

## РЕШЕНИЕ ЗАДАЧИ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ МОДИФИЦИРОВАННЫМ АЛГОРИТМОМ ЛЕТУЧИХ МЫШЕЙ

В работе рассмотрен алгоритм летучих мышей для решения задачи глобальной оптимизации. Данный метаэвристический алгоритм относится к алгоритмам роевого интеллекта, которые интенсивно развиваются в последние годы. Целью работы является усовершенствование алгоритма летучих мышей, исследование эффективности его работы и применение для решения оптимизационных задач. Предложен модифицированный вариант канонического алгоритма, в котором для вычисления скорости движения летучих мышей применена методика алгоритма роя частиц. Проведены вычислительные эксперименты по сравнению точности и скорости сходимости канонического и модифицированного алгоритмов. Установлено, что предложенный вариант алгоритма является более эффективным при нахождении глобального минимума унимодальных и мультимодальных тестовых функций. Исследована зависимость эффективности работы модифицированного алгоритма летучих мышей от задаваемых параметров. В качестве варьируемых параметров выбраны начальные значения громкости и интенсивности звукового сигнала, испускаемого летучими мышами. Модифицированный алгоритм применен для решения практической задачи минимизации стоимости расходных материалов, необходимых для изготовления сосуда высокого давления. Проведено сопоставление полученного решения данной оптимизационной задачи с результатами других авторов, которые применяли как классические, так и популяционные алгоритмы.

**Ключевые слова:** алгоритм летучих мышей, популяционный алгоритм, роевой интеллект, глобальная оптимизация.

### НОМЕНКЛАТУРА

$A_i$  – громкость звукового сигнала;  
 $A_i^{(t+1)}$  – громкость сигнала  $i$ -ой летучей мыши на  $(t+1)$  итерации;  
 $\bar{A}(t)$  – среднее значение громкости звуковых сигналов всех летучих мышей;  
 $c_1, c_2$  – заданные константы;  
 $f(x)$  – целевая функция;  
 $F(x)$  – вспомогательная целевая функция;  
 $g_j(x)$  – ограничения-неравенства;  
 $i$  – номер летучей мыши;  
 $L$  – длина цилиндрической части сосуда высокого давления;  
 $m$  – размерность пространства;  
 $N$  – количество летучих мышей в популяции;  
 $r_i$  – интенсивность звукового сигнала;  
 $r_i^{(t+1)}$  – интенсивность сигнала  $i$ -ой летучей на  $(t+1)$  итерации;  
 $rand$  – случайное число из интервала  $[0, 1]$ ;  
 $R$  – внутренний радиус цилиндрической оболочки сосуда высокого давления;  
 $t$  – номер итерации алгоритма;  
 $tMax$  – максимальное число итераций алгоритма;  
 $T_s$  – толщина стенки цилиндра сосуда высокого давления;  
 $T_h$  – толщина сферической головки сосуда высокого давления;  
 $v_i$  – скорость летучей мыши;  
 $v_i^{(t)}$  – скорость перемещения  $i$ -ой летучей мыши на итерации  $t$ ;  
 $x_i$  – текущее положение летучей мыши;  
 $x_i^{(t)}$  – положение  $i$ -ой летучей мыши на итерации  $t$ ;

$x_{min}$  – минимальное значение координаты летучей мыши;  
 $x_{max}$  – максимальное значение координаты летучей мыши;  
 $x^*$  – точка глобального минимума;  
 $\bar{x}$  – положение летучей мыши, в котором найдено лучшее глобальное решение;  
 $\hat{x}_j$  – положение летучей мыши, в котором найдено ее лучшее решение;  
 $X$  – множество допустимых решений;  
 $\alpha$  – заданный коэффициент из интервала  $[0, 1]$ ;  
 $\beta$  – случайное число из интервала  $[0, 1]$ ;  
 $\gamma$  – заданный коэффициент;  
 $\varepsilon$  – случайное число из интервала  $[-1, 1]$ ;  
 $\mu_j$  – константы штрафа;  
 $\xi_1, \xi_2$  – случайные числа из интервала  $[0, 1]$ ;  
 $\chi$  – коэффициент сжатия;  
 $\omega_i$  – частота звукового сигнала;  
 $\omega_{min}$  – минимальное значение частоты звукового сигнала;  
 $\omega_{max}$  – максимальное значение частоты звукового сигнала.

### ВВЕДЕНИЕ

Решение многих актуальных проблем в различных прикладных и фундаментальных науках сводится к задаче глобальной оптимизации. Существующие численные методы оптимизации можно условно разделить на две группы: детерминистические и стохастические. Детерминистические методы, как правило, требуют нахождения градиента целевой функции и зависят от выбора начальных значений, при фиксированном значении которых всегда приводят к тому же результату. Однако целевая функция в задачах глобальной оптимизации может быть недифференцируемой либо разрывной, или же нелинейной либо мультимодальной, усложняя тем

самым решение задачи. Для эффективного решения подобных задач широко используют стохастические методы, которые не накладывают дополнительных ограничений на поставку задачи.

Стохастические поисковые алгоритмы начали интенсивно развиваться в 1980-х гг., в литературе их также называют метаэвристическими, популяционными, многоагентными и алгоритмами роевого интеллекта (Swarm intelligence) [1]. Данные алгоритмы предполагают одновременную обработку нескольких возможных вариантов решения оптимизационной задачи. Еще одной особенностью подобных алгоритмов является их бионическая природа. Многоагентные алгоритмы описывают динамику поведения групп различных социальных животных, птиц, насекомых и прочих агентов некоторой биологической системы, которые локально взаимодействуют между собой и с окружающей средой. Коллективное поведение группы агентов позволяет им решать различные сложные практические задачи в природе, что свидетельствует про эффективность алгоритмов роевого интеллекта, моделирующих их поведение. Поэтому актуальной является задача построения и исследования популяционных алгоритмов, а также расширение области их применения [2].

Среди многообразия популяционных алгоритмов можно выделить следующие группы: алгоритмы, инспирированные живой природой (nature-inspired algorithms); алгоритмы, инспирированные неживой природой; алгоритмы, инспирированные человеческим обществом и другие [3]. К наиболее известным алгоритмам, вдохновленным живой природой, относятся: муравьиный алгоритм (Ant Colony Optimization, 1992), метод роя частиц (Particle Swarm Optimization, 1995), бактериальная оптимизация (Bacterial Optimisation, 2002), алгоритм пчелиного роя (Bees Algorithm, 2005), алгоритм светлячков (Firefly Algorithm, 2007), поиск косяком рыб (Fish School Search, 2008), алгоритм кукушки (Cuckoo Search, 2009), алгоритм летучих мышей (Bat Algorithm, 2010) и др. Алгоритмы, инспирированные неживой природой, имитируют процессы, происходящие в физике, химии и др., в частности это: стохастический диффузионный поиск (Stochastic Diffusion Search, 1989), дифференциальная эволюция (Differential Evolution, 1997), электромагнитный алгоритм (ElectroMagnetism-like Algorithm, 2003), гармонический поиск (Harmony Search, 2008), алгоритм гравитационного поиска (Gravitational Search, 2009) и др. К алгоритмам, инспирированным человеческим обществом, принадлежат: алгоритм эволюции разума (Mind Evolutionary Computation, 1998), культурный алгоритм (Cultural Algorithm, 1994) и др.

Объектом данного исследования является процесс решения задачи глобальной оптимизации при помощи алгоритмов роевого интеллекта. Одним из новых алгоритмов данного типа является алгоритм, инспирированный поведением летучих мышей (алгоритм летучих мышей), предложенный Xin-She Yang в 2010 г. [4]. Этот алгоритм является потенциально более мощным, чем алгоритм роя частиц, генетический алгоритм, а так же гармонический поиск. Более того, гармонический поиск и алгоритм роя частиц являются особыми случаями алгоритма летучих мышей при соответствующих упрощениях [5].

Предметом исследования выступает метаэвристический алгоритм летучих мышей решения оптимизационных задач.

Целью данной работы является усовершенствование алгоритма летучих мышей, исследование эффективности его работы и применение для решения задач глобальной оптимизации.

## 1 ПОСТАНОВКА ЗАДАЧИ

Дана целевая функция  $f(x)$ , определенная на множестве допустимых решений  $X \subseteq R^m$ .

Требуется найти глобальный условный минимум функции  $f(x)$  на множестве  $X$ :

$$f(x^*) = \min_{x \in X \subseteq R^m} f(x), \quad (1)$$

где  $x = (x_1, x_2, \dots, x_m)^T$ ,  $X = \{x \mid x_i \in [x_{\min}, x_{\max}], i \in \overline{1, m}\}$ .

Задача поиска максимума функции  $f(x)$  сводится к задаче поиска минимума:

$$f(x^*) = \max_{x \in X \subseteq R^m} f(x) = -\min_{x \in X \subseteq R^m} (-f(x)).$$

## 2 ОБЗОР ЛИТЕРАТУРЫ

Алгоритм летучих мышей относится к биоинспирированным алгоритмам роевого интеллекта, который основывается на их уникальных эхолокационных возможностях. Большинство видов летучих мышей обладают совершенными средствами эхолокации, которые используются ими для обнаружения добычи и препятствий. Для этого во время полета они испускают частотно-модулированные звуковые сигналы и улавливают эхо, отраженное от окружающих объектов и предметов. В разных семействах летучих мышей параметры испускаемых звуковых импульсов различаются, определяя тем самым стратегии их охоты и поведения.

Рассматриваемый алгоритм летучих мышей в целом является достаточно эффективным, однако, как и любой другой популяционный алгоритм, он не является универсальным методом решения всех оптимизационных задач. Для усовершенствования и расширения области применения алгоритма на данный момент предложено немало его модификаций. Так, автор алгоритма в 2011 г. обобщил его для многоэкстремальных оптимизационных задач, и получил многокритериальный алгоритм летучих мышей (Multi-objective Bat Algorithm) [6]. Nakamura R. и др. в 2012 г. разработали бинарный алгоритм летучих мышей (Binary Bat Algorithm) для решения задач комбинаторной оптимизации [7]. Многие модификации алгоритма летучих мышей, получают путем гибридизации как с различными популяционными алгоритмами, так и с абсолютно другими методами. Среди таких алгоритмов, например: гибридный алгоритм летучих мышей (Hybrid Bat Algorithm), который получен Fister I. J. и др. в 2013 г. [8] в результате гибридизации с алгоритмом дифференциальной эволюции; комбинированный алгоритм летучих мышей и гармонического поиска, предложили G. Wang и L. Guo [9] и т. д.

Следует отметить, что вариации алгоритма летучих мышей применяются для решения широкого круга за-

дач, в частности это: любые оптимизационные задачи, а также задачи классификации и кластеризации объектов, анализ данных, обработка изображений, нечеткая логика и др. С обзором различных модификаций алгоритма и примерами его практического применения можно ознакомиться в [10]. Таким образом, рассматриваемый популяционный алгоритм, инспирированный поведением летучих мышей, представляет интерес для дальнейших исследований.

### 3 МАТЕРИАЛЫ И МЕТОДЫ

Основными составляющими алгоритма являются: особенности перемещения летучих мышей в пространстве, и характеристики испускаемых ими звуковых сигналов.

Работа алгоритма основывается на следующей модели поведения летучих мышей [4]:

1. Все летучие мыши используют эхолокацию, чтобы определять расстояние, а также различать еду/добычу и препятствия.

2. Текущее положение каждой летучей мыши обозначим  $x_i$ , из него она перемещается случайным образом со скоростью  $v_i$ . В процессе движения летучие мыши испускают звуковые сигналы, имеющие частоту  $\omega_i$  и громкость  $A_i$ . При этом они могут изменять как частоту, так и интенсивность испускаемых импульсов  $r_i \in [0,1]$ , в зависимости от близости к цели.

3. Громкость звукового сигнала изменяется от большего начального значения  $A_0$  до меньшего заданного  $A_{\min}$ .

Положения всех летучих мышей в начальный момент задаются случайным образом. Дальнейшее перемещение каждой из них происходит согласно следующей миграционной процедуре:

$$\omega_i = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \cdot \beta, \quad (2)$$

$$v_i^{(t+1)} = v_i^{(t)} + (x_i^{(t)} - \bar{x}) \cdot \omega_i, \quad (3)$$

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}, \quad (4)$$

где  $\omega_i \in [\omega_{\min}, \omega_{\max}]$ .

Для определения новых положений летучих мышей, каждая из них осуществляет локальный поиск в окрестности своего текущего положения путем случайного блуждания:

$$x_i^{(new)} = x_i^{(old)} + \varepsilon \cdot \bar{A}(t). \quad (5)$$

Эволюция параметров, характеризующих испускаемые звуковые сигналы, осуществляется по формулам:

$$A_i^{(t+1)} = \alpha \cdot A_i^{(t)}, \quad (6)$$

$$r_i^{(t+1)} = r_i^{(0)} \cdot (1 - e^{-\gamma t}), \quad (7)$$

здесь рекомендуемые значения констант  $\alpha$  и  $\gamma$  равны 0,9.

Согласно (6)–(7), с увеличением числа итераций громкость испускаемых звуковых сигналов будет уменьшать-

ся, а их интенсивность увеличиваться, моделируя этим приближение летучей мыши к цели:

$$A_i^{(t)} \rightarrow 0, \quad r_i^{(t)} \rightarrow r_i^{(0)} \quad \text{при } t \rightarrow \infty.$$

Алгоритм летучих мышей для решения задачи глобальной оптимизации (1) представим в виде следующего псевдокода:

1. Инициализация популяции: задать начальное положение  $x_i$  и скорость  $v_i$  каждой  $i$ -ой летучей мыши,  $i = \overline{1, N}$ .

2. Инициализация характеристик звуковых импульсов: задать частоту  $\omega_i$ , интенсивность  $r_i$  и громкость  $A_i$  для каждой  $i$ -ой летучей мыши,  $i = \overline{1, N}$ .

3. Пока  $t \leq t_{\max}$

4. Для  $i = \overline{1, N}$

5. Перемещение каждой  $i$ -ой летучей мыши по формулам (2)–(4).

6. Если  $rand > r_i$

7. Выполнение локального поиска в окрестности текущего положения путем случайного блуждания по формуле (5).

8. Конец если.

9. Если  $rand < A_i$  и  $f(x_i^{(new)}) < f(\bar{x})$

10. Новое положение принято  $x_i^{(t+1)} \leftarrow x_i^{(new)}$ .

11. Уменьшить  $A_i$  и увеличить  $r_i$  согласно (6)–(7).

12. Конец если

13. Обновить  $\bar{x}$ .

14. Конец для.

15. Увеличить счетчик числа итераций  $t \leftarrow t + 1$ .

16. Конец пока.

В каноническом варианте алгоритма случайно выбираются те летучие мыши, которые осуществляют локальный поиск, в результате которого обновляется каждая из их координат. Чтобы повысить эффективность поиска в многомерном случае, можно по аналогии выбирать случайно пространственные координаты, подлежащая изменению. Такая модификация предполагает внесение изменений в 7 строке алгоритма на следующие:

для  $j = \overline{1, m}$

если  $rand > r_i$

$$x_{ij}^{(new)} = x_{ij}^{(old)} + \varepsilon \cdot \bar{A}(t)$$

конец если

конец для

Также в процессе миграции летучих мышей целесообразно использовать информацию не только о глобальном лучшем решении, выбранном среди всех, но и о лучшем решении, найденном каждой летучей мышью. Модифицируем способ вычисления скорости перемещения летучих мышей, используя подход, предложенный М. Clerc и J. Kennedy для алгоритма роя частиц [11]. Результирующая скорость каждой летучей мыши определяется следующими формулами:

$$\chi = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|}, \quad \varphi = c_1 + c_2,$$

$$\begin{aligned} \bar{v}_i^{(t+1)} &= \chi(v_i^{(t)} + c_1 \varepsilon_1 (\bar{x}_i - x_i^{(t)}) + c_2 \varepsilon_2 (\bar{x} - x_i^{(t)})), \\ v_i^{(t+1)} &= \bar{v}_i^{(t)} + (x_i^{(t)} - \bar{x}) \cdot \omega_i, \end{aligned}$$

где  $c_1 = c_2 = 2,05$ .

Представленные модификации позволили получить более эффективный вариант алгоритма летучих мышей для нахождения глобального минимума целевой функции.

#### 4 ЭКСПЕРИМЕНТЫ

Для проведения вычислительных экспериментов выбрали ряд тестовых функций:

1. Функция де Джонга:

$$f_1(x) = \sum_{i=1}^m x_i^2, \quad -10 \leq x_i \leq 10.$$

Глобальный минимум функции  $f_1(x^*) = 0$  при  $x^* = (0, 0, \dots, 0)$ .

2. Функция Розенброка:

$$f_2(x) = \sum_{i=1}^{m-1} \left( (x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2 \right), \quad -5 \leq x_i \leq 5.$$

Глобальный минимум функции  $f_2(x^*) = 0$  при  $x^* = (1, 1, \dots, 1)$ .

3. Функция Растригина:

$$f_3(x) = \sum_{i=1}^m \left( x_i^2 - 10 \cos(2\pi x_i) \right) + 10m, \quad -5 \leq x_i \leq 5.$$

Глобальный минимум функции  $f_3(x^*) = 0$  при  $x^* = (0, 0, \dots, 0)$ .

4. Функция Экли:

$$f_4(x) = 20 + e - 20 \exp \left( -\frac{1}{5} \sqrt{\frac{1}{m} \sum_{i=1}^m x_i^2} \right) - \exp \left( \frac{1}{m} \sum_{i=1}^m \cos(2\pi x_i) \right),$$

$$-20 \leq x_i \leq 20.$$

Глобальный минимум функции  $f_4(x^*) = 0$  при  $x^* = (0, 0, \dots, 0)$ .

Среди представленных функций выбраны как унимодальные функции  $f_1(x)$  и  $f_2(x)$ , так и мультимодальные функции  $f_3(x)$  и  $f_4(x)$ .

Для иллюстрации практического применения алгоритма рассмотрим инженерную задачу оптимизации конструкции цилиндрического сосуда высокого давления. Целевая функция определяет стоимость расходных материалов, необходимых для его изготовления. Дизайн конструкции сосуда определяется следующими параметрами: толщина стенки цилиндра  $T_h(x_1)$ , толщина сфери-

ческой головки  $T_h(x_2)$ , внутренний радиус цилиндрической оболочки  $R(x_3)$ , длина цилиндрической части  $L(x_4)$  (см. рис. 1) [12].

В результате получаем задачу нелинейной оптимизации:

$$f(x) = 0,6224x_1x_3x_4 + 1,7781x_2x_3^2 + 3,1661x_1^2x_4 + 19,84x_1^2x_3 \rightarrow \min, \quad (8)$$

$$g_1(x) = -x_1 + 0,0193x_3 \leq 0, \quad (9)$$

$$g_2(x) = -x_2 + 0,00954x_3 \leq 0, \quad (10)$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3} \pi x_3^3 + 12960000 \leq 0, \quad (11)$$

$$g_4(x) = x_4 - 240 \leq 0, \quad (12)$$

$$x_1 \geq 1 \times 0,0625, \quad x_2 \leq 99 \times 0,0625, \quad x_3 \geq 10, \quad x_4 \leq 200. \quad (13)$$

В задаче (8)–(13) переменные  $x_1$  и  $x_2$  являются дискретными величинами, которые должны быть кратны 0,0625 в соответствии имеющейся толщиной листового проката стали.

Для решения представленной задачи используем метод штрафов и перейдем к следующей задаче:

$$F(x) = f(x) + \sum_{j=1}^4 \mu_j \cdot \max(0, g_j^2(x)) \rightarrow \min.$$

#### 5 РЕЗУЛЬТАТЫ

Канонический и модифицированный алгоритмы реализованы в среде MATLAB R2012b. При сравнении результатов решения задачи (1) каноническим алгоритмом летучих мышей (ВА) и модифицированным (MdBA), каждый алгоритм применялся для каждой функции 50 раз. Используются следующие значения констант и параметров:  $t_{\max} = 1000$ ,  $N = 30$ ,  $\omega_{\min} = 0$ ,  $\omega_{\max} = 2$ ,  $A_i^{(0)} = 0,8$ ,  $r_i^{(0)} = \text{rand}$ ,  $\alpha = 0,9$ ,  $\gamma = 0,9$ . Полученные результаты для набора тестовых функций приведены в табл. 1.

Алгоритм летучих мышей содержит ряд варьируемых параметров, от значения которых, может сильно зависеть его эффективность. Исследуем зависимость эффективности работы модифицированного алгоритма от значений громкости  $A_i^{(0)}$  и интенсивности  $r_i^{(0)}$  звукового сигнала.

В первой серии экспериментов зафиксировали  $r_i^{(0)} = 0,5$ , а значение громкости изменяли в диапазоне от 0,1 до 1 при  $m = 10$ . В следующих экспериментах полагали  $A_i^{(0)} = 0,5$ , а интенсивность сигнала изменяли также от 0,1 до 1. Результаты вычислительных экспериментов представлены в табл. 2–3.

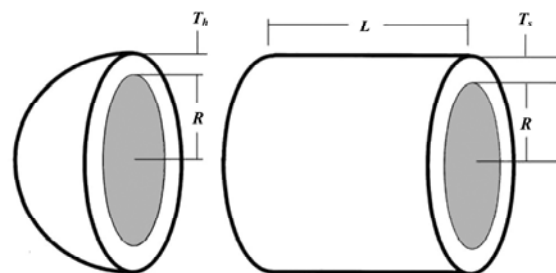


Рисунок 1 – Конструкция сосуда высокого давления

Таблица 1 – Результаты вычислительных экспериментов для тестовых функций

Функция	Размерность пространства, $m$	Алгоритм	Лучшее полученное значение целевой функции	Среднее среди полученных значений целевой функции	Худшее полученное значение целевой функции
$f_1(x)$	5	BA	7,5366e-03	3,9121e-01	1,3720e+00
		MdBA	1,7115e-10	4,7063e-09	3,6887e-08
	10	BA	4,3757e-01	1,8943e+00	4,0154e+00
		MdBA	1,0420e-07	5,1940e-07	1,7923e-06
	20	BA	2,2767e+00	5,6863e+00	9,1286e+00
		MdBA	1,5330e-06	7,8947e-06	2,5199e-05
$f_2(x)$	5	BA	2,8084e+00	3,2013e+01	1,0299e+02
		MdBA	2,5161e-06	2,3596e-01	3,9309e+00
	10	BA	5,8462e+01	2,2555e+02	5,6098e+02
		MdBA	3,4659e-03	8,7276e-01	9,6060e+00
	20	BA	1,6786e+02	8,7887e+02	1,7818e+03
		MdBA	7,0456e+00	1,1763e+01	1,9499e+01
$f_3(x)$	5	BA	6,0018e+00	1,6457e+01	2,7852e+01
		MdBA	3,8424e-01	2,3481e+00	6,9647e+00
	10	BA	3,3129e+01	5,3871e+01	7,2348e+01
		MdBA	1,9900e+00	6,7658e+00	1,5919e+01
	20	BA	1,0428e+02	1,4207e+02	1,6817e+02
		MdBA	5,9703e+00	1,7632e+01	3,6819e+01
$f_4(x)$	5	BA	5,7667e-01	2,3916e+00	3,8869e+00
		MdBA	1,8981e-05	3,3047e-02	1,6462e+00
	10	BA	2,0987e+00	3,2280e+00	4,0536e+00
		MdBA	3,7299e-04	2,4011e-02	1,1552e+00
	20	BA	2,6300e+00	4,2518e+00	3,5454e+00
		MdBA	1,5550e-03	2,1306e-01	2,1712e+00

Таблица 2 – Зависимость решения от начального значения громкости сигнала

	$A_i^{(0)}$									
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0
$f_1(x)$	7,96e-08	8,45e-08	1,10e-07	1,26e-07	1,57e-07	1,58e-07	1,69e-07	2,18e-07	2,09e-07	2,51e-07
$f_2(x)$	1,21e+01	6,28e+00	9,73e+00	8,11e+00	6,97e+00	4,85e+00	3,97e+00	3,71e+00	5,35e+00	4,95e+00
$f_3(x)$	6,94e+00	9,35e+00	9,55e+00	1,01e+01	1,23e+01	1,29e+01	1,53e+01	1,54e+01	1,47e+01	1,70e+01
$f_4(x)$	3,48e-04	3,76e-04	4,20e-04	4,45e-04	4,63e-04	4,78e-04	5,16e-04	5,63e-04	2,36e-02	9,68e-02

Таблица 3 – Зависимость решения от начального значения интенсивности сигнала

	$r_i^{(0)}$									
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0
$f_1(x)$	2,18e-07	1,97e-07	1,75e-07	1,57e-07	1,41e-07	5,41e-07	5,06e-04	8,84e-02	1,36e+00	1,19e+01
$f_2(x)$	8,77e-01	9,63e-01	7,90e-01	1,56e+00	5,84e+00	1,67e+01	2,65e+01	1,47e+02	3,89e+02	4,27e+03
$f_3(x)$	1,74e+01	1,29e+01	1,19e+01	1,18e+01	1,47e+01	1,33e+01	1,21e+01	2,80e+01	4,98e+01	6,74e+01
$f_4(x)$	1,20e+00	2,22e-01	1,70e-01	6,38e-02	5,11e-04	3,38e-02	9,47e-02	1,65e+00	3,77e+00	5,19e+00

Рассмотрим изменение значения целевой функции в процессе поиска экстремума для функций Растригина и Экли, соответствующие графики приведены на рис. 2. В каноническом и модифицированном алгоритмах полагали  $m=10$ , и с учетом результатов табл. 2–3  $A_i^{(0)} = 0,35$ ,  $r_i^{(0)} = 0,4$ .

Модифицированный алгоритм летучих мышей применен для решения задачи о дизайне сосуда высокого давления (8)–(13) и получено следующее лучшее решение:

$$f(x^*) = 6059,1316103768,$$

$$x^* = (0,8125, 0,4375, 42,0984440172, 176,6366206192).$$

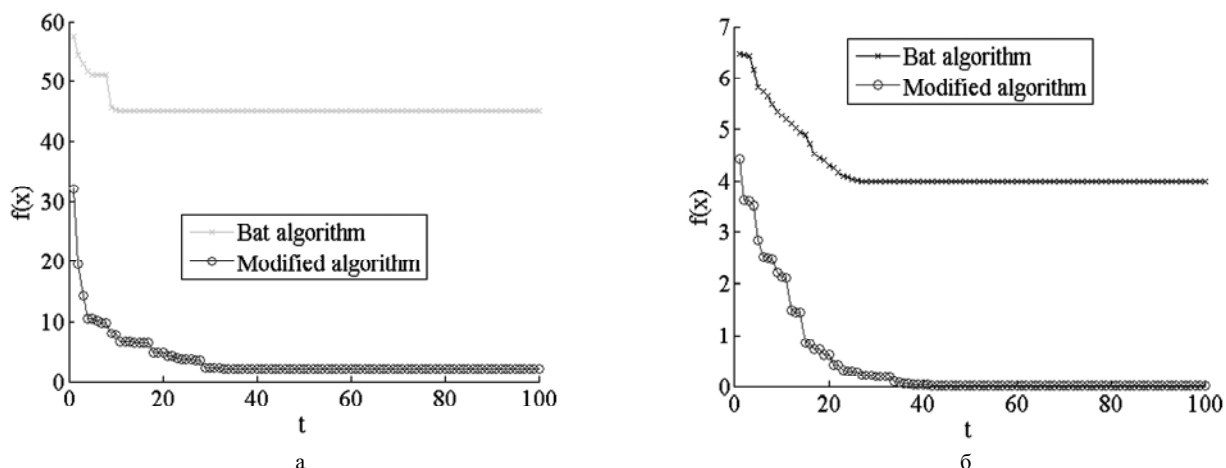


Рисунок 2 – Графики изменения значений функции:  
 а – Растригина, б – Экли

## 6 ОБСУЖДЕНИЕ

Результаты вычислительных экспериментов показали, что модифицированный алгоритм летучих мышей является более эффективным по сравнению с каноническим вариантом для решения задачи глобальной оптимизации. Графики изменения значений целевой функции показали, что модифицированный алгоритм характеризуется более высокой скоростью сходимости и точностью нахождения оптимального значения. Недостатком модифицированного алгоритма является время его выполнения, которое больше времени выполнения канонического алгоритма. Исследована зависимость эффективности модифицированного алгоритма от таких параметров алгоритма как громкость и интенсивность звукового сигнала. Можно сделать вывод, что влияние данных параметров сильно зависит от вида целевой функции. Поэтому оптимальные значения громкости и интенсивности сигнала должны выбираться в каждом конкретном случае индивидуально, при этом отправными могут быть такие значения:  $A_i^{(0)} = 0,25 \dots 0,35$ ,  $r_i^{(0)} = 0,3 \dots 0,4$ .

Решением задачи о минимизации расходов на изготовление сосуда высокого давления занимались многие авторы, применяя различные методы. Сопоставление результатов решения данной оптимизационной задачи предложенным модифицированным алгоритмом летучих мышей с некоторыми уже известными результатами приведено в табл. 4.

Из данных, представленных в табл. 4 видно, что лучшие результаты при решении задачи (8)–(13) получены с использованием популяционных алгоритмов. Значение целевой функции, полученное модифицированным алгоритмом летучих мышей меньше значений, полученных многими другими алгоритмами, использующими принципы роевого интеллекта. Это свидетельствует об эффективности предложенного алгоритма. Однако, на сегодняшний день известны новые результаты с лучшим решением, например, найденное алгоритмом серых волков [20]. Тем не менее, модифицированный алгоритм летучих мышей является достаточно конкурентоспособным для решения практических задач и предполагает дальнейшее развитие.

## ВЫВОДЫ

В работе рассмотрена задача нахождения глобального минимума целевой функции при помощи алгоритма, инспирированного поведением летучих мышей, который относится к алгоритмам роевого интеллекта.

Канонический алгоритм летучих мышей усовершенствован за счет применения методики алгоритма роя частиц для вычисления скорости движения агентов. Результаты вычислительных экспериментов показали, что модифицированный алгоритм летучих мышей характеризуется более высокой скоростью сходимости и точностью нахождения оптимального значения целевой функции.

Таблица 4 – Результаты решения задачи о дизайне сосуда высокого давления

Авторы	Метод	$f(x^*)$
E. Sandgren (1988) [13]	Метод ветвей и границ	8129,1036
B. Kannan, S. Kramer (1994) [14]	Метод множителей Лагранжа	7198,0428
Q. He, L. Wang (2007) [15]	Алгоритм роя частиц	6061,0777
E. Mezura-Montes и др. (2007) [16]	Модифицированный метод дифференциальной эволюции	6059,701660
L. Cagnina и др. (2008) [12]	Алгоритм роя частиц	6059,714335
H. Bernardino и др. (2008) [17]	Генетический алгоритм, скрещенный с алгоритмом искусственной иммунной системы (GA-AIS)	6059,85
A. Kaveh, S. Talatahari (2010) [18]	Муравьиный алгоритм	6059,7258
B. Akay, D. Karaboga (2012) [19]	Алгоритм искусственной пчелиной колонии	6059,714339
S. Mirjalili и др. (2014) [20]	Алгоритм серых волков	6051,5639
Данная работа	Модифицированный алгоритм летучих мышей	6059,131610

При помощи модифицированного алгоритма решения практическая задача о минимизации расходов на изготовление сосуда высокого давления. Найденное решение является одним из лучших, среди многих вариантов решений, полученных другими авторами с применением различных популяционных алгоритмов.

Таким образом, предложенный модифицированный алгоритм летучих мышей является эффективным методом поиска экстремума целевой функции любой сложности. Учитывая универсальность и преимущества данного алгоритма, дальнейшие исследования будут направлены на его применение в методах кластеризации, в частности для k-means.

## СПИСОК ЛІТЕРАТУРИ

1. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой : учебное пособие / А. П. Карпенко. – Москва : Издательство МГТУ им. Н. Э. Баумана, 2014. – 448 с.
2. Субботін С. О. Нейтеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей : монографія / С. О. Субботін, А. О. Олійник, О. О. Олійник ; під заг. ред. С. О. Субботіна. – Запоріжжя : ЗНТУ, 2009. – 375 с.
3. Карпенко А. П. Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов / А. П. Карпенко // Приложение к журналу «Информационные технологии». – 2012. – № 7. – С. 1–32.
4. Yang X. S. A new metaheuristic bat-inspired algorithm / X. S. Yang // Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). – 2010. – Vol. 284. – P. 65–74.
5. Частикова В. А. Алгоритм летучих мышей для решения задачи глобальной оптимизации / В. А. Частикова, Е. Ф. Новикова // Научные труды КубГТУ (электронный сетевой политематический журнал) [Электронный ресурс]. – № 2. – 2015. – Режим доступа : <http://ntk.kubstu.ru/file/348>.
6. Yang X. S. Bat algorithm for multi-objective optimization / X. S. Yang // International Journal of Bio-Inspired Computation. – 2011. – Vol. 3, № 5. – P. 267–274. DOI:10.1504/IJBIC.2011.042259.
7. M BBA: a binary bat algorithm for feature selection / [Nakamura R., Pereira L., Costa K. and other] // Conference on graphics, patterns and images (25th SIBGRAPI), August 22–25, 2012 : IEEE Publication. – P. 291–297. DOI:10.1109/SIBGRAPI.2012.47.
8. Fister I. J. A hybrid bat algorithm / I. J. Fister, D. Fister, X. S. Yang // Electrotechnical Review. – 2013. – Vol. 80, № 1–2. – P. 1–7.
9. Wang G. A Novel hybrid bat algorithm with harmony search for global numerical optimization / Gaige Wang, Lihong Guo // Journal of Applied Mathematics : Hindawi Publishing Corporation [Электронный ресурс]. – Vol. 2013. DOI: 0.1155/2013/696491.
10. Yang X. S. Bat algorithm: literature review and applications / Xin-She Yang, Xingshi He // International Journal of Bio-Inspired Computation. – 2013. – Vol. 5. – № 3. – P. 141–149. DOI:10.1504/IJBIC.2013.055093.
11. Clerc M. The particle swarm – explosion, stability, and convergence in a multidimensional complex space / M. Clerc, J. Kennedy // IEEE Transactions on Evolutionary Computation. – 2002. – Vol. 6. – № 1. – P. 58–73. DOI:10.1109/4235.985692.
12. Cagnina L. C. Solving engineering optimization problems with the simple constrained particle swarm optimizer / L. C. Cagnina, S. C. Esquivel, C. A. C. Coello // Informatica. – 2008. – Vol. 32, № 3. – P. 319–326.
13. Sandgren E. Nonlinear integer and discrete programming in mechanical design / E. Sandgren // Proceedings of the ASME Design Technology Conference : Kissimine. – 1988. – P. 95–105.
14. Kannan B. An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design / B. Kannan, S. N. Kramer // Journal of mechanical design. – 1994. – Vol. 116, № 2. – P. 405–411. DOI:10.1115/1.2919393.
15. He Q. An effective co-evolutionary particle swarm optimization for constrained engineering design problems / Qie He, Ling Wang // Engineering applications of artificial intelligence. – 2007. – Vol. 20, № 1. – P. 89–99.
16. Multiple trial vectors in differential evolution for engineering design / Mezura-Montes E., Coello Coello C. A., Velázquez-Reyes J., Muñoz-Dávila L. // Engineering Optimization. – 2007. – Vol. 39, № 5. – P. 567–589.
17. Bernardino H. S. A new hybrid AIS-GA for constrained optimization problems in mechanical engineering / H. S. Bernardino, H. J. C. Barbosa, A. C. C. Lemonge, L. G. Fonseca // Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), 1–6 June 2008. – P. 1455–1462. DOI:10.1109/CEC.2008.4630985.
18. Kaveh A. An improved ant colony optimization for constrained engineering design problems / A. Kaveh, S. Talatahari // Engineering Computations. – 2010. – Vol. 27, № 1. – P. 155–182.
19. Akay B. Artificial bee colony algorithm for large-scale problems and engineering design optimization / Bahriye Akay, Dervis Karaboga // Journal of intelligent manufacturing. – 2012. – Vol. 23, № 4. – P. 1001–1014.
20. Mirjalili S. Grey wolf optimizer / S. Mirjalili, S. M. Mirjalili, A. Lewis // Advances in engineering software. – 2014. – Vol. 69. – P. 46–61.

Статья поступила в редакцию 03.09.2015.

После доработки 11.10.2015.

Красношлик Н. О.

Канд. техн. наук, доцент кафедры прикладной математики та інформатики Черкаського національного університету імені Богдана Хмельницького, Черкаси, Україна

## РОЗВ'ЯЗУВАННЯ ЗАДАЧІ ГЛОБАЛЬНОЇ ОПТИМІЗАЦІЇ МОДИФІКОВАНИМ АЛГОРИТМОМ КАЖАНІВ

У роботі розглянуто алгоритм кажанів для розв'язування задачі глобальної оптимізації. Даний метаевристичний алгоритм відноситься до алгоритмів ройового інтелекту, які інтенсивно розвиваються в останні роки. Метою роботи є удосконалення алгоритму кажанів, дослідження ефективності його роботи і застосування для розв'язування оптимізаційних задач. Запропоновано модифікований варіант канонічного алгоритму, в якому для обчислення швидкості руху кажанів застосована методика алгоритму рою часток. Проведено обчислювальні експерименти з порівняння точності і швидкості збіжності канонічного та модифікованого алгоритмів. Встановлено, що запропонований варіант алгоритму є більш ефективним при знаходженні глобального мінімуму унімодальних і мультимодальних тестових функцій. Досліджено залежність ефективності роботи модифікованого алгоритму кажанів від заданих параметрів. В якості варіюваних параметрів обрані початкові значення гучності та інтенсивності звукового сигналу, що випускається кажанами. Модифікований алгоритм застосований для розв'язання практичної задачі мінімізації вартості витратних матеріалів, необхідних для виготовлення посудини високого тиску. Проведено співставлення отриманого розв'язку даної оптимізаційної задачі з результатами інших авторів, які використовувати як класичні, так і популяційні алгоритми.

**Ключові слова:** алгоритм кажанів, популяційний алгоритм, ройовий інтелект, глобальна оптимізація.

Krasnoshlyk N. O.

PhD, Associate professor of department of informatics and applied mathematics, Bohdan Khmelnytsky National University of Cherkasy, Cherkasy, Ukraine

#### A MODIFIED BAT ALGORITHM FOR SOLVING GLOBAL OPTIMIZATION PROBLEM

In this paper we consider a bat algorithm for solving the problem of global optimization. This metaheuristic algorithm applies to swarm intelligence algorithms, which are developing rapidly in recent years. The aim of the work is to improve the bat algorithm, study its efficiency and application for solving optimization problems. A modified version of the algorithm in which to calculate speed of the bats used the technique of particle swarm optimization is proposed. The computational experiments have been conducted to compare the accuracy and the speed of convergence of the canonical and the modified algorithms. It was found that the proposed version of the algorithm is more effective in finding the global minimum of unimodal and multimodal test functions. The dependence of the efficiency of modified bat algorithm from the set parameters is investigated. As variable parameters chosen initial values of the loudness and pulse emission rate emitted by bats. The modified algorithm is applied to solve practical problem of minimize the cost of pressure vessel design. The comparison of the solution of the optimization problem with the results of other authors who used both classical and population algorithms was conducted.

**Keywords:** bat algorithm, population algorithm, swarm intelligence, global optimization.

#### REFERENCES

1. Karpenko A. P. Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdohnovlennyye prirodoy : uchebnoe posobie. Moscow, Izdatel'stvo MGTU im. N. Je. Bauman, 2014, 448 p.
2. Subbotin S. O., Olijnyk A. O., Olijnyk O. O. Neiteratyvni, evoljucijni ta mul'tyagentni metody syntezu nechitkologichnyh i nejmromerzhnyh modelej : monografija ; pid zag. red. S. O. Subbotina, Zaporizhzhja, ZNTU, 2009, 375 p.
3. Karpenko A. P. Populjacionnyye algoritmy global'noj poiskovoy optimizatsii. Obzor novyh i maloizvestnyh algoritmov, *Prilozhenie k zhurnalu «Informacionnyye tehnologii»*, 2012, No. 7, pp. 1–32.
4. Yang X. S. A new metaheuristic bat-inspired algorithm, *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, 2010, Vol. 284, pp. 65–74.
5. Chastikova V. A., Novikova E. F. Algoritmy letuchih myshej dlja reshenija zadachi global'noj optimizatsii, *Nauchnyye trudy KubGTU (jelektronnyj setevoy politematicheskij zhurnal)*, No. 2, 2015, <http://ntk.kubstu.ru/file/348>.
6. Yang X. S. Bat algorithm for multi-objective optimization, *International Journal of Bio-Inspired Computation*, 2011, Vol. 3, No. 5, pp. 267–274. DOI:10.1504/IJBIC.2011.042259.
7. Nakamura R., Pereira L., Costa K., Rodrigues D., Papa J., Yang X. BBA: a binary bat algorithm for feature selection, *Conference on graphics, patterns and images (25th SIBGRAPI)*, August 22–25, 2012, IEEE Publication, pp. 291–297. DOI:10.1109/SIBGRAPI.2012.47.
8. Fister I. J., Fister D., Yang X. S. A hybrid bat algorithm, *Electrotechnical Review*, 2013, Vol. 80, No. 1–2, pp. 1–7.
9. Wang G., Guo L. A Novel hybrid bat algorithm with harmony search for global numerical optimization, *Journal of Applied Mathematics*, Hindawi Publishing Corporation [Электронный ресурс], Vol. 2013, <http://dx.doi.org/10.1155/2013/696491>.
10. Yang X. S., He X. Bat algorithm: literature review and applications, *International Journal of Bio-Inspired Computation*, 2013, Vol. 5, No. 3, pp. 141–149. DOI:10.1504/IJBIC.2013.055093.
11. Clerc M., Kennedy J. The particle swarm – explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation*, 2002, Vol. 6, No. 1, pp. 58–73. DOI:10.1109/4235.985692.
12. Cagnina L. C., Esquivel S. C., Coello C. A. C. Solving engineering optimization problems with the simple constrained particle swarm optimizer, *Informatica*, 2008, Vol. 32, No. 3, pp. 319–326.
13. Sandgren E. Nonlinear integer and discrete programming in mechanical design, *Proceedings of the ASME Design Technology Conference*, Kissimmee, 1988, pp. 95–105.
14. Kannan B., Kramer S. N. An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, *Journal of mechanical design*, 1994, Vol. 116, No. 2, pp. 405–411. DOI:10.1115/1.2919393.
15. He Q., Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems, *Engineering applications of artificial intelligence*, 2007, Vol. 20, No. 1, pp. 89–99.
16. Mezura-Montes E., Coello Coello C. A., Velázquez-Reyes J., Muñoz-Dávila L. Multiple trial vectors in differential evolution for engineering design, *Engineering Optimization*, 2007, Vol. 39, No. 5, pp. 567–589.
17. Bernardino H. S., Barbosa H. J. C., Lemonge A. C. C., Fonseca L. G. A new hybrid AIS-GA for constrained optimization problems in mechanical engineering, *Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, 1–6 June 2008, pp. 1455–1462. DOI:10.1109/CEC.2008.4630985.
18. Kaveh A., Talatahari S. An improved ant colony optimization for constrained engineering design problems, *Engineering Computations*, 2010, Vol. 27, No. 1, pp. 155–182.
19. Akay B., Karaboga D. Artificial bee colony algorithm for large-scale problems and engineering design optimization, *Journal of intelligent manufacturing*, 2012, Vol. 23, No. 4, pp. 1001–1014.
20. Mirjalili S., Mirjalili S. M., Lewis A. Grey wolf optimizer, *Advances in engineering software*, 2014, Vol. 69, pp. 46–61.