2. Algoritmi genetici şi strategii evolutive

2.1 Algoritmi genetici

Structura unui algoritm genetic standard:

1. Se inițializează aleator populația de cromozomi.
2. Se evaluează fiecare cromozom din populație.
3. Se creează o nouă generație de cromozomi folosind operatori genetici (selecție, încrucişare, mutație).
4. Se şterg o parte din membrii populației actuale, pentru a fi înlocuiți cu cei din noua generație.
5. Se evaluează noii cromozomi şi se inserează în noua populație.
6. Dacă timpul de căutare nu s-a terminat, se merge la pasul 3. În caz contrar, se opreşte execuția algoritmului.

Operatorii genetici:

- **Selecția** – stabileşte cei mai buni cromozomi din populație
  - selecția proporțională sau principiul ruletei

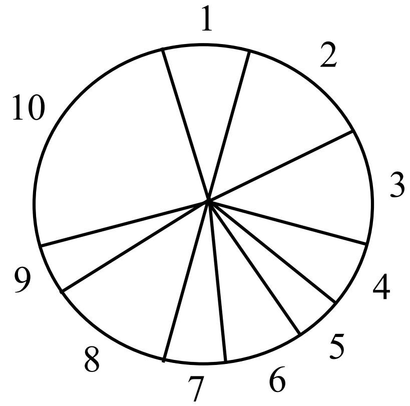

Fig. 2.1 *Principiul ruletei*

1. Se stabileşte funcţia de evaluare $Eval(x_i)$ pentru fiecare cromozom x_i din populaţie
2. Se sumează toate funcţiile de evaluare $Eval = \sum_i Eval(x_i)$
3. Cromozomilor li se atribuie aleator numerele naturale *I*

   Repetă până la crearea numărului suficient de perechi de cromozomi:

4. Se generează aleator *n* şi *m*, astfel ca $1 \le n, m \le Eval$
5. Se alege cromozomul x_i, unde *i* este cel mai mic număr care satisface relaţia: $\sum_{j \le i} Eval(x_j) \ge n$
6. Se alege cromozomul x_j, la fel ca la pasul 5, cu *m* în loc de *n*
7. Se stabileşte perechea de cromozomi x_i şi x_j.

- <u>selecţia prin trunchiere</u>
- <u>selecţia de tip turnir</u>

- **Încrucişarea** – stabileşte modul de schimbare a materialului genetic între cromozomii părinţi
  - <u>încrucişarea într-un singur punct</u>

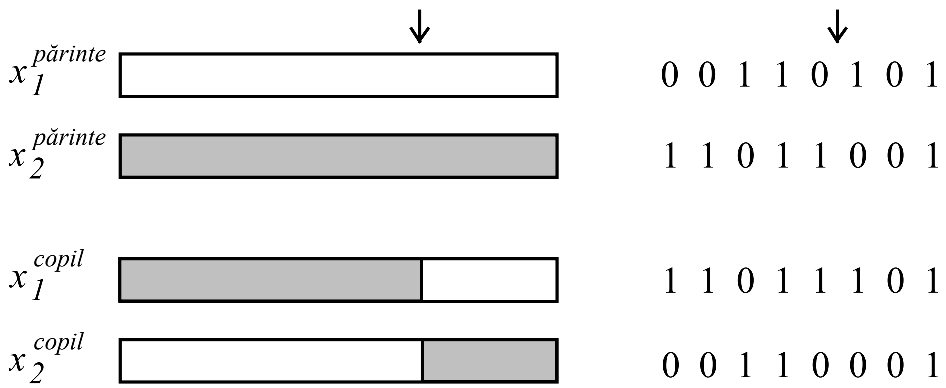

Fig. 2.2 *Încrucişarea într-un singur punct*

- <u>încrucişarea în două puncte</u>
- <u>încrucişarea uniformă</u>

Se face cu o anumită probabilitate χ (între 60% şi 100%)

- **Mutația** – permite găsirea unor soluții noi
  - se alege aleator un bit dintr-un cromozom copil şi se modifică valoarea lui
  Se face cu o anumită probabilitate μ, foarte mică, având valori cuprinse în general între 0,1 şi 1%, pe bit.
  Determină o creştere a diversității populației, având un efect contrar operatorului de selecție, care reduce diversitatea în populația de cromozomi.

Teorema fundamentală a algoritmilor genetici (Teorema Schemei)

Schema - secvență de caractere de lungime l din alfabetul $\{0,1,*\}$. Un exemplu de schemă este $s = 0 * 1\ 1\ 0 *$.

Ordinul schemei s, notat cu $o(s)$, reprezintă numărul de simboluri diferite de "don't care" din schemă. Avem $o(s) = 4$.

Lungimea definitorie a schemei s, notată cu $\delta(s)$, reprezintă distanța dintre primul şi ultimul simbol specific diferit de simbolul "don't care". Pentru cazul nostru $\delta(s) = 5 - 1 = 4$.

Funcția de performanță asociată schemei s, notată cu $f(s)$, reprezintă evaluarea medie a membrilor populației.

Probabilitatea de supraviețuire a schemei în urma încrucişării, notată cu $p(s)$, în cazul unui singur punct de încrucişare, satisface inegalitatea: $p(s) \geq 1 - \chi \cdot \frac{\delta(s)}{l-1}$.

Numărul de apariții ale schemei s în populația de la momentul t, notat cu $m(s, t)$, devine la momentul t+1:

$$m(s,t+1) = m(s,t) \cdot n \cdot \frac{f(s)}{\sum_{i=1}^{n} f(i)} = m(s,t) \cdot \frac{f(s)}{f^{mediu}}$$

Numărul reprezentanților schemei s creşte odată cu trecerea timpului, dacă $f(s) > f^{mediu}$. Dacă ținem seama şi de efectul combinat al operatorilor genetici de încrucişare şi mutație se poate formula <u>Teorema Schemei</u> prin următoarea expresie:

$$m(s,\ t+1) \geq m(s,\ t) \cdot \frac{f(s)}{f^{mediu}} \cdot \left(1 - \chi \cdot \frac{\delta\ (s)}{l-1} - o(s) \cdot \mu\right)$$

Exemplu de căutare a minimului unei funcții multimodale

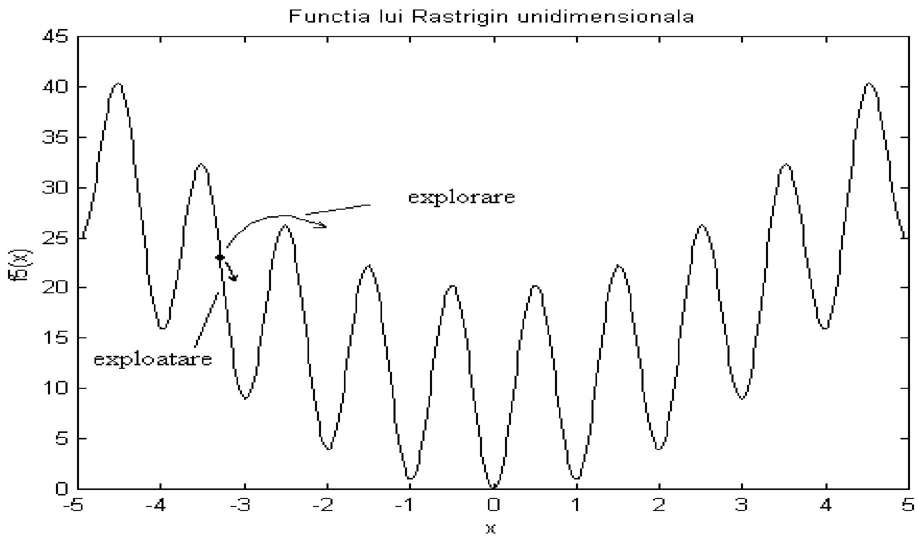

Fig. 2.3 *Funcția lui Rastrigin unidimensională*

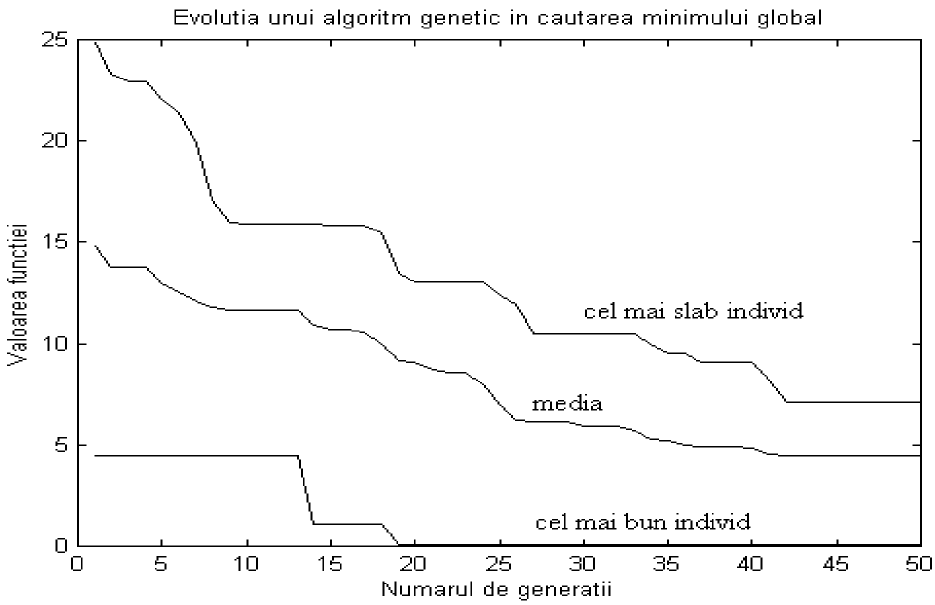

Fig. 2.4 *Evoluția în timp a populației de cromozomi*

2.2 Strategii evolutive

Structura unei strategii evolutive de tip SE(1+1):

1. Se selectează aleator o populație inițială de vectori-părinți $\mathbf{x_i}$, $i = \overline{1,P}$, dintr-un domeniu permis, în raport cu fiecare variabilă. Distribuția inițială este uniformă.
2. Se creează un vector-urmaş, $\mathbf{x_i}'$, prin adăugarea la fiecare componentă a lui $\mathbf{x_i}$ a unei variabile aleatoare cu medie zero şi dispersie predefinită.
3. Se face selecția prin compararea valorilor funcției $F(\mathbf{x_i})$ şi $F(\mathbf{x_i}')$. Cei P vectori care au cele mai mici erori devin părinți pentru noua generație.
4. Procesul de generare şi selecție a generațiilor se opreşte la atingerea unei soluții satisfăcătoare sau la terminarea timpului de calcul disponibil.

- Strategia prezentată este de tip SE(1+1), deoarece un urmaş provine dintr-un singur părinte şi ambii sunt puşi în competiția supraviețuirii.

- Între părinți şi urmaşi există mai degrabă o <u>legătură comportamentală</u> decât una genetică. Mecanismul descris modelează modificarea naturală posibilă a mai multor caracteristici fenotipice, lucru realizat prin variația tuturor componentelor unui părinte pentru a crea un urmaş.

- Dezavantajele metodei constau în rata scăzută de convergență şi o probabilitate destul de mare de a atinge minime locale, comparativ cu algoritmii genetici.

- Rata de convergență estimată este definită de raportul dintre distanța medie parcursă spre soluția optimală şi numărul de încercări necesare atingerii acesteia.

- Pentru o funcție de eroare pătratică $E(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$, s-a demonstrat că rata optimă de convergență este obținută pentru $\sigma \approx 1{,}224 \cdot \frac{r}{n}$, unde σ este dispersia perturbației normale (gaussiene) cu media zero şi *r* este distanța euclidiană obişnuită.

- Modelele fundamentale utilizate în strategiile evolutive *multi-părinți/multi-urmaşi* sunt SE(μ+λ) şi respectiv SE(μ,λ). În primul model, μ părinți creează λ urmaşi şi toate soluțiile concurează la supraviețuire. Cea mai bună soluție devine părinte pentru noua generație. În al doilea model numai λ urmaşi rămân în competiție iar părinții sunt complet înlocuiți în fiecare generație. Durata de viață a unei soluții devine astfel limitată la o singură generație. Creşterea populației duce la mărirea vitezei de optimizare raportată la un număr fix de generații.

- Adaptarea dispersiei în timpul căutării pe baze neeuristice a condus la ideea ca fiecare soluție să conțină şi un *vector de perturbare* σ, care să ofere date asupra mutațiilor lui **x**. De exemplu, dacă σ este vectorul dispersiilor fiecărei dimensiuni a lui **x**, atunci noul vector-soluție (**x** ', σ ') se poate obține din relațiile:

$$\sigma_i' = \sigma_i \cdot e^{\alpha' \cdot N(0,1) + \alpha \cdot N_i(0,1)}$$

$$\mathbf{x_i'} = \mathbf{x_i} + N(0, \sigma_i'), \quad i = \overline{1,n} .$$

$N(0,1)$ şi $N_i(0,1)$ sunt variabile aleatoare cu distribuție normală iar α şi α' sunt mulțimile de parametri care definesc mărimea pasului global, respectiv individual. În acest mod, strategiile evolutive se pot autoadapta la mărimea hipersuprafeței de eroare, rata de convergență crescând în consecință.

- Dacă notăm cu P mulțimea părinților şi cu U mulțimea urmaşilor, atunci putem scrie cele două mulțimi ca fiind formate din elementele **I**:

$$P = \{\mathbf{I}_1,\ \mathbf{I}_2,\ \ldots,\ \mathbf{I}_\mu\} \quad \text{şi} \quad U = \{\mathbf{I}_{\mu+1},\ \mathbf{I}_{\mu+2},\ \ldots,\ \mathbf{I}_{\mu+\lambda},\} .$$

$\mathbf{I} = (\mathbf{x},\ \sigma)$ este un individ, unde **x** este un vector soluție n - dimensional şi σ este vectorul varianței mutațiilor, tot un vector n - dimensional.

- Operatorii evolutivi sunt:

Operatorul de <u>selecție</u> , $S(\Omega) \to P$, care selectează cei mai buni μ indivizi din mulțimea Ω, pentru a produce părinți în noua generație. $\Omega = P \cup U$ pentru strategia SE(μ+λ) şi $\Omega = U$ pentru strategia SE(μ, λ).

Operatorul de <u>recombinare</u>, $R(\mathbf{I}',\ \mathbf{I}'') \to \mathbf{I}$ selectează indivizii **I'** şi **I''** din mulțimea P, cu aceeaşi probabilitate de împerechere. Se poate utiliza fie *recombinarea discretă* în care avem relațiile:

$$\mathbf{x}_\mathrm{i} = \mathbf{x}_\mathrm{i}' \ \text{sau}\ \mathbf{x}_\mathrm{i}'' , \quad \text{şi} \quad \sigma_\mathrm{i} = \sigma_\mathrm{i}' \ \text{sau}\ \sigma_\mathrm{i}'' ,\ \forall \mathbf{i},$$

fie *recombinarea intermediară*, unde:

$$\mathbf{x}_\mathrm{i} = \frac{1}{2} \cdot (\mathbf{x}_\mathrm{i}' + \mathbf{x}_\mathrm{i}'') \quad \text{şi} \quad \sigma_\mathrm{i} = \frac{1}{2} \cdot (\sigma_\mathrm{i}' + \sigma_\mathrm{i}'') ,\ \forall \mathbf{i}$$

Operatorul de *mutație* $M(\mathbf{I}) \to \mathbf{I}$ determină modificări în structura individului care urmează a fi generat. El are acelaşi rol ca şi la algoritmii genetici, iar individul $\mathbf{I}'(\mathbf{x}', \sigma')$ se obține folosind relațiile date pe folia SE 2-6.

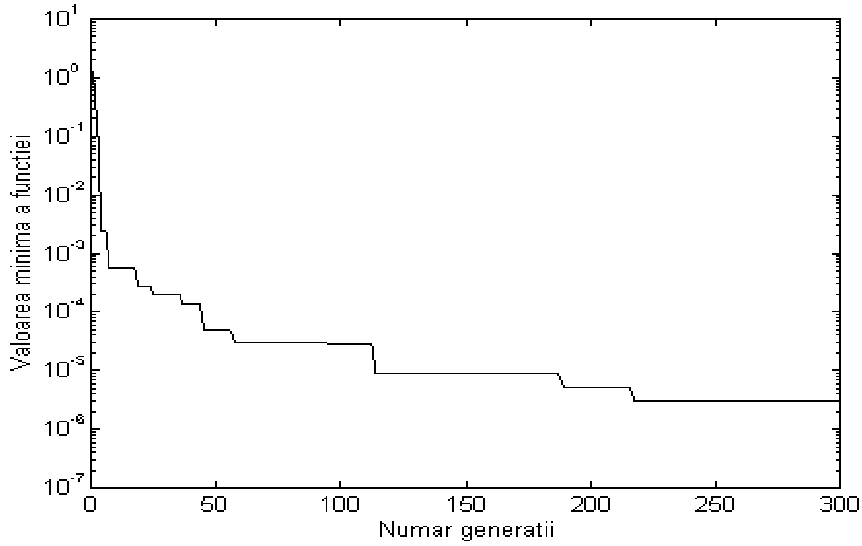

Fig. 2.5 *Rezolvarea modelului sferei cu un algoritm genetic*

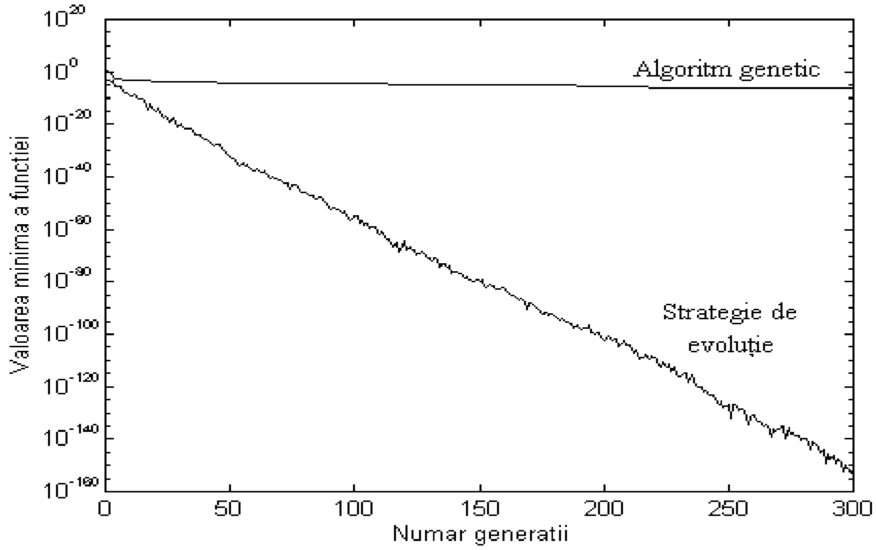

Fig. 2.6 *Comparația vitezelor de convergență*

În privința succesului căutării, pentru funcția lui Rastrigin, de exemplu, algoritmul genetic a găsit minimul global în toate cele 20 de încercări făcute, în timp ce strategia evolutivă SE(8,50) de mai sus, numai în 60% din cazuri.