



A XI-a Conferință Națională multidisciplinară – cu participare internațională,
"Profesorul Dorin PAVEL – fondatorul hidroenergeticii românești",
SEBEȘ, 2011

DETERMINAREA DISTANȚEI MINIME ÎNTRE DOUĂ PUNCTE FOLOSIND ALGORITMI GENETICI

Ancuța DOBÎRCĂU, Daniela BORDENŢEA, Honoriu VĂLEAN

FINDING THE SHORTEST PATH BETWEEN TWO POINTS USING GENETHIC ALGORITHMMS

This paper describes some fundamentals of genetic algorithms. A genetic algorithm (or GA for short) is a heuristic search technique used in computing and Artificial Intelligence to find optimized solutions to search problems using techniques inspired by evolutionary biology: mutation, selection, reproduction [inheritance] and recombination.

Keywords: genetic algorithm
Cuvinte cheie: algoritm genetic

1. Introducere

Algoritmii genetici fac parte din categoria algoritmilor de calcul evoluționist și sunt inspirați de teoria lui Darwin asupra evoluției. Ideea de calcul evoluționist a fost introdusă în 1960 de I. Rechemberg în lucrarea intitulată „Evolution Strategies”.

Algoritmii genetici au fost inventați și dezvoltați de către John Holland, relevantă fiind cartea sa „Adaptation in Natural and Artificial Systems” publicată în 1975. Obiectivele urmărite de Holland sunt îmbunătățirea înțelegerii procesului de adaptare naturală și proiectarea sistemelor artificiale care au proprietăți similare cu sistemele naturale. În 1992 John Koza folosește algoritmi genetici pentru a dezvolta programe care rezolvă anumite sarcini. El își numește metoda „programare genetică”.

Algoritmii genetici sunt programe inteligente capabile să soluționeze probleme folosind conceptul evoluției speciilor, fiind aplicați cu succes într-o varietate de aplicații care necesită optimizarea globală a soluției. Algoritmii genetici sunt proceduri adaptive care găsesc soluția problemei pe baza unui mecanism de selecție naturală și evoluție genetică. Algoritmii genetici sunt folosiți în cazul în care găsirea soluției optime nu este ușoară sau cel puțin ineficientă datorită caracteristicilor căutării probabilistice. Algoritmii genetici codifică o soluție posibilă într-o singură structură de date numită cromozom și aplică operatori genetici la aceste structuri astfel încât să mențină informațiile critice.

2. Fundamente biologice

Toate organismele vii sunt formate din celule. În fiecare celulă se află același set de **cromozomi**. Un cromozom este compus din **gene**. Fiecare genă codifică o trăsătură, de exemplu culoarea ochilor, iar valorile pe care le poate lua o genă se numesc allele. Fiecare genă se găsește în cromozom la o poziție numită locus. Setul complet al materialului genetic (toți cromozomii) poartă numele de genom. Un set particular de gene ale genomului se numește genotip.

În timpul procesului de reproducere, în prima fază survine recombinarea (sau crossover). Genele de la părinți formează într-un fel întreg cromozomul nou. Urmașii nou creați pot suferi mutații. Mutația înseamnă că elementele ADN-ului sunt puțin schimbate. Aceste schimbări sunt cauzate în principal de copierea eronată a genelor de la părinți.

3. Spațiul de căutare a soluțiilor

Rezolvarea unei probleme presupune găsirea celei mai bune soluții din mulțimea soluțiilor posibile. Spațiul tuturor soluțiilor posibile se numește spațiul de căutare sau spațiul stărilor. Fiecare punct din acest spațiu reprezintă o soluție posibilă. Căutarea unei soluții este o problemă foarte complicată dar există multe metode pentru găsirea soluției adecvate, printre care se numără și algoritmii genetici.

4. Algoritmii genetici

Algoritmii genetici au ca punct de plecare o mulțime inițială de soluții aleasă aleator și care se numește populație. În această populație fiecare individ, numit cromozom reprezintă o soluție posibilă a

problemei. Cromozomii evoluează de la o generație la alta, în fiecare generație fiind evaluați prin utilizarea unor funcții de fitness (măsurile de potrivire). Pentru crearea populației următoare, se selectează cei mai buni cromozomi din generația curentă, iar noii cromozomi sunt formați folosind unul din cei trei operatori genetici esențiali: selecția, încrucișarea și mutația.

Selecția asigură că anumiți cromozomi din generația curentă sunt copiați, în acord cu valoarea funcției de fitness, în noua generație, ceea ce înseamnă că acei cromozomi cu o importanță mare au o probabilitate mare să contribuie la formarea noii generații.

Crossover este alt operator genetic care reprezintă procesul prin care se formează doi cromozomi pentru populația următoare, pe baza a doi cromozomi din populația curentă.

Mutația este procesul prin care un cromozom din populația curentă este modificat și salvat în noua populație.

Noile generații sunt formate până la îndeplinirea unor condiții, de exemplu până se ajunge la un anumit număr de populații.

Pașii de bază a unui algoritm genetic sunt prezentați în cele ce urmează:

[Start] Se generează aleator populația cu n cromozomi

[Fitness] Se evaluează fitness-ul $f(x)$ pentru fiecare cromozom x al populației

[Populație Nouă] Se creează o nouă populație repetând pașii următori până când noua populație e completă

[Selecție] Se selectează doi cromozomi părinți din populație, conform cu fitness-ul (cu cât e mai bun fitness-ul, cu atât sunt mai mari șansele ca ei să fie selectați)

[Crossover] Părinții se încrucișează cu o anumită probabilitate de încrucișare pentru a rezulta urmași

[Mutație] Se aplică mutații urmașilor

[Acceptare] Se plasează tânăra generație într-o nouă populație

[Înlocuire] Utilizarea noii generații în rularea algoritmului

[Test] Dacă este satisfăcută condiția de terminare, stop algoritm și se returnează cea mai bună soluție din populația curentă.

[Loop] Revenire la pasul 2.

5. Operatorii Algoritmilor Genetici

5.1. Codificarea cromozomilor

Cromozomii, care reprezintă codificarea problemei, trebuie să conțină informațiile despre soluția problemei și depind foarte mult de problemă. Există mai multe moduri de codificare utilizate, cum ar fi codificarea binară, codificarea prin valoare sau sub formă de permutare. De exemplu, un cromozom poate avea următoarele forme:

Cromozomul A 1001100101011101011010011
(codificare binară)

Cromozomul A 1.234 5.3243 0.4456 2.3293 2.4545
(codificare sub formă de valori)

Cromozomul A 1 5 3 2 6 4 7 9 8 (codificare sub formă de permutare)

Funcția de evaluare numită și funcția de "fitness" este funcția care ne permite să dăm o încredere la fiecare cromozom din populație. Această funcție este de obicei funcția care reprezintă descrierea problemei.

5.2. Încrușișarea (crossover)

Încrușișarea este o parte importantă a algoritmilor genetici. În momentul în care s-a luat o hotărâre asupra modului în care se realizează codificarea, se poate trece la partea de încrușișare. În procesul de încrușișare se selectează gene ale cromozomilor părinte și se creează urmași. Cel mai simplu mod de a realiza asta este să se aleagă aleator un punct de încrușișare și tot ceea ce este înaintea acestui punct să se copieze de la primul părinte iar ce este după, să se copieze de la al doilea părinte. Încrușișarea poate să se prezinte astfel:

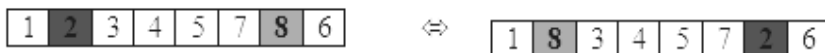


Există și alte modalități de a realiza încrușișarea, una dintre ele fiind prin alegerea mai multor puncte de încrușișare.

5.3. Mutația

Alături de încrușișare, mutația este o altă parte importantă a algoritmilor genetici. Ea se aplică după realizarea încrușișării și are ca

scop evitarea ajungerii soluțiilor într-un optim local. În cazul unei codificări sub formă de permutare, mutația se prezintă astfel:



6. Selecția

Cromozomii sunt selectați din populație pentru a fi părinți într-o încrucișare. Conform teoriei lui Darwin, cei mai buni părinți supraviețuiesc și se înmulțesc creând noi urmași. Dintre metodele cel mai des întâlnite de selecție amintim:

- Selecția aleatoare în care doi părinți sunt aleși aleator din populație.

- Selecția cu ajutorul ruletei ponderate în care părinții sunt selectați în conformitate cu fitness-ul lor. Cu cât sunt mai buni, cu atât șansa lor de a fi aleși este mai mare. Imaginați-vă o ruletă în care sunt dispuși toți cromozomii din populație. Fiecărui cromozom îi corespunde un sector al ruletei, direct proporțional, ca mărime, cu fitness-ul cromozomului respectiv. În acest fel cromozomii cu fitness mai mare au atașate sectoare mai mari iar cei cu fitness mic au atașate sectoare mai mici. La aruncarea bilei pe ruletă există mai multe șanse de alegere pentru cromozomii cu fitness mare.

- Selecția aranjată este folosită în cazul în care fitness-ul cromozomilor diferă foarte mult. Astfel se ordonează cromozomii crescător după fitness, apoi se renumerează cu numere întregi din intervalul $[1, \dots, \text{dimensiunea populației}]$. Cromozomul cu fitness-ul cel mai mic are numărul 1 etc., iar cromozomul cu fitness-ul cel mai mare are numărul egal cu dimensiunea populației. Aceste numere se consideră fitness-uri și pe ele se aplică selecția roții de ruletă.

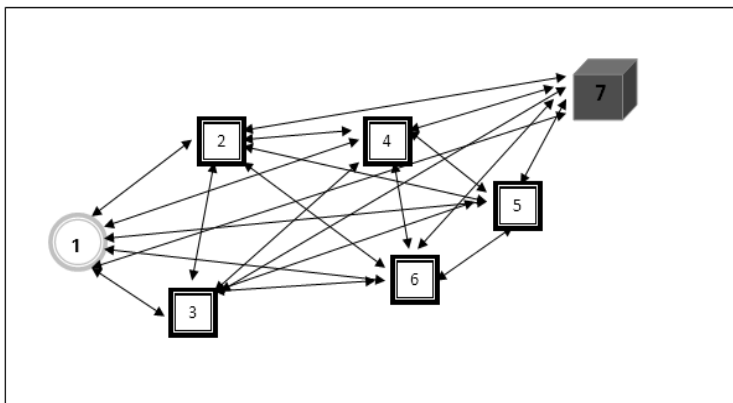
- Elitism. Când se creează o nouă populație prin încrucișare și mutații, există o șansă mare de a pierde cei mai buni cromozomi. De aceea se recomandă copierea celor mai buni cromozomi în noua populație, fără a-i schimba.

7. Aplicație

Pentru a înțelege mai bine algoritmi genetici și modul de utilizare a lor, s-a formulat următoarea problemă: Există un depozit în care se află poziționate n cutii, fiecare cutie conținând același produs. Există un manipulator care trebuie să culegă toate cele n cutii pentru a

le pune pe un palet, paletul urmând a fi pus într-un loc special destinat lui și aflat în punctul Y, prin urmare, manipulatorul pornește din punctul X și trebuie să ajungă în punctul Y, trecând prin toate punctele în care se află cutiile. Poziția cutiilor este cunoscută, prin urmare se poate determina distanța dintre două cutii. Se dorește determinarea traseului de lungime minimă pe care îl parcurge manipulatorul.

O prezentare grafică a problemei se poate observa în figura de mai jos.



Pentru simplificarea problemei, s-a luat în considerare un număr de 5 cutii.

1 – reprezintă poziția manipulatorului

7 – reprezintă poziția unde trebuie poziționat paletul

2, 3, 4, 5, 6 – poziția celor 5 cutii

Cunoscând aceste poziții se poate realiza o matrice distanță care să conțină distanțele dintre două cutii.

Codificarea cromozomului:

Lungimea cromozomului este egală cu 7.



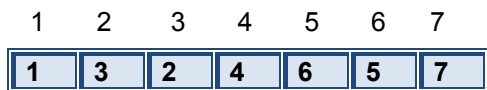
Codificarea genelor:

Un cromozom va fi reprezentat printr-o permutare de tipul 1 2 3 4 5. O permutare reprezintă ordinea în care manipulatorul culege cutiile.

Orice cromozom are genă [1] = 1 și genă [7] = 7. În gene se pun identificatorii cutiilor pe care manipulatorul urmează să le culegă.

Codificarea genelor				
Gena 2	Gena 3	Gena 4	Gena 5	Gena 6
2 – Cutia 2	2 – Cutia 2	2 – Cutia 2	2 – Cutia 2	2 – Cutia 2
3 – Cutia 3	3 – Cutia 3	3 – Cutia 3	3 – Cutia 3	3 – Cutia 3
4 – Cutia 4	4 – Cutia 4	4 – Cutia 4	4 – Cutia 4	4 – Cutia 4
5 – Cutia 5	5 – Cutia 5	5 – Cutia 5	5 – Cutia 5	5 – Cutia 5
6 – Cutia 6	6 – Cutia 6	6 – Cutia 6	6 – Cutia 6	6 – Cutia 6
7 – Depozit 7	7 – Depozit 7	7 – Depozit 7	7 – Depozit 7	7 – Depozit 7

Exemplu:



Prin decodificarea genelor cromozomului de mai sus se obține ruta:

1 -> 3 -> 2-> 4-> 6-> 5-> 7

Se consideră că populația inițială este formată din 20 de indivizi și se realizează aleator.

Funcția de fitness care se consideră în cazul acestei selecții este:

$$F = \begin{cases} \frac{1}{\sum_{i=1}^n d(P_i, P_j)}, & \text{dacă ruta e convenabilă} \\ 0, & \text{dacă ruta e irealizabilă} \end{cases}$$

unde: P_i - poziția cutiei i , iar P_f – poziția cutiei j

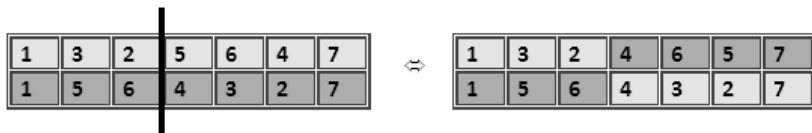
$$d(P_i, P_j) = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2}$$

Vom căuta să obținem cromozomi cu un fitness cât mai mic.

Grupul de elită conține primii 3 indivizi din populația curentă cu cele mai bune valori ale indicelui de fitness.

Modul de încrucișare în cazul problemei de față: se alege aleator un punct de încrucișare (poziția de tăiere – linia verticală); se copiază primele două părți în cei doi descendenți; a doua parte se completează prin inserarea de valori de la celălalt părinte în ordinea în

care apar acolo, începând cu punctul de după tăietură, sărind valorile care se găsesc deja în descendent



Mutația



În urma aplicării operatorilor genetici definiți anterior se obține noua generație de 20 de indivizi.

Acest algoritm se va repeta pentru un număr de 10 generații, număr după care se consideră că s-a găsit un număr de soluții care să satisfacă cerința problemei.

După 10 generații se vor selecta numărul de indivizi care satisfac cerința funcției de fitness.

BIBLIOGRAFIE

- [1] * * * <http://obitko.com/tutorials/genetic-algorithms/> Introduction to Genetic Algorithms
- [2] Popescu Rozica Maria, *Algoritmi genetici*, ISBN 978-606-577-027-0.
- [3] Nagib, Gi., Gharieb, W., *Path Planning for a mobile robot using genetic algorithms*.
- [4] Johannes, F., Knabe, Wegner, K., Maria, J., *Schilstra Genetic Algorithms and their Application to the Artificial Evolution of Genetic Regulatory Networks*, Tutorial ICSB 2007, University of Hertfordshire, UK.

Drd.Ing. Ancuta DOBÎRCĂU
 Drd.Ing. Daniela BORDENŢEA
 membru AGIR
 Prof.Dr.Ing. Honoriu VĂLEAN
 Universitatea Tehnică din Cluj-Napoca
 Catedra de Automatică
 Email: ancuta.dobircau@aut.utcluj.com