Міністерство освіти та науки України

Національний університет «Львівська політехніка»

Кафедра систем автоматизованого проектування



Звіт

до лабораторної роботи №2

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

на тему

**«МЕТОДИ ЕВОЛЮЦІЙНОГО ПОШУКУ»**

Варіант №9

Виконав:

ст. гр. КНСП-11

Цогла О.Р.

Прийняв:

Кривий Р.З.

Львів 2020

**Мета роботи:**

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

**Теоретичні відомості:**

Генетичні алгоритми - один із методів розв'язання оптимізаційних задач, що базується на природному відборі (процес що нагадує біологічну еволюцію). Генетичний алгоритм повторює знову і знову модифікацію популяції окремих розв'язків (особин). На кожному кроці алгоритм випадковим чином вибирає із популяції особин, що стануть батьками і будуть використовуватися при створенні нащадків для формування наступного її покоління. Через послідовність поколінь, популяція розвивається в напрямку оптимального розв'язку. Генетичні алгоритми використовують на кожній ітерації три основні типи дій для створення наступного покоління з поточної популяції:

* Селекція (Selection) - вибір особин, що називаються батьками і які беруть участь у створенні наступного покоління.
* Кросовер (Crossover) - комбінація двох батьків для формування нащадків.
* Мутація (Mutation) - внесення випадкових змін до батьківських особин, щоб сформувати нових нащадків.
* Також важливими є поняття:
* Функція пристосованості (Fitness function) - функція, яку необхідно оптимізувати (у Matlab здійснюється пошук її мінімуму).
* Особина (Individual) - значення, для яких можна обчислити функцію пристосованості. Значення цієї функції для певної особини і буде величиною її пристосованості.
* Популяція (Population) - набір особин. Наприклад, якщо розмір популяції 100 і у функції пристосованості є три параметри, то популяцію можна зобразити у вигляді матриці 100 x 3. Популяція може містити однакові особини.
* Покоління (Generation) - популяція, яка формується на кожному черговому кроці роботі генетичного алгоритму.
* Пристосованість (Fitness value) - значення функції пристосованості для конкретної особини. Чим менше це значення, тим краще.

**Індивідульне завдання:**

**Виконання завдання:**

Для тестування розробленого програмного забезпечення було вибрано наступні тестові функції:

1. , min

2. , min

3. , max

4. , max

5. , 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.

**Програма для Matlab:**

Функція з налаштуваннями ga для 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 для Задачі 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

Функція для статистичних даних:

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

**Результати порівняльного аналізу:**

|  |  |  |  |
| --- | --- | --- | --- |
| Функція №1 (min) | | | |
|  | Задача 1 | Задача 2 | Реальне значення |
| Час виконання | 0,3234 | 0,1146 |  |
| Середнє значення | -5,4086 | -5,4114 | -5,4167 |
| 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 |  |
| Середнє значення | -2,0262 | -2,2284 | -3,0801 |
| 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 |  |
| Середнє значення | 5,0181 | 5,0182 | 5,0191 |
| 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 |  |
| Середнє значення | 1,0905 | 1,0905 | 1,4506 |
| 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:**

Під час виконання цієї лабораторної роботи я ознайомився із роботою функції ga пакету Matlab.

В Matlab вже реалізовані методи для використання генетичних алгоритмів під час пошуку оптимумів функції. Нам залишається тільки викликати їх з правильними параметрами і проаналізувати результат.

Початкова вибірка формувалася 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.