Прізвище: Сливка

Ім’я: Наталія

Група: КНСП-11

Дата прийняття роботи

у системі Git: 02.06.2018

Дисципліна: Методи нечіткої логіки та еволюційні алгоритми при автоматизованому проектуванні

Перевірив: Кривий Р.З.

**Звіт до лабораторної роботи № 4**

**«Використання генетичних алгоритмів з бітовим представленням хромосом»**

**МЕТА РОБОТИ**

Ознайомитися з основними теоретичними відомостями, вивчити еволюційні оператори схрещування та мутації, що використовуються при знаходженні значень цільової функції.

**КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ**

У загальному розумінні генетичні алгоритми (genetic algorithms) – це алгоритми, що використовують механізмами еволюції живої природи – природний відбір і генетичне наслідування. Генетичні алгоритми сьогодні застосовуються в різних галузях. Зокрема їх успішно використовують для розв’язування ряду важливих задач в економіці, бізнесі, техніці. З їх допомогою були розроблені промислові проектні рішення, що сприяли значній економії коштів і ресурсів. Фінансові компанії широко використовують ці засоби для прогнозування розвитку фінансових ринків для управління пакетами цінних паперів.

До основних характеристик ГА належать: розмір популяції (population size), оператор селекції (selection), оператор кросовера (crossover) і правила його використання, оператор мутації (mutation) і його параметри, оператор редукції (reduction), правило (критерій) зупинки процесу виконання генетичного алгоритму (stopping criteria). Оператори селекції, кросовера, мутації і редукції ще називають генетичними операторами.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ**

Розробити програмне забезпечення для знаходження значень цільової функції .

**Код головного файлу програми:**

function main()

clc; clear;

XX = -11:1:54;

YY = zeros(size(XX));

for i = 1:1:length(XX)

YY(i) = FitnessFcn(XX(i));

end

plot(XX, YY)

startPopulation = randi([-10,53],10, 1);

disp('Finding MIN');

myGa(startPopulation, @FitnessFcn, -1);

disp('Find MAX');

myGa(startPopulation, @MaxFitnessFcn, 1)

end

function myGa(startPopulation, func, k)

options = gaoptimset(...

'EliteCount', 4, ...

'PopulationSize', 10, ...

'InitialPopulation', startPopulation, ...

'MutationFcn', @MutationFcn, ...

'CrossoverFcn', @CrossoverFcn, ...

'OutputFcns', {@OutputFcn} ...

);

global RET;

RET = struct('generation', 0, 'population', struct, 'fvals', struct);

[x,fval,exitflag,output,population,scores] = ga(func, 1, options);

disp('Start population:');

PrintIter(RET.population.s0, RET.fvals.s0, 1, 5, k);

disp('?Genertion 1:');

PrintIter(RET.population.s1, RET.fvals.s1, 1, 5, k);

disp('Generation 2:');

PrintIter(RET.population.s2, RET.fvals.s2, 1, 5, k);

disp('Result:');

PrintIter(population, scores, 1, 5, k);

fprintf('f(%d) = %d\n\n', x, (-1)\*k\*fval);

end

function PrintIter(curp, fval, nvars, psize, k)

for i = 1:1:psize

fprintf('[');

for j=1:1:nvars

fprintf('%3d ', curp(i,j));

end

fprintf(']=>%d\t', (-1)\*k\*fval(i));

end

fprintf('\n');

end

**Результат виконання лабораторного завдання.**

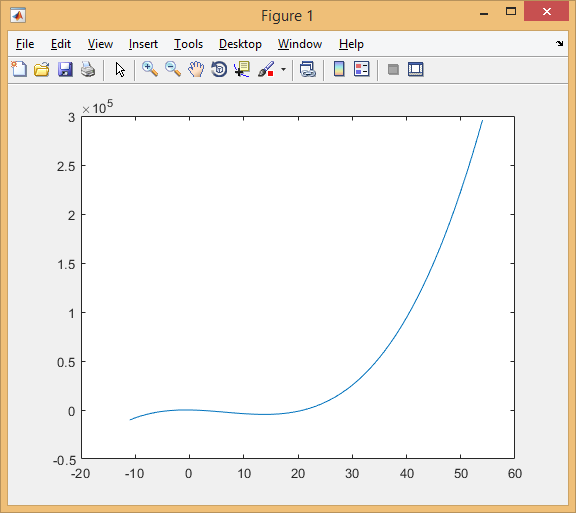


Рис. 1 Графік функції

Finding MIN

Optimization terminated: average change in the fitness value less than options.FunctionTolerance.

Start population:

[ 19 ]=>-2330 [ 4 ]=>-1070 [ 2 ]=>-358 [ 44 ]=>137570 [ 37 ]=>68032

?Genertion 1:

[ 19 ]=>-2330 [ 19 ]=>-2330 [ 4 ]=>-1070 [ -4 ]=>-766 [ 3 ]=>-682

Generation 2:

[ 19 ]=>-2330 [ 19 ]=>-2330 [ 4 ]=>-1070 [ -4 ]=>-766 [ 19 ]=>-2330

Result:

[ 13 ]=>-4472 [ 13 ]=>-4472 [ 13 ]=>-4472 [ 13 ]=>-4472 [ 13 ]=>-4472

f(13) = -4472

Find MAX

Optimization terminated: average change in the fitness value less than options.FunctionTolerance.

Start population:

[ 19 ]=>-2330 [ 4 ]=>-1070 [ 2 ]=>-358 [ 44 ]=>137570 [ 37 ]=>68032

?Genertion 1:

[ 44 ]=>137570 [ 43 ]=>125758 [ 37 ]=>68032 [ 32 ]=>35162 [ 3 ]=>-682

Generation 2:

[ 44 ]=>137570 [ 44 ]=>137570 [ 43 ]=>125758 [ 42 ]=>114602 [ 44 ]=>137570

Result:

[ 53 ]=>276368 [ 53 ]=>276368 [ 53 ]=>276368 [ 53 ]=>276368 [ 53 ]=>276368

f(53) = 276368

**Висновки.**

У ході виконання даної лабораторної роботи я ознайомилась з основними теоретичними відомостями, вивчила еволюційні оператори схрещування та мутації. Як результат, розробила програму для визначення значень цільової функції.