

НОВЫЙ АДАПТИВНЫЙ МЕТОД МУЛЬТИМЕТЕТИЧЕСКОЙ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ ДЛЯ СЛАБОСВЯЗАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

М.К. Сахаров

msakharov@bmstu.ru

МГТУ им. Н.Э. Баумана, Москва, Российская Федерация

Аннотация

Предложен новый параллельный адаптивный мультимететический популяционный метод для эффективного решения задач глобальной оптимизации на слабосвязанных вычислительных системах. Рассмотрены существующие подходы к синтезу адаптивных популяционных алгоритмов и методы их распараллеливания; выделены особенности класса слабосвязанных параллельных вычислительных систем, которые необходимо учитывать при разработке алгоритмов для данных систем. Предложенный алгоритм основан на двухуровневой схеме адаптации. Верхний уровень является статическим и позволяет провести настройку свободных параметров базового алгоритма до начала оптимизации на основе информации о целевой функции, полученной методами ландшафтного анализа. Нижний уровень адаптации динамический и реализован за счет гибридизации базового алгоритма с алгоритмами локальной оптимизации (мемами). Предложена новая статическая схема балансировки загрузки для отображения алгоритма на параллельные вычислительные узлы слабосвязанных вычислительных систем. Предложенная схема балансировки учитывает как особенности алгоритма, так и архитектуру используемой вычислительной системы, что позволяет повысить эффективность решения задачи оптимизации по сравнению с традиционными схемами балансировки. Проведено подробное исследование эффективности предложенного алгоритма и его программной реализации с использованием многомерных тестовых функций различных классов

Ключевые слова

Мультимететический алгоритм, ландшафтный анализ, алгоритм эволюции разума, глобальная оптимизация, параллельный алгоритм, грид-вычисления

Поступила 14.02.2019

© Автор(ы), 2019

Введение. Задачи глобальной оптимизации возникают во многих областях инженерной деятельности: в автомобилестроении это может быть оптимизация профиля кузова; в авиастроении — оптимизация профиля лопатки турбины двигателя; в металлургии — оптимизация режимов литья и добавления примесей и др. [1]. Целевые функции в таких задачах зачастую имеют высокую размерность и вычислительную сложность, а также нетривиальный ландшафт. Известно значительное число различных алгоритмов решения таких задач. На верхнем уровне иерархии алгоритмы глобальной оптимизации подразделяют на классические и неклассические (популяционные) [1].

Популяционные алгоритмы в сравнении с классическими имеют неоспоримые преимущества при решении мультимодальных задач высокой размерности: универсальность применения, простота реализации и масштабируемости и др. Экспериментальные исследования показывают, что чем больше априорной информации о задаче заложено в алгоритм, тем выше может быть его эффективность [2]. В настоящее время активно развиваются методы, позволяющие извлекать информацию об особенностях целевой функции и/или подбирать оптимальные значения свободных параметров популяционных алгоритмов или наиболее подходящие методы оптимизации для уточнения промежуточных или итоговых решений. К таким методам относятся меметические алгоритмы [3] и методы ландшафтного анализа [4]

В широком смысле меметические алгоритмы (МА) можно рассматривать как гибридизацию одного из популяционных алгоритмов глобального поиска и одного или нескольких классических или популяционных алгоритмов локальной оптимизации. Как у всех популяционных алгоритмов, для эффективной работы МА необходимо проводить настройку их свободных параметров, например управлять процедурой выбора мемов, отвечающих специфике конкретной задачи. Согласно результатам исследований [5], выбор мемов оказывает большое влияние на производительность МА, вследствие чего были предложены так называемые мультимеметические адаптивные алгоритмы [6]. Отличительной чертой данного класса методов является использование нескольких мемов в процессе поиска оптимума. Решение об использовании того или иного мема для конкретного индивида в популяции принимается динамически. Категории стратегий выбора мемов (гиперэвристики) — случайная; жадная; с функцией выбора — предложены в работах [7, 8].

Кроме меметических алгоритмов, еще один подход к повышению эффективности решения задач оптимизации — ландшафтный анализ (ЛА) це-

левой функции. Вместо динамической адаптации алгоритма предлагается за счет небольшого числа дополнительных испытаний (1...10 % от общего числа испытаний) извлечь из неизвестной целевой функции информацию о ее ландшафте и топологии [9]. Другими словами, методы ЛА позволяют обнаружить области, где целевая функция имеет высокую мултимодальность, или области, где значения целевой функции существенно не изменяются. Классический метод ЛА [4] предлагает в общей сложности 50 числовых признаков ландшафта функции $\Phi(X)$, сгруппированных в пять так называемых свойств: 1) выпуклость; 2) степень кривизны; 3) ярусность; 4) мултимодальность; 5) метамодельные свойства. Известны и другие методы ЛА, например метод клеточного отображения [9], метод информационного содержания [10] и пр.

При решении вычислительно сложных задач время вычисления одного значения целевой функции может достигать нескольких часов [11]. К таким задачам относятся, например, задачи оптимального управления динамической системой, где высокая сложность целевой функции обусловлена необходимостью решать систему дифференциальных уравнений в рамках каждого испытания, а высокая размерность пространства поиска — необходимостью высококачественной аппроксимации функции управления. Этот факт заставляет не только использовать различные параллельные вычислительные системы, но и проектировать алгоритмы, ориентируясь на конкретный тип архитектуры параллельной системы.

Грид-системы из персональных компьютеров являются отдельным направлением в области слабосвязанных вычислительных систем (ССВС) [12]. Эти системы представляют собой объединение нескольких компьютеров для решения единой вычислительно сложной задачи, разбитой на подзадачи. Основное преимущество грид-систем в том, что они могут состоять из разнородных компьютеров, находящихся далеко друг от друга. Задачу объединения этих разнородных компьютеров выполняет промежуточное программное обеспечение, которое виртуально связывает все компьютеры через интернет или локальную сеть. Грид-системы, состоящие из персональных компьютеров, получили название *Desktop Grid*. Одно из перспективных направлений создания *Desktop Grid* — реализация технологии распределенных вычислений в рамках локальных коммуникационных сетей предприятий и организаций *Enterprise Desktop Grid*. Такой подход позволяет небольшим компаниям без существенных инвестиций в оборудование и инфраструктуру получить необходимые вычислительные мощности.

В работе предложен новый параллельный адаптивный алгоритм для решения задач глобальной оптимизации на ССВС вида *Enterprise Desktop*

Grid. Представлены результаты исследования эффективности предложенного алгоритма и его программной реализации с использованием многомерных тестовых функций различных классов на ССВС из девяти персональных компьютеров.

Постановка задачи и базовый алгоритм *SMEC*. Рассмотрим задачу глобальной безусловной минимизации

$$\min_{X \in R^{|X|}} \Phi(X) = \Phi(X^*) = \Phi^*, \quad (1)$$

где $X = (x_1, x_2, \dots, x_{|X|})$ — $|X|$ -мерный вектор варьируемых параметров; $R^{|X|}$ — $|X|$ -мерное арифметическое пространство; $\Phi(X)$ — скалярная целевая функция; $\Phi(X^*) = \Phi^*$ — ее искомое минимальное значение.

Задана область

$$D_0 = \{ X \mid x^{\min} \leq x_i \leq x^{\max}, i \in [1 : |X|] \} \quad (2)$$

для генерации начальной популяции решений. Выбор границ области D_0 осуществляется пользователем.

Для решения задачи (1), (2) в качестве базового алгоритма оптимизации выбран простой алгоритм эволюции разума — *SMEC* [13]. Алгоритм имеет высокий потенциал развития для организации параллельных вычислений особенно в ССВС, так как в отличие от большинства других методов не подразумевает больших объемов обмена информацией между эволюционирующими группами. Алгоритм *SMEC* построен на основе операций инициализации групп, локальных состязаний и диссимилиации.

Популяция алгоритма состоит из лидирующих групп $S^b = (S_1^b, S_2^b, \dots, S_{|S^b|}^b)$

и отстающих групп $S^w = (S_1^w, S_2^w, \dots, S_{|S^w|}^w)$, число индивидов в которых полагаем одинаковым и равным $|S|$. Соотношение чисел лидирующих $|S^b|$ и отстающих $|S^w|$ групп в алгоритме определяется параметром η (свободный параметр алгоритма).

Операция инициализации групп создает группы S^b , S^w и размещает их в области поиска. Рассмотрим схему операции на примере группы S_i .

1. Генерируем случайный вектор $X_{i,1}$, компоненты которого равномерно распределены в области поиска D . Отождествляем этот вектор с индивидом $s_{i,1}$ группы S_i .

2. Определяем начальные координаты остальных индивидов данной группы по формуле

$$X_{i,j} = X_{i,1} + N_{|X|}(0, \sigma), j \in [2 : |S|], \quad (3)$$

где $N_{|X|}(0, \sigma)$ — $(|X| \times 1)$ -вектор независимых вещественных случайных чисел, распределенных по нормальному закону, σ — свободный параметр алгоритма.

Операция локальных состязаний реализует локальный поиск минимума целевой функции в каждой группе S^b , S^w . Схема этой операции для группы S_i имеет следующий вид.

1. Берем информацию о текущем индивиде-победителе группы S_i . Пусть это будет индивид s_{i,j_b} , $j_b \in [1 : |S|]$.

2. Определяем новые координаты всех оставшихся индивидов $s'_{i,j}$, $j \in [1 : |S|]$, $j \neq j_b$, данной группы по правилу вида (3).

3. Вычисляем значения целевой функции для всех индивидов данной группы $\Phi'_{i,j} = \Phi(X'_{i,j})$, $j \in [1 : |S|]$.

4. Определяем нового победителя группы s'_{i,k_b} , $k_b \in [1 : |S|]$, как индивида данной группы, который имеет минимальное текущее значение целевой функции $\Phi(X)$.

5. Записываем информацию о новом победителе группы s'_{i,k_b} .

Операция диссимилиации управляет глобальным поиском. Схема операции приведена ниже.

1. Считываем текущие значения целевой функции $\Phi(X)$ победителей групп S^b , S^w .

2. Выполняем сравнение указанных значений между собой. Если значение функции $\Phi(X)$ для некоторой лидирующей группы S_i^b меньше значения этой функции для одной из отстающих групп S_j^w , то последняя группа занимает место группы S_i^b в наборе лидирующих групп S^b , а группа S_i^b — место группы S_j^w среди отстающих групп S^w . Если группа S_j^w на протяжении ω итераций является отстающей, то удаляем группу S_j^w из популяции. Здесь ω — свободный параметр алгоритма.

3. С использованием операции инициализации взамен каждой удаленной группы инициализируем новую группу.

Операции локальных состязаний и диссимилиации итерационно повторяем до тех пор, пока имеет место уменьшение минимального значения целевой функции лидирующих групп.

К недостаткам простого алгоритма эволюции разума *SMEC* следует отнести: отсутствие механизмов использования априорной информации

о решаемой задаче в процессе решения; отсутствие рекомендаций по выбору значений свободных параметров алгоритма для различных классов решаемых задач; отсутствие механизмов динамической адаптации алгоритма к особенностям решаемых задач в процессе решения.

Предлагаемые модификации алгоритма *SMEC*. Алгоритм *MЗMEC-P*. Предложена двухуровневая адаптивная мультимететическая модификация простого алгоритма эволюции разума *SMEC*. Адаптация верхнего уровня реализована за счет модификации исходного алгоритма *SMEC* в мультипопуляционный, а также за счет модификации операции инициализации с использованием встроенной процедуры ЛА. Данный уровень адаптации статический и позволяет провести настройку свободных параметров алгоритма до начала оптимизации на основе информации о целевой функции, полученной методами ЛА. Нижний уровень адаптации динамический: в предложенной мультимететической гибридации алгоритма *SMEC* с несколькими методами локального поиска выбор наиболее подходящих мемов осуществляется динамически с использованием функции выбора.

*Мультипопуляционная модификация алгоритма *SMEC*.* Значения свободных параметров алгоритма *SMEC* существенно влияют на эффективность его работы [14]. В практически значимых задачах ландшафт целевой функции может быть разнородным, так что для различных подобластей поиска оптимальными являются различные значения свободных параметров.

Вместо единой популяции рассматриваем мультипопуляцию, состоящую из множества субпопуляций $K = (K_1, K_2, \dots, K_{|K|})$. Процесс эволюции каждой субпопуляции управляется индивидуальными значениями свободных параметров алгоритма (среднее квадратичное отклонение σ_l , частота удаления отстающих групп ω_l , соотношение чисел η_l отстающих $|S^w|$ и лидирующих $|S^b|$ групп в популяции; $l \in [1 : |K|]$). Искомое приближение к решению Φ^* определяем на основе решений, найденных всеми субпопуляциями, как минимальное значение этих решений:

$$\hat{\Phi}^* = \min_l \hat{\Phi}_l^*, l \in [1 : |K|]. \quad (4)$$

Модификация операции инициализации с помощью процедуры ландшафтного анализа. Предложен метод ЛА, встроенный в операцию инициализации мультипопуляционного алгоритма *SMEC*. Модифицированную операцию инициализации алгоритма с учетом процедуры ЛА можно представить следующей последовательностью шагов.

1. Генерируем N псевдослучайных векторов размерности $|X|$ в области D_0 .
2. Для каждого вектора $X_r, r \in [1:N]$, вычисляем соответствующее значение целевой функции Φ_r . Сортируем векторы $X_r, r \in [1:N]$ по возрастанию значения Φ_r .
3. Равномерно разделяем множество векторов (X_1, X_2, \dots, X_N) на $|K|$ подмножеств.
4. Полученные группы отождествляем с субпопуляциями K_l , а образующие группы векторы отождествляем с главными индивидами каждой субпопуляции.
5. Для каждой субпопуляции $K_l, l \in [1:|K|]$ вычисляем значение ее диаметра d_l — максимальное евклидово расстояние между двумя любыми ее индивидами (рис. 1).
6. Линейно аппроксимируем зависимость диаметра d_l от номера группы l , используя метод наименьших квадратов: $d(l) = kl + b$.
7. Вычисляем длину диагонали гиперкуба D_0 .
8. На основе полученных результатов определяем значения свободных параметров алгоритма *SMEC* для каждой субпопуляции K_l .

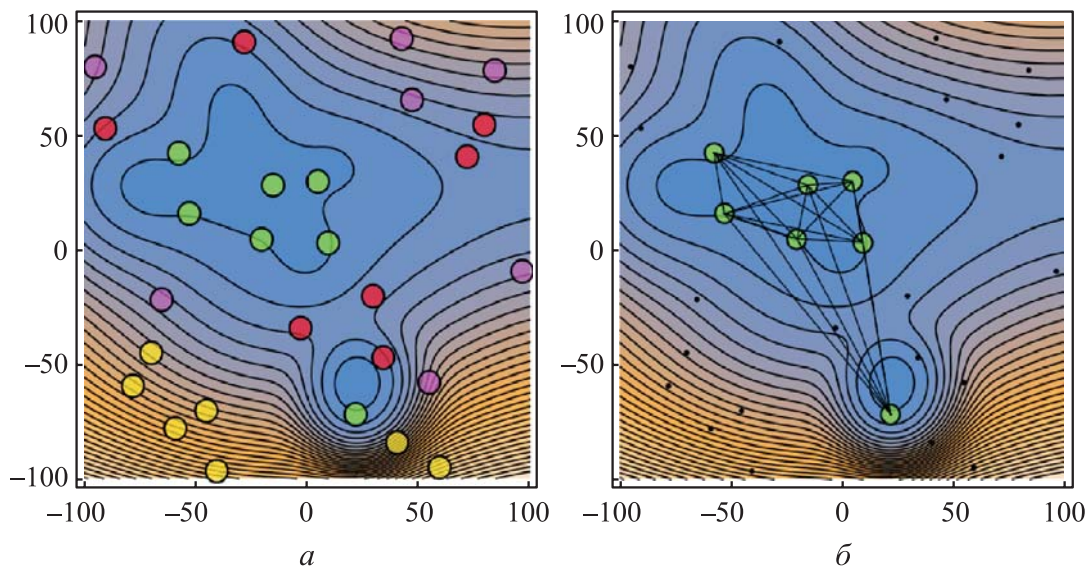


Рис. 1. Определение диаметра субпопуляции для тестовой составной функции № 1 из набора *CEC'14*:

a — распределение индивидов четырех субпопуляций; b — определение диаметра первой субпопуляции

На основе проведенных исследований предложены правила определения значений свободных параметров алгоритма *SMEC* (табл. 1). Возможны три варианта зависимости диаметра d от номера группы l : 1) $d(l)$ является

возрастающей функцией ($k > \operatorname{tg} \theta$); 2) $d(l)$ убывает с ростом l ($k < -\operatorname{tg} \theta$); 3) $d(l)$ не возрастает и не убывает при увеличении l в некотором заданном диапазоне ($-\operatorname{tg} \theta < k < \operatorname{tg} \theta$). Рекомендованные значения коэффициента $\theta = 3^\circ \dots 7^\circ$. Величина ζ позволяет оценить плотность расположения векторов X с минимальными значениями целевой функции Φ . Рекомендованные значения коэффициента $\zeta = 2-3$.

Каждая ситуация (см. табл. 1) соответствует определенному расположению субпопуляций и образующих их индивидов в пространстве поиска, что позволяет качественно оценить топологию исследуемой целевой функции.

Мультимеметическая модификация алгоритма SМЕС. Предложена мультимеметическая модификация мультипопуляционного алгоритма SМЕС. Рассматривается рой доступных мемов $M = (m_j, j \in [1 : |M|])$, набор которых определяется пользователем. Для определения наиболее подходящего мема используем гиперэвристику на основе функции выбора ϕ . Предложена функция выбора ϕ , построенная на основе трех метрик, отражающих различные аспекты качества мемов, и ориентированная на интерактивное взаимодействие с пользователем:

$$\phi(m_j) = \lambda_1 \phi_1(m_j) + \lambda_2 \phi_2(m_j) + \lambda_3 \phi_3(m_j), \quad j \in [1 : |M|].$$

Функция ϕ представляет собой аддитивную свертку функций $\phi_1(m_j)$, $\phi_2(m_j)$, $\phi_3(m_j)$ с весами λ_1 , λ_2 , λ_3 . Компонента $\phi_1(m_j)$ формализует текущую величину улучшения целевой функции, достигнутую мемом m_j . Компонента $\phi_2(m_j)$ позволяет учесть вероятность случайного выбора того или иного мема независимо от его успешности. Третья компонента $\phi_3(m_j)$ предназначена для включения экспертных знаний в пул мемов и отражает вероятность успешности того или иного мема по мнению пользователя. Полагаем, что лучшему мему соответствует большее значение функции $\phi(m_j)$.

Схему модифицированной операции локальных состязаний с учетом использования мемов для одной из субпопуляций K_l описывает следующая последовательность шагов.

1. Берем информацию о текущем индивиде-победителе группы $S_{l,i}$. Пусть это будет индивид s_{l,i,j_b} , $j_b \in [1 : |S|]$.

2. Если текущий номер итерации кратен f_{l_s} (свободный параметр алгоритма; рекомендуемое значение $f_{l_s} = 10$), то переходим к п. 3, иначе переходим к п. 5.

Выбор значений свободных параметров на основе результатов ЛА

$d_D / d(1)$	$k > \text{tg } \theta$	$-\text{tg } \theta < k < \text{tg } \theta$	$k < -\text{tg } \theta$
$\frac{d_D}{d(1)} > \zeta$	<p><i>Ситуация I.</i> Субпопуляции вложены одна в другую. Правила определения свободных параметров: $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 0, 25$; $b_l = 0, 75$</p>	<p><i>Ситуация III.</i> Субпопуляции имеют сходные диаметры и ориентированы произвольно. Правила определения свободных параметров: $\sigma_l = \text{rand}(a_l, b_l)$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 0, 1$; $b_l = 0, 9$</p>	<p><i>Ситуация V.</i> Субпопуляции вложены одна в другую. Правила определения свободных параметров: $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 1$; $b_l = -0, 75$</p>
$\frac{d_D}{d(1)} \leq \zeta$	<p><i>Ситуация II.</i> Субпопуляции вложены одна в другую. Правила определения свободных параметров: $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 0, 25$; $b_l = 0, 75$</p>	<p><i>Ситуация IV.</i> Субпопуляции имеют сходные диаметры и ориентированы произвольно. Правила определения свободных параметров: $\sigma_l = \text{rand}(a_l, b_l)$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 0, 1$; $b_l = 0, 9$</p>	<p><i>Ситуация VI.</i> Субпопуляции вложены одна в другую. Правила определения свободных параметров: $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20$; $\eta_l = 50$; $a_l = 1$; $b_l = -0, 75$</p>

3. Запускаем каждый мем m_j из набора доступных мемов M из положения главного индивида s_{l,i,j_b} . Каждый запуск мема ограничен числом итераций локального поиска λ_{ls} (свободный параметр алгоритма; рекомендуемое значение $\lambda_{ls} = 10$).

4. Для каждого мема m_j определяем значение функции выбора $\phi(m_j)$ и выбираем мем-победитель m_j^* .

5. Определяем новые координаты всех оставшихся индивидов $s'_{l,i,j}$, $j \in [1:|S|]$, $j \neq j_b$, данной группы по правилу вида (3).

6. Если текущий номер итерации кратен f_{ls} , то переходим к п. 7, иначе переходим к п. 9.

7. Запускаем выбранный мем m_j^* из положений всех индивидов группы $s'_{l,i,j}$ и определяем координаты уточненных решений. Каждый запуск мема ограничен числом итераций локального поиска λ_{ls} .

8. Вычисляем значение целевой функции Φ для всех индивидов субпопуляции K_l .

9. Записываем информацию о новых победителях всех групп субпопуляции $S_{l,j}$.

Схема распараллеливания. Схема распараллеливания алгоритма МЗМЕС-Р основана на независимой эволюции субпопуляций $K = (K_1, K_2, \dots, K_{|K|})$ на различных вычислительных узлах, в то время как операции инициализации и ЛА, настройки параметров субпопуляций и формирования итогового решения осуществляются последовательно на главном узле.

При решении задачи на параллельных вычислительных системах и на ССВС, в частности, одной из основных проблем является задача оптимального отображения [15] — оптимальное распределение групп подзадач по процессорам. Чаще всего для приближенного решения задачи оптимального отображения используется метод балансировки загрузки вычислительной системы. Основная идея метода состоит в распределении вычислений по процессорам так, чтобы их суммарная вычислительная и коммуникационная загрузки были примерно одинаковы.

Рассмотрим гетерогенную параллельную ССВС, состоящую из одного главного вычислительного узла P_0 и множества $|P|$ доступных вычислительных узлов $P = \{P_1, \dots, P_{|P|}\}$. Для обеспечения согласованности алгоритма с архитектурой ССВС полагаем, что число субпопуляций $|K|$ равно числу доступных вычислительных узлов $|P|$.

Для алгоритма *МЗМЕС-Р* предложена схема статической балансировки загрузки на основе результатов операции ЛА, которая формирует субпопуляции $K = (K_1, K_2, \dots, K_{|K|})$ приблизительно одинаковой средней вычислительной сложности для распределения по доступным вычислительным узлам.

Статическую балансировку загрузки осуществляем по приведенной ниже схеме.

1. Для каждой субпопуляции $K_l, l \in [1:|K|]$, проводим анализ оценок времени вычисления $t_r, r \in [1:\gamma]$, целевой функции $\Phi(X_r)$ на наличие выбросов.

2. Все найденные выбросы t_r^* исключаем из субпопуляций и добавляем в субпопуляцию $K_{|K|+1}$.

3. Вычислительные узлы P_k сортируем по производительности, и первую субпопуляцию K_1 отправляем на самый мощный узел P_1 .

4. Выполняем перераспределение индивидов между смежными субпопуляциями, начиная с субпопуляции K_2 , так, чтобы среднее время расчета целевой функции главных индивидов было приблизительно одинаковым, и распределяем сбалансированные субпопуляции $K_l, l \in [2:|K|]$, по оставшимся вычислительным узлам.

5. На каждом узле запускаем модифицированный алгоритм *SMES* со значениями свободных параметров (см. табл. 1).

Использование предложенного метода балансировки загрузки позволяет учесть не только структуру алгоритма *МЗМЕС-Р*, но и особенности архитектуры ССВС. После окончания расчетов на каждом узле ССВС найденное оптимальное решение возвращается на главный узел.

Программная реализация. Программная реализация алгоритма *МЗМЕС-Р* выполнена в среде *Wolfram Mathematica*, которая содержит инструменты для управления распределенной вычислительной сетью, включая инициализацию вычислительных узлов, распределение задач и управление распределенной памятью.

Предложенная программная реализация алгоритмов представляет собой модули языка *Wolfram Language* с заданным набором входных параметров. На выходе каждый модуль возвращает следующую информацию: наилучший найденный вектор варьируемых параметров \check{X}^* ; соответствующее значение целевой функции $\Phi(\check{X}^*) = \Phi^*$; число итераций λ работы алгоритма. При необходимости могут быть отображены результаты всех или части промежуточных итераций работы алгоритмов.

В работе использовано $M = 4$ метода нулевого порядка: метод Нелдера — Мида [16], метод Хука — Дживса [17], метод случайного поиска с постоянным радиусом и случайными направлениями [18] и метод Монте-Карло [18]. Каждый метод был реализован в виде самостоятельного модуля в среде *Wolfram Mathematica*.

Экспериментальное исследование эффективности. Для исследования эффективности алгоритма *МЗМЕС-Р* и его программной реализации использованы одно- и многоэкстремальные многомерные тестовые функции [19]. Вычислительные эксперименты проведены с использованием метода мультистарта (100 запусков) для размерностей $|X| = \{8, 16, 32\}$. Значения свободных параметров алгоритма: число групп $\gamma = 50$; число индивидов в каждой группе $|S| = 50$; число итераций стагнации вычислительного процесса $\lambda_{stop} = 100$. Требуемая точность локализации известного глобального минимума $\varepsilon_p = 10^{-2}$.

В качестве критериев эффективности работы алгоритма использована оценка p вероятности локализации глобального минимума, наименьшее найденное значение целевой функции Φ^* и среднее найденное значение целевой функции $\bar{\Phi}$. Испытания проводились на ССВС, состоящей из девяти одинаковых персональных компьютеров (один главный узел и $N = 8$ подчиненных узлов) со следующими характеристиками: четырехъядерный процессор *Intel Core i5-6600* и 8 ГБ оперативной памяти. Использована система *Wolfram Mathematica 11* под управлением операционной системы *Win 10 Pro X64*.

Результаты вычислительных экспериментов представлены на рис. 2 и в табл. 2, 3. Алгоритм *МЗМЕС-Р* и его программная реализация успешно справились с локализацией глобального минимума всех тестовых функций размерностью $|X| = 8$ (оценка вероятности локализации глобального минимума $p = 100\%$). Для более высоких размерностей ($|X| = \{16, 32\}$) оценка вероятности локализации p незначительно снижается. В табл. 2 и 3 приведены среднее ($\bar{\Phi}$) и наименьшее найденное (Φ^*) значения для всех тестовых функций.

Для алгоритма *МЗМЕС-Р* и его программной реализации проведено экспериментальное исследование ускорения S вычислений [15]. Большинство использованных тестовых функций не является достаточно вычислительно сложными для эффективного распараллеливания, поэтому для искусственного повышения вычислительной сложности каждое испытание выполнено 1000 раз.

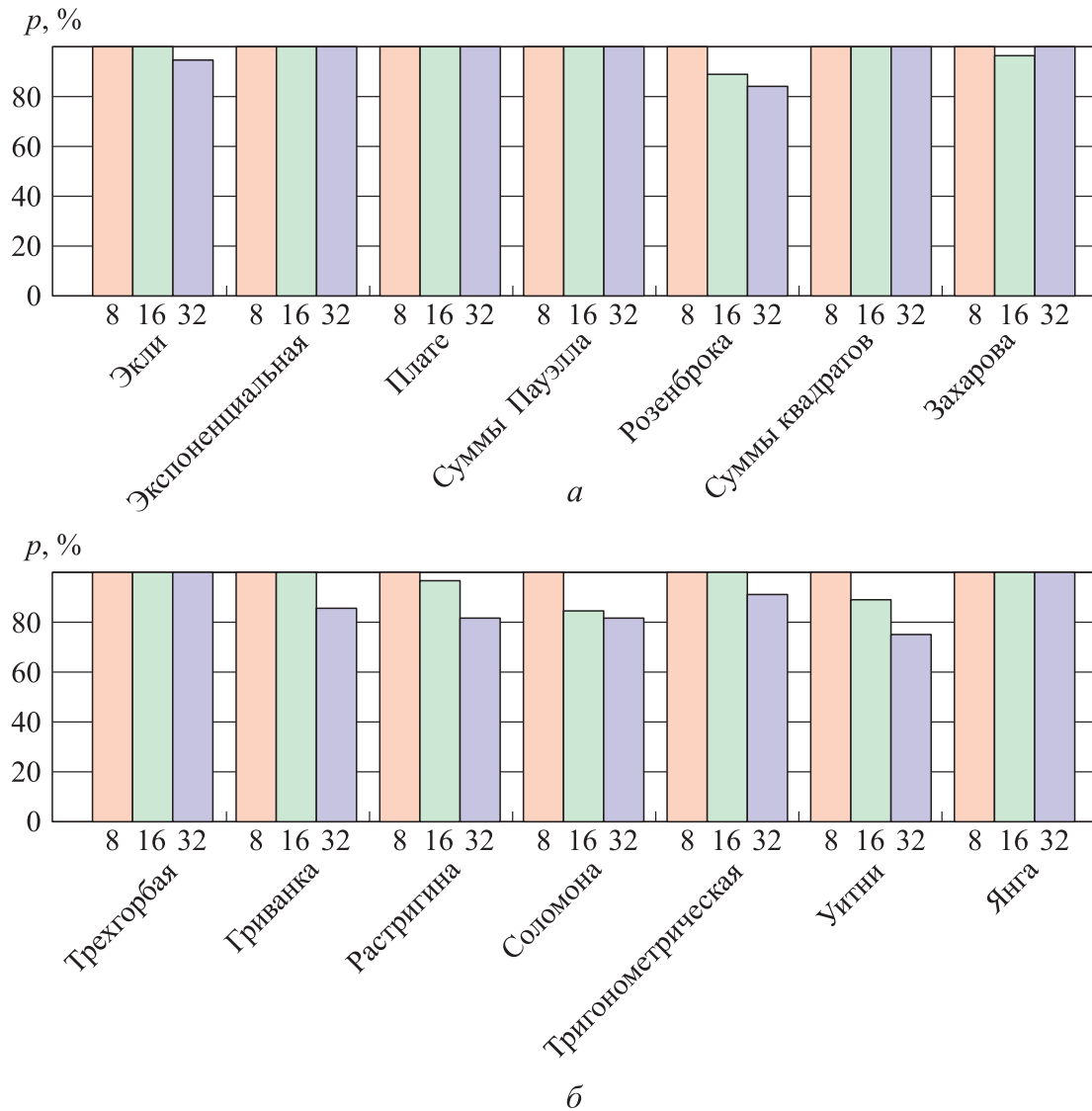


Рис. 2. Оценка вероятности p локализации глобального минимума по результатам мултистарта для алгоритма МЗМЕС-Р для одно- (а) и многоэкстремальных (б) функций

Таблица 2

Одноэкстремальные многомерные тестовые функции

Функция	$\Phi^* / \bar{\Phi}$ при размерности $ X $		
	8	16	32
Экли (Ackley)	2,62E-4 / 7,44E-4	6,74E-3 / 1,56E-2	6,74E-3 / 1,56E-2
Экспоненциальная (Exponential)	-1,00E+0 / -1,00E+0	-1,00E+0 / -1,00E+0	-1,00E+0 / -1,00E+0
Плате (Plateau)	3,00E+1 / 3,00E+1	3,00E+1 / 3,00E+1	3,00E+1 / 3,00E+1

Окончание табл. 2

Функция	$\Phi^* / \bar{\Phi}$ при размерности $ X $		
	8	16	32
Суммы Пауэлла (Powell Sum)	1,03E-11 / 4,94E-9	8,31E-8 / 2,78E-3	8,31E-8 / 2,78E-3
Розенброка (Rosenbrock)	1,36E-4 / 1,59E-2	4,91E-2 / 1,14E-1	4,91E-2 / 1,14E-1
Суммы квадратов (Sum Squares)	5,02E-7 / 1,32E-6	4,07E-6 / 6,72E-3	4,07E-6 / 6,72E-3
Захарова (Zakharov)	2,03E-6 / 1,19E-5	5,67E-4 / 1,39E-3	5,67E-4 / 1,39E-3

Таблица 3

Многоэкстремальные многомерные тестовые функции

Функция	$\Phi^* / \bar{\Phi}$ при размерности $ X $		
	8	16	32
Трехгорбая (Alpine)	6,65E-5 / 1,23E-4	3,83E-3 / 9,04E-2	3,83E-3 / 9,04E-2
Гриванка (Griewank)	1,46E-8 / 4,62E-8	2,81E-6 / 3,78E-5	2,81E-6 / 3,78E-5
Растригина (Rastrigin)	2,21E-5 / 5,79E-5	4,27E-4 / 7,96E-3	4,27E-4 / 7,96E-3
Соломона (Salomon)	9,98E-2 / 9,99E-2	2,06E-1 / 7,69E-1	2,06E-1 / 7,69E-1
Уитни (Whitley)	2,57E-9 / 6,35E-3	5,38E-6 / 8,15E-3	5,38E-6 / 8,15E-3
Тригонометрическая (Trigonometric)	1,00E+0 / 1,00E+0	1,00E+0 / 1,00E+0	1,00E+0 / 1,00E+0
Янга (Xin-She Yang)	3,34E-3 / 3,71E-3	1,47E-2 / 2,85E-1	1,47E-2 / 2,85E-1

Сравнительное исследование проводилось для 2, 4 и 8 вычислительных узлов при размерности целевых функций $|X|=32$. Оценки ускорения алгоритма *МЗМЕС-Р* для некоторых тестовых функций представлены на рис. 3. Кривая синего цвета соответствует ускорению в идеальном теоретическом случае, кривая красного цвета — результатам исследования. Относительно невысокое ускорение связано с тем, что существенная часть алгоритма выполняется последовательно (ЛА и балансировка), а также используется синхронное условие окончания расчетов: главный узел ожидает, пока все подчиненные узлы закончат вычисления.

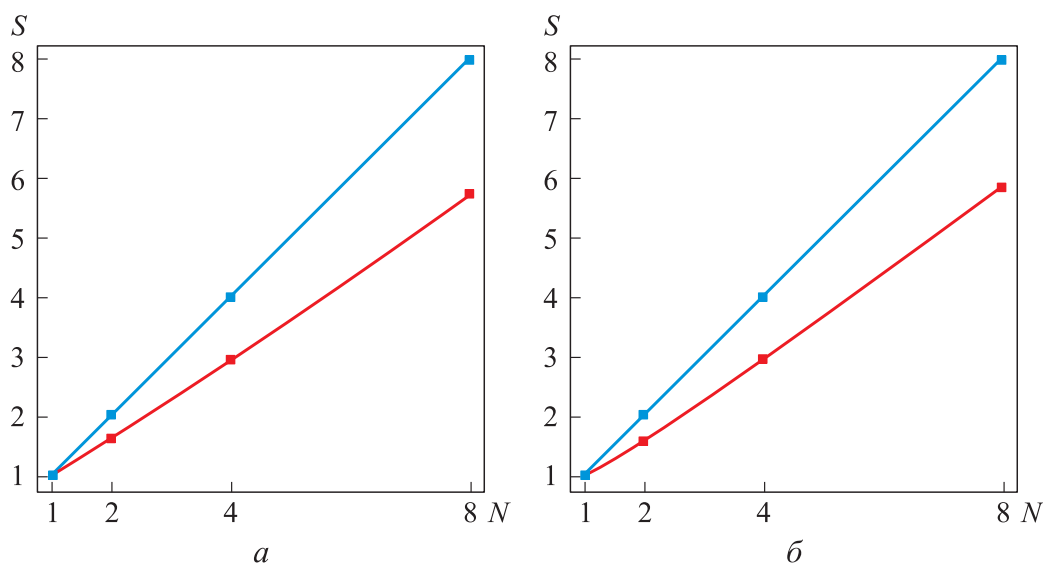


Рис. 3. Ускорение S алгоритма по итогам мултистарта для алгоритма $M3MEC-P$ для тестовых многоэкстремальных функций Захарова (а) и Уитни (б)

Заключение. Предложены новые параллельные адаптивные метод и алгоритм $M3MEC-P$ для решения задач глобальной оптимизации на основе алгоритма эволюции разума с использованием процедур ЛА, настройки его свободных параметров и мултииметической гибридизации, ориентированные на архитектуру ССВС. Разработан параллельный программный комплекс, реализующий предложенные метод и алгоритм $M3MEC-P$ и осуществляющий управление вычислительными узлами ССВС. Комплекс обладает модульной структурой, которая позволяет расширять список используемых мемов, вводить дополнительные вспомогательные операции, например новые методы генерации начальной популяции. Разработанные программные модули предназначены для запуска в среде *Wolfram Mathematica*.

Выполнено экспериментальное исследование эффективности алгоритма $M3MEC-P$ и его программной реализации. Вычислительные эксперименты проведены с использованием одно- и многоэкстремальных тестовых функций различных размерностей и классов. Результаты экспериментов демонстрируют высокую эффективность алгоритма $M3MEC-P$. Разработанный параллельный алгоритм и его программная реализация подтвердили эффективность при решении практически значимой вычислительно сложной задачи глобальной оптимизации: определение оптимального графика импульсной вакцинации при моделировании динамики заболеваемости [20].

Дальнейшее развитие работы направлено на расширение пула доступных мемов и исследование различных гиперэвристик, а также использование асинхронных критериев останова.

ЛИТЕРАТУРА

- [1] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. М., Изд-во МГТУ им. Н.Э. Баумана, 2014.
- [2] Sakharov M., Karpenko A. A new way of decomposing search domain in a global optimization problem. In: Abraham A., Kovalev S., Tarassov V., Snasel V., Vasileva M., Sukhanov A. (eds). *Proceedings of the Second International Scientific Conference “Intelligent Information Technologies for Industry” (IITI’17). IITI 2017. Advances in Intelligent Systems and Computing*, vol 679. Springer, Cham, 2017, pp. 398–407.
DOI: https://doi.org/10.1007/978-3-319-68321-8_41
- [3] Neri F., Cotta C., Moscato P. Handbook of memetic algorithms. Springer, 2012.
DOI: 10.1007/978-3-642-23247-3
- [4] Vassilev V., Fogarty T., Miller J. Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application. In: Ghosh A., Tsutsui S. (eds). *Advances in Evolutionary Computing. Natural Computing Series*. Berlin, Heidelberg, Springer, 2003, pp. 3–44. DOI: https://doi.org/10.1007/978-3-642-18965-4_1
- [5] Krasnogor N. Studies on the theory and design space of memetic algorithms. Ph.D. Thesis, University of the West of England, 2002.
- [6] Moscato P. On evolution, search, optimization, genetic algorithms and martial arts. Towards memetic algorithms. Technical Report Concurrent Computation Program. California Institute of Technology, 1989.
- [7] Ang J.H., Tan K.C., Mamun A.A. An evolutionary memetic algorithm for rule extraction. *Expert Syst. App.*, 2010, vol. 37, iss. 2, pp. 1302–1315.
DOI: <https://doi.org/10.1016/j.eswa.2009.06.028>
- [8] Ong Y.S., Lim M.H., Zhu N., et al. Classification of adaptive memetic algorithms: a comparative study. *IEEE Trans. Syst., Man, Cybern. B Cybern.*, 2006, vol. 36, iss. 1, pp. 141–152. DOI: 10.1109/TSMCB.2005.856143
- [9] Kerschke P., Preuss M., Hernández C., et al. Cell mapping techniques for exploratory landscape analysis. EVOLVE — a bridge between probability, set oriented numerics, and evolutionary computation V. *Advances in Intelligent Systems and Computing*, vol. 288, Cham, Springer, 2014, pp. 115–131. DOI: https://doi.org/10.1007/978-3-319-07494-8_9
- [10] Muñoz M.A., Kirley M., Halgamuge S.K. Exploratory landscape analysis of continuous space optimization problems using information content. *IEEE T. Evolut. Comput.*, 2015, vol. 19, iss. 1, pp. 74–87. DOI: 10.1109/TEVC.2014.2302006
- [11] Карпенко А.П., Сахаров М.К. Мультимемеевая глобальная оптимизация на основе алгоритма эволюции разума. *Информационные технологии*, 2014, № 7, с. 23–30.
- [12] Sakharov M.K., Karpenko A.P. Adaptive load balancing in the modified mind evolutionary computation algorithm. *Supercomputing Frontiers and Innovations*, 2018, vol. 5, no. 4, pp. 5–14. DOI: 10.14529/jsfi180401
- [13] Sun Ch., Sun Y., Wang W. A survey of MEC: 1998–2001. *IEEE SMC*, 2002, vol. 6, pp. 445–453. DOI: 10.1109/ICSMC.2002.1175629

- [14] Sakharov M., Karpenko A. Performance investigation of mind evolutionary computation algorithm and some of its modifications. In: Abraham A., Kovalev S., Tarassov V., Snášel V. (eds). *Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (ITI'16). Advances in Intelligent Systems and Computing*, vol. 450. Cham, Springer, 2016, pp. 475–486.
DOI: https://doi.org/10.1007/978-3-319-33609-1_43
- [15] Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб., БХВ-Петербург, 2002.
- [16] Nelder J.A., Mead R. A simplex method for function minimization. *Comput. J.*, 1965, vol. 7, iss. 4, pp. 308–313. DOI: <https://doi.org/10.1093/comjnl/7.4.308>
- [17] Hooke R., Jeeves T.A. Direct search solution of numerical and statistical problems. *JACM*, 1961, vol. 8, iss. 2, pp. 212–229. DOI: 10.1145/321062.321069
- [18] Карпенко А.П., Белоус В.В. Методы оптимизации (базовый курс). М., Изд-во МГТУ им. Н.Э. Баумана, 2016.
- [19] Momin J., Yang X.S. A literature survey of benchmark functions for global optimization problems. *IJMMNO*, 2013, vol. 4, no. 2, pp. 150–194.
DOI: 10.1504/IJMMNO.2013.055204
- [20] Сахаров М.К. Исследование модели контроля заболеваемости с использованием импульсной вакцинации. *Научные технологии и интеллектуальные системы*. М., МГТУ им. Н.Э. Баумана, 2018, с. 116–120.

Сахаров Максим Константинович — старший преподаватель кафедры «Системы автоматизированного проектирования» МГТУ им. Н.Э. Баумана (Российская Федерация, 105005, Москва, 2-я Бауманская ул., д. 5, стр. 1).

Просьба ссылаться на эту статью следующим образом:

Сахаров М.К. Новый адаптивный метод мультимеметической глобальной оптимизации для слабосвязанных вычислительных систем. *Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение*, 2019, № 5, с. 95–114.
DOI: 10.18698/0236-3933-2019-5-95-114

**NEW ADAPTIVE MULTI-MEMETIC GLOBAL OPTIMIZATION
ALGORITHM FOR LOOSELY COUPLED SYSTEMS**

M.K. Sakharov

msakharov@bmstu.ru

Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

This study introduces a new parallel adaptive multi-memetic population-based algorithm for solving global optimization problems efficiently on loosely coupled computing systems. The exist-

Keywords

Multi-memetic algorithm, landscape analysis, mind evolutionary computation, global optimization, parallel algorithms, grid computing

ent approaches to the synthesis of adaptive population-based algorithms were considered along with the parallelization techniques; distinct features of the loosely coupled computing systems were identified; those features have to be considered carefully when designing algorithms for this class of systems. The proposed algorithm is based on the two level adaptation strategies. The upper level is a static one and allows one to adjust numeric values of the basic algorithm's free parameters before the optimization process using the information about an objective function obtained by means of the landscape analysis. The lower level is a dynamic one and was implemented by means of hybridization of the basic algorithm and several local search methods (memes). The work also presents a new static load balancing method for mapping the proposed algorithm onto parallel computing nodes. The proposed load balancing method takes into consideration both the optimization algorithm's distinct features and the computing system's architecture. This results in higher efficiency of the optimization algorithm comparing to the traditional load balancing methods. A wide performance investigation of the proposed optimization algorithm and its software optimization was carried out in this work with a use of high-dimensional benchmark functions of various classes

Received 14.02.2019

© Author(s), 2019

REFERENCES

- [1] Karpenko A.P. *Sovremennye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy* [Modern search optimization algorithms. Algorithms, inspired by nature]. Moscow, BMSTU Publ., 2014.
- [2] Sakharov M., Karpenko A. A new way of decomposing search domain in a global optimization problem. In: Abraham A., Kovalev S., Tarassov V., Snasel V., Vasileva M., Sukhanov A. (eds). *Proceedings of the Second International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'17). IITI 2017. Advances in Intelligent Systems and Computing*, vol 679. Cham, Springer, 2017, pp. 398–407.
DOI: https://doi.org/10.1007/978-3-319-68321-8_41
- [3] Neri F., Cotta C., Moscato P. *Handbook of memetic algorithms*. Springer, 2012.
DOI: 10.1007/978-3-642-23247-3
- [4] Vassilev V., Fogarty T., Miller J. Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application. In: Ghosh A., Tsutsui S. (eds). *Advances in Evolu-*

tionary Computing. Natural Computing Series. Berlin, Heidelberg, Springer, 2003, pp. 3–44. DOI: https://doi.org/10.1007/978-3-642-18965-4_1

[5] Krasnogor N. Studies on the theory and design space of memetic algorithms. Ph.D. Thesis, University of the West of England, 2002.

[6] Moscato P. On evolution, search, optimization, genetic algorithms and martial arts. Towards memetic algorithms. Technical Report Concurrent Computation Program. California Institute of Technology, 1989.

[7] Ang J.H., Tan K.C., Mamun A.A. An evolutionary memetic algorithm for rule extraction. *Expert Syst. App.*, 2010, vol. 37, iss. 2, pp. 1302–1315. DOI: <https://doi.org/10.1016/j.eswa.2009.06.028>

[8] Ong Y.S., Lim M.H., Zhu N., et al. Classification of adaptive memetic algorithms: a comparative study. *IEEE Trans. Syst., Man, Cybern. B Cybern.*, 2006, vol. 36, iss. 1, pp. 141–152. DOI: [10.1109/TSMCB.2005.856143](https://doi.org/10.1109/TSMCB.2005.856143)

[9] Kerschke P., Preuss M., Hernández C., et al. Cell mapping techniques for exploratory landscape analysis. EVOLVE — a bridge between probability, set oriented numerics, and evolutionary computation V *Advances in Intelligent Systems and Computing*, vol. 288, Cham, Springer, 2014, pp. 115–131. DOI: https://doi.org/10.1007/978-3-319-07494-8_9

[10] Muñoz M.A., Kirley M., Halgamuge S.K. Exploratory landscape analysis of continuous space optimization problems using information content. *IEEE T. Evolut. Comput.*, 2015, vol. 19, iss. 1, pp. 74–87. DOI: [10.1109/TEVC.2014.2302006](https://doi.org/10.1109/TEVC.2014.2302006)

[11] Karpenko A.P., Sakharov M.K. Multi-memes global optimization based on the algorithm of mind evolutionary computation. *Informatsionnye tekhnologii* [Information Technologies], 2014, no. 7, pp. 23–30 (in Russ.).

[12] Sakharov M.K., Karpenko A.P. Adaptive load balancing in the modified mind evolutionary computation algorithm. *Supercomputing Frontiers and Innovations*, 2018, vol. 5, no. 4, pp. 5–14. DOI: [10.14529/jsfi180401](https://doi.org/10.14529/jsfi180401)

[13] Sun Ch., Sun Y., Wang W. A survey of MEC: 1998–2001. *IEEE SMC*, 2002, vol. 6, pp. 445–453. DOI: [10.1109/ICSMC.2002.1175629](https://doi.org/10.1109/ICSMC.2002.1175629)

[14] Sakharov M., Karpenko A. Performance investigation of mind evolutionary computation algorithm and some of its modifications. In: Abraham A., Kovalev S., Tarassov V., Snášel V. (eds). *Proceedings of the First International Scientific Conference “Intelligent Information Technologies for Industry” (IITI’16). Advances in Intelligent Systems and Computing*, vol. 450. Cham, Springer, 2016, pp. 475–486. DOI: https://doi.org/10.1007/978-3-319-33609-1_43

[15] Voevodin V.V., Voevodin V.I. Parallelnye vychisleniya [Parallel computations]. St. Petersburg, BKhV-Peterburg Publ., 2002.

[16] Nelder J.A., Mead R. A simplex method for function minimization. *Comput. J.*, 1965, vol. 7, iss. 4, pp. 308–313. DOI: <https://doi.org/10.1093/comjnl/7.4.308>

[17] Hooke R., Jeeves T.A. Direct search solution of numerical and statistical problems. *JACM*, 1961, vol. 8, iss. 2, pp. 212–229. DOI: [10.1145/321062.321069](https://doi.org/10.1145/321062.321069)

[18] Karpenko A.P., Belous V.V. *Metody optimizatsii (bazovyy kurs)* [Optimization methods (base course)]. Moscow, BMSTU Publ., 2016.

[19] Momin J., Yang X.S. A literature survey of benchmark functions for global optimization problems. *IJMMNO*, 2013, vol. 4, no. 2, pp. 150–194.

DOI: 10.1504/IJMMNO.2013.055204

[20] Sakharov M.K. Issledovaniye modeli kontrolya zabolevayemosti s ispolzovaniyem impulsnoy vaktsinatsii [Study on disease control method using pulse vaccination]. *Naukoemkie tekhnologii i intellektualnye sistemy* [High Technology and Intelligent Systems]. Moscow, BMSTU Publ., 2018, pp. 116–120 (in Russ.).

Sakharov M.K. — Senior Lecturer, Department of Computer Added Design, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, str. 1, Moscow, 105005 Russian Federation).

Please cite this article in English as:

Sakharov M.K. New adaptive multi-memetic global optimization algorithm for loosely coupled systems. *Herald of the Bauman Moscow State Technical University, Series Instrument Engineering*, 2019, no. 5, pp. 95–114 (in Russ.).

DOI: 10.18698/0236-3933-2019-5-95-114

Издательство МГТУ им. Н.Э. Баумана
105005, Москва, 2-я Бауманская ул., д. 5, стр. 1
press@bmstu.ru www.baumanpress.ru

Подписано в печать 26.09.2019

Формат 70 × 108/16

Усл.-печ. л. 10,0

Отпечатано в типографии МГТУ им. Н.Э. Баумана
105005, Москва, 2-я Бауманская ул., д. 5, стр. 1
baumanprint@gmail.com