



Звіт
до лабораторної роботи № 2

на тему:

"МЕТОДИ ЕВОЛЮЦІЙНОГО ПОШУКУ"

з курсу “ Методи нечіткої логіки та еволюційні алгоритми при
автоматизованому проектуванні ”

Виконав:
студент групи КНСП-11
Дербіж А. В.

Перевірив: викладач каф. САП,
асист. Кривий Р.З

МЕТА РОБОТИ

Ознайомитися з основними теоретичними відомостями за темою роботи. Вивчити роботу функції `ga` пакету `Matlab`. Ознайомитися з принципом роботи методів еволюційного пошуку, та їх реалізації в `MatLab`.

КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Еволюційні алгоритми — напрям в штучному інтелекті (розділ еволюційного моделювання), що використовує і моделює біологічну еволюцію. Розрізняють різні алгоритми: генетичні алгоритми, еволюційне програмування, еволюційні стратегії, системи класифікаторів, генетичне програмування тощо. Всі вони моделюють базові положення в теорії біологічної еволюції — процеси відбору, мутації і відтворення. Поведінка агентів визначається довкіллям. Множину агентів прийнято називати популяцією. Така популяція еволюціонує відповідно до правил відбору відповідно до цільової функції, що задається довкіллям. Таким чином, кожному агентові (індивідуумові) популяції призначається значення його придатності в довкіллі. Розмножуються лише найпридатніші види. Рекомбінація і мутація дозволяють агентам змінюватись і пристосовуватися до середовища. Такі алгоритми належать до адаптивних пошукових механізмів.

Моделювання еволюції можна розділити на дві категорії:

1. Системи, які використовують лише еволюційні принципи. Вони успішно використовувалися для завдань виду функціональної оптимізації і можуть легко бути описані на математичній мові. До них належать еволюційні алгоритми, такі як еволюційне програмування, генетичні алгоритми, еволюційні стратегії.
2. Системи, які є біологічно реалістичніші, але які не виявилися корисними в прикладному сенсі. Вони більше схожі на біологічні системи і менш направлені на вирішення технічних завдань. Вони володіють складною і цікавою поведінкою, і, мабуть, незабаром отримають практичне вживання. До цих систем відносять так зване штучне життя.

Еволюційні алгоритми, в сучасному вигляді, з'явилися наприкінці 1960-х на початку 1970-х (існують посилання на раніші дослідження). Еволюційні алгоритми можна поділити на три групи:

- Еволюційне програмування: фокусується більше на адаптації індивідів, аніж на еволюції генетичної інформації. Зазвичай, еволюційне програмування застосовує безстатеве розмноження та мутації, тобто, внесення невеликих змін в поточний розв'язок та методи селекції ґрунтовані на прямій конкуренції.
- Еволюційні стратегії (ЕС): Важливою особливістю еволюційних стратегій є використання само-адаптивних механізмів для контролю процесу мутації. Ці механізми зосереджені не лише на еволюції шуканих розв'язків, а й на еволюції параметрів мутації.
- Генетичний алгоритм (ГА): Основною особливістю генетичних алгоритмів є використання оператора рекомбінації (схрещення) як основного механізму пошуку. Це ґрунтується на припущенні, що частини оптимального розв'язку можуть бути знайдені незалежно та рекомбіновані для отримання кращого розв'язку.

Еволюційні алгоритми знайшли широке застосування. Однією з найпоширеніших галузей застосування є комбінаторна оптимізація. Так, еволюційні алгоритми з успіхом було застосовано для розв'язання класичних NP-повних проблем, таких як задача комівояжера, задача пакування рюкзака, розбиття чисел, максимальна незалежна множина та розфарбовування графів.

До інших не класичних задач, для розв'язання яких застосовано еволюційні алгоритми, належать планування, складання розкладів, обчислення маршрутів, задачі розташування та транспортування. Також еволюційні алгоритми використовують для оптимізації структур та електронних схем, в медицині та в економіці.

В останні роки активно розвивається використання еволюційних алгоритмів для передбачення кристалічних структур з допомогою програмного забезпечення USPEX. Приклади передбачених структур і матеріалів можна знайти на сайті.

Можливість використання еволюційних алгоритмів у галузі музики активно досліджується насамперед у Австрії, а саме при спробі моделювання та відтворення гри на музичних інструментах видатними особистостями різних епох.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

Розробити за допомогою пакету Matlab програмне забезпечення, що реалізує 2 методи еволюційного пошуку. Основні еволюційні оператори для реалізації еволюційних методів обрати з таблиці 1 відповідно до варіанту.

Варіант №5

5	1	рулетка	однорідне	гауссовська
	2	ранжирування	порівняльне	нерівномірна

Таб.1 Еволюційні оператори для виконання завдання

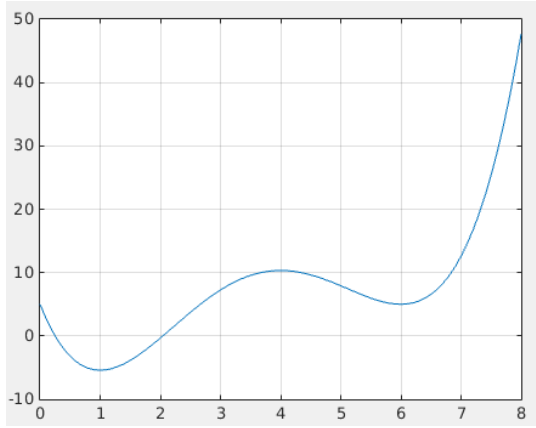
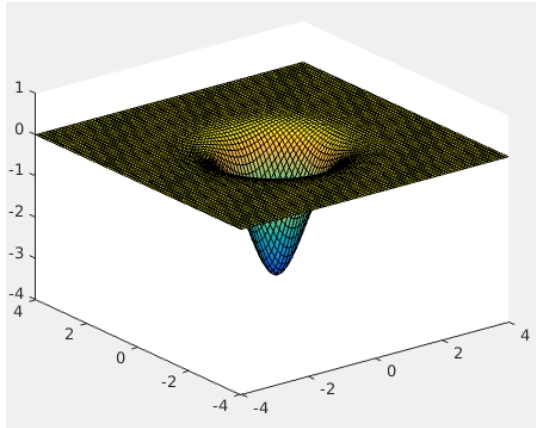
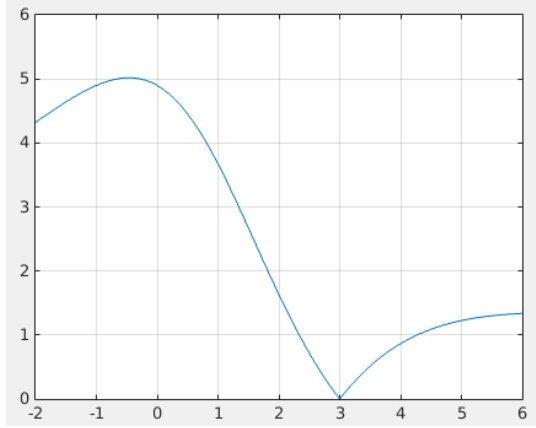
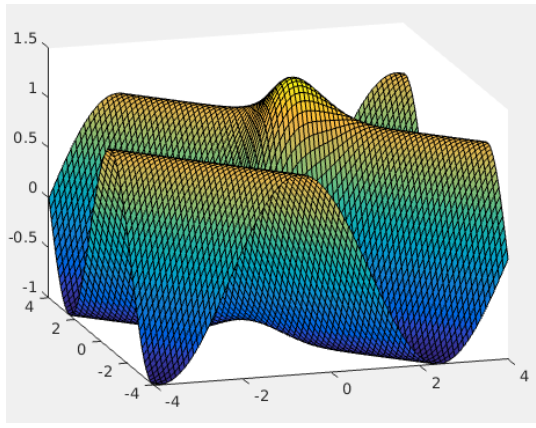
Інші параметри, необхідні для еволюційного пошуку, обрати самостійно. Вибір параметрів обґрунтувати.

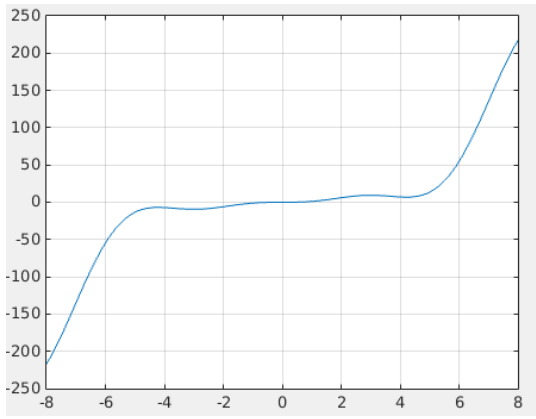
Виконати тестування розробленого програмного забезпечення за допомогою вирішення задач оптимізації тестових функцій. Тестові функції у і (не менше п'яти) для виконання тестування програми обрати самостійно. Вибір тестових функцій обґрунтувати.

Порівняти одержані результати оптимізації різних функцій за допомогою обох реалізованих еволюційних методів. Результати порівняльного аналізу звести до таблиці, попередньо розробивши систему критеріїв порівняння методів еволюційного пошуку.

РЕЗУЛЬТАТИ ВИКОНАННЯ РОБОТИ

Обрані функції:

№	Функція	Графік	Ціль
1	$f(x) = 5 - 24x + 17x^2 - \frac{11}{3}x^3 + \frac{1}{4}x^4$		min
2	$f(x, y) = (y - 3)\exp(-x^2 - y^2)$		min
3	$f(x) = \frac{6\sqrt[3]{6(x-3)^2}}{(x-1)^2 + 8}$		max
4	$f(x, y) = \exp(-x^2 - y^2) + \sin(x + y)$		max

5	$f(x) = 0.3x^3 + x^2 \cdot \sin(x)$		min
---	-------------------------------------	---	-----

Порівняння генетичних алгоритмів здійснюється при сталих значеннях діапазону вибору початкової популяції, розміру популяції та максимальної кількості ітерацій. Результати порівнюються за часом виконання алгоритму та точністю отриманих значень. Оскільки генетичні алгоритми мають випадковий характер, то за еталонне значення приймається мінімальне з отриманих після декількох застосувань алгоритму, а за результат — середнє арифметичне. Чим більшою буде похибка, тим ймовірніше, що повторне застосування обраного алгоритму не дасть бажаного результату.

Функція з налаштуваннями ga згідно Задачі 1

```
function [ time, fval ] = ga_var1( func, nvars, range, population_size, generations )
% Задача 1 %
options = gaoptimset();
options = gaoptimset(options, 'SelectionFcn', @selectiontournament);
options = gaoptimset(options, 'CrossoverFcn', @crossoversinglepoint);
options = gaoptimset(options, 'MutationFcn', @mutationgaussian);
%options = gaoptimset(options, 'PopInitRange', range);
options = gaoptimset(options, 'Generations', generations);
options = gaoptimset(options, 'PopulationSize', population_size);

tic
[fx, fval] = ga(func, nvars, options);
time = toc;

end
```

Функція з налаштуваннями ga згідно Залачі 2

```
function [ time, fval ] = ga_var2( func, nvars, range, population_size, generations )
% Задача 2 %
options = gaoptimset();
options = gaoptimset(options, 'SelectionFcn', @selectionroulette);
options = gaoptimset(options, 'CrossoverFcn', @crossoverintermediate);
options = gaoptimset(options, 'MutationFcn', @mutationuniform);
options = gaoptimset(options, 'PopInitRange', range);
options = gaoptimset(options, 'Generations', generations);
options = gaoptimset(options, 'PopulationSize', population_size);
```

tic

[x, fval] = ga(func, nvars, options);

time = toc;

end

Повертає статистичні дані для вказаної функції

function [ret_fun] = ga_static(func, nvars, k)

RANGE = [-50; 50];

POPULATION = 20;

GENERATIONS = 100;

F(1) = {@ga_var1};

F(2) = {@ga_var2};

N = 10;

for var = 1:1:2

time = []; fval = [];

ga_var = F{var};

for i = 1:1:N

[time_ret, fval_ret] = ga_var(func, nvars, RANGE, POPULATION, GENERATIONS);

time(i) = time_ret;

fval(i) = fval_ret;

end;

ret_fun(1, var) = roundn(mean(time), -4);

*ret_fun(2, var) = k*roundn(mean(fval), -4);*

*ret_fun(3, var) = k*roundn(min(fval), -4);*

*ret_fun(4, var) = abs((ret_fun(2, var) - ret_fun(3, var))/...
ret_fun(3, var))*100;*

end;

end

Функція №1 (min)			
	Задача 1	Задача 2	Реальне значення
Час виконання	0,3234	0,1146	-5,4167
Середнє значення	-5,4086	-5,4114	
Мінімальне значення	-5,4166	-5,4167	
Відносна похибка (%)	0,1477	0,0978	

Функція №2 (min)			
	Задача 1	Задача 2	Реальне значення
Час виконання	0,3153	0,1684	-3,0801
Середнє значення	-2,0262	-2,2284	
Мінімальне значення	-3,0722	-3,0800	
Відносна похибка (%)	34,0473	27,6494	

Функція №3 (max)			
	Задача 1	Задача 2	Реальне значення
Час виконання	0,2932	0,1142	5,0191
Середнє значення	5,0181	5,0182	
Мінімальне значення	5,0191	5,0191	
Відносна похибка (%)	0,0199	0,0179	

Функція №4 (max)			
	Задача 1	Задача 2	Реальне значення
Час виконання	0,2812	0,1375	1,4506
Середнє значення	1,0905	1,0905	
Мінімальне значення	1,3304	1,4506	
Відносна похибка (%)	18,0322	24,8242	

Функція №5 (min)			
	Задача 1	Задача 2	Реальне значення
Час виконання	0,2812	0,1135	
Середнє значення	-8,27E+07	-2,72E+04	---
Мінімальне значення	-1,13E+08	-3,26E+04	
Відносна похибка (%)	26,9582	16,4903	

ВИСНОВОК

На цій лабораторній роботі, я ознайомився з основними теоретичними відомостями за темою роботи. Вивчив роботу функції ga пакету Matlab. Ознайомився з принципом роботи методів еволюційного пошуку, та їх реалізації в MatLab.

Початкова вибірка вибиралася із широкого діапазону значень, щоб віддалити її від оптимуму. Оптимум досягається швидше, якщо використовувати параметри із задачі 2, при цьому як мінімальне, так і середнє значення ближчі до реального оптимуму.