МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра “Системи автоматизованого проектування”



Звіт

до лабораторних робіт № 2

з курсу «**Методи нечіткої логіки та еволюційні алгоритми при автоматизованому проектуванні**»

на тему «**МЕТОДИ ЕВОЛЮЦІЙНОГО**

**ПОШУКУ**»

Виконав:

Студент групи КНС-13

Писаренко В.Б.

Прийняв:

асистент каф. САПР

Кривий Р.З.

## Львів – 2017

**1. МЕТА РОБОТИ**

**Мета:** Ознайомитися з основними теоретичними відомостями за темою роботи. Вивчити роботу функції ga пакету Matlab. 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

**2. РЕЗУЛЬТАТ ВИКОНАННЯ**

**Варіант №2**



Робота з GENETIC ALGORITHM TOOL  
  
слід в командному рядку MATLAB виконати команду gatool. Після цього запуститься пакет генетичних алгоритмів і на екрані з'явиться основне вікно утиліти.  
  
Основне вікно gatool  
  
 У основному вікні gatool в полі Fitness function вказується оптимізується функція у вигляді @my\_fun, де my\_fun.m - назва M-файлу, в якому попередньо слід описати оптимізується функція.

Приклади використаних п’яти тестових функцій :  
  
1) z = x (1) ^ 2 - 2 \* x (1) \* x (2) + 6 \* x (1) + x (2) ^ 2 - 6 \* x (2);

2) z = x(1)^3 + 1/2\*x(1)\*x(2) - 2\*x(1) + x(2)^3 - 3\*x(2);

3) z = x(1)^1/2 - 4\*x(2) - 12\*x(1) + 3\*x(2)^3 - 5\*x(1);

4) z = x(1)^2\*x(2) - 1/3\*x(1)\*x(2) + 15\*x(1)\* x(2) - 6\*x(2)\*x

5) z = x(2)^5 + x(1)\*x(2) + 4\*x(2) - 2\*x(1)^1/3 - 10\*x(1);

Використанні функції у зв’язку з простотою їх реалізації. Використовуються лінійні квадратичні функції двох змінних, для швидкості отримання результату алгоритму.

В поле Number of variables вказується вхідний вектор довжини 2.  
В панелі Constraints можна задати обмеження або обмежує нелінійну функцію. Але спробуємо не накладати ніяких обмежень.

Змінюємо основні параметри пакету GATool :  
- Популяція (вкладка Population);  
- Масштабування (вкладка Fitness Scaling);  
- Оператор відбору (вкладка Selection);  
- Оператор репродукції (вкладка Reproduction);  
- Оператор мутації (вкладка Mutation);  
- Оператор схрещування (вкладка Crossover);  
- Перенесення особин між популяціями (вкладка Migration);  
- Спеціальні параметри алгоритму (вкладка Algorithm settings);  
- Завдання гібридної функції (вкладка Hybrid function);  
- Завдання критерію зупинки алгоритму (вкладка Stopping criteria);  
- Висновок різної додаткової інформації по ходу роботи генетичного алгоритму (вкладка Plot Functions);  
- Висновок результатів роботи алгоритму у вигляді нової функції (вкладка Output function);  
- Завдання набору інформації для виведення в командне вікно (вкладка Display to command window);  
- Спосіб обчислення значень оптимізованої і обмежує функцій (вкладка User function evaluation).  
  
Розглянемо докладніше всі перераховані вище вкладки панелі Options і елементи, які вони містять.  
  
У вкладці настройки популяцій обираємо тип математичних об'єктів, до якого будуть ставитися особини всіх популяцій (подвійний вектор)

Також вкладка популяції дозволяє налаштовувати розмір популяції (зі скількох особин складатиметься кожне покоління) і яким чином буде створюватися початкове покоління (Feasible population).   
  
Вкладка Selection дозволяє вибрати оператор відбору батьківських особин на основі даних з функції масштабування. Спочатку обираємо :  
  
- Roulette - імітується рулетка, в якій розмір кожного сегмента встановлюється відповідно до його ймовірністю;

А у другому прикладі:

- Stochastic uniform - будується лінія, в якій кожному з батьків ставиться у відповідність її частина певного розміру (в залежності від ймовірності одного з батьків), потім алгоритм пробігає піт лінії кроками однакової довжини і вибирає батьків в залежності від того, на яку частину лінії потрапив крок.  
  
Вкладка Reproduction уточнює яким чином відбувається створення нових особин. Пункт Elite count дозволяє вказати число особин, які гарантовано перейдуть в наступне покоління. Пункт Crossover fraction вказує частку особин, які створюються шляхом схрещування. Решта частки створюється шляхом мутації.  
  
У вкладці оператора мутації вибирається тип оператора мутації. Доступними параметрами:

У другому випадку для випадкової вибірки обираємо :  
  
  
- Uniform - вибираються випадковим чином компоненти векторів і замість них записуються випадкові числа з допустимого діапазону;

Вкладка Crossover дозволяє вибрати тип оператора схрещування (арифметичне у першому випадку) або порівняльне (Two points),

Інші значення залишаємо за замовчуванням, для кращої реалізації:

У вкладці Migration можна налаштовувати правила, згідно з якими особи будуть переміщатися між подпопуляціямі в межах однієї популяції. Подпопуляціі створюються, якщо в якості розміру популяції вказано вектор, а не натуральне значення. В даній вкладці можна вказати напрямок міграції (forward - в наступну подпопуляцію, both - в попередню і наступну), частку мігруючих особин і частоту міграції (скільки поколінь проходить між міграціями). Якщо створення подпопуляцій не потрібно, цю вкладку завжди варто залишати без змін.  
  
Вкладка спеціальних опцій алгоритму дозволяє налаштовувати параметри рішення системи нелінійних обмежень, що накладаються на алгоритм. Значення параметра Initial penalty визначає початкове числове значення критики алгоритму, Penalty factor використовується як множник цього значення у випадках, коли розробника не влаштовує точність оптимізації або при виході за межі, визначені у вкладці обмежень. Як правило, ці опції детально налаштовуються для вирішення завдань високої складності.  
  
Вкладка Hybrid function дозволяє задати ще одну функцію мінімізації, яка буде використовуватися після закінчення роботи алгоритму. Але ми її не використовуємо.

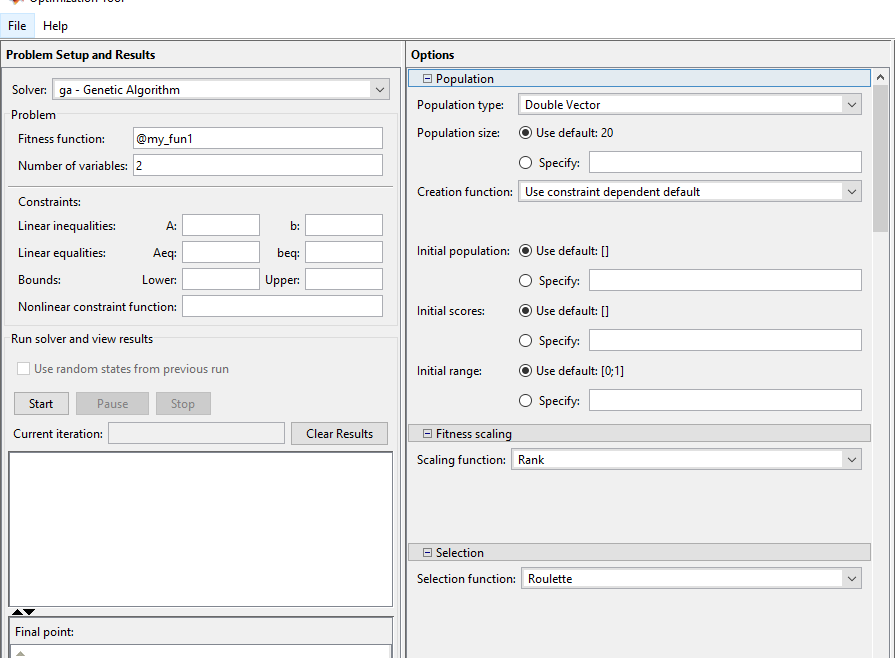


Рис 1 : Результат роботи з пакетом gamatlab

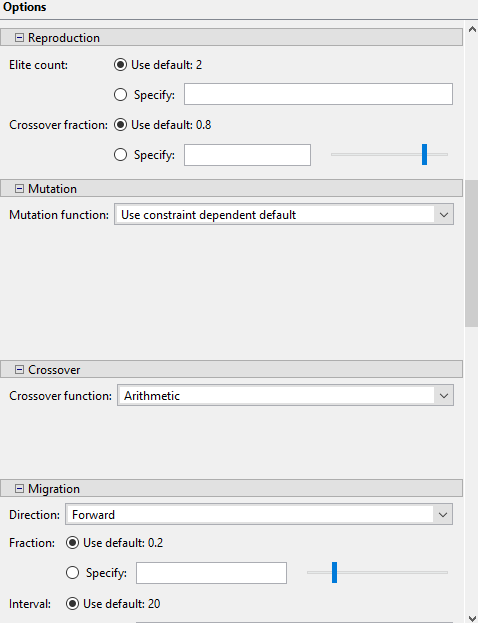


Рис 2 : Результат роботи з пакетом gamatlab

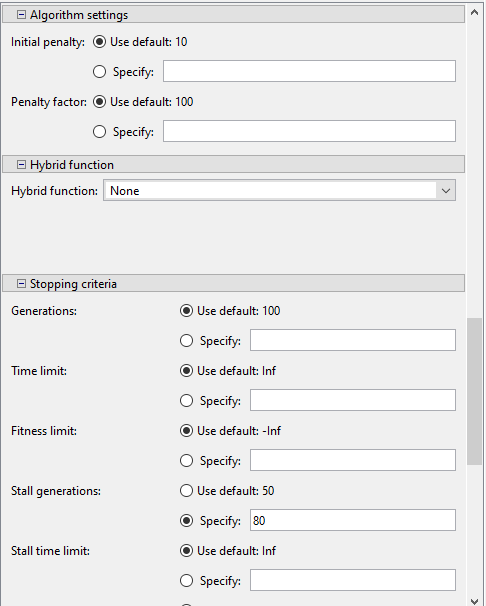


Рис 3 : Результат роботи з пакетом gamatlab

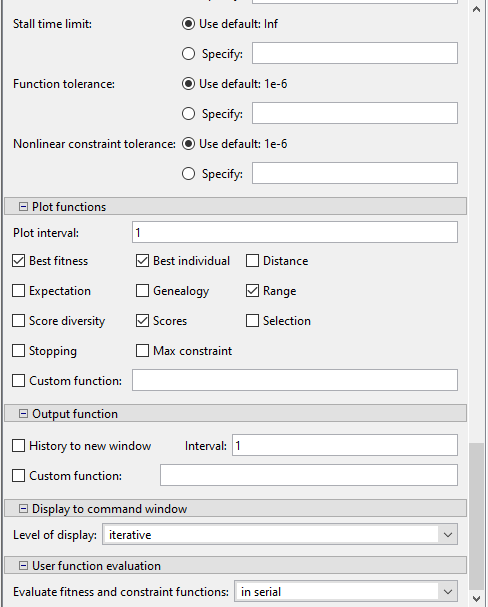


Рис 4 : Результат роботи з пакетом gamatlab

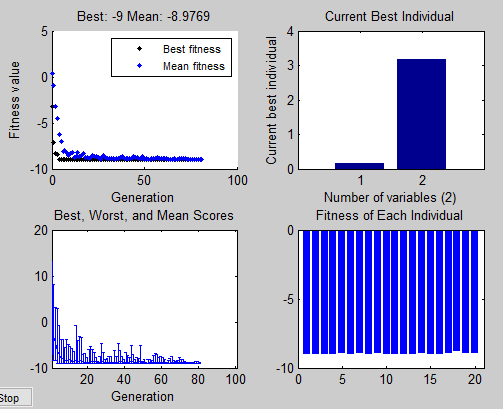


Рис 5 : Результат виконання 1 функції 1 алгоритмом

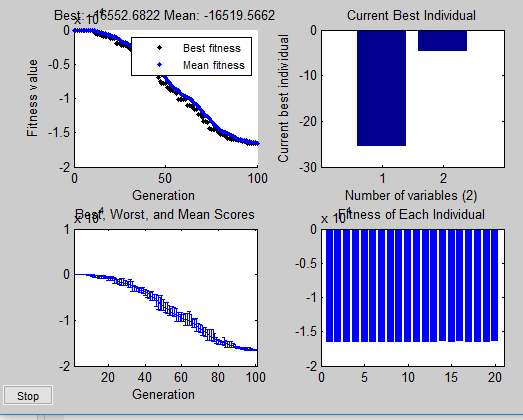


Рис 6 : Результат виконання 2 функції 1 алгоритмом

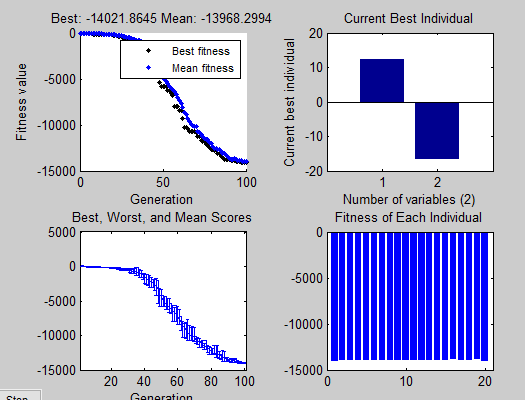


Рис 7 : Результат виконання 3 функції 1 алгоритмом

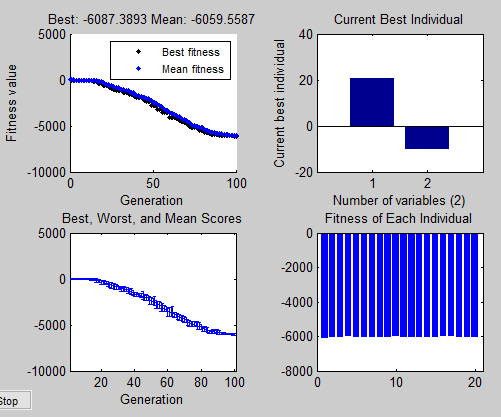


Рис 8 : Результат виконання 4 функції 1 алгоритмом

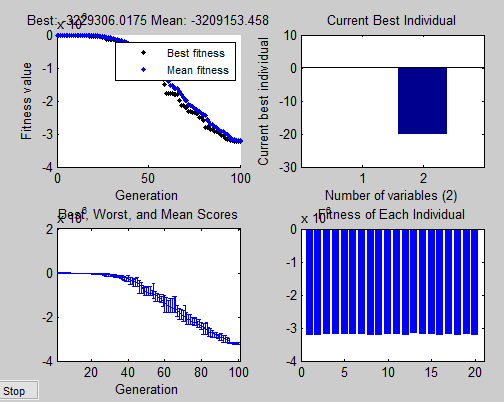


Рис 9 : Результат виконання 5 функції 1 алгоритмом

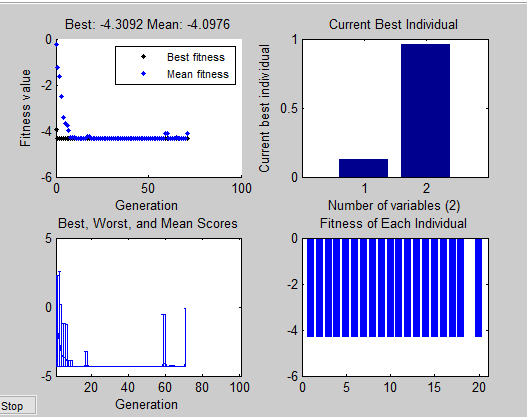


Рис 10 : Результат виконання 1 функції 2 алгоритмом

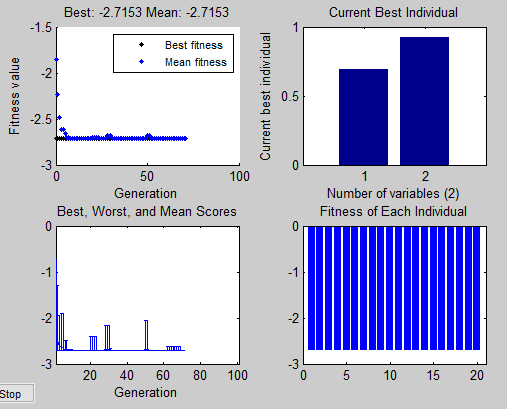


Рис 11 : Результат виконання 2 функції 2 алгоритмом

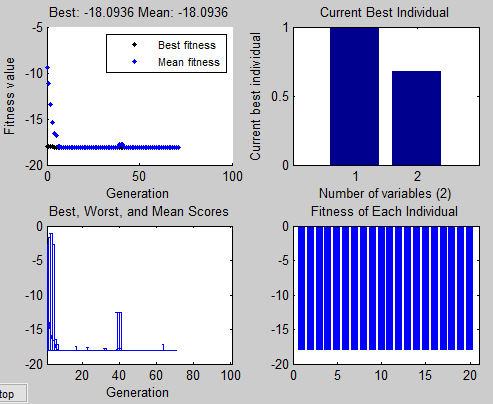


Рис 12 : Результат виконання 3 функції 2 алгоритмом

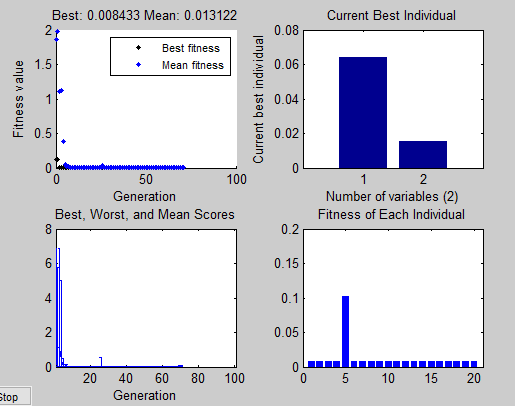


Рис 13 : Результат виконання 4 функції 2 алгоритмом

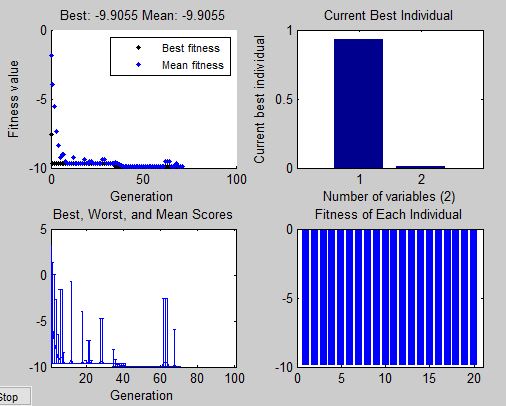


Рис 14 : Результат виконання 5 функції 2 алгоритмом

**4.ВИСНОВОК.**

У результаті проведення лабораторної роботи я вивчив роботу функції ga пакету Matlab. Розробив за допомогою пакету Matlab програмне забезпечення, що реалізує 2 методи еволюційного пошуку.