УДК 519.6+004.02

Л.П. Вакал

ПОИСК ОПТИМАЛЬНЫХ УЗЛОВ ДЛЯ СЕГМЕНТНОЙ АППРОКСИМАЦИИ

Введение

При сегментной аппроксимации промежуток приближения разбивается на ряд подынтервалов (сегментов) и на каждом из них функция аппроксимируется некоторым аналитическим выражением (многочленом, дробно-рациональной функцией и др.). Непрерывность аппроксимирующей функции в узлах разбиения промежутка аппроксимации на сегменты в общем случае не требуется. Существует немало прикладных задач, например задачи геометрического моделирования плоских и пространственных объектов с кусочно-гладкими границами, сжатия численной информации, построения функциональных преобразователей и др. [1–3], в которых аппроксимирующая функция может быть разрывной — лишь бы погрешность приближения на всем промежутке была достаточно малой.

Принципиально различается два случая сегментной аппроксимации: с фиксированными узлами разбиения на подынтервалы и со свободными узлами. Погрешность сегментного приближения намного меньше, если используются свободные узлы. В случае фиксированных узлов задача сегментной аппроксимации линейна, а в случае свободных узлов — нелинейная задача многомерной оптимизации.

В данной статье предлагается алгоритм чебышевского кусочно-многочленного приближения функций со свободными узлами, в котором для поиска оптимальных узлов используется метод дифференциальной эволюции (ДЭ). Он относится к группе эволюционных алгоритмов, моделирующих базовые процессы биологической эволюции (отбор, мутацию и воспроизводство). Алгоритм ДЭ прост в реализации и использовании (содержит мало варьируемых параметров), легко распараллеливается.

Постановка задачи

Пусть $f(x)$ — непрерывная на отрезке $[\alpha,\beta]$ функция, r — натуральное число, $P(x)=a_0+a_1x+\ldots+a_nx^n$ — многочлен степени не выше n, T — совокупность разбиений $\tau=\{\alpha\le t_0<t_1<\ldots<t_r\le\beta\}$ отрезка $[\alpha,\beta]$ на r сегментов $[t_{j-1},\ t_j]$ $(j=\overline{1,r})$, где $t_0,t_1,\ldots,t_r$ — узлы разбиения. Обозначим $L(t_{j-1},t_j)$ наилучшее чебышевское (равномерное) приближение функции $f(x)$ многочленами степени n на сегменте $[t_{j-1},t_j]$

$$L(t_{j-1},t_j)=\min_P \max_{x\in[t_{j-1},t_j]}\left|f(x)-P(x)\right|=\max_{x\in[t_{j-1},t_j]}\left|f(x)-P_j(x)\right|. \qquad (1)$$

Функция вида

$$S(x)=\sum_{j=1}^{r}P_j(x)\cdot\Theta[(x-t_{j-1})(t_j-x)],\ \Theta(x)=\begin{cases}0, x<0\\1, x\ge 0,\end{cases} \qquad (2)$$

где каждый из многочленов $P_j(x)$ — многочлен наилучшего равномерного приближения для функции $f(x)$ на соответствующем сегменте $[t_{j-1},t_j]$, называется чебышевским кусочно-многочленным приближением для функции $f(x)$ на отрезке $[\alpha,\beta]$. В общем случае функция $S(x)$ является разрывной в узлах $t_1,\ldots,t_{r-1}$.

Рассматривается случай чебышевской кусочно-многочленной аппроксимации со свободными узлами. Заданными считаются степень многочленов n и число сегментов r, но сами сегменты, образующие покрытие отрезка $[\alpha, \beta]$, не фиксируются.

Положим $L(\tau) = \max\limits_{1 \le j \le r} L(t_{j-1}, t_j)$. Разбиение τ^* отрезка $[\alpha, \beta]$ называется оптимальным, а его узлы $t_0^*,\ t_1^*, \ldots,\ t_r^*$ — оптимальными узлами, если

$$L(\tau^*) = \inf_{\tau \in \mathrm{T}} L(\tau). \tag{3}$$

Для любой непрерывной на отрезке $[\alpha, \beta]$ функции оптимальное разбиение существует, хотя может быть и неединственным [4, 5]. Задача сегментной аппроксимации заключается в нахождении оптимального разбиения τ^* и значения погрешности аппроксимации $L(\tau^*)$ для этого разбиения. Эта погрешность будет минимальной для заданного числа сегментов r.

Разбиение τ называется разбиением с равными уклонениями, если

$$L(t_{j-1}, t_j) = L(t_j, t_{j+1}),\ \ j = \overline{1, r-1}. \tag{4}$$

Известно [5], что для любой непрерывной на отрезке $[\alpha, \beta]$ функции $f(x)$ разбиение с равными уклонениями существует и является оптимальным (но оптимальное разбиение может и не быть разбиением с равными уклонениями). Ставится задача — найти для заданной функции $f(x)$ разбиение с равными уклонениями. Следует заметить, что если функция $f(x)$ удовлетворяет требованиям $f(x) \in C^{n+1}[\alpha, \beta]$, $f^{(n+1)}(x) \ne 0$ для $x \in [\alpha, \beta]$, то при выполнении условий (4) чебышевское кусочное приближение $S(x)$ многочленами нечетной степени n будет непрерывным [5].

Разработан ряд алгоритмов чебышевской сегментной аппроксимации многочленами со свободными узлами [5–8], в основе которых лежит требование выполнения условий оптимальности (4). Например, в [6] соотношения (4) рассматриваются как система трансцендентных уравнений относительно неизвестных $t_1, \ldots, t_{r-1}$ ($t_0 = \alpha$, $t_r = \beta$), и задача сводится к решению системы методом Ньютона, сходимость которого напрямую зависит от начального выбора узлов.

В данной статье для поиска разбиения с равными уклонениями и построения соответствующего чебышевского кусочно-многочленного приближения с минимальной погрешностью для функции $f(x)$ на отрезке $[\alpha, \beta]$ предлагается подход, в котором неизвестные свободные узлы $t_1, \ldots, t_{r-1}$ (узлы $t_0 = \alpha$ и $t_r = \beta$ известны) находятся как решение задачи минимизации

$$\begin{aligned}
&\Lambda(t_1, \ldots, t_{r-1}) = \min, \\
&\left| L(t_j, t_{j+1}) - L(t_{j-1}, t_j) \right| \le \Lambda,\ \ j = \overline{1, r-1}, \\
&\Lambda \ge 0, \alpha < t_1 < \ldots < t_{r-1} < \beta
\end{aligned}$$

с помощью метода дифференциальной эволюции.

Алгоритм вычисления оптимальных узлов для сегментной аппроксимации

Метод дифференциальной эволюции предназначен для нахождения глобального оптимума недифференцированных, нелинейных, мультимодальных функций многих переменных и принадлежит к прямым методам оптимизации, т.е. в ходе его работы требуется вычисление только значения целевой функций (критерия оптимизации), но не ее производных.

Предложенный Р. Сторном и К. Прайсом [9] алгоритм ДЭ относится к классу стохастических алгоритмов (т.е. работает с использованием случайных чисел) и состоит из таких шагов. На первом шаге генерируется некоторое множество векторов (поколение популяции), представляющих собой возможные решения задачи оптимизации. Далее на каждой итерации алгоритма (эпохе эволюционного процесса) создается новое поколение. Для каждого вектора старого поколения, называемого базовым вектором (target vector), генерируется мутантный вектор (mutant vector) с использованием трех других случайных векторов и операций сложения и вычитания их координат. Затем над мутантным вектором выполняется операция скрещивания (crossover), в ходе которой некоторые его координаты замещаются координатами базового вектора. Полученный после скрещивания вектор называется пробным (trial vector). Если он оказывается лучше базового вектора (значение целевой функции улучшилось), то в новом поколении базовый вектор заменяется на пробный, в противном случае базовый вектор сохраняется в новом поколении. Такое правило отбора гарантирует неизменность размера популяции в процессе работы алгоритма. На каждой эпохе эволюционного процесса в целях контроля скорости поиска оптимального решения определяется лучший вектор поколения. Условиями окончания эволюционного процесса могут быть, например, достижение удовлетворительного значения критерия оптимизации, исчерпание заданного максимального числа поколений и др.

В целом алгоритм ДЭ представляет собой одну из возможных «непрерывных» модификаций генетического алгоритма. В то же время он имеет существенную особенность, во многом определяющую его свойства. В качестве источника шума при мутации в алгоритме ДЭ используется не внешний генератор случайных чисел, а «внутренний», реализованный как разность между случайно выбранными векторами текущей популяции. Благодаря этому алгоритм ДЭ обладает способностью динамически моделировать особенности рельефа оптимизируемой функции, подстраивая под них распределение «встроенного» источника шума. Именно этим объясняется способность алгоритма быстро проходить сложные овраги, обеспечивая эффективность даже в случае сложного рельефа.

Для вычисления оптимальных узлов и нахождения чебышевского кусочно-многочленного приближения с минимальной погрешностью предлагается следующий алгоритм дифференциальной эволюции.

1. Генерируется начальное поколение векторов $V_i = (v_{1i}, \ldots, v_{r-1,i})$, $i = \overline{1, Np}$, где Np — размер популяции (один из параметров настройки алгоритма). Координаты каждого вектора — упорядоченные по возрастанию случайные числа из интервала (α, β)

$$v_{ji} = \text{rand}(0,1) \cdot (\beta - \alpha), \quad i = \overline{1, Np}, \quad j = \overline{1, r-1},$$

где $\text{rand}(0,1)$ — случайное число из интервала $(0,1)$. Заметим, что координата с индексом j соответствует узлу t_j.

2. Для базового вектора V_i $(i = \overline{1, Np})$ из старого поколения выбирается три случайных вектора V_b, V_c, V_d $(b \neq c \neq d \neq i)$ и генерируется мутантный вектор $\widetilde{V}_b$ по правилу

$$\widetilde{V}_b = V_b + Fm \cdot (V_c - V_d),$$

где Fm — некоторая положительная вещественная константа из интервала $[0, 2]$, которая называется силой мутации и является параметром алгоритма. Сила мутации Fm определяет амплитуду возмущений, вносимых в вектор V_b внешним шумом.

3. Задается вероятность скрещивания Cr (еще один параметр алгоритма ДЭ), с которой потомок U_i наследует искаженный мутацией генетический признак от «родительского» вектора V_b. Соответствующий признак от базового вектора V_i (второго «родителя») наследуется с вероятностью $(1-Cr)$. Далее вычисляются координаты пробного вектора U_i по формуле

$$u_{ji} = \begin{cases} \widetilde{v}_{jb}, & \text{если } \mathrm{rand}(0,1) \le Cr \vee j = j_{\mathrm{rand}}, \\ v_{ji}, & \text{если } \mathrm{rand}(0,1) > Cr \wedge j \ne j_{\mathrm{rand}}. \end{cases}$$

4. Для включения в новое поколение выбирается тот из векторов U_i и V_i, значение целевой функции которого меньше. Такой оператор выбора гарантирует, что лучшее значение целевой функции не может быть пропущено, что приводит к быстрой сходимости алгоритма.

Целевая функция F для каждого вектора вычисляется по формуле

$$F(V_i) = \max_{1 \le j \le r} \left| \rho_{j-1} - \rho_j \right|, \; i = \overline{1, Np},$$

где $\rho_j = L(\nu_{j-1,i}, \nu_{ji})$ — погрешность наилучшей чебышевской многочленной аппроксимации (1) для функции $f(x)$ на сегменте $[v_{j-1,i}, v_{ji}]$, $j = \overline{1, r}$ ($\nu_{0i} = \alpha$, $\nu_{ri} = \beta$). Заметим, что выбор такой целевой функции обусловлен стремлением найти разбиение с равными уклонениями (погрешностями аппроксимации на сегментах), так как в этом случае будет выполнено достаточное условие оптимальности (4).

Для вычисления ρ_j ($j = \overline{1, r}$) применяется алгоритм [10], реализующий второй метод Ремеза [11]. Поскольку алгоритм предназначен для приближения функций, заданных на сетке, то каждый сегмент предварительно покрывается равномерной сеткой из m точек.

5. Алгоритм ДЭ завершает эволюционный процесс, если выполняется одно из условий:

— значение целевой функции лучшего вектора в поколении меньше заданного ε;

— исчерпано максимальное число поколений популяции $p_{\max}$;

— происходит стагнация итерационного процесса, т.е. относительный разброс значений целевой функции в популяции меньше заданной величины δ

$$\max_{i=\overline{1, Np}} F(V_i) - \min_{i=\overline{1, Np}} F(V_i) < \delta \cdot \min_{i=\overline{1, Np}} F(V_i).$$

В противном случае осуществляется переход к шагу 2. По умолчанию задается $\varepsilon = 10^{-12}$, $p_{\max} = 200$, $\delta = 10^{-4}$.

6. Для лучшего вектора узлов из последнего поколения, участвовавшего в эволюционном процессе, на каждом из сегментов вычисляются коэффициенты многочлена наилучшего приближения (1). Результатом работы алгоритма является кусочно-многочленный аппроксимант, который имеет наименьшую погрешность приближения функции $f(x)$ среди всех аппроксимантов вида (2) с таким же числом сегментов r на отрезке $[\alpha, \beta]$.

Следует заметить, что ввиду стохастического характера алгоритма ДЭ для получения приемлемого решения задачи следует выполнить алгоритм несколько раз.

Анализ результатов вычислительных экспериментов

Для проверки эффективности предложенного алгоритма ДЭ для поиска оптимальных узлов чебышевского кусочного приближения выполнена серия вычислительных экспериментов по аппроксимации функций многочленами разных степеней и с различным числом узлов разбиения отрезка аппроксимации на сегменты. Анализ полученных результатов свидетельствует о том, что с помощью алгоритма ДЭ удается найти более точные значения оптимальных узлов и построить чебышевское кусочно-многочленное приближение с меньшей погрешностью, чем с помощью сложных детерминистических алгоритмов.

В качестве иллюстрации рассмотрим аппроксимацию функции $f(x)=(1+x^2)^{-1}$ на отрезке $[-5, 5]$ многочленами степеней $n = 3, 4, 5$ с числом сегментов $r=\overline{2, 6}$. Для вычисления оптимальных узлов и чебышевских кусочно-многочленных приближений применялся алгоритм ДЭ с такими настройками: размер популяции $Np = 50$, сила мутации $Fm = 0{,}4$, вероятность скрещивания $Cr = 0{,}9$, количество точек сетки $m = 501$, число запусков алгоритма равно десяти (значения ε, $p_{\max}$, δ — по умолчанию). В табл. 1, где приведены полученные по алгоритму ДЭ результаты, первое число в клетках таблицы означает погрешность сегментной аппроксимации $\rho=\max\limits_{1\le j\le r}\rho_j$, второе — значение целевой функции, остальные числа — внутренние узлы (внешние узлы совпадают с концами отрезка аппроксимации). В табл. 1 приведены также результаты детерминистического алгоритма [7], для которого дополнительно вычислены максимальные значения разностей погрешностей аппроксимации на сегментах для сравнения со значениями целевой функции алгоритма ДЭ.

Таблица 1

r	Детерминистический алгоритм			Алгоритм ДЭ		
	$n = 3$	$n = 4$	$n = 5$	$n = 3$	$n = 4$	$n = 5$
2	0,04476 0 0	0,0375 0 0	0,0215 0 0	0,047709 0 0	0,037547 0 0	0,037547 0 0
3	0,0142 $1{,}5\cdot10^{-4}$ – 0,6745 0,6616	0,00265 $3{,}5\cdot10^{-5}$ – 0,5643 0,7845	0,00137 $1{,}2\cdot10^{-5}$ – 0,6738 0,6737	0,0141892 $2{,}6\cdot10^{-12}$ – 0,667998 0,667998	0,0026042 $6{,}6\cdot10^{-12}$ – 0,777154 0,777154	0,0013649 $6{,}0\cdot10^{-12}$ – 0,674394 0,674394
4	0,00586 $9{,}0\cdot10^{-5}$ – 1,2839 0 1,2839	0,000905 $1{,}4\cdot10^{-7}$ – 1,4899 0 1,4899	0,000313 $7{,}7\cdot10^{-6}$ – 1,1361 0 1,1361	0,0058119 $1{,}5\cdot10^{-13}$ – 1,280404 0 1,280404	0,00090483 $4{,}3\cdot10^{-13}$ – 1,489922 0 1,489922	0,00031280 $8{,}3\cdot10^{-13}$ – 1,142365 0 1,142365
5	0,00286 $1{,}6\cdot10^{-4}$ – 1,6614 – 0,4102 0,4018 1,6341	0,000466 $7{,}1\cdot10^{-6}$ – 1,7745 – 0,5374 0,5399 1,7775	0,000103 $1{,}6\cdot10^{-6}$ – 1,7225 – 0,4019 0,4102 1,7287	0,0027244 $7{,}8\cdot10^{-12}$ – 1,656935 – 0,407128 0,407128 1,656935	0,00046049 $8{,}7\cdot10^{-11}$ – 1,779121 – 0,539921 0,539921 1,779121	0,00010275 $8{,}8\cdot10^{-12}$ – 1,725198 – 0,405822 0,405822 1,725198
6	0,000451 $5{,}8\cdot10^{-6}$ – 2,4581 – 0,7826 0 0,7826 2,4581	0,000239 $1{,}5\cdot10^{-6}$ – 2,0411 – 0,7452 0 0,7452 2,0411	$4{,}42\cdot10^{-5}$ $2{,}7\cdot10^{-7}$ – 2,0329 – 0,7841 0 0,7841 2,0329	0,00044794 $6{,}0\cdot10^{-13}$ – 2,455901 – 0,780985 0 0,780985 2,455901	0,00023846 $7{,}9\cdot10^{-13}$ – 2,038807 – 0,744142 0 0,744142 2,038807	$4{,}4296\cdot10^{-5}$ $4{,}2\cdot10^{-13}$ – 2,032335 – 0,784319 0 0,784319 2,032335

Выполнено сравнение предложенного ДЭ и непрерывного генетического алгоритма (ГА) по времени выполнения (в секундах) и числу итераций (поколений популяции). Как показывают данные табл. 2, где приведены усредненные результаты сегментной аппроксимации для случая $r = 3$, число итераций ГА меньше, чем алгоритма ДЭ, однако в ГА для нахождения оптимальных узлов требуется больше времени, чем в алгоритме ДЭ. В среднем время выполнения одной итерации в алгоритме ДЭ в два раза меньше, чем в ГА.

Таблица 2

Np	Генетический алгоритм			Алгоритм ДЭ		
	Число итераций	Время выполнения, с	Время одной итерации, с	Число итераций	Время выполнения, с	Время одной итерации, с
20	37,1	0,9	0,024	66,3	0,8	0,012
30	51,3	1,7	0,033	56,7	1,0	0,018
40	49,9	2,2	0,044	52,8	1,1	0,021
50	37,2	2,1	0,056	49,3	1,4	0,028
60	36,6	2,3	0,063	52,3	1,7	0,033
70	33,9	2,5	0,074	53,4	2,1	0,039

Наименьшее время для нахождения решения затрачивается в алгоритме ДЭ при использовании популяций небольшого размера $Np = 20, 30$ (см. табл. 2). В то же время в экспериментах установлено, что для малочисленных популяций завершение алгоритма происходит преимущественно вследствие выполнения условия стагнации итерационного процесса, например, для $Np = 30$ — в среднем в 41 % случаев при $r = 3$ и в 60 % случаев при $r = 4$, а для $Np = 20$ — в 77 % и 98 % случаев соответственно. Это связано с тем, что малый размер популяции не соответствует сложности решаемой задачи, поскольку не обеспечивается достаточное разнообразие векторов для поиска оптимального решения. Поэтому в случае окончания алгоритма ДЭ по условию стагнации рекомендуется увеличить размер популяции, например вдвое, и произвести рестарт алгоритма.

В экспериментах исследовалось также влияние на скорость сходимости алгоритма ДЭ таких его параметров, как сила мутации Fm и вероятность скрещивания Cr. Сила мутации характеризует максимально возможное расстояние, на которое может расшириться область поиска оптимума по одной переменной за одно поколение популяции. Установлено, что с увеличением силы мутации Fm растет и число поколений, необходимых для завершения алгоритма. В то же время при малых значениях Fm высок процент выходов из алгоритма по условию стагнации итерационного процесса. Рекомендуемые значения для силы мутации лежат в диапазоне $0{,}4 \le Fm \le 0{,}6$. Что касается вероятности скрещивания Cr, то с увеличением значения этого параметра число поколений, необходимое для нахождения оптимума целевой функции, уменьшается. И наоборот, при уменьшении значения Cr каждое новое поколение все меньше отличается от старого, поэтому для получения оптимального решения следует увеличить максимальное количество поколений $p_{\max}$. По результатам вычислительных экспериментов рекомендуется выбирать значения этого параметра в интервале $0{,}8 \le Cr \le 1$.

Заключение

В настоящей работе представлен алгоритм дифференциальной эволюции, адаптированный для решения задачи поиска оптимальных узлов и соответствующего чебышевского кусочно-многочленного приближения. Алгоритм ДЭ — один из лучших стохастических алгоритмов, который стабильно находит оптимум функции за минимальное время благодаря способности динамически моделировать особенности рельефа оптимизируемой функции. Он прост в реализации и использовании (содержит мало параметров, требующих подбора).

Результаты вычислительных экспериментов показали, что алгоритм ДЭ позволяет точнее найти значения оптимальных узлов и получить меньшую погрешность чебышевской сегментной аппроксимации, чем сложные детерминистические алгоритмы. Кроме того, время выполнения одной итерации алгоритма ДЭ в среднем в два раза меньше, чем непрерывного генетического алгоритма.

Проанализировано влияние варьируемых параметров (размера популяции, силы мутации, вероятности скрещивания) на скорость сходимости алгоритма и сформулированы рекомендации по выбору их наилучших значений.

Поскольку расположение оптимальных узлов для кусочно-многочленной аппроксимации отображает особенности поведения приближаемой функции, то найденные с помощью алгоритма ДЭ узлы могут использоваться при аппроксимации функций многочленными сплайнами с фиксированными узлами (с необходимыми условиями гладкости в узлах). В этом случае оптимальные узлы следует считать фиксированными и по ним вычислять соответствующее сплайн-приближение.

Л.П.Вакал

ПОШУК ОПТИМАЛЬНИХ ВУЗЛІВ ДЛЯ СЕГМЕНТНОЇ АПРОКСИМАЦІЇ

Запропоновано алгоритм диференціальної еволюції для пошуку оптимальних вузлів чебишовської (рівномірної) кусково-поліноміальної апроксимації. Наведено результати апроксимації за допомогою алгоритму. Виконано порівняння запропонованого алгоритму з детерміністичними алгоритмами сегментного наближення многочленами та з неперервним генетичним алгоритмом.

L.P.Vakal

OPTIMAL KNOTS SEARCHING FOR SEGMENT APPROXIMATION

It is proposed a differential evolution algorithm of optimal knots searching for Chebyshev (uniform) piecewise-polynomial approximation. It is presented approximation results with using the algorithm. A comparision of this algorithm with deterministic algorithms of segment polynomial approximation and with continuous genetic algorithm is made.

1. *Попов Б.А.* Равномерное приближение сплайнами. — Киев : Наук. думка, 1989. — 272 с.
2. *Исаев В.К., Плотников С.А.* Обратная задача Чебышева и сплайны Чебышева // Труды Математического института РАН. — 1995. — **211**. — С. 164–185.
3. *Каленчук-Порханова А.А., Вакал Л.П.* Наилучшая чебышевская аппроксимация для сжатия численной информации // Компьютерная математика. — 2009. — № 1. — С. 99–107.
4. *Lawson C.L.* Characteristic properties of the segmented rational minmax approximation problem // Numer. Math. — 1964. — **6**, N 4. — P. 293–301.
5. *Вершик А.М.*, *Малоземов В.Н.*, *Певный А.Б.* Наилучшая кусочно-полиномиальная аппроксимация // Сибирский математический журнал. — 1975. — **16**, № 5. — С. 925–938.
6. *Вопросы* теории и элементы программного обеспечения минимаксных задач. — Л. : Изд-во Ленинградского университета, 1977. — 192 с.
7. *Nürnberger G.*, *Sommer M.*, *Strauss H.* An algorithm for segment approximation // Numer. Math. — 1986. — **48**, N 4. — P. 463–477.
8. *Meinardus G.*, *Nürnberger G.*, *Sommer M.*, *Strauss H.* Algorithm for piecewise polynomials and splines with free knots // Math. of Computation — 1989. — **53**, N 187. — P. 235–247.
9. *Storn R., Price K.* Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces // Journal of Global Optimization. — 1997. — **11**. — P. 341–359.
10. *Каленчук-Порханова А.А., Вакал Л.П.* Аппарат аппроксимации в составе программного обеспечения суперкомпьютера с кластерной архитектурой // Искусственный интеллект. — 2009. — № 1. — С. 158–165.
11. *Ремез Е.Я.* Основы численных методов чебышевского приближения. — Киев : Наук. думка, 1969. — 623 с.
12. *Vakal L.P.* Solving uniform nonlinear approximation problem using continuous genetic algorithm // Journal of Automation and Information Sciences. — 2016. — **48**, N 6. — P. 49–59.

Получено 06.07.2016

Статья представлена к публикации акад. НАН Украины А.В. Палагиным.