**Schema rezolvare de probleme**

1. Date de intrare
2. Reprezentare

Ce se cere? (vector, matrice, etc)

Spatiul solutiilor, Nr de puncte din acest spatiu, domeniul de definitie

Avem / Nu avem constrangeri(conditie pe care fiec pct din acea conditii trb sa indeplineasca). Dc avem constrangeri le notam

1. Functie de evaluare(Scop: de minimizat, maximizat, valoarea optima)
2. Functia fitness(formula matematica. f\_obiectiv(x) = )
3. Generam populatia initiala(Lucrăm cu populații cu dimensiune fixă, *dim* = dimensiunea populației curente. Aleator uniform, din spațiul soluțiilor, Dc are restrictii: cu evaluarea și verificarea fezabilității pentru fiecare individ. Dc sunt generati indivizi nefezabili sunt ignorati si generati altii)

Conditie de terminare – atingerea rezultatului optim sau un numar limitat de iteratii(de ex 200 pt inceput, rulam de mm ori, dc rezultatele difera mult marim resursele)

Conditie de diversitare genetica - sa nu fie toti indivizii cu aceeasi calitate pt a folosi FPS cu sigma scalare

Dimensiune populatie – initial alegem 100(trb sa fie par si sa fie mai mic de 10% din spat solut), pe parcus analizam si reducem sau marim resursele, pe baza experientei din alta problema.

1. Selectia parintilor

Folosim FPS cu sigma scalare pt ca ne am asigurat ca avem min 2 calitati diferite si diminueaza dizavantajele a lui FPS standart(le folosim doar dc avem valorile calitatilor > 0) va fi folosita impreuna cu ruleta multibrat.

1. Recombinare

Probabilitatea de recombinare 0.6 - 0.8

In aceasta problema nu exista constrangeri, deci voi folosi schema generala de recombinare fara constrangeri \ exista constrangeri folosim deci voi folosi schema generala de recombinare cu constrangeri

(in cazul in care apar constrangeri; 1. Repetam aplicarea 2. Pastram parintii in loc de copii neviabili)

Aleg operatorul de recombinare uniforma. Daca dupa mai multor incercari rezultatele nu sunt satisfacatoare incerc operatorul multipunct.

(Dc avem reprezentare ca sir de nr intregi deci avem de ales dintre - unipunct/multipunct/uniform. Multi - pct de taiere depind de valoare lui n dc n nu e foarte mare folosim 2 sau 3 pct. La reprez prin permutari: PMX, OCX)

1. Mutatie

Problemă fără constrîngeri cu reprezentarea sir de numere(mai avem si permutari aici nu putem face mutatie pe fiec gena facem pe intreg individ) 🡪 schema generală de mutație la nivel de populație: fără constrîngeri. Mutația este la nivel de poziție (genă) .

Operatorul – depinde de reprezentare avem fluaj(pt atribute ordinale) si resetare aleatoare(pt atribute cardinale

Probabilitate: 0.1

1. Selectia generatiei urmatoare - Elitism
2. Alegem cel mai bun individ

**6. 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. Date de intrare

N – nave de transport (de la 1 la n)

Masa maxima - Vector de lungime n, pe pozitia I avem capacitatea de transport pt nava i

1. Reprezentare

Ce se cere?

Facem un vect cu n elem, pt fiec elem a vect domeniul de def e 1- 5

Fenotip – sir de nr intregi, nu avem nevoie de decodificare. Fenotipul = genotip

Spatial solutiilor – multimea 1,2,3,4,5 la puterea n

Nr de puncte din acest spatiu – 5^n

Nu avem constrangeri(conditie pe care fiec pct din acea conditii trb sa indeplineasca)

1. Functie de evaluare

facem vector cu 5 elem, initializat cu 0 – rep capacitatea de transport alocata fiecarei exploatari. Dupa care parcurgem cromozonul. Nava 3 e alocata pt exploatarea 3, adunam capacitatea navei 3 la explatarea 3, etc.

calculam diferentele astfel incat capacitățile totale de transport alocate fiecărei exploatări să fie cît mai apropiate. Dif max si min a capacitatilor alocate

1 / Dif max si min a capacitatilor alocate + 1 (rezultatul optim este 1)

1. Generam populatia initiala

Conditie de terminare – atingerea rezultatului optim sau un numar limitat de iteratii(de ex 200 pt inceput, rulam de mm ori, dc rezultatele difera mult marim resursele)

Dimensiune populatie – 100(trb sa fie par si sa fie mai mic de 10% din spat solut), pe parcus analizam si reducem sau marim resursele, pe baza experientei din alta problema. Conditie: sa avem min 2 calitati diferite. Verificam: dc min si max sunt = atunci toate calit sunt egale

1. Selectia parintilor

Folosim FPS cu sigma scalare pt ca ne am asigurat ca avem min 2 calitati diferite si diminueaza dizavantajele a lui FPS standart(sau FPS standart)(le folosim doar dc avem valorile calitatilor > 0) va fi folosita impreuna cu ruleta multibrat.

1. Recombinare

Probabilitatea de recombinare 0.7; 0.8

In aceasta problema nu exista constrangeri, deci voi folosi schema generala de recombinare fara constrangeri

(dc problema e cu constrangeri trb sa zicem ce facem in cazul in care apar constrangeri; 1. Repetam aplicarea 2. Pastram parintii in loc de copii neviabili)

Aleg operatorul de recombinare uniforma. Daca dupa mai multor incercari rezultatele nu sunt satisfacatoare incerc operatorul multipunct.

(Operatorul de recombinare – depinde de reprezentare, dc avem reprezentare ca sir de nr intregi deci avem de ales dintre - unipunct/multipunct/uniform)

1. Mutatie

Problemă fără constrîngeri cu reprezentarea sir de numere(mai avem si permutari aici nu putem face mutatie pe fiec gena facem pe intreg individ) 🡪 schema generală de mutație la nivel de populație: fără constrîngeri.

Operatorul – depinde de reprezentare avem fluaj(pt atribute ordinale) si resetare aleatoare(pt atribute cardinale

Probabilitate: 0.1 (0 – 0.5)

1. Selectia generatiei urmatoare

Elitism

1. Alegem cel mai bun individ

**8. O universitate organizează o expediție științifică pe mare pentru vizitarea unor insule virgine din arhipelagul X, cu plecare dintr-un port al arhipelagului și revenire în același port. Nava nu poate fi reaprovizionată în timpul expediției. Resursele disponibile pe navă pot asigura vizitarea a n insule din cele 1.000 care formează arhipelagul (n << 1.000) și parcurgerea unei distanțe totale de maxim k kilometri. Folosiți un algoritm genetic pentru a alege insulele vizitate și ordinea de vizitare. Este preferată vizitarea unor insule aflate la distanțe cît mai mari. Distanțele între insule (inclusiv portul de plecare) se preiau sub forma unei matrice de ponderi dintr-un fișier text, unde D(i,j) = distanța între insula i și insula j.**

Date de intrare:

1000 – nr de insule

K - parcurgerea unei distanțe totale de maxim kilometri

1. Reprezentare

Solutia care ni se cere – lista/vector de n elem in care fiec elem e nr unei insule, ordinea acestei liste este oridne de vizitare(prerezentare cu siruri de nr intregi).

Domeniul de definitie : 1 – 1k

Spatiul solutiilor – produsul cartezian din multimea asta cu ea insasi de n ori

Nr de puncte din acest spatiu – 1000^n

Spatiul n dimensional

Restrictii – indivizi mai mari decat k sun nefezabili, ii eliminam

Functia de evaluare:

Scop – functie de minimizat, valoare optima este 0

1 / 1 + lungimea circuitului indicat de cromozon – optim este 1

1. Generare populatie initialia
2. Conditie de terminare – nr de generatii limitat, dc intalnim un individ cu valoarea optima sau lungimea circuitului reprezentat de cel mai bun cromozon sa fie in intervalul k – epsilon(10, 15, 20km. Dc nu obtinem folosim 15 sau 20 de km)
3. Conditie de diversitare genetica sa nu fie toti indivizii cu aceeasi calitate pt a folosi FPS cu sigma scalare
4. Selectia parintilor FPS cu sigma scalare si ruleta multibrat
5. Recombinare - Problemă cu constrîngeri ⇨ schema generală de recombinare cu constrîngeri(orice descendent nefezabil este înlocuit cu părintele corespunzător lui)
6. Operator de recombinare – unipunct sau multipunct(pct de taiere depind de valoare lui n dc n nu e foarte mare folosim 2 sau 3 pct)
7. Probabilitatea de recombinare 0.7; 0.8
8. Mutatie: Problemă cu constrîngeri ⇨ schema generală de mutație cu constrîngeri
9. Sunt attribute cardinale, resetare aleatoare
10. Probabilitate : 0.1
11. Dc e individ neviabil repetam
12. Selectia generatiei urmatoare - Elitism
13. Alegem cel mai bun individ – solutia algoritmului genetic

Problema convoi

Date de intrare: n vagoane, m containere, vector cu masele containerelor

Problema nu are restictii

Ce se cere? (vector, matrice, etc) fiecarui container i se asociaza un vagon in care va fi transportat

Spatiul solutiilor – n^m

Genotip: vector cu m elemente, din multimea n

Fct de evaluare – minimizare diferentelor intre masele totale incarcate in fiecare container. Valoarea optima este 0.

Functia fitness: 1 / ( 1 + max(vagoane) – min(vagoane) ) unde vagoane este suma maselor incarcate in container

Generam populatia initiala aleator uniform din spatiul solutiilor

Conditii de opriere – numarul maxim de iteratii, obtinerea rezultatului optim. Pe parcursul ultimelor k generatii nu a fost imbunatatita valoarea maxima a calitatii.

De exemplu 200 initial, rulam de mai multe ori, daca rezultatele difera mult marim resursele.

Populatia initiala 100. Pe parcurs analizam si marim sau micsoram resursele, pe baza experientei din alte probleme

Conditie de diversitate genetica ca sa nu avem toti indivizi cu aceleasi calitati pt a utiliza FPS cu sigma scalare.

Selectia parintilor – mecanismul sus cu distributia de probabilitate FPS cu sigma scalare

Recombinare – problema este fara constrngeri deci folosim schema generala de recombinare fara constrangeri

Probabilitatea de recombinare: 0.7

Reprezentare: sir de numere intregi - Alegem operatorul de recombinare uniforma. Daca dupa mai multe incercari rezultatele nu sunt satisfacatoare incerc operatorul multipunct.

Mutatie

Problema fara constrangeri deci folosim schema generala de mutatie fara constrangeri. Mutatia este la nivel de pozitie(gena).

Operatorul de mutatie – resetarea aleatoare pt ca avem atribute cardinale

Probabilitate: 0.1

Selectia generatiei urmatoare – elitism

Alegem indivizii cei mai buni din populatia finala

Problema fabrica

Date de intrare: cantitati disponibile, 5 combinatii, profit per combinatie

Se cere: vector de 5 elemente unde pe fiecare pozitie e cantitatea pachetelor pt maximizarea profitului

Spatiul solutiilor

Nr de puncte din acest spatiu

Domeniul de definitie: 1...5

Avem constrangeri - Cantitățile disponibile din fiecare materie

Scop: maximizarea profitului

Functie obiectiv – profit.

Genotip fezabil – nu depaseste cantitatile maxime disponibile de materii prime

Generam populatia initiala aleator uniform din spatiul solutiilor cu evaluarea si verificarea fezabilitatilor pt fiecare individ. Dc sunt generati indivizi nefezabili sunt ignorati si generati altii.

Selectia parintilor – din dim indivizi in populatia curenta alegem dim parinti. Folosim mecanismul SUS cu fps sigma scalare

Conditii de oprire: nr maxim de iteratii, obtinerea individului optim, dupa pe parcursul ultimelor k generatii calitatile nu se modifica semnificativ. In populatia curenta toti indivizii au aceeasi calitate

Recombinare: probabilitatea de recombinare 0.8 – 0.9

In aceasta probleme exista constrangeri deci folosim schema de recombinare generala pt probleme cu constrangeri. Unde pastram parintii in loc de copiii nefezabili

Reprezentare – sir de numere intregi, cu operatorul de recombinare – recombinare uniforma. Daca acesta nu satisface cerinte noastre il schimbam pe recombinare multipunct

Mutatie – in aceasta problema exista constrangeri deci folosim schema de mutatie generala pt problemele cu constrangeri, daca este obtinut individ nefezabil, renunta la mutatie pt acel individ.

Operator de mutatie – fluaj pt ca avem atribute ordinale

Probabilitate – 0.1

Selectia generatiei urm – elitism

Alegem cei mai buni indivizi