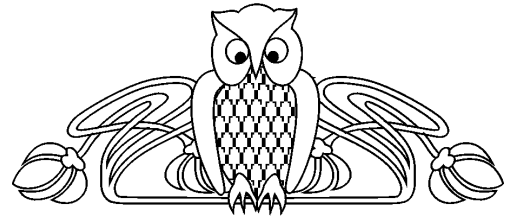




УДК 681.3.06, 681.322

## ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ РЕШЕНИЯ ЗАДАЧ ОПТИМИЗАЦИИ НА ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЁННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ



А. Н. Савин<sup>1</sup>, И. В. Дружинин<sup>2</sup>, А. А. Ерофтьев<sup>1</sup>

<sup>1</sup>Саратовский государственный университет  
E-mail: savinan@info.sgu.ru, eroftiev.andrey@gmail.com

<sup>2</sup>Московский государственный технический университет  
им. Н. Э. Баумана  
E-mail: druzhininiv@gmail.com

### The Application of a Genetic Algorithm to Global Optimization Problem Solving on Parallel and Distributed Computing Systems

A. N. Savin, I. V. Druzhinin, A. A. Eroftiev

Представлены результаты адаптации метода нахождения глобального минимума многоэкстремальной целевой функции многих переменных с ограничениями с помощью генетического алгоритма для систем параллельных и распределённых вычислений. Предложены два варианта распараллеливания генетического алгоритма. Исследована зависимость надёжности и производительности параллельных версий алгоритма от их параметров и количества узлов параллельной вычислительной системы. Показано, что предложенные параллельные варианты генетического алгоритма позволяют за небольшое время надёжно находить глобальный минимум целевой функции.

This article presents the results of the adaptation of method of searching the global minimum of multiextremal criterion function of multiple variables with constraints based on genetic algorithm for parallel and distributed computing systems. Two variants of genetic algorithm parallelization are proposed. The reliability and performance of parallel versions of an algorithm, depending on its parameters and the number of nodes in parallel computer system is investigated. The reliability of finding the global minimum of criterion function in a small amount of time with proposed parallel variants of genetic algorithm is represented.

**Ключевые слова:** глобальная оптимизация, генетический алгоритм, многоэкстремальная целевая функция, параллельные вычислительные системы.

**Key words:** global optimization, genetic algorithm, multiextreme criterion function, parallel computing system.

### ВВЕДЕНИЕ

В настоящее время при разработке новых материалов и устройств часто требуется решать оптимизационные задачи [1], которые, как известно, сводятся к поиску экстремумов целевой функции (ЦФ) различными методами [2]. Наличие всевозможных ограничений на оптимизируемые параметры и многоэкстремальность ЦФ, как правило, приводит к большим вычислительным затратам и соответственно к невозможности нахождения решения за приемлемое время при использовании одного компьютера [1].

Данная проблема может быть решена при применении современных параллельных и распределённых вычислительных систем (ВС) в сочетании с использованием эффективных распараллеленных алгоритмов оптимизации.

В данной работе была поставлена задача изучения возможности распараллеливания алгоритма нахождения глобального минимума многоэкстремальной ЦФ многих переменных с ограничениями типа равенств, основанного на использовании генетических алгоритмов (ГА) [3].

### 1. ОБЩИЙ ВИД ЗАДАЧИ ПОИСКА ГЛОБАЛЬНОГО МИНИМУМА ПРИ НАЛИЧИИ ЯВНЫХ ОГРАНИЧЕНИЙ

Поиск глобального минимума функции  $f : R^n$  при наличии явных ограничений осуществляется на некотором собственном подмножестве  $\Omega$  метрического пространства  $R^n$ :

$$f(x) = f(x_1, \dots, x_n) \rightarrow \min, x \in \Omega, \Omega \subset R^n, \quad (1)$$

где подмножество  $\Omega$  определяется ограничениями типа равенств:

$$q(x) = 0, \quad \text{где } q : R^n. \quad (2)$$

Пусть имеется некоторое множество  $X$ , состоящее из элементов  $x$ , принадлежащих подмножеству  $\Omega \subset R^n$ , и на нём определена скалярная функция  $f(x)$ . Говорят, что  $f(x)$  имеет локальный минимум



в точке  $x^*$ , если существует некоторая конечная  $\epsilon$ -окрестность этой точки, в которой выполняется условие

$$f(x^*) < f(x), \quad \|x - x^*\| \leq \epsilon.$$

У функции может быть много локальных минимумов. Если выполняется неравенство  $f(x^*) < f(x)$ ,  $x \in X$ , где  $x \neq x^*$  — любая точка множества  $X$ , то говорят о глобальном минимуме функции  $f(x)$  [4].

## 2. ПОИСК ГЛОБАЛЬНОГО МИНИМУМА С ИСПОЛЬЗОВАНИЕМ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Поиск глобального минимума функции (1) при наличии ограничений (2) на варьируемые параметры может быть осуществлён с помощью генетических алгоритмов (ГА) [3].

ГА представляют собой модель процесса эволюции в природе. В них используются аналоги механизмов генетического наследования и естественного отбора.

Эволюционная теория утверждает, что каждый биологический вид целенаправленно развивается и изменяется для того, чтобы наилучшим образом приспособиться к окружающей среде. Основным механизмом эволюции — это естественный отбор. Его суть состоит в том, что более приспособленные особи имеют больше возможностей для выживания и размножения и, следовательно, приносят больше потомства, чем плохо приспособленные особи. При этом благодаря передаче генетической информации (генетическому наследованию) потомки наследуют от родителей основные их качества. Таким образом, потомки сильных индивидуумов также будут относительно хорошо приспособленными, а их доля в общей массе особей будет возрастать. После смены нескольких десятков или сотен поколений средняя приспособленность особей данного вида заметно возрастает. В теории ГА используется биологическая терминология в упрощенном виде:

- хромосома — вектор чисел;
- ген — бит хромосомы;
- популяция — совокупность особей.

В качестве критерия жизнеспособности особи выступает функция приспособленности  $F$  — отображение совокупности хромосом особи на множество вещественных чисел. Точное выражение для функции приспособленности составляется индивидуально для каждой задачи.

В этом случае элементы  $x$  подмножества  $\Omega$  в (1) можно рассматривать как особи некоторой популяции точек  $n$ -мерного пространства, а значения функции  $f(x)$  в этих точках можно рассматривать как значение функции приспособленности  $F = -f(x)$ .

Основные шаги работы классического ГА можно записать в следующем виде:

1. Генерация начальной популяции из  $k$  особей.
2. Вычисление функции приспособленности каждой особи и отбор наименее приспособленных особей.
3. Выбор среди оставшихся особей двух родителей.
4. Применение к родительским хромосомам оператора скрещивания с вероятностью  $p_c$ ; при отсутствии скрещивания дочерние хромосомы равны родительским.
5. Применение к полученным дочерним хромосомам оператора мутации с вероятностью  $p_m$ ; в отсутствие мутации хромосомы не изменяются.
6. Повторение шагов 3–5, пока популяция вновь не будет содержать  $k$  особей.
7. Повторение шагов 1–6 до тех пор, пока не будут достигнут критерий окончания процесса.

Скрещивание (кроссовер) — это операция, при которой две хромосомы обмениваются своими частями. В модели, описываемой ГА, кроссовер выполняет оператор скрещивания  $C$ . Существует несколько алгоритмов оператора скрещивания [3], самые простые из них — одно- и многоточечный.

В одноточечном варианте происходит разрыв двух родительских хромосом в случайной позиции, после чего они обмениваются полученными участками. При многоточечном кроссовере хромосомы обмениваются участками, полученными в результате нескольких разрывов.

Мутацией называется случайное изменение одного или нескольких генов хромосомы.

В качестве критерия окончания алгоритма могут выступать 2 условия:

- достижение заданного максимального количества итераций (поколений);
- схождение популяции.



Схождением популяции называется такое ее состояние, когда все особи популяции практически одинаковы и находятся в области некоторого экстремума.

ГА контролируется следующими параметрами:

- объемом популяции  $k$ ;
- вероятностью мутации  $p_m$ ;
- вероятностью скрещивания  $p_c$ ;
- числом новых особей на каждом этапе развития популяции  $m$ ;
- максимальным числом итераций алгоритма  $N$ ;
- при использовании критерия сходимости популяции — точностью схождения  $\epsilon$ .

В данной работе требовалось исследовать применение ГА на параллельных ВС к нахождению глобальных экстремумов многоэкстремальных функций нескольких переменных в условиях наличия явных и неявных ограничений.

### 3. АДАПТАЦИЯ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ОПТИМИЗАЦИИ ДЛЯ ИСПОЛЬЗОВАНИЯ НА ПАРАЛЛЕЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

Существует несколько вариантов использования ГА на параллельных ВС. В работе [5] было предложено два варианта:

1. На  $i$ -м этапе получения потомков с помощью разделения на  $m/2$  процессов основного потока выполнения каждый процесс проводит скрещивание двух особей, затем при вычислении функции приспособленности каждый узел ВС отыскивает значения  $F_i$  (функция приспособленности на  $i$ -м этапе) для порождённых на нём особей.
2. Параллельное развитие нескольких популяций на узлах ВС с последующим выбором наилучшего результата.

Второй вариант является приемлемым в случае легко вычисляемых целевых функций (ЦФ), когда потери на пересылках данных в коммуникационной среде параллельной ВС могут быть сравнимы со временем вычисления функции приспособленности  $F$ . Также одновременная эволюция нескольких популяций может скомпенсировать возможное схождение популяции к локальному экстремуму.

В первом случае было предложено осуществлять скрещивание и вычисление значений функции приспособленности, а следовательно, и ЦФ (1), одновременно на нескольких машинах (клиентская часть алгоритма) и проверять условие сходимости на выделенной машине (серверная часть алгоритма).

Блок-схема первого параллельного варианта ГА представлена на рис. 1. Пунктиром выделена часть алгоритма, выполняемая на клиентских машинах. По достижении условия сходимости или заданного количества итераций в серверной части осуществляется выбор решения с наибольшим значением функции приспособленности, что соответствует наименьшему значению ЦФ, что должно обеспечивать глобальность найденного минимума.

Так как вычисление значений функции приспособленности  $F$ , а значит, и ЦФ производится одновременно на  $m/2$  вычислительных узлах, то при использовании  $m/2$  клиентских машин можно уменьшить временные затраты при поиске глобального минимума примерно в  $m/2$  раз. Более точное значение коэффициента ускорения  $S$  можно получить из закона Амдала:

$$S = \frac{1}{\alpha + \frac{1-\alpha}{p}},$$

где  $\alpha$  — доля вычислений, выполняемая в последовательной части,  $p$  — количество рабочих узлов в параллельной системе.

В начале работы алгоритма производится  $k$  вычислений функции приспособленности  $F$ , составляющие последовательную часть вычислений. Учитывая, что на каждой итерации алгоритма на каждом из  $m/2$  вычислительных узлов выполняется два вычисления функции приспособленности, при общем числе итераций  $I$  получим  $2\frac{m}{2}I = mI$  вычислений в параллельной части.

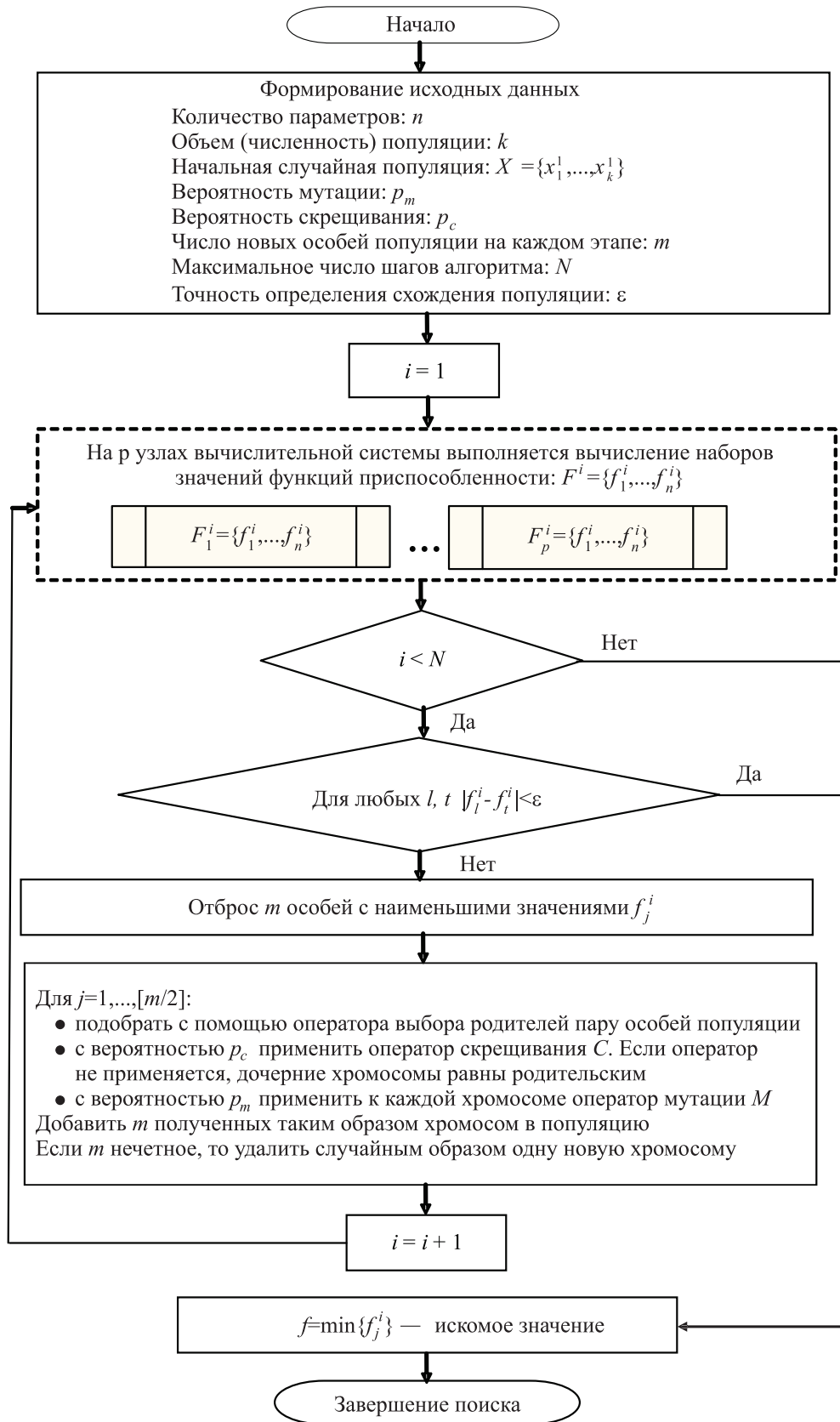


Рис. 1. Блок-схема первого варианта параллельного ГА, используемого для поиска глобального экстремума ЦФ



Таким образом, коэффициент ускорения вычислений без учета затрат на пересылку данных между вычислительными узлами равен

$$S = \frac{mI + k}{2I + k}. \quad (3)$$

Второй вариант параллельной реализации ГА предполагает одновременный запуск алгоритма на  $p$  вычислительных узлах. По окончании работы алгоритма на всех рабочих узлах параллельной ВС производится сбор данных о полученных значениях экстремума и выбор наибольшего значения функции приспособленности  $F$ , соответствующего глобальному минимуму ЦФ.

Блок-схема второго варианта распараллеливания ГА приведена на рис. 2. Пунктиром выделена часть алгоритма, выполняемая на клиентских машинах.

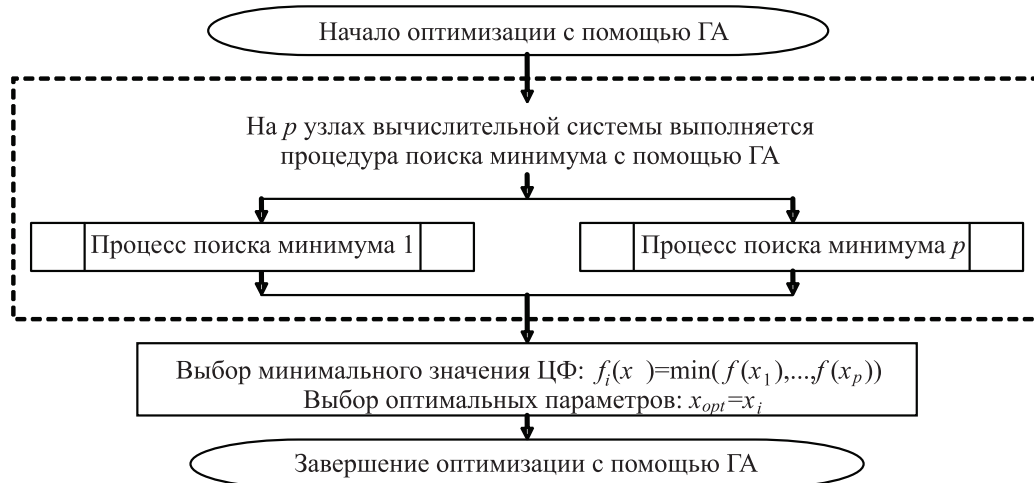


Рис. 2. Блок-схема второго варианта параллельного ГА, используемого для поиска глобального экстремума ЦФ

Каждый процесс поиска минимума, запускаемый на рабочем узле ВС, представляет собой классическую реализацию ГА. Параметры алгоритма при этом задаются одинаковыми для всех процессов.

Такой способ поиска с помощью ГА должен обеспечивать надёжное нахождение глобального минимума ЦФ за счёт многократного повторения поиска в заданной ограничением области.

При этом, так как многократный процесс поиска минимума осуществляется одновременно на соответствующем числе узлов, то затраченное на выполнение всего параллельного алгоритма время приблизительно должно равняться времени выполнения одного процесса и практически не должно зависеть от числа процессов. Соответственно при использовании  $p$  рабочих узлов можно уменьшить временные затраты при поиске глобального минимума примерно в  $p$  раз.

Для тестирования предложенных параллельных вариантов классического генетического алгоритма оптимизации разработана программа на языке программирования C++.

#### 4. ОЦЕНКА ЭФФЕКТИВНОСТИ ОПТИМИЗАЦИИ С ИСПОЛЬЗОВАНИЕМ ПАРАЛЛЕЛЬНЫХ ВАРИАНТОВ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Оценка надёжности и эффективности распараллеленного алгоритма оптимизации методом имитации отжига по схеме больцмановского тушения осуществлялась на многоэкстремальных ЦФ при количестве параметров оптимизации  $n$ , равном 4, 6 и 8:

- число параметров оптимизации  $n = 4$ :

$$f(x_1, x_2, x_3, x_4) = \sum_{j=1}^4 \{x_j^2 - 0.1 \cos(18x_j)\}, \quad (4)$$

где  $-0.5 < x_j < 1$ ,  $j = 1, \dots, 4$ ,  $f_{\min} = -0.4$  при  $x_j = 0$ ,  $j = 1, \dots, 4$ ;



- число параметров оптимизации  $n = 6$ :

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = \sum_{j=1}^6 \{x_j^2 - 0.1 \cos(18x_j)\}, \quad (5)$$

где  $-0.5 < x_j < 1, j = 1, \dots, 6, f_{\min} = -0.6$  при  $x_j = 0, j = 1, \dots, 6$ ;

- число параметров оптимизации  $n = 8$ :

$$f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = \sum_{j=1}^8 \{x_j^2 - 0.1 \cos(18x_j)\}, \quad (6)$$

где  $-0.5 < x_j < 1, j = 1, \dots, 8, f_{\min} = -0.8$  при  $x_j = 0, j = 1, \dots, 8$ .

Процесс поиска глобального минимума завершился при выполнении следующих условий: при сходимости популяции, которое оценивалось по максимальному расстоянию между значениями функции приспособленности —  $\max_k(|\Delta F|) \leq 0.00001$  или по достижении максимального количества итераций алгоритма —  $N = 10000$ .

Для оценки производительности разработанного приложения использовалось количество вычислений ЦФ, так как основное время при поиске глобального минимума с помощью ГА затрачивается на получение её значений.

Статистическая достоверность полученных данных обеспечивалась усреднением по результатам 100 запусков процесса поиска глобального минимума при каждом сочетании параметров.

Тестирование первого варианта параллельного ГА для функций (4), (5) и (6) осуществлялось при различном числе узлов параллельной ВС  $p$ . Также исследовалось влияние объема популяции  $k$ , числа новых особей на каждом этапе развития популяции  $m$ , вероятности скрещивания  $p_c$  и вероятности мутации  $p_m$  на процесс поиска глобального минимума. Пределы изменения этих параметров представлены в табл. 1.

Таблица 1

Диапазоны изменения параметров ГА в численном эксперименте

	$k$	$m$	$p_c$	$p_m$
min	10	2	0.1	0.05
max	100	16	1.0	0.1

На рис. 3–6 приведены усредненные по итогам 100 запусков ВС результаты тестирования первого варианта параллельного ГА.

На рис. 3 приведены зависимости найденных минимальных значений ЦФ с указанием 95% доверительного интервала ( $a$ – $b$ ) и числа вычислений ЦФ (4), (5), (6) ( $e$ ) от размера популяции  $k$ . При этом остальные исследуемые параметры алгоритма фиксировались ( $m = 16, p_c = 0.75, p_m = 0.1$ ).

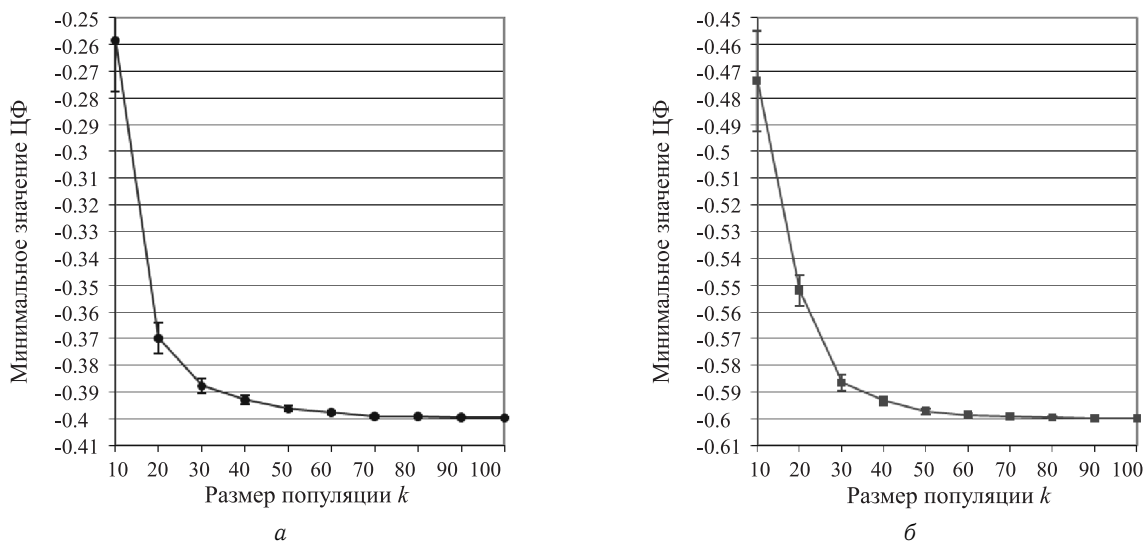
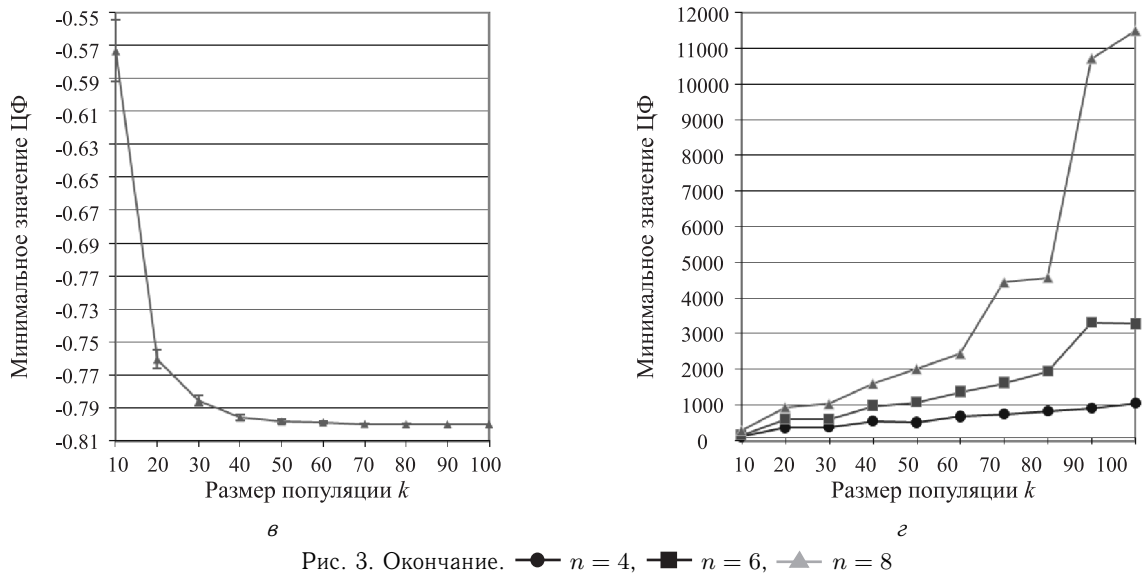


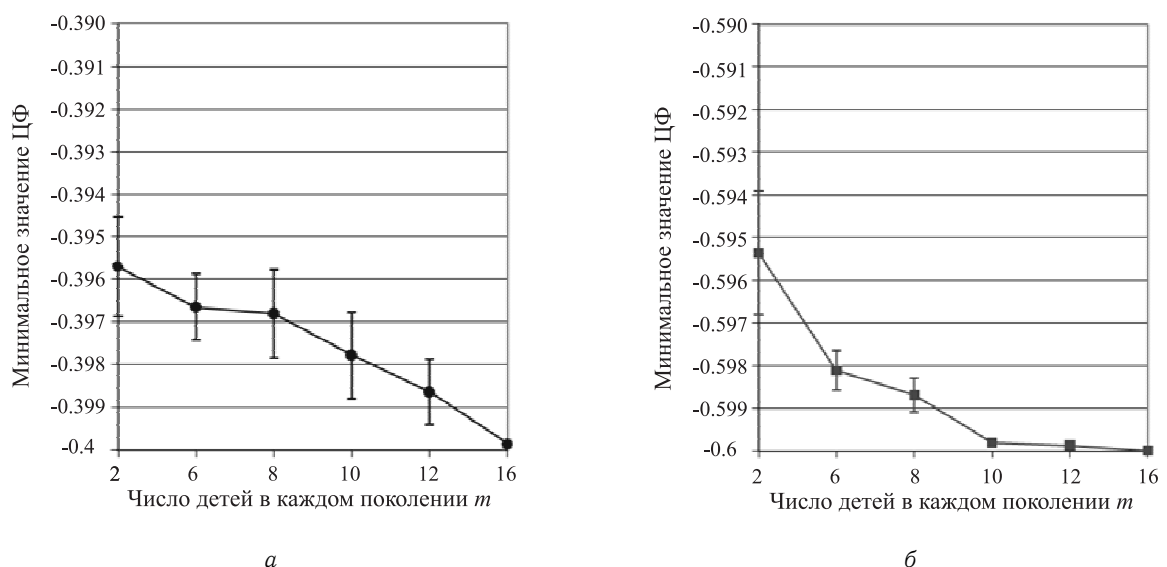
Рис. 3. Зависимость найденного минимального значения ЦФ от размера популяции  $k$  и количества параметров оптимизации  $n$ : ● —  $n = 4$ , ■ —  $n = 6$



Как видно, увеличение размера популяции  $k$  с 10 до 100 приводит к значительному повышению точности нахождения глобального минимума, но при этом быстро растет количество вычислений ЦФ. Оптимальным, с точки зрения точности и времени вычислений, является диапазон значений  $k \in [50, 60]$ . Коэффициент ускорения, рассчитанный по (3) для середины оптимального диапазона  $k$ , составил 7.41, 7.7 и 7.83 для ЦФ (4), (5) и (6) соответственно.

На рис. 4 приведены зависимости найденных минимальных значений ЦФ с указанием 95% доверительного интервала ( $a$ - $b$ ) и числа вычислений ЦФ (4), (5), (6) ( $z$ ) от числа детей в каждом поколении  $m$ . При этом остальные исследуемые параметры алгоритма имели следующие значения:  $k = 50$ ,  $p_c = 0.75$ ,  $p_m = 0.1$ .

Из рис. 4 следует, что увеличение числа детей в каждом поколении  $m$  с 2 до 16 приводит к значительному повышению точности нахождения глобального минимума. При этом количество вычислений ЦФ имеет минимум в диапазоне значений  $m \in [6, 8]$  при удовлетворительной точности для всех исследованных ЦФ. Коэффициент ускорения, рассчитанный по (3) для середины оптимального диапазона  $m$ , составил 3.28, 3.39 и 3.45 для ЦФ (4), (5) и (6) соответственно.



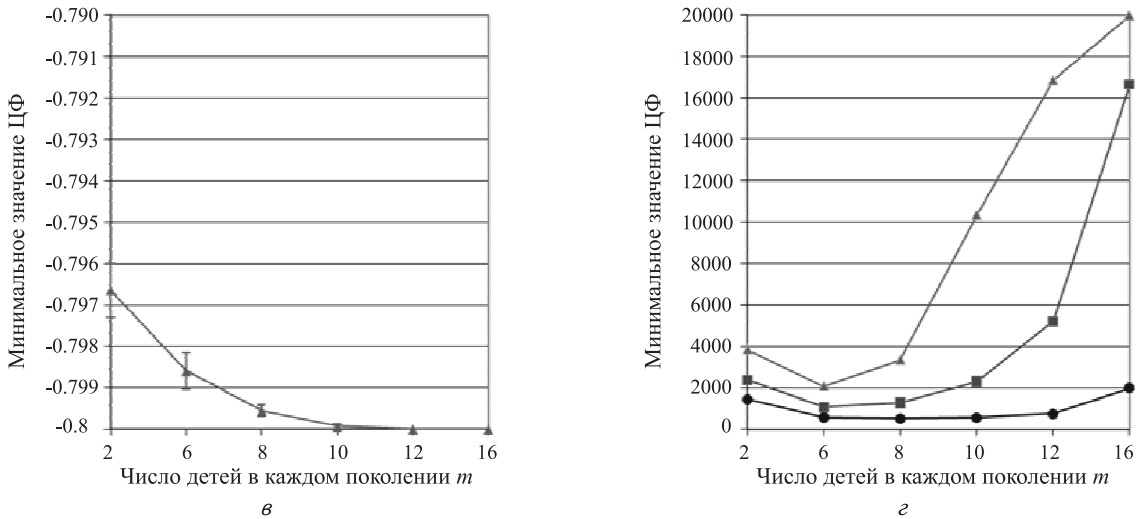


Рис. 4. Окончание. ●  $n = 4$ , ■  $n = 6$ , ▲  $n = 8$

На рис. 5 приведены зависимости найденных минимальных значений ЦФ с указанием 95% доверительного интервала (а-в) и числа вычислений ЦФ (4), (5) и (6) (г) от вероятности применения кроссовера  $p_c$ . При этом остальные исследуемые параметры алгоритма фиксировались ( $k = 50$ ,  $m = 10$ ,  $p_m = 0.05$ ).

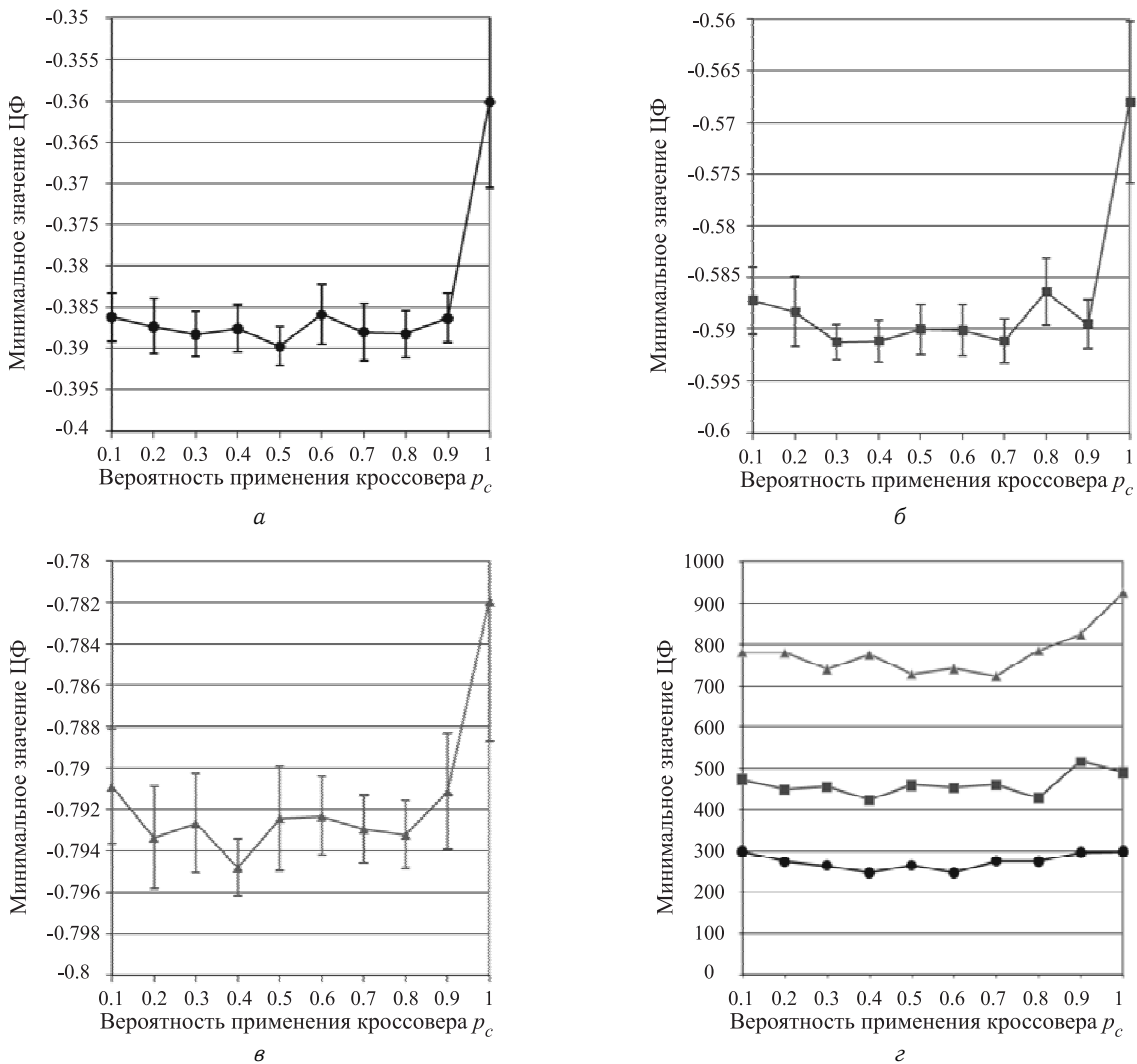


Рис. 5. Зависимость найденного минимального значения ЦФ от вероятности применения кроссовера  $p_c$  и количества параметров оптимизации  $n$ : ●  $n = 4$ , ■  $n = 6$ , ▲  $n = 8$





Из рис. 5 видно, что точность нахождения глобального минимума максимальна в диапазоне значений  $p_c \in [0.3, 0.7]$ . Число вычислений ЦФ также минимально при этих значениях  $p_c$  для всех рассмотренных ЦФ. Коэффициент ускорения, рассчитанный по (3) для середины оптимального диапазона  $p_c$ , составил 4.35, 4.6 и 4.74 для ЦФ (4), (5) и (6) соответственно.

На рис. 6 приведены зависимости найденных минимальных значений ЦФ с указанием 95% доверительного интервала ( $a$ – $в$ ) и числа вычислений ЦФ (4), (5), (6) ( $г$ ) от вероятности мутации  $p_m$ . При этом остальные исследуемые параметры алгоритма имели значения:  $k = 50$ ,  $m = 10$ ,  $p_c = 0.75$ .

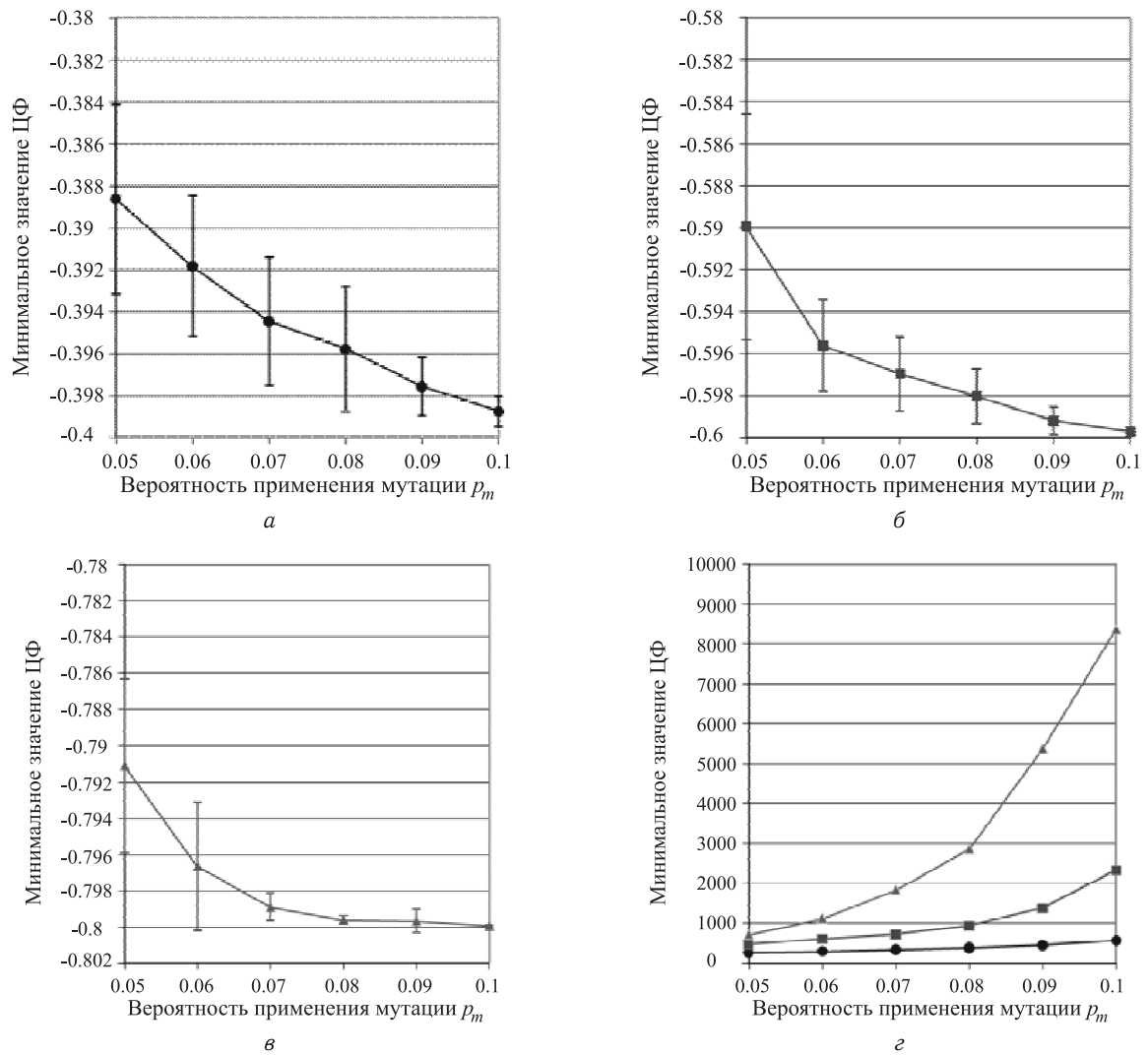


Рис. 6. Зависимость найденного минимального значения ЦФ от вероятности применения мутации  $p_m$  и количества параметров оптимизации  $n$ : ●  $n = 4$ , ■  $n = 6$ , ▲  $n = 8$

Как видно из рис. 6, увеличение вероятности применения оператора мутации  $p_m$  с 0.05 до 0.1 приводит к значительному повышению точности нахождения глобального минимума, но при этом быстро растет количество вычислений. Оптимальным, с точки зрения точности и времени вычислений, является диапазон значений  $p_m \in [0.06, 0.07]$ . Коэффициент ускорения, рассчитанный по (3) для середины оптимального диапазона  $p_m$ , составил 4.46, 4.72 и 4.86 для ЦФ (4), (5) и (6) соответственно.

Таким образом, в результате численного эксперимента были определены оптимальные диапазоны параметров первого варианта распараллеливания ГА, обеспечивающие высокую точность при приемлемом времени поиска глобального минимума ЦФ (4), (5) и (6) (табл. 2).

Оценка эффективности второго варианта параллельного ГА для функций (4), (5) и (6) осуществлялась путем про-

Таблица 2  
Оптимальные диапазоны изменения параметров параллельного ГА

	$k$	$m$	$p_c$	$p_m$
min	50	6	0.3	0.06
max	60	8	0.7	0.07



ведения серии экспериментов при различном числе узлов параллельной ВС  $p$  и фиксированных значениях остальных параметров:  $k = 50, m = 10, p_c = 0.75, p_m = 0.05$ .

Как видно из результатов тестирования (рис. 7), с ростом числа узлов ВС  $p$  с 1 до 30 наблюдается уменьшение доверительного интервала в 7.89, 5.64 и 7.87 раз соответственно для ЦФ (4), (5) и (6), что говорит о значительном повышении надежности нахождения глобального минимума. При этом число вычислений ЦФ, а значит, и время работы параллельной ВС, практически не изменяется и составляет в среднем 257, 450, 740 для ЦФ (4), (5), (6) соответственно.

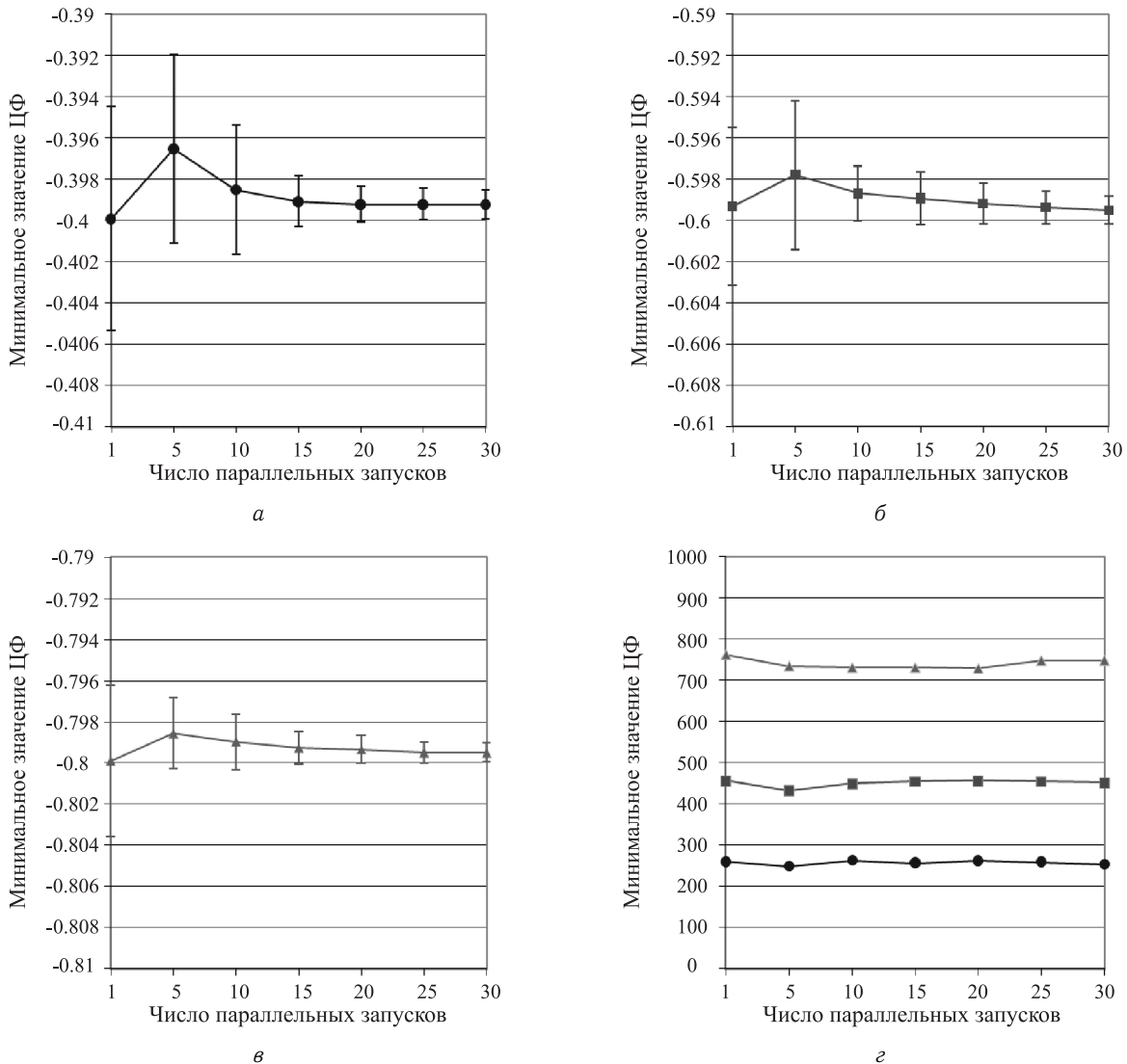


Рис. 7. Зависимость найденного минимального значения ЦФ от количества параллельных запусков параметров оптимизации  $n$ : ● —  $n = 4$ , ■ —  $n = 6$ , ▲ —  $n = 8$

Следовательно, вариант распараллеливания ГА путём множественного запуска на параллельной ВС обеспечивает значительное увеличение вероятности нахождения глобального экстремума за счёт более полного исследования области факторного пространства ЦФ. При этом время выполнения параллельного алгоритма, пропорциональное числу вычислений ЦФ, практически не зависит от числа узлов ВС  $p$  и примерно равно времени выполнения алгоритма на одном узле ВС при одинаковых  $n$ .

## ВЫВОДЫ

В работе предложены два варианта распараллеливания генетических алгоритмов оптимизации для случаев сложных и простых ЦФ.

Первый вариант распараллеливания ГА основан на одновременном вычислении на нескольких узлах ВС изменяемых точек в факторном пространстве. В результате численного эксперимента были



определены оптимальные диапазоны параметров первого варианта распараллеливания ГА, обеспечивающие небольшое время поиска глобального минимума ЦФ и высокую надёжность его нахождения при выполнении на параллельной ВС.

Тестирование второго варианта распараллеливания ГА, основанного на множественном запуске на параллельной ВС, показало значительное увеличение вероятности нахождения глобального экстремума за счёт более полного исследования области факторного пространства ЦФ. Время выполнения параллельного алгоритма при этом практически не зависело от числа узлов ВС и примерно равнялось времени выполнения алгоритма на одном узле ВС.

Таким образом, предложенные варианты распараллеливания генетического алгоритма оптимизации могут эффективно применяться для решения задач поиска как глобального минимума многоэкстремальной ЦФ при наличии ограничений на варьируемые параметры (первый вариант), так и его области (второй вариант).

*Работа выполнена при финансовой поддержке программы Министерства образования и науки РФ «Подготовка и переподготовка профильных специалистов на базе центров образования и разработок в сфере информационных технологий в Южном и Северо-Кавказском федеральных округах» (государственный контракт № 07.Р20.11.0029).*

### Библиографический список

1. Орлянская И. В. Современные подходы к построению методов глобальной оптимизации // Электронный журнал «Исследовано в России». С. 2097–2108. URL : <http://zhurnal.ape.relarn.ru/articles/2002/189.pdf> (дата обращения : 2.12.2011). [Orlyanskaya I. V. Modern approaches to global optimization methods building // Online magazine «Issledovano v Rossii». P. 2097–2108. URL : <http://zhurnal.ape.relarn.ru/articles/2002/189.pdf> (last checked : 2.12.2012).]
2. Банди Б. Методы оптимизации. Вводный курс. М. : Радио и связь, 1988. 127 с. [Brian D. Bunday Basic Optimisation Methods. London : Edward Arnold, 1984. 128 p.]
3. Панченко Т. В. Генетические алгоритмы : учеб.-метод. пособие / под ред. Ю. Ю. Тарасевича / Астрахан. ун-т. Астрахань, 2007. 87 с. [Panchenko T. V. Genetic algorithms : the methodical manual / Ed. by Yu. Yu. Tarasevich / Astrakhan. Univer. Astrakhan, 2007. 87 p.]
4. Калиткин Н. Н. Численные методы. М. : Наука, 1978. 512 с. [Kalitkin N. N. Numerical methods. Moscow : Nauka, 1978. 512 p.]
5. Eroftiev A. A., Timofeeva N. E., Savin A. N. Parallel Computing in Application to Global Optimization Problem Solving // Grid and Visualization Systems : MIPRO, 2011 Proc. of the 34th Intern. Convention. Zagreb, Croatia : DENONA, 2011. P. 185–190.

УДК 501.1

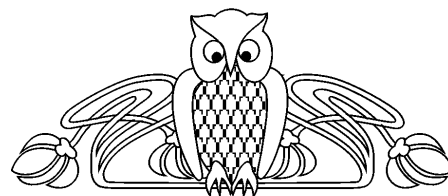
## ГЛАВНЫЕ ИДЕАЛЫ В ПОЛУРЕШЕТКЕ КОНГРУЭНЦИЙ ЦЕПИ

Е. О. Фомина

Саратовский государственный университет  
E-mail: janekao@mail.ru

Показано, что главные идеалы, порождаемые однотипными конгруэнциями цепи, изоморфны как решетки. Подсчитано количество элементов, атомов и коатомов в главном идеале порождаемой данной конгруэнцией цепи.

**Ключевые слова:** цепь, конгруэнция, полурешетка, главный идеал, атом, коатом.



### Principal Ideals in the Congruence Semilattice of a Path

E. O. Fomina

It is proven that principal ideals generated by congruences of a path having the same type are isomorphic lattices. The number of elements, atoms and coatoms is found for the principal ideal generated by a given congruence of a path.

**Key words:** path, congruence, semilattice, principal ideal, lattice, atom, coatom.

Ориентированным графом (далее орграфом) называется пара  $G = (V, \alpha)$ , где  $V$  — конечное непустое множество, а  $\alpha$  — отношение на  $V$ . Множество  $V$  называется множеством вершин, отношение  $\alpha$  — отношение смежности, а пары, входящие в  $\alpha$ , — дуги орграфа  $G$ . Если  $(u, v) \in \alpha$ , то говорят, что вершина  $v$  смежна с вершиной  $u$ . Основные понятия приводятся в соответствии с [1].

Пусть  $\varepsilon$  — некоторое отношение эквивалентности на множестве вершин  $V$ . Факторграфом ор-