|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **НУЛП, ІКНІ, САП** | | Тема | оцінка | підпис |
| СПКс-11 | 2 | МЕТОДИ ЕВОЛЮЦІЙНОГО ПОШУКУ |  |  |
| Шкіндер А. В. | |
| Методи нечіткої логіки та еволюційні алгоритми при автоматизованому проектуванні | | Викладач: | |
| Кривий Р.З. | |

**Мета роботи:**ознайомитися з основними теоретичними відомостями про методи еволюційного пошуку. Вивчити роботу функції ga пакету Matlab.

**Завдання:**Розробити за допомогою пакету Matlab програмне забезпечення, що реалізує 2 методи еволюційного пошуку.





В – 14

**Виконання лабораторного завдання**

**Тестові функції.**

1. Функція

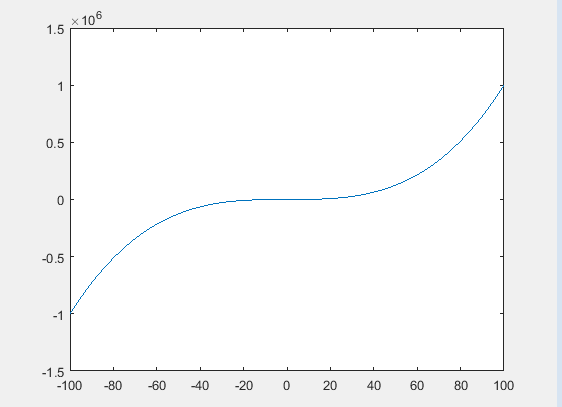
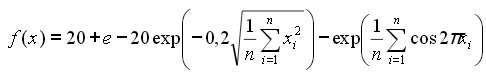


Рис.1. Функція однієї змінної.

Мінімум функції знаходиться при x = 0, а значення функції рівне 0.

2) Функція Екклі:

**

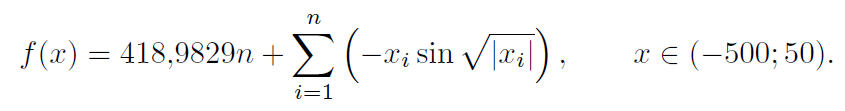
Для двох змінних вона буде мати вигляд:



Рис.2. Функція Екклі для двох змінних.

Мінімум функції знаходиться в точці (0,0), а значення функції 0.

* 3) Функція Швефеля:



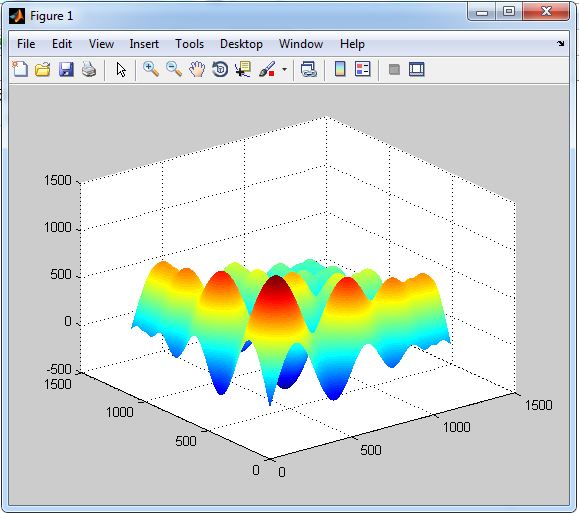
[X1, X2] = meshgrid(-500:.5:50);

Z =418.9829 -X1.\*sin(sqrt(abs(X1)))-X2.\*sin(sqrt(abs(X2)));

mesh(Z)

x = fminsearch(@fun3,[0,50])

y = fun3(x)



*Рис.3. Функція Швефеля*

Мінімум функції знаходиться при x1 = 5.2392 , х2 = 65.5479, а значення функції рівне y = 351.4026.

**Знаходження мінімуму функції за допомогою генетичного алгоритму.**

Механізм роботи з генетичними алгоритмами в середовищі MATLAB реалізований двома способами:  
1.Виклик функції генетичних алгоритмів  
2.Використання комплекту GeneticAlgorithmTool

Я використовував другий спосіб. В 1 задачі відбір Рулетка(Roulette), схрещування арифметичне(Arithmetic), мутація проста(Uniform). В 2 задачі відбір пороговий (Stochastic uniform), схрещування порівняльне (реалізовувалось самостійно crossoverComparative), мутація випадкова (реалізовувалось самостійно mutationRandom).

**Задача 1**

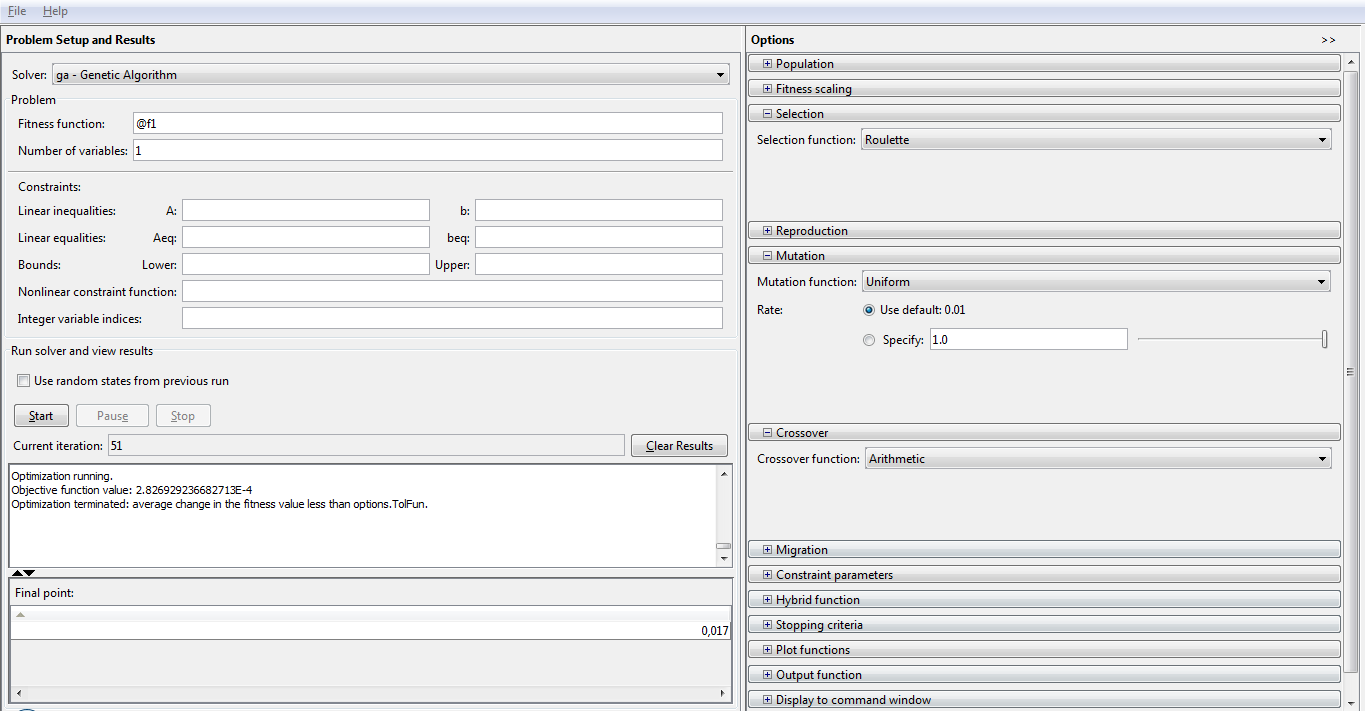


Рис.4. Налаштування комплекту GeneticAlgorithmTool для задачі 1.

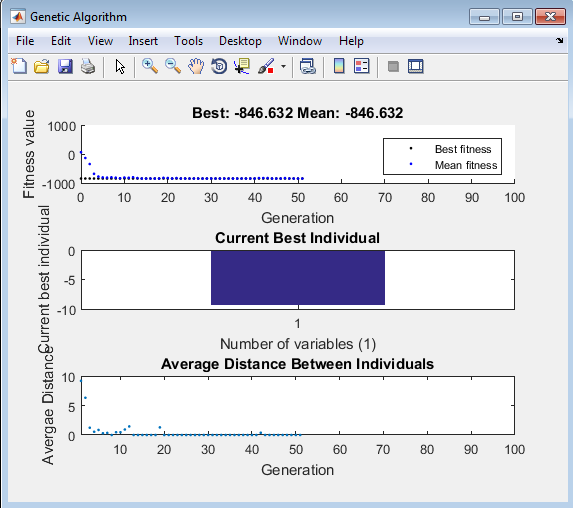


Рис.5. Графіки знаходження мінімуму функції однієї змінної для 1 задачі.

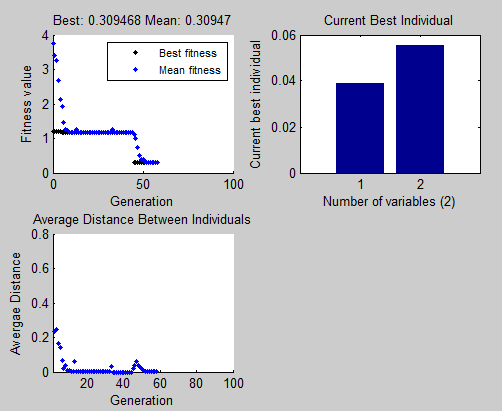


Рис.6. Графіки знаходження мінімуму функції Екклі для 1 задачі.

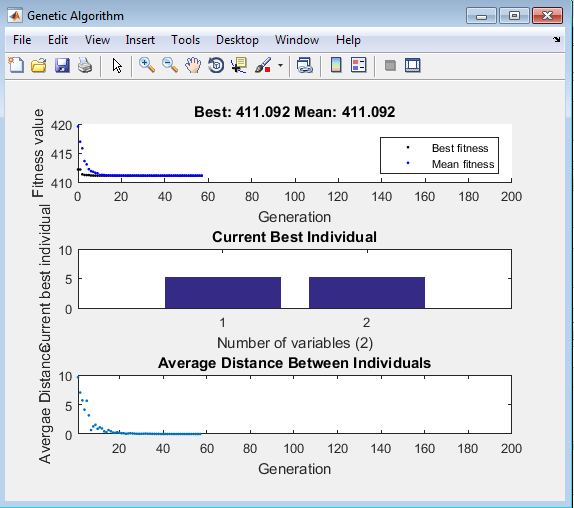


Рис.7. Графіки знаходження мінімуму функції Швефеля для 1 задачі.

**Задача 2**

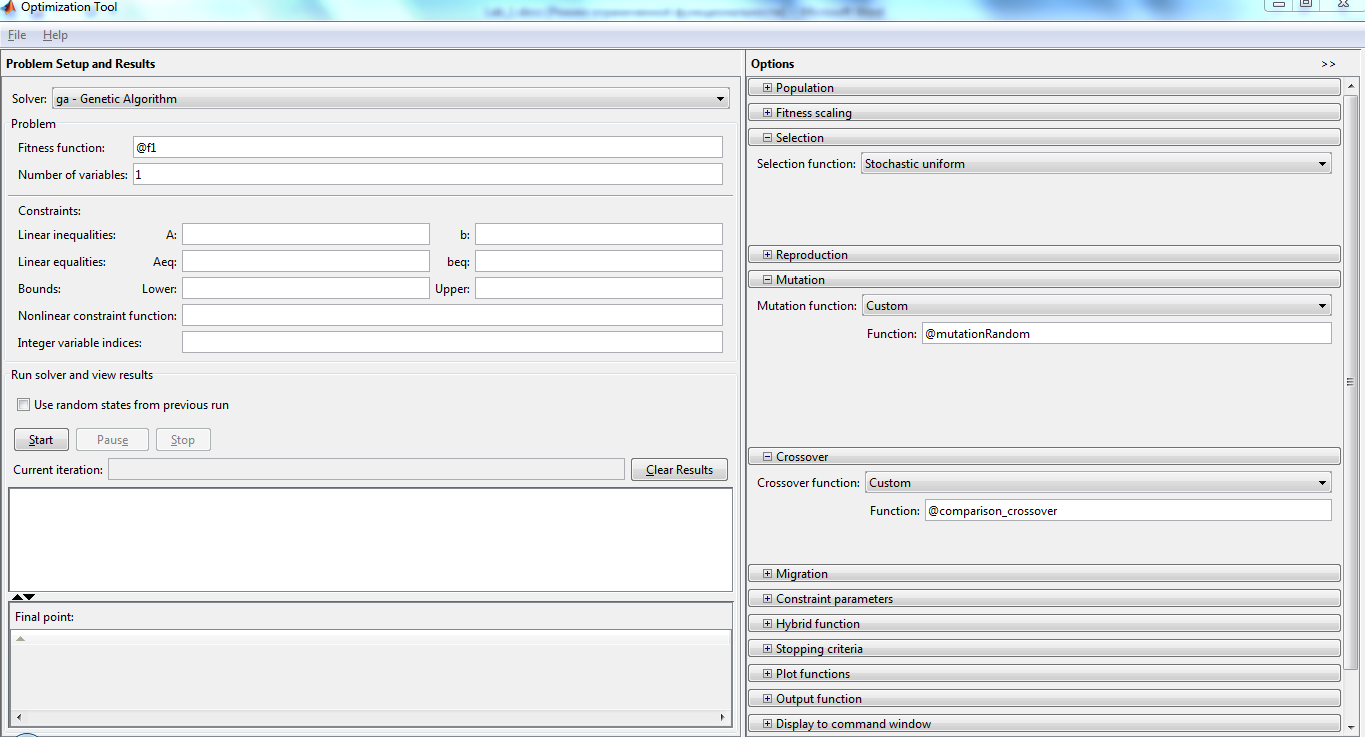


Рис.8. Налаштування комплекту GeneticAlgorithmTool для задачі 2.

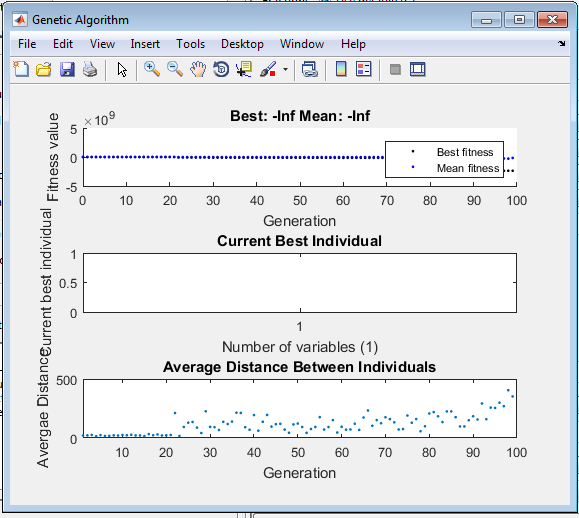


Рис.9. Графіки знаходження мінімуму функції однієї змінної для 2 задачі.

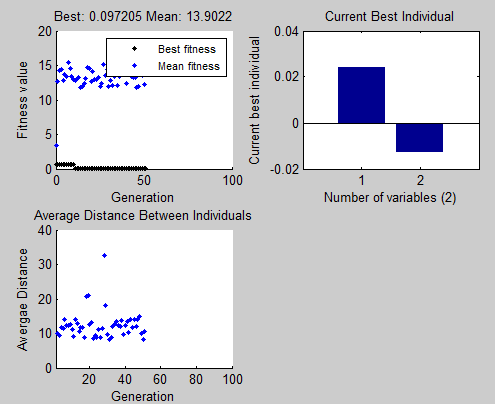


Рис.10. Графіки знаходження мінімуму функції Екклі для 2 задачі.

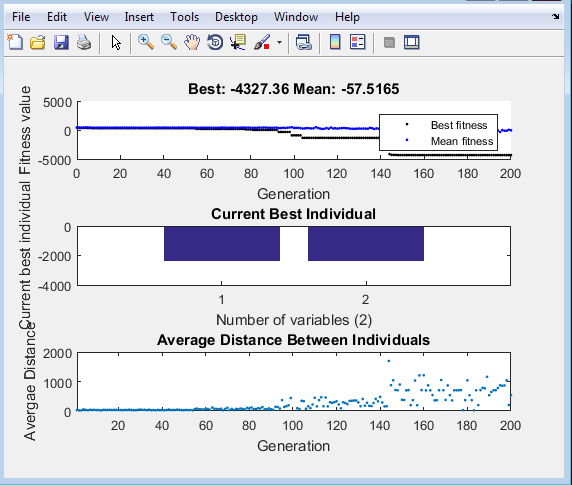


Рис.11. Графіки знаходження мінімуму функції Швефеля для 2 задачі.

Табл.1.Таблиця похибок генетичного алгоритму для тестових функцій

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Відбір : турнірний  Схрещування арифметичне  Мутація: проста | | | Відбір : турнірний  Схрещування: діагональне  Мутація: випадкова | | |
| Назва функції  (к-сть змінних) | (1) | Екклі (2) | Швефеля  (2) | (1) | Екклі (2) | Швефеля  (2) |
| Мінімум функції | 0 | 0 | 351.4 | 0 | 0 | 351.4 |
| Мінімум функції за допомогою ГА (середнє при 5 запусканнях | 0,01 | 0,30 | 411.11 | 0.001 | 0.097 | 417.3 |
| Похибка,% | 1 | 30 | 15.75 | 0.1 | 9.7 | 18.59 |

**Код реалізованих операторів**

**mutationRandom.m**

function mutationChildren = mutationRandom(parents ,options,NVARS, ...

FitnessFcn, state, thisScore,thisPopulation); %випадкова мутація

T=100;

ti=state.Generation;

a=2.0\*randn\*((log10(T)-log10(ti))/log10(T));

b=0;

deltaH=randn/(a-b)+b;

thisPopulation

for i=1:length(parents)

child=thisPopulation(parents(i),:);

mutationPoints=find(rand(1,length(child))<0.01);

child(mutationPoints)=~child(mutationPoints);

mutationChildren(i,:)= child+deltaH;

end

end

**crossoverComparative.m**

function xoverKids = crossoverComparative(parents, options, nvars, FitnessFcn, ...

unused,thisPopulation)%порівняльне схрещування

leng = length(parents)/2;

for j = 1:nvars

for i = 1:leng

if (parents(i) == parents(i + 14))

xoverKids(i,j) = parents(i);

else

k = rand();

if (k <= 0.5) xoverKids(i,j) = parents(i);

else xoverKids(i,j) = parents(i + 14);

end

end

end

end

end

**Код тестових функцій:**

function y=f1(x)  
y = x.^3 + 2.\*x - 7;  
end

function y = fun2(x) % функція Екллі

y = 20 + exp(1) - 20 \* exp(-0.2\*((1/2)\*(x(1).^2 + x(2).^2)).^0.5)-exp((1/2)\*(cos(2\*pi\*x(1))+cos(2\*pi\*x(2))));

end

function y = f3(x) % Швефеля  
y = 418.9829 -x(1).\*sin(sqrt(abs(x(1))))-x(2).\*sin(sqrt(abs(x(2))));  
end

**Висновок:** виконавши дану лабораторну роботу я ознайомився з основними теоретичними відомостями про методи еволюційного пошуку, вивчив роботу функції gatool пакету Matlab і реалізував еволюційні оператори згідно завдання.