

# Tehnici de optimizare in programare

## Raport tema de curs

Chisalescu Bogdan si Chivu Gheorghe-Iulian  
Grupa 432A

### 1 Introducere

Algoritmii genetici sunt o subclasa a algoritmilor evolutivi ce se pot aplica in orice problema de optimizare. Acestia sunt algoritmi matematici inspirati din teoria evolutiei speciilor ce se bazeaza pe un set de caracteristici. Acest set include existenta unei populatii (indivizii reprezentand posibile solutii), o modalitate de a genera noi indivizi in populatie (prin recombinarea a doi indivizi deja existenti si/sau mutatia unui individ) si o modalitate ce simuleaza selectia naturala (un algoritm prin care se decide ce indivizi raman in populatie si modul in care se modifica dimensiunea populatiei). Prezenta lucrare are ca scop implementarea unui algoritm genetic specific apartinand unei lucrari de studiu de cercetare. Lucrarea de cercetare se intituleaza "An Effective Real-Parameter Genetic Algorithm with Parent Centric Normal Crossover for Multimodal Optimisation" si este realizata de Pedro J. Ballester si Jonathan N. Carter in anul 2004.

Urmatoarele sectiuni ce vor fi descrise sunt o imagine a modului in care am reusit sa intelegem notiunile teoretice din spatele acestui studiu, a modului in care am reusit sa implementam algoritmul si in care am interpretat rezultatele obtinute; si nu o reprezentare exacta a studiului, deoarece am incercat pe cat posibil sa reproducem algoritmul si performantele sale.

## 2 Descrierea Algoritmului

Algoritmul descris de studiul de cercetare se intituleaza "Scaled Probabilistic Crowding Genetic Algorithm with Parent Centric Normal crossover" la care ne vom referi pe tot parcursul lucrarii ca SPC-PNX.

SPC-PNX foloseste un model in care numarul de indivizi din populatie este constant iar genele fiecarui individ sunt numere reale. In fiecare generatie doi parinti sunt selectati din aceasta populatie pentru a crea  $\lambda$  copii prin recombinare. Apoi se desfasoara etapa de selectie (descrisa in sectiunea 2.3) pentru a stabili indivizii ce vor ramane in populatie.

### 2.1 Alegerea parintilor

In general pentru algoritmi genetici, alegerea parintilor depinde de valoarea functiei fiecarui individ. SPC-PNX foloseste o selectie aleatoare a parintilor dupa o distributie uniforma, pentru a selecta doi parinti din populatia curenta.

### 2.2 Recombinarea

Pentru fiecare  $\lambda_i$  copil creat prin aceasta metoda procedam astfel pentru a-i calcula componenta  $j$  ( $y_j$ ) : Prima data se extrage un numar aleator  $\omega$  care apartine intervalului  $[0,1]$ , daca  $\omega < 0.5$  atunci folosim forma  $y_j^{(1)}$ , daca  $\omega \geq 0.5$  atunci folosim forma  $y_j^{(2)}$ . Odata ce una din forme a fost aleasa, aceasta va fi folosita pentru toate cele  $j$  componente ale copilului. Aceste forme se calculeaza astfel:

$$y_j^{(1)} = N(x_j^{(1)}, |x_j^{(2)} - x_j^{(1)}|/\eta) \quad y_j^{(2)} = N(x_j^{(2)}, |x_j^{(2)} - x_j^{(1)}|/\eta) \quad (1)$$

Unde:

- $N(\mu, \sigma)$  reprezinta un numar aleator dintr-o distributie normala cu media  $\mu$  si dispersia  $\sigma$
- $x_j^{(i)}$  reprezinta componenta numarul  $j$  a parintelui  $i$
- $\eta$  este un parametru variabil. Cu cat  $\eta$  este mai mare, copilul se afla mai aproape de arealul parintilor.

### 2.3 Selectia (Scaled Probabilistic Crowding Replacement)

Se selecteaza aleator  $\kappa$  indivizi din populatia curenta, pentru a concura cu cei  $\lambda$  copii pentru un loc in populatie.

Pentru fiecare copil ( $x^{ofp}$ ) se determina care individ din cei  $\kappa$  selectati se afla mai aproape de el ( $x^{cst}$ ). Apoi se calculeaza pentru fiecare probabilitatea de eliminare din populatie astfel:

$$p(x^{ofp}) = \frac{f(x^{ofp}) - f_{best}}{f(x^{ofp}) + f(x^{cst}) - 2f_{best}} \quad p(x^{cst}) = \frac{f(x^{cst}) - f_{best}}{f(x^{ofp}) + f(x^{cst}) - 2f_{best}} \quad (2)$$

Unde  $f_{best}$  este valoarea functiei celui mai bun individ din multimea formata din copilul  $\lambda_i$  si cei  $\kappa$  indivizi selectati.

## 3 Pseudocodul algoritmului

Pentru o mai buna intelegere a ideilor expuse am ales sa includem si pseudocodul sub o maniera asemanatoare descrierii metodelor din lucrarile de laborator.

1. Initializarea unei populatii de dimensiune  $N$  cu indivizi  $M$ -dimensionali. Pentru fiecare individ  $x$ , componenta  $x_j$  ia o valoare aleatoare din intervalul  $[-10, -5]$ .
2. Se aleg aleator dupa o distributie uniforma din populatie 2 indivizi ce vor deveni parinti.
3. Pentru fiecare copil se alege aleator un numar real  $\omega \in [0, 1]$ .
4. Daca  $\omega \leq 0.5$  genele copilului vor rezulta din forma 2 din setul de ecuatii (1), altfel genele vor rezulta din forma 1 a aceluiasi set.
5. Pentru fiecare copil se aleg aleator un numar de  $k$  indivizi din populatie, se determina cel mai apropiat individ si pentru acesta si copilul curent se calculeaza probabilitatile descrise de setul (2) de ecuatii, cel cu probabilitatea mai mica va fi pastrat in populatie.
6. Daca optimul este mai mare decat  $10^{-20}$  sau numarul de sondari ale functiei obiectiv este mai mic de  $10^6$  se trece la pasul 2, altfel se returneaza cel mai bun individ ca solutie.

## 4 Rezultatele obtinute si concluzii

SPC-PNX a fost rulat pe o serie de 6 functii obiectiv multimodale: functia Elipsiod (ELP), Schwefel (SCH), Rosenbrock (ROS), Ackley (ACKL), Rastrigin (RTG) si Rotated Rastrigin (RRTG). RRTG reprezinta functia RTG inasa careia i-a fost aplicata o transformare liniara pentru a anula avantajele ce rezulta in urma tendintei algoritmului de a cauta pe axele de coordonate. Rezultatele obtinute sunt prezentate in urmatorul tabel, unde "B.F." (best found) reprezinta cea mai buna solutie gasita de algoritm, "SUCCES" reprezinta de cate ori a fost gasita cea mai buna solutie intr-o serie de rulari succesive ale algoritmului.

M	N	$\lambda$	FUNC	BEST	MED.	WORST	B.F.	SUCCES	RULARI
20	35	1	ELP	$10^6$	-	-	$10^{-15}$	1/10	10
20	35	1	SCH	$10^6$	-	-	$10^{-1}$	1/10	10
20	80	1	ROS	$10^6$	-	-	$10^1$	28/50	50
20	50	1	ACKL	$10^6$	-	-	$10^{-4}$	3/10	10
20	400	4	RTG	$10^6$	-	-	$10^2$	10/10	10
20	400	4	RRTG	$10^6$	-	-	$10^2$	10/10	10

Consideram ca performantele mai reduse comparativ cu cele din studiul de cercetare sunt datorate neimplementarii corecte a algoritmului sau faptului ca algoritmul ajunge sa se blocheze intr-un minim local. Am rulat SPC-PNX si pentru o configuratie cu  $M = 10$ , in acest caz obtinand performante mai bune pentru functiile ELP, SCH si ACKL.