УДК 004.94; 517.9:519.6

РОЙОВИЙ ІНТЕЛЕКТ У ТЕХНІЧНИХ ОБ'ЄКТАХ І СИСТЕМАХ

М. В. Литовченко[1], В. В. Хайдуров[1]

[1] Навчально-науковий Фізико-технічний інститут

Анотація

У роботі було розглянуто прикладні та звичайні функціональні задачі математичної глобальної однокритеріальної оптимізації. На тестування було запропоновано три алгоритми оптимізації, а саме: бджолиний алгоритм, алгоритм зграї кажанів, китовий алгоритм. У результаті було представлено те як різні алгоритми впоралися з прикладною задачею для знаходження оптимальних значень параметрів конструкції для моделювання зварної балки.

Ключові слова: прикладні задачі в оптимізаційній постановці, ройовий інтелект, бджолиний алгоритм, алгоритм зграї кажанів, китовий алгоритм, глобальна оптимізація, The Bees Algorithm, Bat Ispired Algorithm, The Whale Optimization Algorithm, WOA, складні об'єкти та системи

Вступ

Ройовий інтелект, як і інші методи з відкритої природи, має значну актуальність у оптимізації складних систем. Особливо в умовах сьогодення, де дослідження зосереджені на знаходженні оптимальних рішень у великих і складних просторах, ройовий інтелект ефективно розв'язує прикладні задачі глобальної оптимізації, де потрібно знаходити найкращі рішення великого обсягу даних із складними функціональними взаємозалежностями. Системи, які описуються нелінійними залежностями, можуть бути складними для оптимізації традиційними методами й алгоритмами глобальної багатовимірної оптимізації. Слід зазначити, що алгоритми і методи ройового інтелекту швидко адаптуються до змін у середовищі або вимогах завдання, що робить їх потужним інструментом для оптимізації складних систем у змінних умовах.

1. Постановка задачі

Більшість прикладних задач оптимізації зводяться до знаходження глобальних екстремумів функцій у класичній постановці, яка подається так [1]:

$$y = F(X) \to min\,(max), X \in G, G \subset R^n,$$

де G – простір пошуку значень аргументів X, n – розмірність простору пошуку глобального екстремуму функції. X – факторні незалежні змінні, які встановлюють причинно-наслідковий зв'язок з залежною результуючою змінною y. На функцію $F(X)$ часто накладаються функціональні обмеження, які подаються у вигляді різних функціональних залежностей.

Складність цільової функції $F(X)$ і відповідних обмежень $g_i(X)$ визначається конкретним технічним завданням. Від вигляду $F(X)$ і $g_i(X)$ залежить вибір оптимізаційного методу, який буде застосовуватись для пошуку $F(X^*)$. Слід зазначити, що залежність виду:

$$y = F(X) \to min$$

у задачах машинного навчання часто подається у вигляді сумарної або середньої помилки, яка полягає у відхиленні між реальними даними та модельними, які пропускаються через функціональну залежність.

У даній роботі розглядатимуться завдання пошуку глобального мінімуму задач без функціональних обмежень на цільову функцію/функціонал, та задачі з обмеженнями у вигляді функцій, а також методи їх пошуку.

2. Алгоритми прикладних оптимізаційних задач

2.1. Бджолиний Алгоритм

Метод оптимізації медоносних бджіл, який ще називають бджолиним алгоритмом, починається з визначення розміру популяції та кількості бджіл які будуть займатися пошуком локального вирішення оптимізаційної задачі [2], як це відбувається у медоносних бджіл коли бджоли розвідники повертаються у вулик та розповідають іншим своїм бджолам працівникам про якість знайденого джерела пилку. Початкова популяція створюється завдяки генерації випадкових значень на визначеному межами просторі рішень. Відсоток того скільки буде бджіл працівників визначається вуликом, а в алгоритмі це визначають параметри e та b, які відповідають за кількість обраних ділянок для пошуку, що описуються як найкращі та кращі ділянки з усіх рішень популяції, відповідно. Кількість найкращих ділянок не може бути більшою за кількість кращих, так як найкращі обираються з числа кращих ділянок.

2.2. Алгоритм зграї кажанів

Алгоритм зграї кажанів був натхнений природою ехолокаційних здібностей мікрокажанів, бо зазвичай великі кажані не мають такої здібності [3]. Ехолокацію мікрокажани використовують для пошуку здобичі та орієнтування у просторі при полюванні за здобиччю, що в нашому випадку може бути використано для пошуку оптимального рішення з простору рішень. Для алгоритму було виведено деякі константи з природи ехолокації мікрокажанів, такі як: частота звукових імпульсів f, довжина хвилі ультразвукових імпульсів λ, та швидкість звуку у повітрі v. Але так як ці параметри взаємо залежні за формулою:

$$\lambda = \frac{v}{f}, \tag{1}$$

в алгоритмі для тестування було використано тільки f – частота ультразвукових імпульсів. Частота ультразвукових імпульсів впливає на те як далеко зайде уявний кажан при переміщенні. Для алгоритму також визначають параметри які впливають на рух уявного кажана у просторі рішень, а саме: v – швидкість руху кажана у просторі рішень, A – гучність ультразвукового імпульсу для пошуку оптимального рішення, та r – швидкість ультразвукового імпульсу для пошуку. Якщо швидкість v впливає на те як далеко уявний кажан може долетіти на просторі рішень, то гучність A та швидкість r імпульсу регулюють те наскільки близько уявний кажан знаходиться від здобиччі . Зміна параметрів A та r відбувається за допомогою допоміжних змінних α та γ відповідно за формулами (2) та (3) в межах $0 < \alpha < 1$ та $\gamma > 0$. Змінюються параметри A та r до певних обмежень що вказані в (4).

$$A_i^{t+1} = \alpha A_i^t \tag{2}$$

$$r_i^{t+1} = r_i^0 \left[1 - \exp -\gamma * t\right] \tag{3}$$

$$A_i^t \to 0, r_i^t \to r_i^0, \ as\ t \to \infty, \tag{4}$$

де t – номер ітерації, r_i^0 – початкове значення r_i, яке може бути близьким до нуля або $r_i^0 \in [0,1]$. Початкове значення A_i^0 при цьому повинно бути $[1,2]$, а $A_{min} = 0$.

Для кожного кажана значення цих параметрів різне, тобто маємо вектори розмірністю n де кожен елемент вектора це значення для окремого кажана. Значення параметрів які були використанні при тестуванні продемонстровані у табл. 1. Для переміщення кажанів по простору рішень використовуються відповідні формули (5), але для локального пошуку рішення використовують вже іншу формулу (6) в якій використовується не певне значення гучності ультразвукового імпульсу, а середня гучність по всім кажанім на поточній ітерації.

$$\begin{gathered} f_i = f_{min} + (f_{max} - f_{min})\beta, \\ v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i, \\ x_i^t = x_i^{t-1} + v_i^t, \end{gathered} \tag{5}$$

де x_* – найкращий результат пошуку на поточній ітерації.

$$x_{new} = x_{old} + \epsilon A^t, \tag{6}$$

де $\epsilon \in [-1,1]$ – випадкове число, а $A^t =< A_i^t >$ – середнє значення гучності ультразвукового імпульсу всіх кажанів на поточній ітерації.

Таблиця 1. Таблиця визначених значень параметрів для алгоритму зграї кажанів.

Параметр	Значення
n	100
f_{min}	0
f_{max}	100
A_i	$\in [1,2]$
A_{min}	0
r_i	$\in [0,1]$
α	0.9
γ	0.9

2.3. Китовий Алгоритм

Алгоритм оптимізації китів черпає натхнення з природи полювання горбатих китів. Для вполювання зграї маленьких риб, кити пускають бульбашки та рухаються по спіралі для того щоб загнати здобич у вузьке місце поближче до поверхні води де й атакують свою здобич [4]. Так як оптимальне рішення у просторі рішень зазвичай не відоме, тоді алгоритм китів припускає що найкращий кандидат з найкращим результатом на просторі рішень є здобиччю. Після визначення найкращого пошукового агента (кита) інші агенти намагаються оновити своє розташування у напрямку визначеної позиції найкращого агента. Така поведінка описується наступними рівняннями:

$$\vec{D} = |\vec{C}\vec{X}^*(t) - \vec{X}(t)|, \tag{7}$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}, \tag{8}$$

де $\vec{A}$ та $\vec{C}$ вектори коефіцієнти які обраховуються за (9), (10), X^* – вектор позиції найкращого рішення яке є на поточній ітерації, $\vec{X}$ – позиційний вектор одного кита на просторі пошуку, $\cdot$ – поелементне множення.

Вектори коефіцієнти $\vec{A}$ та $\vec{C}$ обраховуються за такими формулами:

$$\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a}, \tag{9}$$

$$\vec{C} = 2 \cdot \vec{r}, \tag{10}$$

де $\vec{a}$ лінійно зменшується від 2 до 0 вздовж максимальної кількості ітерацій, та $\vec{r}$ – випадковий вектор в проміжку $[0,1]$. Це допомагає іншим пошуковим агентам рухатися до позиції найкращого рішення та симулювати оточення здобичі спіральними рухами.

Цей процес можна розширити до n-вимірного простору рішень, де пошукові агенти будуть рухатися навколо найкращого рішення на поточній ітерації.

Процес оптимізації можна розділити на два етапи: Метод атаки бульбашковою сіткою (фаза експлуатації) та Пошук здобиччі (фаза розвідки).

Фаза експлуатації поділяється на наступні два кроки:

1. Зменшувально-оточувальний механізм: цей механізм досягається завдяки зменшенню значення параметра $\vec{a}$ з рівняння (9), таким чином зменшується значення вектору-коефіцієнту $\vec{A}$. Завдяки цьому нова позиція пошукового агенту може бути визначена де завгодно між поточною позицією агента та позицією найкращого агента на поточній ітерації.
2. Спіральне оновлення позиції: така поведінка досягається завдяки наступному рівнянню, що імітує гвинтоподібний рух горбатих китів:

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t),$$

де $\vec{D}' = |\vec{X}^*(t) - \vec{X}(t)|$ та вказує відстань i-того кита (агента пошуку) до здобиччі (кращого рішення на поточній ітерації), b – константа для визначення форми логарифмічної спіралі, а l – випадкове число в межах $[-1, 1]$ та $\cdot$ – поелементне множення. Така модель виглядає наступним чином:

$$\begin{aligned}&\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}, p < 0.5\\ &\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t), p \geq 0.5,\end{aligned}$$

де p – випадкове число в межах $[0, 1]$.

У фазі пошуку здобиччі пошукові агенти шукають здобич випадковим чином. На відміну від фази експлуатації, в цій фазі рух відбувається до випадково обраного пошукового агента, а не до найкращого. Цей рух відбувається у обох напрямках, до та від еталонного кита (пошукового агента). Математична модель цього процесу описується наступним чином:

$$\vec{D} = |\vec{C} \cdot \vec{X_{rand}} - \vec{X}|,$$

$$\vec{X}(t+1) = \vec{X_{rand}} - \vec{A} \cdot \vec{D},$$

де $\vec{X_{rand}}$ – випадково обраний пошуковий агент до якого буде виконуватися рух.

3. Практичні результати роботи

3.1. Тестування алгоритмів ройового інтелекту на класичних задачах оптимізації

1. *Двовимірна функція Біла.* Тестова функція Біла має такий математичний вигляд [5]:

$$f(x) = (1,5 - x + xy)^2 \to min,$$

де $x \in [-4,5; 4,5]$, $y \in [-4,5; 4,5]$. Глобальний мінімум в точці $(x^*, y^*) = (3; 0,5)$. $f^* = 0$.

2. *Двовимірна функція Голдстейн-прайса.*

$$\begin{aligned}s_1(x,y) &= (x+y+1)^2,\\ s_2(x,y) &= 19 - 14x + 3x^2,\\ s_3(x,y) &= -14y + 6xy + 3y^2,\\ s_4(x,y) &= s_2(x,y) + s_3(x,y),\\ s_5(x,y) &= (2x-3y)^2,\\ s_6(x,y) &= 18 - 32x + 12x^2,\\ s_7(x,y) &= 48y - 36xy + 27y^2,\\ s_8(x,y) &= s_6(x,y) + s_7(x,y),\\ g_1(x,y) &= 1 + s_1(x,y)\, s_2(x,y),\\ g_2(x,y) &= 30 + s_3(x,y)\, s_6(x,y),\\ f(x,y) &= g_1(x,y)\, g_2(x,y),\end{aligned}$$

де $x \in [-2; 2]$, $y \in [-2; 2]$.

3. *Функція Розенброка.*

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

де $x_i \in (-\infty; +\infty)$.

4. *Функція Растригіна.*

$$f(X) = An + \sum_{i=1}^{n} \left(x_i^2 - A \cdot \cos\left(2\pi x_i\right)\right),$$

де $A = 10$, $x_i \in [-5,12; 5,12]$, $i = \overline{1; n}$. Глобальний мінімум в точці $X^* = \left(x_1^*, x_2^*, ..., x_n^*\right) = (0; 0; ...; 0)$. $f\left(X^*\right) = 0$.

5. *Функція Аклея.*

$$\begin{aligned}f_1(x,y) &= -20e^{-\frac{1}{5}\sqrt{\frac{x^2+y^2}{2}}},\\ f_2(x,y) &= -e^{-0,5(\cos 2\pi x + \cos 2\pi y)},\\ f(x,y) &= f_1(x,y) + f_2(x,y),\end{aligned}$$

де $x, y \in [-5; +5]$.

3.2. Тестування алгоритмів ройового інтелекту на прикладних задачах оптимізації

Задача оптимізації конструкції зварної балки. Математичне формулювання задачі наведено нижче. Потрібно знайти глобальий мінімум функції виду:

$$f(X) = 1,10471x_1^2 + 0,04811x_3x_4\left(14 + x_2\right). \quad (11)$$

Обмеження на цільову функцію виглядають так:

$$\begin{aligned}&\tau(X) - \tau max \leq 0,\\ &\sigma(X) - \sigma_{max} \leq 0, x_1 - x_4 \leq 0,\\ &0,10471x_1^2 + 0,04811x_3x_4\left(14 + x_2\right) - 5.0 \leq 0,\\ &0,125 - x_1 \leq 0,\\ &\delta(X) - \delta_{max} \leq 0,\\ &P - P_C(X) \leq 0,\\ &0,1 \leq x_i \leq 2, i = 1; 4\\ &0,1 \leq x_i \leq 10, i = 2; 3,\end{aligned} \quad (12)$$

Таблиця 2. Таблиця результатів оптимізації параметрів конструювання зварної балки.

Метод	Змінні конструкції				$f(x)$
	x_1	x_2	x_3	x_4	
(1) BOA	0,2443	6,2177	8,2915	0,2443	2,3811
(2) BIA	0,2445	6,2177	8,2915	0,2444	2,3810
(3) WOA	0,2444	6,2177	8,2914	0,2443	2,3811

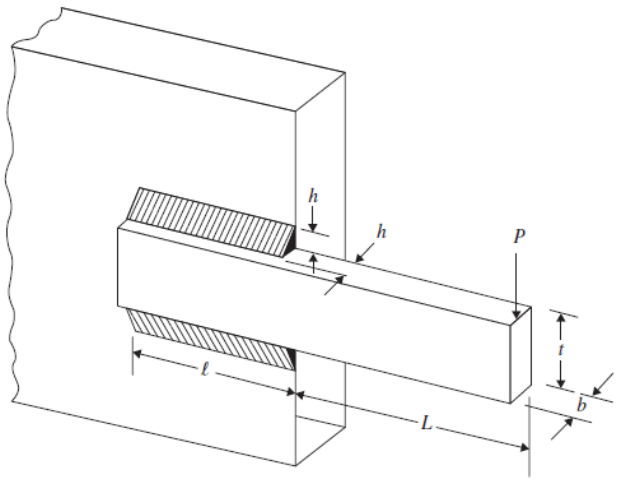

Рис. 1. Вигляд зварної балки

де

$$\tau(X) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2},$$
$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \tau'' = \frac{MR}{J}, M = P\left(L + \frac{x_2}{2}\right),$$
$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2},$$
$$J = 2\left\{\frac{x_1x_2}{\sqrt{2}}\left[\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)^2\right]\right\}, \quad (13)$$
$$\sigma(X) = \frac{6PL}{x_4x_3^2},$$
$$\delta(X) = \frac{4PL^3}{Ex_3^3x_4},$$
$$P_c(X) = \frac{4.013\sqrt{EG\left(x_3^2x_4^6/36\right)}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right).$$

Оптимальний розв'язок поставленої задачі (11), (12), (13) подається у вигляді точки:

$$X^* = (0,2444; 6,2177; 8,2915; 0,2444),$$
$$f^* = 2,3810.$$

Результати тестування на прикладній задачі для всіх методів/алгоритмів подаються у табл. 2.

3.3. Аналіз отриманих результатів тестування

У результаті можна сказати що є сенс на об'єднання бджолиного та китового алгоритмів щоб отримати результат ще швидше та зменшити тенденцію бджолиного алгоритму до сповільнення при збільшені розмірності вектору позиції на просторі пошуку. Але й об'єднання китового алгоритму та алгоритму зграї кажанів теж має сенс щоб зменшити тенденцію зграї кажанів до застрягання в локальному мінімумі та покращити рух китів у напрямку найкращого рішення.

Висновки

У роботі було протестовано три методи й алгоритми ройового інтелекту: бджолиний алгоритм, алгоритм зграї кажанів, та алгоритм китів. Методи й алгоритми тестувалися на різного роду тестових функціях для знаходження глобального екстремуму що відносяться до задач глобальної багатовимірної однокритеріальної оптимізації, та на задачі інженерного спрямування для знаходження оптимальних значень параметрів для моделювання зварної балки.

Перелік використаних джерел

1. *Rao S. S.* Engineering Optimization: Theory and Practice, Fourth Edition. — 2009. — URL: https://onlinelibrary.wiley.com/doi/book/10.1002/9780470549124.
2. *Pham D.* The Bees Algorithm – A Novel Tool for Complex Optimisation Problems // Intelligent Production Machines and Systems. — 2006.
3. *Yang X.-S.* A New Metaheuristic Bat-Inspired Algorithm // Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). — 2010.
4. *Seyedali Mirjalili A. L.* The Whale Optimization Algorithm // Advances in Engineering Software. — 2016.
5. *Wikipedia.* Multidimensional functions for testing global optimization methods and algorithms. — 2024. — [Online; accessed 25-Match-2024]. https://en.wikipedia.org/wiki/Test_functions_for_optimization.