ) для всех лингвистических термов входных переменных выбираются гауссовы функции принадлежности 1-го рода  $Gs1FN$ , а для всех термов выходных переменных — трапецидальные функции принадлежности  $TrpFN$ , то начальное значение вектора  $\mathbf{S}_0$  будет определяться выражением (6)

$$\begin{aligned} \mathbf{S}_0 &= \{S_{in}^i(q), S_{out}^j(k)\} = \\ &= \{S_{in}^1(1), S_{in}^1(2), S_{in}^1(3), S_{in}^1(4), S_{in}^2(1), S_{in}^2(2), S_{in}^2(3), S_{in}^3(1), S_{in}^3(2), \\ &\quad S_{out}^1(1), S_{out}^1(2), S_{out}^1(3), S_{out}^1(4), S_{out}^1(5)\} = \\ &= \{Gs1FN, Gs1FN, Gs1FN, Gs1FN, Gs1FN, Gs1FN, Gs1FN, Gs1FN, \\ &\quad Gs1FN, Gs1FN, TrpFN, TrpFN, TrpFN, TrpFN, TrpFN\}. \end{aligned} \quad (6)$$

**Этап 7. Формирование комплексной целевой функции  $J_C$  для оценивания эффективности разрабатываемой нечеткой системы.** На данном этапе определяется тип, параметры и оптимальное (граничное) значение комплексной целевой функции  $J_C$ , используемой для поиска оптимальных функций принадлежности. Поскольку от выбранных на этапе 6 типов ФПЛТ зависит общее количество параметров лингвистических термов (поддающихся параметрической оптимизации на последующих этапах проектирования НС), то в процессе структурной оптимизации вектора  $\mathbf{S}$  целесообразно использовать как критерий  $J_1$ , оценивающий эффективность решения поставленной перед НС задачи (управления, принятия решений и др.), так и критерий  $J_2$ , учитывающий степень сложности дальнейшей параметрической оптимизации разрабатываемой системы. Таким образом, задача поиска оптимальных ФПЛТ сводится к задаче многокритериальной оптимизации [12, 43–45], для решения которой необходимо найти оптимальную структуру вектора  $\mathbf{S}$  с учетом минимизации двух критериев:  $J_1$  и  $J_2$ . При решении данной задачи целесообразно использовать априорный подход к решению задач многокритериального эволюционного поиска на основе агрегирования целевых функций [13, 46, 47], согласно которому необходимо осуществлять поиск оптимума единой комплексной целевой функции (глобального критерия)  $J_C$ , сформированной на основе критериев  $J_1$  и  $J_2$  с предварительной оценкой их значимости. В соответствии с данным подходом текущее значение комплексной целевой функции  $J_C$  в процессе поиска оптимальных функций принадлежности НС целесообразно рассчитывать на основе выражения (7)

$$J_C = J_1 + k_{J_2} J_2, \quad (7)$$

где  $k_{J_2}$  — масштабный коэффициент при  $J_2$ , определяющий важность учета данной целевой функции в процессе вычислительного поиска и обеспечивающий масштабирование (нормирование) значений  $J_2$ .

Например, при проектировании одномерной (MISO) нечеткой системы автоматического управления ( $\mathbf{Y} = y$ ) в качестве критерия  $J_1$  может быть выбрано

обобщенное интегральное отклонение реальной переходной характеристики данной системы  $V_R(t, \mathbf{S})$  от желаемой переходной характеристики эталонной модели  $V_D(t)$  [10, 22], которое задается выражением (8)

$$J_1(t, \mathbf{S}) = \frac{1}{t_{\max}} \int_0^{t_{\max}} [(E_V)^2 + k_{11}(\dot{E}_V)^2 + k_{12}(\ddot{E}_V)^2] dt, \quad (8)$$

где  $E_V$  — отклонение  $V_R(t, \mathbf{S})$  от  $V_D(t)$ ,  $E_V = V_D(t) - V_R(t, \mathbf{S})$ ;  $t_{\max}$  — общее время переходного процесса САУ;  $k_{11}$ ,  $k_{12}$  — соответствующие весовые коэффициенты при составляющих  $(\dot{E}_V)^2$  и  $(\ddot{E}_V)^2$ .

В свою очередь, значения критерия  $J_2$ , оценивающего сложность дальнейшей параметрической оптимизации НС, могут рассчитываться в зависимости от количества оптимизируемых параметров лингвистических термов, например, для НС типа Мамдани или Такаги–Сугено — на основе зависимостей (9) или (10) соответственно:

$$J_2 = \sum_{i=1}^n \sum_{q=1}^{\tau_i} k_{in}^i(q) + \sum_{j=1}^m \sum_{k=1}^{\tau_j} k_{out}^j(k), \quad (9)$$

$$J_2 = \sum_{i=1}^n \sum_{q=1}^{\tau_i} k_{in}^i(q). \quad (10)$$

Здесь  $k_{in}^i(q)$ ,  $k_{out}^j(k)$  — количество оптимизируемых параметров  $q$ -го лингвистического терма для  $i$ -й входной переменной и  $k$ -го лингвистического терма для  $j$ -й выходной переменной в зависимости от их функций принадлежности. Например, при использовании соответствующего лингвистического терма  $A = (a, b, c)$  с треугольной функцией принадлежности (см. табл. 1) [40, 41] необходимо осуществлять параметрическую оптимизацию трех параметров  $(a, b, c)$  для каждого лингвистического терма, что при четырех термах для каждой входной и семи термах для каждой выходной переменных потребует оптимизации в целом  $(12n + 21m)$  параметров.

Кроме того, на данном этапе предварительно устанавливается оптимальное (граничное) значение критерия  $J_{1opt}$ , а также выбираются соответствующие значения  $J_{Copt}$ ,  $J_{2opt}$  и  $k_{J2}$ , исходя из требований и особенностей задачи проектирования НС. Например, коэффициент  $k_{J2}$  может быть выбран исходя из следующих условий:

$$\frac{J_{Copt}}{J_{1opt}} < 2, \quad \frac{J_{Copt}}{J_{1opt}} > 2 \quad \text{или} \quad \frac{J_{Copt}}{J_{1opt}} = 2.$$

**Этап 8. Итеративный глобальный поиск оптимальной структуры вектора функций принадлежности  $S_{opt}$  с использованием комбинации поисковых алгоритмов.** Для решения задач подобного класса целесообразно использовать алгоритмы эволюционной оптимизации [11, 13, 16]. Авторы предлагают осуществлять поиск глобального экстремума целевой функции  $J_C \rightarrow \min$  с использованием комбинации биоинспирированных эволюционных алгоритмов, т.е. с использованием не одного, а нескольких биоинспирированных эволюционных алгоритмов. Следовательно, на данном этапе для нахождения глобального оптимума решаемой задачи поочередно или одновременно с использованием параллельных процессоров проводится процедура итеративного поиска на основе нескольких

(двух или более) различных биоинспирированных эволюционных алгоритмов. Для этого предварительно выбранные алгоритмы адаптируются под специфику задачи оптимизации функций принадлежности НС.

Результаты предварительных исследований показывают, что разные биоинспирированные эволюционные алгоритмы при структурной оптимизации вектора  $\mathbf{S}$  дают различные решения. Сравнительный анализ таких альтернативных решений позволит определить оптимальную (в смысле заданной целевой функции) структуру вектора  $\mathbf{S}$ . Для нахождения оптимального вектора ФПЛТ могут быть выбраны следующие биоинспирированные алгоритмы глобальной оптимизации: генетический [28], биогеографический [38], дифференциальной эволюции [36], динамических сетей [48], искусственных иммунных систем [34], а также алгоритмы, имитирующие распространение сорняков [49], поведение кукушек [50] и др.

**Этап 9. Выбор оптимальной структуры вектора  $\mathbf{S}_{opt}$  на основе альтернативных результатов структурной оптимизации функций принадлежности.** На данном этапе проводится анализ полученных на этапе 8 результатов поиска оптимальных функций принадлежности с помощью различных биоинспирированных алгоритмов глобальной оптимизации и на основе сравнения значений комплексной целевой функции  $J_C$ , а также отдельных ее компонентов  $J_1$  и  $J_2$  выбирается наилучший вариант вектора  $\mathbf{S}$ . При этом также можно учитывать количество итераций, затраченное каждым из алгоритмов для достижения оптимального значения комплексной целевой функции  $J_{Copt}$  в процессе поиска.

На рис. 2 в виде блок-схемы представлена информационная модель реализации приведенного вычислительного метода структурной оптимизации нечетких систем путем поиска оптимальных ФПЛТ.

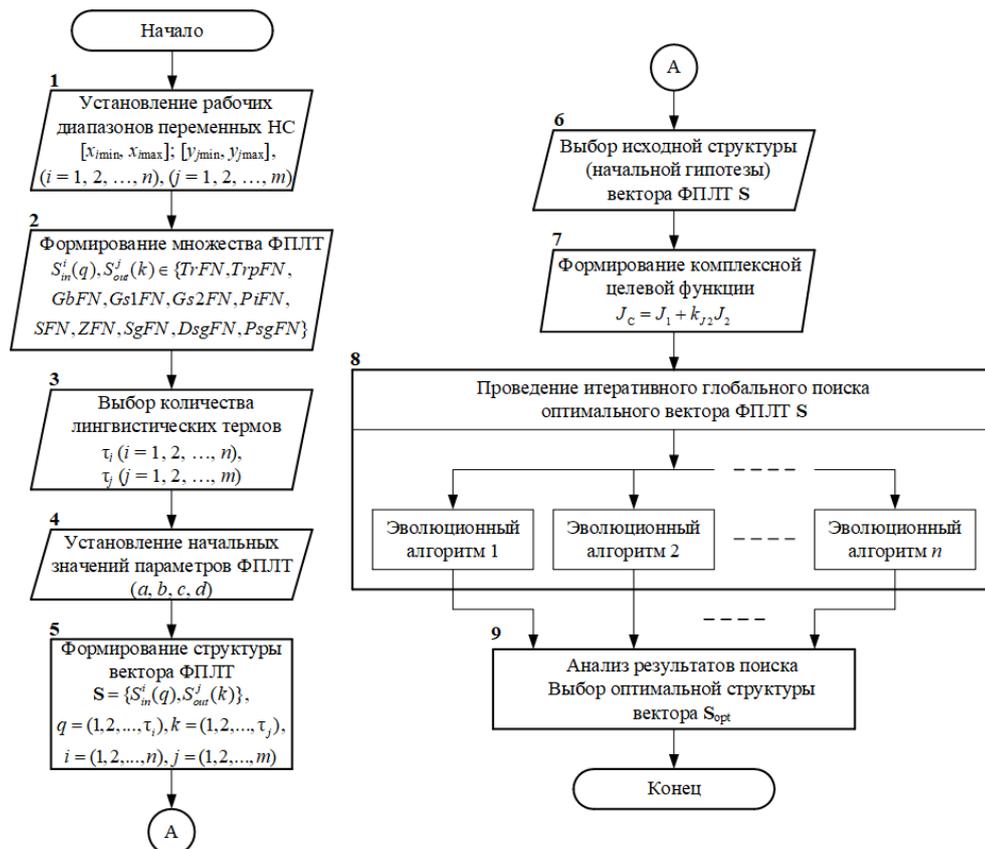


Рис. 2

Далее (разд. 3) в качестве примера рассмотрим особенности применения таких (хорошо апробированных и зарекомендовавших себя) биоинспирированных алгоритмов глобальной оптимизации для поиска оптимальных ФПЛТ НС, как генетический [28], искусственных иммунных систем [34], биогеографический [38].

### 3. Биоинспирированные эволюционные алгоритмы глобальной оптимизации, адаптированные для решения задачи поиска оптимальных ФПЛТ НС

**Генетический алгоритм (Genetic Algorithm — GA).** Генетические алгоритмы моделируют процессы естественного отбора живой природы и для нахождения оптимальных решений в задачах глобального поиска используют механизмы отбора (селекции), скрещивания, наследования и мутации [27, 28]. При этом в данных алгоритмах множество векторов-решений задачи, каждый из которых состоит из оптимизируемых переменных, представляется популяцией особей (хромосом), состоящих из определенного набора генов.

Для оценивания пригодности особей (хромосом) и эффективности решения оптимизационной задачи используется соответствующая функция приспособленности (фитнесс-функция) [29].

Процедура поиска оптимального вектора функций принадлежности  $S_{\text{opt}}$  разрабатываемой нечеткой системы на основе GA состоит в следующей последовательности шагов.

**Шаг 1. Инициализация GA.** На этапе инициализации создается начальная популяция хромосом (вариантов вектора  $S$ ) и выбираются: фитнес-функция  $f$  и критерий окончания поиска, типы генетических операторов селекции, скрещивания и мутации, а также размер начальной популяции  $l_{\text{max}}$ , вероятности скрещивания  $P_C$  и мутации  $P_M$ . Созданная начальная популяция  $N_0$  состоит из  $l_{\text{max}}$  хромосом:

$$N_0 = \{S_1, S_2, \dots, S_l, \dots, S_{l_{\text{max}}}\}, \quad (11)$$

где каждая  $l$ -я хромосома  $S_l$  представляет собой определенный вариант вектора ФПЛТ (3) либо (4), переменные которого  $S_{in}^i(q), S_{out}^j(k)$  (гены хромосомы) задаются случайным образом в рамках сформированного множества функций принадлежности (см. табл. 1). В данном случае применяется вещественное кодирование хромосом (генотип идентичен фенотипу) [27]. Так как генетические алгоритмы являются алгоритмами максимизации функции, то для минимизации комплексной целевой функции (7) фитнес-функцию  $f$  целесообразно рассчитывать как

$$f = \frac{1}{J_C}. \quad (12)$$

В качестве условия окончания поиска целесообразно выбрать достижение оптимального значения фитнес-функции  $f \geq f_{\text{opt}}$  либо выполнение максимального количества итераций  $Z_{\text{max}}$ . В качестве оператора селекции целесообразно выбирать пропорциональную селекцию, в качестве оператора скрещивания — одноточечное либо двухточечное скрещивание, в качестве оператора мутации — простую мутацию.

**Шаг 2. Оценивание хромосом популяции и проверка критерия окончания поиска.** На данном шаге для каждого  $l$ -го варианта вектора  $S_l$  ( $l = \{1, 2, \dots, l_{\text{max}}\}$ ) рассчитывается значение фитнес-функции  $f_l(Z)$  на основе выражения (12), а также осуществляется проверка условия окончания процесса поиска. При достижении оптимального значения фитнес-функции  $f_l(Z) \geq f_{\text{opt}}$  для какого-либо вектора  $S_l$  или при выполнении максимального количества итераций  $Z = Z_{\text{max}}$  осуществляется переход на шаг 7, иначе — на шаг 3.

**Шаг 3. Селекция хромосом для скрещивания.** При использовании механизма пропорциональной селекции рассчитывается среднее значение [30] фитнес-функции популяции  $f_M$  на основе выражения (13)

$$f_M = \frac{1}{l_{\max}} \sum_{l=1}^{l_{\max}} f_l, \quad (13)$$

и для каждого  $l$ -го варианта вектора ФПЛТ  $S_l$  проверяется выполнение условия

$$\frac{f_l}{f_M} > 1. \quad (14)$$

Если условие (14) выполняется, то данный  $l$ -й вектор допускается для скрещивания на шаге 4, а если не выполняется, то соответствующий вектор исключается из текущей популяции.

**Шаг 4. Скрещивание хромосом.** Среди допущенных до скрещивания векторов-хромосом выбираются родительские пары на основе значения вероятности скрещивания  $P_C$  с помощью механизма случайного отбора [28]. Отобранные родительские пары скрещиваются с помощью одноточечного либо двухточечного оператора скрещивания [27]. В свою очередь, точки разрыва вектора ФПЛТ при скрещивании выбираются случайным образом.

**Шаг 5. Мутация хромосом.** Подверженные мутации векторы ФПЛТ выбираются случайным образом на основе значения вероятности мутации  $P_M$  среди родительских особей популяции [30]. Далее, выбранные векторы мутируют с помощью оператора простой мутации [32].

**Шаг 6. Формирование нового поколения.** На данном шаге формируется новое поколение из хромосом-потомков, полученных в результате операций скрещивания и мутации на шагах 4 и 5. Для того чтобы популяция нового поколения имела постоянное количество хромосом (альтернативных вариантов оптимизируемого вектора  $S$ )  $l_{\max}$ , на данном шаге также добавляются недостающие векторы, функции принадлежности (гены) которых генерируются случайным образом [28]. Далее осуществляется переход на шаг 2 и реализуется следующая итерация  $Z$  процесса оптимизации.

**Шаг 7. Окончание процедуры поиска.**

На рис. 3 приведена блок-схема генетического алгоритма, адаптированного под специфику задачи поиска оптимальных ФПЛТ.

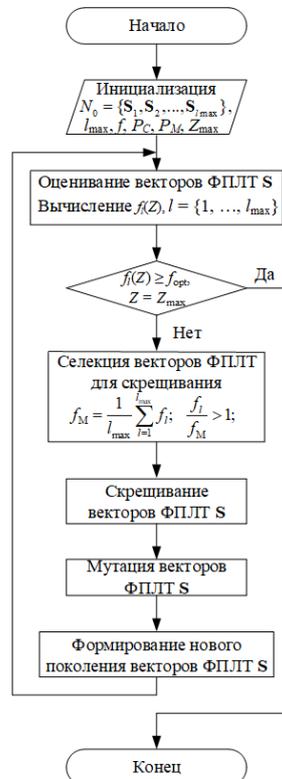


Рис. 3

**Алгоритм искусственных иммунных систем (Artificial Immune Systems — AIS).** Алгоритмы искусственных иммунных систем в процессе поиска оптимальных решений используют идеи, заимствованные из иммунологии, имитируя работу иммунных систем живых организмов [34]. В данных алгоритмах векторы-решения представлены иммунными клетками (антителами), аффинность (эффективность борьбы с антигенами) которых непосредственно зависит от значений целевой функции в решаемой задаче. Для нахождения оптимальных решений в процессе глобального поиска AIS-алгоритмы оперируют механизмами клонирования, мутации и селекции [34, 35].

Процедура поиска оптимального вектора функций принадлежности  $\mathbf{S}_{\text{opt}}$  разрабатываемой нечеткой системы на основе AIS состоит из следующих шагов.

**Шаг 1. Инициализация AIS-алгоритма.** На этапе инициализации создается начальная популяция из  $l_{\text{max}}$  иммунных клеток (вариантов вектора  $\mathbf{S}$ ), выбираются функция аффинности  $f$  и критерий окончания поиска. Также задаются параметры биологических операторов клонирования, мутации и селекции: число клеток памяти  $N_m$ , число клеток в популяции с наихудшей аффинностью  $N_w$ , параметр оператора клонирования  $\rho$  или  $N_c$  в зависимости от типа оператора клонирования, а также параметр оператора мутации  $r$  [34]. Созданная начальная популяция  $N_0$  состоит из  $l_{\text{max}}$  клеток, соответствующих определенным вариантам, например, вектора ФПЛТ (3), переменные которого  $S_{in}^i(q)$ ,  $S_{out}^j(k)$  (параметры клетки) задаются случайным образом в рамках сформированного множества функций принадлежности.

Функцию аффинности  $f$  подобно фитнес-функции в генетических алгоритмах целесообразно рассчитывать на основе выражения (12). В качестве условия окончания поиска целесообразно выбрать достижение оптимального значения аффинности  $f \geq f_{\text{opt}}$  либо выполнение максимального количества итераций  $Z_{\text{max}}$ .

**Шаг 2. Оценивание клеток популяции и проверка критерия окончания поиска.** На данном этапе для каждого  $l$ -го варианта вектора  $\mathbf{S}_l$  ( $l = \{1, 2, \dots, l_{\text{max}}\}$ ) выполняется расчет значения функции аффинности  $f_l(Z)$ , а также осуществляется проверка условия окончания процесса оптимизации. При достижении оптимального значения функции аффинности  $f_l(Z) \geq f_{\text{opt}}$  для какого-либо вектора  $\mathbf{S}_l$  выполняется переход на шаг 9, а иначе — на шаг 3.

**Шаг 3. Клонирование клеток памяти.** На данном этапе выполняется упорядочение вариантов вектора  $\mathbf{S}$  (клеток популяции) по убыванию их аффинности, например  $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_{l_{\text{max}}}$ , где  $f(\mathbf{S}_1) = f_{\text{max}}(Z)$ ,  $f(\mathbf{S}_{l_{\text{max}}}) = f_{\text{min}}(Z)$  [35]. Далее из упорядоченной популяции (с новой нумерацией индексов вариантов векторов  $\mathbf{S}_l$  в соответствии с рейтингом их аффинности) выбирается  $N_m$  первых векторов ФПЛТ с наилучшей аффинностью. При использовании пропорционального оператора клонирования для каждого  $l$ -го вектора  $\mathbf{S}_l$  ( $l = \{1, 2, \dots, N_m\}$ ) генерируется  $N_l$  клонов на основе выражения (15) [34]

$$N_l = \frac{\rho l_{\text{max}}}{l}, l = (1, 2, \dots, N_m). \quad (15)$$

При использовании равномерного оператора клонирования для каждого  $l$ -го вектора (клетки памяти) генерируется заранее выбранное количество  $N_l = N_c$  клонов [35].

**Шаг 4. Мутация клонов клеток памяти.** Для всех  $N_l$  клонов каждой  $l$ -й клетки памяти ( $l = \{1, 2, \dots, N_m\}$ ) выполняется процедура мутации — изменение  $r$  случайно выбранных ФПЛТ (параметров клетки) случайным образом в рамках множества функций принадлежности [34].

**Шаг 5. Оценивание клонов-мутантов клеток памяти и проверка критерия окончания поиска.** На данном шаге для всех  $N_l$  клонов-мутантов каждой  $l$ -й клетки памяти ( $l = \{1, 2, \dots, N_m\}$ ) выполняется расчет значения функции аффинности  $f(Z)$  и проверка условия окончания процесса оптимизации. При достижении оптимального значения функции аффинности  $f(Z) \geq f_{\text{opt}}$  для какого-либо клон-мутанта выполняется переход на шаг 9, иначе — на шаг 6.

**Шаг 6. Селекция и обновление клеток.** Среди всех  $N_l$  клонов-мутантов каждой  $l$ -й клетки памяти ( $l = \{1, 2, \dots, N_m\}$ ) выбирается клон-мутант с наилучшей аффинностью  $f_{\text{max}}$  и сравнивается с данной клеткой памяти. Если аффинность выбранного клон-мутанта выше аффинности его клетки памяти, то данный клон-мутант становится новой клеткой памяти [35].

**Шаг 7. Формирование новой популяции клеток.** На данном шаге выполняется упорядочение обновленных векторов ФПЛТ по убыванию их аффинности [34]. Затем из упорядоченной популяции удаляется  $N_w$  последних векторов с наихудшей аффинностью с их последующей заменой новыми вариантами, сгенерированными случайным образом.

**Шаг 8. Оценивание новых клеток популяции и проверка условия окончания поиска.** На данном этапе для всех сгенерированных новых вариантов вектора  $S$  выполняется расчет значения функции аффинности  $f(Z)$ , а также осуществляется проверка критерия окончания процесса оптимизации. При достижении оптимального значения функции аффинности  $f(Z) \geq f_{\text{opt}}$  для какого-либо нового варианта вектора ФПЛТ или выполнении максимального количества итераций  $Z_{\text{max}}$  осуществляется переход на шаг 9, а иначе — на шаг 3.

**Шаг 9. Окончание процедуры поиска.**

На рис. 4 приведена блок-схема алгоритма искусственных иммунных систем, адаптированного под специфику задачи поиска оптимальных ФПЛТ.

**Биогеографический алгоритм.** Алгоритмы оптимизации на основе биогеографии (Biogeography-Based Optimization — ВВО) моделируют закономерности географического распределения животных, растений и микроорганизмов по средам обитания в живой природе, и в процессе поиска оптимальных решений используют механизмы иммиграции, эмиграции, а также мутации видов [23, 38]. Векторы-решения представлены средами обитания видов (островами) с характеризующими их переменными, а эффективность решения задачи оптимизации оценивается с помощью индекса пригодности среды обитания (Habitat Suitability Index — HSI) [38, 39].

Процедура поиска оптимального вектора функций принадлежности  $S_{\text{opt}}$  разрабатываемой нечеткой системы на основе ВВО состоит из следующих шагов.

**Шаг 1. Инициализация ВВО-алгоритма.** На этапе инициализации создается экосистема из  $l_{\text{max}}$  сред обитания (островов), выбираются функция индекса пригодности среды обитания  $f$  и условие окончания поиска.

Кроме того, задаются основные параметры алгоритма: максимальный уровень иммиграции  $\lambda_{\text{max}}$ , максимальный уровень эмиграции  $\nu_{\text{max}}$ , коэффициент оператора мутации  $r$ , максимально возможное количество видов на острове  $N_{\text{max}}$ , соответствующие оптимальному значению индекса пригодности среды обитания

$f_{opt}$ , а также коэффициент  $k_N$  взаимосвязи количества видов  $N$  на острове от HSI [38]. Также, задаются зависимости миграции видов от количества видов на островах  $N$ :  $\lambda(N)$  и  $\nu(N)$ . В свою очередь, созданная экосистема  $N_0$  состоит из  $l_{max}$  островов, соответствующих определенным вариантам вектора ФПЛТ, переменные которого  $S_{in}^i(q), S_{out}^j(k)$  (характеризующие остров) задаются случайным образом в рамках сформированного множества ФПЛТ [39]. HSI  $f$  целесообразно рассчитывать на основе выражения (12). В качестве условия окончания поиска можно выбрать (а) достижение оптимального значения индекса пригодности  $f \geq f_{opt}$  или (б) выполнение максимального количества итераций  $Z_{max}$ . Зависимости миграции видов от количества видов на островах  $N$  могут иметь линейный характер [38] с максимальными значениями иммиграции и эмиграции (рис. 5) равными единице ( $\lambda_{max} = \nu_{max} = 1$ ).

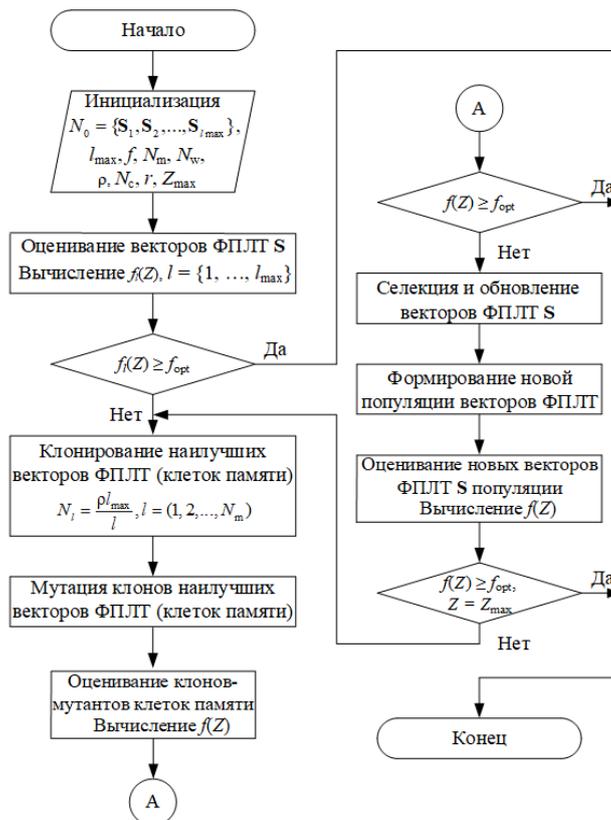


Рис. 4

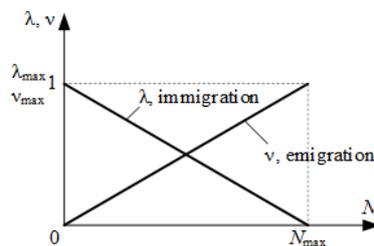


Рис. 5

**Шаг 2. Оценивание островов экосистемы и проверка условия окончания поиска.** На данном этапе для каждого  $l$ -го варианта вектора (острова)  $S_l$  ( $l = \{1, 2, \dots, l_{max}\}$ ) выполняется расчет значения HSI  $f_l(Z)$  на основе выражения (12), а так-

же осуществляется проверка условия окончания процесса оптимизации. При достижении оптимального значения индекса пригодности  $f_l(Z) \geq f_{\text{opt}}$  для какого-либо вектора ФПЛТ или выполнении максимального количества итераций  $Z_{\text{max}}$  выполняется переход на шаг 7, иначе — на шаг 3.

**Шаг 3. Определение уровней миграции видов.** Для каждого  $l$ -го варианта вектора  $S$  на основе значения индекса пригодности  $f_l$  определяется количество видов  $N_l$  и соответствующие уровни иммиграции  $\lambda_l$  и эмиграции  $\nu_l$ . При использовании линейных зависимостей количество видов на острове и уровни миграции рассчитываются на основе выражений (16)–(18) [38]

$$N_l = k_N f_l, l = (1, 2, \dots, l_{\text{max}}); \quad (16)$$

$$\lambda_l = \lambda_{\text{max}} \left( 1 - \frac{N_l}{N_{\text{max}}} \right), l = (1, 2, \dots, l_{\text{max}}); \quad (17)$$

$$\nu_l = \frac{\nu_{\text{max}} N_l}{N_{\text{max}}}, l = (1, 2, \dots, l_{\text{max}}). \quad (18)$$

**Шаг 4. Миграция видов на островах.** Миграция видов с острова на остров осуществляется следующим образом: случайно выбранная переменная  $S_{MF}$  вектора ФПЛТ, выбранного для иммиграции, заменяется соответствующей переменной вектора ФПЛТ, выбранной для эмиграции [39]. При выборе векторов для иммиграции видов поочередно перебираются все  $l_{\text{max}}$  вариантов (островов экосистемы) и каждый  $l$ -й вариант вектора  $S$  может быть выбран с вероятностью, пропорциональной  $\lambda_l$ . Таким же образом, для каждого выбранного для иммиграции вектора осуществляется выбор соответствующего вектора эмиграции видов среди остальных  $l_{\text{max}} - 1$  векторов популяции. При этом каждый вариант вектора ФПЛТ может быть выбран для эмиграции видов с вероятностью, пропорциональной  $\nu_l$  [38].

**Шаг 5. Оценивание островов экосистемы и проверка условия окончания поиска.** На данном этапе для каждого  $l$ -го вектора  $S_l$  ( $l = \{1, 2, \dots, l_{\text{max}}\}$ ) рассчитывается значение HSI  $f_l(Z)$ , а также осуществляется проверка условия окончания процесса оптимизации. При достижении оптимального значения индекса пригодности  $f_l(Z) \geq f_{\text{opt}}$  для какого-либо варианта вектора ФПЛТ выполняется переход на шаг 7, иначе — на шаг 6.

**Шаг 6. Мутация видов на островах.** Для каждого  $l$ -го острова экосистемы с вероятностью  $P_M$  осуществляется процедура мутации видов — изменение случайно выбранной переменной  $S_{MF}$  случайным образом в рамках сформированного множества ФПЛТ [42]. При этом вероятность мутации  $P_M$  рассчитывается [38] на основе выражений (19)–(22)

$$P_M = r \left( \frac{1 - P_l}{P_{l_{\text{max}}}} \right); \quad (19)$$

$$P_l = \frac{\nu_N}{\sum_{N=1}^{N_{\text{max}}+1} \nu_N}; \quad (20)$$

$$\nu_N = \begin{cases} \frac{N_{\text{max}}!}{(N_{\text{max}} + 1 - N)!(N - 1)!} & \text{при } N \in (1, \dots, N'), \\ \nu_{N_{\text{max}} + 2 - N} & \text{при } N \in (N' + 1, \dots, N_{\text{max}} + 1); \end{cases} \quad (21)$$

$$N' = \text{ceil}\left(\frac{N_{\max} + 1}{2}\right). \quad (22)$$

Здесь  $P_l$  — вероятность, что  $l$ -й остров имеет  $N$  видов;  $v_N$  — параметр, характеризующий взаимосвязь количества видов  $N$  и вероятности  $P_l$ ;  $N'$  — ближайшее целое число, которое больше или равно  $(N_{\max} + 1)/2$  (рассчитывается с помощью операции ceil).

Далее выполняется переход на шаг 2.

### Шаг 7. Окончание процедуры поиска.

На рис. 6 приведена блок-схема биогеографического алгоритма, адаптированного под задачу поиска оптимальных ФПЛТ.

Для исследования эффективности предложенного метода структурной оптимизации ФПЛТ в части 2 данной работы проведен поиск оптимальных функций принадлежности для нечеткой САУ скоростью движения многоцелевого гусеничного мобильного робота (МР), способного перемещаться по наклонным и вертикальным ферромагнитным поверхностям [51] (рис. 6).

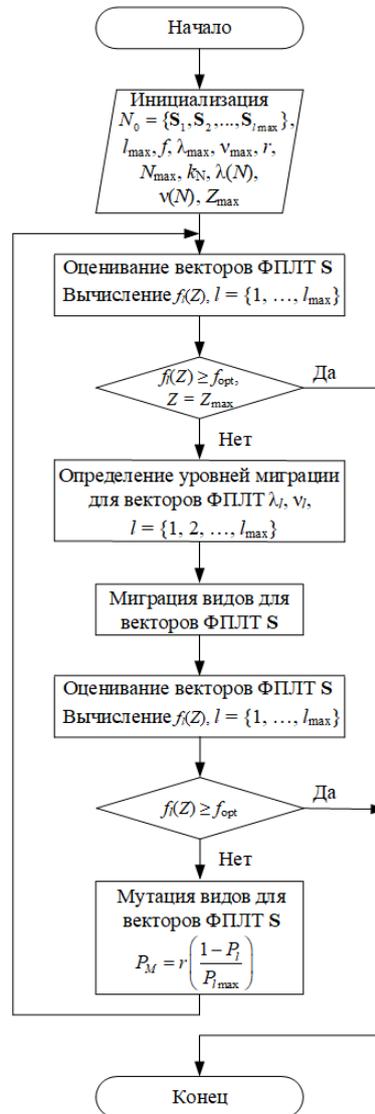


Рис. 6

## Заключення

В настоящей публикации представлена разработка и исследование метода поиска оптимальных функций принадлежности нечетких систем на основе комбинации биоинспирированных эволюционных алгоритмов глобальной оптимизации. Полученный метод позволяет формировать оптимальные ФПЛТ при решении компромиссных задач многокритериальной структурной оптимизации различных НС (управления, принятия решений и др.) для повышения их эффективности и снижения степени сложности дальнейшей параметрической оптимизации. Особенностью предложенного метода является проведение итеративных поисковых процедур на основе комбинации различных (двух или более) биоинспирированных эволюционных алгоритмов с последующим анализом полученных результатов и выбором наилучшего варианта вектора ФПЛТ, соответствующего глобальному оптимуму решаемой задачи. В частности, в данной работе рассмотрены процедуры итеративного глобального поиска оптимального вектора функций принадлежности с использованием таких (адаптированных под специфику решаемой задачи) биоинспирированных эволюционных алгоритмов, как генетический, искусственных иммунных систем и биогеографический.

Во второй части работы будут представлены результаты исследования эффективности реализации предложенного метода (см. рис. 2) на конкретном примере нечеткой системы управления движением многоцелевого гусеничного мобильного робота, способного перемещаться по наклонным и вертикальным ферромагнитным поверхностям.

*О.В. Козлов, Ю.П. Кондратенко*

### ПОШУК ОПТИМАЛЬНИХ ФУНКЦІЙ НАЛЕЖНОСТІ НЕЧІТКИХ СИСТЕМ НА ОСНОВІ БІОІНСПІРОВАННИХ ЕВОЛЮЦІЙНИХ АЛГОРИТМІВ. Частина 1. ПОКРОКОВИЙ МЕТОД

Сучасні дослідження в області створення та розвитку інтелектуальних систем на основі нечіткої логіки ведуться переважно в напрямку розробки високоефективних методів їх синтезу і структурно-параметричної оптимізації. В останні роки у зв'язку з інтенсивним розвитком інформаційних технологій та апаратних засобів обчислювальної техніки досить перспективними для вирішення завдань синтезу й оптимізації нечітких систем є біоінспіровані інтелектуальні методи глобального пошуку, до яких відносяться еволюційні та ройові методи, що моделюють процеси природного відбору, а також колективної поведінки різних груп соціальних тварин, комах і мікроорганізмів в природі. Дана стаття присвячена розробці і дослідженню методу пошуку оптимальних функцій належності нечітких систем на основі біоінспірованих еволюційних алгоритмів глобальної оптимізації. Отриманий метод дозволяє знаходити оптимальні функції належності лінгвістичних термів при вирішенні компромісних задач багатокритеріальної структурної оптимізації різних нечітких систем з метою підвищення їх ефективності і зниження ступеня складності подальшої параметричної оптимізації. У запропонованому методі для знаходження глобального оптимуму розв'язуваної задачі ітеративні процедури проводяться на основі комбінації декількох різних біоінспірованих еволюційних алгоритмів з подальшим аналізом отриманих результатів і вибором найкращого варіанту вектору функцій належності. Викладено теоретичні основи та інформаційна модель реалізації обчислювального покрокового методу структурної оптимізації нечітких систем, а також наведено різні варіанти проведення його пошукових процедур. Зокрема, розглянуто особливості застосування та адаптації під розв'язувану задачу пошуку таких біоінспірованих еволюційних алгоритмів, як генетичний, штучних імунних систем і біогеографічний.

**Ключові слова:** нечітка система, функції належності, метод структурної оптимізації, біоінспіровані еволюційні алгоритми, генетичний алгоритм, алгоритм штучних імунних систем, біогеографічний алгоритм.

## SEARCH OF OPTIMAL MEMBERSHIP FUNCTIONS OF FUZZY SYSTEMS BASED ON BIOINSPIRED EVOLUTIONARY ALGORITHMS. Part I. STEP-BY-STEP METHOD

Contemporary research in the field of creation and development of intelligent systems based on fuzzy logic is carried out mainly in the direction of developing highly efficient methods for their synthesis and structural-parametric optimization. In recent years, due to the intensive development of information technologies and computer hardware, bioinspired intelligent techniques of global search are quite promising for solving problems of synthesis and optimization of fuzzy systems, which include evolutionary and swarm methods, that simulate the processes of natural selection, as well as collective behavior of various groups of social animals, insects and microorganisms in nature. This paper is devoted to the development and study of a method of optimal membership functions search for fuzzy systems based on bioinspired evolutionary algorithms of global optimization. The obtained method allows finding the optimal membership functions of linguistic terms at solving the compromise problems of multicriteria structural optimization of various fuzzy systems in order to increase their efficiency, as well as to reduce the degree of complexity of further parametric optimization. In the proposed method for finding the global optimum of the problem being solved, the iterative procedures are carried out on the basis of combination of several different bioinspired evolutionary algorithms with subsequent analysis of the results obtained and the choice of the best variant of the membership function vector. The paper outlines the theoretical foundations and information model for the implementation of the computational step-by-step method for structural optimization of fuzzy systems, as well as presents various options for carrying out its search procedures. In particular, the features of the application and adaptation to the search problem to be solved of such bioinspired evolutionary algorithms as genetic, artificial immune systems and biogeographic are discussed.

**Keywords:** fuzzy system, membership functions, structural optimization method, bioinspired evolutionary algorithms, genetic algorithm, artificial immune systems algorithm, biogeographic algorithm.

1. Recent developments and new directions in soft computing. L.A. Zadeh, A.M. Abbasov, R.R. Yager, S.N. Shahbazova, M.Z. Reformat, Eds. STUDEFUZ 317, Cham: Springer, 2014. 466 p. DOI 10.1007/978-3-319-06323-2
2. Advance trends in soft computing. M. Jamshidi, V. Kreinovich, J. Kacprzyk, Eds. Cham: Springer-Verlag, 2013. 468 p. DOI 10.1007/978-3-319-03674-8
3. Kondratenko Y.P., Korobko O.V., Kozlov O.V. Synthesis and optimization of fuzzy controller for thermoacoustic plant. *Lotfi A. Zadeh et al. (Eds.) Recent Developments and New Direction in Soft-Computing Foundations and Applications, Studies in Fuzziness and Soft Computing*. Berlin, Heidelberg: Springer-Verlag, 2016. **342**. P. 453–467. DOI 10.1007/978-3-319-32229-2\_31
4. Zadeh L.A. The role of fuzzy logic in modeling, identification and control. *Modeling Identification and Control*. 1994. 15(3). P. 191–203. DOI: 10.1142/9789814261302\_0041
5. Mamdani E.H. Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of IEEE*. 1974. **121**. P. 1585–1588. DOI: 10.1049/piee.1974.0328
6. Kosko B. Fuzzy Systems as Universal Approximators. *IEEE Trans. on Computers*. 1994. **43**, N 11. P. 1329–1333. DOI: 10.1109/12.324566
7. Gupta M, Behera L., Venkatesh K.S. PSO based modeling of Takagi-Sugeno fuzzy motion controller for dynamic object tracking with mobile platform. *International Multiconference Computer Science and Information Technology (IMCSIT)*. 2010. P. 37–43. DOI: 10.1109/IMCSIT.2010.5680034
8. Safaee B., Mashhadi S.K.M. Fuzzy membership functions optimization of fuzzy controllers for a quad rotor using particle swarm optimization and genetic algorithm. *Proceedings of 2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*. 2016. P. 256–261. DOI 10.1109/ICCIAutom.2016.7483170
9. Kondratenko Y.P., Kozlov O.V. Mathematical model of ecopyrogenesis reactor with fuzzy parametrical identification. *Lotfi A. Zadeh et al. (Eds.) Recent Developments and New Direction in Soft-Computing Foundations and Applications, Studies in Fuzziness and Soft Computing*. Berlin, Heidelberg: Springer-Verlag, 2016. **342**. P. 439–451. DOI 10.1007/978-3-319-32229-2\_30
10. Kondratenko Y.P., Kozlov O.V., Korobko O.V. Two modifications of the automatic rule base synthesis for fuzzy control and decision making systems. Medina J., Ojeda-Aciego M., Verdegay J.L., Pelta D.A., Cabrera I.P., Bouchon-Meunier B., Yager R.R. (Eds.) *Information processing and management of uncertainty in knowledge-based systems. Theory and foundations. Book series: Communications in computer and information science*. 854. Berlin, Heidelberg: Springer International Publishing. 2018. P. 570–582. DOI [https://doi.org/10.1007/978-3-319-91476-3\\_47](https://doi.org/10.1007/978-3-319-91476-3_47)

11. Simon D. Evolutionary optimization algorithms: biologically inspired and population-based approaches to computer intelligence. *John Wiley & Sons, Inc.*, Hoboken, New Jersey, USA, 2013. 772 p. ISBN: 978-0-470-93741-9
12. Kondratenko Y., Khalaf P., Richter H., Simon D. Fuzzy real-time multi-objective optimization of a prosthesis test robot control system. Yuriy P. Kondratenko, Arkadii A. Chikrii, Vyacheslav F. Gubarev, Janusz Kacprzyk (Eds.) *Advanced Control Techniques in Complex Engineering Systems: Theory and Applications. Dedicated to Professor Vsevolod M. Kuntsevich. Studies in Systems, Decision and Control.* **203**. – Cham: Springer Nature Switzerland AG, 2019, P. 165–185. DOI: 10.1007/978-3-030-21927-7\_8
13. Субботін С.О., Олійник А.О., Олійник О.О. Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей. Запоріжжя: ЗНТУ. 2009. 375 с. ISBN 978–966–7809–96–6
14. Zhu J., Lauri F., Koukam A., Hilaire V. Fuzzy logic control optimized by artificial immune system for building thermal condition. Siarry P., Idoumghar L., Lepagnot J. (Eds.) *Swarm Intelligence Based Optimization. ICSIBO 2014. Lecture Notes in Computer Science.* **8472**. Springer, Cham, 2014. P. 42–49. DOI: 10.1007/978-3-319-12970-9\_5
15. Melendez A., Castillo O. Evolutionary optimization of the fuzzy integrator in a navigation system for a mobile robot. *Recent Advances on Hybrid Intelligent Systems.* 2013. P. 21–31. [https://doi.org/10.1007/978-3-642-33021-6\\_2](https://doi.org/10.1007/978-3-642-33021-6_2)
16. Smiley A., Simon D. Evolutionary optimization of atrial fibrillation diagnostic algorithms. *International Journal of Swarm Intelligence.* 2016. **2**, N 2/3/4. P. 117–133. DOI: 10.1504/IJSI.2016.10002173
17. Kondratenko Y.P., Kozlov O.V. Generation of rule bases of fuzzy systems based on modified ant colony algorithms. *Journal of Automation and Information Sciences.* 2019. **51**, N 3. P. 4–25. DOI: 10.1615/JAutomatInfScien.v51.i3.20
18. Hampel R., Wagenknecht M., Chaker N. Fuzzy control: Theory and practice. New York: Physika-Verlag, Heidelberg. 2000. 410 p. DOI 10.1007/978-3-7908-1841-3.
19. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М. : Горячая линия. Телеком. 2006. 452 с.
20. Kacprzyk J. Multistage fuzzy control: A prescriptive approach. *John Wiley & Sons, Inc.*, New York, NY, USA, 1997. 338 p.
21. Kozlov O., Kondratenko G., Gomolka Z., Kondratenko Y. Synthesis and optimization of green fuzzy controllers for the reactors of the specialized pyrolysis plants. Kharchenko V., Kondratenko Y., Kacprzyk J. (Eds.) *Green IT Engineering: Social, Business and Industrial Applications. Studies in Systems, Decision and Control.* 2018. **171**. Springer, Cham. P. 373–396. DOI 10.1007/978-3-030-00253-4\_16
22. Kondratenko Y.P., Simon D. Structural and parametric optimization of fuzzy control and decision making systems. Zadeh L., Yager R., Shahbazova S., Reformat M., Kreinovich V. (Eds.) *Recent Developments and the New Direction in Soft-Computing Foundations and Applications. Studies in Fuzziness and Soft Computing.* 2018. **361**. Springer, Cham. P. 273–289. DOI: [https://doi.org/10.1007/978-3-319-75408-6\\_22](https://doi.org/10.1007/978-3-319-75408-6_22)
23. Ovreiu M., Simon D. Biogeography-based optimization of neuro-fuzzy system parameters for diagnosis of cardiac disease. *Proceedings of Genetic and Evolutionary Computation Conference.* 2010. P. 1235–1242. DOI: 10.1145/1830483.1830706
24. Nabi A., Singh N.A. Particle swarm optimization of fuzzy logic controller for voltage sag improvement. *Proceedings of 2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS).* 2016. **01**. P. 1–5. DOI 10.1109/ICACCS.2016.7586345
25. Alves R.T., Delgado M.R., Lopes H.S., Freitas A.A. An artificial immune system for fuzzy-rule induction in data mining. Yao X. et al. (Eds.) *Parallel Problem Solving from Nature — PPSN VIII. PPSN 2004. Lecture Notes in Computer Science.* 2004. **3242**. Springer, Berlin, Heidelberg. P. 1011–1020. [https://doi.org/10.1007/978-3-540-30217-9\\_102](https://doi.org/10.1007/978-3-540-30217-9_102)
26. Castillo O., Ochoa P., Soria J. Differential evolution with fuzzy logic for dynamic adaptation of parameters in mathematical function optimization. Angelov P., Sotirov S. (Eds.) *Imprecision and Uncertainty in Information Representation and Processing. Studies in Fuzziness and Soft Computing.* 2016. **332**. Springer, Cham. P. 361–374. [https://doi.org/10.1007/978-3-319-26302-1\\_21](https://doi.org/10.1007/978-3-319-26302-1_21)
27. Khan S. et al. Design and implementation of an optimal fuzzy logic controller using genetic algorithm. *Journal of Computer Science.* 2008. **4**, N 10. P. 799–806. DOI: 10.3844/jcssp.2008.799.806
28. Cordon O., Gomide F., Herrera F., Hoffmann F., Magdalena L. Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy Sets and Systems.* 2004. **141**, N 1. P. 5–31. [https://doi.org/10.1016/S0165-0114\(03\)00111-8](https://doi.org/10.1016/S0165-0114(03)00111-8)
29. Liu X.-H., Kuai R., Guan P., Ye X.-M., Wu Z.-L. Fuzzy-PID control for arc furnace electrode regulator system based on genetic algorithm. *Proceedings of the Eighth International Conference on Machine Learning and Cybernetics.* Baoding, 2009. P. 683–689. DOI: 10.1109/ICMLC.2009.5212457
30. Zhao J., Han L., Wang L., Yu Z. The fuzzy PID control optimized by genetic algorithm for trajectory tracking of robot arm. *2016 12th World Congress on Intelligent Control and Automation (WCICA).* Guilin, China, 2016. P. 556–559. DOI: 10.1109/WCICA.2016.7578443
31. Chen Ch., Li M., Sui J., Wei K., Pei Q. A genetic algorithm-optimized fuzzy logic controller to avoid rear-end collisions. *Journal of advanced transportation.* 2016. **50**. P. 1735–1753. <https://doi.org/10.1002/atr.1426>

32. Li H.F., Feng Z.G., Wang J. GA based design of fuzzy control law for hypersonic vehicle. *2008 2nd International Symposium on Systems and Control in Aerospace and Astronautics*. 2008. P. 1–4. DOI: 10.1109/ISSCAA.2008.4776229
33. Li L., Zhu Q., Gao Sh. Design and realization of waterjet propelled craft autopilot based on fuzzy control and genetic algorithms. *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation*. Luoyang, China, 2006. P. 1362–1366. DOI: 10.1109/ICMA.2006.257826
34. Prakash A., Deshmukh S.G. A multi-criteria customer allocation problem in supply chain environment: An artificial immune system with fuzzy logic controller based approach. *Expert Systems with Applications*. 2011. **38**, N 4. P. 3199–3208. <https://doi.org/10.1016/j.eswa.2010.09.008>
35. Visconti A., Tahayori H. Artificial immune system based on interval type-2 fuzzy set paradigm. *Applied Soft Computing*. 2011. **11**, N 6. P. 4055–4063. <https://doi.org/10.1016/j.asoc.2010.12.011>
36. Hachicha N., Jarboui B., Siarry P. A fuzzy logic control using a differential evolution algorithm aimed at modelling the financial market dynamics. *Information Sciences*. 2011. **181**. P. 79–91. <https://doi.org/10.1016/j.ins.2010.09.010>
37. Minku F.L., Ludermir T. Evolutionary strategies and genetic algorithms for dynamic parameter optimization of evolving fuzzy neural networks. *Evolutionary Computation, The 2005 IEEE Congress*. 2005. **3**. P. 1951–1958. DOI: 10.1109/CEC.2005.1554934
38. Thomas, G., Lozovyy, P., Simon, D. Fuzzy Robot Controller Tuning with Biogeography-Based Optimization. *Modern Approaches in Applied Intelligence: 24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems*. IEA/AIE 2011, Syracuse, NY, USA, June 28 – July 1, Proceedings, Part II, 2011. P. 319–327. DOI: 10.1007/978-3-642-21827-9\_33
39. Zhang, M., Jiang, W., Zhou, X. et al. A hybrid biogeography-based optimization and fuzzy C-means algorithm for image segmentation. *Soft Computing*. 2019. **23**. P. 2033–2046. <https://doi.org/10.1007/s00500-017-2916-9>
40. Piegat A. Fuzzy modeling and control. Springer, Heidelberg, 2001. 728 p. DOI 10.1007/978-3-7908-1824-6
41. Kondratenko Y.P., Kondratenko N.Y. Soft computing analytic models for increasing efficiency of fuzzy information processing in decision support systems. Chapter in book: *Decision Making: Processes, Behavioral Influences and Role in Business Management*, R. Hudson (Ed.), *Nova Science Publishers*. New York, 2015. P. 41–78.
42. Kondratenko Y.P., Klymenko L.P., Al Zu'bi E.Y.M. Structural optimization of fuzzy systems' rules base and aggregation models. *Kybernetes*. 2013. **42**, N 5. P. 831–843. DOI: <http://dx.doi.org/10.1108/K-03-2013-0053>
43. Катренко А.В., Пасічник В.В., Пасько В.П. Теорія прийняття рішень : підручник. Київ: Видавнича група BHV, 2009. 448 с.
44. Kondratenko Y., Kondratenko, G., Sidenko, I. Multi-criteria decision making for selecting a rational IoT platform. *Proceedings of 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies*. DESSERT 2018, 24–27 May 2018, Kiev, Ukraine. P. 147–152. DOI: 10.1109/DESSERT.2018.8409117
45. Nazarenko A.M., Karpusha M.V. Modeling and identification in the problems of multicriteria optimization with linear and quadratic performance criteria under statistical uncertainty. *Journal of Automation and Information Sciences*. 2014. **46**, N 3. P. 17–29. DOI: 10.1615/JAutomatInfScien.v46.i3.30
46. Kondratenko Y.P., Al Zubi E.Y.M. The optimisation approach for increasing efficiency of digital fuzzy controllers. *Annals of DAAAM for 2009 & Proceeding of the 20th Int. DAAAM Symp. «Intelligent Manufacturing and Automation»*. Published by DAAAM International, Vienna, Austria, 2009. P. 1589–1591.
47. Kondratenko Y., Kondratenko G., Sidenko I. Multi-criteria decision making and soft computing for the selection of specialized IoT platform. Chertov O., Mylovanov T., Kondratenko Y., Kacprzyk J., Kreinovich V., Stefanuk V. (Eds.) *Recent Developments in Data Science and Intelligent Analysis of Information. Proceedings of the XVIII International Conference on Data Science and Intelligent Analysis of Information*. June 4–7, 2018, Kyiv, Ukraine. ICDSIAI 2018, Advances in Intelligent Systems and Computing, Vol. 836, Springer International Publishing, 2019. P. 71–80. DOI: 10.1007/978-3-319-97885-7\_8
48. Puris A., Bello R., Molina D., Herrera F. Variable mesh optimization for continuous optimization problems. *Soft Computing*. 2011. **16**, N 3. P. 511–525. DOI: 10.1007/s00500-011-0753-9
49. Mehrabian A.R., Lucas C. A novel numerical optimization algorithm inspired from weed colonization. *Ecological Informatics*. 2006. **1**. P. 355–366. DOI: 10.1016/j.ecoinf.2006.07.003
50. Li J., Li Y., Tian S. et al. An improved cuckoo search algorithm with self-adaptive knowledge learning. *Neural Comput & Applic*. 2020. **32**. P. 11967–11997. DOI: 10.1007/s00521-019-04178-w
51. Kondratenko Y.P., Zaporozhets Y.M., Rudolph J., Gerasin O.S., Topalov A.M., Kozlov O.V. Features of clamping electromagnets using in wheel mobile robots and modeling of their interaction with ferromagnetic plate. *Proc. of the 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. 2017. **1**. Bucharest, Romania. P. 453–458. DOI: 10.1109/IDAACS.2017.8095122

Получено 06.10.2020