

# Raport tema nr 3

Bucătaru Andreea A2

Bulboacă Maria A2

## 1 Descrierea problemei

Să se rezolve *Problema comisului voiajor (TSP)* cu o metodă euristică și cu un algoritm genetic.

## 2 Algoritmul utilizat

Pentru rezolvarea acestei probleme, vom implementa un algoritm euristic și unul genetic. Acesta din urmă are la bază o strategie evolutivă, plecând de la o populație inițială de cromozomi care, pe măsură ce avansează, evoluează cu ajutorul operatorilor genetici. Doar cei mai puternici indivizi (cromozomi) vor supraviețui timp de mai multe generații.

### 2.1 Detalii de implementare pentru algoritmul genetic

Populația este reprezentată în memorie printr-o matrice care reține mai mulți cromozomi, fiecare cromozom fiind o listă de orașe (gene).

Inițializarea constă în crearea random a unor cromozomi. Generarea aleatoare a populației inițiale se face pe baza unei liste de orașe și distanțe date, interschimbând orașele între ele.

Funcția fitness este utilizată pentru a măsura calitatea cromozomilor. Ea este formulată plecând de la funcția numerică de optimizat. Trebuie să fie pozitivă și este construită pentru maximizare. În cazul problemei noastre, valoarea fitness-ului pentru orașul  $x$  va fi  $1/y$ , unde  $y$  reprezintă costul distanței cromozomului  $x$ .

Scopul selecției este de a alege pentru supraviețuire cei mai adaptați indivizi din populație cu speranța că descendenții acestora vor avea un fitness mai ridicat. O presiune de selecție ridicată duce la crearea unei diversități scăzute în populația creată. Pe de altă parte, o presiune de selecție scăzută încetinește evoluția. În algoritmul implementat este folosită schema de selecție "Roata norocului".

Operatorii genetici folosiți sunt mutația (modificarea unei gene alese aleatoriu) și încrucișarea (schimbul de informație genetică între doi cromozomi).

### 2.2 Pseudocodul algoritmului genetic

1. Se generează random o populație inițială.
2. Cât timp populația evoluează:
  - a) Se evaluează populația curentă.
  - b) Se selectează cromozomii pentru următoarea generație.
  - c) Cromozomii se supun încrucișării și mutației.
3. Se returnează cea mai bună soluție găsită.

**2.3 Algoritmul euristic** folosește o abordare Greedy pentru a obține drumul de cost minim. Plecăm din fiecare oraș, alegând de fiecare dată cel mai apropiat oraș, prin care nu am trecut, față de cel în care ne aflăm la un moment dat, iar când terminăm drumul curent comparăm cu cel mai bun găsit până acum.

## 2.4 Pseudocodul algoritmului euristic

1. Se pornește din fiecare oraș.
2. Cât timp mai sunt orașe nevizitate:
  - a) Se alege cel mai apropiat oraș, care nu a fost vizitat.
  - b) Se compară cu cel mai bun drum găsit până atunci.
3. Se returnează cea mai bună soluție.

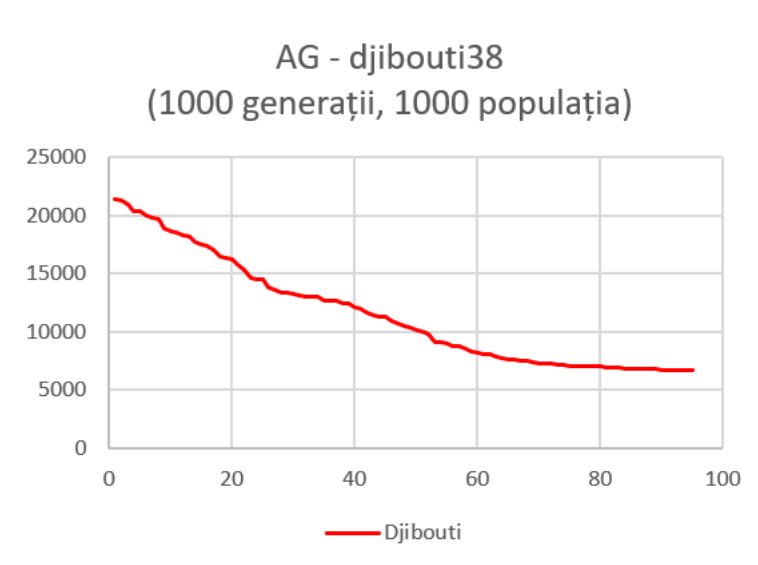
## 3 Rezultate experimentale

Algoritm	Date de test	Nr orașe	Nr rulări	Minim	Medie	Maxim
Genetic	Djibouti	38	30	6693	8235	11510
	Berlin	52		11919	13492	15637
	EIL	101		2075	2289	2408

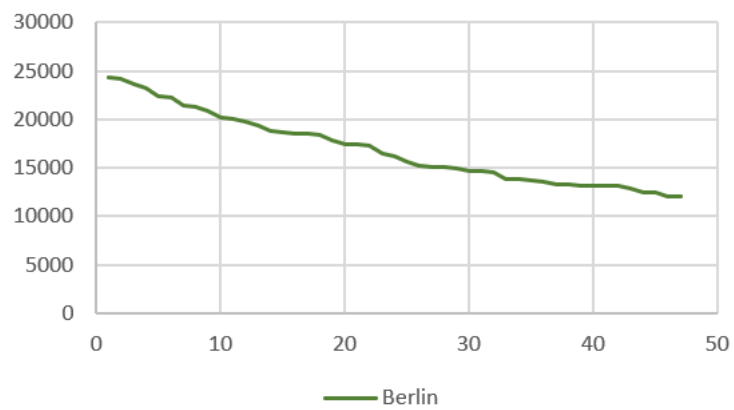
Algoritm	Date de test	Nr orașe	Nr rulări	Rezultat
Euristic	Djibouti	38	1	6770.08
	Berlin	52		8182.19
	EIL	101		736.36

Rezultatele algoritmului genetic au fost obținute cu o populație inițială de 1000 de cromozomi, 1000 generații,  $P\_CROSS = 0.05$  și  $P\_MUTATION = 0.25$ .

Influența valorilor parametrilor: Dacă mutația sau încrucișarea cromozomilor se realizează prea des, un individ nu poate evolua la fel de bine, pentru că ar ajunge să fie complet diferit. De asemenea, cu cât lăsăm populația să evolueze mai mult, cu atât o să obținem rezultate mai bune.



AG - berlin52  
(1000 generații, 1000 populația)



AG - eil101  
(1000 generații, 1000 populația)

