УДК 519.245

О.О. Евсютин, А.А. Шелупанов, В.Д. Бабишин, К.А. Соседко

Непрерывная оптимизация с помощью гибридной модели клеточных автоматов и обучаемых автоматов

Представлен алгоритм непрерывной оптимизации функций многих переменных, основанный на вычислительной модели клеточного автомата с целевой функцией. Эффективность данной вычислительной модели зависит от выбора правила развития клеточного автомата. При этом использование композиций правил показывает большую эффективность по сравнению с использованием отдельных правил. Отличительной особенностью данного исследования является использование динамической композиции, формируемой в процессе развития клеточного автомата с целевой функцией. Выбор правила на каждом шаге развития клеточного автомата осуществляется с помощью вычислительной модели обучаемого автомата. Результаты вычислительных экспериментов, проведенных со стандартными тестовыми функциями, показали, что данное решение позволяет повысить точность оптимизации.

Ключевые слова: непрерывная оптимизация, клеточные автоматы, обучаемые автоматы, клеточный автомат с целевой функцией.

doi: 10.21293/1818-0442-2019-22-1-50-54

Задачи оптимизации достаточно часто возникают в различных прикладных областях, когда требуется выбрать наилучший вариант из множества возможных. В настоящее время известно много методов решения подобных задач. Их можно разделить на два больших класса: детерминированные и стохастические. Детерминированные методы позволяют гарантировать оптимальность решения с заданной точностью, в то время как стохастические методы в общем случае не дают информации о точности найденного решения, однако на практике зачастую показывают высокую эффективность для сложных целевых функций. В связи с этим методы оптимизации, относящиеся ко второму классу, также называют эвристическими.

Значительная часть стохастических методов оптимизации основана на том, что поиск оптимального решения имитирует поведение сложных физических, биологических или социальных систем, состоящих из значительного числа взаимодействующих друг с другом однотипных элементов [1–4].

Удобной средой моделирования подобных систем являются клеточные автоматы. В работах [5–7] приведены алгоритмы оптимизации, основанные на соответствующем математическом аппарате.

Помимо [5–7], известно достаточно много исследований, в которых математический аппарат клеточных автоматов применяется для построения алгоритмов оптимизации. Отметим несколько работ в качестве примера.

В исследовании [8] клеточный автомат используется для улучшения генетического алгоритма. Для этого хромосомы генетического алгоритма записываются в ячейки двумерного клеточного автомата, и при скрещивании каждая хромосома берет генетический материал у соседей из окрестности Мура.

В работах [9, 10] клеточные автоматы объединяются с алгоритмами оптимизации, относящимися к роевому интеллекту. Статья [9] описывает модифицированный алгоритм муравьиной колонии, в

котором многомерное пространство поиска рассматривается как клеточная структура. В [10] клеточно-автоматный подход применен для улучшения алгоритма роящихся частиц. Для этого авторы работы вводят понятие окрестности частицы, под которой понимается множество частиц роя, наиболее близких к данной.

В исследовании [11] представлен клеточно-автоматный подход к улучшению мембранного алгоритма, схожий с подходом, использованным в [8] применительно к генетическому алгоритму.

Однако во всех данных работах клеточные автоматы применяются лишь в качестве вспомогательной вычислительной модели, служащей для повышения эффективности базовой метаэвристики.

В свою очередь, отличительной особенностью алгоритма, представленного в [5, 6], является использование динамики клеточного автомата для непосредственного поиска оптимума в пространстве решений. Для этого вводится и исследуется модификация классической модели клеточного автомата, названная клеточным автоматом с целевой функцией. Динамикой данной модели управляет композиция локальных правил перехода (правил развития клеточного автомата).

В [5] используется постоянная композиция правил развития, выбранная из нескольких вариантов как обеспечивающая наибольшую скорость сходимости процесса оптимизации. Результаты исследования по выбору композиции правил развития клеточного автомата с целевой функцией представлены в [6].

В [7] предлагается использовать адаптивный выбор правила развития клеточного автомата с целевой функцией. Для этого на основании текущего оптимального значения целевой функции и значений, достигнутых на двух предыдущих шагах, выбирается одно из двух правил развития клеточного автомата. Данный подход носит отчасти эмпирический характер, но в некоторых случаях позволяет избежать «застывания» в локальных оптимумах и

улучшить показатели сходимости. Однако, несмотря на это, получаемая таким образом композиция правил развития клеточного автомата не является оптимальной.

Целью настоящего исследования является создание улучшенного алгоритма оптимизации на основе клеточных автоматов с оптимальной композицией правил развития. Для получения оптимальной композиции правил развития клеточного автомата в процессе его развития предлагается использовать вычислительную модель обучаемых автоматов.

Вычислительная модель клеточных автоматов

Классическую математическую модель клеточного автомата можно описать совокупностью компонентов $\langle Z^n, A, Y, \sigma \rangle$, где Z^n – это пространство целочисленных координат клеток решетки; A – целочисленный алфавит внутренних состояний, определяющий конечное множество возможных значений отдельно взятой клетки; $Y = (y_1, y_2, \dots, y_k)$, $y_i \in Z^n$, $i = \overline{1, k}$ – шаблон соседства, определяющий вид окрестности, одинаковый для каждой клетки решетки; $\sigma: A^k \rightarrow A$ – локальная функция перехода (правило развития), задаваемая аналитически или в виде множества параллельных подстановок и одновременно применяемая ко всем клеткам решетки в ходе развития клеточного автомата.

Под окрестностью клетки понимается совокупность тех клеток решетки, текущие состояния которых повлияют на состояние данной клетки в следующий момент времени. Клетка, относительно которой строится окрестность, называется центральной клеткой. Каждый из элементов шаблона соседства представляет собой вектор относительных индексов, задающих расположение одной из клеток окрестности относительно центральной клетки, т.е. $y_i^j \in \{-r, \dots, 0, \dots, r\}$, $i = \overline{1, k}$, $j = \overline{1, n}$, где r – целое число [5, 12].

Клеточный автомат с целевой функцией

Модель клеточного автомата с целевой функцией расширяет классическую модель клеточного автомата и описывается совокупностью компонентов $\langle Z^n, L, A, Y, \sigma, U, \Phi \rangle$. Данное расширение классической модели является специфичным для задачи непрерывной оптимизации, его динамика направлена на достижение глобального оптимума заданной целевой функции. В представленной математической модели Z^n , L и Y соответствуют аналогичным компонентам классической модели; алфавит внутренних состояний A модифицирован и имеет вид $A = \mathcal{R}^m \times D$, где D – набор меток, записываемый как некоторый отрезок ряда неотрицательных целых чисел; $\sigma: (\mathcal{R}^m \times D)^k \rightarrow \mathcal{R}^m$ – локальная функция перехода, которая может быть задана лишь аналитически вследствие перехода от целочисленных значений к вещественным; $U: \mathcal{R}^m \rightarrow D$ – правило разметки клеток; $\Phi = \Phi(x_1, \dots, x_m)$ – целевая функция.

Каждая клетка решетки такого клеточного автомата содержит вектор вещественных значений $x \in \mathcal{R}^m$, рассматриваемый как элемент области определения целевой функции Φ , и целочисленную метку, определяющую качество содержащегося в векторе x решения в смысле близости к лучшему решению, полученному на данном шаге развития клеточного автомата.

Вычислительная модель обучаемых автоматов

Вычислительная модель обучаемых автоматов основана на парадигме автомата, работающего в неизвестной среде. В каждый момент дискретного времени обучаемый автомат совершает одно из множества заранее заданных действий. Выбор действия осуществляется в соответствии с изменяющимся во времени распределением вероятностей. Среда, в которой работает автомат, отвечает реакцией на каждое выбранное им действие. Автомат изменяет вероятности выбора действий в зависимости от того, какой была реакция среды на выбранное действие: положительной или отрицательной.

Обучаемый автомат описывается тройкой $\langle A, B, T \rangle$, где A – это множество действий автомата, B – множество входных значений, определяющих реакцию среды на действие автомата, и T – алгоритм обучения. Множеству действий A ставится в соответствие вектор вероятностей P , определяющий вероятности выбора отдельных действий. Алгоритм обучения обновляет вектор вероятностей в каждый момент времени в зависимости от реакции среды на действия автомата [13].

Известны различные приложения обучаемых автоматов, в том числе к задаче непрерывной оптимизации [14, 15].

В настоящем исследовании данная вычислительная модель использована для выбора правил развития клеточного автомата с целевой функцией в ходе процесса оптимизации.

Алгоритм непрерывной оптимизации на основе гибридной модели клеточных автоматов и обучаемых автоматов

В [5] введено два правила развития клеточного автомата с целевой функцией σ_I и σ_{II} , а также правило разметки клеток U . Правило развития σ_I определяет новое состояние каждой клетки автомата посредством усреднения состояний «хороших» клеток окрестности, а правило развития σ_{II} изменяет элементы вектора, содержащегося в клетке, на некоторую величину, рассчитываемую на основе значений соответствующих элементов соседних клеток.

Обучаемый автомат, предназначенный для выбора правила развития на каждом шаге клеточного автомата, построен следующим образом.

Множество действий A содержит два элемента по числу имеющихся правил развития клеточного автомата: $A = \{a_1, a_2\}$, где a_1 = “выполнить правило σ_I ”, a_2 = “выполнить правило σ_{II} ”.

Реакция среды на действие обучаемого автомата также может принимать два значения: $B = \{0, 1\}$, где 0

соответствует положительной реакции, а 1 – отрицательной.

Алгоритм обучения представляет собой правило обновления вектора вероятностей $\mathbf{P}$ после реакции среды на действие обучаемого автомата.

Если из множества A в момент времени t выбрано действие a_i , то вектор вероятностей $\mathbf{P}$ в момент времени $t+1$ обновляется следующим образом:

$$p_j(t+1) = \begin{cases} p_j(t) + a \cdot (1 - p_j(t)), & \text{if } i = j, \\ p_j(t) - a \cdot p_j(t), & \text{if } i \neq j, \end{cases} \quad (1)$$

если реакция среды на действие автомата положительна ($\beta=0$),

$$p_j(t+1) = \begin{cases} p_j(t) \cdot (1-b), & \text{if } i = j, \\ \frac{b}{k-1} + p_j(t)(1-b), & \text{if } i \neq j, \end{cases} \quad (2)$$

если реакция среды на действие автомата отрицательна ($\beta=1$).

Здесь a и b – это параметры алгоритма обучения, причем $0 < b < a < 1$.

Сформулируем улучшенный алгоритм оптимизации на основе клеточного автомата с целевой функцией и обучаемого автомата. Для определенности в качестве задачи оптимизации возьмем минимизацию целевой функции.

Вход: параметры клеточного автомата с целевой функцией; начальный вектор вероятностей выбора действий обучаемого автомата $\mathbf{P}=[p_1 \ p_2]$; параметры алгоритма обучения a, b, τ ; границы пространства поиска $[a, b]^m \subset \mathcal{H}^m$; число шагов развития клеточного автомата T .

Выход: вектор $\mathbf{x} \in [a, b]^m$, являющийся точкой искомого оптимума.

Шаг 1. Заполнить случайными значениями из области $[a, b]^m$ состояния всех клеток решетки клеточного автомата.

Шаг 2. Для каждой клетки решетки вычислить значение целевой функции, записывая получаемые значения в матрицу $\mathbf{M}=(m_{ij})_{i=1, j=1}^{h_1, l_2}$.

Шаг 3. Найти в матрице $\mathbf{M}$ элемент с минимальным значением. Записать данное значение в $\Phi_{\min}$, а координаты элемента – в (p, q) .

Шаг 4. Найти в матрице $\mathbf{M}$ элемент с максимальным значением. Записать данное значение в $\Phi_{\max}$.

Шаг 5. Вычислить метку каждой клетки решетки.

Шаг 6. Для t от 1 до T выполнить следующее:

Шаг 6.1. Выбрать действие обучаемого автомата в соответствии с вектором вероятностей $\mathbf{P}$.

Шаг 6.2. Выполнить выбранное действие обучаемого автомата – обновить состояние каждой клетки решетки клеточного автомата в соответствии

с выбранным правилом развития и шаблоном соседства $\mathbf{Y}$.

Шаг 6.3. Обновить матрицу $\mathbf{M}$ и значения $\Phi_{\min}$, $\Phi_{\max}$ и (p, q) .

Шаг 6.4. Обновить метку каждой клетки решетки.

Шаг 6.5. Если $t=1$, присвоить $\beta=0$ и перейти к шагу 6.8. В противном случае перейти к следующему шагу.

Шаг 6.6. Если $\Phi_{\min}^t = \Phi_{\min}^{t-1}$, то присвоить $\beta=1$ и перейти к шагу 6.8. В противном случае перейти к следующему шагу.

Шаг 6.7. Если $\frac{\Phi_{\min}^{t-2} - \Phi_{\min}^{t-1}}{\Phi_{\min}^{t-1} - \Phi_{\min}^t} < \tau$, присвоить

$\beta=0$. В противном случае присвоить $\beta=1$.

Шаг 6.8. Если $\beta=0$, обновить вектор вероятностей $\mathbf{P}$ по формуле (1), в противном случае – по формуле (2).

Шаг 7. Вернуть состояние клетки с координатами (p, q) в качестве вектора $\mathbf{x}$ и завершить алгоритм.

На первом шаге развития клеточного автомата с целевой функцией реакция среды на выбранное действие обучаемого автомата по умолчанию принимается положительной. На всех последующих шагах реакция среды определяется в зависимости от изменения скорости сходимости.

Вычислительные эксперименты

В таблице приведены результаты сравнения представленного алгоритма с исходным алгоритмом [5] на стандартных тестовых функциях многих переменных. В обоих случаях был использован клеточный автомат с решеткой 10×10 клеток и окрестностью Мура 3×3 клетки. Параметры обучаемого автомата в новом алгоритме были заданы следующим образом: начальное состояние вектора вероятностей $\mathbf{P}=[0,5 \ 0,5]$, параметры алгоритма обучения $a=0,9$, $b=0,1$, $\tau=0,5$. Параметры клеточного автомата с целевой функцией μ и ω варьировались для разных функций.

Каждый из алгоритмов был запущен 30 раз на каждой из тестовых функций. В таблице приведены усредненные значения найденных оптимумов, а также лучшие значения.

Из данной таблицы можно увидеть, что новый алгоритм с адаптивным выбором правила развития клеточного автомата опережает исходный алгоритм [5] в большинстве случаев либо показывает одинаковый результат, проигрывая только для одной целевой функции.

Кроме того, преимуществом нового алгоритма является большая стабильность работы, что выражается в меньшем разбросе выходных значений. Разница между средними значениями получаемых решений и лучшими решениями в серии запусков меньше, чем у алгоритма [5].

Результаты вычислительных экспериментов

Целевая функция	Значение μ	Значение ω	Исходный алгоритм [5]		Новый алгоритм	
			Среднее	Лучшее	Среднее	Лучшее
Φ_1	0,20	1,00	-12558,8132	-12569,4866	-12569,4866	-12569,4866
Φ_2	0,15	0,75	$9,0933 \cdot 10^{-2}$	0	$3,4249 \cdot 10^{-6}$	0
Φ_3	0,50	0,70	$5,2374 \cdot 10^{-4}$	0	$3,8203 \cdot 10^{-5}$	0
Φ_4	0,40	0,55	$9,7298 \cdot 10^{-4}$	0	$2,0775 \cdot 10^{-3}$	0
Φ_5	0,10	0,95	-99,4094	-99,5339	-99,5540	-99,6012
Φ_6	0,10	1,00	-78,3315	-78,3323	-78,3323	-78,3323
Φ_7	0,90	0,95	16,7522	2,3542	29,0445	1,8310
Φ_8	0,30	0,80	$3,2277 \cdot 10^{-7}$	0	$1,5271 \cdot 10^{-8}$	0
Φ_9	0,30	0,80	$6,4751 \cdot 10^{-5}$	0	$9,8860 \cdot 10^{-6}$	0
Φ_{10}	0,50	0,10	0,1041	0,0256	0,2553	0,0482

Заключение

В данной статье представлен новый алгоритм непрерывной оптимизации функций многих переменных, основанный на объединении вычислительных моделей клеточных автоматов и обучаемых автоматов и представляющий собой улучшение ранее полученного алгоритма, опубликованного в статье [5].

Отличительной особенностью нового алгоритма является использование обучаемого автомата для выбора правил развития клеточного автомата с целевой функцией в ходе процесса оптимизации. Результаты вычислительных экспериментов, проведенных со стандартными тестовыми функциями, показали, что данное решение позволило повысить эффективность процесса оптимизации в части точности получаемых решений.

Литература

- Kennedy J. Particle Swarm Optimization / J. Kennedy, R. Eberhart // Proceedings of the 1995 IEEE International Conference on Neural Networks. – Perth: IEEE, 1995. – P. 1942–1948.
- Dorigo M. The Ant System: Optimization by a colony of cooperating agents / M. Dorigo, V. Maniezzo, A. Colomni // IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). – 1996. – Vol. 26, No. 1. – P. 29–41.
- Storn R. Differential Evolution – a simple and efficient heuristic for global optimization over continuous spaces / R. Storn, K. Price // Journal of Global Optimization. – 1997. – Vol. 11, No. 44. – P. 341–359.
- Karaboga D. A survey: algorithms simulating bee swarm intelligence / D. Karaboga, B. Akay // Artificial Intelligence Review. – 2009. – Vol. 31, No. 1–3. – P. 61–85.
- The Algorithm of Continuous Optimization Based on the Modified Cellular Automaton / O. Evsutin, A. Shelupanov, R. Meshcheryakov, D. Bondarenko, A. Rashchupkina // Symmetry. – 2016. – Vol. 8, No. 9. – Article number 84.
- Евсютин О.О. Клеточный автомат с целевой функцией // Актуальные проблемы вычислительной и прикладной математики: тр. междунар. конф., посвященной 90-летию со дня рождения акад. Г.И. Марчука. – Новосибирск: Институт вычислительной математики и математической геофизики СО РАН, 2015. – С. 230–235.
- Бондаренко Д.О. Непрерывная оптимизация с помощью клеточного автомата с адаптивным выбором правила развития / Д.О. Бондаренко, О.О. Евсютин, А.В. Ращупкина // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2015. – № 4(38). – С. 119–122.
- Canyurt O.E. Cellular genetic algorithm technique for the multicriterion design optimization / O.E. Canyurt; P. Hajela // Structural and Multidisciplinary Optimization. – 2010. – Vol. 40, No. 1–6. – P. 201–214.
- Du T.S. A new cellular automata-based mixed cellular ant algorithm for solving continuous system optimization programs / T.S. Du, P.S. Fei, Y.J. Shen // Proceedings of the 4-th International Conference on Natural Computation. – Jinan: IEEE, 2008. – P. 407–411.
- Gholizadeh S. Layout optimization of truss structures by hybridizing cellular automata and particle swarm optimization // Computers and Structures. – 2013. – Vol. 125. – P. 86–99.
- Han M. An evolutionary membrane algorithm for global numerical optimization problems / M. Han, C. Liua, J. Xing // Information Sciences. – 2014. – Vol. 276. – P. 219–241.
- Кудрявцев В.Б. Основы теории однородных структур / В.Б. Кудрявцев, А.С. Подколзин, А.А. Болотов. – М.: Наука, 1990. – 296 с.
- Beigy H. A mathematical framework for cellular learning automata / H. Beigy, M.R. Meybodi // Advances in Complex Systems. – 2004. – Vol. 7, No. 3–4. – P. 295–319.
- Santharam G. Continuous Action Set Learning Automata for Stochastic Optimization / G. Santharam, P.S. Sastry, M.A.L. Thathachar // Journal of the Franklin Institute. – 1994. – Vol. 331, No. 5. – P. 607–628.
- Hashemi A.B. A Note on the Learning Automata Based Algorithms for Adaptive Parameter Selection in PSO / A.B. Hashemi, M.R. Meybodi // Applied Soft Computing Journal. – 2011. – Vol. 11, No. 1. – P. 689–705.

Евсютин Олег Олегович

Канд. техн. наук, доцент, ст. науч. сотр.
каф. безопасности информационных систем (БИС)
Томского государственного ун-та систем управления
и радиоэлектроники (ТУСУР)
Красноармейская ул., 146, г. Томск, Россия, 634045
ORCID 0000-0002-8257-2082
Тел.: +7-923-403-09-21
Эл. почта: eoo@keva.tusur.ru

Шелупанов Александр Александрович

Д-р техн. наук, профессор, ректор ТУСУР
Ленина пр., 40, г. Томск, Россия, 634050
Тел.: +7 (382-2) 51-05-30
Эл. почта: saa@tusur.ru

Бабишин Владимир Денисович

Д-р техн. наук, ст. науч. сотр., преподаватель аспирантуры
АО «Корпорация «ВНИИЭМ»
Вольная ул., 30, г. Москва, Россия, 105187
Тел.: +7-926-311-85-38
Эл. почта: Vova.babishin@yandex.ru

Соседко Ксения Андреевна

Методист аспирантуры АО «Корпорация «ВНИИЭМ»
Вольная ул., 30, г. Москва, Россия, 105187
Тел.: +7-915-021-27-84
Эл. почта: ksenya-1991@bk.ru

Evsutin O.O., Shelupanov A.A., Babishin V.D., Sosedko K.A.
Continuous optimization using a hybrid model of cellular automata and learning automata

The article presents an algorithm for continuous optimization of several variables functions, based on a computational model of cellular automata with an objective function. An effectiveness of this computational model depends on choice of the cellular automata development rule. At the same time, the use of rule compositions shows greater efficiency compared with the use of individual rules. A distinctive feature of this study is the use of a dynamic composition, formed during the development of cellular automata with an objective function. The rule choice at each step of the cellular automata development is carried out by a learning automata computational model. The results of computational experiments conducted with standard test functions show that this solution can improve the accuracy of optimization.

Keywords: continuous optimization, cellular automata, learning automata, cellular automaton with an objective function.

doi: 10.21293/1818-0442-2019-22-1-50-54

References

1. Kennedy J., Ebenhart R. Particle Swarm Optimization. Proc. of the 1995 IEEE International Conference on Neural Networks. Perth, IEEE, 1995. pp. 1942–1948.
2. Dorigo M., Maniezzo V., Colomi A. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 1996, vol. 26, no. 1, pp. 29–41.
3. Storn R., Price K. Differential Evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997, vol. 11, no. 4 pp. 341–359.
4. Karaboga D., Akay B. A survey: algorithms simulating bee swarm intelligence. *Artificial Intelligence Review*, 2009, vol. 31, no. 1–4, pp. 61–85.
5. Evsutin O., Shelupanov A., Meshcheryakov R., Bondarenko D., Raschupkina A. The Algorithm of Continuous Optimization Based on the Modified Cellular Automaton. *Symmetry*, 2016, vol. 8, no. 9, article number 84.
6. Evsutin O.O. Kletochnyj avtomat s celevoj funkciej [Cellular automaton with an objective function]. Aktual'nye problemy vychislitel'noj i prikladnoj ma-tematiki: Trudy Mezhdunarodnoj konferencii, posvjashhen-noj 90-letiju so dnja rozhdenija akademika G.I. Marchuka [Advanced Mathematics, Computations and Applications 2015. Proc. of the international conference]. Novosibirsk, Institute of Computational Mathematics and Mathematical Geophysics SB RAS, 2015, pp. 230–235 (in Russ.).

7. Bondarenko D.O., Evsutin O.O., Raschupkina A.V. Continuous optimization using cellular automata by choosing the evolution rule. *Proceedings of TUSUR University*, 2015, no. 4(38), pp. 119–122 (in Russ.).

8. Canyurt O.E., Hajela P. Cellular genetic algorithm technique for the multicriterion design optimization. *Structural and Multidisciplinary Optimization*, 2010, vol. 40, no. 1–6, pp. 201–214.

9. Du T.S., Fei P.S., Shen Y.J. A new cellular automata-based mixed cellular ant algorithm for solving continuous system optimization programs. Proc. of the 4-th International Conference on Natural Computation. Jinan, IEEE, 2008, pp. 407–411.

10. Gholizadeh S. Layout optimization of truss structures by hybridizing cellular automata and particle swarm optimization. *Computers and Structures*, 2013, vol. 125, pp. 86–99.

11. Han M., Liua C., Xing J. An evolutionary membrane algorithm for global numerical optimization problems. *Information Sciences*, 2014, vol. 276, pp. 219–241.

12. Kudryavcev V.B., Podkolzin A.S., Bolotov A.A. Osnovy teorii odnorodnyh struktur [The foundations of the theory of homogeneous structures]. Moscow, Nauka, 1990, 296 p. (in Russ.).

13. Beigy H., Meybodi M.R. A mathematical framework for cellular learning automata. *Advances in Complex Systems*, 2004, vol. 7, no. 3–4, pp. 295–319.

14. Santharam G., Sastry P.S., Thathachar M.A.L. Continuous Action Set Learning Automata for Stochastic Optimization. *Journal of the Franklin Institute*, 1994, vol. 331, no. 5, pp. 607–628.

15. Hashemi A.B., Meybodi M.R. A Note on the Learning Automata Based Algorithms for Adaptive Parameter Selection in PSO. *Applied Soft Computing Journal*, 2011 vol. 11, no. 1, pp. 689–705.

Oleg O. Evsutin

Candidate of Engineering, Senior Researcher,
Department of Information Systems Security, Tomsk State
University of Control Systems and Radioelectronics (TUSUR)
146, Krasnoarmeyskaya st., Tomsk, Russia, 634045
ORCID 0000-0002-8257-2082
Phone: +7-923-403-09-21
Email: eoo@keva.tusur.ru

Alexander A. Shelupanov

Doctor of Engineering, Professor, Rector TUSUR
40, Lenin pr., Tomsk, Russia, 634050
Phone: +7 (382-2) 51-05-30
Email: saa@tusur.ru

Vladimir D. Babishin

Doctor of Engineering, Senior Researcher, Professor
Graduate School JC «VNIIE Corporation»
30, Volnaia st., Moscow, Russia, 105187
Phone: +7-926-311-85-38
Email: Vova.babishin@yandex.ru

Ksenia A. Sosedko

Supervisor, Graduate School JC «VNIIE Corporation»
30, Volnaia st., Moscow, Russia, 105187
Phone: +7-915-021-27-84
Email: ksenya-1991@bk.ru