Genetický algoritmus

Komentář [k1]:

1. Evolúcia v prírode

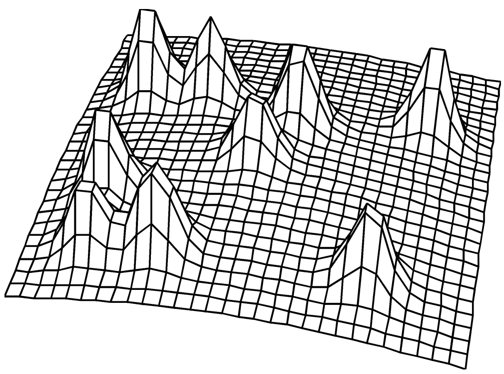
Biologická evolúcia je progresívna zmena genetického obsahu jedincov populácie v priebehu mnohých generácií. Evoluèný proces obsahuje tieto dve zložky:

(1) **Prirodzený výber** (Darwin), proces v ktorom "silnejšie" jedinci populácie majú viac potomkov v nasledujúcej generácii, ako "slabší" jedinci.
(2) **Náhodný genetický drift**, náhodne udalosti v živote jedincov populácie. Takéto udalosti sú napr. náhodná mutácia alebo náhodná smr zjavne "silného" jedinca predtým, ako dostal príležitos reprodukcie. Stochastické efekty náhodného genetického driftu sú významné hlavne pre malá populácie.

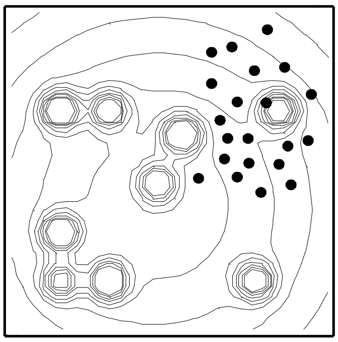
V biológii je "sila" (fitness) definovaná ako relatívna schopnos prežitia a reprodukovania sa v danom **prostredí** a v danej **populácii**. "Sila" môže by chápaná ako atribút genotypu.

Povrch "sily" (A) - vhodný prístup ako vizualizova akty selekcie v populácii, ktorá je v evolúcii. Populácia je reprezentovaná ako "oblak" bodov (B) na povrchu "sily"

Potomkovia rodièov s väèšou silou budú viac poèetnejšie a "silnejšie" ako potomkovia menej "silnejších" rodièov.



A



B

2. Abstrakcia prirodzeho výberu

(1) Jedinci populácie sú reprezentované **chromozómami** - lineárne re azce symbolov

$$P = \{x_1, x_2, \ldots, x_p\}$$

$$x \in \{\alpha, \beta, \ldots\}^n$$

(2) Každý chromozóm populácie je ohodnotený pozitívnym reálnym èíslom, ktoré sa nazýva "**sila**" (fitness)

$$f: P \to R_+$$

(3) **Sexuálna reprodukcia** (proces množenia). Dva kvázináhodne vybraní jedinci z populácie (v závislosti od ich "sily", "silnejšie" jedince sú vybrané s väèšou pravdepodobnos ou) produkujú potomkov (deti). Hlavné èasti reprodukcie sú:

(1) **selekcia** rodièov,
(2) **kríženie** a **mutácia**,
(3) **návrat** potomkov do populácie.

$$\left(x_1^{new}, x_2^{new}\right) = O_{repro}\left(x_1^{old}, x_2^{old}\right)$$

Pascalovský pseudokód prirodzeného výberu

```
P:=randomly generated population
   of chromosomes;
t:=0;
while t<t_max do
begin Q:=∅;
      while |Q|<|P| do
      begin select quasirandomly two
            parents x1,x2∈P;
            (x1',x2'):=O_repro(x1,x2);
            Q:=Q∪{x1',x2'};
      end;
      P:=Q;
end;
```

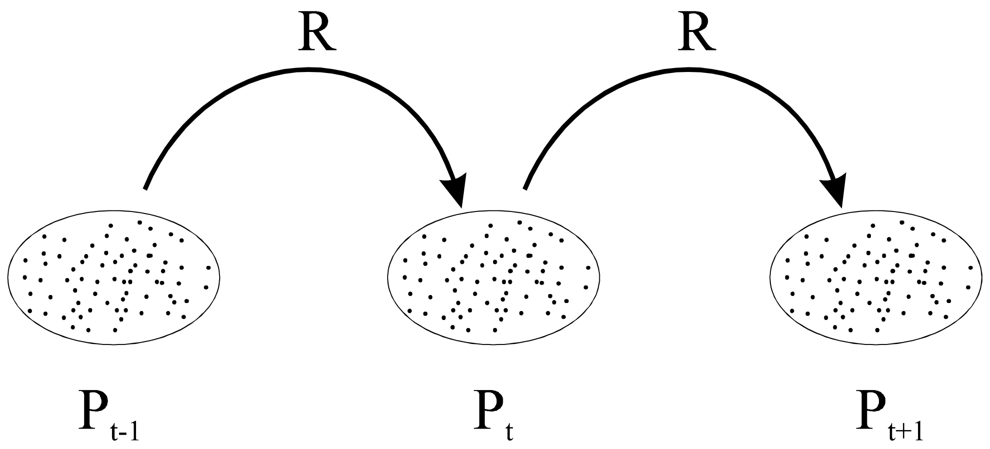
Pascalovský pseudokód obecnejšieho prístupu k prirodzenému výberu (Evoluèné algoritmy)

Funkcia (zobrazenie) R priradí **stochasticky** (vzh¾adom k "sile") aktuálnej populácii P_{t-1}

```
P0:=randomly generated population of
    solutions (chromosomes);
t:=0;
while t<tmax do
begin all solutions of Pt are
      evaluated by fitness;
      t:=t+1;
      Pt:=R(Pt-1);
end;
```

nasledujúcu populácii P_t. Efektívnos evoluèných algoritmov je urèená hlavne realizáciou funkcie R.

Evolúcia v prírode
Rekurentná konštrukcia populácie z prechádzajúcej populácie



Genetický algoritmus (GA)

Objavený **John H. Holland** (1975)

Literatúra:

1. J.H. Holland: *Adaptation in Natural and Artificial Systems. An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Michigan University Press, Ann Arbor 1975.

2. D.E. Goldberg: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Pub. Co., New York 1989.

3. L. Davis (Ed.): *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York 1991.

Základné princípy GA

1. Populácia je zložená z daného poètu chromozómov - binárnych vektorov

$$P = \left\{\alpha_1, \alpha_2, \ldots, \alpha_p\right\} \subseteq \{0,1\}^n$$

2. Nad množinou $\{0,1\}^n$ (táto množina obsahuje všetky binárne vektory dåžky n) je definovaná funkcia F

$$F : \{0,1\}^n \to R$$

ktorá priradí každému binárnemu vektoru $\alpha \in \{0,1\}^n$ reálne èíslo $y \in R$.

3. H¾adáme maximum (minimum) funkcie F nad množinou $\{0,1\}^n$

$$\alpha_{opt} = \arg \max_{\alpha \in \{0,1\}^n} F(\alpha)$$

Vo všeobecnosti, jedná sa o NP-problém (t~2^n)

4. Každý chromozóm - bitový vektor populácie P je ohodnotený "**silou**" (kladné reálne èíslo)

$$f : P \to R_+$$

Toto zobrazenie vyhovuje podmienke monotónnosti

$$\forall \alpha_1, \alpha_2 \in P : F(\alpha_1) \le F(\alpha_2) \Rightarrow f(\alpha_1) \le f(\alpha_2)$$

Renormalizovaná "sila" je urèená vz ahom

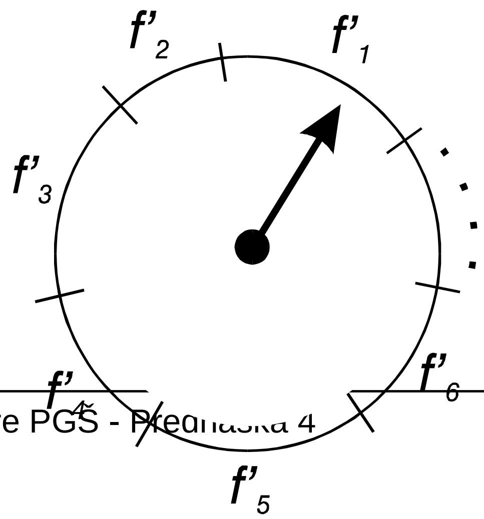
$$f'(\alpha) = \frac{f(\alpha)}{\sum\limits_{\beta \in P} f(\beta)}$$

kde

$$\forall \alpha \in P : 0 \le f'(\alpha) \le 1$$

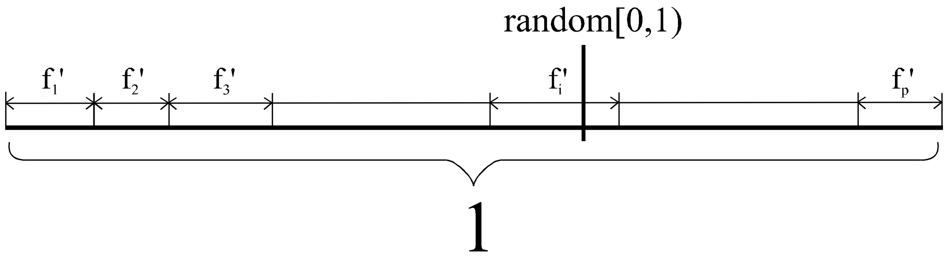
$$\sum_{\alpha \in P} f'(\alpha) = 1$$

5. Kvázináhodný výber "najsilnejších" chromozómov je realizovaný tzv. **ruletou**



i-tý chromozóm je vybraný vtedy ak

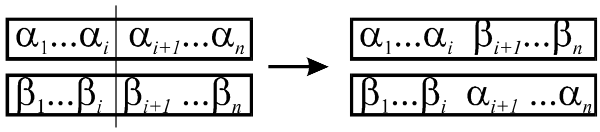
$$\sum_{k=1}^{i-1} f'_k \le \text{random}[0,1) < \sum_{k=1}^{i} f'_k$$



6. **Kríženie** - priradí každej páre chromozómov iný pár chromozómov

$$(\alpha',\beta') = O_{cross}(\alpha,\beta)$$

1-bodové kríženie



7. **Mutácia**, priradí (stochasticky) každému chromozómu iný chromozóm

$$\alpha' = O_{mut}(\alpha)$$

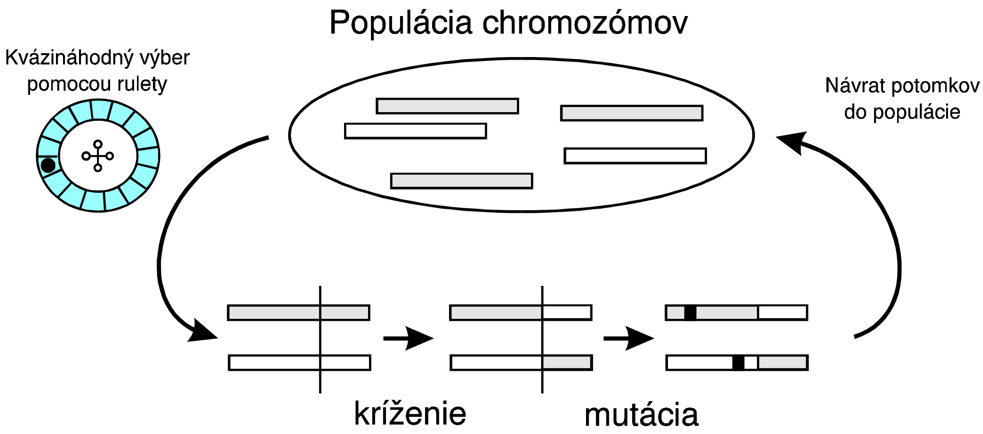
$$\boxed{\alpha_1 \ldots \alpha_{i-1}\ \alpha_i\ \alpha_{i+1} \ldots \alpha_n} \rightarrow \boxed{\alpha_1 \ldots \alpha_{i-1}\ \alpha'_i\ \alpha_{i+1} \ldots \alpha_n}$$

$$\alpha'_i = 1 - \alpha_i$$

Mutácia jedného bitu chromozómu α je realizovaná s pravdepodobnos ou $0<P_{mut}$ $<<1$

```
for i:=1 to n do
if random[0,1)<P_mut then
α_i^new:=1-α_i^old else
α_i^new:=α_i^old;
```

Diagramatické znázornenie GA



8. **Výsledné riešenie** poskytnuté GA je také riešenie, ktoré má najväèšiu "silu"

$$\exists t_0 \; \forall t \geq t_0 : \tilde{\alpha} = \arg\max_{\alpha \in P_t} f(\alpha)$$

Oèakáva sa, že platí

$$\boxed{\alpha_{opt} \approx \tilde{\alpha}}$$

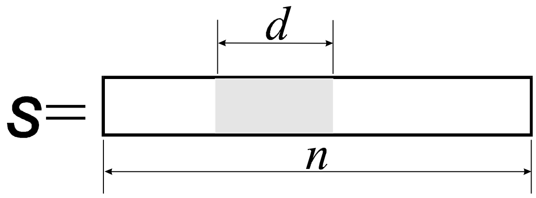
4. Ako pracuje GA?

Schéma - sekvencia bitov v chromozóme, ktorá je dôležitá pre ïalšou evolúciu populácie

$$s = (\#\#\#\#1101\#\#\#\#\#)$$

symbol # je nahradený buï 0 alebo 1.

Pravdepodobnos výskytu schémy



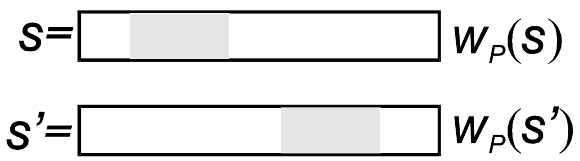
v populácii *P* je

$$w_P(s) = \frac{|P|}{2^{n-d}}$$

Napr. pre $|P|=10^3$, n=30, and d=4 dostaneme

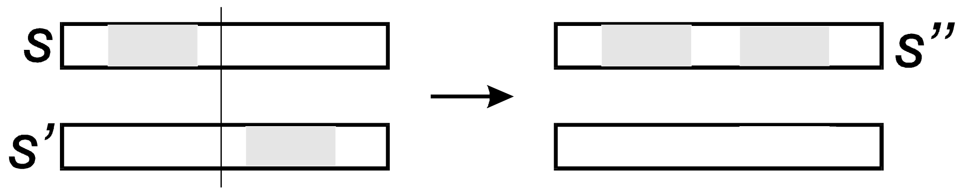
$$w_P(s) \approx 6.0 \times 10^{-5}.$$

Majme dve neprekrývajúce sa schémy *s* and *s′*



Našim cie¾om je ma   také schéma *s″*, ktoré obsahuje obe schémy *s* a *s′*

Pravdepodobnos   výskytu tejto schémy je ve¾mi malá. Pravdepodobnos   výskytu sa podstatne zvýši, ak schéma *s″* je vytvorená pomocou kríženia schém *s* a *s′*



Záver: Kríženie dramaticky zvyšuje pravdepodobnos výskytu zložitejších schém, ktoré sú dôležité pre dosiahnutie správneho riešenia.

Kríženie v poèiatoènej etape GA pôsobí ako mutácia, avšak v neskorších štádiách evolúcie, kde urèité chromozómy (riešenia) majú väèšiu silu ako ostatné, potom kríženie hrá dôležitú úlohu v "zosilnení" týchto doèasne "silných" riešení.

5. Ilustraèný príklad

1. **Binárna reprezentácia** reálnych èísel. Študujme bitový re azec dåžky n

$$\alpha = \left(\alpha_1\alpha_2\ldots\alpha_n\right) \in \{0,1\}^n$$

Môžeme by interpretovaný ako binárne èíslo, dekadické celé èíslo priradené tomuto binárnemu èíslo má hodnotu

$$\operatorname{int}(\alpha) = \sum_{i=1}^{n} \alpha_i 2^{n-i}$$

$$= \alpha_1 2^{n-1} + \alpha_2 2^{n-2} + \ldots + \alpha_{n-1} 2 + \alpha_n$$

Reálne èíslo x∈[a,b] je aproximované èíslom

$$\operatorname{real}(\alpha) = a + \frac{b-a}{2^n - 1}\operatorname{int}(\alpha)$$

Príklad. x∈[0,1] a n=3

No.	α	int(α)	real(α)
1	000	0	0
2	001	1	1/7
3	010	2	2/7
4	011	3	3/7
5	100	4	4/7
6	101	5	5/7
7	110	6	6/7
8	111	7	1

2. **Optimalizaèný problém** funkcie s jednou reálnou premenou

$$F : [a,b] \to R$$

H¾adáme globálne minimu

$$x_{opt} = \arg\max_{x\in[a,b]} F(x)$$

Predpokladajme, že reálna premenná $x \in [a,b]$ je binárne reprezentovaná, potom vyššie uvedený optimalizaèný problém je transformovaný na diskrétny optimalizaè-ný problém nad množinou binárnych re azcov dåžky n

$$\alpha_{opt} = \arg \max_{\alpha \in \{0,1\}^n} F(\mathrm{real}(\alpha))$$

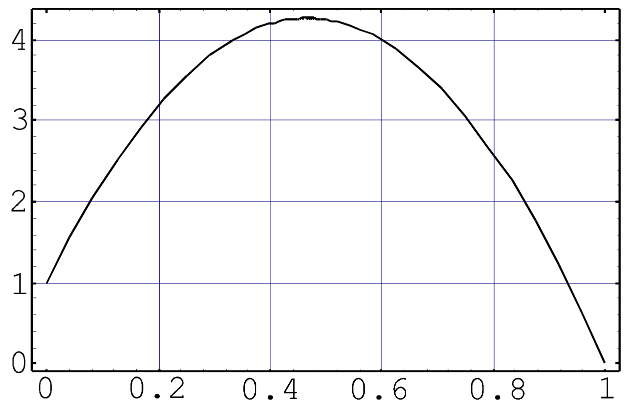
kde

$$x_{opt} \approx \mathrm{real}(\alpha_{opt})$$

3. Študujme kvadratickú funkciu

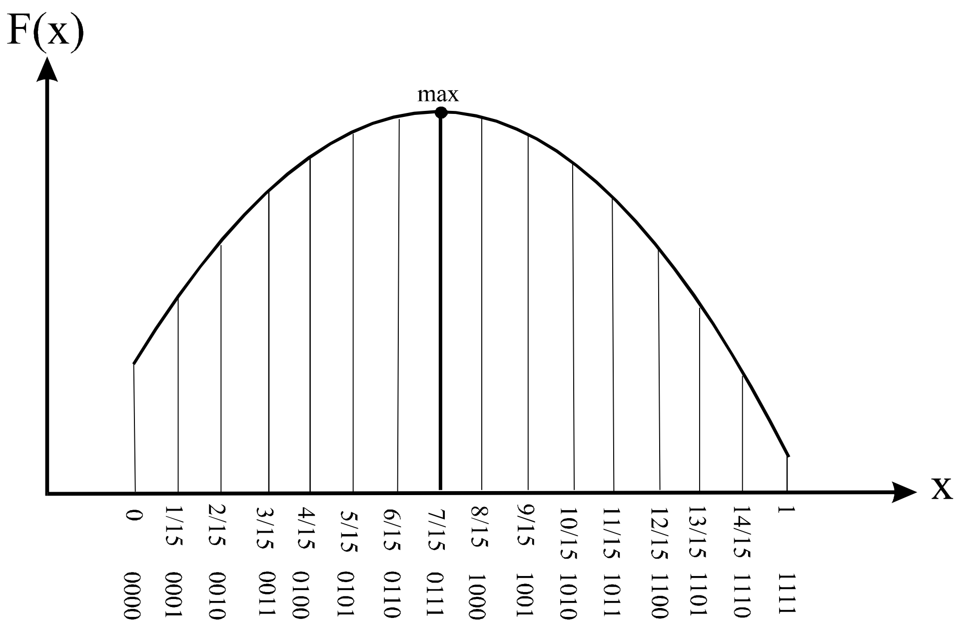
$$F(x) = 14x - 15x^2$$

pre $x \in [0,1]$. Táto funkcia má globálne maximum pre $x_{opt}=7/15$.

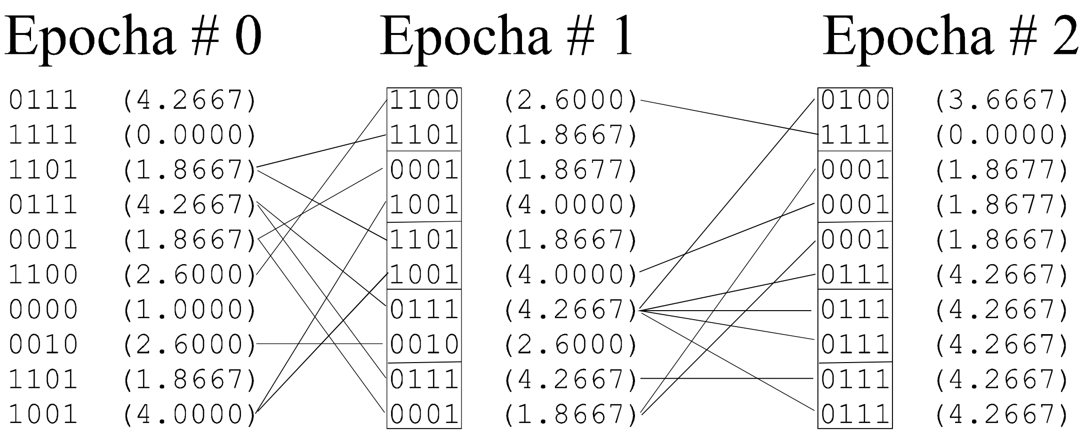


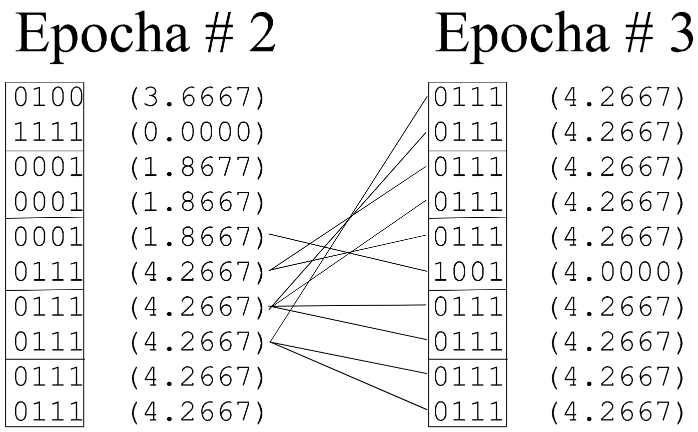
Reálna premenná x je aproximovaná binárnou reprezentáciou špecifiko-vanou $n=4$. Optimálne binárne riešenie je $\alpha_{opt}=(0111)$.

Grafický priebeh funkcie F(x) a binárna reprezentácia jednotlivých bodov z intervalu [0,1]



Prvé tri generácia genetického algoritmu



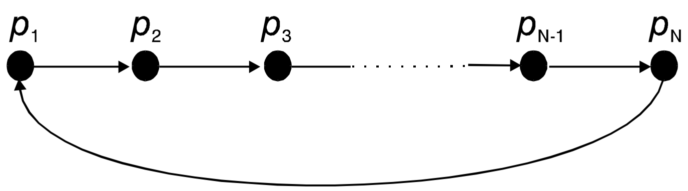


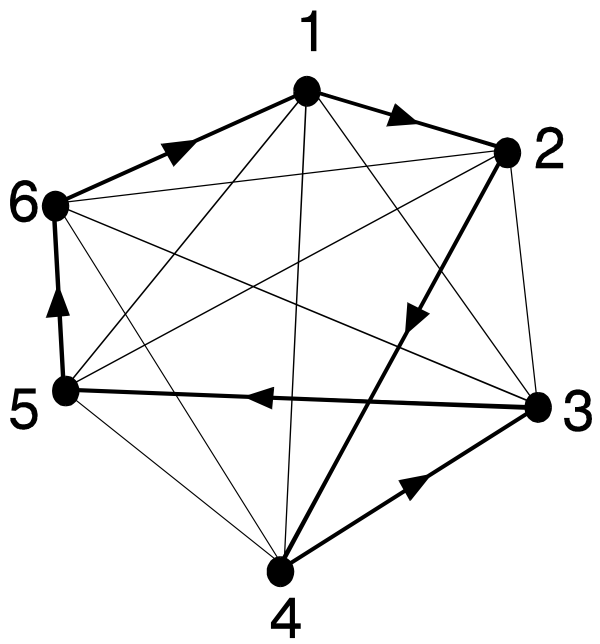
Problém obchodného cestujúceho (TSP)

Grafovo-teoretická formulácia: Nech *G* je úplný graf obsahujúci *N* vrcholov (miest) a *N*(*N*-1)/2 hrán (spojov, ciest). Každá hrana (*i,j*), spájajúca vrcholy *i* a *j* , je ohodnotená vzdialenos ou *d*(*i,j*).

Cyclická cesta obchodného cestujúceho, ktorý musí navštívi všetkých N miest (práve jeden-krát) a vráti sa do poèiatoèného mesta, je urèená permutáciou *N* objektov

$$P = \left(p_1, p_2, \ldots, p_{N-1}, p_N\right)$$





P=(1,2,4,3,5,6)

Každá cesta P=(p_1,p_2,...p_N) je ohodnotená vzdialenos ou

$$d(P) = d(p_1, p_N) + \sum_{i=2}^{N} d(p_{i-1}, p_i)$$

TSP: Hlada takú cestu P , ktorá poskytuje **minimálnu** vzdialenos $d_{\min}= d(P_{min})$

$$P_{min} = \arg\min_{P} d(P)$$

Toto je tzv. NP-obtiažny problém, jeho CPU èas rastie rýchlejšie ako polynomiálne pre $N\to\infty$

$$t_{CPU} \approx N!$$

Použitie presných (enumeratívnych) metód k riešeniu TSP je beznádejne obtiažny numerický problém pre väèšie hodnoty N.

Táto skutoènos je hlavným dôvodom preèo TSP je riešený GA (alebo iným evoluèným algoritmom).

Ako modifikova GA pre úèely TSP?

1. Populácia je tvorená množinou

permutácií *N* objektov.

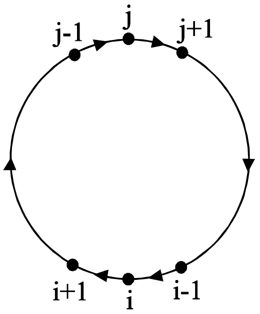
2. Každý chromozóm *P* (permutácia) je ohodnotený "silou" $f(P)$

$$d(P_1) \le d(P_2) \Rightarrow f(P_1) \ge f(P_2)$$

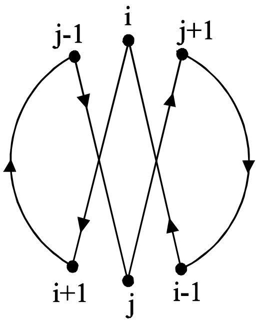
3. Mutácia chromozómov je realizovaná transpozíciou dvoch náhodne vybraných komponent

$$P\ = (p_1, \ldots, p_i, \ldots, p_j, \ldots p_N)$$
$$P' = (p_1, \ldots, p_j, \ldots, p_i, \ldots p_N)$$

Diagramatická vizualizácia mutácie



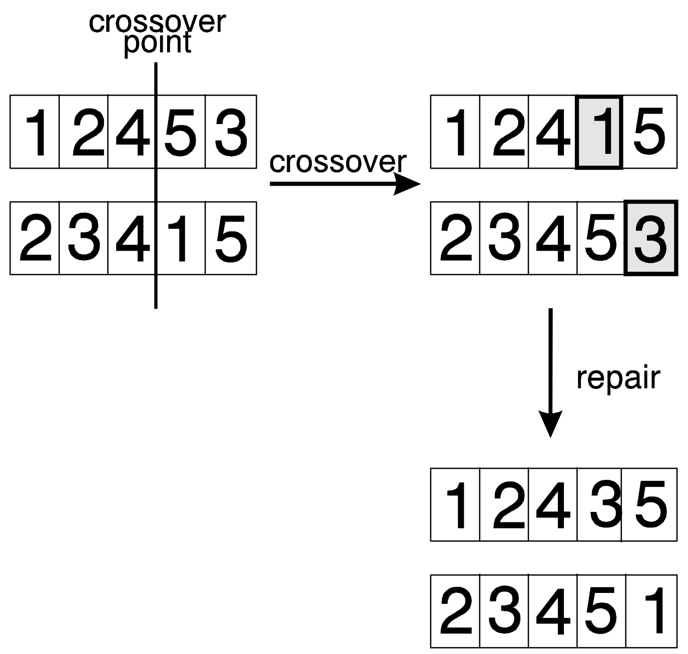
$$P=\left(p_1 \dots p_{i-1} p_i p_{i+1} \dots p_{j-1} p_j p_{j+1} \dots p_N\right)$$



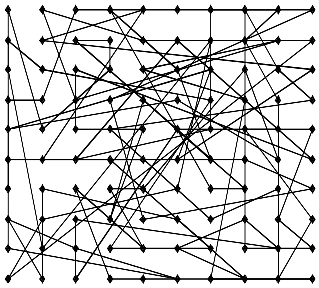
$$P=\left(p_1 \dots p_{i-1} p_j p_{i+1} \dots p_{j-1} p_i p_{j+1} \dots p_N\right)$$

4. Kríženie dvoch chromozómov je realizované
výmenou dvoch náhodnevybraných èastí hromozómov. Vytvorené chromozómy nemusia by permutáciami, preto opravný

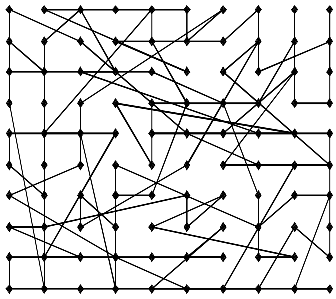
proces je aplikovaný na vytvorené chromozómy.



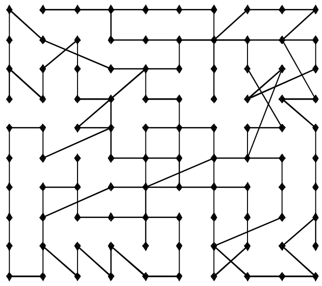
Štvorcová mriežka typu 10×10
(100 vrcholov)
Ve¾kos populácie=200 chromozómov



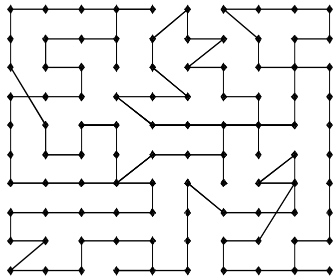
Epoch=0



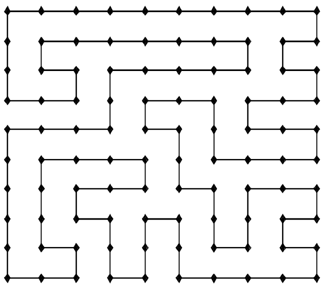
Epoch=100



Epoch=200



Epoch=300



Epoch=1500

Závery

1. GAs je **robustný** a **efektívny** numerický prostriedok pre nájdenie globálneho minima funkcií, ktoré sú vysoko multimodálne.

2. V praktických aplikáciách je doporuèo-vané používa   tzv. **hybridný GA,** ktoré kombinujú GA s klasickými optimalizaènými metódami. Obvykle, ich efektívnos   sa dramaticky zvýši. Prinajhoršom, doporuèuje sa použi   GA ako **generátor poèiatoèných riešení** pre klasické optimalizaèné metódy.

3. Binárna reprezentácia premenných v GA môže by   zamenená za "reálnu" reprezen-táciu, t.j. chromozóm je vektor reálnych premenných.

4. Implementácia GA v C++ alebo Pascale je ve¾mi jednoduchá.