

# ОБРАБОТКА ИЗОБРАЖЕНИЙ, РАСПОЗНАВАНИЕ ОБРАЗОВ

## К ВОПРОСУ ОБ ЭФФЕКТИВНОСТИ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ФУНКЦИЙ МНОГИХ ПЕРЕМЕННЫХ

Коварцев А.Н., Попова-Коварцева Д.А.

Самарский государственный аэрокосмический университет имени академика С.П. Королёва  
(национальный исследовательский университет)

### Аннотация

Рассматривается проблема построения эффективных параллельных алгоритмов глобальной оптимизации. Приведены результаты качественного анализа возможности преодоления экспоненциального роста сложности задачи глобальной оптимизации для функций общего вида при использовании часто применяемых алгоритмических приёмов ускорения сходимости (условия Липшица, редукции, локальной техники). Показано, что построение эффективных алгоритмов глобальной оптимизации для размерностей 100 и более переменных возможно для частных постановок задач, учитывающих специфические особенности оптимизируемой функции. В качестве примера приведён класс «хороших» функций.

**Ключевые слова:** глобальная оптимизация, гладкие функции многих переменных, параллельные алгоритмы, анализ сложности алгоритмов.

### Введение

Алгоритмы глобальной оптимизации находят широкое применение в разнообразных областях науки и техники, везде, где необходимо получить наилучший результат целевой функции, оценивающей качество принимаемого решения. В силу важности проблемы в настоящее время разработано колоссальное количество алгоритмов и методов решения задачи многоэкстремальной оптимизации. С некоторыми из них можно ознакомиться в достаточно подробной обзорной статье [1].

Сегодня разработка методов глобальной оптимизации стимулируется развитием электронно-вычислительных средств и во многом связана с доступностью параллельных компьютерных систем высокой производительности. Появилось большое количество различных подходов к распараллеливанию алгоритмов глобальной оптимизации. Однако в публикациях, посвящённых этой тематике, редко обсуждается проблема преодоления «проклятия размерности» задачи многомерной глобальной оптимизации.

Ещё в 70-е годы прошлого века для многоэкстремальной задачи оптимизации  $k$  раз непрерывно дифференцируемой функции  $n$  переменных была получена нижняя оценка стохастической сложности [2]:

$$\tilde{N}(\varepsilon) \geq c(n, k) \left( \frac{1}{\varepsilon} \right)^{n/k}, \quad (1)$$

где  $\varepsilon$  – заданная погрешность решения оптимизационной задачи по координате,  $c(n, k)$  – некоторый коэффициент, зависящий от свойств функции. Из выражения (1) ясно, что при заданной точности увеличение размерности задачи оптимизации приводит к катастрофическому росту её сложности. В связи с чем непонятно, за счёт каких скрытых ресурсов, используя кластерные системы, авторы рассчитывают преодолеть экспоненциальный рост сложности задачи глобальной оптимизации, когда количество

процессоров в кластере можно увеличить в лучшем случае линейно.

Теория алгоритмов допускает два способа повышения алгоритмической эффективности решаемой задачи:

- 1) существенно снизить требования по точности к формируемым решениям (искать квазиоптимальные решения);
- 2) несколько изменить исходную постановку задачи.

Первый способ малоперспективен, поскольку загромождение точности выхолащивает саму постановку задачи глобальной оптимизации. В то же время с ростом  $n$  многие авторы вынужденно снижают требования к точности при поиске глобального экстремума [3]. Второй способ, в сущности, связан с выделением класса многоэкстремальных функций, допускающих алгоритмы поиска глобального экстремума полиномиальной сложности и имеющих практическое значение. В данной работе производится попытка качественного анализа эффективности задач глобальной оптимизации.

### 1. Постановка задачи

Без ограничения общности рассмотрим задачу глобальной оптимизации в следующей постановке:

$$f(x) \rightarrow \min_{x \in D_0},$$

где  $D_0 = \bigotimes_{i=1}^n [0, 1]$  – допустимое множество задачи оптимизации (многомерный единичный гиперкуб),  $f(\cdot)$  – многоэкстремальная функция цели. Обычно  $f(x)$  является гладкой невыпуклой функцией, обладающей определёнными свойствами. Как и для большинства алгоритмов оптимизации, положим, что функция удовлетворяет условию Липшица с константой  $L$ , т.е. выполняется неравенство

$$|f(x) - f(z)| \leq L \|x - z\|_\infty. \quad (2)$$

Здесь и далее  $\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$ .

Большинство методов глобальной оптимизации используют идею неравномерного покрытия допустимого множества, которая с учётом свойств оптимизируемой функции позволяет существенно сократить сложность поиска в допустимом множестве. В [3] был предложен метод половинных делений, предлагающий удобный для анализа задачи глобальной оптимизации алгоритм формирования неравномерной сетки покрытия допустимого множества. Методология этого метода используется в данной работе.

### 2. Метод половинных делений

Для метода половинных делений характерно непропорциональное деление области поиска на параллелепипеды меньшей размерности: единичный куб на две прямоугольные области, затем на четыре квадрата четвертной размерности и т.д. (рис. 1). В методе половинных делений в пространстве поиска формируется нерегулярная сетка испытаний исследуемой функции. При этом в зависимости от размерности пространства  $n$  по-разному изменяется шаг дискретизации. Например, для  $n=1$  имеет место следующая последовательность деления шага дискретизации:  $1/2; 1/4; 1/8; 1/16; \dots$ ; для  $n=2$ :  $1/2; 1/4; 1/4; 1/8; 1/8; \dots$  и т.д.

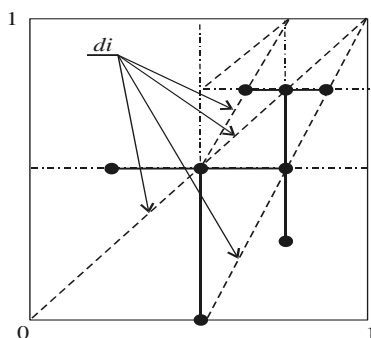


Рис. 1. Деление допустимого множества

В процессе расчётов начальный параллелепипед  $D_0$  делится на прямоугольные параллелепипеды  $D_i$  с центрами  $c_i$  и главными диагоналями  $d_i$ . Метод половинных делений [3] сводится к построению последовательности  $B_k = \{D_1, D_2, \dots, D_k\}$  параллелепипедов  $D_i \subset D_0$ , причём значения функции вычисляются в соответствующей последовательности центров параллелепипедов  $N_k = \{c_1, c_2, \dots, c_k\}$ . Для функций, удовлетворяющих условию Липшица, на каждом шаге алгоритма выбирается параллелепипед, для которого

$$\varphi_s = \min_{1 \leq i \leq k} f(c_i) - (L/2) \|d_i\|_\infty. \quad (3)$$

Используя условие Липшица, некоторые параллелепипеды исключаются из дальнейшего рассмотрения. Для остановки алгоритма можно использовать условие

$$\|d_i\|_\infty \leq \varepsilon. \quad (4)$$

Определим число обращений к функции до достижения заданного размера «измельчения» параллелепипеда (до заданного уровня точности по аргументу решения задачи оптимизации).

Пусть  $n = 1$ . Процесс половинных делений параллелепипедов формирует двоичное дерево, и в этом случае сложность алгоритма можно оценить величиной

$$N_m(1) = 2^m - 1,$$

где  $m$  – высота бинарного дерева.

Для  $n = 2$  уровни дерева делений будем группировать по длине шага дискретизации. Другими словами, на один уровень поставим все вершины, полученные в процессе делений, когда центры параллелепипедов предшественника и приемника отстоят друг от друга на одинаковом расстоянии  $h_m$ .

Все вершины дерева делений на каждом уровне условно разобьём на две группы: внутренние и внешние (рис. 2), порождающие последующее деление параллелепипедов на следующем уровне.

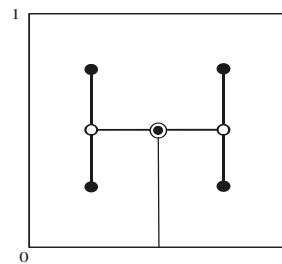


Рис. 2. Шаблон дерева делений для  $n = 2$

Несложно показать, что на  $m$ -м уровне дерева делений количество вершин распределяется следующим образом:  $S_m = S_m^{внут} + S_m^{внеш} = 2^{2m-1} + 2^m$ . Тогда общее число вершин, включая  $m$ -й уровень дерева, можно подсчитать по формуле

$$N_m(2) = 2 + 2^3 + \dots + 2^{(2m-1)} + 2^2 + 2^4 + \dots + 2^m = 2 \cdot 2^{2(m-1)} - 1.$$

Для  $n = 3$  шаблон дерева делений представлен на рис. 3. На втором шаге ( $h_2 = 0,25$ ) построения дерева делений, для параллелепипеда с центром  $c_0 = (0,5; 0,5; 0,5)$  (рис. 3) шаблон дерева делений содержит 6 внутренних вершин и 8 внешних. Причём справедливо  $S_{m+1}^{внут} = 6 \cdot S_m^{внут}$ ,  $S_{m+1}^{внеш} = 8 \cdot S_m^{внеш}$ . Тогда общее число вершин, включая на  $m$ -й уровень, можно подсчитать по формуле

$$N_m(3) = 6(8^0 + 8^1 + \dots + 8^{(m-1)}) + 8^1 + 8^2 + \dots + 8^m = 2 \cdot 2^{3(m-1)} - 1.$$

В общем случае, используя метод математической индукции, можно показать, что дерево делений до  $m$ -го уровня включительно содержит

$$N_m(n) = 2 \cdot 2^{n(m-1)} - 1 \quad (5)$$

точек вычисления функции. Для параллелепипедов  $m$ -го уровня дерева делений справедливо

$\|d_i\|_\infty = 2h_m = 2/2^m$ , учитывая условие остановки (4), получаем  $2^{m-1} \geq 1/4\epsilon$ , тогда

$$N(n) = 2 \cdot \left(\frac{1}{4\epsilon}\right)^n - 1. \tag{6}$$

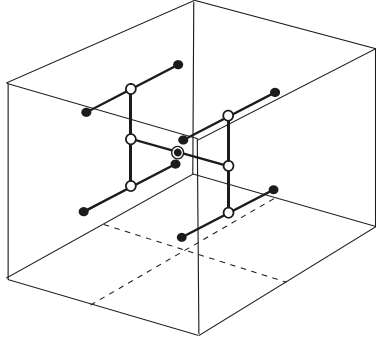


Рис. 3. Шаблон дерева делений для  $n=3$

### 3. Анализ сложности алгоритмов глобальной оптимизации

Если не делать специальных предположений о свойствах оптимизируемой функции, то из формул (5) и (6) следует вывод об экспоненциальном росте сложности задачи глобальной оптимизации. Например, при оптимизации непрерывной функции с точностью до  $\epsilon = 0,0078$ , оценки числа испытаний функции в зависимости от размерности задачи оптимизации представлены в табл. 1.

Таблица 1. Зависимость числа испытаний функции от размерности  $n$

$n$	2	3	4	5	6
$N(n)$	2047	65535	2097151	67108863	2,147E+09

При таком росте сложности задачи оптимизации очевидно, что невозможно повысить эффективность алгоритма за счёт распараллеливания вычислительных процессов. Однако большинство методов оптимизации используют те или иные условия, позволяющие в процессе оптимизации исключать из рассмотрения целые зоны из области поиска, в нашем случае – параллелепипеды. Если не учитывать специфические особенности оптимизируемой функции, то в общем случае универсальных правил прореживания области поиска не так уж много. Обычно это условия, использующие ограниченность скорости роста функции и, возможно, производных более высоких порядков. В любом случае механизм прореживания областей деления приводит к удалению части ветвей дерева делений. Предположим, что для некоторой функции на каждом уровне дерева делений для дальнейшего поиска остаются в среднем только  $\alpha$  (в долях) параллелепипедов, из числа внешних вершин каждого уровня. Причём  $(1-\alpha)$  параллелепипедов отбраковываются, например, с использованием критерия Липшица. Можно показать, что в этом случае средняя сложность задачи глобальной оптимизации оценивается величиной

$$\tilde{N}(n) = \beta \cdot (\alpha 2^n)^{m-1} - 1, \tag{7}$$

где  $\beta = \frac{(\alpha + 1)2^n - 2}{\alpha 2^n - 1}$ , а  $\alpha$  должно удовлетворять условию  $\alpha > 1/2^n$ .

Если положить  $\mu = \lfloor \log_2 \alpha \rfloor$ , выражение (7) примет вид

$$\tilde{N}(n) = \beta \cdot (2^{n-\mu})^{m-1} - 1,$$

или в зависимости от заданной точности

$$\tilde{N}(n) = \beta \cdot \left(\frac{1}{4\epsilon}\right)^{n-\mu} - 1. \tag{8}$$

Из формулы (8) следует, что даже при очень благоприятных условиях, связанных либо со свойствами функции, либо эффективными алгоритмическими решениями, когда  $\alpha$  равно 0,5 или даже 0,25, сложность задачи глобальной оптимизации остаётся экспоненциальной, но на несколько порядков ниже, чем для случая (6). Этот факт позволяет разрабатывать приемлемые по сложности алгоритмы распараллеливания задачи глобальной оптимизации для сравнительно небольших размерностей.

### 4. Метод редукции

В последнее время появилось большое количество алгоритмов, основанных на редукции сложности, когда решение исходной задачи заменяется решением нескольких более простых задач. В большинстве случаев ставка делается на эффективность одномерных алгоритмов глобальной оптимизации, например, на информационно-статистический подход Р.Г. Стронгина [4]. К методам редукции пространства поиска непосредственно примыкают разнообразные методы «развёрток», например, с помощью кривых Пеано.

Метод редукции рассмотрим в классической постановке. Пусть многомерная задача глобальной оптимизации решается с помощью редукции вида

$$\min_{x_1, \dots, x_n} f(x_1, \dots, x_n) = \min_{x_1} \min_{x_2} \dots \min_{x_n} f(x_1, \dots, x_n). \tag{9}$$

Для  $n = 1$  из (8) следует, что

$$\tilde{N}(1) = \beta \cdot (1/4\epsilon)^{1-\mu} - 1,$$

тогда в многомерном случае справедливо

$$\tilde{N}(n) = \left[ \beta \cdot \left(\frac{1}{4\epsilon}\right)^{1-\mu} - 1 \right]^n. \tag{10}$$

В табл. 2 сведены оценки сложности задачи глобальной оптимизации функции многих переменных для разных методов её решения и свойств самой функции.

Из таблицы видно, что и метод редукции принципиально не изменяет экспоненциальный характер роста сложности задачи глобальной оптимизации. Однако даже при незначительном уменьшении коэффициента «прореживания»  $\alpha$  существенно расширяется диапазон размерности задачи глобальной оптимизации, когда за счёт распараллеливания можно добиться ощутимых конечных результатов.

Более того, при использовании информационно-статистического алгоритма глобальной оптимизации

Р.Г. Стронгина [4] функции одной переменной диапазон размерностей, допускающий распараллеливание, расширяется ещё больше. В данном случае эффективное «прореживание» области поиска реализуется не только за счёт использования условия Липшица, но и за счёт учёта плотности вероятностей, описывающей апостериорную информацию об оптимизируемой функции.

Таблица 2. Оценка эффективности задачи глобальной оптимизации

<i>n</i>	2	3	4	5	6
<i>N(n)</i> для непрерывной функции	2047	65535	2097151	67108863	2,15E+09
<i>N(n)</i> липшицевой функции, $\alpha=0,5$	127	3412	102984	3215632	1,02E+08
<i>N(n)</i> метод редукции, $\alpha=0,75$	862	14371	273379	5499292	1,14E+08
<i>N(n)</i> метод редукции, $\alpha=0,6$	80	388	2163	12818	78622

### 5. Класс «хороших» функций

Если «проклятие размерности» не удаётся преодолеть за счёт изощрённых стратегий оптимизации, то согласно канонам теории алгоритмов можно попробовать изменить условия задачи, например, ввести класс функций, для которых задача многомерной оптимизации имеет полиномиальную сложность.

Действительно, при проектировании технических систем нечасто встречаются целевые функции, имеющие большое количество локальных минимумов с приблизительно равными значениями. Такие функции обычно предлагаются в тестах, предназначенных для апробации алгоритмов оптимизации.

Определим класс «хороших» функций формально. К ним отнесём:

1. Функции, удовлетворяющие условию Липшица (2).
2. Функции, имеющие конечное число экстремумов с явно выраженным преобладанием глобального минимума над локальными.

В принципе, такую функцию можно построить. Например, для этих целей, пригодна следующая функция:

$$f(x) = \sum_{k=1}^M \varphi_k(x), \tag{11}$$

где  $\varphi_k(x) = \gamma_k \exp\left(-\frac{(x - a_k)^2}{2\sigma_k}\right)$ .

Следует отметить, что практически для всех алгоритмов глобальной оптимизации существует начальная стадия адаптации алгоритма к исследуемой функции, хотя бы потому, что константа Липшица редко бывает известна заранее.

Пусть этап адаптации для алгоритма половинных делений продолжается до высоты дерева делений  $m_1$ , т.е. до параллелепипеда диаметра  $d_f$ , откуда  $m_1 = \log_2 1/d_f$ . Тогда на первом этапе будет произведено  $N_1(n) = 2(1/2d_f)^n - 1$  испытаний функции.

Первый этап при небольших значениях  $m_1$  может оказаться не слишком трудоёмким. На втором этапе работы алгоритма предположим, что в дереве делений реализуются только  $r$  веток, «привязанных» к локальным минимумам, а остальные – «прореживаются». Высоту «прореженной» части дерева обозначим  $m_2$ , тогда

$$m_2 = m - m_1 = \log_2 1/4\epsilon - \log_2 1/2d_f = \log_2 d_f / 2\epsilon,$$

откуда  $N_2(n) = r m_2 = r \log_2 d_f / 2\epsilon$ . В итоге имеем

$$N(n) = 2(1/2d_f)^n - 1 + r \log_2 d_f / 2\epsilon. \tag{12}$$

В таблице 3 приведены результаты оценок сложности задач глобальной оптимизации для «хорошей» функции в зависимости от  $n$ . Как и прежде, условием остановки работы алгоритма является достижение точности по аргументу  $\epsilon = 0,0078$ . Из таблицы видно, что для «хорошей» функции наблюдается ещё большее замедление экспоненциального роста сложности алгоритма половинных делений, но сложность по-прежнему экспоненциальная.

Таблица 3. Оценка эффективности задач глобальной оптимизации

<i>n</i>	2	3	4	5	6
«хорошая» функция, $d_f=0,125, r=20$	91	187	571	2107	8251
«хорошая» функция, эксперимент	828	1409	2232	2410	4699

В методах глобальной оптимизации часто используется так называемая локальная техника [1], суть которой сводится к запуску стандартных локальных алгоритмов из множества точек, принадлежащих областям притяжения локальных экстремумов. Эффективность у алгоритмов локальной оптимизации существенно выше, чем у методов глобальной оптимизации (обычно имеет место полиномиальная сложность), поэтому часто в методах глобальной оптимизации, так или иначе, используется локальная техника.

Для оценки влияния локальной техники на эффективность алгоритмов глобальной оптимизации были произведены вычислительные эксперименты, сделанные на основе модифицированного алгоритма половинных делений для «хорошей» функции вида (11). Для локальной оптимизации использовался метод деформированных многогранников. В таблице 3 приведены результаты экспериментов.

Видно, что рост сложности алгоритма глобальной оптимизации, использующего локальную технику, для «хорошей» функции имеет более пологий характер, который, главным образом, определяется полиномиальным ростом сложности алгоритмов локальной оптимизации.

### Заключение

В последнее время, вероятно, в связи с развитием высокопроизводительных вычислительных средств появилось большое количество публикаций, посвящённых задаче глобальной оптимизации. В них пред-

лагаются разнообразные схемы организации параллельных вычислений. Однако не следует забывать, что преодоление «проклятия размерности» за счёт распараллеливания вычислительных процессов в общем случае – неразрешимая задача, а использование кластерных систем – дорогостоящее «удовольствие».

Приведённые в статье результаты, естественно, носят качественный характер. Однако анализ задачи глобальной оптимизации показывает, что разработка параллельных алгоритмов оптимизации в общем случае не имеет серьёзных перспектив, когда речь идёт о размерностях в 100 и более переменных. В то же время, учёт специфических особенностей оптимизируемой функции в оригинальных алгоритмах решения частных задач глобальной оптимизации способен преодолеть экспоненциальный характер роста сложности и повысить эффективность параллельных алгоритмов.

#### Литература

1. Орлянская, И.В. Современные подходы к построению методов глобальной оптимизации / И.В. Орлянская И.В. // Электронный журнал «Исследовано в России», 2007. – С. 189-192: [web-сайт] <<http://zhurnal.ape.relarn.ru/articles/2002/189.pdf>>
2. Немировский, А.С. Сложность задач и эффективность методов оптимизации / А.С. Немировский, Д.Б. Юдин – М.: Наука, 1979. – 384 с.
3. Евтушенко, Ю.Г.. Метод половинного деления для глобальной оптимизации функции многих переменных / Ю.Г. Евтушенко, В.А. Ратькин // Техническая кибернетика. – 1987. – № 1. – С. 119-127.
4. Стронгин, Р.Г. Поиск глобального оптимума / Р.Г. Стронгин. – М.: Знание, 1990. – 240 с.
5. Гришагин, В.А. Параллельный метод решения многомерных многоэкстремальных задач с невыпуклыми ограничениями / В.А. Гришагин, Я.Д. Сергеев // V международный научно-практический семинар «Высокопроизводительные параллельные вычисления на кластерных системах». – Н. Новгород, ННГУ. 2005. – С. 74-82.

#### References

1. Orlyanskaya, I.V. Current approaches to the development of methods of global optimization / I.V. Orlyanskaya // Electronic Journal "Investigated in Russia". – 2007. – P. 189-192: [web-site] <http://zhurnal.ape.relarn.ru/articles/2002/189.pdf>. – (in Russian).
2. Nemirovsky, A.S. Complexity of the tasks and the effectiveness of optimization methods / A.S. Nemirovsky, D.B. Yudin. – Moscow: "Nauka" Publisher, 1979. – 384 p. – (in Russian).
3. Yevtushenko, Y.G. Method of half division for global optimization of functions of many variables, Yevtushenko Y.G., Ratkin V.A. // Technical Cybernetics. - 1987. № 1. - P. 119-127. – (in Russian).
4. Strongin, R.G. Search of global optimum / R.G. Strongin. – Moscow: "Znanie" Publisher, 1990. – 240 p. – (in Russian).
5. Grishagin, V.A. A parallel method for solving multidimensional multiextremal problems with nonconvex limitations / V.A. Grishagin, Ya.D. Sergeyev // V international scientific-practical seminar «High performance parallel computing on cluster systems». Nizhny Novgorod, "Nizhny Novgorod State University" Publisher. 2005. – P. 74-82. – (in Russian).

### ON EFFICIENCY OF PARALLEL ALGORITHMS FOR GLOBAL OPTIMIZATION OF FUNCTIONS OF SEVERAL VARIABLES

A.N. Kovartsev, D.A. Popova-Kovartseva

Samara State Aerospace University (National Research University)

#### Abstract

We consider the problem of constructing efficient parallel algorithms for global optimization. The results are of given the qualitative analysis of the possibility of overcoming the exponential growth of global optimization problems for functions of general form, using the commonly used algorithmic techniques to accelerate convergence (Lipschitz conditions, reduction, local technique). Shown that the construction of efficient algorithms for global optimization, for the dimensions of 100 or more variables, is possible for specific problems, taking into account the specific characteristics of the function being optimized. The class of "good" functions is shown, as an example.

Key words: global optimization, smooth function, parallel algorithms, analysis of algorithms.

#### Сведения об авторах



**Коварцев Александр Николаевич**, 1952 года рождения. В 1975 году окончил Куйбышевский авиационный институт (ныне – Самарский государственный аэрокосмический университет имени академика С.П. Королёва) по специальности «Прикладная математика». Доктор технических наук (2000 год), профессор, работает заведующим кафедрой программных систем СГАУ. Директор школы информатики СГАУ. Коварцев А.Н. – специалист в области автоматизации проектирования, разработки и тестирования сложных программных средств; надёжности программного обеспечения; моделирования параллельных вычислений и глобальной оптимизации. Имеет более 80 научных публикаций.

**Alexander Nikolaevich Kovartsev** (b. 1952) graduated with honour (1975) from the S.P. Korolyov Kuibyshev Aviation Institute (presently, S. P. Korolyov Samara State Aerospace

University (SSAU)), majoring in Applied Mathematics. He received his Doctor of Engineering (2000) degrees from Samara State Aerospace University. Professor, chair of SSAU's sub-department software systems SSAU. Director of the School of Computer SSAU. His current research interests include computer-aided design, development and testing of complex software; simulation of parallel computing and global optimization. He has more than 80 scientific publications.



**Попова-Коварцева Дарья Александровна**, 1984 года рождения. В 2007 году с отличием окончила Самарский государственный аэрокосмический университет имени академика С.П. Королёва по специальности «Автоматизированные системы обработки информации и управления». В настоящее время – аспирант кафедры программных систем. Область научных интересов: автоматизации программирования, системный анализ.

**Daria Alexandrovna Popova-Kovartseva**, born in 1984. In 2007 she graduated from S.P. Korolyov Samara State Aerospace University (SSAU), majoring in computing science. At present time – post-graduate student of the sub-department software systems SSAU. Research interests: automated programming, systems analysis.

---

*Поступила в редакцию 29 марта 2011 г.*