7. prednáška

Kombinatoriálne optimalizačné problémy

Formulácia základných kombinatoriálnych optimalizačných problémov

Jeden z prvých veľkých úspechov evolučných algoritmov bolo ich použitie pre riešenie notoricky obtiažnych kombinatoriálnych problémov, ktorých CPU čas rastie buď faktoriálovo alebo exponenciálne s rastom dimenzie problému (NP = "**n**on**p**olynomialy" obtiažne).

Dva typy kombinatoriálnych problémov

Typ 1. Funkcia f je definovaná nad symetrickou grupou S_N zloženou zo všetkých permutácií N objektov

$$f : S_N \to R$$

kde permutácie z S_N sú definované ako N-tica celých rôznych čísel

$$P = \left(p_1, p_2, \ldots, p_N \right)$$

Optimalizačný problém má tvar

$$P_{opt} = \arg \min_{P \in S_N} f\left(P\right)$$

Riešenie tohto optimalizačného problému je obvykle veľmi obtiažne v dôsledku skutočnosti, že symetrická grupa S_N obsahuje $N!$ permutácií. Z týchto dôvodov môže nastať situácia, že pre veľké N CPU čas zhruba rastie ako $N!$.

Typ 2. Nech R_{set}={1,2,...,r} je množina obsahujúca prvých r celých čísel, jej priamy súčin R_{set}^N obsahuje N-tice

$$\pi = (\pi_1, \pi_2, ...\pi_N)$$

Jednotlivé komponenty tohto výrazu sú celé čísla z uzavretého intervalu [1,r]. Funkcia

$$f : R_{set}^N \rightarrow R$$

zobrazuje N-tice π na reálne čísla, optimalizačný problém má tvar

$$\pi_{opt} = arg \min_{\pi \in R_{set}^N} f(\pi)$$

V dôsledku toho, že kardinalita množiny R_{set}^N je r^N, optimalizačný problém môže patriť medzi obtiažne s exponenciálnym rastom CPU času pre veľké N. Poznamenajme, že pre r=2 sa tento optimalizačný problém redukuje na obyčajný binárny problém.

Úloha obchodného cestujúceho

Typ 1 kombinatoriálneho problému bude ilustrovaný známou úlohou obchodného cestujúceho (TSP, Traveling Salesman Problem).

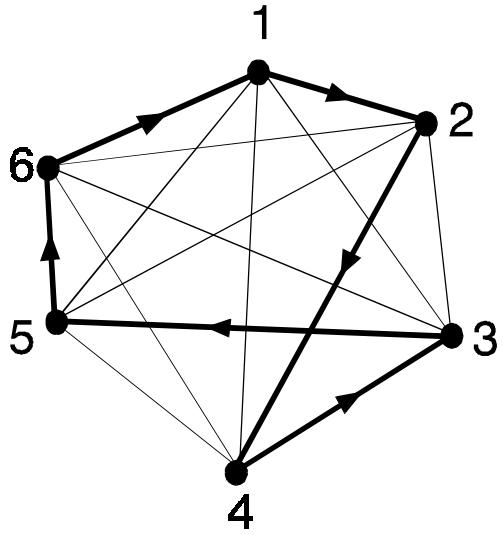
Grafovo-teoretická formulácia je nasledovná

Nech G je úplný graf obsahujúci N vrcholov a $N(N-1)/2$ hrán (spojov).

Každá hrana $[i,j]$, spájajúca i-ty s j-tym vrcholom (mestom) je ohodnotená kladným číslom $d(i,j)$, ktoré sa interpretuje ako vzdialenosť medzi danými vrcholmi - mestami.

Služobná cesta obchodného cestujúceho (Hamiltónov cyklus) navštívi každé mesto len raz, pričom sa vracia do východzieho mesta.

Táto cesta je určená pomocou permutácie N objektov - miest, odpovedá ceste, ktorá je zahájená v meste p_1, potom pokračuje postupne v mestách p_2, p_3, ..., p_N, a na záver sa vráti do východzieho mesta p_1.



Každej ceste je priradená jej dĺžka

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

Cieľ úlohy TSP je nájsť takú cestu - permutáciu P_{opt} , ktorá poskytuje minimálnu dĺžku cesty $f_{opt}=f(P_{opt})$.

Mutácia permutácie - cesty

Cesta P môže byť zmenená na novú cestu P' pomocou stochastického operátora mutácie O_{mut} špecifikovaného pravdepodobnosťou P_{mut}.

$$P \in S_N \Rightarrow P' = O_{mut}(P) \in S_N$$

$$\lim_{P_{mut} \to 0} O_{mut}(P) = P$$

```
procedure Permutation_Mutation;
begin for i:=1 to N do p'i:=pi;
      for i:=1 to N do
      if random<Pmut then
      begin j:=1+random(N);
            aux:=p'i;
            p'i:=p'j;
            p'j:=aux
      end;
end;
```

$O_{\text{mut}}(2,1,4,3,5) \rightarrow (2,5,4,3,1)$

Kríženie pernutácií - ciest

K tomu, aby sa tento typ kombinatotiálnych úloh mohol študovať genetickým algoritmom, musíme zaviesť ešte operáciu kríženia medzi dvoma permutáciami

$$(P',Q') = O_{cross}(P,Q)$$

ktoré zachováva ich charakter permutácie, t.j. P' a Q' sú taktiež permutácie. Študujme dve permutácie P a Q n objektov, vyberme bod kríženia a tak, že $1<a<n$

$$P = (p_1,...p_a, p_{a+1},..., p_n),\ Q = (q_1,...q_a, q_{a+1},..., q_n)$$

Keď vymeníme segmenty, ktoré nasledujú za bodom kríženia, dostaneme dva nové objekty

$$\hat{P} = (p_1,...p_a, q_{a+1},..., q_n),\ \hat{Q} = (q_1,...q_a, p_{a+1},..., p_n)$$

Žiaľ, takto vytvorené objekty nemusia byt už permutácie.

Môže nastať situácia, že nejaké celé číslo sa v objektoch $\hat{P}$ alebo $\hat{Q}$ vyskytuje dvakrát. Z tohto dôvodu je potrebné aplikovať "opravný proces" (nazývaný čiastočné priradenie - partial matching), ktorý z objektov $\hat{P}$ a $\hat{Q}$ vytvorí normálne permutácie (poznamenajme, že v literatúre je popísaných mnoho rôznych druhov kríženia dvoch permutácií a spôsobov ich opráv).

Pre ilustráciu študujme kríženie medzi dvoma permutáciami

$$P = (3,4,5,1,2,6,10,8,9,7)$$
$$Q = (1,2,6,10,8,3,4,5,7,9)$$

predpokladajme, že bod kríženia je a=4. Ak vymeníme medzi dvoma permutáciami segmenty po bode kríženia dostaneme

$$\hat{P} = (3,4,5,1,8,3,4,5,7,9),$$
$$\hat{Q} = (1,2,6,10,2,6,10,8,9,7)$$

Vidíme, že objekty $\hat{P}$ a $\hat{Q}$ nie sú permutácie, pretože obsahujú niektoré indexy dvakrát.

K oprave týchto objektov na permutácie zostrojíme zobrazenia f_{QP} a f_{PQ}

$$f_{PQ} = \begin{pmatrix} 2\,6\,10\,8\,7\,9 \\ 8\,3\,4\,5\,9\,7 \end{pmatrix}$$

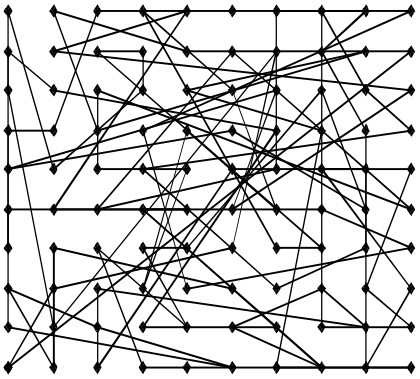
$$f_{QP} = \begin{pmatrix} 8\,3\,4\,5\,9\,7 \\ 2\,6\,10\,8\,7\,9 \end{pmatrix}$$

Objekt $\hat{P}$ opravíme pomocou zobrazenia f_{QP} a objekt $\hat{Q}$ pomocou zobrazenia f_{PQ}. Prvé tri indexy v P' zameníme za $3\rightarrow6$, $4\rightarrow10$ a $5\rightarrow8\rightarrow2$. Podobne, prvé indexy 2, 6 a 10 v Q' zameníme za $2\rightarrow8\rightarrow5$, $6\rightarrow3$ a $10\rightarrow4$, dostaneme opravené objekty, ktoré už sú permutácie a sú považované za výsledok výmeny medzi permutáciami P a Q

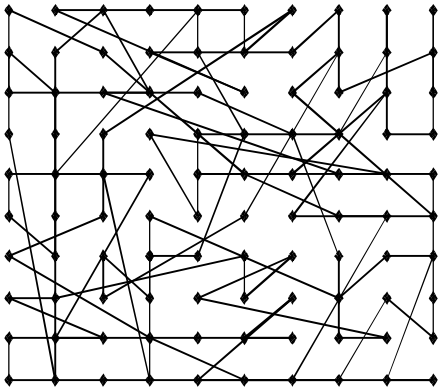
$$P' = (6,10,2,1,8,3,4,5,7,9)$$

$$Q' = (1,5,3,4,2,6,10,8,9,7)$$

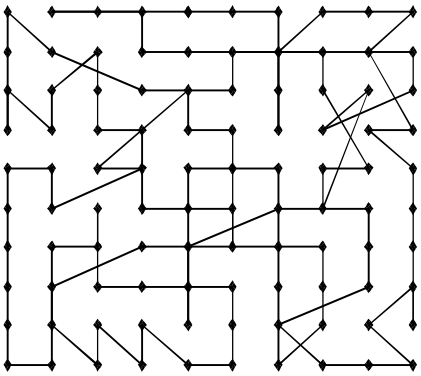
Úloha obchodného cestujúceho na mriežke 10×10



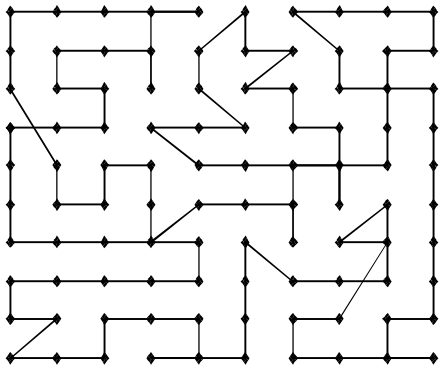
f_{opt}=462, Epoch=0



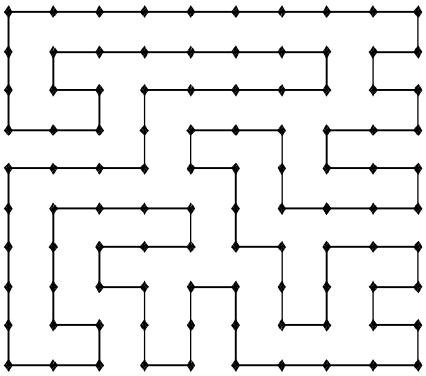
f_{opt}=298, Epoch 500



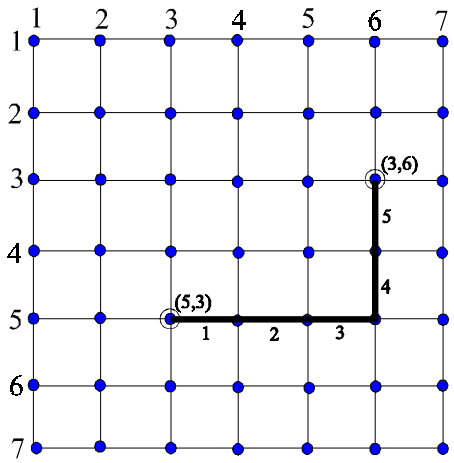
f_{opt}=164, Epoch=1000



f_{opt}=124, Epoch=2000



f_{opt}=100, Epoch=5000

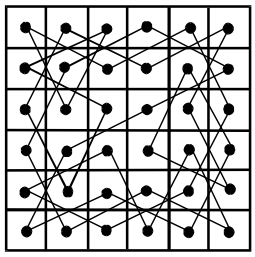


Ortogonálna mriežka bodov typu N×N, pričom vzdialenosť medzi jednotlivými bodmi sa počíta pomocou Hammingovej (L_1) vzdialenosti. Pre tento špeciálny prípad optimálna vzdialenosť je

$$f_{opt} = \begin{cases} N^2 \,(\text{pre párne } N) \\ \\ N^2 + 1 \,(\text{pre nepárne } N) \end{cases}$$

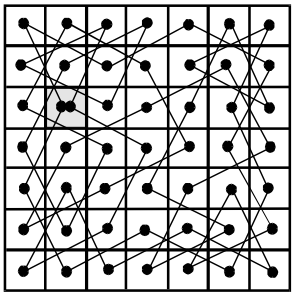
Pohyb koňom po šachovnici

Chessboard 6x6

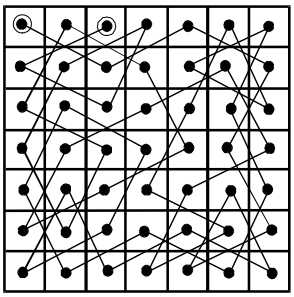


A

Chessboard 7x7

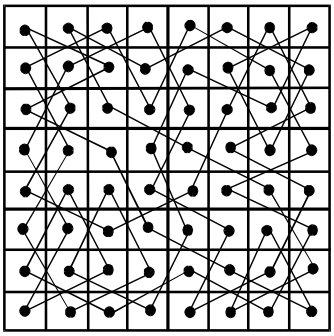


B

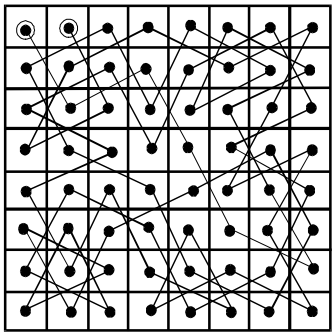


C

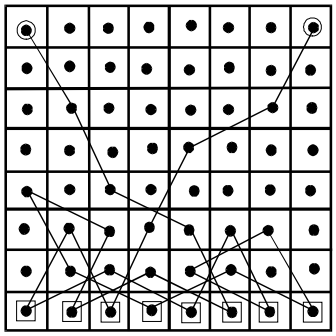
Chessboard 8x8



D



E



F

Problém rozkladu čísla

NPP, Number Partitioning Problem

Nech $Q=\{q_1,q_2,...,q_N\}$ je množina obsahujúca N kladných reálnych čísel a nech π je zobrazenie Q na množinu $R_{set}=\{1,2,...,r\}$

$$\pi : Q \rightarrow R_{set}$$

Toto zobrazenie π môže byť jednoznačne vyjadrené ako N-tica $\pi = (\pi_1, \pi_2, ..., \pi_N) \in R_{set}^N$. Jej jednotlivé komponenty sú interpretované tak, že celé číslo $\pi_i \in R_{set}$je priradené objektu $q_i \in Q$. Vzhľadom k tomuto zobrazeniu množina Q môže byť rozložená na disjunktné podmnožiny

$$Q = Q_1 \cup Q_2 \cup ... \cup Q_r$$
$$Q_i = \{q \in Q; \pi(q) = i\}$$

Podmnožina Q_i obsahuje všetky elementy - čísla množiny Q, ktoré sú zobrazené funkciou π na celé číslo i. Definujme účelovú funkciu

$$f(\pi) = \max_i \sum_{q \in Q_i} q - \min_i \sum_{q \in Q_i} q$$

Vyjadruje rozdiel medzi maximálnou a minimálnou sumou čísel z podmnožín $Q_1, Q_2, ..., Q_N$. Cieľom NPP úlohy je hľadať také zobrazenie π, ktoré minimalizuje účelovú funkciu f

$$\pi_{opt} = \arg \min_{\pi \in R_{set}^N} f(\pi)$$

Mutáciou zobrazenia π zostrojíme iné zobrazené π'

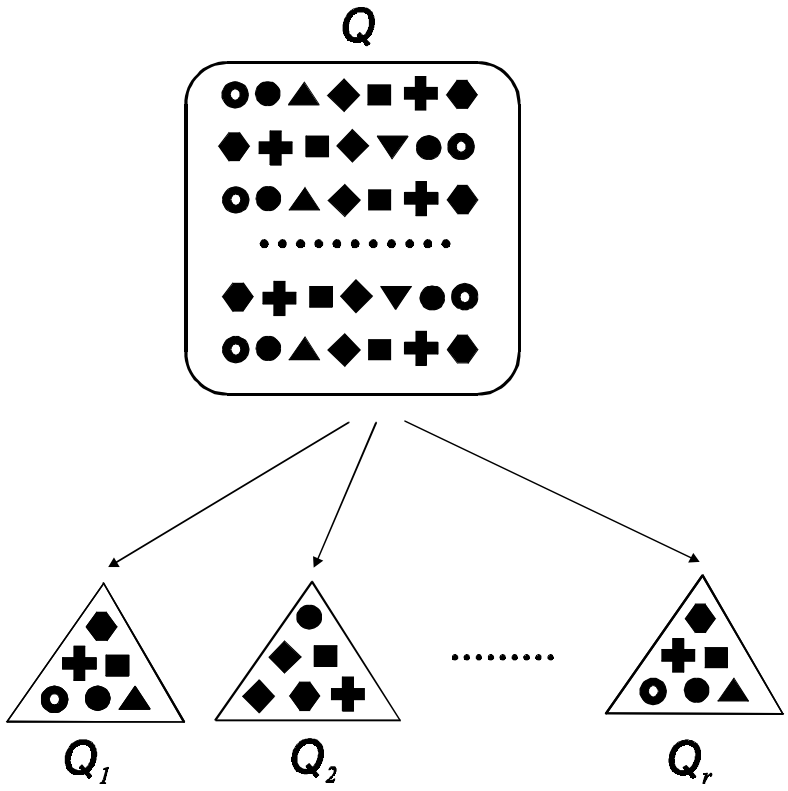
$$\pi' = O_{mut}(\pi)$$

```
procedure Mapping_Mutation;
begin for i:=1 to N do
      if random<P_mut then
      π'_i:=1+random(r) else π'_i:=π_i;
end;
```

Operácia **krížemia** pre tento typ chromozómov je veľmi jednoduchá a môže sa priamo stotožniť s podobnou operáciou pre binárne vektory.

Ako názorne interpretovať túto úlohu?

Predstavme si, že máme hromadu obsahujúcu N vecí, pričom každej veci je priradená cena $q_1,q_2,...,q_N$. Cieľom je rozdeliť túto kopu vecí na r hromád tak, aby ich ceny (súčty cien jednotlivých vecí patriacich do tej ktorej hromady) vykazovali minimálne rozdiely medzi sebou.



Alternatívny popis permutácie

Permutácia N objektov (kombinatoriálne úlohy typu 1) môže byť zadaná ako binárny vektor dĺžky kN, kde k je také celé číslo, ktoré zabezpečuje, aby binárny vektor dĺžky k bol schopný reprezentovať číslo N.

Tento binárny vektor je ľahko prepísaný na vektor N celých čísel z intervalu $[0,2^k-1]$.

Potom permutácia P je definovaná tak, že celočíselné komponenty sú usporiadané do nerastúcej alebo neklesajúcej postupnosti.

Nech N=5, k=3, binárny vektor

(101|011|001|111|101)

jeho celočíselná verzia má tvar (5,4,1,7,5). Nech permutácia P usporiada túto N-ticu do rastúcej postupnosti, potom

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

Nevýhodou tohto prístupu pre kódovanie permutácií je, že rovnaká permutácia je určená rôznymi binárnymi reťazcami. Toto zvýšenie redundancie kódovania môže v niektorých prípadoch byť dokonca výhodné, ako spôsob zníženia pravdepodobnosti "zamrznutia" v lokálnom minime.