

ЭЛЕКТРОНИКА, ФОТОНИКА,
ПРИБОРОСТРОЕНИЕ
И СВЯЗЬ

ELECTRONICS, PHOTONICS,
INSTRUMENT MAKING
AND COMMUNICATIONS

УДК 519.688

DOI: 10.17212/2782-2001-2026-2-115-132

Эвристические методы поиска минимума функции при наличии дополнительных ограничений^{*}

А.А. МИЦЕЛЬ

РФ, 634034, г. Томск, проспект Ленина, 40, Томский государственный университет
систем управления и радиоэлектроники

artur.a.mitsel@tusur.ru

Статья посвящена сравнительному анализу алгоритма роя частиц, генетического алгоритма и предложенного автором эвристического алгоритма поиска минимума функций многих переменных при наличии дополнительных ограничений. Алгоритмы апробировались на тестовых функциях Швепеля, Растригина, Розенброка и функции сферы. Результаты моделирования показали, что каждый из трех алгоритмов имеет свои преимущества и недостатки. Для всех алгоритмов погрешность решения увеличивается с ростом количества переменных. Наибольшую погрешность вычисления оптимального решения для всех тестовых функций дал роевой алгоритм, наименьшую погрешность обеспечил генетический алгоритм. Наибольшее время счета оказалось для генетического алгоритма, наименьшее время счета обеспечил авторский эвристический алгоритм для функции сферы. Наибольшее нормированное время счета показал генетический алгоритм – от 28 для функции Швепеля до 35,8 для функции сферы. Наименьшее нормированное время расчета оптимального решения соответствует авторскому эвристическому алгоритму (от 1 для функции сферы до 1,15 для функции Швепеля). Нормированное время счета роевого алгоритма примерно одинаково для первых трех тестовых функций и составляет 30, для функции сферы – 26,7. Для прикладной задачи экономики поиска оптимального плана выпуска продукции по критерию минимальной трудоемкости наименьшую погрешность обеспечил генетический алгоритм (3,5), наибольшую погрешность дал роевой алгоритм (7,5). Погрешность авторского эвристического алгоритма составила 3,9. Время счета оптимального плана выпуска продукции по критерию минимальной трудоемкости роевым алгоритмом практически совпало с временем вычисления оптимального решения авторским алгоритмом. Наибольшее время требуется генетическому алгоритму. Для задачи максимизации прибыли погрешности и время счета примерно такие же.

Ключевые слова: методы условной оптимизации, эвристические алгоритмы, математическое моделирование, тестовые функции, функция Швепеля, функция Растригина, функция Розенброка, функция сферы, оптимальный план производственной программы

^{*} Статья получена 05 сентября 2025 г.

Исследование выполнено при финансовой поддержке Российского научного фонда (проект № 25-21-00123).

ВВЕДЕНИЕ

Задача поиска минимума функции при наличии дополнительных ограничений относится к классу задач условной оптимизации. Алгоритмы условной оптимизации функций многих переменных являются более сложными и трудоемкими по сравнению с алгоритмами безусловной оптимизации. Традиционные методы, например градиентный спуск или аналитические подходы, зачастую оказываются недостаточно эффективными при наличии сложных многоэкстремальных ландшафтов. Эвристические методы поиска минимума функций основаны на приближенных алгоритмах, которые ищут решения, приближающиеся к глобальному или локальному минимуму, используя различные стратегии поиска. Рассмотрим вкратце состояние проблемы применительно к решению прикладных задач. Исследование эвристических методов применительно к решению прикладных задач имеет обширную теоретическую и прикладную базу. В работах [1, 2] заложены основы генетических алгоритмов, которые впоследствии нашли применение в оптимизации портфельных инвестиций и управлении запасами [3, 4]. Методы роя частиц (PSO – particle swarm optimization) [5–7] активно используются для решения задач прогнозирования рыночных тенденций и калибровки моделей для решения прикладных задач.

Монография [8] посвящена использованию эвристических методов оптимизации, в частности алгоритма «порогового принятия» (*Threshold Accepting*), для решения сложных задач в статистике и эконометрике. В работе [9] показано применение эволюционных алгоритмов, генетических алгоритмов и других методов оптимизации в инженерии, бизнесе, экономике и финансах. Работа [10] посвящена применению генетического программирования для прогнозирования состояния предприятий, а в работе [11] авторы применили бинарный роевой алгоритм для ребалансировки портфелей на фондовом рынке. Пример использования эвристического алгоритма для решения оптимизационной задачи в аграрной экономике описан в работе [12]. Авторы работы [13] предложили новые технологии решения задачи разбиения, использующие математические методы, в которых заложены принципы природных механизмов принятия решений. Задача разбиения представляется в виде адаптивной системы на основе интеграции эволюционного и муравьиного подходов к поиску решения. В работе [14] рассмотрены эволюционные алгоритмы (*Simulated Annealing*, *Threshold Accepting*), нейронные сети, генетические алгоритмы, *Tabu Search* и гибридные методы и показано, почему стандартные численные методы часто не работают и как эвристики помогают в оценке и моделировании.

Решению задач условной оптимизации посвящены работы [15, 16]. Так, в работе [15] рассматриваются гибридные методы, сочетающие классические подходы (метод штрафных функций, теория множителей Лагранжа) с метаэвристическими алгоритмами. Приведен пример схемы на основе метода роя частиц и метода множителей Лагранжа с добавками. В работе [16] изложены результаты решения многомерной задачи условной оптимизации четырьмя методами: классическим градиентным спуском, точным аналитическим решением системы уравнений, генетическим алгоритмом, а также методом «подтягивания к среднему». Необходимость постановки данной задачи возникает при решении систем линейных алгебраических уравнений (СЛАУ), построении регрессионных моделей, обучении искусственных нейронных сетей.

Настоящая статья посвящена сравнительному анализу алгоритма роя частиц, генетического алгоритма и предложенного автором эвристического алгоритма поиска минимума функций многих переменных на тестовых функциях.

В качестве примера приводятся результаты оптимизации годовой производственной программы предприятия машиностроительного предприятия. Обсуждаются достоинства и недостатки рассмотренных эвристических алгоритмов.

1. КРАТКОЕ ОПИСАНИЕ АЛГОРИТМОВ

1.1. МЕТОД РОЯ ЧАСТИЦ

Роевой алгоритм широко используется для решения оптимизационных задач в различных областях. Оптимизация методом роя частиц (PSO), предложенная в работах [17, 18], основана на модели поведения птиц в стае и рыб в косяке. В PSO группа частиц (например, стая птиц) выполняет процесс поиска в пространстве решений. Каждая частица после смещения в пространстве решений делится своей текущей наилучшей позицией с одной или несколькими частицами роя. В результате весь рой (все решения) приближается к оптимуму целевой функции [19–21].

Этапы роевого алгоритма с учетом его модификации для решения задачи поиска минимума при наличии дополнительных ограничений.

Этап 1. Задание начальных координат частиц роя:

$$x_{ij}^0 \sim U(l_i, u_i), \quad i = 1, \dots, n; \quad j = 1, \dots, K.$$

Здесь $U(l_i, u_i)$ – равномерно распределенная случайная величина в диапазоне $[l_i, u_i]$; l_i, u_i – границы пространства поиска.

При наличии дополнительных ограничений частицы выбираются из условия

$$a \leq Ax^j \leq b, \quad j = 1, \dots, K. \quad (1)$$

Здесь A – матрица ограничений размерностью $m \times n$; a, b – векторы левой и правой части ограничений размерностью m ; x^j – координаты j -й частицы.

Этап 2. Задание начальных интервалов изменения координат частиц роя:

$$v_{ij}^0 \sim U(v_{l_i}, v_{u_i}), \quad i = 1, \dots, n; \quad j = 1, \dots, K.$$

Здесь $v_{l_i} = -(u_i - l_i)$, $v_{u_i} = (u_i - l_i)$.

При наличии дополнительных ограничений частицы выбираются из условия

$$a \leq A(x^j + v^j) \leq b, \quad j = 1, \dots, K,$$

где v^j – интервал изменения координат j -й частицы.

Этап 3. Выбор лучшей частицы в рое на итерации t :

$$p^t = \arg \left(\min_{j=1, \dots, N} f(x^j) \right).$$

В конце каждой итерации выбирается наилучшее решение g^t .

Этап 4. Обновление интервалов изменения координат частиц на итерации t :

$$v_{ij}^t = wv_{ij}^{t-1} + c_1 U(0, 1) \left(p_i^t - x_{ij}^t \right) + c_2 U(0, 1) \left(g_i^t - x_{ij}^t \right),$$

$$i = 1, \dots, n; \quad j = 1, \dots, K.$$

Здесь w, c_1, c_2 – параметры алгоритма.

Этап 5. Обновление координат частиц на итерации t :

$$x_{ij}^t = x_{ij}^{t-1} + v_{ij}^t, \quad i = 1, \dots, n; \quad j = 1, \dots, K.$$

При наличии дополнительных ограничений частицы выбираются из условия

$$a \leq Ax^j \leq b, \quad j = 1, \dots, K.$$

В качестве критерия остановки используется заданная величина погрешности ε :

$$\|x^t - x^{t-1}\| \leq \varepsilon, \quad t = 1, \dots, T. \quad (2)$$

В формуле (2) переменная t означает номер итерации; T – время поиска корма.

1.2. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Генетический алгоритм (ГА) [22–25] для решения задачи многомерной оптимизации включает следующие шаги с учетом дополнительных ограничений.

1. Создание начальной популяции $x_{i,k}^0$, $k = 1, \dots, K$; $i = 1, \dots, n$ из K (особей). Здесь n – количество переменных. Для этого случайным образом генерируется K целых чисел из интервала $[0; 2^L - 1]$, где L – количество битов.

2. Вещественное кодирование переменных по формулам:

$$Z = (2^L - 1);$$

$$x_i = l_i + U(1, Z) \frac{u_i - l_i}{Z}; \quad i = 1, \dots, n,$$

где l_i , u_i – границы интервала i -й переменной; $U(1, Z)$ – равномерно распределенная случайная величина, принимающая целые значения в диапазоне $[1, Z]$.

3. Подготовка хромосом x^j , $j = 1, \dots, K$, удовлетворяющих ограничениям (1).

4. Оценка степени приспособленности каждой особи (рассчитать значения целевой функции $f_k(x) = f(x^k)$, $x^k = (x_{1,k}, x_{2,k}, \dots, x_{n,k})^T$).

5. Выбор K родителей из популяции при помощи рулетки (вероятность выбора родителя должна зависеть от степени его приспособленности). Веса рассчитываются по формуле

$$w_j = \frac{|f(x^j)|}{\sum_{k=1}^K |f(x^k)|}, \quad j = 1, \dots, K.$$

При поиске минимума функции вероятности можно вычислять по формуле

$$p_k = \frac{1 - w_k}{\sum_{j=1}^K (1 - w_j)}, \quad k = 1, \dots, K.$$

6. Выбор из родительского пула пары родителей для репродукции. При помощи оператора кроссинговера получают потомка. Создается случайный потомок в пределах гиперпрямоугольника, определяемого точками родителей.

Пусть выбраны родительские хромосомы x_i^1 и x_i^2 . Тогда i -я хромосома потомка в однопараметрическом операторе смешения определяется случайно из интервала $[x_i^1 - \alpha D_i, x_i^2 + \alpha D_i]$, где $D_i = x_i^2 - x_i^1$, при этом предполагается $x_i^1 < x_i^2$. Результирующие потомки определяются следующим образом:

$$y_i^1 = (1 - \mu_i)x_i^1 + \mu_i x_i^2; \quad y_i^2 = (1 - \mu_i)x_i^2 + \mu_i x_i^1,$$

где $\mu_i = (1 + 2\alpha)r_i - \alpha$, r_i – случайное число из интервала $[0, 1]$, $\alpha \in [0, 1]$ – параметр алгоритма.

7. Производится мутация. Этот оператор используется для улучшения тонкой настройки системы. Это реализуется в хромосоме с наименьшей вероятностью (плохой хромосоме). Будем использовать экспоненциальную мутацию [25]. Мутация k -го гена происходит следующим образом:

$$y_k = (1 - \tilde{g})l_k + \tilde{g}u_k,$$

где

$$\tilde{g} = \begin{cases} g - g \left(\frac{g-r}{g} \right)^b, & \text{если } r < g, \\ g, & \text{если } r = g, \\ g + (1-g) \left(\frac{r-g}{1-g} \right)^b, & \text{если } r > g. \end{cases} \quad g = \frac{x_k - l_k}{u_k - x_k},$$

Здесь r – случайное число из интервала $[0, 1]$.

8. Выполняется элитарный отбор особей в новую популяцию. В новую популяцию отбираются лучшие особи из родительского пула и лучшие особи из пула потомков, удовлетворяющие ограничениям (1).

В качестве критерия остановки используется заданная величина погрешности ε :

$$\|x^t - x^{t-1}\| \leq \varepsilon, \quad t = 2, \dots, N.$$

Здесь переменная t означает номер поколения, N – количество поколений.

1.3. АВТОРСКИЙ ЭВРИСТИЧЕСКИЙ АЛГОРИТМ

Шаг 1. Инициализация начального решения

$$x_i^0 = l_i + r_i(u_i - l_i), \quad i = 1, \dots, n,$$

где n – количество переменных; l_i, u_i – границы интервала i -й переменной; r_i – случайное число из интервала $[0, 1]$.

Шаг 2. Формирование начального массива решений размером K :

$$x_{ij} = l_i + r_i(u_i - l_i), \quad i = 1, \dots, n; \quad j = 1, \dots, K,$$

удовлетворяющего ограничениям (1).

Шаг 3. Выбор лучшего решения X^t на каждом шаге цикла из условия

$$x^t = \arg \left[\min_{1 \leq j \leq K} f(x_t^j) \right], \quad t = 1, \dots, N,$$

и вычисление кумулятивного решения $gm^t = gm^{t-1} + x^t$, $t = 2, \dots, N$. Здесь $gm^1 = x^1$.

Шаг 4. Проверка условия завершения поиска:

$$\left\| \frac{gm^t}{t} - \frac{gm^{t-1}}{t-1} \right\| \leq \varepsilon, \quad t = 2, \dots, N.$$

Если выполняется это условие, то поиск закончен и в качестве решения берем

$$xm = \frac{gm^t}{t}. \quad (3)$$

Далее приведены результаты моделирования.

2. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ

Сравнительный анализ алгоритмов поиска минимума функций выполнялся на следующих тестовых функциях:

1) функция Швевеля

$$f(x) = 420,9687n + \sum_{i=1}^n \left[x_i \sin(\sqrt{|x_i|}) \right]; \quad x_i \in [500; 800];$$

2) функция Растригина

$$f(x) = 10n + \sum_{i=1}^n \left[x_i^2 - A \cos(2\pi x_i) \right]; \quad x_i \in [3; 15];$$

3) функция Розенброка

$$f(x) = \sum_{i=1}^{n-1} \left[(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2 \right]; \quad x_i \in [3; 15];$$

4) функция сферы

$$f(x) = \sum_{i=1}^n x_i^2; \quad x_i \in [3; 15].$$

Для моделирования задачи условной оптимизации была создана матрица ограничений со случайными элементами

$$A_{ji} = [U(10)], \quad i = 1, \dots, n; \quad j = 1, \dots, m,$$

где $U(10)$ – равномерно распределенная случайная величина в интервале $(0; 10)$; n – количество переменных; m – количество ограничений; $[.]$ – целая часть числа.

Векторы ограничений $b1$ и $b2$ вычислялись по формулам:

$$b1_j = \sum_{i=1}^n A_{ji} l_i; \quad b2_j = \sum_{i=1}^n A_{ji} u_i; \quad j = 1, \dots, m; \quad j = 1, \dots, m.$$

Здесь $l_i, u_i, i = 1, \dots, n$ – границы интервалов изменения переменных.

При моделировании задачи условной оптимизации количество ограничений задавалось равным $m = 4$. Расчеты выполнялись в пакете Mathcad для различного количества переменных n (от 10 до 30) с шагом 5.

Моделирование проводилось для следующих исходных данных.

Параметры роевого алгоритма: $K = 100$ – количество частиц в стае; $N = 300$ – количество итераций (размер цикла); остальные параметры: $w = 0,6$; $c1 = 0,05$; $c2 = 0,2$.

Параметры генетического алгоритма: $K = 100$ – размер популяции; $L = 10$ – длина хромосомы; $N = 300$ – количество поколений; остальные параметры: $\alpha = 0,5$; $\beta = 0,5$.

Параметры эвристического алгоритма: $K = 100$ – количество решений; $N = 300$ – количество итераций (размер цикла).

Погрешность решения (условие завершения поиска решения) для всех алгоритмов задавалась равной $\epsilon = 0,001$.

В процессе моделирования рассчитывалась погрешность решения dm по формуле

$$dm = \sqrt{\frac{1}{n} \sum_{i=1}^n (xm_i - xT_i)^2}, \quad (4)$$

где xm – приближенное решение; xT – точное решение; n – количество переменных.

На рис. 1а – 1г приведены результаты вычисления оптимальных решений для различных тестовых функций.

Видно, что с увеличением количества переменных погрешность в среднем растет. Наибольшую погрешность для всех тестовых функций показал роевой алгоритм. Так для функции Швепеля погрешность роевого алгоритма возрастает от 30 для $n = 10$ до 60 для $n = 30$. Для других функций эта погрешность меньше и составляет от 3,5 для $n = 10$ до 5,5 для $n = 30$ для функции Растригина и функции сферы, а для функции Розенброка – от 4 для $n = 10$ до 6,6 для $n = 30$.

Погрешность генетического алгоритма наименьшая для всех тестовых функций и находится в пределах от 18 для $n = 10$ до 20 для $n = 30$ для функции Швепеля. Для других функций эта погрешность существенно меньше и составляет от 1,2 для $n = 10$ до 2,8 для $n = 30$ для функции Растригина и функции сферы соответственно, для функции Розенброка – от 2 для $n = 10$ до 3,9 для $n = 30$.



Рис. 1а. Погрешности вычисления dm для функции Швепеля

Fig. 1a. Errors in calculating dm for the Schwefel function

Погрешность авторского алгоритма составила от 18 для $n = 10$ до 37 для $n = 30$ для функции Швепеля; для других функций эта погрешность составила от 2 для $n = 10$ до 4,5 для $n = 30$.

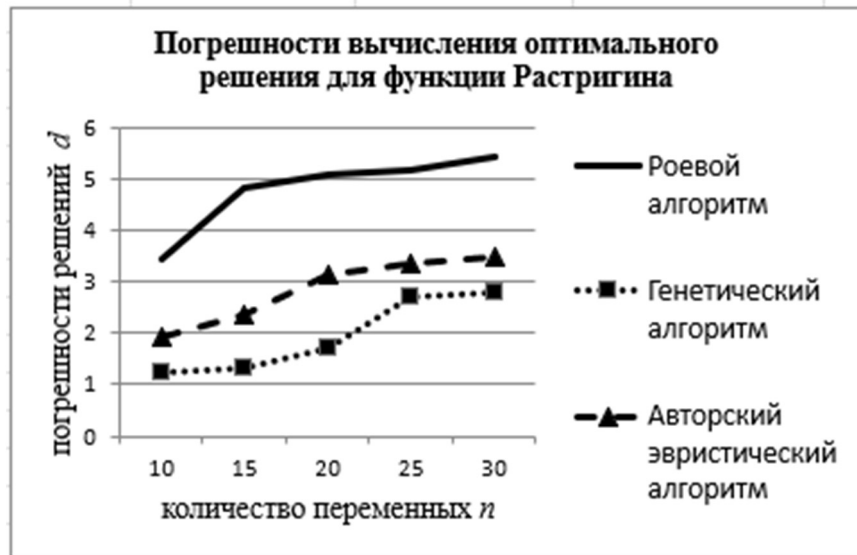


Рис. 1б. Погрешности вычисления dm для функции Растригина

Fig. 1b. Errors in calculating dm for the Rastrigin function



Рис. 1в. Погрешности вычисления dm для функции Розенброка

Fig. 1c. Errors in calculating dm for the Rosenbrock function



Рис. 1г. Погрешности вычисления dm для функции сферы

Fig. 1d. Errors in calculating dm for the spherical function

На рис. 2а, 2б приведены средние погрешности

$$ds = \frac{1}{5} \sum_{i=1}^5 dm_i \quad (5)$$

для трех алгоритмов поиска минимума функций. Усреднение проводилось по количеству переменных (от $n = 10$ до $n = 30$ с шагом $h = 5$). Видно, что наибольшая погрешность получена для функции Швепеля – от 20 для генетического алгоритма до 50 для роевого алгоритма; для авторского алгоритма погрешность составила 30. Для остальных функций погрешность существенно меньше. Для функции Розенброка эта погрешность составила от 3 до 6; наименьшая погрешность от 2 до 5 достигнута для функции Растргина.

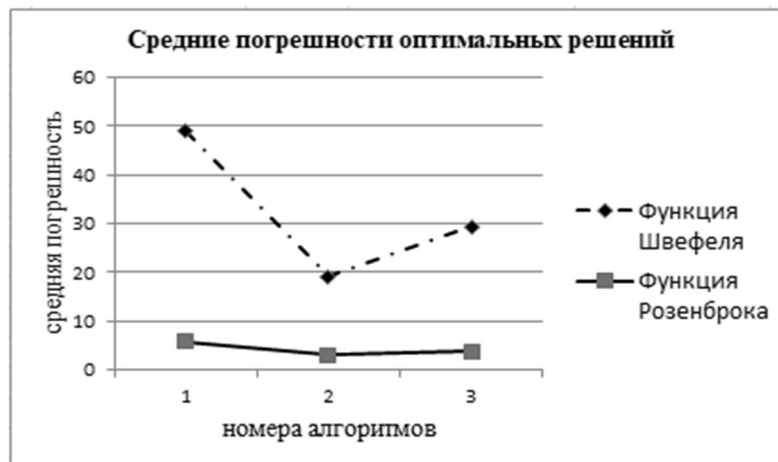


Рис. 2а. Средние погрешности вычисления оптимального решения:

1 – роевой алгоритм; 2 – генетический алгоритм; 3 – эвристический алгоритм

Fig. 2a. Average errors in calculating the optimal solution:

1 is a swarm algorithm; 2 is a genetic algorithm; 3 is a heuristic algorithm



Рис. 2б. Средние погрешности вычисления оптимального решения:

1 – роевой алгоритм; 2 – генетический алгоритм; 3 – эвристический алгоритм

Fig. 2b. Average errors in calculating the optimal solution:

1 is a swarm algorithm; 2 is a genetic algorithm; 3 is a heuristic algorithm

Представляет интерес информация о времени счета. Наименьшее время показал авторский алгоритм для функции сферы (3,27 с). В таблице приведены нормированные данные о времени счета для количества переменных $n = 30$. Нормировка проводилась на наименьшее время счета 3,27 с. Наибольшее время счета показал генетический алгоритм – от 28 для функции Швевеля до 35,8 для функции сферы. Наименьшее время расчета оптимального решения соответствует авторскому эвристическому алгоритму (от 1 для функции сферы до 1,15 для функции Швевеля). Время счета роевого алгоритма примерно одинаково для первых трех тестовых функций и составляет 30, для функции сферы – 26,7.

Нормированное время вычисления оптимального решения для $n = 30$

The normalized time for calculating the optimal solution for $n = 30$

Алгоритмы	Функция Швевеля	Функция Растригина	Функция Розенброка	Сферическая функция
Роевой	29,1	29,9	28,8	26,7
Генетический	28,0	35,5	33,0	35,8
Авторский	1,15	1,07	1,13	1

3. ОПТИМИЗАЦИЯ ГОДОВОЙ ПРОИЗВОДСТВЕННОЙ ПРОГРАММЫ МАШИНОСТРОИТЕЛЬНОГО ПРЕДПРИЯТИЯ

Под производственной программой понимается номенклатура и объем выпуска продукции. Ее формирование – одна из центральных задач текущего планирования. Каждое предприятие заинтересовано в формировании оптимальной производственной программы. Под ней понимают программу, которая в наибольшей степени учитывает запросы потребителей, отвечает структуре ресурсов предприятия и обеспечивает наилучшие результаты его деятельности по принятым критериям. В качестве критериев рассматриваются трудоемкость и прибыль. Модель задачи описана в работе Мицеля А.А. и Зединой М.А. [26], исходные данные взяты из работы Лихачева Л.Н. и др. [27]. Приведем краткое описание экономико-математической модели.

Машиностроительное предприятие выпускает 24 вида продукции. Необходимо сформировать такой план производства, который обеспечивал бы максимум прибыли при минимальной трудоемкости программы. Решение такой задачи не является оптимальным ни для одного из частных критериев. В работе [26] решается двухкритериальная задача методом справедливого компромисса. В настоящей статье будем решать две однокритериальные задачи: по критерию минимизации трудоемкости и по критерию максимизации прибыли.

Известны нормы расхода материалов первого и второго видов на выпуск единицы продукции и общая трудоемкость единицы изделия, цены (оптовые) на единицу выпускаемой продукции, себестоимость и прибыль единицы изделия, минимальное и максимальное количество каждого изделия в производственной программе, определяемое спросом, допустимая годовая трудоемкость, возможный объем расхода ресурсов на производство продукции, контрольное (предельное) значение себестоимости, а также ограничения по объему реализуемой продукции и контрольному значению прибыли.

Модель по трудоемкости имеет следующий вид:

$$\begin{aligned}
 ftr(x) &= \sum_{i=1}^{24} tr_i x_i \rightarrow \min, \\
 \left\{ \begin{array}{l} \sum_{i=1}^{24} tr_i x_i \leq Tr; \quad \sum_{i=1}^{24} m1_i x_i \leq M1; \quad \sum_{i=1}^{24} m2_i x_i \leq M2, \\ \sum_{i=1}^{24} c_i x_i \leq C; \quad \sum_{i=1}^{24} a_i x_i \geq Va; \quad \sum_{i=1}^{24} pr_i x_i \geq Pr, \\ Low_i \leq x_i \leq Upp_i, \quad i \in \overline{1; 24}. \end{array} \right. \quad (6)
 \end{aligned}$$

В модели по прибыли в качестве критерия используется

$$ptr(x) = \sum_{i=1}^{24} pr_i x_i \rightarrow \max.$$

Здесь x_i – годовое количество i -го изделия в производственной программе (целочисленная величина); tr_i – общая трудоемкость единицы i -го изделия; $m1_i$ – норма расхода 1-го лимитирующего вида материала на единицу i -го изделия; $m2_i$ – норма расхода 2-го лимитирующего вида материала на единицу i -го изделия; c_i – себестоимость на единицу i -го изделия; a_i – цена (оптовая) единицы i -го изделия; pr_i – прибыль единицы i -го изделия; Tr – максимально допустимая годовая трудоемкость производственной программы; $M1$, $M2$ – максимально возможный объем расхода на производственную программу 1-го и 2-го лимитирующего вида материалов, обусловленный возможностями его поставки и имеющимися запасами; C – контрольное (предельное) значение по себестоимости; Va – контрольное значение по объему производства продукции; Pg – контрольное значение по прибыли; Low_i – минимальное количество i -го изделия; Upp_i – максимальное количество i -го изделия (определяется спросом).

Первая группа ограничений (первые три неравенства в системе (6) представляет собой ограничения на ресурсы, вторая группа ограничений (четвертое, пятое и шестое неравенства системы (6) – это ограничения по основным показателям деятельности предприятия, третья группа ограничений (последние соотношения в системе (6) – это ограничения по спросу (сбыту). Исходные данные взяты из работы Лихачева Л.Н. и др. [30] (1999).

На рис. 3 приведены погрешности решения задачи поиска оптимального плана производственной программы по критерию минимальной трудоемкости. Наибольшую погрешность ($db = 7,5$) дает роевой алгоритм, наименьшую – генетический алгоритм ($db = 3,5$). Погрешность эвристического алгоритма составила 3,9.

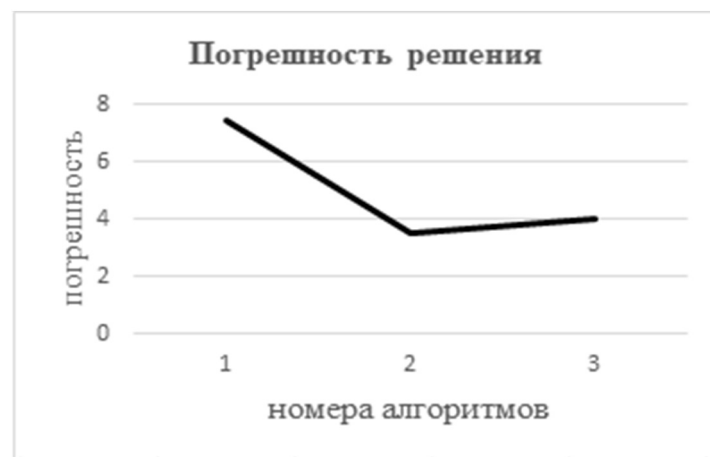


Рис. 3. Погрешности вычисления оптимального плана выпуска продукции по критерию минимальной трудоемкости:

1 – роевой алгоритм; 2 – генетический алгоритм; 3 – авторский эвристический алгоритм

Fig. 3. Errors in calculating the optimal production plan based on the minimum labor intensity criterion:

1 is a swarm algorithm; 2 is a genetic algorithm; 3 is the author's heuristic algorithm

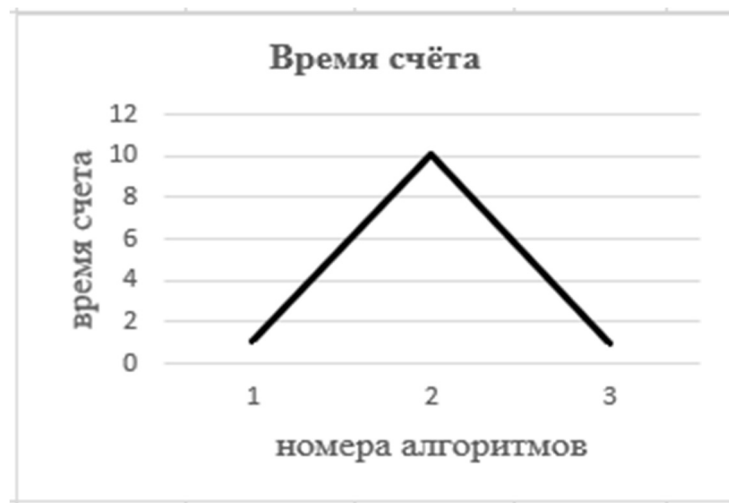


Рис. 4. Время (нормированное) вычисления оптимального плана выпуска продукции по критерию минимальной трудоемкости:
1 – роевой алгоритм; 2 – генетический алгоритм; 3 – эвристический алгоритм

Fig. 4. Time (normalized) for calculating the optimal production plan based on the minimum labor intensity criterion:
1 is a swarm algorithm; 2 is a genetic algorithm; 3 is the author's heuristic algorithm

На рис. 4 приведено время счета (нормированное на минимальное время, соответствующее авторскому алгоритму) оптимального плана выпуска продукции. Как и следовало ожидать, наибольшее время требуется генетическому алгоритму. Время счета роевого алгоритма практически совпало с временем вычисления оптимального решения авторским алгоритмом. Для задачи максимизации прибыли погрешности и время счета примерно такие же.

ЗАКЛЮЧЕНИЕ

Каждый из трех алгоритмов имеет свои преимущества и недостатки. Для всех алгоритмов погрешность решения увеличивается с ростом количества переменных. Для всех тестовых функций наибольшую погрешность дает роевой алгоритм: для функции Швепеля погрешность роевого алгоритма составила 30 для $n = 10$ и увеличилась до 60 для $n = 30$. Для других функций эта погрешность меньше и составляет от 3,5 для $n = 10$ до 5,5 для $n = 30$ для функции Растригина и функции сферы, а для функции Розенброка от 4 для $n = 10$ до 6,6 для $n = 30$.

Погрешность генетического алгоритма оказалась наименьшей для всех тестовых функций и находилась в пределах от 18 для $n = 10$ до 20 для $n = 30$ для функции Швепеля. Для других функций эта погрешность существенно меньше: от 1,2 для $n = 10$ до 2,8 для $n = 30$ для функции Растригина и функции сферы соответственно, для функции Розенброка – от 2 для $n = 10$ до 3,9 для $n = 30$.

Погрешность авторского алгоритма меньше погрешности роевого алгоритма для всех тестовых функций и составила от 18 для $n = 10$ до 37 для $n = 30$

для функции Швепеля; для других функций это погрешность была от 2 для $n = 10$ до 4,5 для $n = 30$.

Наибольшее время счета вычисления оптимального решения для всех тестовых функций дал генетический алгоритм, а наименьшее время – авторский алгоритм. Так, для функции Швепеля время счета генетическим алгоритмом для $n = 30$ составило 28, а для функции сферы – 36. Время счета роевым алгоритмом для всех функций примерно одинаково и находится в пределах 29–30.

Для прикладной задачи экономики поиска оптимального плана выпуска продукции по критерию минимальной трудоемкости наименьшую погрешность обеспечил генетический алгоритм (3,5 количества изделий), наибольшую погрешность дал роевой алгоритм (7,5 количества изделий). Для авторского эвристического алгоритма погрешность составила 3,9 количества изделий. Наибольшее время вычисления показал генетический алгоритм (нормированное минимальное время авторского алгоритма составило 10), время счета роевого алгоритма близко к времени счета авторским алгоритмом (1,08).

СПИСОК ЛИТЕРАТУРЫ

1. Holland J.H. Adaptation in natural and artificial systems. – Ann Arbor: University of Michigan Press, 1975.
2. Goldberg D.E. Genetic algorithms in search, optimization, and machine learning. – Addison-Wesley, 1989.
3. Drake A.E., Marks R.E. Genetic algorithms in economics and finance: Forecasting stock market prices and foreign exchange – A review // Chen S.H., ed. Genetic algorithms and genetic programming in computational finance. – Springer, 2002. – P. 29–54. – DOI: 10.1007/978-1-4615-0835-9_2.
4. Liu Z., Nishi T. Data-driven evolutionary computation for service constrained inventory optimization in multi-echelon supply chains // Complex & Intelligent Systems. – 2023. – Vol. 10 (1). – P. 825–846. – DOI: 10.1007/s40747-023-01179-0.
5. Kennedy J., Eberhart R. Particle swarm optimization // Proceedings of ICNN'95 – International Conference on Neural Networks. – 1995. – Vol. 4. – P. 1942–1948. – DOI: 10.1109/CNN.1995.488968.
6. Poli R., Kennedy J., Blackwell T. Particle swarm optimization: An overview // Swarm Intelligence. – 2007. – Vol. 1 (1). – P. 33–57. – DOI: 10.1007/s11721-007-0002-0.
7. Natural computing in computational finance. Vol. 3 / A. Brabazon, M. O'Neill, D.G. Marin-gers. – Springer, 2010.
8. Winker P. Optimization heuristics in econometrics: Applications of threshold accepting. – Wiley, 2001. – 333 p. – ISBN 0471856312.
9. Vasant P.V. Meta-heuristics optimization algorithms in engineering, business, economics, and finance. – Hershey, PA: Information Science Reference, 2013. – DOI: 10.4018/978-1-4666-2086-5.
10. Гречин С.Б., Трифонов Ю.В. Применение генетического программирования для прогнозирования состояния предприятий // Вестник Нижегородского университета им. Н.И. Лобачевского. – 2007. – № 5. – С. 130–133.
11. Пантелеев А.В., Милютин С.А. Оптимизация инвестиционного портфеля бинарным методом роя пчел // Моделирование и анализ данных. – 2024. – Т. 14, № 3. – С. 87–104. – DOI: 10.17759/mda.2024140305.
12. Ефремов А., Белоцкая Ю. Эвристический алгоритм оптимизации выполнения комплекса механизированных работ в полеводстве в условиях цифровизации // Аграрная экономика. – 2023. – № 4. – С. 65–81. – DOI: 10.29235/1818-9806-2023-10-65-81.
13. Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Гибридный алгоритм разбиения на основе природных механизмов принятия решений // Искусственный интеллект и принятие решений. – 2012. – № 2. – С. 3–15.
14. Gilli M., Winker P. A review of heuristic optimization methods in econometrics. – Preprint. – 2008. – 48 p. – (COMISEF Working Papers Series; WPS-001 09/09/2008).
15. Лисин А.В., Яковенко К.С. Гибридные схемы решения задач условной оптимизации с использованием метаэвристик // Математические структуры и моделирование. – 2017. – № 1 (41). – С. 88–92.
16. Крючков М.В. Сравнительный анализ некоторых алгоритмов решения многомерной задачи условной оптимизации // Вестник ИжГТУ имени М.Т. Калашникова. – 2014. – № 2 (62). – С. 153–155.

17. Казакова Е.М. Применение метода роя частиц в задачах оптимизации // Вестник КРАУНЦ. Физико-математические науки. – 2022. – Т. 39, № 2. – С. 150–174.
18. Сидорова В.Т., Остащенко А.П. Применение алгоритма роя частиц для оптимизации компенсации реактивной мощности в сетях низкого напряжения // Электрические станции. – 2024. – № 10. – С. 22–27.
19. Eberhart R., Kennedy J. A new optimizer using particle swarm theory // MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. – IEEE, 1995. – P. 39–43. – DOI: 10.1109/MHS.1995.494215.
20. Cleghorn C.W., Engelbrecht A.P. Particle swarm convergence: An empirical investigation // 2014 IEEE Congress on Evolutionary Computation (CEC). – IEEE, 2014. – P. 2524–2530. – DOI: 10.1109/CEC.2014.6900439.
21. Levy flight distribution: a new metaheuristic algorithm for solving engineering optimization problems / E.H. Houssein, M.R. Saad, F.A. Hashim, H. Shaban, M. Hassaballah // Engineering Applications of Artificial Intelligence. – 2020. – Vol. 94. – P. 103–131. – DOI: 10.1016/j.engappai.2020.103731.
22. Батищев Д.И. Генетические алгоритмы решения экстремальных задач. – Н. Новгород: Нижегород. гос. ун-т, 1995. – 62 с.
23. Панченко Т.В. Генетические алгоритмы. – Астрахань: Астрахан. ун-т, 2007. – 87 с.
24. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2006. – 320 с.
25. Abdoun O., Abouchabaka J., Tajani C. Analyzing the performance of mutation operators to solve the travelling salesman problem // International Journal of Emerging Sciences. – 2012. – Vol. 2 (1). – P. 61–77.
26. Мицель А.А., Зедина М.А. Оптимизация годовой производственной программы предприятия методом справедливого компромисса // Экономический анализ: теория и практика. – 2012. – № 41 (296). – С. 54–60.
27. Практикум по применению экономико-математических моделей для формирования продуктовой (производственной) программы коммерческой организации / Л.Н. Лихачева, И.Н. Щелина, О.С. Воищева, С.С. Щекунских. – Воронеж: ВГУ, 1999.

Мицель Артур Александрович, доктор технических наук, профессор, Томский государственный университет систем управления и радиоэлектроники, каф. АСУ, профессор. Основное направление научных исследований – математическое моделирование, оптимизация, искусственный интеллект. Имеет более 70 печатных работ. E-mail: artur.a.mitsel@tusur.ru

Mitsel Artur A., D.Sc. (Eng.), professor, Tomsk State University of Control Systems and Radioelectronics, Department of ACS, Professor. The main field of his scientific research is mathematical modeling, optimization, and artificial intelligence. He has more than 70 published works. E-mail artur.a.mitsel@tusur.ru

DOI: 10.17212/2782-2001-2026-2-115-132

Heuristic methods for finding the minimum of a function in the presence of additional constraints*

A.A. MITSEL

Tomsk State University of Control Systems and Radioelectronics, 40 Lenin Avenue, Tomsk, 634034, Russian Federation

artur.a.mitsel@tusur.ru

Abstract

The article is devoted to a comparative analysis of the particle swarm algorithm, the genetic algorithm, and the heuristic algorithm proposed by the author for finding the minimum of functions of many variables with additional constraints. The algorithms were tested on the test functions of Schwefel,

* Received 05 September 2025.

The research was funded by the Russian Science Foundation (project No. 25-21-00123).

Rastrigin, Rosenbrock, and the sphere function. The simulation results showed that each of the three algorithms has its own advantages and disadvantages. For all algorithms, the error in the solution increases with the number of variables. The particle swarm algorithm had the highest error in calculating the optimal solution for all test functions, while the genetic algorithm had the lowest error. The longest calculation time was for the genetic algorithm, and the shortest calculation time was for the author's heuristic algorithm for the sphere function. The genetic algorithm showed the longest normalized calculation time, ranging from 28 for the Schwefel function to 35.8 for the sphere function. The lowest normalized calculation time of the optimal solution corresponds to the author's heuristic algorithm (from 1 for the sphere function to 1.15 for the Schwefel function). The normalized calculation time of the swarm algorithm is approximately the same for the first three test functions and is equal to 30, while it is 26.7 for the sphere function. For the applied economic problem of finding the optimal production plan based on the minimum labor intensity criterion, the genetic algorithm provided the lowest error (3.5), while the swarm algorithm provided the highest error (7.5). The error of the author's heuristic algorithm was 3.9. The time of calculation of the optimal production plan by the criterion of minimum labor input by the swarm algorithm practically coincided with the time of calculation of the optimal solution by the author's algorithm. The greatest time is required by the genetic algorithm. For the problem of maximizing the profit, the errors and the time of calculation are approximately the same.

Keywords: constrained optimization methods, heuristic algorithms, mathematical modeling, test functions, Schwefel function, Rastrigin function, Rosenbrock function, sphere function, optimal production program plan

REFERENCES

1. Holland J.H. *Adaptation in natural and artificial systems*. Ann Arbor, University of Michigan Press, 1975.
2. Goldberg D.E. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, 1989.
3. Drake A.E., Marks R.E. Genetic algorithms in economics and finance: Forecasting stock market prices and foreign exchange – A review. Chen S.H., ed. *Genetic algorithms and genetic programming in computational finance*. Springer, 2002, pp. 29–54. DOI: 10.1007/978-1-4615-0835-9_2.
4. Liu Z., Nishi T. Data-driven evolutionary computation for service constrained inventory optimization in multi-echelon supply chains. *Complex & Intelligent Systems*, 2023, vol. 10 (1), pp. 825–846. DOI: 10.1007/s40747-023-01179-0.
5. Kennedy J., Eberhart R. Particle swarm optimization. *Proceedings of ICNN'95 – International Conference on Neural Networks*, 1995, vol. 4, pp. 1942–1948. DOI: 10.1109/ICNN.1995.488968.
6. Poli R., Kennedy J., Blackwell T. Particle swarm optimization: An overview. *Swarm Intelligence*, 2007, vol. 1 (1), pp. 33–57. DOI: 10.1007/s11721-007-0002-0.
7. Brabazon A., O'Neill M., Maringereds D.G., eds. *Natural computing in computational finance*. Vol. 3. Springer, 2010.
8. Winker P. *Optimization heuristics in econometrics: Applications of threshold accepting*. Wiley, 2001. 333 p. ISBN 0471856312.
9. Vasant P.V. *Meta-heuristics optimization algorithms in engineering, business, economics, and finance*. Hershey, PA, Information Science Reference, 2013. DOI: 10.4018/978-1-4666-2086-5.
10. Grechin S.B., Trifonov Yu.V. Primenenie geneticheskogo programmirovaniya dlya prognozirovaniya sostoyaniya predpriyatii [The use of genetic programming for forecasting company performance]. *Vestnik Nizhegorodskogo universiteta im. N.I. Lobachevskogo = Vestnik of Lobachevsky State University of Nizhni Novgorod*, 2007, no. 5, pp. 130–133.
11. Panteleev A.V., Milyutina S.A. Optimizatsiya investitsionnogo portfelya binarnym metodom roya pchel [Investment portfolio optimization by binary bee swarm method]. *Modelirovanie i analiz dannykh = Modeling and Data Analysis*, 2024, vol. 14, no. 3, pp. 87–104. DOI: 10.17759/mda.2024140305.
12. Efremov A., Belotskaya Yu. Evristicheskii algoritm optimizatsii vypolneniya kompleksa mekhanizirovannykh rabot v polevodstve v usloviyakh tsifrovizatsii [Heuristic algorithm for optimizing the implementation of a complex of mechanized works in field farming in conditions of digitalization]. *Agrarnaya ekonomika = Agrarian Economics*, 2023, no. 4, pp. 65–81. DOI: 10.29235/1818-9806-2023-10-65-81.
13. Kureichik V.M., Lebedev B.K., Lebedev O.B. Gibridnyi algoritm razbieniya na osnove prirodnykh mekhanizmov prinyatiya reshenii [A hybrid partitioning algorithm based on natural mechanisms of decision making]. *Iskusstvennyi intellekt i prinyatie reshenii = Artificial Intelligence and Decision-Making*, 2012, no. 2, pp. 3–15.
14. Gilli M., Winker P. *A review of heuristic optimization methods in econometrics*. Preprint. 2008. 48 p. *COMISEF Working Papers Series*; WPS-001 09/09/2008.

15. Lisin A.V., Yakovenko K.S. Gibridnye skhemy resheniya zadach uslovnoi optimizatsii s ispol'zovaniem metaevristik [Hybrid methods for solving constrained optimization problems using metaheuristics]. *Matematicheskie struktury i modelirovanie = Mathematical Structures and Modeling*, 2017, no. 1 (41), pp. 88–92.
16. Kryuchkov M.V. Sravnitel'nyi analiz nekotorykh algoritmov resheniya mnogomernoi zadachi uslovnoi optimizatsii [Comparative analysis of some algorithms for solving problems of multidimensional constrained optimization]. *Vestnik IzhGTU imeni M.T. Kalashnikova*, 2014, no. 2 (62), pp. 153–155. (In Russian).
17. Kazakova E.M. Primenenie metoda roya chastits v zadachakh optimizatsii [A concise overview of particle swarm optimization methods]. *Vestnik KRAUNTS. Fiziko-matematicheskie nauki = Bulletin KRASEC. Physical and Mathematical Sciences*, 2022, vol. 39, no. 2, pp. 150–174. (In Russian).
18. Sidorova V.T., Ostashenkov A.P. Primenenie algoritma roya chastits dlya optimizatsii kompensatsii reaktivnoi moshchnosti v setyakh nizkogo napryazheniya [Application of the particle swarm algorithm for optimizing reactive power compensation in low-voltage networks]. *Elektricheskie stantsii = Power Plants*, 2024, no. 10, pp. 22–27.
19. Eberhart R., Kennedy J. A new optimizer using particle swarm theory. *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. IEEE, 1995, pp. 39–43. DOI: 10.1109/MHS.1995.494215.
20. Clegghorn C.W., Engelbrecht A.P. Particle swarm convergence: An empirical investigation. *2014 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2014, pp. 2524–2530. DOI: 10.1109/CEC.2014.6900439.
21. Houssein E.H., Saad M.R., Hashim F.A., Shaban H., Hassaballah M. Levy flight distribution: a new metaheuristic algorithm for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, 2020, vol. 94, pp. 103–131. DOI: 10.1016/j.engappai.2020.103731.
22. Batishchev D.I. *Geneticheskie algoritmy resheniya ekstremal'nykh zadach* [Genetic algorithms for solving extremal problems]. Nizhny Novgorod, Nizhny Novgorod State University Publ., 1995. 62 p.
23. Panchenko T.V. *Geneticheskie algoritmy* [Genetic algorithms]. Astrakhan, Astrakhan University Publ., 2007. 87 p.
24. Gladkov L.A., Kureichik V.V., Kureichik V.M. *Geneticheskie algoritmy* [Genetic algorithms]. Moscow, Fizmatlit Publ., 2006. 320 p.
25. Abdoun O., Abouchabaka J., Tajani C. Analyzing the performance of mutation operators to solve the travelling salesman problem. *International Journal of Emerging Sciences*, 2012, vol. 2 (1), pp. 61–77.
26. Mizel A.A., Zedina M.A. Optimizatsiya godovoi proizvodstvennoi programmy predpriyatiya metodom spravedlivogo kompromissa [Optimization of annual production program of enterprise by fair compromise method]. *Ekonomicheskii analiz: teoriya i praktika = Economic Analysis: Theory and Practice*, 2012, no. 41 (296), pp. 54–60.
27. Likhacheva L.N., Shchepina I.N., Voishcheva O.S., Shchekunskikh S.S. *Praktikum po primeniyu ekonomiko-matematicheskikh modelei dlya formirovaniya produktovoi (proizvodstvennoi) programmy kommercheskoi organizatsii* [Practical training in the application of economic and mathematical models for the formation of a product (production) program of a commercial organization]. Voronezh, VSU Publ., 1999.

Для цитирования:

Мицель А.А. Эвристические методы поиска минимума функции при наличии дополнительных ограничений // Системы анализа и обработки данных. – 2026. – № 2 (102). – С. 115–132. – DOI: 10.17212/2782-2001-2026-2-115-132.

For citation:

Mitsel A.A. Evristicheskie metody poiska minimuma funktsii pri nalichii dopolnitel'nykh ogranichenii [Heuristic methods for finding the minimum of a function in the presence of additional constraints]. *Sistemy analiza i obrabotki dannykh = Analysis and Data Processing Systems*, 2026, no. 2 (102), pp. 115–132. DOI: 10.17212/2782-2001-2026-2-115-132.