

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Програмна інженерія та інформаційні технології управління»

Індивідуальне домашнє завдання №1
з предмету «Нечіткі технології обробки інформації»

Виконав:

ст. групи КН-34г

Чепурний А. С.

Перевірів:

проф. каф. ПШТУ

Голоскоков О. Є.

Харків 2018

ЗМІСТ

1	Теоретичні основи генетичних алгоритмів	3
1.1	Елементи теорії генетики	3
1.2	Генетичні алгоритми	3
1.2.1	Основні поняття генетичних алгоритмів	3
1.2.2	Опис алгоритму	4
2	Постановка задачі	6
3	Виконання задачі	7
3.1	Дослідження цільової функції	7
3.2	Проектування генетичного алгоритму	7
3.3	Аналіз результатів роботи генетичного алгоритму	8
	Висновки	11
	Список джерел інформації	12

1 ТЕОРЕТИЧНІ ОСНОВИ ГЕНЕТИЧНИХ АЛГОРИТМІВ

1.1 Елементи теорії генетики

Теорія еволюції Дарвіна передбачає, що еволюція природи — це процес оптимізації живих організмів. Відповідно до цієї теорії, природний відбір — це головна умова еволюції в природі.

Суть природного відбору полягає в тому, що більш адаптовані біологічні істоти мають більше шансів на виживання та розмноження. Таким чином, потімки «кращих» біологічних індивідів будуть більш пристосовані порівняно зі своїми ровесниками. Отже, через декілька (десятків, сотень, тисяч) поколінь частка адаптованості збільшується в загальній масі [1].

1.2 Генетичні алгоритми

1.2.1 Основні поняття генетичних алгоритмів

Генетичний алгоритм — це еволюційний алгоритм пошуку, що використовується для вирішення задач оптимізації і моделювання шляхом послідовного підбору, комбінування і варіації шуканих параметрів з використанням механізмів, що нагадують біологічну еволюцію.

Генетичний алгоритм є комбінацією переборного і градієнтного методів вирішення задач. Механізми схрещування і мутації в якомусь змісті реалізують переборну частину методу, а добір кращих розв'язань — градієнтний спуск [2].

Характерними відмінностями генетичного алгоритму є:

- обробка не параметрів задачі, а їх закодовану форму;
- пошук рішення ґрунтується не на точці, а на деякій популяції;
- при оптимізації використовують тільки цільову функцію;
- правила відбору особин популяції імовірнісні.

Основні терміни генетичного алгоритму:

- особа — деяке рішення задачі;
- хромосома (*chromosome*) — числовий вектор, що відповідає особі;

- ген (*gen*) — це атомарний елемент хромосоми;
- популяція (*population*) — сукупність особин виду, множина допустимих розв'язків;
- покоління (*generation*) — чергова популяція;
- функція адаптації (*fitness function*) — порівняльний показник якості адаптації особи;
- оператор селекції (*selection*) — це елемент генетичного алгоритму, який відповідає за спосіб вибору осіб, які беруть участь у створенні наступного покоління;
- оператор схрещування (*crossover*) — це операція, при якій із двох хромосом породжується одна чи декілька нових хромосом;
- оператор мутації (*mutation*) — це операція перетворення хромосоми, що випадково змінює одну чи декілька її генів.

1.2.2 Опис алгоритму

Задача кодується таким чином, щоб її рішення могло бути представлене у виді вектора. Випадковим чином створюється деяка кількість початкових векторів. Вони оцінюються з використанням функції адаптації, у результаті чого кожному векторові привласнюється визначене значення адаптації, що визначає ймовірність виживання організму, представленого даним вектором. Після цього з використанням отриманих значень пристосованості вибираються вектори (селекції), допущені до схрещування. До цих векторів застосовуються генетичні оператори (у більшості випадків схрещування та мутація), створюючи в такий спосіб наступне покоління. Особи наступного покоління також оцінюються, потім виробляється селекція, застосовуються генетичні оператори і т.д. Так моделюється еволюційний процес, що продовжується кілька життєвих циклів (поколінь), поки не буде виконаний критерій зупинки алгоритму. Таким критерієм може бути:

- отримання глобального, або субоптимального рішення;
- вичерпання числа поколінь, відпущених на еволюцію;

– вичерпання часу, відпущеного на еволюцію.

Таким чином, можна виділити наступні етапи генетичного алгоритму:

- 1 Створення початкової популяції.
- 2 Обчислення функцій адаптації для особей популяції.
- Початок циклу.
 - 1 Селекція.
 - 2 Схрещування і/або мутація.
 - 3 Обчислення функцій пристосованості для всіх особей.
 - 4 Формування нового покоління.
- Якщо виконуються умови зупинки, то алгоритм зупиняється, інакше перехід до початку циклу.

2 ПОСТАНОВКА ЗАДАЧІ

Завданням роботи є знаходження мінімального рішення задачі, яке виражене цільовою функцією на області допустимих рішень, використовуючи генетичні алгоритми та пакет програмного забезпечення MATLAB.

Необхідно провести дослідження створеного алгоритму, а саме оцінити точність результату роботи алгоритму та визначити вплив розміру популяції.

Згідно з індивідуальним завданням, цільовою функцією є ступенчата функція Де Йонга

$$f(x) = \sum_{i=1}^n [|x_i|], \quad x \in (-5.12; 5.12),$$

де $|x_i|$ — модуль цілої частки x_i .

3 ВИКОНАННЯ ЗАДАЧІ

3.1 Дослідження цільової функції

Графік ступінчастої функції Де Йонга для двох параметрів представлено на рисунку 3.1.

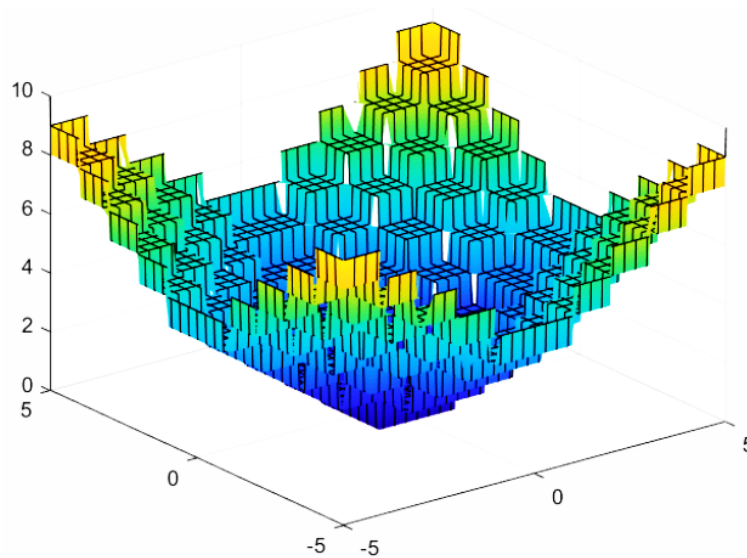


Рисунок 3.1 — Графік функції Де Йонга

Функція має глобальний мінімум в області $x, y \in (-1; 1)$.

3.2 Проектування генетичного алгоритму

Для розрахунку було взято популяцію розміром в 80 осіб, що генерується автоматично у діапазоні згідно з завданням.

Функція селекції — стохастична (*selectionstochunif*) — кожна особа розміщується на колесі рулетки та має розмір секції, пропорційний до її функції адаптації; алгоритм один раз вибирає випадкове число і кожен раз крутить колесо на це число градусів, вибираючи нову популяцію [3].

Функція мутації — мутація по Гаусу (*mutationgaussian*) — до числа додається випадкове число з розподілу Гаусу [3].

Функція схрещування — рівномірний кросовер (*crossoverscattered*) — кожний ген має власну імовірність бути успадкованим від батьків [3].

Скрипт для середовища MATLAB:

```

options = gaoptimset;
options = gaoptimset(options, 'Display', 'iter');
options = gaoptimset(options, 'PopulationSize', 80);
options = gaoptimset(options, 'Generations', 15);
options = gaoptimset(options, 'PlotFcn', @gaplotbestf);

LB = [-5.12]; % Lower bound
UB = [+5.12]; % Upper bound
x = ga(@dejong, 2, [], [], [], [], LB, UB, [], [], options)

function y = dejong(x)
    r = abs(fix(x));
    y = sum(r, 2);
end

```

3.3 Аналіз результатів роботи генетичного алгоритму

Результатом виконання скрипту є

$$x_1 = 0.234, x_2 = -0.781,$$

$$f\left(\begin{bmatrix} x_1 & x_2 \end{bmatrix}\right) = 0,$$

отже генетичний алгоритм успішно знайшов глобальний мінімум функції. Вихідна інформація роботи алгоритму представлена у таблиці 3.1.

Таблиця 3.1 — Процес виконання алгоритму

№	Кількість особин	Найкращий $f(x)$	Середній $f(x)$	Застійні покоління
1	160	0	4.487	0
2	240	0	2.763	1
3	320	0	1.9	2
4	400	0	1.212	3
5	480	0	0.825	4
6	560	0	0.6	5
7	640	0	0.225	6
8	720	0	0.0875	7
9	800	0	0.075	8
10	880	0	0.075	9
11	960	0	0.075	10
12	1040	0	0.075	11
13	1120	0	0	12
14	1200	0	0	13
15	1280	0	0	14

Функція приймає значення глобального мінімуму на першому поколінні, отже доречним буде зменшити розмір популяції. Процес оптимізації був перерваним через досягнення заданої кількості поколінь. Графік динаміки функції адаптації представлено на рисунку 3.2.

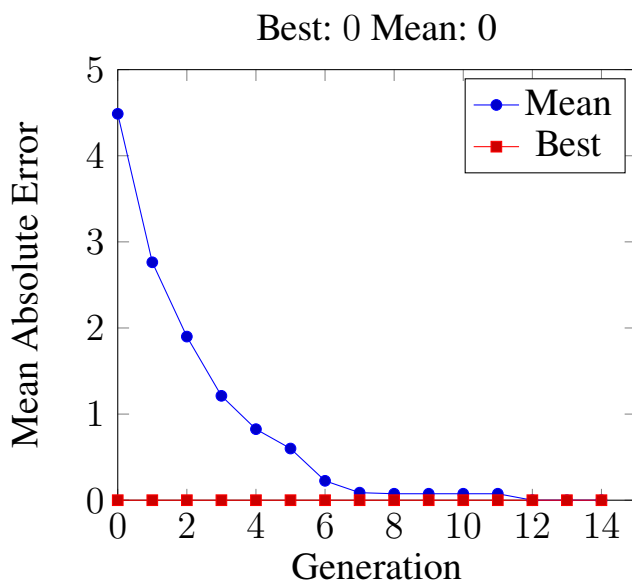


Рисунок 3.2 — Графік динаміки функції адаптації (розмір популяції 80)

Для перевірки припущення про зменшення розміру популяції, розмір було змінено з 80 до 5. При таких умовах глобальний мінімум був досягнутий на 2 генерації (рисунок 3.3).

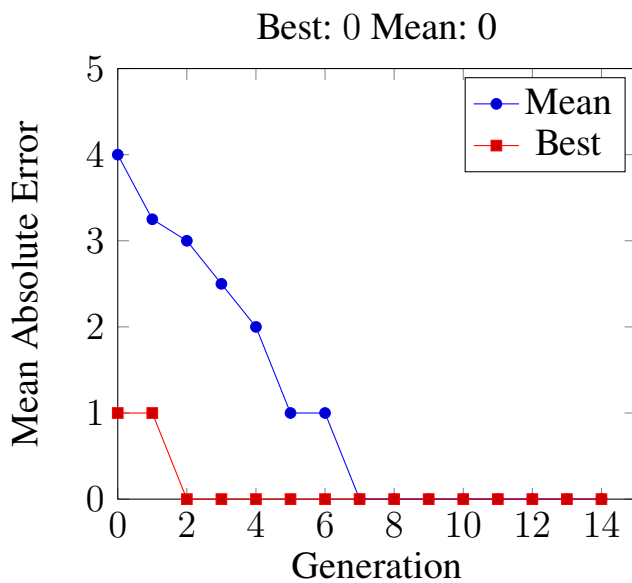


Рисунок 3.3 — Графік динаміки функції адаптації (розмір популяції 5)

ВИСНОВКИ

За допомогою генетичних алгоритмів у пакеті MATLAB була вирішена задача знаходження мінімуму функції і досліджено вплив розміру популяції на точність обчислень. З використанням пакету MATLAB можна швидко дослідити роботу генетичного алгоритму, побудувати графік динаміки фітнес-функції.

СПИСОК ДЖЕРЕЛ ІНФОРМАЦІЇ

- 1 І. О. Огінова та О. Є. Пахомов. “Теорія еволюції”. В: ДНУ, 2011.
- 2 А. Ю. Конончук. *Нейронні мережі та генетичні алгоритми*. 2008.
- 3 *Genetic Algorithm Options*. MATLAB & Simulink, 2017. URL: <https://www.mathworks.com/help/gads/genetic-algorithm-options.html>.