

Масштабируемые параллельные алгоритмы глобальной оптимизации со смешанной локально-глобальной стратегией*

К.А.Баркалов, В.В.Рябов, С.В.Сидоров

Данная работа продолжает развитие информационно-статистического подхода к минимизации многоэкстремальных функций при невыпуклых ограничениях, получившего название индексного метода глобальной оптимизации. При этом решение многомерных задач сводится к решению эквивалентных им одномерных. Редукция основана на использовании кривых Пеано, однозначно отображающих единичный отрезок вещественной оси на гиперкуб. Также используется схема построения множества кривых Пеано («вращаемые развертки»), которую можно эффективно применять при решении задачи на кластере с десятками и сотнями процессоров. Основное внимание уделяется применению смешанной локально-глобальной стратегии для ускорения сходимости параллельного алгоритма.

1. Введение

Данная работа продолжает развитие известного подхода к минимизации многоэкстремальных функций при невыпуклых ограничениях, описанного в работах [1–6] и получившего название индексного метода глобальной оптимизации. Подход основан на раздельном учете каждого ограничения задачи и не связан с использованием штрафных функций. При этом решение многомерных задач сводится к решению эквивалентных им одномерных. Соответствующая редукция основана на использовании кривых Пеано (называемых также развертками Пеано), однозначно отображающих единичный отрезок вещественной оси на гиперкуб. Используется схема построения множества кривых Пеано («вращаемые развертки»), которую можно применять при решении задачи на кластерных системах с десятками и сотнями процессоров. Данная схема распараллеливания дополняется применением смешанной локально-глобальной стратегии поиска, заключающейся в чередовании итераций индексного метода и его локально-адаптивной модификации, что приводит к ускорению сходимости. Приведены результаты экспериментов по сравнению смешанного алгоритма с исходным индексным методом, а также с известным алгоритмом DIRECT [9], убедительно подтверждающие достоинство параллельной схемы построения множественных отображений в совокупности со смешанной стратегией поиска. Эксперименты выполнены на вычислительном кластере ННГУ им.Н.И.Лобачевского, установленном в ходе выполнения нацпроекта «Образование».

2. Постановка задачи

Рассмотрим задачу глобальной оптимизации вида

$$\begin{aligned} \varphi = \varphi(y) = \min \{ \varphi(y) : y \in D, g_j(y) \leq 0, 1 \leq j \leq m \}, \\ D = \{ y \in R^N : a_i \leq y_i \leq b_i, 1 \leq i \leq N \}, \end{aligned} \quad (1)$$

где целевая функция $\varphi(y)$ (в дальнейшем обозначаемая также $g_{m+1}(y)$) и левые части ограничений $g_j(y), 1 \leq j \leq m$, удовлетворяют условию Липшица с соответствующими константами $L_j, 1 \leq j \leq m+1$, а именно

$$|g_j(y_1) - g_j(y_2)| \leq L_j |y_1 - y_2|, \quad 1 \leq j \leq m+1, \quad y_1, y_2 \in D.$$

Используя кривые типа развертки Пеано $y(x)$, однозначно отображающие отрезок $[0, 1]$ на N -мерный гиперкуб D

$$D = \{ y \in R^N : -2^{-1} \leq y_i \leq 2^{-1}, 1 \leq i \leq N \} = \{ y(x) : 0 \leq x \leq 1 \},$$

* Работа выполнена при поддержке совета по грантам Президента Российской Федерации (грант № МК-1536.2009.9), а также Федерального агентства по науке и инновациям, госконтракт № 02.740.11.5018.

исходную задачу можно редуцировать к следующей одномерной задаче:

$$\varphi(y(x)) = \min \{ \varphi(y(x)) : x \in [0, 1], g_j(y(x)) \leq 0, 1 \leq j \leq m \}.$$

Рассматриваемая схема редукции размерности сопоставляет многомерной задаче с липшицевой минимизируемой функцией и липшицевыми ограничениями одномерную задачу, в которой соответствующие функции удовлетворяют равномерному условию Гельдера (см. [1]), т.е.

$$|g_j(y(x')) - g_j(y(x''))| \leq K_j |x' - x''|^{1/N}, \quad x', x'' \in [0, 1], \quad 1 \leq j \leq m+1,$$

где N есть размерность исходной многомерной задачи, а коэффициенты K_j связаны с константами Липшица L_j исходной задачи соотношениями $K_j \leq 4L_j \sqrt{N}$.

Различные варианты индексного алгоритма для решения одномерных задач и соответствующая теория сходимости представлены в работах [3], [5].

3. Использование множественных отображений

Редукция многомерных задач к одномерным с помощью разверток имеет такие важные свойства, как непрерывность и сохранение равномерной ограниченности разностей функций при ограниченности вариации аргумента. Однако при этом происходит потеря части информации о близости точек в многомерном пространстве, так как точка $x \in [0, 1]$ имеет лишь левых и правых соседей, а соответствующая ей точка $y(x) \in R^N$ имеет соседей по 2^N направлениям. А при использовании отображений типа кривой Пеано близким в N -мерном пространстве образам y', y'' могут соответствовать достаточно далекие прообразы x', x'' на отрезке $[0, 1]$. Как результат, единственной точке глобального минимума в многомерной задаче соответствует несколько (не более 2^N) локальных экстремумов в одномерной задаче, что, естественно, ухудшает свойства одномерной задачи.

Сохранить часть информации о близости точек позволяет использование множества отображений

$$Y_L(x) = \{y^1(x), \dots, y^L(x)\} \quad (2)$$

вместо применения единственной кривой Пеано $y(x)$ (см. [2], [4]). Каждая кривая Пеано $y^i(x)$ из $Y_L(x)$ может быть получена в результате некоторого сдвига вдоль главной диагонали гиперинтервала D . Таким образом сконструированное множество кривых Пеано позволяет получить для любых близких образов y', y'' , отличающихся только по одной координате, близкие прообразы x', x'' для некоторого отображения $y^i(x)$.

3.1 Вращаемые развертки

К числу недостатков ставшей уже классической схемы построения множественных разверток (далее будем называть их сдвиговыми развертками или С-развертками) можно отнести, во-первых, наличие дополнительного ограничения, порождающего сложную допустимую область на одномерных отрезках. А во-вторых, при построении С-разверток число разверток L (а следовательно, и число параллельно решаемых задач) зависело от требуемой точности ε поиска решения задачи. С позиции экономии вычислительных ресурсов было невыгодно использовать число разверток большее, чем $\lceil \log_2(\varepsilon^{-1}) \rceil$. Например, при решении задачи с точностью 10^{-3} по координате целесообразно было выбирать число разверток, не больше 10.

Преодолеть эти недостатки, сохранив информацию о близости точек в N -мерном пространстве, позволяет схема построения множественных отображений, предложенная в [7]. Отличительной чертой этой схемы является посторонние множества кривых Пеано не с помощью сдвига вдоль главной диагонали гиперкуба, а поворотом развертки вокруг начала координат. При этом найдется отображение $y^i(x)$, которое точкам многомерного пространства y', y'' , которым при исходном отображении соответствовали достаточно далекие прообразы на отрезке $[0, 1]$, будет сопоставлять более близкие прообразы x', x'' .

Развертки, порождаемые в соответствии с новой схемой, будем называть вращаемыми развертками или В-развертками.

Максимальное число различных поворотов развертки, отображающей N -мерный гиперкуб на одномерный отрезок, составляет 2^N . Использование всех из них является избыточным, тре-

буется выбрать лишь часть из всех возможных вариантов. В предложенной схеме преобразование развертки осуществляется в виде поворота на угол $\pm\pi/2$ в каждой из координатных плоскостей. Число подобных пар поворотов определяется числом координатных плоскостей пространства, которое равно $C_N^2 = \frac{N(N-1)}{2}$, а общее число преобразований будет равно $N(N-1)$.

Учитывая исходное отображение, приходим к заключению, что данный способ позволяет строить до $N(N-1)+1$ развертки для отображения N -мерной области на соответствующие одномерные отрезки. При этом дополнительное ограничение, которое возникало при построении S -разверток [2], отсутствует. В случае необходимости данный способ построения множества отображений может быть легко «отмасштабирован» для получения большего (вплоть до 2^N) числа разверток.

3.2 Параллельный индексный метод

Использование множества отображений $Y_L(x) = \{y^1(x), \dots, y^L(x)\}$ приводит к формированию соответствующего множества одномерных многоэкстремальных задач

$$\min \{ \varphi(y^l(x)) : x \in [0, 1], g_j(y^l(x)) \leq 0, 1 \leq j \leq m \}, 1 \leq l \leq L. \quad (3)$$

Каждая задача из данного набора может решаться независимо, при этом любое вычисленное значение $z = g_v(y^l)$, $y^l = y^l(x)$ функции $g_v(y)$ в i -й задаче может интерпретироваться как вычисление значения $z = g_v(y^s)$, $y^s = y^s(x)$ для любой другой s -й задачи без повторных трудоемких вычислений функции $g_v(y)$. Подобное информационное единство позволяет решать исходную задачу (1) путем параллельного решения индексным методом L задач вида (3) на наборе отрезков $[0,1]$. Каждая одномерная задача решается на отдельном процессоре. Для организации взаимодействия на каждом процессоре создается L очередей, в которые процессоры помещают информацию о выполненных итерациях. Используемая схема не содержит какого-либо единого управляющего процессора, что увеличивает надежность выполняемых вычислений.

Подробное описание решающих правил параллельного индексного алгоритма глобальной оптимизации приведено в работе [8].

3.3 Локально-адаптивный и смешанный алгоритмы

Локально-адаптивный алгоритм является модификацией индексного метода глобального поиска, состоящей в том, что, начиная с некоторого шага, при выборе точек итераций используется дополнительная информация – текущие оценки плотности вероятности для расположения точки искомого оптимума. Оценки плотности определяются по значениям функционалов задачи, вычисленных в точках выполненных итераций. Таким образом, плотность переоценивается после каждой итерации, причем максимумы плотности соответствуют окрестностям точек текущих оптимальных значений.

В случае локально-адаптивного метода правила вычисления характеристик $R(i)$ индексного метода, приведённые, например, в [6], преобразуются к виду

$$R'(i) = \frac{R(i)}{\sqrt{(z_i - z_v^*)(z_{i-1} - z_v^*)} / \mu_v + 1.5^{-\alpha}}, \quad v(x_{i-1}) = v(x_i) \quad (4)$$

$$R'(i) = \frac{R(i)}{(z_i - z_v^*) / \mu_v + 1.5^{-\alpha}}, \quad v(x_{i-1}) > v(x_i) \quad (5)$$

$$R'(i) = \frac{R(i)}{(z_{i-1} - z_{v-1}^*) / \mu_{v-1} + 1.5^{-\alpha}}, \quad v(x_{i-1}) < v(x_i), \quad (6)$$

где величины $z_v^* = \begin{cases} \varphi^*(y'(x)), & v = m + 1 \\ g_v^*(y'(x)), & g_v^*(y'(x)) > 0, v \leq m, \\ 0 & g_v^*(y'(x)) \leq 0, v \leq m \end{cases}$ а целочисленный параметр

$\alpha, 0 \leq \alpha \leq 30$, указывает “степень локальности” метода. При $\alpha=0$ метод имеет “глобальный”, а при $\alpha=30$ – локальный характер.

Смешанный алгоритм является модификацией индексного метода глобального поиска, состоящей в том, что, начиная с некоторого шага итерации, определяемые правилами индексного метода, чередуются с итерациями, определяемыми правилами локально-адаптивного алгоритма. Частота чередования является параметром метода.

Смешанная схема адаптирована к параллельному индексному методу с вращаемыми развёртками.

4. Реализация в программной системе Global Expert

На кафедре математического обеспечения ЭВМ ННГУ им. Лобачевского разрабатывается программный комплекс параллельной глобальной оптимизации Global Expert, одной из особенностей которого является быстрая работа с большими объёмами поисковой информации (т.е. совокупностью всех точек, в которых вычислялись функции задачи), включая собственную подсистему подкачки, учитывающую особенности алгоритмов глобального поиска. В частности для ускорения работы для каждой загруженной в оперативную память страницы используется кэширование с помощью очереди характеристик $R(i)$, чтобы извлекать интервалы с наибольшей характеристикой быстрее. В смешанной схеме удобно для каждого элементарного интервала хранить две характеристики: $R(i)$ (для глобального индексного метода) и $R'(i)$ (для итераций локально-адаптивного алгоритма, см. (4)–(6)). Это потребовало добавления к каждой странице данных ещё одной очереди характеристик для $R'(i)$. Соответственно модифицирована и схема управления всеми очередями характеристик в условиях подкачки. Хотя стратегия вытеснения и подгрузки страниц в оперативную память для смешанной схемы может быть ещё более оптимизирована, но с учётом преобладания в большинстве случаев глобальных итераций (обычно каждая четвёртая итерация – локально-адаптивная) текущая стратегия по-прежнему достаточно эффективна.

Смешанная локально-глобальная схема поддерживается и в режиме возобновления вычислений после предшествующей остановки поиска (например, по исчерпанию вычислительных ресурсов).

5. Результаты экспериментов

При сравнении методов глобальной оптимизации, ориентированных на определённые классы задач, является актуальным вопрос выбора тестовых задач для сравнения. В литературе встречается немало различных классов липшицевых многоэкстремальных функций для минимизации, многие из которых генерируются автоматически. Среди подобных генераторов одним из наиболее востребованных является GKLS, описанный, например, в [12] и доступный для свободного скачивания. Отличительной особенностью GKLS является прозрачный способ задания сложности генерируемых задач: количество локальных минимумов, размеры областей притяжения и многое другое. Всё это во многом определило использование GKLS в данной работе.

5.1 Классы тестовых функций

В работе [13] для сравнения методов глобальной оптимизации описывается 6 классов тестовых задач по 100 функций каждый. В таблице 1 приведены параметры GKLS-генератора для каждого класса. Здесь N – размерность, M – количество локальных минимумов, f^* – значение функции в точке глобального оптимума, d – расстояние точки глобального оптимума от вершины базового параболоида, r_g – радиус области притяжения глобального оптимума.

Таблица 1. Параметры GKLS-генератора для классов тестовых функций.

Класс	N	M	f^*	d	r_g
1-simple	2	10	-1.0	0.66	0.33
2-hard	2	10	-1.0	0.90	0.20
3-simple	3	10	-1.0	0.66	0.33
4-hard	3	10	-1.0	0.90	0.20
5-simple	4	10	-1.0	0.66	0.33
6-hard	4	10	-1.0	0.90	0.20

Область поиска – $[-1.0, 1.0]^N$.

5.2 Сравнение с алгоритмами DIRECT и LBDIRECT

Известный алгоритм глобальной оптимизации DIRECT и его модификация LBDIRECT (locally-biased DIRECT) подробно описаны в [9–11], а из работы [13] использованы результаты экспериментов для приведённых классов тестовых функций для двух указанных алгоритмов, а также для исходного индексного метода глобального поиска, который также известен как алгоритм Стронгина с индексной схемой учёта ограничений (обозначим его AG, как в [13]). Введём также следующие сокращения: индексный метод с вращаемыми развёртками – AG-R, индексный метод с вращаемыми развёртками и смешанной стратегией – AG-R-mixed, индексный метод с одной развёрткой и смешанной стратегией – AG-mixed.

Как и в работе [13], здесь для каждой задачи использовалось правило остановки $\|y - y^*\| \leq \rho$, т.е. достижение известного глобального оптимума с заданной точностью. Для задач размерности $N=2,3$ использовалось $\rho = 0.01 \cdot \sqrt{N}$, а для $N=4$ – $\rho = 0.02 \cdot \sqrt{N}$.

Параметры индексного метода и его модификаций: плотность развёртки $m=10$, параметр надёжности для AG-mixed – $r=4.3$, для AG-R-mixed – $r=3.2, 3.8$ (в зависимости от числа развёрток). В AG-R-mixed каждая четвёртая итерация – локально-адаптивная с параметром $\alpha = 15$.

В таблицах 2 и 3 приведены максимальные и средние количества испытаний, затраченные методами для достижения глобального оптимума с заданной точностью. В случае, если не все 100 задач были решены, в скобках указано количество нерешённых задач. Прочерк означает неприменимость метода с указанным количеством развёрток для задач слишком маленькой размерности, поскольку максимальное число развёрток в текущей реализации $L=N(N-1)$.

Таблица 2. Сравнение AG-R-mixed и других алгоритмов в худшем случае.

Класс задач	N	Максимальное число испытаний									
		Direct	LB Direct	AG	AG-mixed $r=4.3$	AG-R $L=2$ $r=3.8$	AG-R $L=4$ $r=3.8$	AG-R $L=6$ $r=3.2$	AG-R-mixed $L=2$ $r=3.8$	AG-R-mixed $L=4$ $r=3.8$	AG-R-mixed $L=6$ $r=3.2$
1-simple	2	127	165	239	207	587	-	-	221	-	-
2-hard	2	1159	2665	938	90000 (2)	90000 (2)	-	-	90000 (1)	-	-
3-simple	3	1179	1717	3945	1287	3505	5330	4276	1301	961	1013
4-hard	3	77951	85931	26964	90000 (2)	11672	15057	18535	7513	9025	90000 (1)
5-simple	4	90000 (1)	90000 (15)	27682	9684	31611	36551	17399	7057	5221	9697
6-hard	4	90000 (43)	90000 (65)	90000 (1)	82923	90000 (2)	90000 (2)	90000 (3)	90000 (2)	61613	81081

Таблица 3. Сравнение AG-R-mixed и других алгоритмов в среднем.

Класс задач	N	Максимальное число испытаний									
		Direct	LB Direct	AG	AG-mixed $r=4.3$	AG-R $L=2$ $r=3.8$	AG-R $L=4$ $r=3.8$	AG-R $L=6$ $r=3.2$	AG-R-mixed $L=2$	AG-R-mixed $L=4$	AG-R-mixed $L=6$

									$r=3.8$	$r=3.8$	$r=3.2$
1-simple	2	68.1	70.7	90.1	82.3	198.1	-	-	94.4	-	-
2-hard	2	208.6	304.3	333.1	1968.1	2168.7	-	-	1080.0	-	-
3-simple	3	238.1	355.3	817.7	380.2	1359.1	1963.9	1239.7	346.7	370.5	336.8
4-hard	3	5857.2	9990.6	3541.8	3809.6	4483.3	5652.7	4125.5	1798.9	1689.8	2543.1
5-simple	4	>12206	>23452	3950.4	1644.1	4179.2	5752.7	3470.0	1387.5	1536.1	1315.6
6-hard	4	>57333	>65326	>22315	19788	>24038	>27548	>28791	>18642	16416.1	16934.9

Полученные данные убедительно показывают в целом превосходство последовательного смешанного индексного метода с вращаемыми развёртками, особенно с ростом размерности и сложности решаемых задач. Заметим, однако, что подбор одинакового для всех классов задач параметра надёжности r затруднён: 2-й, 4-й и 6-й классы задач порой решаются на 98-99%, а для одной-двух оставшихся задач небольшое варьирование r приводит к их решению. Здесь намеренно не приводятся результаты такого варьирования, хотя в результатах, взятых из [13], такое варьирование присутствует.

5.3 Результаты вычислений на кластере

Исследование эффективности и масштабируемости параллельного смешанного индексного метода с вращаемыми развёртками (AG-R-mixed) проведём при решении существенно многомерных задач (данная схема распараллеливания допускает распределение вычислений максимум на $N(N-1)$ вычислительных ядер). В данной работе было выбрано несколько задач.

Первая – задача условной многоэкстремальной оптимизации на основе одной из функций Растригина ($N=8$):

$$\left\{ \begin{array}{l} f(y) = \sum_{i=1}^n (y_i^2 - 10 \cdot \cos(2 \cdot \pi \cdot y_i)) \rightarrow \min \\ g_1(y) = \sum_{i=1}^n y_i - 0.5 \leq 0 \\ g_2(y) = \sum_{i=1}^n y_i^2 - 1.0 \leq 0 \\ g_3(y) = \sum_{i=1}^n \sin(i \cdot \pi \cdot y_i) - 0.3 \leq 0 \\ y \in [-1.0, 1.0]^N \end{array} \right.$$

Глобальный минимум в этой задаче достигается в точке $(0, \dots, 0)^N$ со значением $f=0.0$. Результаты решения задачи последовательным и параллельным индексным методом со смешанной стратегией приведены в таблице 4. В обоих случаях метод завершал работу по достижении заданной относительной точности $\varepsilon = 0.0035$. Ещё большая точность приводит ко включению подкачки в последовательном методе и, следовательно, к большому росту вычислительных затрат, поэтому для более корректного сравнения была выбрана именно эта точность.

Таблица 4. Сравнение последовательного и параллельного AG-R-mixed на задаче с ограничениями.

	последовательный, 56 развёрток	параллельный, 56 процессов	
		всего (максимум на один процесс)	ускорение с поправкой на разное число испытаний*
Количество испытаний	309'640	244'356 (4891)	
Количество вычислений критерия	33'874	22'722 (789)	
Ускорение по испытаниям		63.3	49.95
Ускорение по вычислениям критерия		42.93	28.8

* Поправка в данном случае – это умножение на величину p/s , где p – число испытаний, сделанных параллельным методом, а s – последовательным.

Время вычислений	40006.6 сек.	626.8 сек.	
Ускорение по времени		63.83	50.37
Найденное минимальное значение целевой функции	0.016412	0.022182	

На задачах большей размерности сравнить последовательную и параллельную реализации AG-R-mixed становится значительно сложнее, поскольку в последовательном случае количество хранимой поисковой информации пропорционально числу развёрток, из-за чего включается подкачка, а в параллельном – все данные распределяются по процессам, что позволяет не использовать подкачку.

Выход: сравнивать последовательный AG-mixed (с одной развёрткой) и параллельный AG-R-mixed (с p развёртками, где p – число процессов). Но в этом случае необходимо брать различные параметры надёжности ($r=3.0$ для AG-mixed, $r=2.5$ – для AG-R-mixed) и разную относительную точность в условиях останковки. Таким образом, адекватное измерение ускорения по числу испытаний не представляется возможным. Однако, можно измерить ускорение по времени и эффективность нахождения глобального оптимума при одинаковом условии останковки по числу испытаний. В таблице 5 приведены результаты безусловной минимизации функции Растригина $f(y) = \sum_{i=1}^n (y_i^2 - \cos(18 \cdot y_i^2)) \rightarrow \min$ при $N=11, 15$ и 20 . Минимум этой функции достигается в точке $(0, \dots, 0)^N$ с оптимальным значением $f^* = -N$.

Таблица 5. Результаты минимизации многомерных функций Растригина.

	AG-mixed			AG-R-mixed (100, 110 процессов)			
	Кол-во испытаний	Время, сек.	Оценка f^*	Кол-во испытаний	Время, сек. (процессов)	Оценка f^*	Ускорение по времени
N=11	5'000'000	544'068.8	-10.997543	4'993'599	9'321.5 (110)	-10.995598	112.01
N=15	3'000'000	605'788.4	-14.835308	3'000'016	7'414.6 (100)	-14.925409	81.7
N=20	3'000'000	605'956.7	-19.989635	3'000'056	6'654.7 (100)	-19.048833	91.06

Некруглое число испытаний, сделанное параллельным методом, связано с его асинхронностью и, следовательно, недетерминированным срабатыванием условия останковки на всех процессах.

6. Заключение

В рамках данной работы программно реализована модификация последовательного и параллельного индексного метода с вращаемыми развёртками путём внедрения смешанной локально-глобальной стратегии. Проведено исследование эффективности этой стратегии в последовательном случае и эффективности распараллеливания индексного метода с её использованием. Результаты наглядно подтверждают ускорение сходимости, а эффективный параллелизм позволяет ещё дальше отодвинуть порог сложности решаемых задач.

Отметим также, что существуют параллельные реализации метода DIRECT и его модификаций. Сравнение с ними параллельного индексного метода может стать одним из дальнейших направлений исследований.

Литература

1. Стронгин Р.Г. Поиск глобального оптимума. М.: Знание, 1990.
2. Стронгин Р.Г. Параллельная многоэкстремальная оптимизация с использованием множества развёток // Ж. вычисл. матем. и матем. физ. 1991. Т.31. №8. С. 1173 – 1185.
3. Стронгин Р.Г., Баркалов К.А. О сходимости индексного алгоритма в задачах условной оптимизации с ε -резервированными решениями // Математические вопросы кибернетики. М.: Наука, 1999. С. 273 – 288.

4. Strongin R.G., Sergeyev Ya.D. Global optimization with non-convex constraints. Sequential and parallel algorithms. Kluwer Academic Publishers, Dordrecht, 2000.
5. Баркалов К.А., Стронгин Р.Г. Метод глобальной оптимизации с адаптивным порядком проверки ограничений // Ж. вычисл. матем. и матем. физ. 2002., Т.42, №9. С. 1338–1350.
6. Баркалов К.А. Ускорение сходимости в задачах условной глобальной оптимизации. Нижний Новгород: изд-во Нижегородского гос. ун-та, 2005.
7. Баркалов К.А., Рябов В.В., Сидоров С.В. Использование кривых Пеано в параллельной глобальной оптимизации // Материалы Девятой международной конференции-семинара "Высокопроизводительные параллельные вычисления на кластерных системах", Владимир, 2009, стр. 44 - 47.
8. Стронгин Р.Г., Гергель В.П., Баркалов К.А. Параллельные методы решения задач глобальной оптимизации // Известия высших учебных заведений. Приборостроение. – Т. 52. №10, 2009.–С. 25–32.
9. Jones D.R., Perttunen C.D., Stuckman B.E. Lipschitzian optimization without the Lipschitz constant // Journal of Optimization Theory and Applications. – 1993. – N. 79. – P. 157–181.
10. Gablonsky M.J. Modifications of the DIRECT Algorithm // Ph.D. thesis, North Carolina State University, Raleigh, NC, 2001.
11. Gablonsky M.J., Kelley C.T. A locally-biased form of the DIRECT Algorithm // Journal of Global Optimization. – 2001. – Vol. 21. – P. 27–37.
12. Gaviano M., Kvasov D.E., Lera D., Sergeyev Ya.D. Software for generation of classes of test functions with known local and global minima for global optimization: [<http://si.deis.unical.it/~yaro/GKLS.html>] // ACM TOMS. – 2003. – Vol. 29, N. 4. – P. 469–480.
13. Lera D., Sergeyev Ya.D. Lipschitz and Hölder global optimization using space-filling curves // Applied Numerical Mathematics. – January 2010. – Vol. 60, N. 1–2, P. 115–129.