

---

---

# МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ

УДК 004.89.896

## НЕЙРОЭВОЛЮЦИОННЫЙ МЕТОД ИНТЕЛЛЕКТУАЛИЗАЦИИ ПРИНЯТИЯ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

*Статья поступила в редакцию 10.07.2015, в окончательном варианте 25.08.2015.*

*Хлопкова Ольга Андреевна*, аспирант, Московский государственный университет экономики, статистики и информатики, 119501, Российская Федерация, г. Москва, ул. Нежинская, 7, e-mail: ohlopkova@nifi.ru

В статье представлен адаптивный метод интеллектуализации принятия решений в условиях неопределенности, неполной информации и динамической среды. В отличие от существующих нейроэволюционных методов принятия решений, таких как GNARL, ENS3, NEAT (и его модификаций, в том числе HyperNEAT), предлагаемый метод применим для оптимизации как структуры, так и параметров нейросетей любой топологии. Он лишен типичных недостатков, связанных с использованием нейроэволюционного подхода: конкурирующих представлений; незащищенных инноваций; проблемы начального размера. Предлагаемый метод позволяет снизить риск преждевременной сходимости и «паралича» нейронной сети, обеспечивает исключение остановки вычислительного процесса на локальных оптимумах. Автором описаны основные аспекты предлагаемого метода: прямой способ генетического кодирования и определяемая им структура хранения данных; параметры хромосом; вид функции приспособленности; алгоритмы селекции, мутации и рекомбинирования. Дано теоретическое обоснование введения этапа локальной оптимизации, наделяющего метод свойствами меметичности. Приведены результаты успешного применения метода для решения классических нейроэволюционных задач, в том числе задач адаптивного управления, реализации логических функций, трудноформализуемых задач восстановления поврежденных данных. Эффективность метода доказана результатами его практического использования в задачах защиты информационной системы от DDoS-атак, рейтинговой оценки размещаемых в базе знаний материалов; оптимизации работы систем 3D-печати.

**Ключевые слова:** генетические алгоритмы, нейронные сети, искусственный интеллект, нейроэволюция, меметичные алгоритмы, системы поддержки принятия решений, неопределенность, эволюционные вычисления, гибридные интеллектуальные системы, TWEANN, COGANN, 3D-принтеры

## NEUROEVOLUTIONAL METHOD OF DECISION MAKING INTELLECTUALIZATION IN CONDITIONS OF INDETERMINACY

*Khlopkova Olga A.*, post-graduate student, Moscow State University of Economics, Statistics and Informatics, 7 Nezhinskaya St., Moscow, 119501, Russian Federation, e-mail: ohlopkova@nifi.ru

This article presents the adaptive method of intellectualization of decision support systems under conditions of indeterminacy, incomplete information and dynamic environment. As distinct from existing neuroevolutional methods such as GNARL, ENS3, NEAT (and its modifications including HyperNEAT), the method is applicable for the optimization of both the structure and parameters of the neural networks of any topology. It is deprived of the main neuroevolution-related drawbacks such as competing conventions, unprotected innovations, initialization and topology minimization, topological innovations. This method reduces the risk of premature convergence, network paralysis and solves the local minima problem. This article describes the main aspects of the method: the scheme structure defined by the direct encoding, chromosomes parameters, fitness function definition and algorithms of selection, mutations and recombination. The article introduces the theoretical rationale for the memetic phase of the parameters and structure local tuning. The article gives examples of successful application of the method to classical neuroevolutional problems such as

adaptive control, constructing of logical functions, damaged data recovery. The method proved its efficiency in appliance to practical problems of information system protection from DDoS-attacks, rating calculation of materials in knowledge base and improvement of the 3D-printing systems.

**Keywords:** genetic algorithms, neural networks, artificial intelligence, neuroevolution, memetic algorithms, decision support systems, indeterminacy, evolutionary computation, hybrid intelligent systems, TWEANN, COGANN, 3D-printers

**Введение.** Методы нейроэволюции служат для самоорганизации искусственных систем, автоматизации решения оптимизационных задач и наделения программных комплексов принятия решений адаптивным поведением. Они нашли применение при разработке большого числа средств управления динамическими нелинейными системами в условиях неопределенности и наличия лишь ограниченной информации.

Самоорганизующиеся модули демонстрируют высокие показатели качества в системах принятия решений (СПР), оперирующих неполными, трудноформализуемыми и неточными данными. Нейроэволюционные методы принятия решений все чаще применяются в предметных областях, где отсутствует физическое или статистическое понимание поведения системы, наблюдается большой статистический разброс входных данных; нелинейность механизма их обобщения. Несмотря на активные исследования в рассматриваемой области, пока остается открытым вопрос создания эффективной искусственной нейронной сети (ИНС) с динамической структурой, обладающей высокими показателями точности вывода, скорости обучения и функционирования, надежности и масштабируемости.

В связи с этим целью данной статьи является разработка формализованного метода, лишенного основных недостатков, присущих большинству современных нейроэволюционных методов.

**Общая характеристика структуры хранения данных, используемая в предлагаемом методе.** Нейроэволюция – форма настройки нейросети с использованием эволюционных алгоритмов. Как показывает практика, эволюционные методы настройки топологии ИНС и ее параметров TWEANN [9] (Topology and Weight Evolving Artificial Neural Network) позволяют сократить количество примеров в обучающей выборке и количество нейронов сети, оптимизировать топологию. К основным проблемам, связанным с практическим применением COGANN (Combination Of Genetic Algorithms And Neural Networks, сочетание генетических алгоритмов и нейронных сетей) [9], относятся следующие: конкурирующие представления; незащищенные инновации; проблемы начального размера и топологических инноваций; преждевременной сходимости; «паралича» ИНС и необходимости «преодоления» локального оптимума [18].

*Конкурирующие представления (проблема перестановок)* – это проблема, связанная с отсутствием биекции «генотип – фенотип», при котором один и тот же фенотип топологически может быть представлен в генотипе несколькими способами. Применение оператора кроссинговера для двух схожих фенотипов с большой вероятностью генерирует поврежденную ИНС с повторяющимися или отсутствующими элементами.

*Незащищенные инновации* – это явление отсеивания в ходе нейроэволюции вариантов, которые являются удачными, но их первые несколько эпох демонстрируют низкую приспособленность из-за ненастроенных весовых коэффициентов и параметров функции активации.

*Проблема начального размера и топологических инноваций* во многих методиках возникает из-за того, что начальная популяция формируется набором случайных топологий. При этом затрачивается время на отсеивание заведомо нежизнеспособных особей, а сформированные популяции имеют тенденцию преждевременной сходимости к решениям, размер которых не оптимален. *Преждевременная сходимость* означает остановку генетическо-

го алгоритма до достижения им глобального оптимума. «Паралич» нейросети – неконтролируемый рост значений весовых коэффициентов в процессе обучения и, как следствие, активизация нейронов только при очень больших входных значениях.

Большинство существующих на данный момент алгоритмов применимы только для нейросетей определенного типа. Например, метод GNARL [13] работает только с рекуррентными ИНС. В методе ENS3 [9] заложены ограничения на вид функции активации. Метод HyperNEAT[15] не предназначен для реализации эволюции топологии.

Такая ситуация определяет целесообразность и востребованность разработки достаточно универсального метода, допускающего гибкую «адаптацию» к решаемым задачам. При этом эволюция архитектуры и топологии нейросети осуществляется на протяжении всего процесса ее функционирования. Направление эволюции зависит от изменений свойств динамически управляемого объекта: качественные и количественные изменения могут идти как по пути усложнения, так и упрощения. Экспоненциальное снижение быстродействия нейросети с ростом количества ее элементов препятствует широкому распространению больших нейросетей. Такие сети сложно проектировать и настраивать. Эволюционная оптимизация количества нейронов и слоев упрощает поиск компромисса между скоростью достижения результата и его точностью.

В качестве структуры хранения данных для генетического кодирования в предлагаемом методе был выбран список смежности. Размер занимаемой им памяти линейно зависит от количества нейронов ( $N$ ) и количества синаптических связей между ними  $S$  – как  $O(N + |S|)$ . Компактность данной структуры открывает возможность оперирования топологией глубоких ИНС с экстремально большим, порядка десятков тысяч, количеством нейронов. Список смежности ставит в соответствие каждому  $i$ -ому прондексированному нейрону список пар вида  $\{s_{ij}, w_{ij}\}$ , где  $s_{ij}$  – индекс нейрона, связанного синапсом с  $j$ -ым нейроном, а  $w_{ij}$  – вес этой связи. Для входных нейронов этот список пуст. Для каждого нейрона список смежности хранит также пороговое значение  $\theta$ , вид и параметры функции активации  $f$ . Важным свойством описываемой структуры является избыточность. Помимо перечисленных параметров она содержит следующую информацию:  $IN_i$  – минимальное количество узлов ИНС, через которые должен пройти входной сигнал, чтобы достигнуть  $i$ -го нейрона;  $OUT_i$  – минимальное количество узлов ИНС, через которые должен пройти сигнал, сгенерированный  $i$ -ым нейроном, чтобы достигнуть выхода.

Хранение и пересчет этих параметров способствуют решению проблем *конкурирующих представлений* и *незащищенных инноваций*, свойственных схеме прямого кодирования. Использование прямого кодирования подразумевает фиксированное взаимно-однозначное соответствие между фенотипом, образующим структурные элементы ИНС, и соответствующими участками хромосомы. Биекция «фенотип – генотип» позволяет отслеживать изменения популяции и причины этих изменений; применять к хромосомам уже имеющийся аппарат генетического поиска. Из наиболее очевидных минусов такой схемы можно отметить зависимость длины генотипа от количества нейронов и связей. Следствием является низкая эффективность – за счет увеличения пространства поиска для больших и сверхбольших ИНС.

При косвенном кодировании в генотипе содержатся только правила построения фенотипа. Несмотря на свойственную косвенному кодированию компактность представления, в изложенном методе реализована схема прямого кодирования. Это обусловлено недостатками косвенного кодирования: низкой производительностью и медленной сходимостью; практической невозможностью отслеживания изменений генотипа, послуживших причинами заданных изменений фенотипа; ограничениями, накладываемыми на генетические операторы. Каждый  $i$ -ый элемент списка смежности имеет вид:  $\{i, \{s_{i0}, w_{i0}\}, \dots, \{s_{iM}, w_{iM}\}, \theta, f_{type}, \{p_0, \dots, p_K\}, IN_i, OUT_i\}$ .

Нейросети любой топологии и размерности кодируются при помощи данного универсального представления. На рисунке 1 приведен пример структуры простой рекуррентной ИНС с контекстом и единственным скрытым слоем.

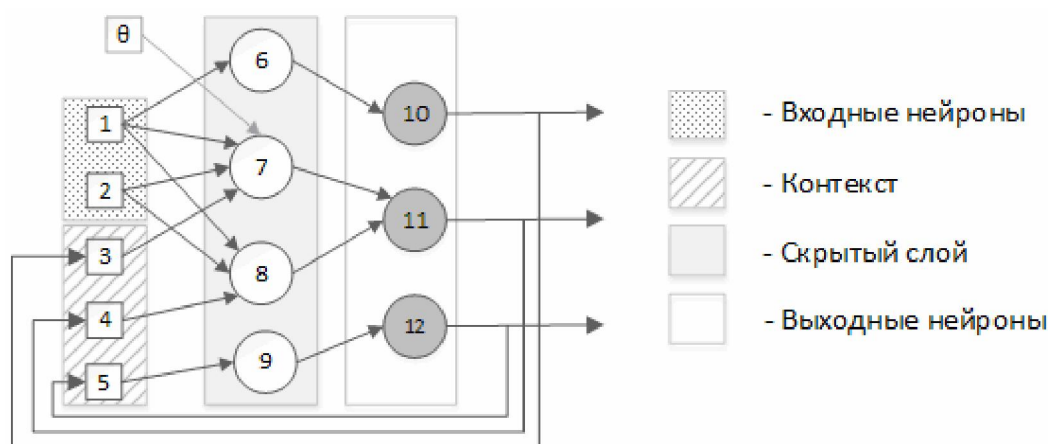


Рис. 1. Рекуррентная ИНС с единственным скрытым слоем

Структуре такой нейросети соответствует список смежности, представленный в таблице 1. Нейрону с индексом  $i$  сопоставлены вектор пар  $\overline{\{s_{ij}, w_{ij}\}}$  индексов и весовых коэффициентов, порог  $\theta$ , вид функции активации  $f_{type}$ , вектор параметров  $\overline{p}$ , значения  $IN_i$  и  $OUT_i$  для расчета минимального расстояния до входа и выхода ИНС.

Таблица 1

Список смежности рекуррентной ИНС, представленной на рисунке 1

| $i$ | $\overline{\{s_{ij}, w_{ij}\}}$ | $\theta$ | $f_{type}$ | $\overline{p}$   | $IN_i$ | $OUT_i$ |
|-----|---------------------------------|----------|------------|------------------|--------|---------|
| 1   | –                               | 0        | $mth$      | 1,5; 3; 1,68; 2  | 0      | 2       |
| 2   | –                               | 0        | $mth$      | 1,7; 2,5; 1,5; 2 | 0      | 2       |
| 3   | {10; 0,45}                      | 0        | $sigm$     | 1,07             | 2      | 2       |
| 4   | {11; 0,8}                       | 0        | $sigm$     | 2                | 2      | 2       |
| 5   | {12; 0,91}                      | 0        | $sigm$     | 0,89             | 2      | 2       |
| 6   | {1; 0,21}                       | 0        | $th$       | 2,01             | 1      | 1       |
| 7   | {1; 0,5}, {2; 0,42}, {3; 0,7}   | -0,3     | $th$       | 1,68             | 1      | 1       |
| 8   | {1; 0,1}, {2; 0,3}, {4; 0,8}    | 0        | $th$       | 2                | 1      | 1       |
| 9   | {5; 1}                          | 0        | $th$       | 2                | 1      | 1       |
| 10  | {6; 0,98}                       | 0        | $th$       | 2                | 2      | 0       |
| 11  | {7; 0,54}, {8; 0,5}             | 0        | $th$       | 1,3              | 2      | 0       |
| 12  | {9; 0,75}                       | 0        | $th$       | 1,3              | 2      | 0       |

**Особенности реализации генетического алгоритма.** Оператор кроссингвера (скрещивания) необходим для смешивания генетического материала хромосом-предков и формирования на его основе хромосомы-потомка. Так как хромосомы, подвергающиеся кроссингверу, предварительно проходят этап отбора, можно ожидать, что на каждой итерации алгоритма приспособленность хромосом-потомков возрастает. Кроме того, кроссингвер привносит в популяцию разнообразие, так как генерирует новые, ранее не встречавшиеся решения.

Пусть необходимо применить оператор одноточечного кроссинговера к скрытым слоям двух ИНС, изображенных на рисунке 2:  $[4, 5, 6, 7] \times [7, 4, 5, 6]$ . Одним из возможных результатов является комбинация  $[4, 5, 5, 6]$ , в которой недостает нейрона 7 и продублирован нейрон 5. То есть отсутствует один из функциональных компонентов ИНС. Данный пример иллюстрирует проблему *конкурирующих представлений*, приводящую к появлению в популяции нежизнеспособных особей и значительному снижению результативности работы нейроэволюционного алгоритма.

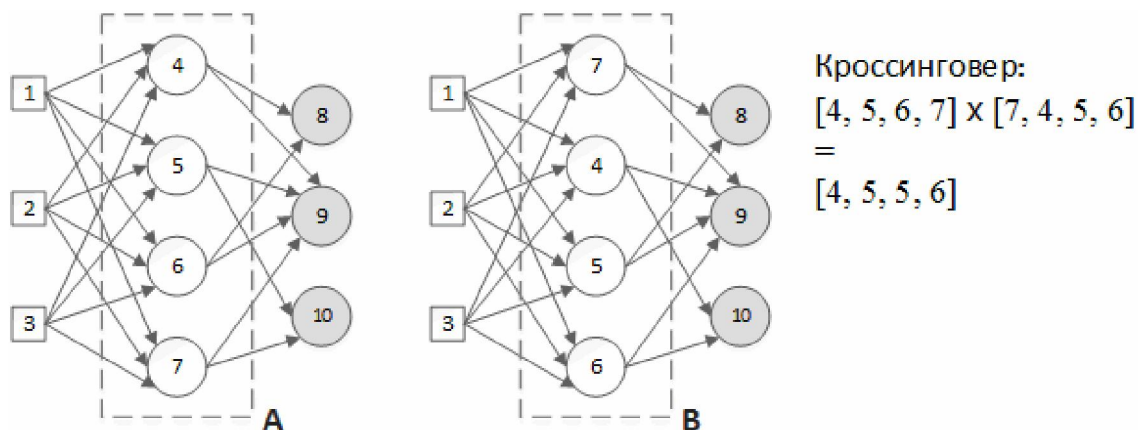


Рис. 2. Иллюстрация проблемы конкурирующих представлений

Индексация узлов ИНС и использование параметров  $IN_i$  и  $OUT_i$  позволяют успешно решить проблему *конкурирующих представлений*. Во-первых, новые узлы, добавляемые в ИНС операторами мутации и кроссинговера, обладают большими индексами, чем узлы, присутствовавшие в ИНС в предыдущее эпохи нейроэволюции. Во-вторых, наличие сведений о кратчайшем пути к входным и выходным узлам препятствует скрещиванию участков нейросети, несущих разную функциональную нагрузку. Например, становится невозможным кроссинговер между  $S$ -плоскостями и  $S$ -плоскостями неокогнитрона; между слоями свертки и подвыборки во сверточной ИНС или слоями параллельных подсетей Сиамской ИНС [6].

Мутация в измененных за предыдущие эпохи участках ИНС является менее вероятной, чем в не эволюционировавших участках. Индексирование узлов снижает риск удаления из популяции новых элементов, что решает проблему *незащищенных инноваций*. На эффективность нейроэволюции непосредственно влияют подбор генетических операторов и вид функции приспособленности, описанные в рамках соответствующих этапов эволюционного алгоритма (рис. 3).

Рассмотрим эти этапы более подробно.

1. Инициализация. Формирование начальной популяции  $P^0$  с достаточным разнообразием особей. Проблемы *начального размера* и *топологических инноваций* решаются ограничением на количество нейронов нейросетей нулевой эпохи. Для списка смежности реализована операция удаления нейронов. А функция приспособленности, накладывающая штраф на размер ИНС, гарантирует, что в ходе эволюции количество узлов нейросети не будет строго возрастающим.

2. Преобразование генотипа в фенотип: декодирование хромосом особей в множество ИНС.

3. Оценка приспособленности. Для функции приспособленности  $F(x)$  в пространстве поиска  $X$  требуется найти такое значение аргумента  $x^*$ , при котором  $F(x)$  достигнет своего

наибольшего значения:  $x^* = \arg \max F(x), x \in X$ . Функция приспособленности  $F_i$  особи  $i$  в эпоху  $t$  вычисляется по формуле:

$$F_i^t = T_i E_i^t N_i^t M_i^t (T_{\max} N_{\max}^t M_{\max}^t \rho_{\min}(i, j))^{-1},$$

где  $T_i$  – количество эпох эволюции данной особи;  $T_{\max}$  – максимально возможное количество эпох;  $E_i^t$  – средняя квадратическая ошибка вывода нейросети  $i$ ;  $N_i^t$  – количество нейронов данной особи;  $N_{\max}^t$  – максимальное количество нейронов особей эпохи  $t$ ;  $M_i^t$  – количество связей между нейронами данной особи;  $M_{\max}^t$  – максимальное количество связей между нейронами особей (вместе с  $N_i^t$  параметры  $N_{\max}^t$  и  $M_i^t$  формируют штраф на размер ИНС);  $\rho_{\min}(i, j)$  – минимальное расстояние между  $i$ -ой хромосомой и другими хромосомами популяции (штраф для схожих генотипов).

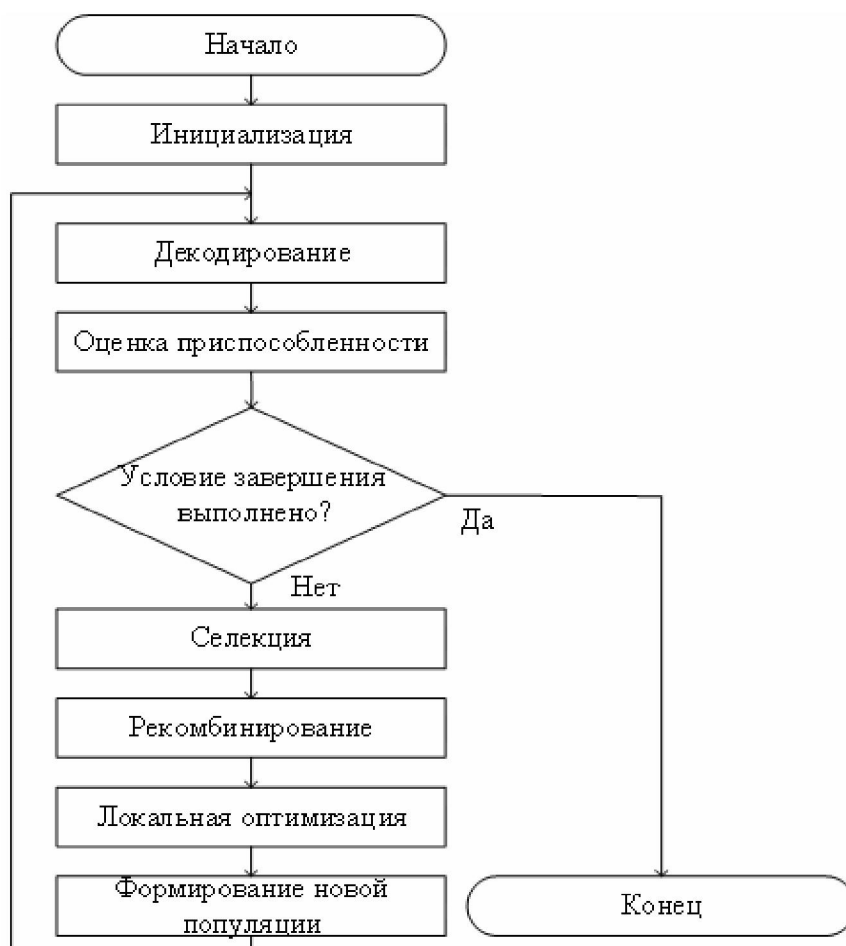


Рис. 3. Блок-схема организации процесса нейроэволюции

Для предотвращения преждевременной сходимости и ситуации, в которой средние и наилучшие особи формируют примерно одинаковое количество потомков. Значение  $F$  масштабируется по следующей формуле:

$$\tilde{F} = (F + F_{avg} - c\sigma)^{\mathcal{G}},$$

где  $F_{avg}$  – средняя приспособленность популяции;  $c = const \in [1,5]$ ;  $\sigma$  – среднее квадратичное отклонение приспособленностей по популяции;  $\mathcal{G} = 1 + (2\tau^2 - 1)^{-1}$ , здесь  $\tau \in [-0.1, 0.5]$  – число, подбираемое с учетом специфики задачи.

4. Проверка условия завершения эволюции. Независимыми друг от друга условиями завершения могут быть следующие: по количеству эпох; по исчерпанию времени эволюции (или количества обращений к функции оптимизации); по достижению наилучшей комбинации генов; по выходу функции приспособленности на «плато», то есть по отсутствию ее изменения в течение заданного количества эпох. Если выполнено любое из этих условий, то алгоритм завершает работу. Иначе выполняется следующий шаг.

5. Селекция. Оператор отбора хромосом  $SL$  (*selection*) для новой популяции реализован вероятностным методом [9] в сочетании с методом «элит» [9]. Наиболее удачные особи заносятся в пул «хороших» решений, остальные выбираются для рекомбинирования с вероятностью  $P_{SL}$ :

$$P_{SL}(i) = F_i / \left( \sum_{j=1}^N F_j \right).$$

Пул «хороших» решений необходим для поддержания разнообразия популяции и предотвращения быстрой сходимости алгоритма к неоптимальному решению (локальному оптимуму). Хромосома для фенотипа  $p^*$  запоминается в пуле при выполнении условий:

$$(\forall p \in P)((\rho(p, p^*) = \rho_{max}) \cap (F(p^*) \in F^*) \cup (F(p^*) = F_{max})),$$

где  $\rho_{max}$  – максимальное в данную эпоху расстояние между особями;  $F_{max}$  – максимальная приспособленность.

6. Рекомбинирование – применение генетических операторов (кроссинговера и мутации) к отобраным на предыдущем шаге особям.

*Кроссинговер*  $CR$  (*crossingover*) – генетический оператор, влияющий на размер популяции. Автором используется *многоточечный кроссинговер* [9]. В общем случае для  $D$  точек пересечения результатом скрещивания особей  $p_i^t = \{a_1, a_2, \dots, a_n\}$ ,  $p_j^t = \{b_1, b_2, \dots, b_n\} \in P^t$  эпохи  $t$  станут два генотипа последующей популяции  $t+1$ :

$$p_i^{t+1} = \{a_1, \dots, a_{d_1}, b_{d_1+1}, \dots, b_{d_2}, a_{d_2+1}, \dots, a_{d_D}, \dots\},$$

$$p_j^{t+1} = \{b_1, \dots, b_{d_1}, a_{d_1+1}, \dots, a_{d_2}, b_{d_2+1}, \dots, b_{d_D}, \dots\} \in P^{t+1}.$$

Точки пересечения  $d_k, k \in [1, D]$  выбираются случайным образом.

*Мутация* (*mutation*)  $MT$  – диссипативный компонент эволюции, «выводящий» популяцию из локальных экстремумов, привносящий разнообразие и новую информацию в генофонд.

Мутация оперирует отдельными генами особи, среди которых есть числовые и номинальные признаки. Оператор  $MT$  реализует следующие функции: добавление (последовательное либо параллельное) или удаление нейрона; добавление или удаление связи между нейронами; изменение параметров нейрона (значения порога  $\theta$ , вида функции активации  $f(S)$  и ее параметров). Вероятность мутации отдельного гена  $P(g_i)$  эпохи  $t$  рассчитывается следующим образом:

$$P(g_i) = (\alpha \cdot e^{\gamma} (\beta L)^{-1})^{0.5} N^{-1},$$

где  $\alpha, \beta, \gamma = const$ ;  $L$  – длина хромосомы;  $N$  – размер популяции.



Выбор формулы данного вида обусловлен тем, что для эффективной эволюции вероятность мутации должна зависеть от размера популяции обратно пропорционально. Одной из основных целей оператора мутации является поддержание разнообразия особей. Однако при этом на маленьких популяциях частые мутации негативно сказываются на сходимости к оптимуму.

Хромосомы большой длины обеспечивают вариативность популяции. Поэтому вероятность мутации тем выше, чем меньше параметров содержит хромосома. Параметры  $\alpha, \beta, \gamma$  последней формулы выбираются до начала эволюции и необходимы для тонкой настройки процесса мутации.

7. Локальная оптимизация. Данный этап наделяет алгоритм свойствами меметичности [10, 22] и обосновывает целесообразность использования прямого метода кодирования хромосом.

Преимуществами эволюционной настройки параметров ИНС являются следующие: независимость от архитектуры нейросети и характеристик функций активации; отсутствие необходимости в применении обучающей выборке. При использовании эволюционного алгоритма необходимо подобрать адекватную оценку качества работы ИНС, подходящую для любой особи в популяции.

Этап локальной оптимизации состоит из следующих шагов: эволюционная донастройка весов и параметров функций активации ИНС популяции; пересчет функций приспособленности; возвращение к предыдущим значениям параметров в случае снижения приспособленности.

Эти шаги проиллюстрированы блок-схемой на рисунке 4.

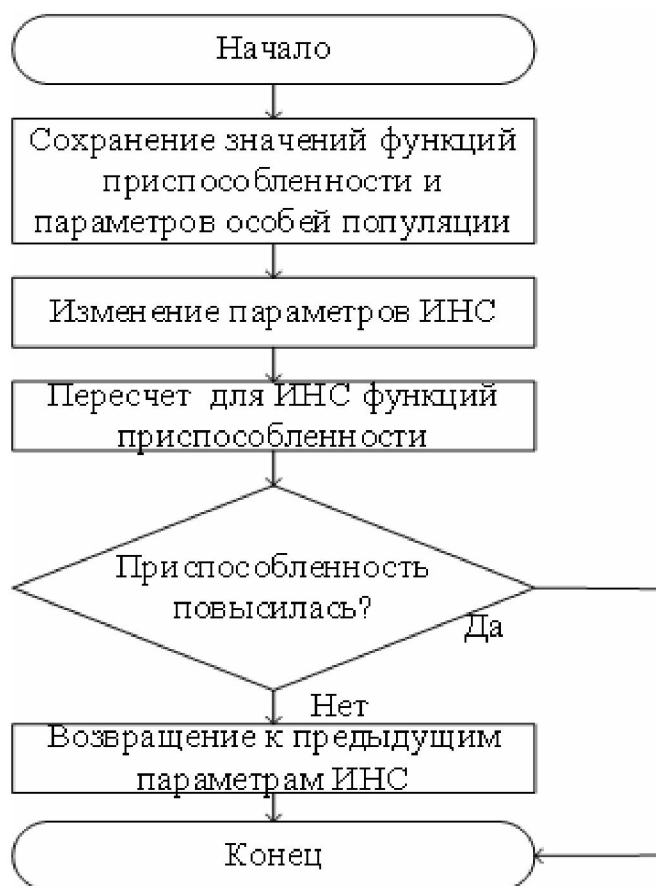


Рис. 4. Блок-схема этапа локальной оптимизации



8. Генерирование новой популяции  $P^{t+1}$  и переход на новую эпоху эволюции  $t+1$  (шаг 2 по схеме на рис. 3):

$$P^{t+1} = MT \cdot CR \cdot SL(P^t, F^t).$$

**Результаты тестирования предлагаемого метода.** Описанный метод, с одной стороны, способен рационально исследовать пространство поиска и избегать остановки на локальных экстремумах; с другой – эффективно использовать найденные «хорошие» решения и на их основе постепенно улучшать результаты.

В соответствии с теоремой схем [11, 21] применение оператора мутации лишь в редких случаях снижает приспособленность представлений меньшего порядка, а кроссинговер редко снижает приспособленность представлений с меньшей определяющей длиной. Эти факторы влияют на вероятность применения мутации и кроссинговера.

Описанные операторы селекции и рекомбинирования в совокупности с функцией приспособленности и пулом «хороших» особей предназначены для самоадаптации алгоритма к уровню сложности задачи.

Можно утверждать, что описанный метод обладает требуемыми для практического применения качествами. Эмпирически это подтверждается тестированием на разнотипных классических нейроэволюционных задачах, среди которых были задачи адаптивного управления; реализация логических функций; трудноформализуемые задачи восстановления поврежденных данных.

В качестве задачи адаптивного управления была выбрана задача об «обратном маятнике» теории автоматического управления. Обратный маятник имеет центр масс выше своей точки опоры и находится на конце жесткого стержня. Точка опоры кданного стержня закреплена на тележке.

В начальный момент времени маятник отклоняется на заданный угол от положения равновесия. Задача заключается в установлении устойчивого состояния маятника путем перемещения тележки некоторым усилием. При этом следует избегать концов отрезка, по которому может перемещаться тележка.

Было рассмотрено три варианта системы стабилизации (табл. 2).

1) классическая формулировка задачи. Один обратный маятник, тележка с которым перемещается по заданному отрезку. Для усложнения задачи на вход системы управления подавалась информация о текущем угле балансируемого маятника, но не о его угловой скорости;

2) двойной обратный маятник. Тележка с двумя обратными маятниками, закрепленными на стержнях различной длины;

3) обратный маятник на плоскости. Задача, в которой тележка перемещается не по прямой, а в двумерном пространстве.

Для проверки эффективности предложенной нейроэволюционной модели в качестве классификатора линейно-неразделимых образов [15] была использована функция XOR, являющаяся частным случаем задачи о классификации точек единичного гиперкуба. Хотя сама по себе реализация логических функций при помощи ИНС считается тривиальной задачей, она была выбрана для тестирования нейроэволюционной системы по двум соображениям.

(1) Проверка оптимизации параметров: весовых коэффициентов, вида функций активации и пороговых значений. Реализация функции XOR наглядно продемонстрировала точность настройки весовых коэффициентов и оптимальность выбора в качестве функции активации пороговой функции. (2) Проверка оптимальности структуры ИНС. Для успешного решения задачи XOR многослойный перцептрон должен иметь два входа, скрытый слой из четырех нейронов и один выход. Сгенерированная в ходе нейроэволюции ИНС обладает данной архитектурой. Это является доказательством эффективности процесса организации эволюции и правильности подобранных генетических операторов.

Задачи восстановления поврежденных данных (зашумленных сигналов, поврежденных изображений) за счет слабой формализуемости являются весьма показательными при испытании нейроэволюционных моделей. В качестве такой задачи было выбрано восстановление поврежденного изображения размера не более  $R \times C$ , в котором  $X\%$  пикселей неизвестны.

Процесс восстановления осуществляется следующим образом: на вход обученной ИНС подается поврежденное изображение,  $X\%$  пикселей которого повреждены. После этого на выходе ИНС формируется восстановленный образ. В нем  $X\%$  пикселей замещены значениями весов соответствующих нейронов.

Результаты решения приведенных тестовых задач при помощи разработанного метода и методов, указанных в первом разделе, представлены в таблице 2.

Таблица 2

**Сравнение эффективности нейроэволюционных методов  
для различных типов задач**

| Задача<br>Метод    | Количество эпох/неудач |                          |                               |      |                            |
|--------------------|------------------------|--------------------------|-------------------------------|------|----------------------------|
|                    | Обратный маятник       | Двойной обратный маятник | Обратный маятник на плоскости | XOR  | Восстановление изображения |
| Предложенный метод | 2/0                    | 25/1                     | 37/2                          | 11/0 | 146/2                      |
| CE [11]            | 5/0                    | 150/14                   | 230/20                        | 19/0 | 441/4                      |
| CNE [11]           | 4/0                    | 800/30                   | 1312/27                       | 31/2 | 659/0                      |
| DXNN [17]          | 3/0                    | 614/28                   | 802/12                        | 9/0  | 235/0                      |
| EANT [14]          | 7/0                    | 975/14                   | 1040/1                        | 14/0 | 148/23                     |
| ENS3 [8]           | 8/0                    | 741/16                   | 1021/5                        | 15/1 | 360/0                      |
| EPNet [20]         | 3/0                    | 543/20                   | 717/45                        | 10/0 | 152/0                      |
| ESP [20]           | 2/0                    | 19/1                     | 33/2                          | 17/1 | 554/0                      |
| GNARL [7]          | 4/0                    | 591/13                   | 719/34                        | 16/0 | 408/7                      |
| ICONE [12]         | 6/0                    | 903/6                    | 1452/14                       | 12/0 | 415/0                      |
| NEAT [8]           | 6/0                    | 24/1                     | 79/1                          | 8/0  | 210/0                      |
| SANE [18]          | 11/0                   | 63/3                     | 104/3                         | 9/0  | 276/4                      |

**Применение разработанного метода к некоторым практически важным задачам.** Эффективность и надежность предлагаемого метода доказаны применением к ряду практически важных задач: защита информационной системы (ИС) от DDoS-атак; рейтинговая оценка размещаемых в базе знаний материалов [3, 5]; диагностика систем 3D-печати [19].

Рассмотрим сначала использование предлагаемого метода для защиты ИС, состоящей из трех основных компонентов: базы данных; сервисов, инкапсулирующих логику обработки данных; корпоративного портала. Через корпоративный портал пользователи получают доступ к виртуальным средам для работы над проектами и поручениями, к методам аналитической обработки информации и данным, соответствующим агрегируемому в хранилище материалов (архив). С целью защиты данного портала был реализован классификатор для выявления DDoS-атак. В качестве обучающей выборки для классификатора были использованы файлы логов веб-запросов и находящиеся в открытом доступе базы сетевых атак (securelist.ru). Было сформировано 14 входных параметров для анализа.

1. HTTP-метод запроса, принимающий одно из значений, определенных протоколом – GET, HEAD, POST, PUT и т.д.

2. URI запрашиваемого ресурса в стартовой строке запроса.

3. Версия протокола. Например, HTTP/1.1.

4. Url источника запроса – определяется по значению заголовка referer.

5. User-Agent – информация о клиентском приложении, обрабатываемая специальным парсером для разбиения на отдельные признаки.

6. Корректность заголовков. Значение данного двоичного параметра формируется парсером заголовков.

7. Код состояния сервера – статус сервера в ответе на запрос клиента. Особый интерес представляют следующие значения статуса: 403 – forbidden, 404 – not found, 503 – service unavailable.

8. Ip-адрес клиента. Параметр необходим для фильтрации запросов во время DDoS-атаки, в том числе по первому октету Ip, определяющему RIR (регионального интернет регистратора).

9. Cookie: name=value – строка в ответе на присланный сервером куки (добавленный в HTTP-заголовок или установленный сценарием JavaScript).

10. Ответ клиента на сценарий JavaScript.

11. Ответ клиента на Flash-контент, отправленный сервером.

12. Время поступления запроса. Параметр был добавлен для снижения риска ошибок второго рода: вероятность запроса от легитимного клиента на корпоративный портал в нерабочее время ниже, чем атакующего хоста.

13. Время, прошедшее между двумя запросами клиента. Параметр позволяет блокировать хосты, отправляющие большое количество запросов за короткий промежуток времени.

14. Количество подключений клиента. Учет данного параметра при фильтрации клиентов повышает вероятность блокирования клиентов со статическим Ip, но оказывается полезным для фильтрации атакующих хостов, использующих прокси.

Из данных параметров был сформирован входной вектор ИНС, каждый элемент которого проходил предобработку. Для адаптации диапазона значений переменной и достижения однородности интерпретации входных параметров нейросети компоненты вектора были масштабированы. Исходные значения многих параметров варьируются в большом диапазоне, и работа ИНС с такими данными является некорректной. Поэтому такие значения проходили процедуру нормализации и приводились к одному диапазону. Данный этап предобработки данных позволяет сделать множество параметров пригодными к сравнению между собой.

В процессе нейроэволюции количество параметров было сокращено до 11 (параметры с номерами 2, 3, 5 были отброшены как обладающие низкой значимостью). В 17-ую эпоху эволюции в сетях прямого распространения были сформированы обратные связи и контекст. Таким образом, в ходе нейроэволюции была оптимизирована структура ИНС.

Рекуррентная ИНС показала наилучшие результаты со следующей структурой: входной слой, контекст, первый скрытый слой из 31 нейрона, второй скрытый слой из 14 нейронов, выходной слой (2 нейрона, на выход которых поступает на контекст). Описанная ИНС имеет два выхода: нет DDoS атаки на конкретном обращении к ИС; есть атака.

Архитектура с единственным нейроном в выходном слое могла бы быть потенциально возможной альтернативной данной классификационной нейросети. Значения непрерывной выходной переменной такой регрессионной ИНС могли бы интерпретироваться как «вероятность атаки низкая или отсутствует», «средняя вероятность атаки» и соответственно «высокая вероятность». Представляется, что данная альтернатива могла бы быть полезна в случае последующего анализа решений ИНС модератором.

Накопленная статистика свидетельствует, что при работе сервера в штатном режиме количество пользовательских запросов варьируется в районе сотен запросов в секунду. А во время DDoS-атаки средней силы оно возрастает на два–три порядка. Внедрение данного классификатора в ИС позволило значительно снизить риски отказа серверов из-за DDoS-атак: классификатор способен с вероятностью 99,892 % избежать ошибок первого и второго рода при распознавании (дифференциации) легитимных пользователей и атакующих машин.

Данная оценка формировалась в течение восемнадцати месяцев по результатам обработки данных о злонамеренных и тестовых DDoS-атаках. Тестовые атаки организовывались с целью оценивания работоспособности классификатора и закрепления навыков сотрудников информационного отдела по оперативной обработке критических ситуаций.

Второй областью для практического применения описанного метода является его использование в качестве сервиса для рейтинговой оценки материалов, размещаемых в виртуальной среде для работы над научными проектами, и принятия решения об их дальнейшей обработке. Потребность в данном сервисе обусловлена необходимостью автоматического управления контентом виртуальной среды. Каждому размещаемому в ней материалу сопоставлен набор разнотипных параметров, среди которых следующие: автор, категория, теги, результаты согласования, дата обновления, количество просмотров, комментарии пользователей, оценки модераторов.

Множество входных нейронов ИНС было поставлено в соответствие данным параметрам. После порядка 900 эпох эволюции точность нейросетевого вывода стала достаточной для верификации и последующего внедрения ИНС в слой бизнес-логики сервиса обработки контента. Наилучшее решение было представлено нейросетью прямого распространения, состоящей из пяти слоев. Наличие трех выходов позволило при помощи ИНС одновременно решать несколько задач: расчет непосредственно рейтинговой оценки; принятие решений о перенесении материала в архив, отдельную рубрику или закрытое хранилище.

Таким образом, нейровэволюционная система была интегрирована на корпоративный портал государственного бюджетного учреждения для автоматизации следующих процессов:

- 1) сбор и копирование в архив тех материалов, которые обладают высокими показателями качества (востребованности);
- 2) ранжирование и рубрикация материалов для их быстрого и результативного поиска.
- 3) отсев в закрытое хранилище материалов, не востребованных для дальнейшей работы над проектом.

Внедрение сервиса позволило распределить нагрузку на сервера (за счет структуризации материалов по отдельным библиотекам документов) и повысить качество работы сотрудников в виртуальной среде (за счет сокращения времени на поиск необходимой информации).

Сервис способствует также накоплению статистики с целью выявления наиболее востребованной информации, модификации состава и структуры информации на портале.

Третья область применения изложенного метода – это оптимизация конструкции пластиковых элементов 3D-принтеров в рамках открытого RepRap-сообщества. Проект RepRap (Replicating Rapid Prototyper, самовоспроизводящийся механизм для быстрого прототипирования) был организован для конструирования и распространения 3D-принтеров с открытыми спецификациями и общедоступными наработками.

Под самовоспроизведением понимается возможность изготовления на принтере всех пластиковых элементов его конструкции для сборки точной копии. В качестве материалов для печати чаще всего используются такие термопластики, как ABS (акрилонитрилбутадиенстирол), PLA (полилактид), TPE (термопластичный эластомер). Реже применяются duroпластики, силиконы и другие, менее доступные конечному пользователю материалы.

Процесс изготовления детали состоит из трех ключевых шагов: создание 3D-модели, слайсинг (преобразование модели в g-код для принтера) и непосредственно печать. Большинство существующих и распространяемых в RepRap-сообществе деталей на текущее время имеет ряд существенных недостатков: неоправданно высокую материалоемкость, неоптимальную конфигурацию и недостаточную прочность.

В связи с этим было принято решение о разработке новых версий деталей для оптимизации принтеров Prusa поколения i3. В качестве инструмента оптимизации выступает описанный нейроэволюционный метод: было сгенерировано поколение сетей, имеющих архитектурное сходство с сетями Хопфилда [16].

При наличии множества обратных связей в процессе функционирования динамика такой сети конвергирует к положению равновесия. Это позволяет использовать ИНС для решения задачи оптимизации: максимизации прочности детали при минимизации материалоемкости и отсутствии «поддержки» конструкции для предотвращения деформации во время печати стенок детали, имеющих угол наклона более  $70^\circ$ .

На рисунке 5 представлены исходная деталь для закрепления мотора оси «X» до оптимизации (слева) и после оптимизации (справа) для наглядной демонстрации возможностей метода.

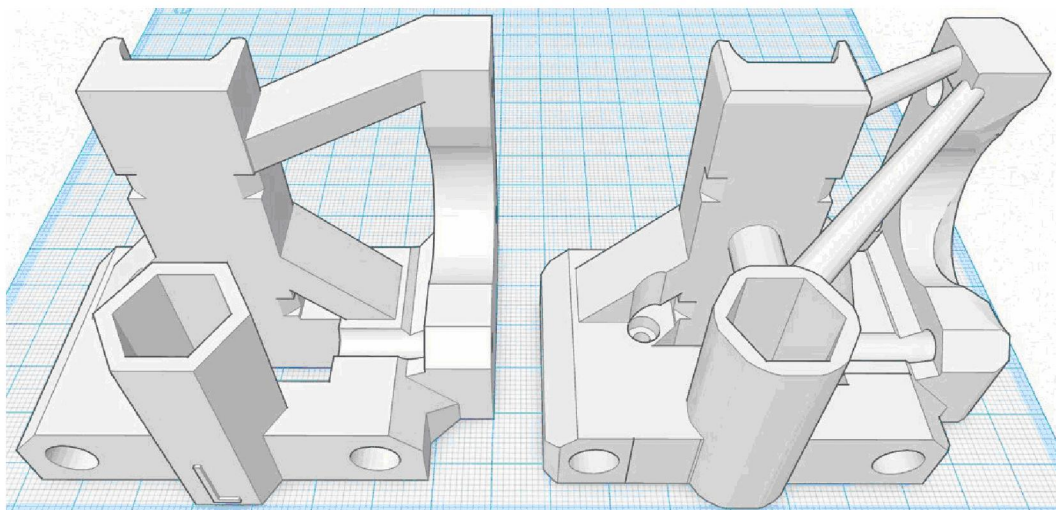


Рис. 5. Сравнение конструкции детали 3D-принтера Prusa до и после оптимизации

При помощи данной ИНС были спроектированы новые версии для 11-ти из 15-ти деталей принтера Prusa i3. Экономия филамента при их распечатывании достигла 27 % по сравнению с исходными образцами.

После распечатывания с теми же параметрами печати, что и у образцов (заполнение типа «honeycomb 50 %», пластик ABS), они были использованы как элементы конструкции оптимизированной версии 3D-принтера. За четыре месяца работы принтер показал лучшее качество печати изделий за счет уменьшения вибраций и большую износоустойчивость по сравнению с неоптимизированной версией.

Таким образом, нейроэволюционная система продемонстрировала точность в задачах многокритериальной оценки и адаптируемость к новым наборам данных в тестовых и практических областях применения. Она показала отличные результаты на этапах тестирования, валидации и эксплуатации.

**Выводы.** Подводя итоги, можно отметить, что предложенный метод подтвердил свою эффективность для построения широкого класса интеллектуальных систем принятия решений и управления динамическими объектами. При изменении задач управления или свойств управляемого объекта исходная топология нейросети может оказаться не оптимальной. Поэтому для корректного функционирования в нестационарной окружающей среде ИНС должна быть реконфигурирована. В соответствии с требованиями к устойчивости нейросетевого вывода метод позволяет оптимизировать не только параметры нейронов, но и перестраивать саму нейросетевую структуру.

Поскольку сложности при выборе топологии зачастую препятствуют применению ИНС на практике, то адаптивная самонастраивающаяся система способна упростить и в определенной степени автоматизировать процесс настройки и обучения ИНС, повысить ее универсальность.

Помимо практического применения, метод представляет интерес в области теоретических исследований – описанные в данной статье генетические операторы применительно к нейронным сетям предполагают возможность комбинирования нескольких ИНС в одну с сохранением их «жизнеспособности». Благодаря этому становится возможной генерация «нестандартных», нерегулярных топологий и гибридных ИНС.

#### Список литературы

1. Брумштейн Ю. М. Анализ моделей и методов выбора оптимальных совокупностей решений для задач планирования в условиях ресурсных ограничений и рисков / Ю. М. Брумштейн, Д. А. Тарков, И. А. Дюдилов // Прикаспийский журнал: управление и высокие технологии. – 2013. – № 3. – С. 69–180.
2. Зунг Н. Т. Распознавание объектов в системе технического зрения мобильного робота: использование библиотеки FLANN и алгоритма SURF / Н. Т. Зунг, И. А. Щербатов // Прикаспийский журнал: управление и высокие технологии. – 2014. – № 4. – С. 65–76.
3. Комлева Н. В. Инновационная информационная среда для обогащения качества образования / Н. В. Комлева, Л. А. Данченко, Т. М. Гулая // Инст. Дж. Форсайта и инновационной политики. – 2010. – № 6. – С. 257.
4. Лукьянов Н. Д. Параметрическая оптимизация автоматических систем стабилизации с помощью генетического алгоритма : автореф. дис. ... канд. техн. наук / Н. Д. Лукьянов. – Иркутск : Иркутский государственный университет путей сообщения, 2014. – 138 с.
5. Хлопкова О. А. Обработка контента в информационных средах на основе нейро-нечеткой модели принятия решений / О. А. Хлопкова, Н. В. Комлева // Экономика, статистика и информатика. Вестник УМО. – 2013. – № 5. – С. 188–192.
6. Bromley J. Signature verification using a “siamese” time delay neural network / J. Bromley, I. Guyon, Y. LeCun, E. Sackinger, R. Shah // International Journal of Pattern Recognition and Artificial Intelligence. – 1993. – № 7. – P. 739–744.
7. Cheetam A. Binary (presence-absence) similarity coefficients / A. Cheetam, J. Hazel // Paleontology. – 1969. – Vol. 43, № 5. – P. 1130–1136.
8. Chen X. A Multi-Facet Survey on Memetic Computation / X. Chen, Ye.-S. Ong, M.-H. Lim, K. C. Tan // IEEE Transactions on Evolutionary Computation. – 2011. – Vol. 15, № 5. – P. 591–607
9. Clune J. On the performance of indirect encoding across the continuum of regularity / J. Clune, K. O. Stanley, R. T. Pennock // IEEE Transactions on Evolutionary Computation. – 2011. – Vol. 15, № 3. – P. 346–367.
10. Del Moral P. On Adaptive Resampling Procedures for Sequential Monte Carlo Methods / P. Del Moral, A. Doucet, A. Jasra // Bernoulli Society for Mathematical Statistics and Probability. – 2012. – Vol. 18, no. 1. – P. 252–278.
11. Fulcher J. Advances in Applied Artificial Intelligence / J. Fulcher. – Hershey : Idea Group Publishing, 2006. – 325 p.

12. Gauci S. Generating Large-Scale Neural Networks Through Discovering Geometric Regularities / S. Gauci // Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2007). – New York : ACM, 2007. – P. 997–1004.
13. Gruau F. Neural network synthesis using cellular encoding and the genetic algorithm. – France : Ecole Normale Supérieure de Lyon, 1994. – 183 p.
14. Karkavitsas G. Automatic Music Genre Classification Using Hybrid Genetic Algorithms / G. Karkavitsas, G. Tsihrintzis. – Springer : Intelligent Interactive Multimedia Systems and Services, 2011. – P. 323–335.
15. Liles W. Introduction to Schema Theory: a survey lecture of pessimistic & exact schema theory / W. Liles, P. Wiegand, G. Mason. – University EC lab Activities, 2002. – 114 p.
16. Moriarty D. E. Forming neural networks through efficient and adaptive coevolution / D. E. Moriarty, R. Miikkulainen // Evolutionary Computation. – 1997. – № 5. – P. 373–399.
17. Pasemann F. Balancing rotators with evolved neurocontroller / F. Pasemann, U. Dieckmann. – Leipzig : Max-Planck-Institute for Mathematic in Sciences, 2007. – 36 p.
18. Rempis C. Evolving Complex Neuro-Controllers with Interactively Constrained Neuro-Evolution / Ch. Rempis. – Osnabrück University, 2012. – 240 p.
19. Salam A. Low-cost 3D Printing for Science, Education and Sustainable Development / A. Salam. – 2013. – 192 p.
20. Sher G. Handbook of Neuroevolution Through Erlang / G. Sher. – Springer Verlag, 2012. – 92 p.
21. Siebel N. Evolutionary reinforcement learning of artificial neural networks / N. Siebel, G. Sommer // International Journal of Hybrid Intelligent Systems. – 2007. – № 4 (3). – P. 171–183.

#### References

1. Brumshteyn Yu. M., Tarkov D. A., Dyudikov I. A. Analiz modeley i metodov vybora optimalnykh sovokupnostey resheniy dlya zadach planirovaniya v usloviyakh resursnykh ogranicheniy i riskov [The models and methods analysis of optimum choice for decisions sets in conditions of resources restrictions and risks]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Management and High Technologies], 2013, no. 3, pp. 169–180.
2. Zung N. T., Shherbatov I. A. Raspoznavanie obektov v sisteme tekhnicheskogo zreniya mobilnogo robota: ispolzovanie biblioteki FLANN i algoritma SURF [Object recognition in machine vision system of mobile robot: the using of library “FLANN” and algorithm “SURF”]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Management and High Technologies], 2014, no. 4, pp. 65–76.
3. Levanina N., Danchenok L., Gulaya T. Innovatsionnaya informatsionnaya sreda dlya obogashcheniya kachestva obrazovaniya [Innovative information environment for enriching education quality]. *Inst. Dzh. Forsayta i innovatsionnoy politiki* [Intst. J. Foresight and Innovation Policy], 2010, no. 6, p. 257.
4. Lukyanov N. *Parametricheskaya optimizatsiya avtomaticheskikh sistem stabilizatsii s pomoshchyu geneticheskogo algoritma* [Parametric optimization of automatic stabilization systems using genetic algorithm], Irkutsk, Irkutsk State University of Railway Transport Publ. House, 2014. 138 p.
5. Khlopkova O. A., Komleva N. V. *Obrabotka kontenta v informatsionnykh sredakh na osnove neyro-nechetkoy modeli prinyatiya resheniy* [Content management in information environments based on neuro-fuzzy of decision-making]. *Ekonomika, statistika i informatika. Vestnik UMO* [Economics, Statistics and Informatics. Bulletin of the UMO], 2013, no. 5, pp. 188–192.
6. Bromley J., Guyon I., LeCun Y., Sackinger E., Shah R. Signature verification using a “siamese” time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 1993, no. 7, pp. 739–744.
7. Cheetam A., Hazel J. Binary (presence-absence) similarity coefficients. *Paleontology*, 1969, vol. 43, no. 5, pp. 1130–1136.
8. Chen X., Ong Ye.-S., Lim M.-H., Tan K. C. A Multi-Facet Survey on Memetic Computation. *IEEE Transactions on Evolutionary Computation*, 2011, vol. 15, no. 5, pp. 591–607
9. Clune J., Stanley K. O., Pennock R. T. On the performance of indirect encoding across the continuum of regularity. *IEEE Transactions on Evolutionary Computation*, 2011, vol. 15, no. 3, pp. 346–367.
10. Del Moral P., Doucet A., Jasra A. On Adaptive Resampling Procedures for Sequential Monte Carlo Methods. *Bernoulli Society for Mathematical Statistics and Probabilit*, 2012, vol. 18, no. 1, pp. 252–278.
11. Fulcher J. *Advances in Applied Artificial Intelligence*, Hershey, Idea Group Publishing, 2006. 325 p.



12. Gauci S. Generating Large-Scale Neural Networks Through Discovering Geometric Regularities. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2007)*, New York, ACM Publ., 2007, pp. 997–1004.
13. Gruau F. *Neural network synthesis using cellular encoding and the genetic algorithm*, France, Ecole Normale Supérieure de Lyon Publ. House, 1994. 183 p.
14. Karkavitsas G., Tsihrintzis G. *Automatic Music Genre Classification Using Hybrid Genetic Algorithms*, Springer, Intelligent Interactive Multimedia Systems and Services Publ. House, 2011, pp. 323–335.
15. Liles W., Wiegand P., Mason G. *Introduction to Schema Theory: a survey lecture of pessimistic & exact schema theory*, University EC lab Activities Publ. House, 2002. 114 p.
16. Moriarty D. E., Miikkulainen R. Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation*, 1997, no. 5, pp. 373–399.
17. Pasemann F., Dieckmann U. *Balancing rotators with evolved neurocontroller*, Leipzig, Max-Planck-Institute for Mathematics in Sciences Publ. House, 2007. 36 p.
18. Rempis C. *Evolving Complex Neuro-Controllers with Interactively Constrained Neuro-Evolution*, Osnabrück University Publ. House, 2012. 240 p.
19. Salam A. *Low-cost 3D Printing for Science, Education and Sustainable Development*, 2013. 192 p.
20. Sher G. *Handbook of Neuroevolution Through Erlang*, Springer Verlag, 2012. 92 p.
21. Siebel N., Sommer G. Evolutionary reinforcement learning of artificial neural networks. *International Journal of Hybrid Intelligent Systems*, 2007, no. 4 (3), pp. 171–183.

УДК 004.652

**АНАЛИЗ ПРИНЦИПОВ ПОСТРОЕНИЯ  
АППАРАТНО-ПРОГРАММНОГО КОМПЛЕКСА  
ДЛЯ ДИАГНОСТИЧЕСКОГО МОДЕЛИРОВАНИЯ  
РАЗНОРОДНЫХ ФИЗИЧЕСКИХ ПРОЦЕССОВ  
В РАДИОЭЛЕКТРОННОЙ АППАРАТУРЕ<sup>1</sup>**

*Статья поступила в редакцию 20.08.2015, в окончательном варианте 14.09.2015.*

**Аminev Дмитрий Андреевич**, кандидат технических наук, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: aminev.d.a@yandex.ru

**Кокин Николай Николаевич**, аспирант, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: nikolay.kokin@mail.ru

**Увайсов Сайгид Увайсович**, доктор технических наук, профессор, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: uvaysov@yandex.ru

**Тихменев Александр Николаевич**, кандидат технических наук, Национальный исследовательский университет «Высшая школа экономики», 101000, Российская Федерация, г. Москва, ул. Мясницкая, 20, e-mail: alectikhmenev@gmail.com

Рассмотрены принципы диагностического моделирования физических процессов в электронных схемах; основные причины преждевременных отказов электронных средств (ЭС); их неудовлетворительной надежности. Выявлена необходимость создания информационно-логической модели для исследований вариантов организации потоков данных при диагностическом моделировании разнородных физических процессов в ЭС. Предложена архитектура информационной системы диагностического моделирования ЭС. Ее особенностью является комплексный подход к моделированию

---

<sup>1</sup> Работа выполнена при финансовой поддержке РФФИ (грант № 14-07-00414 «Информационная система диагностического моделирования физических процессов в электронных схемах»).