

# Адаптивный алгоритм глобальной оптимизации на основе взвешенного усреднения координат и нечетко-нейронных сетей

Кошур В.Д.

Красноярский государственный технический университет, [koshur@fvt.krasn.ru](mailto:koshur@fvt.krasn.ru)

**Аннотация.** Проблема поиска глобального экстремума для негладких, многоэкстремальных целевых функций на ограниченном множестве в пространстве  $R^n$  до сих пор не решена в полной мере. Предлагается новая модификация адаптивного нечетко-нейронного алгоритма глобальной оптимизации. Основопологающая идея состоит в применении технологии построения нечетко-нейронных сетей для оценок инверсных регрессий.

## Введение

Широкая область компьютерного моделирования прямо или косвенно связана с постановкой и решением задач оптимизации. При этом наибольший интерес представляет определение таких значений параметров моделируемой системы, которые соответствуют глобальному минимуму или максимуму заданной целевой функции.

В настоящее время наиболее распространенным и достаточно эффективным алгоритмом поиска глобального экстремума считается генетический алгоритм (ГА) [1-3] и его модификации (МГА) – гибридные ГА [4,5], адаптивные ГА [6-8]. В программный комплекс MatLab 7, вышедший в декабре 2004 года, включен комплект инструментальных средств на основе ГА и поисковой оптимизации. Дополнительным преимуществом ГА и МГА является возможность организации параллельных вычислений с реализацией задач большой размерности на многопроцессорных вычислительных комплексах и кластерных системах [3,9].

К достаточно универсальным алгоритмам глобальной оптимизации также относятся алгоритмы на основе непараметрических оценок инверсных регрессий [10-12], рандомизированного подхода [13,14] и близких по методологии методов непараметрической поисковой оптимизации [15,16] и взвешенного усреднения координат [17].

Проведённые исследования [18] вычислительной эффективности алгоритмов глобальной оптимизации на основе метода усреднения координат и МГА для набора тестовых примеров негладких многоэкстремальных целевых функций показали, что, при дополнительной эвристической подстройке параметров алгоритма метода усреднения координат, он является конкурентоспособным численным методом. В данной работе рассматривается модифицированный алгоритм поиска глобального экстремума на основе взвешенного усреднения координат, нечеткой логики [19-22] и нейронных сетей [5,23,24].

Идея построения и использования инверсных характеристик чрезвычайно плодотворна как при решении задач технической кибернетики [25,26], так и

решении широкого класса задач вычислительной математики [27-29], даже в тех случаях, когда инверсные связи могут быть неоднозначными. Использование технологий мягких вычислений [30,31] позволяют придать подобным вычислительным методам элементы искусственного интеллекта.

Главная идея излагаемого подхода состоит в том, что сложная задача оптимизации сводится к серии существенно более простых задач минимизации, состоящих в настройке параметров системы нечетко-нейронных сетей с помощью стандартных методов минимизации квадратичной ошибки выходов сети [5,22-24] на заданном множестве пробных точек.

Поясним идею на примере минимизации функции одной переменной  $I = f(t)$ . Если на рассматриваемом интервале  $[t_1, t_2]$  существует обратная функция  $t = f^{-1}(I)$ , то, вычисляя  $t^{[k]} = f^{-1}(I^{[k]})$ , при постепенном уменьшении значений целевой функции  $I^{[k]}$  и, проверяя условие  $t_1 \leq t^{[k]} \leq t_2$ , на некотором шаге  $k = N$  можно найти приближенное значение  $t^{[N]}$  для аргумента целевой функции  $t_{\min} = \arg \min f(t)$ ,  $t \in [t_1, t_2]$ .

Если обратной функции не существует, что, как правило, имеет место в приложениях, то, на  $k$ -том итерационном шаге по заданному множеству пробных точек  $t^{(j)[k]}$ ,  $j = 1, 2, \dots, M^{[k]}$  и соответствующих значений целевой функции  $I^{(j)[k]} = f(t^{(j)[k]})$ , всегда можно найти аппроксимирующую инверсную оценку, которая к обратной зависимости имеет лишь косвенное отношение, и реализовать её в виде нечетко-нейронной сети  $FNN^{[k]}$  (Fuzzy-Neural Network) вида  $t^{[k]} = FNN^{[k]}(I^{*[k]}, I^{(j)[k]}, t^{(j)[k]}, W^{[k]})$ , где  $I^{*[k]}$  – входной параметр сети,  $W^{[k]}$  – массив настраиваемых параметров,  $I^{(j)[k]}, t^{(j)[k]}$  – массивы задаваемых параметров сети. После обучения  $FNN^{[k]}$  на заданном множестве значений целевой функции и соответствующих аргументов  $\{I^{(j)[k]}, t^{(j)[k]}\}$ , вычисляется улучшенное значение аргумента при последовательном уменьшении значений целевой функции:  $I^{*[k]} = \min(I^{(1)}, I^{(2)}, \dots, I^{(M^{[k]})}) - \varepsilon^{[k]}$ , где  $\varepsilon^{[k]} \geq 0$ ,  $\varepsilon^{[k]} \xrightarrow{k \rightarrow \infty} 0$ . Для найденного аргумента  $t^{[k]}$  вычисляется значение целевой функции  $I^{pred[k]} = f(t^{[k]})$  и, если  $I^{pred[k]} \leq I^{*[k]}$ , то значение аргумента  $t^{[k]}$  добавляется к списку пробных точек и осуществляется переход на следующий итерационный шаг  $k + 1$ , иначе производится уменьшение параметра  $\varepsilon^{[k]}$ .

## 1. Теоретические основы метода усреднения координат

**1.1. Теорема сходимости [17].** Рассматривается ограниченная, непрерывная почти всюду функция  $f(x) : \Omega \rightarrow R$ , где  $x = (x_1, x_2, \dots, x_n) \in \Omega \subset R^n$ . Ограниченное множество  $\Omega$  представляется как не более чем счетное объединение замкнутых, непересекающихся подобластей, в каждой из которых функция

$f(x)$  непрерывна и ограничена. Необходимо найти точку  $x_{\min}$  реализации глобального минимума

$$f(x_{\min}) = \min f(x), \quad x \in \Omega . \quad (1)$$

Пусть  $\phi(x)$  – любая непрерывная, ограниченная функция. Вводится последовательность непрерывных функций  $P_s(y)$ ,  $s = 1, 2, \dots$ , таких что  $\forall y \in R$  значения  $P_s(y) \geq 0$  и для последовательности вида  $P_s(y)/P_s(z)$  выполняется условие монотонного неограниченного возрастания при увеличении номера  $s$  и любых фиксированных значениях  $y, z$  с условием  $y < z$ . Примерами функций  $P_s(y)$ , в частности, являются функции:  $\exp(-sy)$ ,  $s^{-y}$ ,  $y^{-s}$ , а также класс функций вида  $(1 - y^r)^s$ , для  $r = 1, 2, \dots$  при  $y \in [0; 1]$ .

При выполнении данных условий доказано предельное интегральное соотношение [17]:

$$\phi(x_{\min}) = \lim_{s \rightarrow \infty} \int_{\Omega} \phi(x) p_s(x) dx \quad (2)$$

где  $p_s(x)$  – последовательность нормированных ядер:

$$p_s(x) = P_s(f(x)) / \int_{\Omega} P_s(f(x)) dx . \quad (2a)$$

Если функцию  $\phi(x)$  определить в виде  $\phi_i(x) = x_i$  ( $i = 1, 2, \dots, n$ ), то набор предельных соотношений (2) будет давать координаты точки  $x_{\min} = (x_{1,\min}, x_{2,\min}, \dots, x_{n,\min})$  задачи (1). Этот результат может быть модифицирован и для задачи определения глобального максимума.

**1.2. Вероятностно-статистическая интерпретация.** Смысл предельного соотношения (2) состоит в том, что при фиксированном значении параметра  $s$  неотрицательная, нормированная в области  $\Omega$  весовая функция  $p_s(x)$  вида (2a) может рассматриваться как плотность распределения вероятности некоторой случайной величины. При этом значение плотности распределения возрастает при уменьшении значения функции  $f(x)$ . Тогда соотношение (2) представляет собой предел последовательности математических ожиданий от значений  $\phi(x)$ . При возрастании параметра селективности  $s$ , предел последовательности функций  $p_s(x)$  стремится к  $n$ -мерной дельта-функции с локализацией в точке  $x_{\min}$  [17]. Следует отметить, что если целевая функция  $f(x)$  имеет *не единственный* глобальный минимум в области  $\Omega$ , то, даже если пределы в соотношениях (2)

существуют при  $\phi_i(x) = x_i$  ( $i = 1, 2, \dots, n$ ), получаемые предельные значения координат могут не совпадать ни с одним из значений  $x_{\min}$  задачи (1).

## 2. Алгоритм поисковой оптимизации и его модификации

**2.1.** Вычислительная процедура нахождения приближенного значения координат точки  $x_{\min}$  строится на основе организации итерационного процесса [17]

$$x_i^{[k+1]} = \int_{\Omega^{[k]}} x_i p_s^{[k]}(x) dx, \quad (i = 1, 2, \dots, n), \quad (3)$$

$$p_s^{[k]}(x) = P_s^{[k]}(f(x)) / \int_{\Omega^{[k]}} P_s^{[k]}(f(x)) dx. \quad (3a)$$

Здесь  $k$  - номер шага вычислительного процесса,  $\Omega^{[k]}$  - область усреднения координат на шаге  $k$ . По мере роста параметра селективности  $s$  крутизна ядер  $p_s^{[k]}(x)$  увеличивается, это создает трудности при вычислении интегралов в (3), (3a). Частично эта проблема может быть решена за счет принудительного уменьшения области усреднения. В тоже время следует заметить, что этот прием несет в себе риск потерять искомую точку глобального минимума. Поэтому, представляется, что более эффективным и надежным способом повышения точности вычисления интегралов является последовательное увеличение количества пробных точек  $x^{(j)[k]}$ , ( $j = 1, 2, \dots, M^{[k]}$ ) на  $k$ -том шаге итерации, то есть  $M^{[k]} \geq M^{[k-1]}$ , при этом область усреднения следует оставлять неизменной. Рациональной модификацией численного алгоритма также является ранжирование (по значениям целевой функции) совокупности пробных точек, которые были получены на предыдущем и следующем итерационном шаге. Или, в простейшей реализации, осуществляется прямое добавление на текущем шаге набора новых пробных точек к имеющемуся набору сгенерированных случайных, равномерно распределенных точек в области  $\Omega$ .

Для приближенного вычисления интегралов в (3), (3a) используется суммирование значений подынтегральных выражений на заданном наборе пробных точек. Координаты пробных точек генерируются датчиком случайных чисел с равномерным распределением на интервалах  $[x_i^{[0]} - d_i, x_i^{[0]} + d_i]$ , ( $i = 1, 2, \dots, n$ ). Прямое произведение координатных интервалов образует  $n$ -мерную прямоугольную область  $\Pi$  такую, что  $\Omega \subseteq \Pi$ , с центром в точке  $x^{[0]} = (x_1^{[0]}, x_2^{[0]}, \dots, x_n^{[0]})$ . В набор пробных точек включаются только точки, которые попадают в область  $\Omega$ . После аппроксимации соотношения (3), (3a) принимают вид

$$x_i^{[k+1]} = \sum_{j=1}^{M^{[k]}} x_i^{(j)[k]} p_s^{[k]}(x^{(j)[k]}) V^{(j)[k]}, \quad (i = 1, 2, \dots, n), \quad (4)$$

$$p_s^{[k]}(x^{(j)[k]}) = \frac{P_s^{[k]}(g^{[k]}(x^{(j)[k]}))}{\sum_{j=1}^{M^{[k]}} P_s^{[k]}(g^{[k]}(x^{(j)[k]})) V^{(j)[k]}}, \quad (4a)$$

где  $V^{(j)[k]}$  соответствует малому  $n$ -мерному объему при разбиении области  $\Omega$  на подобласти, связанные с семейством точек интегрирования  $x^{(j)[k]}$ , ( $j = 1, 2, \dots, M^{[k]}$ ). Здесь  $g^{[k]}(x)$  – вспомогательные функции, масштабирующие целевую функцию  $f(x^{(j)[k]})$  к диапазону  $[0, 1]$ , см. ниже, формулы (6). Учитывая, что при генерации пробных точек используется равномерное случайное распределение, меру малых объемов  $V^{(j)[k]}$ , ( $j = 1, 2, \dots, M^{[k]}$ ) приближенно можно полагать постоянной величиной, вынести за знаки сумм в выражениях (4), (4a) и сократить. Таким образом, расчетные формулы принимают следующий вид

$$x_i^{[k+1]} = \sum_{j=1}^{M^{[k]}} x_i^{(j)[k]} p_s^{[k]}(x^{(j)[k]}), \quad (i = 1, 2, \dots, n), \quad (5)$$

$$p_s^{[k]}(x^{(j)[k]}) = \frac{P_s^{[k]}(g^{[k]}(x^{(j)[k]}))}{\sum_{j=1}^{M^{[k]}} P_s^{[k]}(g^{[k]}(x^{(j)[k]}))}. \quad (5a)$$

Следует отметить, что для улучшения аппроксимации  $n$ -кратных интегралов в виде конечных сумм при генерации последовательностей равномерно распределенных пробных точек  $x^{(j)[k]}$ , ( $j = 1, 2, \dots, M^{[k]}$ ) целесообразно использовать специализированные датчики псевдослучайных чисел [32].

Для приведения значений целевой функции  $f(x^{(j)[k]})$  к диапазону  $[0, 1]$  проводится масштабирование и вводятся вспомогательные функции  $g^{[k]}(x)$  вида:

$$g^{[k]}(x^{(j)[k]}) = \frac{f(x^{(j)[k]}) - f_{\min}^{[k]}}{f_{\max}^{[k]} - f_{\min}^{[k]}}, \quad (6)$$

где обозначено

$$f_{\min}^{[k]} = \min(f(x^{(1)[k]}), f(x^{(2)[k]}), \dots, f(x^{(M^{[k]})[k]})), \quad (6a)$$

$$f_{\max}^{[k]} = \max(f(x^{(1)[k]}), f(x^{(2)[k]}), \dots, f(x^{(M^{[k]})[k]})) . \quad (6б)$$

Таким образом, соотношения (5), (5а), (6), (6а), (6б) и конкретизация вида ядер  $P_s^{[k]}(y)$  полностью определяют вычислительный алгоритм глобальной оптимизации на основе взвешенного усреднения координат.

### 3. Адаптивный нечетко-нейронный алгоритм поиска глобального минимума

Методологической основой и, в определенной степени, прототипом для предлагаемой нечетко-нейронной адаптивной системы итерационного поиска координат точки  $x_{\min}$  глобального минимума задачи (1) является построение непараметрических оценок инверсных регрессий [10,11] и рандомизированный подход [13,14], которые предшествовали детальным исследованиям [15-17]. Взвешенное усреднение пробных координат  $x^{(j)[k]}$ , ( $j = 1, 2, \dots, M^{[k]}$ ), полученных на  $k$ -том итерационном шаге, позволяет найти координаты  $(x_1^{[k+1]}, x_2^{[k+1]}, \dots, x_n^{[k+1]})$  точки  $x^{[k+1]}$ , для которой значение целевой функции меньше, чем ранее найденные:  $f(x^{[k+1]}) \approx f_{\min}^{[k]} - \varepsilon^{[k]}$ , при этом оценки инверсных регрессий для каждой координаты могут быть построены в виде:

$$x_i^{[k+1]} = \sum_{j=1}^{M^{[k]}} x_i^{(j)[k]} E_i^{(j)[k]} \left( \frac{f^{*[k]} - f(x^{(j)[k]})}{\beta_i^{(j)[k]}} \right) + \Delta x_i^{[k]}, \quad (i = 1, 2, \dots, n), \quad (7)$$

$$f^{*[k]} = f_{\min}^{[k]} - \varepsilon^{[k]}, \quad \varepsilon^{[k]} \geq 0, \quad \varepsilon^{[k]} \xrightarrow{k \rightarrow \infty} 0 . \quad (7a)$$

Здесь  $E_i^{(j)[k]}(y)$  – нормированные, симметричные относительно нуля, колоколообразные ядра,  $\beta_i^{(j)[k]}$  – коэффициенты размытости ядер,  $\Delta x_i^{[k]}$  – малая случайная добавка такая, что  $\Delta x_i^{[k]} \xrightarrow{k \rightarrow \infty} 0$ . Следует отметить, что приведенный вид соотношений (7) отличается от представленных в работах [10,11]. Отличие заключается в том, что в соотношении (7) допускается возможность независимой индивидуализации вида ядер и параметров размытости не только при переходе к следующей итерации, но и для каждой пробной точки и их координат. По структуре соотношения вида (5) и (7) практически совпадают и соответствуют взвешенному усреднению координат, хотя их методологическая основа различна.

Инверсными соотношениями вида (7) необходимо смоделировать закономерности, которые, как правило, носят неточный и, в ряде случаев, многозначный характер, поэтому целесообразно представить эти инверсные зависимости как нечеткие. Это позволяет придать инверсной регрессии более обобщенный смысл и существенно расширить возможности ее практического применения. Трудности при практической реализации «точных» аналитических соотношений вида (7) связаны с выбором вида ядер  $E_i^{(j)[k]}(y)$ , коэффициентов размытости  $\beta_i^{(j)[k]}$  и параметров пробных движений  $\Delta x_i^{[k]}$ . При нейросетевой реализации этот выбор осуществляется стандартным образом одним из алгоритмов адаптивного обучения нечетко-нейронных сетей [5,22-24].

В качестве прототипа для построения нечетко-нейронной сети применяется метод Takagi-Sugeno [5,33]. Реализованная четырехслойная сеть состоит из слоя фаззификации, слоя применения операторов Т-нормы, слоя нормализации и слоя дефаззификации [5]. Для входных переменных  $z_1, z_2, \dots, z_m$  и действительной величины  $y$ , соответствующей выходному значению, определяется набор нечетких правил вывода в форме:

$$Rule_q : \text{if } z_1 \text{ is } A_{q1} \text{ and } z_2 \text{ is } A_{q2} \dots \text{and } z_m \text{ is } A_m \text{ then } y \text{ is } y_q, \quad (8)$$

где  $q = 1, 2, \dots, Q$  (на  $k$ -том итерационном шаге  $Q = M^{[k]}$ ),  $A_{ql}$  определяют нечеткие множества для  $q$ -го правила и  $l$ -ой лингвистической переменной, соответствующей  $z_l$ ;  $y_q$  представляет действительное число – значение требуемого вывода для  $Rule_q$ . Система нечетко-нейронных сетей, каждая из которых по виду аналогична [34], реализует итерационное улучшение значений координат  $x_i^{[k]} = y_i^{[k]}$ , где

$$y_i^{[k]} = FNN_i^{[k]}(I^{*[k]}, f(x^{(j)[k]}), x^{(j)[k]}, W_i^{[k]}), \quad (i = 1, 2, \dots, n)., \quad (9)$$

Здесь  $W_i^{[k]}$  – вектор настраиваемых параметров сети  $FNN_i^{[k]}$ , входной переменной для семейства нечетко-нейронных сетей (9) является только значение целевой функции  $I^{*[k]}$ , которое определено ниже. Значения целевой функции  $f(x^{(j)[k]})$  и координаты пробных точек  $x^{(q)[k]}$  сетью не обрабатываются, а выступают в роли подмножества заданных параметров сети  $FNN_i^{[k]}$ .

В качестве функций принадлежности используются функции Гаусса:

$$\mu_{(i)(q)(l)}^{[k]}(I) = \exp\left(-\frac{(I - (f(x^{(q)[k]}) + a_{(i)(q)(l)}^{[k]}))^2}{2(b_{(i)(q)(l)}^{[k]})^2}\right), \quad (I = 1, 2, \dots, m), \quad (10)$$

с настраиваемыми параметрами  $W_i^{[k]} = \{a_{(i)(q)(l)}^{[k]}\} \cup \{b_{(i)(q)(l)}^{[k]}\}$ .

Для реализации логического оператора Т-нормы используется операция произведения:

$$u_q = \mu_{q1}\mu_{q2}\dots\mu_{qm} \quad (q = 1, 2, \dots, Q) \quad , \quad (11)$$

где  $\mu_{qi}$  – значение функции принадлежности для входной переменной  $z_i$  относительно нечеткого множества  $A_{qi}$ . Формирование логического вывода по определенной системе нечетких правил (8) и дефаззификация, в форме «центра притяжения» [34], реализуется в виде

$$y_i^{[k]} = \frac{\sum_{q=1}^{M^{[k]}} x_i^{(q)[k]} u_q^{[k]}}{\sum_{q=1}^{M^{[k]}} u_q^{[k]}} \quad . \quad (12)$$

Следует отметить, что при  $m = 1$  соотношение (12) соответствует инверсной регрессии (7) и аналогично по форме взвешенному усреднению координат пробных точек (5).

После обучения нечетко-нейронных сетей  $FNN_i^{[k]}$ , выходы  $y_i^{[k]}$  соответствуют улучшенным значениям координат  $x_i^{[k]}$ , если на вход сетей подается значение  $I^{*[k]} = f^{*[k]}$ , которое определяется в виде (7а). Обучение проводится на множестве значений  $I^{(j)[k]} = f(x^{(j)[k]})$ , при которых семейство сетей  $FNN_i^{[k]}$  вида (9) должны выдавать приближенные значения соответствующих координат пробных точек  $x_i^{(j)[k]}$ .

При численной реализации адаптивного нечетко-нейронного алгоритма глобальной оптимизации целесообразно проводить наращивание координатных сетей  $FNN_i^{[k]}$  по числу нечетких правил на новом итерационном шаге, если число пробных точек  $M^{[k]}$  на новом итерационном шаге включает как уже имеющийся список, так и новый набор пробных точек.

#### 4. Результаты вычислительных экспериментов

С целью апробации разработанного адаптивного нечетко-нейронного алгоритма глобальной оптимизации была проведена серия вычислительных экспериментов. Рассматривались гладкие и негладкие целевые функции различного вида, заданные на ограниченных односвязных областях  $\Omega$  в пространстве  $R^n$ , где  $n = 2, 10, 100, 1000$ . Учитывая сложность визуализации графических образов в многомерных пространствах, ниже приводятся наглядные графические представления численных результатов решения задачи минимизации (1) для много-

экстремальных целевых функций *двух переменных*, заданных в прямоугольной области  $\Pi \subset R^2$ .

**4.1. Гладкая многоэкстремальная функция.** Целевая функция задана аналитическим выражением:

$$f(x, y) = -\frac{1}{\frac{x_1^2 + x_2^2}{200} - \cos(x_1) \cos(x_2 / \text{sqrt}(2)) + 2} \quad (13)$$

Глобальный минимум, равный  $-1$ , соответствует координатам  $x_1 = x_2 = 0$ .

На рис. 1 представлен процесс поиска глобального минимума на поверхности целевой функции (13). Шаги итерационного процесса на плоскости  $x_1, x_2$  с изображением изолиний целевой функции приведены на рис. 1а.

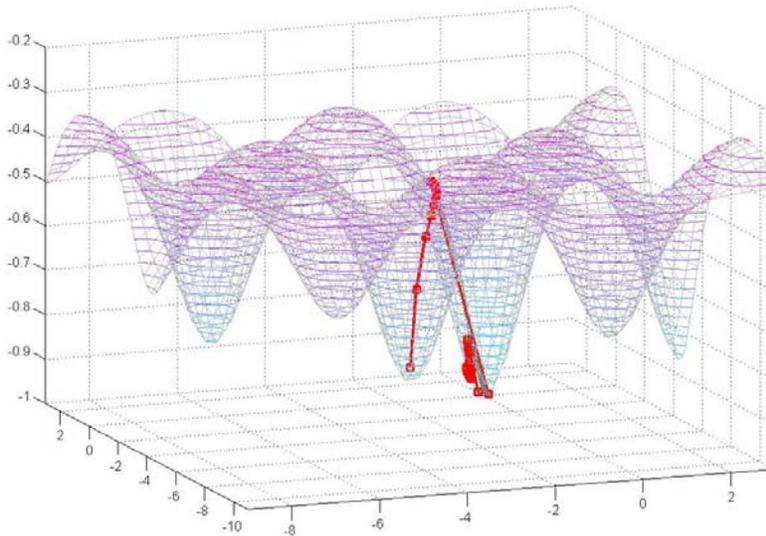


Рис. 1. Поверхность целевой функции (13) и изображение итерационных шагов поиска глобального минимума.

Гистограммы распределения значений целевой функции в пробных точках  $x^{(q)[k]}$ , ( $q = 1, 2, \dots, 100$ ), использованных на итерационных шагах  $k = 1, 2, \dots, 14$ , показаны на рис. 1б. Гистограммы иллюстрируют постепенное выделение глобального экстремума в распределении значений целевой функции.

Для гистограмм горизонтальная ось соответствует приведенным к интервалу  $[0, 1]$  значениям целевой функции, а вертикальная ось – значениям относи-

тельной частоты появления некоторого фиксированного значения целевой функции. Вычислительны эксперименты были проведены:

- для различного числа итераций ( $k = 5, 10, 50, 100, 1000$ ),
- при различных стартовых условиях – начальных параметрах  $W_i^{[k]} (i = 1, 2)$  нечетко-нейронной сети (9),
- для варианта наращивания числа нечетких правил (8) сетей  $FNN_i^{[k]}$ ,
- при фиксации числа нечетких правил (8) и количества используемых пробных точек на каждом итерационном шаге, т.е.  $Q = M^{[k]} = 100, \forall k$ .

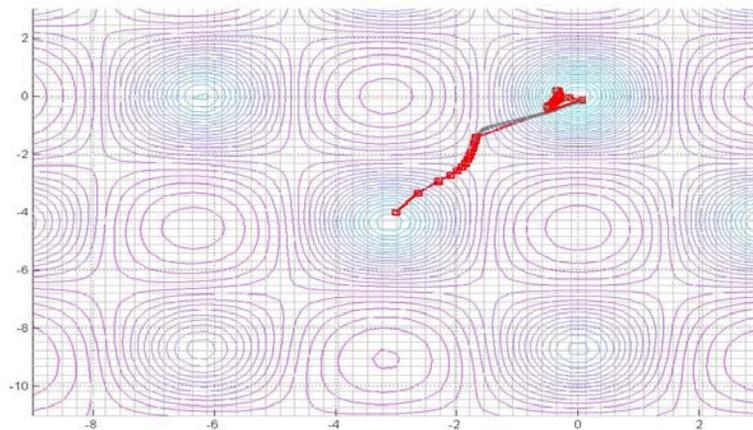


Рис. 1а. Изолинии поверхности целевой функции (13) и последовательные шаги поиска глобального минимума.

Расчеты показали стабильность достижения приближенных значений  $x_{1\min}, x_{2\min}$  при работе адаптивно подстраиваемой нечетко-нейронной системы (9). При проведении данных расчетов, как правило, имела место *немонотонная* сходимость значений координат  $x_1^{[k]}, x_2^{[k]}$  к координатам  $x_{1\min}, x_{2\min}$  точки глобального минимума.

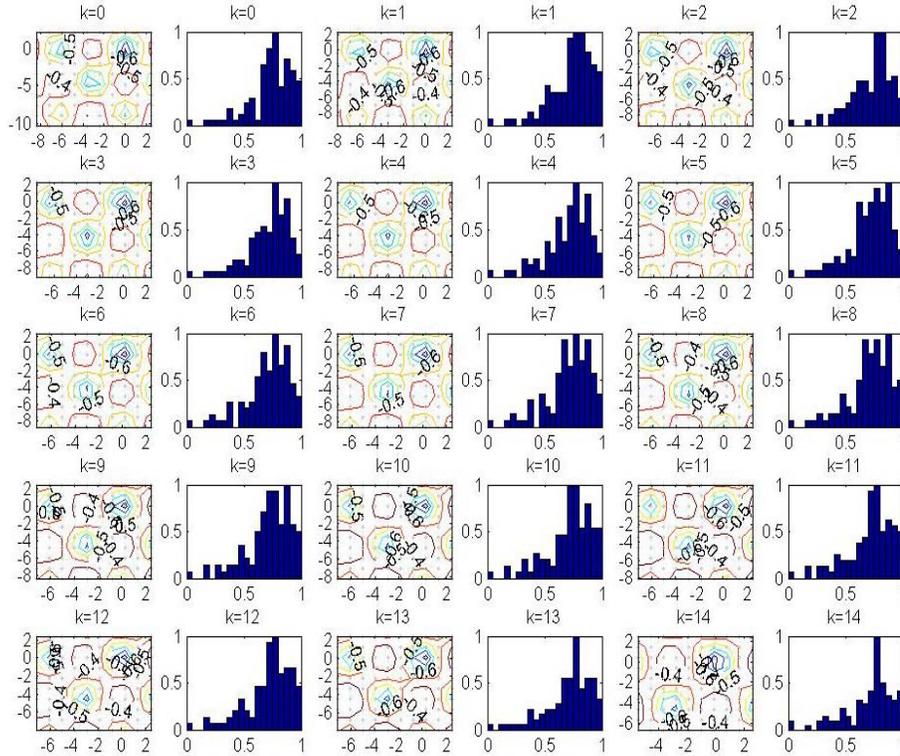


Рис. 16. Гистограммы распределения значений целевой функции (13) в пробных точках для первых 14 шагов итерационного процесса и изолинии, построенные по информации о значениях  $f(x^{(q)[k]})$ .

**4.2. Многоэкстремальная функция Растргина.** Целевая функция Растргина задается аналитическим выражением следующего вида [17]:

$$f(x_1, x_2) = 20 + x_1^2 - 10 \cos(2\pi x_1) + x_2^2 - 10 \cos(2\pi x_2). \quad (14)$$

Результаты расчетов по нахождению глобального минимума для функции Растргина показаны на рис. 2, рис. 2а.

Общий характер вычислительного процесса аналогичен представленным результатам пункта 4.1.

Адаптивный алгоритм глобальной оптимизации

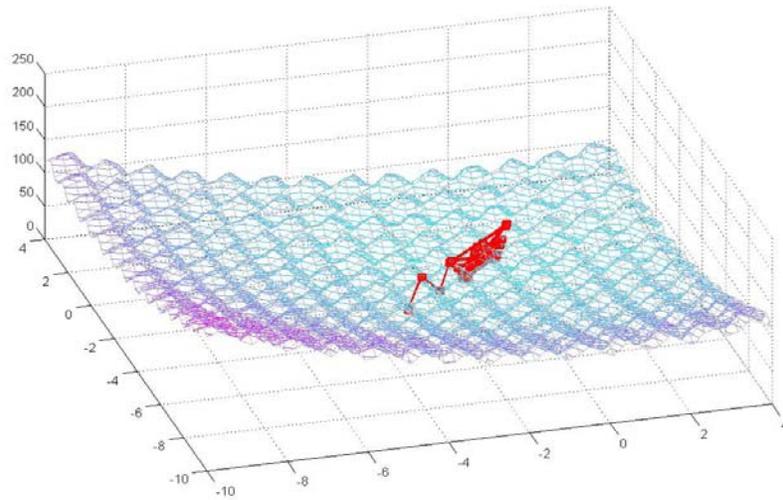


Рис. 2. Поверхность целевой функции (14) и изображение итерационных шагов поиска глобального минимума.

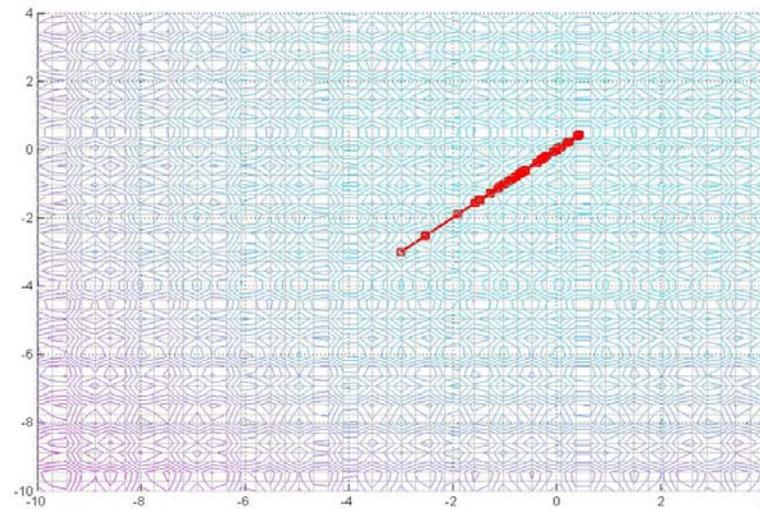


Рис. 2а. Изолинии поверхности целевой функции (14) и последовательные шаги поиска глобального минимума.

**4.3. Негладкая 4-х экстремальная функция.** Целевая функция задается выражениями вида [17]:

$$\begin{aligned}
 f1 &= -5\exp(-3((\text{abs}(x_1 + 1))^{0.6} + (\text{abs}(x_2 + 1))^{0.6})); \\
 f2 &= -10\exp(-2(\text{abs}(x_1) + \text{abs}(x_2))); \\
 f3 &= -7\exp(-2.5((\text{abs}(x_1 - 1))^{0.8} + (\text{abs}(x_2 - 1))^{0.8})); \\
 f4 &= -3\exp(-((\text{abs}(x_1 - 2))^{0.9} + (\text{abs}(x_2 - 2))^{0.9})); \\
 f(x_1, x_2) &= f1 + f2 + f3 + f4.
 \end{aligned}
 \tag{15}$$

Результаты расчетов по нахождению глобального минимума для 4-х экстремальной функции представлены на рис. 3, рис. 3а.

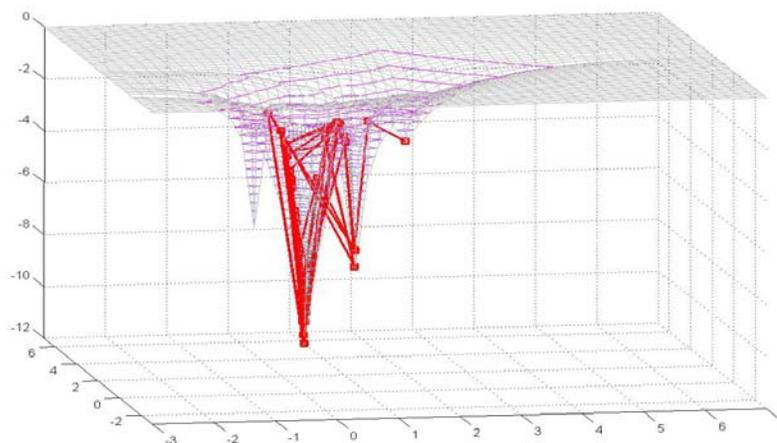


Рис. 3. Поверхность целевой функции (15) и изображение итерационных шагов поиска глобального минимума.

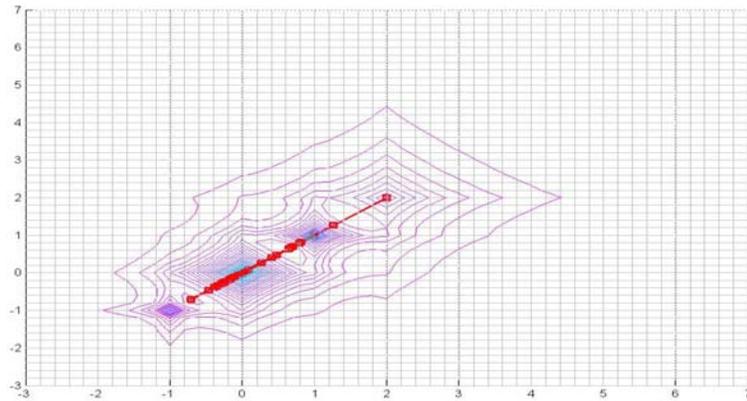


Рис. 3а. Изолинии поверхности целевой функции (15) и последовательные шаги поиска глобального минимума.

**4.4. Негладкая 4-х экстремальная функция, имеющая три равных по значению глобальных минимума.** Данная функция определяется модифицированными соотношениями (15) в виде:

$$\begin{aligned}
 g1 &= -7\exp(-3((\text{abs}(x_1 + 1))^{0.6} + (\text{abs}(x_2 + 1))^{0.6})); \\
 g2 &= -7\exp(-2(\text{abs}(x_1) + \text{abs}(x_2))); \\
 g3 &= -7\exp(-2.5((\text{abs}(x_1 - 1))^{0.8} + (\text{abs}(x_2 - 1))^{0.8})); \\
 f4 &= -3\exp(-((\text{abs}(x_1 - 2))^{0.9} + (\text{abs}(x_2 - 2))^{0.9})); \\
 f(x_1, x_2) &= g1 + g2 + g3 + f4;
 \end{aligned} \tag{16}$$

Результаты расчетов по нахождению глобального минимума для 4-х экстремальной функции с тремя одинаковыми по значению функции точками (1,1), (0,0), (-1,-1) представлены на рис. 4, рис. 4а.

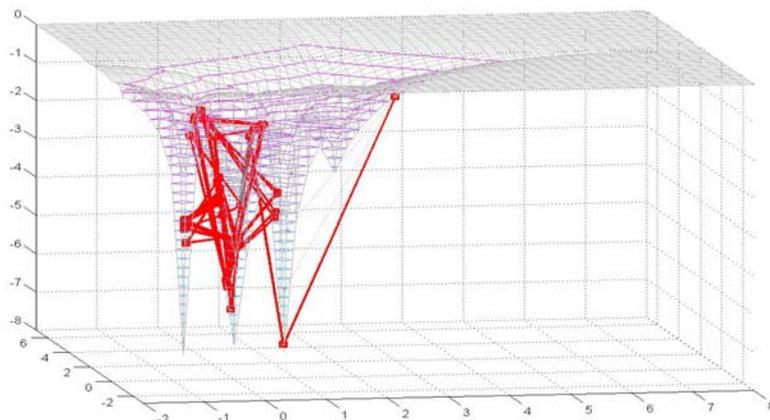


Рис. 4. Поверхность целевой функции (16) и изображение расходящегося процесса поиска неединственного глобального минимума.

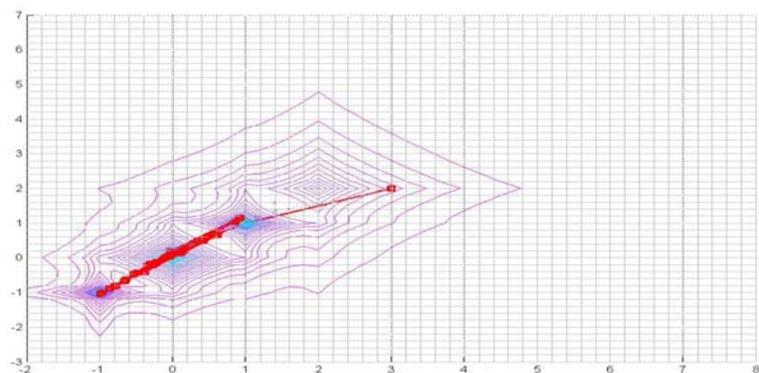


Рис. 4а. Изолинии поверхности целевой функции (16) и последовательные шаги поиска глобального минимума.

Представленные результаты наглядно показывают, что для целевой функции с неединственным глобальным минимумом адаптивный нечетко-нейронных алгоритм (также как и метод взвешенного усреднения координат) не позволяет найти хотя бы один из глобальных минимумов в виду того, что итерационный процесс расходится.

## Заключение

Проведенные вычислительные эксперименты свидетельствуют о высокой эффективности предложенного адаптивного нечетко-нейронного алгоритма нахождения глобального экстремума гладких и негладких целевых функций.

Дополнительным и достаточно весомым аргументом в пользу нечеткого представления инверсных характеристик является возможность проведения поиска глобального экстремума для процедурно вычисляемых целевых функций при наличии помех (или погрешностей) и не только аддитивного типа. Исследование этих вопросов, а также анализ применения данного подхода для решения задач многокритериальной оптимизации и решения негладких нелинейных систем уравнений в пространстве  $R^n$  предполагается провести в последующих работах.

## Литература

1. Holland J.H. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, 1975.
2. Goldberg D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
3. Курейчик В.М. Генетические алгоритмы. Состояние, Проблемы. Перспективы // Известия Академии наук. Теория и системы управления, 1999, № 1. С. 144-160.
4. Кошур В.Д., Ильин В.А. Нейронная сеть Хопфилда как кроссовер генетического алгоритма // Научная сессия МИФИ-2003. V Всероссийская научно-техническая конференция “Нейроинформатика-2003”. Сборник научных трудов. Часть 1. М.: МИФИ, 2003. С. 92-100.
5. Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М: Горячая линия – Телеком, 2004. 454 с.
6. Herrera F., Lozano M., Verdegey J.L. Dynamic and heuristic fuzzy connectives-based crossover operators for controlling the diversity and convergence of real coded genetic algorithms // Int. Journal of Intelligent Systems, 1996, N 11, pp. 1013-1041.
7. Herrera F., Lozano M. Adaptation of genetic algorithm parameters based on fuzzy logic controllers // Genetic Algorithms and Soft Computing, Physica-Verlag, 1996, pp. 95-129.
8. Smith J.L., Fogarty T.C. Operator and parameter adaptation in genetic algorithms // Soft Computing, 1997, N 1(2), pp. 81-87.
9. Vishnevsky M.A., Koshur V.D., Legalov A.I., Mirkes E.M. Protective laminar composites design optimisation using genetic algorithm and parallel processing // Parallel Computing Technologies. Proceedings of 7-th International Conference, PaCT 2003, Nizhni Novgorod, Russia, September 2003, Ed. by Malyshkin V., Springer, 2003, pp. 394-400.
10. Живоглядов В.П., Медведев А.В. Непараметрические алгоритмы адаптации. Фрунзе: Илим, 1974. 134с.
11. Медведев А.В. Непараметрические системы адаптации. Новосибирск: Наука, 1983. 174 с.
12. Рубан А.И. Метод непараметрической оптимизации стохастических объектов. // Системы управления: Сборник научных работ. Вып.1. Томск: Изд-во Томского государственного университета, 1975. С. 101-107.
13. Алексеев В.И., Кориков А.М., Полонников Р.И., Тарасенко В.П. Экстремальная радионавигация. М.: Наука, 1978. 280 с.
14. Алексеев В.И. Субоптимальные рекуррентные алгоритмы оценивания в системах навигации // Изв. вузов СССР. Радиоэлектроника. 1987, Том 30, № 3. С. 34-39.

15. Рубан А.И. Метод непараметрической поисковой оптимизации // Изв. вузов. Физика. 1995, Том 38, № 9. С. 65-73.
16. Рубан А.И. Метод глобальной оптимизации функций в непрерывном пространстве // Информатика и системы управления: Межвузовский сборник научных трудов, посвященный памяти профессора Б.П. Соустина, Вып. 8. Красноярск: НИИ ИПУ, 2002. С. 161-173.
17. Рубан А.И. Глобальная оптимизация методом усреднения координат. Красноярск: ИПЦ КГТУ, 2004. 302 с.
18. Власов П.А. Глобальная оптимизация на основе генетического алгоритма и метода усреднения координат. Магистерская диссертация по направлению 552800 - информатика и вычислительная техника / Научный руководитель: д.ф.-м.н., проф. Кошур В.Д. Красноярск, КГТУ, 2005. 68 с.
19. Zadeh L.A. Fuzzy sets // Information and Control, 1965, Vol. 8. pp. 338-353.
20. Заде Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений. М: Мир, 1976.
21. Nauck D. Neuro-fuzzy systems: review and prospects // Proc. Fifth European Congress on Intelligent Techniques and Soft Computing (EUFIT'97), Aachen, Sep. 8-11, 1997. pp. 1044-1053.
22. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ-Петербург, 2003. 736 с.
23. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. – М: Горячая линия – Телеком, 2002. 382 с.
24. Осовский С. Нейронные сети для обработки информации. М: Финансы и статистика, 2002. 344 с.
25. Цыпкин Я.З. Адаптация и обучение в автоматических системах. М.: Наука, 1968. 400 с.
26. Цыпкин Я.З. Сглаженные рандомизированные функционалы и алгоритмы в теории адаптации и обучения // Автоматика и телемеханика, 1971, № 8. С. 29-50.
27. Ogawa T., Kosugi Y., Kanada H. Neural network based solution to inverse problems // Proceedings of 1998 IEEE World Congress on Computational Intelligence, IEEE, 1998, pp. 2411-2416.
28. Бессольцев Д.А., Кошур В.Д. Решение нелинейных уравнений с помощью инверсных нейронных сетей // Научная сессия МИФИ-2004. VI Всероссийская научно-техническая конференция “Нейроинформатика-2004”. Сборник научных трудов. Часть 2. М.: МИФИ, 2004. С. 45-49.
29. Кошур В.Д. Инверсный метод нейросетевых аппроксимаций для решения задач оптимизации и управления // Научная сессия МИФИ-2004. VI Всероссийская научно-техническая конференция “Нейроинформатика-2004”. Сборник научных трудов. Часть 1. М.: МИФИ, 2004. С. 217.
30. Linkens D.A., Nyongesa H.O. Learning system in intelligent control: an appraisal of fuzzy, neural and genetic algorithm control applications // IEE Proc. Control Theory and Applications. 1996, Vol. 143, N 4, pp. 367-386. [Рус. перевод: Обучающиеся системы в интеллектуальном управлении: сопоставление подходов, основанных на размытой логике, искусственных нейронных сетях и генетических алгоритмах. М.: ВИНТИ. Экспресс-информация “Астронавтика и ракетодинамика”. 1997, Вып. 8, реф. 15; 1997, Вып. 9, реф. 16.]
31. Кошур В.Д. Вычислительные технологии на основе мягких вычислений // Вычислительные технологии, 2005, Том 10, Спец. выпуск. С. 77-82.
32. Иванов В.М. Программная реализация алгоритма генерации ЛП<sub>τ</sub> последовательностей // Информатика и процессы управления. Межвузовский сборник научных статей. Красноярск: КГТУ, 1995. С. 125-130.

33. Takagi T., Sugeno M. Derivation of fuzzy control rules from human operator's control action // Proceedings of the IFAC Symposium on Fuzzy Information, Knowledge Representation and Decision Analysis, July 1983, pp. 55-60.
34. Godjevac J., Steele N. Neuro-fuzzy control of a mobile robot // Neurocomputing, 1999, N 28, pp. 127-143.

Статья поступила 23 января 2006 г.  
После доработки 4 сентября 2006 г.