

МЕТОДЫ ОПТИМИЗАЦИИ | OPTIMIZATION METHOD

Научная статья | Original paper

УДК 519.85:004.021

Анализ эффективности многошагового адаптивного алгоритма оптимизации с прогнозированием

А.В. Пантелеев ✉, Е.А. Хвошнянская

Московский авиационный институт (национальный исследовательский университет)
институт «Компьютерные науки и прикладная математика»

Москва, Российская Федерация

✉ avpantelev@inbox.ru

Резюме

Контекст и актуальность. Задачи поиска глобального экстремума возникают при решении широкого круга прикладных задач оптимизации, в том числе при настройке параметров технических систем и исследовании сложных многоэкстремальных функций. Для таких задач применение классических детерминированных методов часто затруднено из-за наличия локальных экстремумов, сложной структуры целевой функции и отсутствия полной информации о ее свойствах. В связи с этим актуальной является разработка программных средств, реализующих адаптивные и метаэвристические алгоритмы оптимизации. **Цель.** Целью работы является разработка и исследование программного обеспечения для реализации многошагового адаптивного алгоритма оптимизации с прогнозированием, предназначенного для поиска глобального экстремума целевой функции на заданном множестве допустимых решений. **Гипотеза.** Предполагается, что использование прогнозируемого положения решения, памяти об успешных шагах, адаптивного изменения величины шага и генерации новых начальных приближений с помощью распределения Леви позволит повысить устойчивость поиска и уменьшить вероятность преждевременного попадания в окрестности локальных экстремумов. **Методы и материалы.** В работе рассматривается многошаговый адаптивный алгоритм, основанный на последовательном поиске в окрестности прогнозируемого решения, использовании памяти об успешных положениях и генерации новых начальных приближений с применением распределения Леви. Разработанное программное обеспечение позволяет



задавать целевую функцию, область допустимых решений и параметры алгоритма, выполнять вычислительный эксперимент, а также визуализировать процесс поиска и полученные результаты. Для проверки работы программы использовались общепринятые тестовые функции различной степени сложности. **Результаты.** Проведенные вычислительные эксперименты показали, что реализованный алгоритм позволяет находить решения, близкие к точным значениям глобального экстремума, как для простых, так и для многоэкстремальных тестовых функций. Полученные результаты подтверждают работоспособность разработанного программного комплекса и возможность его применения для исследования влияния параметров алгоритма на качество поиска. **Выводы.** Разработанное программное обеспечение может использоваться для решения задач глобальной оптимизации, анализа поведения многошагового адаптивного алгоритма и подбора его параметров. Применение прогнозирования, адаптивного изменения шага и генерации новых начальных точек позволяет повысить устойчивость поиска и снизить вероятность преждевременного попадания в локальный экстремум.

Ключевые слова: метаэвристические алгоритмы, оптимизация, адаптация, прогнозирование, программное обеспечение, тестовые функции

Дополнительные данные. Программный комплекс доступен по адресу: <https://github.com/LizaKhvoshnyanskaya/MAAP-software>

Для цитирования: Пантелеев, А.В., Хвошнянская, Е.А. (2026). Анализ эффективности многошагового адаптивного алгоритма оптимизации с прогнозированием. *Моделирование и анализ данных*, 16(2), 146–165. <https://doi.org/10.17759/mda.2026160208>

Analysis of the efficiency of a multi-step adaptive optimization algorithm with prediction

A.V. Panteleev ✉, **E.A. Khvoshnyanskaya**
Moscow Aviation Institute (National Research University)
Institute of Computer Science and Applied Mathematics
Moscow, Russian Federation
✉ avpanteleev@inbox.ru

Abstract

Context and relevance. Global extremum search problems arise in the solution of a wide range of applied optimization problems, including the tuning of technical system parameters and the study of complex multi-extremal functions. For such problems, the application of classical deterministic methods is often difficult due to the presence of local extreme, the complex structure of the objective function, and the lack of complete information about its properties. In this regard, the development



of software tools implementing adaptive and metaheuristic optimization algorithms is of great relevance. **Objective.** The aim of this work is to develop and study software for implementing a multi-step adaptive optimization algorithm with forecasting, designed to find the global extremum of the objective function within a given set of feasible solutions. **Hypothesis.** It is hypothesized that using a predicted solution position, a history of successful steps, adaptive step size adjustment, and the generation of new initial solutions via the Lévy distribution will improve the stability of the search and reduce the probability of prematurely reaching a local extremum. **Methods and materials.** This paper considers a multi-step adaptive algorithm based on sequential search in the neighborhood of the predicted solution, the use of memory of successful positions, and the generation of new initial solutions using the Lévy distribution. The developed software allows users to specify the objective function, the feasible solution region, and the algorithm parameters; perform computational experiments; and visualize the search process and the results obtained. Benchmark test functions of varying complexity were used to verify the program's performance. **Results.** The computational experiments conducted showed that the implemented algorithm allows finding solutions close to the exact values of the global extremum for both simple and multi-extremal test functions. The results obtained confirm the functionality of the developed software package and the possibility of its application for analysis of the impact of parameter selection on the quality of the search. **Conclusions.** The developed software can be used for the numerical study of global optimization problems, the analysis of the behavior of a multi-step adaptive algorithm, and the selection of its parameters. The use of prediction, adaptive step size adjustment, and the generation of new initial points makes it possible to improve the stability of the search and reduce the probability of premature convergence to a local extremum.

Keywords: metaheuristic algorithms, optimization, adaptation, prediction, software, test functions

Supplemental data. The software package is available at the following address: <https://github.com/LizaKhvoshnyanskaya/MAAP-software>

For citation: Pantelev, A.V., Khvoshnyanskaya, E.A. (2026). Analysis of the efficiency of a multi-step adaptive optimization algorithm with prediction. *Modelling and Data Analysis*, 16(2), 146–165. (In Russ.). <https://doi.org/10.17759/mda.2026160208>

Введение

Решение прикладных задач параметрической оптимизации технических систем и теории оптимального управления динамическими системами, как правило, связано с применением численных методов оптимизации, в развитии которых можно выделить два ключевых подхода.

Первый подход связан с реализацией детерминированных и стохастических процедур поиска экстремума при наложении различных, иногда весьма существенных



ограничений на свойства и структуру целевой функции. Основное преимущество такого подхода заключается в возможности математического обоснования сходимости алгоритма и в ряде случаев получения гарантий по скорости его сходимости. Однако данный подход предъявляет высокие требования к условиям постановки задачи, которые нередко трудно соблюсти в практических применениях, а проверка выполнения этих условий может быть крайне затруднена из-за их сложности. При этом применяются следующие методы и подходы: численные алгоритмы удовлетворения необходимых условий условного или безусловного экстремумов (Поляк, 1983; Bazaraa, Sherali, Shetty, 2006), методы штрафов (внешних штрафов, барьерных функций, комбинированный метод штрафов, метод множителей, метод точных штрафных функций) (Bazaraa, Sherali, Shetty, 2006), методы возможных направлений (Bazaraa, Sherali, Shetty, 2006), методы, использующие идеи моментов и адаптацию (Нестеров, 2010), методы на основе универсального градиентного спуска (Гасников, 2018), диагональные методы (Sergeyev, Kvasov, 2017), интервальные методы (Hansen, Walster, 2004), методы второго порядка и квазиньютоновские методы (Поляк, 1983; Bazaraa, Sherali, Shetty, 2006; Пантелеев, 2025). Достаточно полное единообразное описание методов оптимизации имеется в (Locatelli, 2021; Floudas, Pardalos, 2009).

Второй подход связан с применением метаэвристических методов оптимизации, которые объединяют в себе один или более эвристических методов, управляемых стратегией поиска более высокого уровня. Они способны покидать окрестности локальных экстремумов и выполнять достаточно полное исследование множества допустимых решений. Метаэвристические методы оптимизации позволяют находить решение высокого качества за разумное с практической точки зрения время. В отличие от традиционных методов оптимизации, метаэвристические методы могут применяться в ситуациях, когда практически полностью отсутствует информация о характере и свойствах исследуемой функции. Тем не менее, слабой стороной этой группы методов является, как правило, отсутствие формальных доказательств сходимости и гарантий, что найденное решение действительно близко к глобальному экстремуму. Классификация метаэвристических методов в настоящее время носит условный характер, поскольку характерные группы методов базируются на сходных идеях, и имеется устойчивая тенденция к созданию гибридных алгоритмов, объединяющих в себе сразу несколько хорошо зарекомендовавших себя алгоритмов. Кроме того, один и тот же алгоритм может принадлежать сразу к нескольким группам. Можно выделить следующие группы метаэвристических методов оптимизации: эволюционные методы (Gendreau, Potvin, 2019; Alorf, 2023; Tomar, Bansal, Singh, 2024; Пантелеев, Скавинская, 2023), методы «роевого» интеллекта (Del Ser et al., 2019; Niculina Dragoi, Dafinescu, 2021; Tzanetos, Fister, Dounias, 2020; Карпенко, 2021; Пантелеев, Скавинская, 2023), методы, имитирующие физические процессы (Floudas, Pardalos, 2009; Пантелеев, Скавинская, 2023), биоинспирированные методы (Yang, Chien, Ting, 2015; Карпенко, 2021; Пантелеев, Каранэ, 2024), мультиагентные методы (Пантелеев, Каранэ, 2024; Карпенко, 2021), мультистартовые методы (Floudas, Pardalos, 2009; Пантелеев, Скавинская, 2023).



Разнообразие методов оптимизации и постоянное появление новых алгоритмов и их модификаций объясняется хорошо известной теоремой No Free Lunch (NFL) (Wolpert, Macready, 1997). Эта теорема утверждает отсутствие универсального метода, способного эффективно решать все возможные задачи оптимизации. Поэтому выбор метода определяется конкретной задачей оптимизации: видом целевой функции и ограничений, задающих множество допустимых решений.

В статье предлагается новый метод оптимизации, который объединяет оба описанных подхода. Исследование области допустимых решений использует перелеты Леви (Gutowski, 2001) с целью борьбы с эффектом «застревания» в окрестностях локальных экстремумов. Этап разработки перспективных областей реализуется из прогнозируемой точки, определяемой конечной памятью алгоритма на двух последних шагах (Поляк, 1983; Нестеров, 2010; Гасников, 2018). Поиск из прогнозируемой точки реализует стратегию последовательного сокращения области поиска с ее неполным восстановлением после каждого из проходов (Luus, 2000). Новое положение решения определяется суммарным влиянием прогнозируемой точки и взвешенным влиянием результатов удачных шагов из нее. Процесс завершается после достижения заданного числа проходов. В качестве приближенного решения задачи выбирается наилучшее решение (решения) из полученных в результате реализованных проходов. Алгоритм относится к многошаговым (Поляк, 1983), так как использует концепцию конечной памяти о предыдущих положениях решения.

В работе рассматривается программная реализация многошагового адаптивного алгоритма оптимизации с прогнозированием (МАОП). Алгоритм сочетает последовательный поиск в окрестности прогнозируемой точки, адаптивное изменение величины шага, использование информации об успешных положениях решения и генерацию новых начальных приближений с помощью распределения Леви. Такой подход позволяет совмещать локальное уточнение решения с возможностью перехода к новым перспективным областям поиска.

Материалы и методы

Постановка задачи

Рассматривается задача поиска условного глобального минимума непрерывной целевой функции $f(x)$ на множестве допустимых решений D , т.е. такой точки $x^* \in D$, что

$$f(x^*) = \min_{x \in D} f(x),$$

где $x = (x_1, \dots, x_n)^T$, $D = \{x \mid x_i \in [a_i, b_i], i = 1, \dots, n\}$.

Стратегия поиска решения

Метод использует идеи алгоритма Luus–Jaakola с последовательной редукцией и неполным восстановлением области исследования и ускоренного метода



Нестерова, в котором антиградиент находится не в текущей точке, а в некоторой прогнозируемой точке в процессе поиска. Из начальной (текущей) точки совершается заданное число итераций, в результате которых реализуется проход. Начальная точка для следующего прохода получается при помощи распределения Леви с целью исследования еще неисследованных областей. Поиск завершается после окончания заданного числа проходов. В качестве приближенного решения задачи выбирается наилучшее решение (решения) из полученных в результате реализованных проходов. Согласно классификации алгоритм относится к многошаговым, так как использует концепцию конечной памяти о предыдущих положениях.

Алгоритм решения задачи

Шаг 1. Задать параметры метода:

- максимальное число попыток из текущей точки N ;
- максимальное число проходов P ;
- число итераций за время прохода K ;
- наименьшая величина шага R ;
- параметр уменьшения величины шага α ($0 < \alpha < 1$);
- параметр восстановления величины шага η ($0 < \eta < 1$);
- величина шага, используемого при полетах Леви β ;
- параметр распределения Леви λ ($1 < \lambda \leq 3$).

Шаг 2. Задание начального приближения $x^{(0)} \in D$:

а) используя равномерный закон распределения. Для этого с помощью равномерного распределения на единичном отрезке $[0;1]$ генерировать n случайных точек $P_i^0, i=1, \dots, n$. Используя линейное преобразование, каждую точку отобразить на соответствующий ей промежуток $[a_i, b_i]$: $x_i^{(0)} = (b_i - a_i)P_i^0 + a_i$;

б) положить $x_i^{(0)} = \frac{a_i + b_i}{2}, i=1, \dots, n$.

Шаг 3. Положить $p=1$ (счетчик числа проходов), множество $\text{Pool} = \emptyset$. Задать начальную величину шага $t_0 = \frac{1}{2} \min_{i=1, \dots, n} (b_i - a_i)$.

Шаг 4. Положить $k=0$ (счетчик числа итераций).

Шаг 5. Найти прогнозируемое решение:

$$z^{(k)} = x^{(k)} + (1 - e^{-\frac{k}{5}})(x^{(k)} - x^{(k-1)}) \text{rand}[0,1],$$

где $\text{rand}[0,1]$ – случайная величина, описываемая равномерным законом распределения на отрезке $[0,1]$.

Если для какого-либо значения i не выполняется ограничение, т.е. $z_i^{(k)} \notin [a_i, b_i]$, то положить $z_i^{(k)} = x_i^{(k)}$.

Шаг 6. Реализовать поиск в окрестности прогнозируемого решения.

Шаг 6.1. Положить $s=0$ (число удачных попыток), $j=1$ (число попыток из точки), $J = \emptyset$ (множество номеров удачных попыток из текущей точки).

Шаг 6.2. Генерировать случайный вектор $\xi^j = (\xi_1^j, \dots, \xi_n^j)^T$, где ξ_i^j – случайная величина, равномерно распределенная на отрезке $[-1,1]$, которая получается



из случайной величины η_i^j , равномерно распределенной на $[0,1]$, с использованием преобразования $\xi_i^j = 2\eta_i^j - 1$.

Шаг 6.3. Найти

$$y^j = z^{(k)} + t_k \frac{\xi^j}{\|\xi^j\|},$$

где $\|\xi^j\|$ – евклидова норма вектора.

Шаг 6.4. Если $y^j \in D$ и $f(y^j) < f(z^{(k)})$, то шаг удачный, положить $s = s + 1$, номер j включить в множество J . Если $y^j \notin D$ или $f(y^j) \geq f(z^{(k)})$, то шаг неудачный.

Если $j < N$, то положить $j = j + 1$ и перейти к шагу 6.2.

Если $j = N$, то процедуру поиска с текущей величиной шага завершить.

Шаг 6.5. Если $s = 0$, проверить условие завершения прохода:

- если $t_k \leq R$, проход завершить, включить найденное решение в множество Pool и перейти к шагу 8.;
- если $t_k > R$, положить $t_k = \alpha t_k$ и перейти к шагу 6.1.

Если $s \neq 0$, то упорядочить решения $y^j, j \in J$ по неубыванию значения целевой функции: $y^{(1)}, \dots, y^{(s)}$. Затем наилучшему удачному решению $y^{(1)}$ поставить в соответствие вес, равный 1, а наихудшему $y^{(s)}$, равный $1/s$, т.е. всем удачным решениям сопоставить значения веса, уменьшающиеся по линейному закону $\frac{s+1-j}{s}, j = 1, \dots, s$.

Шаг 6.6. Найти новое положение решения, используя средневзвешенную величину:

$$x^{(k+1)} = z^{(k)} + \frac{\sum_{j=1}^s y^{(j)} \frac{s+1-j}{s}}{(s+1)/2}.$$

Если $x^{(k+1)} \notin D$, то положить $x^{(k+1)} = y^{(1)}$.

Шаг 7. Проверка завершения итераций в рамках текущего прохода.

Если $k < K$, то положить $k = k + 1$ и перейти к шагу 5.

Если $k = K$, то полученное решение включить в множество Pool и перейти к шагу 8.

Шаг 8. Проверка завершения числа проходов.

Если $p < P$, то положить $p = p + 1$ и перейти к шагу 9.

Если $p = P$, процесс поиска завершить. Из множества Pool выбрать решения x^* , соответствующие наименьшему значению целевой функции $f(x^*)$.

Шаг 9. Реализация следующего прохода.

Найти новое начальное значение величины шага $t_0 = \frac{\eta^p}{2} \min_{i=1, \dots, n} (b_i - a_i)$.

Генерировать новое положение начальной точки с помощью распределения Леви:

$$x_i^{(0)} = z_i^{(k)} + \frac{\beta}{p} Levy_i(\lambda), i = 1, \dots, n,$$

где для генерации величины $Levy_i(\lambda)$ требуется:



- генерировать число Q_i с помощью равномерного закона распределения на отрезке $[\varepsilon, b_i - a_i]$, $\varepsilon = 10^{-7}$ – константа различимости;
- вычислить значения $\theta_i = 2\pi Q_i$ и $L_i = Q_i^{-\frac{1}{\lambda}}$;
- вычислить

$$Levy_i(\lambda) = L_i \sin \theta_i, i = 1, \dots, \left\lfloor \frac{n}{2} \right\rfloor; Levy_i(\lambda) = L_i \cos \theta_i, i = \left\lfloor \frac{n}{2} \right\rfloor + 1, \dots, n.$$

Если $x_i^{(0)} \notin [a_i, b_i]$, процесс генерации следует повторить.

Перейти к шагу 4.

Замечания.

1. На шаге 5 находится прогнозируемое решение с целью начала поиска из него, а не из текущего решения. При этом используется схема двухшагового быстрого градиентного метода [Поляк, 1983; Нестеров, 2010; Гасников, 2018]. Для учета информации о двух последних итерациях используется величина $(1 - e^{-\frac{k}{5}})(x^{(k)} - x^{(k-1)})$ с весовым коэффициентом, отличающимся от известного $\frac{k-1}{k+r-1}$, $r \geq 3$. Умножение этого слагаемого на $rand[0,1]$ способствует активизации поиска (при этом детерминированный подход заменяется стохастическим).
2. При поиске из прогнозируемой точки на шаге 6.3 реализуется метод случайного поиска с возвратом при неудачном шаге, дополненный процедурой сохранения в памяти удачных попыток из данной точки, которые ранжируются по величине целевой функции. На шаге 6.5 при нахождении новой величины шага фактически проводится редукция области поиска, как в методе (Luus, 2000).
3. На шаге 6.6 авторами предложена формула для нахождения нового положения решения с использованием средневзвешенной величины удачных решений, найденных на шаге 6.5.
4. На шаге 9 реализуется переход к следующему проходу. При этом для нахождения величины шага используется стратегия метода (Luus, 2000) с частичным восстановлением области поиска (с коэффициентом η).

Программное обеспечение

На основе изложенных алгоритмов разработан программный комплекс поиска глобального экстремума функций. Для его создания использовалась среда разработки Microsoft Visual Studio, язык программирования C# (Пантелеев, Каранэ, 2024).

Описанные в данном разделе методы соответствует задаче поиска минимума целевой функции. Так как заданные тестовые функции имеют глобальный максимум, при разработке программного обеспечения этого алгоритма был заменен знак перед целевыми функциями на противоположный.

Возможности программного комплекса позволяют изучить алгоритмы методов, а также влияние параметров каждого метода на результат его работы. В список выби-раемых функций включены стандартные многоэкстремальные тестовые функции двух переменных, для которых известно точное решение (Пантелеев, Скавинская, 2023).



На рис. 1 представлено главное окно многошагового адаптивного алгоритма оптимизации с прогнозированием, где пользователь может выбрать вид целевой функции, задать множество допустимых решений и параметры метода, просматривать результаты работы.

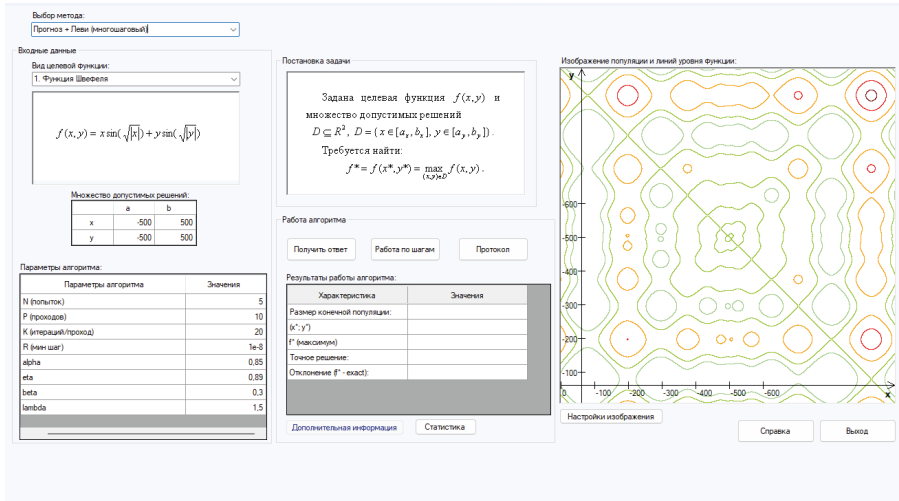


Рис. 1. Главное окно программы многошагового адаптивного алгоритма оптимизации с прогнозированием

Fig. 1. Main window of the multi-step adaptive optimization algorithm with prediction

Результаты

Пример 1. Найдем глобальной максимум функции Шаффера. Зададим множество допустимых решений $x, y \in [-10, 10]$. Выберем следующие параметры алгоритма: максимальное число попыток из текущей точки $N = 100$; максимальное число проходов $P = 10$; число итераций за время прохода $K = 5$; наименьшая величина шага $R = 10^{-8}$; параметр уменьшения величины шага $\alpha = 0,95$; параметр восстановления величины шага $\eta = 0,89$; величина шага, используемого при полетах Леви $\beta = 0,4$; параметр распределения Леви $\lambda = 1,5$.

На рис. 2 представлена популяция на начальной ($k = 1$), промежуточных ($k = 40, k = 75$) и конечной ($k = 100$) итерациях. Красным обозначено наилучшее решение в популяции на данной итерации.

Результаты работы алгоритма: наилучшее решение $(x^*, y^*) = (0,006; -0,007)$; значение целевой функции $f(x^*, y^*) = 0,999913$; отклонение от точного решения $\Delta = 8,7 * 10^{-5}$.

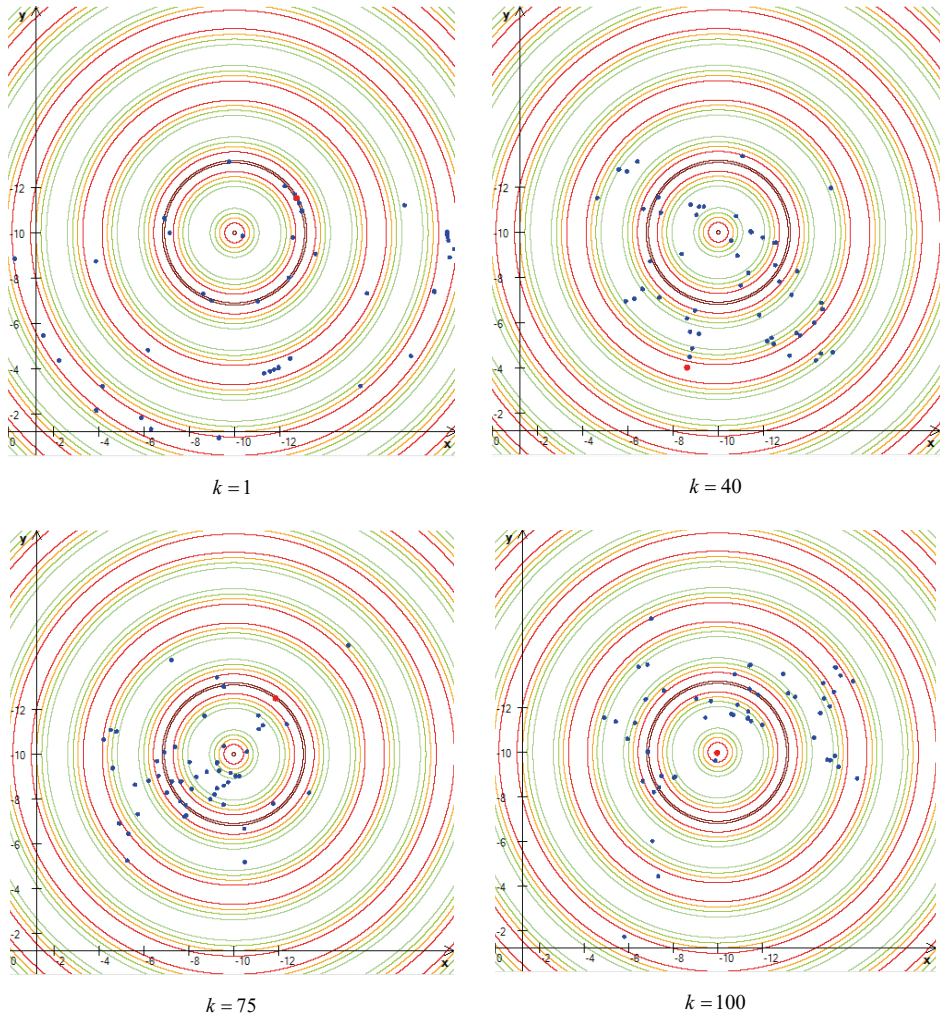


Рис. 2. Начальная, промежуточная и конечная популяции
Fig. 2. Initial, intermediate, and final populations

Пример 2. Найдем глобальной максимум функции Розенброка. Зададим множество допустимых решений $x, y \in [-2, 2]$. Выберем следующие параметры алгоритма: максимальное число попыток из текущей точки $N = 100$; максимальное число проходов $P = 20$; число итераций за время прохода $K = 5$; наименьшая величина шага $R = 10^{-8}$; параметр уменьшения величины шага $\alpha = 0,9$; параметр восстановления величины шага $\eta = 0,89$; величина шага, используемого при полетах Леви $\beta = 0,4$; параметр распределения Леви $\lambda = 1,5$.



На рис. 3 представлена популяция на начальной ($k = 1$), промежуточных ($k = 15, k = 25$) и конечной ($k = 40$) итерациях. Красным обозначено наилучшее решение в популяции на данной итерации.

Результаты работы алгоритма: наилучшее решение $(x^*, y^*) = (-1,006; -1,013)$; значение целевой функции $f(x^*, y^*) = -4,3 * 10^{-5}$; отклонение от точного решения $\Delta = 4,3 * 10^{-5}$.

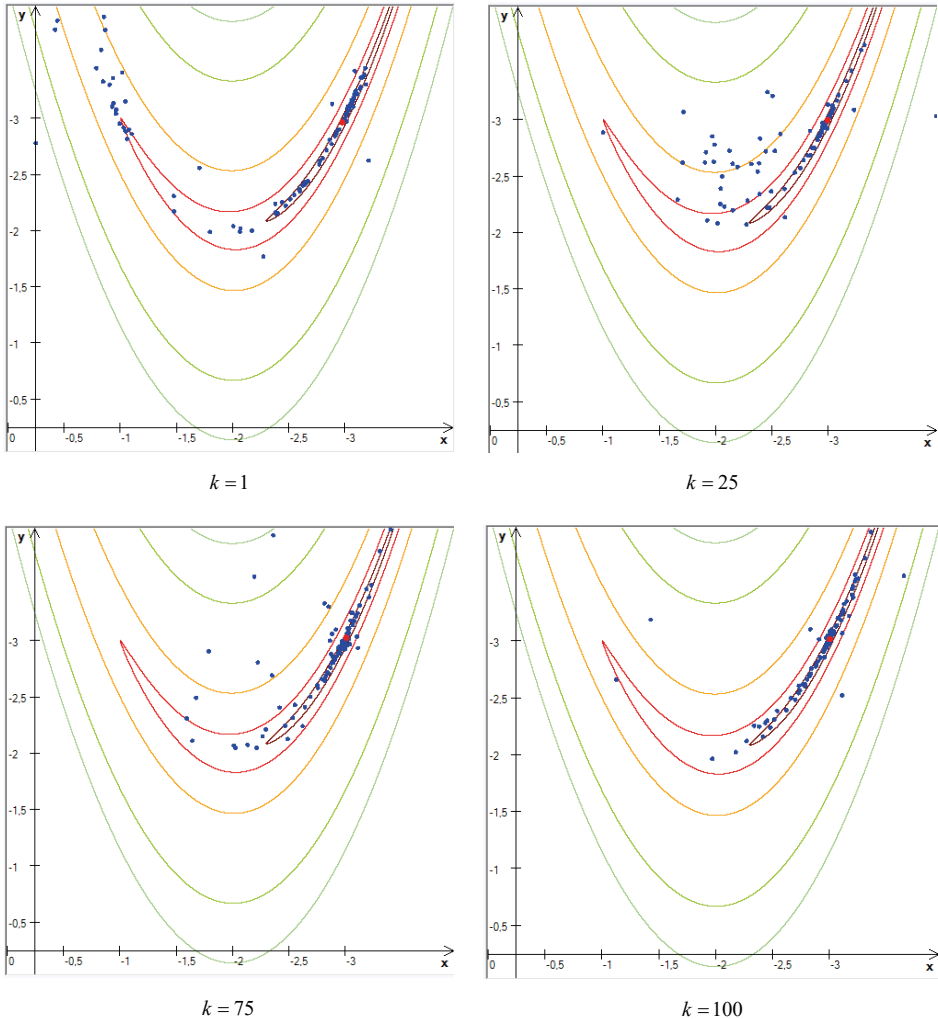


Рис. 3. Начальная, промежуточная и конечная популяции
Fig. 3. Initial, intermediate, and final populations



Пример 3. Оптимизация конструкции зубчатой передачи (Сандгрэн, 1988). Рассматривается редуктор, включающий четыре шестерни для уменьшения угловой скорости ведомого вала. Все шестерни могут иметь от 12 до 60 зубьев. Цель — минимизация стоимости конструкции. Переменными являются $T_d = x_1$ — число зубьев ведущей шестерни D , $T_b = x_2$, $T_a = x_3$ — числа зубьев промежуточных шестерен, $T_f = x_4$ — число зубьев ведомой шестерни. В результате получается конструкция с передаточным отношением $TR = \frac{T_d T_b}{T_a T_f}$.

Постановка задачи оптимизации:

$$f(x) = \left(\frac{1}{6,931} - \frac{\langle x_1 \rangle \langle x_2 \rangle}{\langle x_3 \rangle \langle x_4 \rangle} \right)^2 \rightarrow \min_{x \in D}$$

$$D = [12; 60] \times [12; 60] \times [12; 60] \times [12; 60],$$

где $\langle \cdot \rangle$ — целая часть числа.

Результат решения с параметрами $N = 100$, $P = 80$, $K = 30$, $R = 10^{-8}$, $\alpha = 0,7$; $\eta = 0,89$; $\beta = 12$; $\lambda = 1,5$; $x^* = (16; 19; 43; 49)^T$, $f(x^*) = 2,7001 \times 10^{-12}$. Сравнение с результатами других методов показано в таблице 1.

Таблица 1 / Table 1

Сравнение результатов работы алгоритма МАОП с другими методами
Comparison of the results of the new algorithm with other methods

		Best
(Sandgren, 1988)	[18, 22, 45, 60], 0.146667	5.712×10^{-6}
(Kannan, Kramer, 1994)	[13, 15, 33, 41], 0.144124	2.146×10^{-8}
(Deb, Goyal, 1996)	[19, 16, 49, 43], 0.144281	2.701×10^{-12}
(Gandomi et al., 2013)	[19, 16, 43, 49], 0.144281	2.701×10^{-12}
(Garg, 2019)	[19, 16, 43, 49], 0.14428096	$2.7008571 \times 10^{-12}$
(Nekoo et al., 2022)	[16, 19, 43, 49], 0.14428096	$2.7008571 \times 10^{-12}$
МАОП	[16, 19, 43, 49], 0.14428096	2.7001×10^{-12}

Анализ эффективности метода

Исследуемые методы применялись к некоторым функциям из набора тестовых функций. Для каждой функции проводилась серия из 100 решений одной и той же задачи с одними и теми же значениями параметров. Для полученной выборки $\{f^1, f^2, \dots, f^{100}\}$ вычислялись среднее значение отклонения полученного решения от точного $\bar{\Delta f} = \frac{1}{100} \sum_{i=1}^{100} \Delta f_i$, где $\Delta f_i = |f(x^*) - f^i|$; наименьшее значение отклонения $\Delta f_{best} = \min_i \Delta f_i$; среднеквадратичное отклонение $\bar{\sigma}_f = \sqrt{S_{100}}$,



где $\bar{S}_{100} = \frac{1}{100} \sum_{i=1}^{100} (\Delta f_i - \overline{\Delta f})^2$; количество успехов $n_{\text{усп}}$ (успехом считалось попадание лучшей точки в ε – окрестность точного решения, $\varepsilon = \frac{\max_{i=1, \dots, n} |b_i - a_i|}{1000}$).

Результаты, полученные для каждой функции, представлены в табл. 2—4.

Таблица 2 / Table 2

Анализ выбора параметров метода. Корневая функция
Analysis of method parameter selection. Root function

Параметры метода / Method parameters								$\overline{\Delta f}$	Δf_{best}	$\overline{\sigma}_f$	$n_{\text{усп}}$
N	P	K	R	α	η	β	λ				
40	10	15	10^{-8}	0,8	0,5	0,3	2	0,000177	0,000000	0,000227	100
30	10	15	10^{-8}	0,8	0,5	0,3	2	0,000227	0,000002	0,000284	100
50	10	15	10^{-8}	0,8	0,5	0,3	2	0,000182	0,000005	0,000208	100
40	20	15	10^{-8}	0,8	0,5	0,3	2	0,000152	0,000002	0,000227	100
40	30	15	10^{-8}	0,8	0,5	0,3	2	0,000142	0,000001	0,000168	100
40	30	25	10^{-8}	0,8	0,5	0,3	2	0,000005	0,000000	0,000005	100
40	20	15	10^{-9}	0,8	0,5	0,3	2	0,000146	0,000003	0,000174	100
40	20	15	10^{-8}	0,5	0,5	0,3	2	0,000001	0,000000	0,000003	100
40	20	15	10^{-8}	0,5	0,9	0,3	2	0,000000	0,000000	0,000000	100
40	20	15	10^{-8}	0,5	0,9	0,9	2	0,000009	0,000000	0,000016	100

Таблица 3 / Table 3

Анализ выбора параметров метода. Мульти-функция
Analysis of method parameter selection. Multi-function

Параметры метода / Method parameters								$\overline{\Delta f}$	Δf_{best}	$\overline{\sigma}_f$	$n_{\text{усп}}$
N	P	K	R	α	η	β	λ				
40	10	20	10^{-6}	0,89	0,89	0,3	1,5	0,800518	0,000000	0,888984	20
40	20	20	10^{-6}	0,89	0,89	0,3	1,5	0,265937	0,000000	0,420819	32
40	20	30	10^{-6}	0,89	0,89	0,3	1,5	0,090201	0,000000	0,227720	41



Параметры метода / Method parameters								$\overline{\Delta f}$	Δf_{best}	$\overline{\sigma}_f$	$n_{усп}$
N	P	K	R	α	η	β	λ				
40	30	30	10^{-6}	0,89	0,89	0,3	1,5	0,116452	0,000000	0,329003	41
40	30	30	10^{-8}	0,89	0,89	0,3	1,5	0,171381	0,000000	0,374197	47
40	30	30	10^{-8}	0,5	0,89	0,3	1,5	0,074884	0,000000	0,191694	47
40	30	30	10^{-8}	0,5	0,6	0,3	1,5	0,415003	0,000000	0,457901	23
40	30	30	10^{-8}	0,5	0,99	0,3	1,5	0,064881	0,000000	0,182047	52
40	30	30	10^{-8}	0,5	0,99	0,9	1,5	0,019978	0,000000	0,097809	45
40	30	30	10^{-8}	0,5	0,99	0,9	2,9	0,033619	0,000000	0,123121	47

Таблица 4 / Table 4

Анализ выбора параметров метода. Функция Розенброка
Analysis of method parameter selection. Rosenbrock function

Параметры метода / Method parameters								$\overline{\Delta f}$	Δf_{best}	$\overline{\sigma}_f$	$n_{усп}$
N	P	K	R	α	η	β	λ				
40	5	10	10^{-6}	0,89	0,89	0,3	1,5	0,061874	0,000000	0,351095	32
40	10	10	10^{-6}	0,89	0,89	0,3	1,5	0,000023	0,000000	0,000083	70
40	5	20	10^{-6}	0,89	0,89	0,3	1,5	0,011783	0,000000	0,101855	78
40	10	20	10^{-6}	0,2	0,89	0,3	1,5	0,000138	0,000000	0,000468	48
40	10	20	10^{-6}	0,5	0,89	0,3	1,5	0,000007	0,000000	0,000016	74
40	10	20	10^{-6}	0,99	0,89	0,3	1,5	0,009835	0,000000	0,031232	16
40	10	20	10^{-6}	0,5	0,5	0,3	1,5	0,000002	0,000000	0,000008	96
40	10	20	10^{-6}	0,5	0,2	0,3	1,5	0,000436	0,000000	0,002041	46
40	10	20	10^{-6}	0,5	0,5	0,9	1,5	0,000011	0,000000	0,000029	67
40	10	20	10^{-6}	0,5	0,5	0,9	0,5	0,000025	0,000000	0,000143	78

Анализ работы метода показал, что с большинством тестовых функций метод справляется успешно. В серии из 100 запусков удается найти глобальный максимум функций с достаточно высокой точностью или вообще точное решение. Метод



показал, что увеличение значений N , P и K обычно повышает надёжность метода, но одновременно увеличивает вычислительные затраты.

Рекомендации по выбору параметров

Максимальное количество попыток N определяет число попыток из точки. Чем больше параметр размера количества попыток, тем лучше полученный результат, но в силу возрастающих вычислительных затрат замедляется алгоритм. Рекомендуемое значение $N \geq 40$.

Максимальное число проходов P определяет, сколько раз алгоритм может повторно запускать процедуру поиска с новым начальным приближением. Увеличение P повышает вероятность нахождения более точного решения, поскольку алгоритм получает больше возможностей уточнить результат и выйти из неудачной области поиска. При малом значении P процесс может завершиться преждевременно, особенно для многоэкстремальных функций. Рекомендуемое значение $P \geq 10$.

Число итераций за время прохода K задает глубину локального поиска в рамках одного прохода. При увеличении K повышается точность найденного решения, но возрастает время работы алгоритма. Анализ результатов показал, что при слишком малом значении K алгоритм не всегда успевает приблизиться к искомой точке глобального экстремума. Рекомендуемое значение $K \geq 15$.

Наименьшая величина шага R используется как критерий завершения прохода. Чем меньше значение R , тем более точным может быть найденное решение. Слишком малое значение параметра увеличивает число вычислений и может привести к избыточному уточнению. Рекомендуемое значение $R \in [10^{-9}, 10^{-6}]$.

Параметр уменьшения величины шага α определяет скорость уменьшения шага при неудачном поиске. При больших значениях α шаг уменьшается медленно, что позволяет дольше исследовать область, но может замедлить сходимость. Рекомендуемое значение $0 < \alpha < 1$.

Параметр восстановления величины шага η задает увеличение шага после успешного поиска. Данный параметр позволяет алгоритму адаптивно изменять шаг и продолжать движение в перспективном направлении. При слишком малом значении η шаг восстанавливается недостаточно быстро, а при слишком большом возможны резкие переходы по области поиска. Рекомендуемое значение $0 < \eta < 1$.

Величина шага, используемого при полетах Леви β . Подбирается в зависимости от особенностей решаемой задачи оптимизации: размеров множества допустимых решений, свойств целевой функции.

Параметр распределения Леви λ . Рекомендуемое значение $1 < \lambda \leq 3$.

Обсуждение результатов

Полученные результаты показывают, что разработанное программное обеспечение позволяет наглядно исследовать работу многошагового адаптивного алгоритма оптимизации с прогнозированием. На тестовых функциях алгоритм находил решения, близкие к известным значениям глобального экстремума, что подтверждает корректность его программной реализации.



По сравнению с классическими градиентными методами рассматриваемый алгоритм не требует вычисления производных целевой функции, поэтому может применяться к задачам со сложным рельефом поверхностей уровня. В отличие от других алгоритмов он использует информацию об успешных положениях решения, что делает процесс поиска более направленным.

В сравнении с эволюционными и методами «роевого интеллекта» предложенный алгоритм имеет более простую структуру, так как не требует операторов скрещивания, мутации или моделирования взаимодействия частиц. При этом использование распределения Леви позволяет выполнять переходы к новым областям поиска и снижать вероятность преждевременного попадания в локальный экстремум.

Анализ экспериментов показал, что параметры алгоритма существенно влияют на качество результата. Увеличение значений N , P и K повышает вероятность нахождения точного решения, однако увеличивает время вычислений. Выбор величин параметров α , η , β и λ позволяет регулировать баланс между локальным уточнением решения и исследованием новых областей множества допустимых решений.

Таким образом, результаты подтверждают выдвинутую гипотезу о том, что сочетание прогнозирования, памяти об успешных шагах, адаптивного изменения шага и применение распределения Леви повышает устойчивость алгоритма при решении задач глобальной оптимизации.

Заключение

В работе создано программное обеспечение для реализации многошагового адаптивного алгоритма оптимизации с прогнозированием. Программа позволяет задавать целевую функцию, область допустимых решений и параметры алгоритма, выполнять численный эксперимент и визуализировать процесс поиска решения.

Проведенные эксперименты на тестовых функциях показали, что реализованный алгоритм способен находить решения, близкие к точным значениям глобального экстремума. Использование прогнозирования, памяти об успешных положениях, адаптивного изменения шага и распределения Леви позволяет сочетать локальное уточнение решения с исследованием новых областей множества допустимых решений.

Полученные результаты подтверждают работоспособность разработанного программного комплекса и возможность его применения для анализа поведения алгоритма, подбора параметров и решения задач глобальной оптимизации. Перспективой дальнейших исследований является расширение набора тестовых функций, сравнение с другими метаэвристическими методами и применение программы в прикладных задачах оптимизации сложных технических систем.

Ограничения. Ограничением исследования является то, что проверка программного обеспечения выполнялась на ограниченном наборе тестовых функций с известными значениями экстремума. Это позволяет оценить корректность реализации алгоритма, однако не полностью отражает сложность реальных прикладных задач. Кроме того, качество работы метода зависит от выбора параметров алгоритма,



поэтому при их неудачном задании возможно увеличение времени вычислений или снижение точности результата. Как и другие метаэвристические методы, предложенный алгоритм не гарантирует нахождения глобального экстремума за конечное число итераций для произвольной целевой функции. В дальнейшем исследование может быть расширено за счет более подробного сравнения с другими методами оптимизации и применения программы в прикладных задачах поиска оптимальных законов управления динамическими системами (Пантелеев, Бортакровский, 2016).

Limitations. A limitation of this study is that the software was tested on a set of test functions with known extremes values. While this allows to study the algorithm's correct implementation, it does not fully reflect the complexity of real-world application problems. Furthermore, the performance of the method depends on the choice of algorithm parameters; therefore, if they are set incorrectly, this may lead to increased computation time or reduced accuracy of the result. Like other metaheuristic methods, the proposed algorithm does not guarantee finding the global extremum for an arbitrary objective function within a finite number of iterations. Future research could be expanded through a more detailed comparison with other optimization methods and the application of the software to real-world problems (Pantelev, Bortakovskii, 2016).

Список источников / References

1. Бортакровский, А.С. (2016) *Теория управления в примерах и задачах: Учебное пособие / А.С. Бортакровский, А.В. Пантелеев.* — 2-е издание, стереотипное. — Москва: Общество с ограниченной ответственностью «Научно-издательский центр ИНФРА-М». — (Бакалавриат). — ISBN 978-5-16-011862-8. — EDN VYIWMR.
Bortakovskii, A.S., Pantelev, A.V. (2016) *Control theory in examples and problems.* Moscow: INFRA-M. EDN VYIWMR (In Russ.)
2. Гасников, А.В. (2018) *Современные численные методы оптимизации. Метод универсального градиентного спуска.* — М.: МФТИ.
Gasnikov, A.V. (2018) *Modern Numerical Methods for Optimization: The Universal Gradient Descent Method.* Moscow: MIPT. (In Russ.)
3. Карпенко, А.П. (2021) *Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой.* — М.: МГТУ им. Н.Э. Баумана.
Karpenko, A.P. (2021) *Modern Search Optimization Algorithms: Algorithms Inspired by Nature.* Moscow: Bauman Moscow State Technical University. (In Russ.)
4. Нестеров, Ю.Е. (2010) *Методы выпуклой оптимизации.* — М.: МЦНМО.
Nesterov, Yu.E. (2010) *Methods of Convex Optimization.* — Moscow: MCNMO. (In Russ.)
5. Пантелеев, А.В. (2023) *Метаэвристические стратегии и алгоритмы глобальной оптимизации / А.В. Пантелеев, Д.В. Скавинская.* — Москва: Факториал. — ISBN 978-5-98688-322-9. — EDN XUERAR.
Pantelev, A.V., Skavinskaya, D.V. (2023) *Metaheuristic Strategies and Algorithms for Global Optimization.* Moscow: Factorial. (In Russ.)
6. Пантелеев, А.В. (2025) *Методы оптимизации в машинном обучении / А.В. Пантелеев.* — Москва: Издательство «Доброе слово и Ко». — ISBN 978-5-605-36885-4. — EDN QDZHUZ.



- Panteleev, A.V. (2025) *Optimization Methods in Machine Learning*. Moscow: Dobroe Slovo i Ko. EDN QDZHUZ. (In Russ.)
7. Пантелеев, А.В., Каранэ, М.М.С. (2024) *Мультиагентные и биоинспирированные методы оптимизации технических систем*. — М.: Доброе слово и Ко. EDN: NJFRAO.
Panteleev, A.V., Karane, M.M.S. (2024) *Multi-agent and Bio-inspired Methods for the Optimization of Technical Systems*. Moscow: Dobroe Slovo i Ko. EDN: NJFRAO. (In Russ.)
 8. Поляк, Б.Т. (1983) *Введение в оптимизацию*. — М.: Наука.
Polyak, B.T. (1983) *Introduction to Optimization*. Moscow: Nauka. (In Russ.).
 9. Alorf, A. (2023) A survey of recently developed metaheuristics and their comparative analysis. *Engineering Applications of Artificial Intelligence*, 117. Part A. Art. 105622. DOI: 10.1016/j.engappai.2022.105622.
 10. Bazaraa, M.S., Sherali, H.D., Shetty, C.M. (2006) *Nonlinear Programming: Theory and Algorithms*. — 3rd ed. — Hoboken, NJ, USA: John Wiley & Sons.
 11. Deb, K., Goyal, M. (1996) A combined genetic adaptive search (GeneAS) for engineering design. *Computer Science and informatics*, 26, 30—45.
 12. Del Ser, J., Osaba, E., Molina, D., Yang, X.-S., Salcedo-Sanz, S., Camacho, D., Das, S., Suganthan, P.N., Coello Coello, C.A., Herrera, F. (2019) Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation*, 48, 220—250. DOI: 10.1016/j.swevo.2019.04.008.
 13. *Encyclopedia of Optimization* (2009) / Eds. C.A. Floudas, P.M. Pardalos. — New York, NY, USA: Springer.
 14. Gandomi, A.H., Yang, X.-S.; Alavi, A.H. (2013) Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. *Engineering with Computers*, 29(1), 17—35. DOI: 10.1007/s00366-011-0241-y.
 15. Garg, H. (2019) A hybrid GSA-GA algorithm for constrained optimization problems. *Information Sciences*, 478, 499—523. DOI: 10.1016/j.ins.2018.11.041.
 16. Gutowski, M. (2001) Lévy flights as an underlying mechanism for global optimization algorithms. — Cornell University. arXiv: math-ph/0106003.
 17. *Handbook of Metaheuristics* (2019) / Eds. M. Gendreau, J-Y. Potvin. — New York, NY, USA: Springer. DOI: 10.1007/978-3-319-91086-4.
 18. Hansen, E., Walster, G.W. (2004) *Global Optimization Using Interval Analysis*. — New York, NY, USA: Marcel Dekker.
 19. Kannan, B.K.; Kramer, S.N. (1994) An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. *J. Mech. Des. Trans.*, 116, 318—320.
 20. Locatelli, M., Schoen, F. (2021) (Global) Optimization: Historical notes and recent developments. *EURO Journal on Computational Optimization*, 9. Art. 100012. DOI: 10.1016/j.ejco.2021.100012.
 21. Luus, R. (2000) *Iterative Dynamic Programming*. — London, UK: Chapman & Hall/CRC. DOI: 10.1201/9781420036022.



22. Nekoo, S.R., Acosta, J.A., Ollero, A. (2022) A search algorithm for constrained engineering optimization and tuning the gains of controllers. *Expert Systems with Applications*. 206. 117866. DOI:10.1016/j.eswa.2022.117866.
23. Niculina Dragoi, E., Dafinescu, V. (2021) Review of metaheuristics inspired from the animal kingdom. *Mathematics*, 9(18). Art. 2335. DOI: 10.3390/math9182335.
24. Sangren, E. (1990) Nonlinear integer and discrete programming in mechanical design optimization. *J.Mech.Des.-T.ASME*, 112(2), 223—229.
25. Sergeyev, Y.D., Kvasov, D.E. (2017) *Deterministic Global Optimization: An Introduction to the Diagonal Approach*. — New York, NY, USA: Springer.
26. Tomar, V., Bansal, M., Singh, P. (2024) Metaheuristic algorithms for optimization: A brief review. *Engineering Proceedings*, 59 (1). Art. 238. DOI: 10.3390/engproc2023059238.
27. Tzanetos, A., Fister, I., Dounias, G. (2020) A comprehensive database of nature-inspired algorithms. *Data in Brief*, 31. Art. 105792. DOI: 10.1016/j.dib.2020.105792.
28. Wolpert, D.H., Macready, W.G. (1997) No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), 67—82.
29. Yang, X.S., Chien, S.F., Ting, T.O. (2015) *Bio-inspired Computation and Optimization*. — New York, NY, USA: Morgan Kaufmann. DOI: 10.1016/B978-0-12-801538-4.00001-X.

Информация об авторах

Пантелеев Андрей Владимирович, доктор физико-математических наук, профессор, Московский авиационный институт (МАИ), Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0003-2493-3617>, e-mail: avpanteleev@inbox.ru

Хвошнянская Елизавета Аркадьевна, аспирант, ассистент, Московский авиационный институт (ФГБОУ ВО СПбГУ), (МАИ), Москва, Российская Федерация, ORCID: <https://orcid.org/0009-0009-6901-8720>, e-mail: liza190401@mail.ru

Information about the authors

Andrey V. Panteleev, Doctor of Physical and Mathematical Sciences, Professor, Moscow Aviation Institute (MAI), Moscow, Russian Federation, ORCID: <https://orcid.org/0000-0003-2493-3617>, e-mail: avpanteleev@inbox.ru

Elizaveta A. Khvoshnyanskaya, Graduate Student, Assistant, Moscow Aviation Institute (Federal State Budgetary Educational Institution of Higher Education St. Petersburg State University), (MAI), Moscow, Russian Federation, ORCID: <https://orcid.org/0009-0009-6901-8720>, e-mail: liza190401@mail.ru

Вклад авторов

Пантелеев А.В. — идеи исследования; аннотирование, написание и оформление рукописи; планирование исследования; контроль за проведением исследования.

Хвошнянская Е.А. — применение статистических, математических или других методов для анализа данных; проведение эксперимента; сбор и анализ данных; визуализация результатов исследования.

Все авторы приняли участие в обсуждении результатов и согласовали окончательный текст рукописи.



Contribution of the authors

A.V. Panteleev — research concepts; abstract preparation, manuscript writing and formatting; research planning; supervision of the study.

E.A. Khvoshnyanskaya — application of statistical, mathematical, or other methods for data analysis; conducting the experiment; data collection and analysis; visualization of research results.

All authors participated in the discussion of the results and agreed on the final text of the manuscript.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest

The authors declare no conflict of interest.

Поступила в редакцию 14.05.2026

Поступила после рецензирования 20.05.2026

Принята к публикации 21.05.2026

Опубликована 30.06.2026

Received 2026.05.14

Revised 2026.05.20

Accepted 2026.05.21

Published 2026.06.30