

# PB165 – Grafy a sítě

## 9. Plánování v telekomunikačních sítích

- Optimalizace
  - Lokální prohledávání
  - Evoluční prohledávání
- Stručný náhled na plánování mobilních sítí
  - Topologie pevné sítě
  - Alokace kanálů
  - Umístění základnových stanic
- Shrnutí

- Soustředíme se na *optimalizaci*
- Zpravidla kombinatorické problémy
  - Nejvhodnější topologie privátní datové sítě
  - Rozložení základnových stanic pro GSM sítě
  - Rozdělení frekvencí u bezdrátových sítí
  - Nejlepší topologie „poslední míle“
- Co mají společné
  - Diskrétní kombinatorické problémy
  - Patří mezi NP-těžké úlohy
  - Používáme *heuristická řešení*

- Přibližná řešení NP-těžkých problémů
- Prohledávání
  - Lokální prohledávání a jeho varianty
  - Evoluční prohledávání
- Standardní techniky
  - Lineární programování
  - Dynamické programování atd.
- Proč heuristiky
  - Použití snazší (intuitivnější)
  - Větší šance najít alespoň nějaké řešení

- Není možné rozhodnout, které techniky jsou lepší
- Empirické pozorování (Corne et al. Telecommunications Optimization., John Wiley, 2000):
  - Expertovi na heuristiky vychází lépe řešení pomocí heuristik
  - Expertovi na standardní techniky operačního výzkumu vychází lépe tyto techniky
- Závěr: Nejlepší je použít techniku, které dobře rozumíte
  - A tady je výhoda heuristik, protože jsou snáze pochopitelné

- Řešíme problém, pro který existuje mnoho řešení
- Jednotlivá řešení neznáme, ale umíme je generovat
- Hledáme řešení splňující nějaké podmínky
  - Triviální (brute force) přístup: vygenerujeme všechna řešení a vybereme to „správné“
  - Velikost prostoru toto řešení neumožňuje
- Optimalizace
  - Hledáme řešení, které minimalizuje *cenu* (*cost function*)
- Lokální prohledávání: Začne s nějakým řešením a to „zlepšuje“ prohledáváním jeho okolí
  - Potřebuje definovat to „okolí“
  - Malá změna stavu
    - Přidání či ubrání vrcholu nebo hrany
    - Posun hrany
  - Příslušnou operaci nazýváme *mutace*

- 1 Nalezni nějaké (počáteční) řešení  $c$  a spočti jeho cenu  $f(c)$
- 2 Proveď mutaci  $c \rightarrow c'$  a spočti cenu  $f(c')$
- 3 Pokud  $f(c') \leq f(c)$ , nahraď řešení  $c$  řešením  $c'$
- 4 Je-li splněno kritérium ukončení, skonči, jinak se vrať na bod 2

V podstatě se jedná o slézání z kopce

- V literatuře se často setkáte s pojmem *hillclimbing*, tedy „stoupání do kopce“. V takovém případě se snažíme  $f(c)$  maximalizovat.

Důležitý je výběr počátečního řešení

- Nelze očekávat, že ze špatného řešení se rychle najde dobré

- Lokální prohledávání může „uváznout“
  - Nalezeno *lokální minimum*
  - Z něj se nelze dostat bez porušení pravidla striktního ne-zhoršování ceny
- Dva často používané přístupy
  - Simulované žíhání: s určitou pravděpodobností akceptujeme i horší řešení
  - Tabu prohledávání: Zohledňuje typ mutace, ne jen cenu



# Simulované žíhání (Simulated Annealing)

- Zavádí teplotu  $T$ 
  - Při vyšší teplotě je materiál „tvárný“
  - Vyšší teplota – vyšší pravděpodobnost akceptace horšího řešení
- Spočte funkci  $F(f(c), f(c'), T)$ , např.  $e^{(f(c)-f(c'))/T}$   
( $F > 1$  pro  $f(c') < f(c)$ )
- Vygeneruje náhodné číslo  $r \in (0, 1)$
- Pokud  $F > r$ , pak konfigurace  $c'$  je akceptována
- Dodatečný „trik“: teplota klesá s počtem iterací

- 1 Nalezni nějaké (počáteční) řešení  $c$  a spočti jeho cenu  $f(c)$ ; nastav teplotu  $T$  a parametr chlazení  $q$  ( $0 < q < 1$ )
- 2 Proveď mutaci  $c \rightarrow c'$  a spočti cenu  $f(c')$
- 3 Pokud  $\text{test}(f(c'), f(c), T)$  platí, nahraď řešení  $c$  řešením  $c'$  (test je funkce popsaná výše)
- 4 Uprav teplotu  $T = qT$
- 5 Je-li splněno kritérium ukončení, skonči, jinak se vrať na bod 2

Na počátku vyhledávání se akceptují i výrazně horší řešení

- Pamatuje si předchozí změny
- Zavádí *tabu* (tedy zakázané) změny
  - Např. vrchol, jehož hrany se změnily v posledních krocích, nesmí být už měněn
- Vybere nejlepší z povolených změn
  - Což nemusí být absolutně nejlepší konfigurace
- Podstatný výběr kritérií, podle nichž se zařazuje do zakázaných (*tabu*) seznamů
  - Otázka zkušenosti uživatele této heuristiky

- Důležitá reprezentace a operátor mutace
  - Topologie sítě jako seznam dvoubodových spojení, alternativně jako bitový seznam existujících vs. možných spojení atd.
- Aplikace dodatečné znalosti
  - Např. v počítačové síti nechceme izolované vrcholy
    - Vyloučíme je tedy vždy z uvažování (mutace vedoucí k rozpadu sítě neakceptujeme)
  - Redundance – akceptujeme jen uzly se stupněm alespoň 2
  - Složitější – např. požadavek existence kostry (garantuje spojitost a může být použito přímo při aplikaci operátoru mutace)
  - Znalost *ceny* (resp. funkce jejího výpočtu) může rovněž přímo ovlivňovat operátor mutace
    - Aplikace algoritmů na nalezení kostry s minimální cenou a následně již jen přidáváme hrany k této kostře (i odpovídající reprezentace)

- Při hledání nové „mutace“ nepoužíváme jen jednoho předka, ale celou populaci
  - Prohledáváme paralelně několik okolí
  - Provádíme *rekombinaci*
- Rekombinace
  - Dva nebo více „rodičů“
  - Vhodnou operací se spojí jejich vlastnosti (např. každá část sítě je od jiného „rodiče“)
  - Výsledek se nazývá *rekombinant* (na rozdíl od mutanta)
  - Cílem rekombinace je opět překonat lokální optimum – rekombinace vybere vzdálenou konfiguraci, která dědí pro dvou „dobrých rodičích“
- *Crossover*
  - Máme stav popsán lineárním vektorem
  - Výsledná konfigurace náhodný výběr hodnot z „rodičovských“ konfigurací (vektorů)
  - Rekombinant („dítě“) se může i o 50 % lišit od kteréhokoliv „rodiče“

- 1 Inicializuj populaci vhodně zvolenými konfiguracemi.  
Spočti cenu každé konfigurace.
  - 2 Vyber rodiče (několik párů, resp.  $n$ -tic)
  - 3 Použij rekombinaci a mutaci a vytvoř „děti“
  - 4 Začleň „děti“ do populace
  - 5 Pokud je dosaženo kritéria ukončení, skonči, jinak přejdi na bod 2
- Algoritmus má velmi mnoho stupňů volnosti
    - Výběr rodičů je zpravidla dán jejich cenou (nicméně je vhodný i určitý náhodný prvek)
    - Je možné se omezit jen na rekombinace nebo rekombinace a mutace; mutují se „rodiče“, jen zřídka „děti“
    - Začlenění „děti“ do populace
      - Udržení „genové“ diversity

- Návrh topologie pevné sítě
- Přiřazení kanálů (frekvencí)
- Umístění základnových stanic
- Správa mobility
- Správa volání
- Detekce uživatelů v CDMA sítích
- TDMA alokace slotů
- Ekvalizace dat

Jen některé mají charakter grafových problémů

- Hierarchie prvků mobilní sítě
  - *Mobilní stanice* (MS) komunikují bezdrátově se *základnovými stanicemi* (BTS). Ty jsou kontrolovány *řídícími stanicemi* (BSC), které jsou sdruženy pod *přepínací stanice* (MSC). Samotné přepínací stanice pak tvoří páteř mobilní sítě, protože odpovídají autentizaci, účetnictví, údržbu databází apod.
- Vidíme, že většina sítě je fixní (bezdrátové spojení je nezbytné jen pro kontakt MS s BTS)
- Návrh topologie s minimální cenou souvisí s nalezením minimální kostry
  - Příklad ceny:

$$f = \sum_{\forall n} C_n^{\text{NODE}} + \sum_{\forall p} C_p^{\text{POI}} + \sum_{\forall l \in L^{\text{BTS} \rightarrow \text{BSC}}, L^{\text{BSC} \rightarrow \text{MSC}}, L^{\text{MSC} \rightarrow \text{MSC}}} C_l^{\text{LINK}}$$

s okrajovými podmínkami  $F_l \leq C_l, \forall l, \theta \leq 0.001$  kde  $C_n^{\text{NODE}}$  je cena vrcholů typu  $n$ ,  $C_p^{\text{POI}}$  je cena přípojného místa do veřejné sítě (Point of Interconnect),  $C_l^{\text{LINK}}$  je cena spoje  $l$  typu  $L^{\text{BTS} \rightarrow \text{BSC}}, L^{\text{BSC} \rightarrow \text{MSC}}, L^{\text{MSC} \rightarrow \text{MSC}}$  a  $F_l, C_l$  reprezentují tok a kapacitu spoje  $l$  a  $\theta$  je pravděpodobnost nepřijetí volání.



- Počet frekvenčních kanálů které mohou být použity

$$N_k = \text{int}(\Delta F / F_k)$$

kde  $F_k$  je šířka jednoho pásma

- Počet frekvencí potřebných pro obsluhu uživatelů ( $M$ ) v jednom sektoru každé buňky ( $C$  je počet sektorů)

$$n_s = \text{int}(N_k / C \times M)$$

- Akceptovatelná zátěž v jednom sektoru nebo v celé buňce (pokud jsou použity cirkulární antény)

$$A = n_0 [1 - \sqrt{1 - (P_{sl} / \sqrt{\pi n_0 / 2})^{1/n_0}}] \quad \text{at} \quad P_{sl} \leq \sqrt{\frac{2}{\pi n_0}};$$

$$A = n_0 + \sqrt{\frac{\pi}{2} + 2n_0 \ln((P_{sl} / \sqrt{\pi n_0 / 2}) - \sqrt{\frac{\pi}{2}})} \quad \text{at} \quad P_{sl} > \sqrt{\frac{2}{\pi n_0}}$$

kde  $n_0 = n_s \times n_a$ ;  $n_a$  je počet uživatelů, kteří mohou současně používat jeden kanál. Tento údaj definuje standard.

## Příklad technických parametrů

- Počet uživatelů jedné základní stanice

$$N_{aBTS} = M \times \text{int}(A/\beta).$$

- Nezbytný minimální počet základnových stanic propokrytí dané geografické oblasti

$$N_{BTS} = \text{int}(N_a/N_{aBTS})$$

kde  $N_a$  je počet uživatelů, které má jedna základová stanice obsluhovat.

- Poloměr buňky je definován (za předpokladu rovnoměrného pokrytí)

$$R = \sqrt{\frac{1,21 \times S_0}{\pi N_{BTS}}}.$$

- Bezpečná vzdálenost mezi základnovými stanicemi používajícími stejnou frekvenci

$$D = R\sqrt{3C},$$

- K dispozici je pouze velmi omezený počet kanálů/frekvencí
- Základní idea mobilní sítě: znovupoužití frekvencí v geograficky oddělených částech sítě
  - *Buňka* – část sítě s množinou přidělených frekvencí
  - Celé spektrum je přiděleno *clusteru* buněk
  - Buňky zpravidla hexagonální (pokrytí plochy)
    - Geometrie omezuje počet buněk v clusteru

$$N = i^2 + ij + j^2$$

kde  $i$  a  $j$  jsou celá čísla ( $N$  je tedy 1, 3, 4, 7, ...)

- Problémem je interference

- Podmínky alokace
  - 1 Interference frekvencí mezi buňkami (buňky se stejným kanálem)
  - 2 Interference frekvencí na buňce (minimální spektrální vzdálenost kanálů)
  - 3 Požadavky buňky na počet kanálů
- Problém přidělení kanálů je ekvivalentní zobecněnému problému barvení grafů
- Podvarianty: *pevné* a *dynamické* přidělení kanálů

- Pokrytí terénu signálem při minimalizaci počtu základnových stanic
- Problém velmi podobný nalezení minimální dominující množiny grafu

## Definice

- *Dominující množina grafu je taková množina jeho vrcholů, že všechny ostatní vrcholy jsou spojeny s alespoň jedním vrcholem dominující množiny*

- Mnoho problémů návrhu sítě má charakter optimalizace grafově vyjádřeného problému
- Optimalizace je zpravidla NP-těžký problém, řešený heuristicky
- Populární heuristiky využívají *lokální* nebo *evoluční prohledávání*
  - Negarantují nalezení globálního optima
  - Na druhé straně mohou poskytnout cenné informace o stavovém prostoru možných řešení
  - A nalezená řešení zpravidla akceptovatelná
- Příklady problémů plánování mobilní sítě jako optimalizace grafově orientovaných problémů