|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **НУЛП, ІКНІ, САПР** | | Тема | оцінка | підпис |
| СПКс-11 | 3 | КОМБІНАТОРНА  ОПТИМІЗАЦІЯ ЗА ДОПОМОГОЮ ЕВОЛЮЦІЙНИХ  МЕТОДІВ |  |  |
| Кикуш О.М. | |
| № залікової: | |
| Методи нечіткої логіки та еволюційні алгоритми при автоматизованому проектуванні | | Викладач: | |
| Кривий Р.З. | |

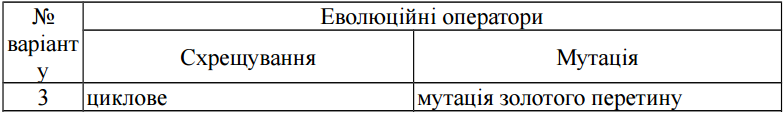
**Мета роботи:**

Ознайомитися з основними теоретичними відомостями, вивчити еволюційні оператори схрещування та мутації, що використовуються при розв’язуванні задач комбінаторної оптимізації.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ**

**Варіант -3**

Розробити за допомогою пакету Matlab програмне забезпечення для вирішення задачі комівояжера. Параметри еволюційного методу обрати з таблиці відповідно до варіанту.



**виконання роботи**

x =

[1x10 double]

fval =

61.6819

reason =

1

output =

problemtype: 'unconstrained'

rngstate: [1x1 struct]

generations: 68

funccount: 6900

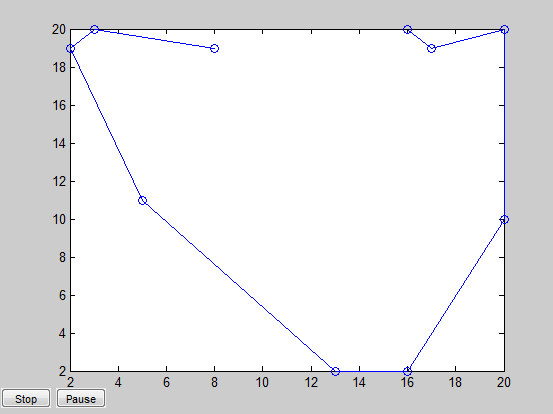


Рис.1. Рішення задачі комівояжера для 10 міст.

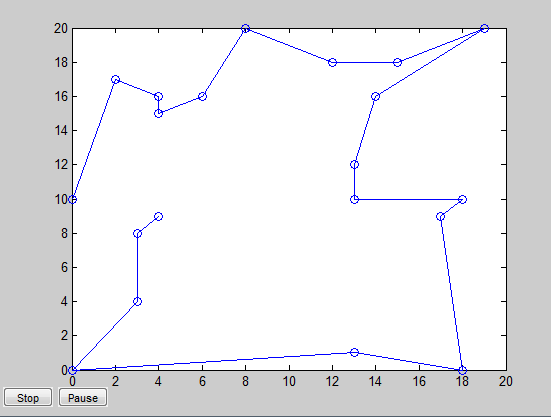


Рис.2. Рішення задачі комівояжера для 20 міст.

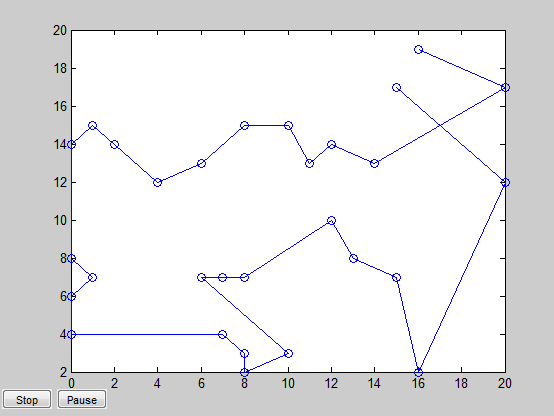


Рис.3. Рішення задачі комівояжера для 30 міст.

Таблиця порівняння залежності кількості міст і популяції

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Кількість міст | 10 | | | 20 | | | 50 | | |
| Покоління | 100 | 200 | 300 | 100 | 200 | 300 | 100 | 200 | 300 |
| Популяція | 20 | 30 | 50 | 20 | 30 | 50 | 20 | 30 | 50 |
| Довжина шляху | 3.04 | 2.14 | 2.64 | 7.46 | 8.03 | 5.86 | 18.6 | 18.68 | 17.21 |
| Час виконання,c | 0.44 | 1.23 | 3.24 | 0.5 | 1.51 | 4.16 | 1.23 | 3.46 | 7.39 |

**Код програми**

cities = 10;%кількість міст

locations = zeros(cities,2);

mas = zeros(cities,2);

x=cell(1,1);

for n = 1: cities

locations(n,1) = randi([0 20],1,1);

mas(1,n) = locations(n,1);

locations(n,2) = randi([0 20],1,1);

end

locations()

x{1}=mas(1,:);

plot(locations(:,1),locations(:,2),'bo')

distances = zeros(cities);

for count1=1:cities,%знаходження відстані між містами

for count2=1:count1,

x1 = locations(count1,1);

y1 = locations(count1,2);

x2 = locations(count2,1);

y2 = locations(count2,2);

distances(count1,count2)=sqrt((x1-x2)^2+(y1-y2)^2);

distances(count2,count1)=distances(count1,count2);

end;

end;

FitnessFcn = @(x) traveling\_fitness(x,distances);

my\_plot = @(options,state,flag) traveling\_s\_plot(options, ...

state,flag,locations);

c = clock;

options = gaoptimset('CreationFcn',@create\_permutations, ...

'CrossoverFcn',@CXcrossover, ...

'MutationFcn',@gold\_mut, ...

'PlotFcn', my\_plot, ...

'Generations',500,'PopulationSize',100);

[x,fval,reason,output] = ga(FitnessFcn,cities,options)

lastPopul = population(length(population))

c2 = clock;

time=c-c2%знаходження часу роботи оптимізації

displayEndOfDemoMessage(mfilename)

**Висновки:** виконавши лабораторну роботу я вивчив еволюційні оператори схрещування та мутації, що використовуються при розв’язуванні задач комбінаторної оптимізації. Реалізував за допомогою пакету Matlab програмне забезпечення для вирішення задачі комівояжера з pmx схрещуванням і класичним інвертуванням.