

УДК 519.71(575.1)

РЕШЕНИЕ ЗАДАЧИ МОНИТОРИНГА НА ОСНОВЕ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ ИММУННЫХ СИСТЕМ

Мухамедиева Д.Т.

д.т.н., проф., ведущий научный сотрудник
Научно-инновационного центра информационно-коммуникационной технологий,
тел.: +(99895) 195-47-52, e-mail: dilnoz134@rambler.ru

Примова Х.А.

старший научный сотрудник-исследователь
Научно-инновационного центра информационно-коммуникационной технологий,
тел.: +(99893) 727-85-61, e-mail: xolida_primova@mail.ru

Решение задач мониторинга и принятия решений в слабоформализуемых системах только с помощью детерминированных и вероятностных методов является недостаточным. Для этого необходимо широко применять методы гибридных интеллектуальных систем и, особенно, методы «мягких» вычислений (SoftCalculation, SoftComputing) и зарождающиеся на этой теоретико-методологической базе направления ComputationalIntelligence – интеллектуальные вычислительные технологии. Предлагается иммунно-нечеткий алгоритм синтеза систем нечеткого вывода (СНВ). Описывается двухэтапный адаптивный алгоритм синтеза СНВ. На первом этапе производится кластеризация исходных нечетких параметров с целью сокращения числа входных параметров нечетких правил, а на втором - осуществляется синтез нечетких моделей (правил вывода) типа Сугэно.

Ключевые слова: иммунный алгоритм, нечеткое множество, мониторинг, слабоформализуемая система, принятие решений.

SOLUTION OF THE PROBLEM OF MONITORING ON THE BASIS OF ARTIFICIAL IMMUNE SYSTEMS Muhamediyeva D.T., Primova X.A.

Solution monitoring tasks and decision-making systems in poorly formalized only by means of deterministic and probabilistic methods is insufficient. To do this, widely used methods of hybrid intelligent systems and, especially, the methods of "soft" computing (Soft Calculation, Soft Computing) and emerging on this theoretical and methodological basis of the direction of Computational Intelligence - intelligent computing technologies. It proposed immuno - fuzzy synthesis of fuzzy inference systems (FIS). We describe a two-stage adaptive algorithm FIS synthesis. In the first stage of initial fuzzy clustering parameters with a view to reducing the number of input parameters of the fuzzy rules, and the second - the synthesis of fuzzy models (inference rules) Sugeno type.

Keywords: immune algorithm, fuzzy set, monitoring, weakly formalized system, the adoption of a decision.

СУНЪИЙ ИММУН ТИЗИМЛАРНИ ҚЎЛЛАГАН ҲОЛДА МОНИТОРИНГ МАСАЛАСИНИ ЕЧИШ Мухамедиева Д.Т., Примова Х.А.

Суст шаклланган тизимларда қарор қабул қилиш ва мониторинг масаласини фақат детерминалашган ва эҳтимоллик усуллари билан ечиш етарли ҳисобланмайди. Бунинг учун гибрид интеллектуал тизим усуллари, айниқса “юмшоқ” ҳисоблаш усуллари (SoftCalculation, SoftComputing) ва ушбу назарий-методологик асосда вужудга келган йўналиш Computational Intelligence – интеллектуал ҳисоблаш технологияларини кенг қўллаш зарур. Норавадан ҳулосалар тизимини синтезлашнинг иммун-норавадан алгоритми тақлиф қилинган. Норавадан ҳулосалар тизимини синтезлашнинг икки босқичли мослашувчан алгоритми тавсифланган. Биринчи босқичда норавадан қодаларнинг қирувчи параметрлари сонини қисқартириш мақсадида берилган норавадан параметрлар кластеризацияси, иккинчи босқичда эса Сугэно типдаги норавадан моделлар (чиқариш қодалари) синтези амалга оширилади.

Таянч иборалар: иммун алгоритмлар, норавадан тўпламлар, мониторинг, яхши шаклланмаган тизимлар, қарор қабул қилиш.

1. Введение

В прикладных интеллектуальных системах мониторинга и поддержки принятия слабоформализуемых решений основными функциональными задачами являются классификация, кластеризация, распознавание образов, прогнозирование, оценка состояний, выявление закономерностей между параметрами различных типов, оптимизация и принятие решений. Отличительными особенностями решаемых задач указанных типов являются: большая размерность, многокритериальность, наличие неопределенностей в исходной информации и ситуации, динамичность изменения параметров внешней среды, предсказуемость которых, зачастую, является затруднительной или невозможной.

Эти особенности для решения указанных задач, наряду с традиционными методами и средствами исследования операций и интеллектуального анализа данных (ИАД), обуславливают использование интеллектуальных технологий, основанных на неформальных эмпирических знаниях экспертов и логических рассуждениях, а также на природно-биологических механизмах обучения, эволюции, адаптации и оптимизации [1-11]. Примерами таких интеллектуальных технологий, получивших наибольшее развитие и использование, являются экспертные системы (ЭС), технологии SoftComputing (SC), включающие нечеткие множества [1, 2], нейронные сети (НС) [12-14] и генетические алгоритмы (ГА) [15,16]; эволюционные алгоритмы (Swarmintelligence – роевого интеллекта: муравьиные, пчелиные, стаи птиц и рыб) [17-29]. В последнее время получают развитие новые интеллектуальные технологии, основанные на природно-биологических механизмах, такие как искусственные иммунные системы (ИИС) [30-32], называемые также иммунными вычислениями или иммунокомпьютингом (ИК) и ДНК-вычисления [27-29]. Из них наибольшее практическое применение получают ИИС. Поэтому представляет интерес исследование возможностей и определения основных направлений по совместному использованию интеллектуальных технологий для создания гибридных интеллектуальных, в том числе эволюционных, алгоритмов и программных средств, которые обеспечат более высокую эффективность решения исследуемых задач. Модели и алгоритмы нечеткого вывода занимают центральное место в задачах принятия решений, управления, прогнозирования, классификации, распознавания и машинного обучения в условиях неопределенности нечеткой природы. Алгоритмы нечеткого вывода реализуются системами нечеткого вывода (СНВ), ядром которых являются продукционные правила типа «Если A , То B » [2,4]. Эти правила формируются на основе лингвистических высказываний экспертов. В совокупности система таких правил отображает эвристическую модель исследуемых задач.

При решении прикладных задач в условиях неопределенности нечеткой, размытой (т.е.

нестохастической) природы информацию, необходимую для построения и реализации системы принятия решений, можно разделить на две части: численную (количественную) и лингвистическую (качественную), поступающую от эксперта. Значительная часть нечетких систем использует второй вид знаний, чаще всего представляемых в форме базы нечетких правил, объединяемых в СНВ. Особенностью моделей таких задач является наличие в них нечеткой базы правил, описывающей структуру нечеткой модели задачи в целом и содержащих основные знания (экспертную информацию) о моделируемой системе, т.е. главную составляющую «интеллекта» рассматриваемой задачи. Поэтому корректное формирование нечеткой базы правил является очень важным условием эффективного решения поставленной задачи. Для решения задач такого класса широкое применение получают подходы, основанные на интеллектуальных технологиях «Soft Computing» [1-4,12-32].

Для того чтобы такая модель была адекватна реальной ситуации, количество формируемых правил в СНВ обычно должно быть равно числу элементов входного вектора - условия A правила. Чрезмерно большое их число приводит к увеличению размерности и, соответственно, сложности решаемой задачи. Кроме того, объем имеющейся доступной информации, в том числе экспертной, о моделируемой системе часто оказывается недостаточным для построения более сложной и адекватной модели. Следует также учитывать наличие объективных ограничений на точность получения исходных данных. Поэтому при их формировании и оценке в процессе построения исследуемых моделей следует использовать принцип разумной полноты и точности. Это обуславливает важность анализа исходной информации и использования процедур разумного сокращения количества правил.

Одним из перспективных подходов к формированию нечетких правил и настройки значений их параметров, в особенности, когда в наличии имеются только численные данные, являются нечеткие нейронные сети (fuzzy-neural) [2]. При всех достоинствах их основным недостатком является длительность построения базы нечетких правил в процессе итеративного обучения нейронных сетей (НС).

С целью устранения этого недостатка предлагается комбинированный метод построения базы нечетких правил с использованием кластеризации на основе нечетких отношений и иммунных алгоритмов. При построении процедур классификации и кластеризации были проанализированы и систематизированы различные критерии [5-9]. Достоинство этого метода заключается в его простоте и высокой эффективности. Кроме того, он позволяет объединять численную информацию, представленную в форме обучающих данных, с лингвистической информацией, имеющей вид базы правил, за счет

дополнения имеющейся базы правилами, созданными на основе численных данных.

Предлагаемый алгоритм синтеза правил СНВ и настройки их параметров реализуется в два этапа.

На первом этапе осуществляется кластеризация (clustering) входных переменных правил. Каждый из сформированных кластеров будет объединять группу исходных входных переменных, сходных по определенным признакам. В этом случае каждый кластер может рассматриваться как обобщенное условие для соответствующего формируемого правила СНВ. Результатом этого этапа являются лингвистические правила СНВ с предварительными, грубыми, значениями их параметров, описывающих математические модели функций принадлежности.

На втором этапе производится уточнение и настройка этих параметров с использованием иммунных алгоритмов обучения.

2. Постановка задачи синтеза правил СНВ

Рассматривается задача нечеткого логического вывода (НЛВ), описываемая нечеткой моделью Сугэно [2,10]:

$$\bigcup_{p=1}^{k_j} \left(\bigcap_{i=1}^n x_i = a_{i,jp} - \text{с весом } w_{jp} \right) \rightarrow y_j = f_j(x_1, x_2, \dots, x_n). \quad (1)$$

Здесь $j = \overline{1, m}$ - номер правила; $a_{i,jp}$ - лингвистический терм, которым оценивается входная переменная x_i , $i = \overline{1, n}$, в строке-конъюнкции с номером jp ($jp = \overline{1, k_j}$) j -го правила;

$w_{jp} = [0, 1]$ - весовой коэффициент правила с номером jp ;

$y_j = f(x_1, x_2, \dots, x_n)$ - выход (заключение) нечеткого правила, который может быть описан, например, полиномом вида

$$y_j = b_{j,0} + b_{j,1} \cdot x_1 + b_{j,2} \cdot x_2 + \dots + b_{j,n} \cdot x_n, \quad j = \overline{1, m}$$

В модели (1) каждая входная переменная, оцениваемая соответствующим нечетким термом $a_{i,jp}$, описывается собственными функциями принадлежности (ФП) нечетким термам (например, квантификаторам типа ОН – очень низкий, Н – низкий, НС – ниже среднего, С – средний, ВС – выше среднего, В – высокий, ОВ – очень высокий). В общем виде ФП описываются выражением

$$\tilde{\mu}^k(x_i^j) = \exp\left(-\frac{1}{2} \left(\frac{x_i^j - c_k^j}{s_k^j}\right)^2\right). \quad (2)$$

Здесь c_k^j, s_k^j - параметры ФП, k – индекс терма, j - номер правила.

Для конкретной рассматриваемой задачи формируется обучающая выборка в виде множества пар «вход - выход» с зафиксированными (измеренными) значениями входных параметров $X^* = \{X_q^*\} = \{x_{1q}^*, \dots, x_{nq}^*\}$ и соответствующих им

выходных. $Y^* = \{y_q^*\}$ $q = \overline{1, N}$ - число выборок. С использованием иммунных алгоритмов требуется найти такие значения параметров c_k^j, s_k^j , при которых отклонения текущих фактических значений заключений правил НЛВ (1) от зафиксированных в обучающей выборке эталонных будут наименьшими.

3. Алгоритм кластеризации входных параметров нечетких правил СНВ

Основным назначением этого алгоритма, реализуемого на первом этапе, является сокращение исходного числа входных параметров правил (1) и, соответственно, числа правил, с использованием процедур кластеризации и формирование правил СНВ с предварительными (грубыми) значениями параметров, описывающих их ФП (2). Известные алгоритмы кластеризации K-Means и Expectation Maximization [5] налагают ограничения на геометрию получаемых кластеров, требуя, в частности, возможности охвата каждого кластера отдельным выпуклым множеством. Такое ограничение налагается используемыми этими алгоритмами предположениями о существовании центров кластеров (K-Means) или функции плотности вероятности для каждого кластера с соответствующими значениями математического ожидания и дисперсии (Expectation Maximization). Поэтому эти алгоритмы не в состоянии адекватно разбить на кластеры невыпуклые множества, тем более вложенные структуры.

Эту проблему решает описываемый ниже алгоритм кластеризации конечного набора элементов произвольного метрического пространства на основании разбиения исходного множества на классы эквивалентности по нечеткому отношению. Он позволяет группировать в кластеры элементы, между которыми есть последовательность «близких» друг к другу элементов, что также соответствует интуитивному представлению о группировке.

Алгоритм кластеризации реализуется в следующей последовательности.

Шаг 1. Разделение множества $\{X_1, X_2, \dots, X_N\}$ на непересекающиеся классы эквивалентности. Представим, что нам известны минимальное и максимальное значения каждой входной и выходной информации. По ним можно определить интервалы, в которых находятся допустимые значения. Для входного сигнала x_i такой интервал обозначим через $[x_{iq}^-, x_{iq}^+]$. Если значения x_{iq}^- и x_{iq}^+ неизвестны, то можно воспользоваться обучающими данными и выбрать из них соответственно минимальное и максимальное значения.

Каждый определенный таким образом интервал разделим на K областей (отрезков), причем значение K для каждого сигнала подбирается индивидуально, а отрезки могут иметь одинаковую или различную длину.

Для оценки значений лингвистических переменных будем использовать указанную выше семиуровневую шкалу квантификаторов термов. Каждый из этих термов представляет нечеткое множество, заданное с помощью соответствующей функции принадлежности.

Пользуясь введенными качественными термами (классификаторами) и знаниями эксперта, представим нечеткие правила в виде таблицы, элементами которой являются функции принадлежности термов нечетких правил.

Используя таблицу и операции \wedge (И - min) и \vee (ИЛИ - max), можно легко записать систему нечетких логических уравнений, связывающих функции принадлежности заключений НЛВ и входных переменных.

В общем случае каждая переменная входного вектора $X_q^* = (x_{1q}^*, \dots, x_{nq}^*)$ $q = \overline{1, N}$ имеет свои собственные функции принадлежности нечетким термам (ОН, Н, НС, С, ВС, В, ОВ), которые используются в правилах СНВ. Для упрощения моделирования для всех переменных входного вектора будем использовать только одну – гауссовую форму функций принадлежности.

Шаг 2. Построение нечетких правил на основе обучающих данных.

Вначале определим степени принадлежности обучающих данных к каждой области, выделенной на шаге 1. Эти степени будут выражаться значениями ФП нечетких множеств соответствующей группы данных.

Основными преимуществами рассмотренного алгоритма являются:

- отсутствие необходимости в априорных предположениях относительно структуры данных (вид и параметры распределения вероятности по кластерам, центров плотности, числа кластеров);
- понятная интерпретация результатов разбиения по кластерам: элементы входят в один кластер, когда между ними есть последовательность близких друг к другу элементов;
- отсутствие ограничений на геометрию кластеров. Полученные при помощи описанного алгоритма кластеры могут иметь произвольную геометрическую форму множеств, в том числе и невыпуклую. Это выгодно отличает данный алгоритм от известных алгоритмов кластеризации (например, модификаций алгоритма K-Means, Expectation Maximization и т.д.).

4. Алгоритм идентификации и настройки параметров нечетких правил СНВ

Рассматриваемый алгоритм реализует второй этап общего алгоритма синтеза СНВ. Для идентификации параметров заключений в правилах (1) предлагается использовать следующий иммунно-нечеткий алгоритм:

1. Фиксируются значения входных и выходных параметров состояния объекта:

$$X_q^* = (x_{1q}^*, \dots, x_{nq}^*), Y^* = \{y_q^*\} \quad q = \overline{1, N}.$$

2. Определяются значения функций принадлежности входных параметров $\mu^q(x_i^*)$ при фиксированных значениях вектора $X_q^* = (x_{1q}^*, \dots, x_{nq}^*)$.

3. Вычисляются значения функций принадлежности выходных параметров $\mu^{y_q}(x_{1q}^*, x_{2q}^*, \dots, x_{nq}^*)$ при фиксированных значениях вектора $X_q^* = (x_{1q}^*, \dots, x_{nq}^*)$.

4. Путем обучения с помощью иммунных алгоритмов подбираются такие значения параметров c_k^q, s_k^q функций принадлежности (2), которые минимизируют величину невязки $E_i = y_q^* - y_q$, т.е. различие между зафиксированными реальными значениями выходных параметров объекта (y_q^*) и значениями выходных параметров (y_q), которые формируются на выходе нечеткого вывода, аппроксимирующего правила (1). В результате определяются значения y_q^* , для которого

$$\mu^{y_q^*}(x_1^*, x_2^*, \dots, x_n^*) = \max_{q=1, n} [\mu^{y_q}(x_1^*, x_2^*, \dots, x_n^*)].$$

Для обучения использован иммунный алгоритм.

Алгоритм обучения состоит из двух фаз. На первой фазе вычисляется модельное значение выхода объекта (y), соответствующее заданной архитектуре сети. На второй фазе вычисляется значение невязки (E_i) и пересчитываются параметры функций принадлежности по следующему алгоритму:

1) Генерация выборки экспериментальных данных (X_q, y_q) , $q = 1, 2, \dots, N$, где $X_q = (x_{1q}, x_{2q}, \dots, x_{nq})$ - входной вектор в q -й паре и y_q - соответствующий выход.

2) Нормирование значений параметров вектора $X_q = (x_{1q}, x_{2q}, \dots, x_{nq})$ на $[0, 1]$, где l - значение масштаба:

$$u_i^q = l \frac{x_{iq} - x^{\min}}{x^{\max} - x^{\min}}.$$

3) Фаззификация нормированных значений $X_q = (x_{1q}, x_{2q}, \dots, x_{nq})$:

$$\mu^k(u_i^q) = \exp\left(-\frac{1}{2} \left(\frac{u_i^q - c_q}{s_q}\right)^2\right) \quad c_q = 0, 1, 2, \dots, l.$$

4) Максимизация: $\mu^*(u_i^q) = \max_j \mu^k(u_i^q)$.

5) Вычисление степени истинности выполнения заключения k -го правила:

$$SP^q = \mu^*(u_1^q) \cdot \mu^*(u_2^q) \cdot \dots \cdot \mu^*(u_n^q).$$

6) Нормирование значений SP^q на $[0, 1]$:

$$\eta^q = l \frac{SP^q - SP^{\min}}{SP^{\max} - SP^{\min}}.$$

7) Фаззификация значений η^q :

$$\mu^k(\eta^q) = \frac{1}{1 + \frac{(\eta^q - c_q)}{s_q}}$$

8) Нахождение максимума функции принадлежности:

$$\mu^*(\eta^q) = \max_j \mu^k(\eta^q)$$

9) Формирование совокупности нечетких правил вида (1) для базы знаний:

если $x_1^1 = u_{11}^1 \cap x_2^1 = u_{21}^1 \dots \cap x_n^1 = u_{n1}^1$ или $x_1^2 = u_{11}^2 \cap x_2^2 = u_{21}^2 \dots \cap x_n^2 = u_{n1}^2$, или

$$x_1^{k_q} = u_{11}^{k_q} \cap x_2^{k_q} = u_{21}^{k_q} \dots \cap x_n^{k_q} = u_{n1}^{k_q}$$

то $y_q = b_{q0} + b_{q1}x_1^q + \dots + b_{qn}x_n^q$, $q = \overline{1, N}$.

Необходимо найти такие значения элементов матрицы коэффициентов заключений правил

$$B = \begin{pmatrix} b_{10} & b_{11} & \dots & b_{1n} \\ b_{20} & b_{21} & \dots & b_{2n} \\ - & - & - & - \\ b_{N0} & b_{N1} & \dots & b_{Nn} \end{pmatrix},$$

которые обеспечивают минимум квадратичной невязки:

$$E = \sum_{q=1}^N (y_q - y_q^f)^2 \rightarrow \min,$$

где y_q^f - результат вывода по нечеткой модели, y_q - эталонное значение выхода объекта.

5. Применение иммунного алгоритма для обучения нечеткой модели

Пусть задана выборка нечетких экспериментальных данных (X_r, y_r) , $r = \overline{1, M}$; здесь $X_r = (x_{r1}, x_{r2}, \dots, x_{rn})$ - входной n -мерный вектор и $y_r = (y_1, y_2, \dots, y_M)$ - соответствующий ему выходной вектор.

В общем виде требуется построить модель, основанную на нечетких правилах вывода (1):

$$\bigcup_{p=1}^{k_j} \left(\bigcap_{i=1}^n x_i = a_{i,jp} - \text{с весом } w_{jp} \right) \rightarrow y_j = b_{m0} + b_{m1}x_1^j + \dots + b_{mn}x_n^j.$$

В процессе построения модели нужно найти такие значения коэффициентов правил

$$B = (b_{ij}), i = \overline{1, m}, j = \overline{0, n},$$

при которых достигается минимум следующего выражения:

$$\sum_{r=1}^M (y_r - y_r^f) \rightarrow \min,$$

где y_r^f - результат нечетких правил вывода с параметром B в r -й строке выборки (X_r) .

Входной матрице X_r соответствует следующий результат нечеткого вывода:

$$y_r^f = \frac{\sum_{i=1}^m \mu_{d_i}(X_r) \cdot d_i}{\sum_{i=1}^m \mu_{d_i}(X_r)}$$

здесь $d_i = b_{i0} + b_{i1}x_{r1} + b_{i2}x_{r2} + \dots + b_{in}x_{rn}$ - выход i -правила; $\mu_{d_i}(X_r)$ - функция принадлежности, соответствующая каждой экспериментальной информации:

$$\begin{aligned} \mu_{d_j}(X_r) &= \mu_{i1}(x_{r1}) \cdot \mu_{i1}(x_{r2}) \cdot \mu_{i1}(x_{r3}) \cdot \dots \cdot \mu_{i1}(x_{rn}) \vee \\ &\vee \mu_{i2}(x_{r1}) \cdot \mu_{i2}(x_{r2}) \cdot \mu_{i2}(x_{r3}) \cdot \dots \cdot \mu_{i2}(x_{rn}) \vee \\ &\dots \dots \dots \vee \\ &\vee \mu_{im}(x_{r1}) \cdot \mu_{im}(x_{r2}) \cdot \mu_{im}(x_{r3}) \cdot \dots \cdot \mu_{im}(x_{rn}). \end{aligned}$$

Функция принадлежности элемента x_i терму a_i^{jp} имеет следующий вид:

$$\mu^{jp}(x_i) = \exp \left(-\frac{1}{2} \left(\frac{a_i^{jp} - b_i^{jp}}{c_i^{jp}} \right)^2 \right),$$

где b_i^{jp}, c_i^{jp} - параметры функции принадлежности.

Для решения задачи (2) используем иммунный алгоритм.

Иммунные алгоритмы (ИА) оптимизации содержат следующие операторы: клонирование, мутация, старение и селекция. Рассмотрим их подробнее.

Алгоритм настройки параметров функций принадлежности $B = (b_1, b_2, \dots, b_q)$ и $C = (c_1, c_2, \dots, c_q)$ и веса правил $W = (w_1, w_2, \dots, w_N)$ состоит из следующих этапов.

1. Формирование исходной популяции, которое реализуется оператором клонирования. Генерирует новое поколение копий антител в будущей популяции. Известны следующие основные операторы клонирования: а) статический оператор клонирования, который просто копирует каждую В-клетку, производя переходную популяцию; б) пропорциональный оператор клонирования, который копирует В-клетки пропорционально их антигенной схожести; в) оператор вероятностного клонирования, в соответствии с которым В-клетки выбираются из текущего поколения в зависимости от вероятности клональной селекции [30].

Для реализации ИА следует задать способ кодирования нечетких моделей. Сведем неизвестные параметры W, B, C в один вектор [31,32]:

$$S = (W, B, C) = (w_1, w_2, \dots, w_N,$$

$$b_{11}, c_{11}, \dots, b_{l_1}, c_{l_1}, b_{n1}, c_{n1}, \dots, b_{n l_n}, c_{n l_n}),$$

где N - общее число строк в нечеткой базе знаний; l_i - количество термов-оценок входной переменной λ_i ,

$$l_1 + l_2 + \dots + l_n = q, i = \overline{1, n};$$

q - общее число термов.

2. Мутация, которая реализуется оператором мутации, действует в зависимости от имеющейся популяции клонов, применяя к каждому антителу

определенное количество одиночных мутаций, осуществляемых случайным образом.

Каждый элемент вектора S может подвергнуться операции мутации с вероятностью p_m . Обозначим мутацию элемента s через $Mu(s)$:

$$Mu(w_j) = RANDOM([\underline{w}, \overline{w}]), j = \overline{1, N},$$

$$Mu(b_p) = RANDOM([\underline{x}_i, \overline{x}_i]),$$

$$Mu(c_p) = RANDOM([\underline{c}_i, \overline{c}_i]),$$

где $\underline{w}, \overline{w}$ - нижняя (верхняя) граница интервала возможных значений весов правил, $[\underline{w}, \overline{w}] \subset [0, 1]$;

$[\underline{\tilde{n}}_i, \overline{c}_i]$ - интервал возможных значений коэффициента концентрации-растяжения функций принадлежности термов-оценок входной переменной x_i , $[\underline{\tilde{n}}_i, \overline{c}_i] \subset (0, +\infty]$, $i = \overline{1, n}$;

$RANDOM([\underline{\xi}, \overline{\xi}])$ обозначает операцию нахождения равномерно распределенного на интервале $[\underline{\xi}, \overline{\xi}]$ случайного числа.

3. Устранение старых особей, которое реализуется оператором старения. Статический оператор старения использует возрастной параметр для максимального количества поколений антител, которым разрешено оставаться в популяции. Когда антитело старше, оно удаляется из системы, даже если оно может оказаться вполне пригодным на последующих итерациях.

При клональной экспансии клонированное антитело наследует возраст его родителя. После этапа мутации только те антитела, которые получили высшее значение аффинности, получают возраст, равный 0. Элитный вариант этого оператора получается путем взятия наилучших антител популяции в поколение с возрастом, равным 0. При этом берется лучшее антитело из популяции в поколение.

4. Замена наихудших антител в популяции новыми случайными антителами, которая реализуется оператором селекции.

Для проведения сравнительного анализа взяты известные модельные задачи, расположенные по электронному адресу:

<http://www.ics.uci.edu/~mllearn/databases/>. К ним относятся следующие задачи: задача ирис (Iris Data Set), задача идентификации стекла (Glass Identification Data Set), задача диагностирования диабета (Pima Indians Diabetes), экологическая задача (Ecoli Data Set), задача Хабермана (Haberma's Survival Data Set), задача определения сорта вина (Wine Data Set), задача определения состояния печени (Liver).

В табл. 1 приводятся параметры перечисленных задач. В табл. 2 для сравнения приведены результаты решения некоторых модельных задач в процентном соотношении на основе различных известных и предлагаемых алгоритмов.

Таблица 1

Параметры модельных задач

Наименование задачи	Количество классов	Количество признаков	Количество объектов
Стекло (Glass)	7	9	214
Хаберман (Haberma)	2	4	306
Ирис (Iris)	3	4	150
Экология (Ecoli)	8	7	336
Вино (Wine)	3	13	178
Печень (Liver)	2	6	345

Таблица 2

Результаты работы предлагаемого и существующих алгоритмов

Задача	Предлагаемый алгоритм	GBC	SGF	SVM	INN	KNN	Conventional RBF network
Стекло (Glass)	87,2	84,27	75,74	71,50	72,01	72,01	69,16
Ирис (Iris)	98,5	98,00	97,33	97,33	96,00	95,33	95,33
Вино (Wine)	100	100	99,44	99,44	95,52	96,07	98,89

5. Заключение

Таким образом, показана целесообразность объединения метода нечеткого вывода и иммунных алгоритмов в задачах с параметрами, содержащими неопределенности различного типа, а также в задачах, для которых характерны интуитивные решения. Предложенный метод позволяет существенно улучшить качество решения задач

мониторинга с нечетко заданными параметрами и критериями. В дальнейшем планируется изучение различных гибридных методов применительно к задачам мониторинга, а также методов автоматического формирования базы нечетких правил. Это позволит существенно повысить эффективность решений задач рассматриваемого класса.

Литература

- [1] *Заде Л.А.* Понятие лингвистической переменной и его применение к принятию приближенных решений. - М.: Мир, 1976. - 165 с.
- [2] *Алиев Р.А., Алиев Р.Р.* Теория интеллектуальных систем и ее применение. – Баку: Изд-во Чашыюглы, 2001. – 720 с.
- [3] *Борисов А.Н., Крумберг О.А., Федоров И.П.* Обработка нечеткой информации в системах принятия решений. - М.: Радио и связь, 1989. - 304 с.
- [4] *Пегат А.* Нечеткое моделирование и управление. – М.: БИНОМ. Лаборатория знаний, 2009. – 798 с. (серия «Адаптивные и интеллектуальные системы»).
- [5] *Забродин В.Ю.* О критериях естественной классификации // НТИ. - Сер. 2. – 1981. - № 8.
- [6] *Витяев Е.Е.* Классификация как выделение групп объектов, удовлетворяющих разным множествам согласованных закономерностей // Анализ разнотипных данных (Вычислительные системы - 99). - Новосибирск, 1983. - С. 44-50.
- [7] *Демин А.В., Витяев Е.Е.* Логическая модель адаптивной системы управления // Нейроинформатика. – 2008. – Т. 3, № 1. - С. 79-107.
- [8] *Vityaev E.E., Lapardin K.A., Khomicheva I.V., Proskura A.L.* Transcription factor binding site recognition by regularity matrices based on the natural classification method. Intelligent- Data Analysis. Special issue: “New Methods in Bioinformatics. Presented at the fifth International Conference on Bioinformatics of Genom Regulation and Structure” eds. Evgenii Vityaev and Nikolai Kolchanov. - V.12(5), IOS Press, 2008. - Pp. 495-512.
- [9] *Vityaev E.E., Kostin V.V., Podkolodny N.A., Kolchanov N.A.* Natural classification of nucleotide sequences // Proc. of the Third International Conference On Bioinformatics of Genome Regulation and Structure (BGRS'2002, Novosibirsk, Russia, July 14-20, 2002). - V3, ICG, Novosibirsk, 2002. - Pp. 197-199.
- [10] *Штовба С.Д.* Введение в теорию нечетких множеств и нечеткую логику. <http://www.matlab.exponenta.ru>.
- [11] *Bekmuratov T.F., Muhamedieva D.T., Bobomuradov O.J.* Model prediction of yield initial conditions. Ninth International Conference on Application of Fuzzy Systems and Soft Computing. ICAFS – 2010. Edited by R.A. Aliev, K.W. Bonfig, M. Jamshidi, W. Pedrycz, I.B. Turksen. b – Quadrat Verlag. Prague, Czech Republic. August 26-27, 2010. – Pp. 321-328.
- [12] *Hopfield J.J., Tank D.W.* “Neural” computation of decisions in optimization problems // Biological Cybernetics, 1985. - Vol. 52, № 3. - Pp. 141-152.
- [13] *Hung D.L., Wang J.* Digital hardware realization of a recurrent neural network for solving the assignment problem // Neurocomputing, 51, 2003. - Pp. 447-461.
- [14] *Holland J.H.* Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence. - London: Bradford book edition, 1994. - 211 p.
- [15] *Bryant K., Benjamin A.* Genetic Algorithms and the Traveling Salesman Problem, Department of Mathematics, HarveyMudd College, 2000.
- [16] *Махотило К.В., Петрашев С.Н., Сергеев С.А.* Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. – Харьков: ОСНОВА, 1997. – 112 с.
- [17] *Cantu-Paz E.* Efficient and Accurate Parallel Genetic Algorithms, Lawrence Livermore National Lab, 2000.
- [18] *Dorigo Marco, Stutzle T.* Ant colony optimization. – Cambridge: The MIT Press, 2004. – 305 p.
- [19] *Craig Reynolds.* Flocks, Herds, and Schools: A Distributed Behavioral Model // Computer Graphics, 21(4), 1987. – Pp. 25–34.
- [20] *Kennedy J., Eberhart R.C.* Particle swarm optimization // In Proceedings of IEEE International Conference on Neural Networks. – 1995. – Pp. 1942-1948.
- [21] *Heppner F., Grenander U.* A stochastic nonlinear model for coordinated bird flocks // The Ubiquity of Chaos, 1990. - Pp. 233–238.
- [22] *Eberhart R.C., Kennedy J.* A new optimizer using particle swarm theory // Proceedings of the Sixth International Symposium on Micro Machine and Human Science MHS'95. – 1995. - Pp. 39-43.
- [23] *Shi Y., Eberhart R.* A modified particle swarm optimizer // The 1998 IEEE International Conference on Evolutionary Computation Proceedings. – 1998. - Pp. 69-73.
- [24] *Shi Y., Eberhart R.* Empirical study of particle swarm optimization // Proceedings of the 1999 IEEE Congress on Evolutionary Computation. – 1999. - Pp. 1945-1950.
- [25] *Clerc M., Kennedy J.* The particle swarm – explosion, stability, and convergence in a multidimensional complex space // IEEE Transactions on Evolutionary Computation. – 2002. - № 6 (1). - Pp. 58-73.
- [26] *Mendes R., Kennedy J., Neves J.* The fully informed particle swarm: Simpler, maybe better // IEEE Transactions on Evolutionary Computation. – 2004. - № 8. - Pp. 204-210.
- [27] *Ibrahim Z., Tsuboi Y., Ono O. and Khalid M.* Dir ect-proportional length-based DNA computing for shortest path problem, International Journal of Computer Sciences & Applications. - 2004. - Vol.1, № 1. - Pp. 46-60.
- [28] *Yamamoto M., Matsuura N., Shiba T., Kawazo Y. and Ohuchi A.* DNA solution of the shortest path problem by concentration control, Lecture Notes in Computer Science. - 2004. - Vol.2340. - Pp.23-212.

- [29] *Zhang F., Liu B., Liu W. and Xu J.* A DNA computing model based on acrydite™ gel technology to solve the timetable problem, IEEE/ICME International Conference on Complex Medical Engineering. - 2007. - Pp.1632-1635.
- [30] *Дасгунта Д.* Искусственные иммунные системы и их применение. – М.: Физматлит, 2006. - 344 с.
- [31] *Dasgupta D.* Artificial Immune Systems and Their Applications, Springer-Verlag, 1998.
- [32] *Мухамедиева Д.Т.* Иммунный алгоритм решения задач классификации и прогнозирования в нечеткой среде // Вестник ТУИТ. – Ташкент, 2012. - Вып.1. - С. 38-41.