Compania minieră spațială MinAstero operează 5 exploatări situate în locuri diferite ale centurii de asteroizi dintre Marte și Jupiter, la mari distanțe între ele. Compania dispune de n nave de transport de tipuri diferite, care trebuie să aducă resursele obținute pe Terra. Pentru fiecare navă se cunoaște capacitatea de transport (masă maximă). Datorită distanțelor mari, fiecare navă trebuie alocată unei singure exploatări. Utilizați un algoritm genetic pentru a stabili cărei exploatări îi este alocată fiecare navă, astfel încît capacitățile totale de transport alocate fiecărei exploatări să fie cît mai apropiate. Capacitățile de transport ale navelor se pot prelua dintr-un fișier text.

1.

Reprezentare:

-matrice de dim(6) dimensiune cu n linii – 5 gene + ultima pozitie pentru calitate

-fiecare linie=cromozon=vector de numere intregi sub forma unei permutari

-Restrictie: fiecare nava—o singura exploatare

Codificare:

-vector de numere intregi intre 1-5 reprezentand exploatarii la care v-a fi alocata nava

Spatiu:

- numere naturale apartinand multimii {1-5}

2.

-maximizarea capacitatii fiecarei nave

-capacitatiile totale de transport cat mai apropiate

(diferenta intre cel mai mare si cel mai mic sa fie minima)

3.

Variatie:

Recombinare: Pentru recombinare se va utiliza ocx

Mutatie: interschimbare

4.

Selectia parintilor: FPS cu Sigma Scalare pentru a imbunatatii performanta alogirtmului genetic prin diminuarea fluctatiilor fitness ului

Selecti generatiei viitoare: Elitism- selecteaza indivizii cu calitataea cea mai buna

5.

Algoritmul se incheie cand numarul de iteratii se termina si se gaseste individul cu calitatea cea mai buna din cei generati in numarul de iteratii primit ca si parametru

Problema 2: . Un produs P este obţinut în două fabrici situate în două orașe (București și Craiova) și este stocat pentru desfacere în trei depozite, unul situat în Ploiești, unul în Pitești și unul la Cluj. Fabrica din București poate produce săptămînal 120 de tone din produsul P, iar fabrica din Craiova poate produce P în cantitate de 140 tone pe săptămînă. Pentru desfacerea produsului, necesarul săptămînal este: pentru depozitul din Ploiești 100 de tone, pentru depozitul din Pitești, 60 de tone, respectiv pentru depozitul din Cluj 80 de tone. În tabelul de mai jos sunt prezentate costurile de transport pe tona de produs. Ploiești Pitești Cluj București 50 70 90 Craiova 60 70 100 Problema de rezolvat: calculul numărului de tone din produsul P care trebuie furnizate de cele două fabrici fiecărui depozit astfel încît costul de transport să fie minim și astfel încît să fie respectate condițiile enunțate mai sus.

Fabricat:

Bucuresti- 120 tone/saptamana

Craiova-140 tone/saptamana

Stocat:

Ploiesti- 100 tone

Pitesti- 60 tone

Cluj- 80 tone

Reprezentare:

Matrice cu dim linii si 6 coloane reprezentand gene si a 7a coloanal reprezentand calitatea fiecarui individ

Fiecare linie reprezentand un cromozon cu 6 gene- fiecare gena reprezinta cantitatea transportata in fiecare din cele 3 fabrici de catre cele 2 fabrici producatoare: enuntare

Fiecare gena va avea valori intre 0 si cantitatea maxima transportata.

Restrictie: Totalul cantitatii care poate fi furnizata sa fie mai mare sau egal cu cererea:

100+60+80=240<=260 -🡪restrictia poate fi respectata

Calitatea fiecarui individ reprezinta costul de transport.

Codificare:

Fiecare individ este format din 6 gene .Cele 6 gene reprezinta cantitatea transportata. Vectorul este format din numere intregi.

Spatiul solutilor: Spatiul solutilor este definit de cantitatea maxima de produs saptamanal ce poate fi produs de fiecare fabrica.

Obiectiv:

Functia obiectiv va calcula cotul total de transport pentru furnizarea cantitatilor de produs necesar fiecarui deposit, astfel incat acesta sa fie cat mai mic.

Costul de transport va fi calculate astfel: Suma de cantitate transportata\*costul transportului

Recombinare:

Unipunct- se va allege un punct aleator. Din doi candidadati parinti vor interschimba sub-diviziunile astfel incat se vor forma 2 noi copii

Mutatie:

Se va utiliza restetarea aleatoare, se va allege un aleatory un punct unde gena v-a fi schimbata cu una noua formata dintr-un numar intre 0 si cantitatea maxima transportata.

Selectia parintilor:

Pentru selectia parintilor se va utiliza algoritmul SUS pentru ca spatial solutilor sa fie cat mai diversificat si cat mai explorat: In metoda SUS se calculeaza suma totala a valorilor fitness ului ale indivizilor din populatie si se stabileste un punct de start intre 0 si suma totala. S e genereaza n puncta de selectie, unde n reprezinta numarul de parinti. Aceste n puncta sunt distribuite uniform axei fitness ului. Se aleg indivizii ce corespund axei de selectie. (indivizii cu valori ale fitness ului mai mari vor fi selectati de mai multe ori iar cei cu valori mai mici de mai putine ori)

Selectia generatiei urmatoare:

Vom utiliza algoritmul Elitist pentru a fi selectati indivizii cu cea mai buna calitate—aceasta metoda asigura conservarea celor mai bune soltii si ajuta la mentinerea procesului realizat pana in present.(prin pastrarea celor mai buni indivizi se Evita regresia populatiei)

Conditii de terminare:

Algoritmul se va incheia la finalul numarului de iteratii transmis ca si parametru.

Problema 3:

Pentru aprovizionarea coloniilor umane din sistemul solar, pe orbita Terrei se formează un tren spațial format din n vagoane. Trenul va vizita succesiv fiecare colonie unde va descărca materialele prevăzute pentru acea colonie. Materialele sînt ambalate în m containere speciale. Pentru fiecare container se cunoaște masa totală. Utilizați un algoritm genetic pentru a aloca fiecare container într-un vagon, astfel încît vagoanele să fie încărcate cu mase totale cît mai apropiate. Masele containerelor pot fi preluate dintr-un fișier text. Nu se ia în considerare o limită superioară de masă pentru vagoane.

Reprezentare:

Fenotip:Un individ reprezinta o solutie, un container atribuit unui vagon

Genotip: Lista de permutari unde fiecare gena reprezinta indexul containerului si ordinea( I ul vectorului) reprezinta alocarea in vagoane

Matrice formata din dim linii si n+1 coloane

Fiecare linie reprezinta un cromozom format din n gene si calitate

Fiecare gena reprezinta containerul atribuit fiecarui vagon, iar calitatea este data de masa totala transportata

Restrictie: Calitatea maxima-calitatea minima= sa fie minima

Codificare: Un candidat solutie este reprezentat sub forma unei permutari, unde v[i] este containerul atribuit, iar I este vagonul in care este atribuit

Spatiul solutilor: Spatiul solutilor este reprezentat de numarul de vagoane la puterea numarului de containere: n^m

Functia obiectiv:

Transportarea cator mai multe containere la capacitate maxime astfel incat masa totala maxima-masa totala minima sa fie minima

Variatie:

Pentru recombinarea populatiei se va allege metoda operatorul OCX: Se vor selecta 2 puncte de taiere in cromozomi, se va pastra sectiunea respective pentru copil. Se parcurge sectiunea taiata a celui de-al doilea parinte si se completeaza copilul, astfel incat sa nu existe duplicate. Se completeaza restul copilului cu genele ramase din cel de-al doilea parinte.

Mutatie: Pentru mutatie se va allege interschimbarea, astfel vor fi selectate 2 gene din parinte care se vor interchimba intre ele.

Selectia:

Selectia parintilor : Se va allege metoda Turneu astfel incat fiecare canditat parinte sa aiba sanse egale, spatial solutilor fiind unul mare. Vor intra in tureu candidatii parinti si vor fi selectati cei cu calitatiile cele mai bune – cel mai bun va fi selectat

Selectia generatiei urmatoare: Se va utiliza metoda Elitista, astfel se vor pastra indivizii cu cea mai buna calitate, asigurand o populatie viitoara buna si evitand regresia.

Conditii de terminare:

Algoritmul se incheie la in functie de convergenta populatiei--- cand se gaseste individul cel mai bun.

PROBLEMA 4:

Cele n orașe stat din Grecia antică se luptă între ele pentru dominație, dar în fața unei amenințări externe hotărăsc să se unească. Pentru a stabili planul de apărare, delegații orașelor urmează să se întîlnească, fiecare oraș desemnîndu-și un singur reprezentant. Cunoscînd rivalitățile istorice dintre orașe, folosiți un algoritm genetic pentru a găsi o modalitate de așezare a delegaților la masa (rotundă) tratativelor astfel încît delegații din orașe rivale să nu fie vecini (se presupune că acest lucru este posibil). Harta orașelor stat între care există animozități este exprimată printr-o matricea pătratică de ordin n, numită CONFLICT: 𝐶𝑂𝑁𝐹𝐿𝐼𝐶𝑇(𝑖,𝑗) = { 0, 𝑑𝑎𝑐ă 𝑖 ș𝑖 𝑗 𝑛𝑢 𝑠𝑒 𝑎𝑓𝑙ă î𝑛 𝑠𝑡𝑎𝑟𝑒 𝑑𝑒 𝑐𝑜𝑛𝑓𝑙𝑖𝑐𝑡 𝑠𝑎𝑢 𝑖 = 𝑗 1, 𝑑𝑎𝑐ă 𝑜𝑟𝑎ș𝑢𝑙 𝑠𝑡𝑎𝑡 𝑖 𝑒𝑠𝑡𝑒 î𝑛 𝑐𝑜𝑛𝑓𝑙𝑖𝑐𝑡 𝑐𝑢 𝑜𝑟𝑎ș𝑢𝑙 𝑠𝑡𝑎𝑡 �

Reprezentare:

Fenotip: un individ reprezinta o solutie, un mod optim de aranjare a oraselor

Genotip: Lista de permutari in care v[i] reprezinta delegatul din orasul I, iar indicele reprezinta pozitia de asezare la masa

Matrice formata din dim linii si n+1 coloane reprezentand asezarea oraselor si calitatea fiecarui cromozon

Codificare: Lista de permutari in care v[i] reprezinta delegatul din orasul I, iar indicele reprezinta pozitia de asezare la masa

Spatiul solutiloR: n factorial, unde n este numarul de orase

Functia obiectiv: Asezarea tuturor delegatilor la masa astfel incat rivalitatiile( calitatea fiecarui individ) sa fie minima

Recombinare: se va allege operatorul OCX- Order Crossover astfel: …Asigura ordinea genelor si Evita dublicarile.

Mutatie:Se va allege metoda de inerschimbare

Selectia parintilor: Turneu- indivizii intra in tuneu si este selectat cel cu calitatea maxima, astfel fiecare candidat solutie are sanse egale , fiind exploatat efficient spatial solutilor

Selectia generatiei urmatoare: se va alaege metoda elitista, fiind selectati indivizii cei mai buni(cu calitatiile maxime—in cazul nostrum, cu numarul rivalitatilor minime)- ofera optimizarea generatiei urmatoare si Evita regresia acesteia

Conditii de terminare: Atunci cand se atinge un numar maxim de generatii si se obtine o solutie satisfacatoare.

PROBLEMA 5: O companie aeriană dorește să achiziționeze 3 tipuri de aeronave, avînd un buget de achiziții de 5.000 de unități. Fiecare tip de avion, notat de la a. la c. are următoarele caracteristici: a. costă 100 de unități, autonomie 6.000 km și raza de detecție TCAS 30 km; b. costă 60 de unități, autonomie 4.200 km și raza de detecție TCAS 48 km; c. costă 50 de unități, autonomie 2.800 km și raza de detecție TCAS 32 km. Calculați cîte aeronave din fiecare tip trebuie cumpărate astfel încît - să nu fie depășită suma disponibilă; - autonomia medie să fie maximă; - valoarea medie razei de detecție TCAS să fie cel puțin 40 km. Observație. Dacă numărul avioanelor din fiecare tip este a, b, respectiv c, atunci autonomia medie este 6000∙𝑎+4200∙𝑏+2800∙𝑐 𝑎+𝑏+𝑐 și valoarea medie a razei de detecție TCAS este 30∙𝑎+48∙𝑏+32∙𝑐 𝑎+𝑏+𝑐 .

Reprezentare:

Matrice cu dim linii si 3 coloane+ calitatea

FENOTIP: Fiecare linie reprezinta un cromozom, un mod de cumparare a avioanelor astfel incat restricitiile impuse sa fie respectate

Fiecare gena reprezinta numarul de avioane cumparate din tipul i

GENOTIP: vector de 3 elemente

Restrictie: valoarea medie a razei de detecție TCAS este 30∙𝑎+48∙𝑏+32∙𝑐/𝑎+𝑏+𝑐>=40

Codificare:

Vector de numere reale, in care indexul reprezinta tipul de avion cumparat, iar gen, v[i] reprezinta numarul de avioane de tipul respective cumparate, calitatea fiecarui individ este data de functia fitness

Spatiul solutilor:

Spatiul solutilor pentru fiecare gena este cuprins intre 0 si buget/cost

X1 apartine 0…5000/100= 0…50

X2 apartine 0.. 5000/60=0…

X3 aparine 0…5000/50

Functia obiectiv:

autonomia medie este 6000∙𝑎+4200∙𝑏+2800∙𝑐/𝑎+𝑏+𝑐 sa fie maxima

Recombinare: Unipunct---fiind indivizi cu nr de gene mic

Mutatie: Resetare aleatoare, se allege o gena careia I se schimba valoarea cu o valoare cuprinsa in sp sol: 0…buget/cost

Selectia parintilor: SUS: se calculeaza suma tuturor fitness-urilor si se allege un punct intre 0 si suma totala. Se genereaza n puncte de selectie, unde n este nr de parinti. Aceste n puncta de selectie sunt distribuite uniform axei fitness-ului. Se aleg indivizi ce corespund axei de selectie.

Selectia generatiei urmatoare: Elitism

Oprire: nr de iteratii

PYTHON:

Def GA\_AVIOANE(n, dim, pr, pm)

#N este nr maxim de iteratii

#dim este dimensiunea populatiei

#pr recombinare

#pm mutatie

Populatie=GenPop(dim)

I=0

While i<nmax

#parinit=selectie\_parinti\_SUS(populatie)

#descendenti=recombinare\_unipunct(parinti)

#desc=mutatie\_aleatoare(descendenti)

#popViitoare=selectie\_elitism(desc)

I=i+1

X=np.argsort(populatie[:,-1])

PROBLEMA 6: Pentru realizarea unui studiu extensiv asupra florei și faunei dintr-un număr de n insule virgine ale arhipelagului X, sînt înființate stații științifice de observare. Distanțele între fiecare două insule sînt date sub forma unei matrice de ponderi (D(i,j) = distanța între insula i și insula j). O companie care asigură distribuția proviziilor către stațiile de observare dorește să creeze 3 depozite de distribuție, în 3 insule diferite dintre cele n, care să deservească toate stațiile apropiate. Folosiți un algoritm genetic pentru a alege cele trei insule unde să fie amplasate depozitele, astfel încît să fie minimizată distanța totală de la fiecare insulă la depozitul cel mai apropiat.

Reprezentare:

Genotip: un vector de 3 elemente reprezentand insulele selectate pentru amplasarea depozitelor

Fenotip: Vector de insule, unde fiecare insula va aparea de un nr de ori dat de genotipul sau

Codificare: Genotipul constă dintr-un vector de trei gene, fiecare gen reprezentând un index de insulă din arhipelag.

Valorile posibile ale genelor variază între 1 și n, unde n reprezintă numărul total de insule din arhipelag.

Fiecare gen indică insula selectată pentru amplasamentul depozitului.

De exemplu, genotipul [2, 1, 3] ar însemna că depozitele vor fi amplasate pe insula a doua, a întâia și a treia în ordine.

Sp sol:

Fiecare insula trebuie sa fie selectata doar o data si sa apartina multimii de nr intregi 1-n

Unde v[i] este insula pe care sunt depozitate depozitele si I este depozitul

Obiectiv:

Functia obiectiv va fi calculul disnatei totale de la fiecare insula la cel mai apropiat deposit 🡪 minimizarea functiei obiectv

Recombinare: unipunct

Mutatie: inerschimbare- ofera o modalitate de exploatare si diversificare a solutiilor in cadrul ga.

Selectia parintilor: Vom utiliza turneu, se vor allege n indivizi ce vor intra in tuneu. Se va allege cel cu calitatea cea mai buna(functia obiectiv maxima)

Selectia generatiei urmatoare: Metoda elitista: se vor pastra indivizii cu calitatiile maxime

Terminare: nr de iteratii

PROBLEMA 7: . Fie un set de n puncte în planul 𝑆, fiecare punct 𝐴 ∈ 𝑆 fiind reprezentat prin coordonatele carteziene (xA, yA), Proiectați și implementați un algoritm genetic pentru a determina 3 puncte coliniare din setul dat, dacă astfel de puncte există. Punctele 𝐴, 𝐵, 𝐶 ∈ 𝑆 sînt coliniare dacă una din următoarele proprietăți este adevărată: - |𝐴𝐵| = |𝐴𝐶| + |𝐶𝐵| - |𝐴𝐶| = |𝐴𝐵| + |𝐵𝐶| - |𝐶𝐵| = |𝐶𝐴| + |𝐴𝐵| unde |𝐴𝐵| = √(xA − xB) 2 + (yA − yB) 2

Reprezentare:

Matrice de dim linii si 7 coloane

Fiecare linie= cromozom, un set de 3 puncte coliniare

Fiecare gena: 3 seturi a cate 2 gene, prima gena reprezentand coordonata X si ce-a de-a doua reprezentand coordonata Y a punctului ales, ultima coloanal: calitatea

Codificare: vector de numere reale, fiecare gena reprezentand o coordonata in spatial solutilor

Spatiul solutiilor: combinari de n luate cate 3

Recombinare: unipunct

Mutatie: fluaj

Parinti: SUS

Viitoare: Elitism

Finish: nr iteratii