Pochôdzky v grafoch

Stanislav Palúch

Fakulta riadenia a informatiky, Žilinská univerzita

11. apríla 2011

Eulerovské sledy a eulerovské ťahy

Definícia

Hovoríme, že sled $s(u, v)$ v súvislom grafe $G = (V, H)$ je **eulerovský**,*ak obsahuje všetky hrany grafu G.*

Poznámka

Pretože ťah je špeciálnym prípadom sledu, je definíciou eulerovského sledu presne vymedzený pojem **eulerovský ťah** *ako taký ťah $t(u, v)$ v súvislom grafe G, ktorý obsahuje všetky hrany grafu G.*
Keďže ťah obsahuje každú hranu grafu G práve raz, postupnosť vrcholov a hrán ťahu $t(u, v)$ predstavuje postup, ako nakresliť diagram grafu G "jedným ťahom".

Definícia

Hovoríme, že graf $G = (V, H)$ je **eulerovský**, *ak v ňom existuje uzavretý eulerovský ťah.*

Eulerova veta o existencii eulerovského ťahu

Veta

(Euler, 1736.) *Súvislý graf* $G = (V, H)$ *je eulerovský práve vtedy, keď stupne všetkých vrcholov grafu* G *sú párne.*

DÔKAZ.

1. Ak v grafe G existuje uzavretý eulerovský ťah $\mathcal{T}$, potom stupeň každého vrchola je párny, pretože počet hrán ktorými ťah $\mathcal{T}$ z každého vrchola v vyšiel sa rovná počtu hrán, ktorými sme do vrchola v vošli.

2. Konštrukciu uzavretého eulerovského ťahu v súvislom grafe, ktorý má všetky vrcholy párneho stupňa popisuje nasledujúci algoritmus:

Algoritmus

- **Krok 1.** *Začni z ľubovoľného vrchola z, polož $\mathcal{T} = (z)$ a postupne predlžuj ťah $\mathcal{T}$ pokiaľ sa dá. Ukončíš vo vrchole z.*
- **Krok 2.** *Nájdi prvý vrchol v ťahu $\mathcal{T}$, ktorý má ešte aspoň jednu hranu nepoužitú*

  Ak taký vrchol v neexistuje, STOP.

  Ťah $\mathcal{T}$ je hľadaným uzavretým eulerovským ťahom.
  ak taky vrchol existuje, pokracuj krokom 3
- **Krok 3.** *Vytvor ťah $\mathcal{S}$ takto:*

  Polož $\mathcal{S} = (v)$ a postupne predlžuj ťah $\mathcal{S}$ doteraz nepoužitými hranami, pokiaľ sa dá. Ukončíš vo vrchole v.
- **Krok 4.** *Rozdeľ ťah $\mathcal{T}$ na z–v ťah $\mathcal{T}_1$ a v–z ťah $\mathcal{T}_2$, t. j. $\mathcal{T} = \mathcal{T}_1 \oplus \mathcal{T}_2$.*

  Polož $\mathcal{T} = \mathcal{T}_1 \oplus \mathcal{S} \oplus \mathcal{T}_2$.

  GOTO Krok 2.

♣

Algoritmus

Fleuryho algoritmus na hľadanie uzavretého eulerovského ťahu v súvislom grafe $G = (V, H)$, v ktorom majú všetky vrcholy párny stupeň.

- **Krok 1.** *Začni v ľubovoľnom vrchole a do ťahu $\mathcal{T}$ zaraď ľubovoľnú s ním incidentnú hranu.*
- **Krok 2.** *Ak sú do ťahu $\mathcal{T}$ zaradené všetky hrany grafu G, STOP.*
- **Krok 3.** *Ako ďalšiu hranu zaraď do ťahu $\mathcal{T}$ takú hranu incidentnú s jeho posledným vrcholom, po vybratí ktorej sa podgraf grafu G* [redacted] *nerozpadne na*
  - *dva netriviálne komponenty*
  - *netriviálny komponent a izolovaný začiatok ťahu $\mathcal{T}$.*
- *GOTO krok 2.*

♣

Algoritmus

Labyrintový algoritmus na hľadanie uzavretého eulerovského ťahu v súvislom grafe $G = (V, H)$, v ktorom majú všetky vrcholy párny stupeň.

- **Krok 1.** *Začni z ľubovoľného vrchola* $u \in V$.

  Nech sled $\mathcal{S}$ *inicializačne pozostáva z jediného vrchola* u.

  Polož $w := u$ *– vrchol* w *je posledný vrchol doteraz vytvoreného sledu* $\mathcal{S}$.

Algoritmus (– pokračovanie)

- **Krok 2.** *Ako ďalšiu hranu vyber podľa nižšie uvedených pravidiel do sledu $\mathcal{S}$ hranu $\{w, v\}$. Zaznač si smer použitia hrany $\{w, v\}$.*

  Ak doteraz vrchol v ešte nebol zaradený do sledu $\mathcal{S}$, označ hranu $\{w, v\}$ ako hranu prvého príchodu.

  Ďalej zaznamenaj tzv. **spätnú postupnosť** *— poradie hrán, v ktorom sa v slede $\mathcal{S}$ vyskytujú po druhýkrát.*

  Pri výbere hrany dodržuj nasledujúce pravidlá:

(L1): *Každú hranu možno v jednom smere použiť iba raz*
(L2): *Poradie zaraďovania hrán:*
– nepoužité hrany
– hrany použité raz
– hrana prvého príchodu (ak niet inej možnosti)

- **Krok 3.** *Ak taká hrana neexistuje – STOP. Spätná postupnosť určuje hľadaný eulerovský ťah.*

- **Krok 4.** *Inak polož $w := v$ a pokračuj krokom 2.*

Úloha čínskeho poštára

Chinese postman problem

Slovná formulácia úlohy čínskeho poštára:
Poštár má vyjsť z pošty, prejsť všetky ulice svojho rajónu a vrátiť sa na poštu tak, aby sa čo najmenej nachodil.

Matematická formulácia úlohy čínskeho poštára.
V súvislom hranovo ohodnotenom grafe nájsť uzavretý eulerovský sled najmenšej dĺžky.

Úloha čínskeho poštára

Poznámka

- *Model cestnej siete poštára – súvislý hranovo ohodnotený graf* $G = (V, H, c)$.
- *Keby mal graf* G *všetky vrcholy párneho stupňa, stačilo by nájsť v* G *uzavretý eulerovský ťah.*
- *Ak má graf* G *vrcholy nepárneho stupňa, je ich* $2t$ *(párny počet).*
- *Pridaním fiktívnych hrán typu* $\{\text{nepárny}, \text{nepárny}\}$ *s dĺžkou rovnajúcou sa vzdialenosti príslušných vrcholov v* G *možno z* G *vyrobiť eulerovský graf alebo multigraf.*
- *Uzavretý eulerovský ťah v rozšírenom grafe predstavuje trasu poštára, pričom fiktívne hrany predstavujú najkratšie cesty medzi ich koncovými vrcholmi a tieto cesty poštár prejde naprázdno – bez roznášania pošty.*
- **Čím meší súčet dĺžok pridaných fiktívnych hrán, tým lepšie výsledné riešenie.**

Úloha čínskeho poštára

Definícia

Nech $G = (V, H, c)$ je hranovo ohodnotený graf.
Párenie *v grafe G je taký jeho podgraf P, v ktorom má každý vrchol stupeň 1.*
Cena párenia *P je súčet ohodnotení jeho hrán.*
Hovoríme, že párenie P je **maximálne párenie** *v grafe G, ak P nie je podgrafom žiadneho iného párenia v G.*
Párenie P je **najpočetnejšie párenie** *v grafe G ak P má zo všetkých párení najväčší počet hrán.*
Párenie P je **úplné párenie** *v G, ak P je faktorovým podgrafom grafu G (P obsahuje všetky vrcholy grafu G).*

a) Maximálne párenie, ktoré nie je ani najpočetnejšie, ani úplné.

b) Najpočetnejšie párenie, ktoré nie je úplné.

c) Úplné párenie v K_6.

Algoritmus

Edmondsov algoritmus na hľadanie najkratšieho uzavretého eulerovského sledu v súvislom hranovo ohodnotenom grafe $G = (V, H, c)$.

- **Krok 1.** *V grafe G nájdi všetky vrcholy nepárneho stupňa. Tých je párny počet $2t$.*
  Z vrcholov nepárneho stupňa zostroj úplný graf K_{2t}. Jeho hrany ohodnoť [redacted] *v pôvodnom grafe G.*
- **Krok 2.** *V grafe K_{2t} nájdi úplné párenie s minimálnou cenou.*
- **Krok 3.** *Hrany párenia pridaj k hranovej množine pôvodného grafu G. Dostaneš tak multigraf $\overline{G}$, v ktorom majú všetky vrcholy párny stupeň. V multigrafe $\overline{G}$ zostroj uzavretý eulerovský ťah $\mathcal{T}$.*
- **Krok 4.** *Hrany párenia v ťahu $\mathcal{T}$ nahraď príslušnými najkratšími cestami v grafe G a označ ich ako prejdené naprázdno. Dostaneš tak najkratší eulerovský uzavretý sled v grafe G.*

♣

Hamiltonovský sled, hamiltonovský cyklus

Definícia

Sled v grafe G sa nazýva **hamiltonovský sled** *v grafe G, ak obsahuje všetky vrcholy grafu G.*

Poznámka

Predchádzajúca definícia definuje i hamiltonovskú cestu i hamiltonovský cyklus, pretože obe sú špeciálnym prípadom hamiltonovského sledu.

Definícia

Hovoríme, že graf G je **hamiltonovský**, *ak v ňom existuje hamiltonovský cyklus.*

Hamiltonovský sled, hamiltonovský cyklus

Neexistuje jednoduché kritérium na zixstenie toho, či je daný graf hamiltonovský.
Máme niekoľko hrubých postačujúcich podmienok:

Veta

Nech v grafe $G = (V, H)$ s aspoň troma vrcholmi pre každé dva také vrcholy vrcholy u, v, ktoré nie sú susedné, platí

$$\deg(u) + \deg(v) \geq |V|.$$

Potom je G hamiltonovský graf.

Veta

Nech v grafe $G = (V, H)$ s aspoň troma vrcholmi platí pre každý vrchol $v \in V$

$$\deg(v) \geq \frac{1}{2}.|V|.$$

Potom je G hamiltonovský graf.

Úloha obchodného cestujúceho

Travelling Salesman Problem - TSP

Úloha obchodného cestujúceho – TSP

Slovná formulácia úlohy obchodného cestujúceho:
Obchodný cestujúci má navštíviť všetkých svojich zákazníkov a vrátiť sa domov tak, aby sa čo najmenej nachodil.

Matematická formulácia úlohy obchodného cestujúceho:
Ak dovoľujeme navštíviť to isté miesto viackrát, úlohu obchodného cestujúceho môžeme formulovať nasledovne:

V súvislom hranovo ohodnotenom grafe nájsť najkratší uzavretý hamiltonovský sled.

Ak zakazujeme navštíviť to isté miesto viackrát, úlohu obchodného cestujúceho formulujeme takto:

V súvislom hranovo ohodnotenom grafe nájsť najkratší hamiltonovský cyklus.

Úloha obchodného cestujúceho – TSP

Najkratší hamilt. cyklus v úplnom grafe s $\triangle$ nerovnosťou

V úplnom grafe $\overline{G}$ už každá permutácia vrcholov definuje hamiltonovský cyklus.

Ak fixujeme prvý vrchol, potom máme $(n-1)!$ rôznych hamiltonovských cyklov.

Pre exaktné hľadanie najkratšieho hamiltonovského cyklu niet podstatne lepšieho algoritmu, ako systematické prehľadanie všetkých $(n-1)!$ permutácií.

Doba výpočtu pri prekontrolovaní 10^9 permutácií/sec.

n	$(n-1)!$	sekundy	minúty	dni	roky
10	3,6E+05	0,36 ms	-	-	-
15	8,7E+10	87,17	1,45	-	-
20	1,2E+17	1,2E+08	2000000	1400	3,9
25	6,2E+23	6,2E+14	1,0E+13	7,2E+09	2,0E+07
30	8,8E+30	8,8E+21	1,5E+20	1,0E+17	2,8E+14
35	3,0E+38	3,0E+29	4,9E+27	3,4E+24	9,4E+21
40	2,0E+46	2,0E+37	3,4E+35	2,4E+32	6,5E+29
45	2,7E+54	2,7E+45	4,4E+43	3,1E+40	8,4E+37
50	6,1E+62	6,1E+53	1,0E+52	7,0E+48	1,9E+46

Doba od Veľkého Tresku $1,4 * 10^{10}$ rokov.

Najkratší hamilt. cyklus v úplnom grafe s $\triangle$ nerovnosťou

Dôsledok: Nutnosť používať algoritmy, ktoré dávajú dostatočne dobré, ale nie zaručene optimálne riešenie – suboptimálne algoritmy, heuristiky.

Algoritmus

Pažravá metóda – Greedy Algorithm. Heuristika na hľadanie suboptimálneho riešenia úlohy obchodného cestujúceho v úplnom grafe $G = (V, H, c)$ s aspoň tromi vrcholmi a s trojuholníkovou nerovnosťou.

- **Krok 1.** *Začni v ľubovoľnom vrchole a do (budúceho) hamiltonovského cyklu vlož najlacnejšiu hranu incidentnú s týmto vrcholom.*
- **Krok 2.** *Ak je vybratých $n-1$ hrán, uzavri cyklus. STOP*
- **Krok 3.** *Inak vyber takú najlacnejšiu nevybranú hranu incidentnú s posledným vrcholom doteraz vybranej postupnosti, ktorá nie je incidentná so žiadnym iným vrcholom vybranej postupnosti.*

  GOTO Krok 2.

♣

Metóda zdvojenia kostry.

Algoritmus

Metóda zdvojenia kostry. (Kim – 1975). Heuristika na hľadanie suboptimálneho riešenia úlohy obchodného cestujúceho v úplnom grafe $G = (V, H, c)$ s trojuholníkovou nerovnosťou.

- **Krok 1.** *V grafe G zostroj najlacnejšiu kostru K.*
- **Krok 2.** *V kostre K zostroj uzavretý sled $\mathcal{S}$, ktorý obsahuje každú hranu práve dvakrát. (Použi napr. Tarryho algoritmus).*
- **Krok 3.** *Z uzavretého sledu $\mathcal{S}$ vytvor hamiltonovský cyklus takto: Postupne prechádzaj sledom $\mathcal{S}$ a keď narazíš na taký vrchol, alebo úsek niekoľkých vrcholov za sebou, ktoré sa už v slede vyskytujú, premosti takýto vrchol alebo úsek priamou hranou.*

♣

Metóda zdvojenia kostry

Veta

Nech $G = (V, H, c)$ je úplný graf, v ktorom platí trojuholníková nerovnosť. Nech $c(MZK)$ je dĺžka hamiltonovského cyklu získaného metódou zdvojenia kostry, nech $c(OPT)$ je dĺžka najkratšieho hamiltonovského cyklu v grafe G. Potom

$$\frac{c(MZK)}{c(OPT)} < 2.$$

Naviac posledný odhad už nemožno zlepšiť – pre každé $\varepsilon > 0$ existuje taký graf G_ε, že preň je $c(MZK)/c(OPT) > 2 - \varepsilon$.

Algoritmus kostry a párenia.

Algoritmus

Algoritmus kostry a párenia. (Christofides – 1976.) *Heuristika na hľadanie suboptimálneho riešenia úlohy obchodného cestujúceho v úplnom grafe* $G = (V, H, c)$ *s trojuholníkovou nerovnosťou.*

- **Krok 1.** *V grafe G zostroj najlacnejšiu kostru K.*
- **Krok 2.** *V kostre K nájdi všetky vrcholy nepárneho stupňa. Tých je $2t$.*
- **Krok 3.** *Z vrcholov nepárneho stupňa zostroj úplný graf K_{2t}, jeho hrany ohodnoť ohodnoteniami príslušných hrán v pôvodnom grafe G.*
- **Krok 4.** *V grafe K_{2t} nájdi úplné párenie s minimálnou cenou.*
- **Krok 5.** *Hrany párenia dodaj k hranovej množine najlacnejšej kostry K. Dostaneš tak graf (multigraf) $\overline{G}$, ktorý má všetky vrcholy párneho stupňa.*
- **Krok 6.** *V grafe (resp. multigrafe) $\overline{G}$ zostroj uzavretý eulerovský ťah $\mathcal{T}$.*
- **Krok 7.** *Z uzavretého ťahu $\mathcal{T}$ vytvor hamiltonovský cyklus takto: Postupne prechádzaj ťahom $\mathcal{T}$ a keď narazíš na taký vrchol, alebo úsek niekoľkých vrcholov za sebou, ktoré sa už v slede vyskytujú, premosti takýto vrchol alebo úsek priamou hranou.*

♣

Algoritmus kostry a párenia.

Veta

Nech $G = (V, H, c)$ je úplný graf, v ktorom platí trojuholníková nerovnosť. Nech $c(MKP)$ je dĺžka hamiltonovského cyklu získaného metódou kostry a párenia, nech $c(OPT)$ je dĺžka najkratšieho hamiltonovského cyklu v grafe G. Potom

$$\frac{c(MKP)}{c(OPT)} < \frac{3}{2}.$$

Naviac posledný odhad už nemožno zlepšiť – pre každé $\varepsilon > 0$ existuje taký graf G_ε, že preň je $c(MKP)/c(OPT) > 3/2 - \varepsilon$.

Poznámka

Nie je známy polynomiálny algoritmus ALG, pre ktorý by bol zaručený lepší pomer $c(ALG)/c(OPT)$ než $3/2$.

Vkladacia heuristika pre TSP

Algoritmus

Vkladacia heuristika na konštrukciu suboptimálneho hamiltonovského cyklu v úplnom grafe $G = (V, H, c)$ s trojuholníkovou nerovnosťou.

- **Krok 1.** *Do cyklu zaraď hranu $h = \{u, v\}$ s najmenšou cenou. Nájdi vrchol $w \in V$, pre ktorý je súčet $c\{u, w\} + c\{w, v\}$ najmenší. Vytvor cyklus $C = (u, \{u, w\}, w, \{w, v\}, v, \{v, u\}, u)$.*

- **Krok 2.** *Ak cyklus C obsahuje všetky vrcholy grafu G, STOP. Inak pokračuj krokom 3.*

- **Krok 3.** *Pre každú hranu $h = \{u, v\}$ cyklu C vypočítaj*

$$z(h) = \min\{c\{u, w\} + c\{w, v\} - c\{u, v\} \mid w \in V - C\}.$$

Vezmi hranu $h = \{u, v\}$ s minimálnym $z(h)$ a w vrchol, pre ktorý nastalo minimum. Vytvor cyklus C' tak, že nahradíš hranu $\{u, v\}$ dvojicou hrán $\{u, w\}$, $\{w, v\}$. Polož $C := C'$.

GOTO Krok 2.

♣

Algoritmus

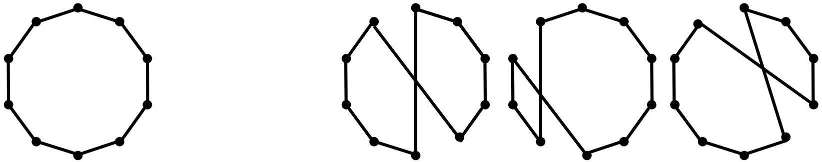
Algoritmus prehľadávania okolí.

- **Krok 1.** *Za počiatočný hamiltonovský cyklus C vezmi ľubovoľný hamiltonovský cyklus (dostať ho môžeš náhodným generátorom alebo ako výsledok niektorej vytvárajúcej heuristiky).*
- **Krok 2.** *Hľadaj $C' \in \mathcal{O}(C)$ také, že $c(C') < c(C)$.*

  Ak pre všetky $C' \in \mathcal{O}(C)$ $c(C') \geq c(C)$, STOP, C je suboptimálny hamiltonovský cyklus.

  Inak pokračuj krokom 3.
- **Krok 3.** *Vezmi $C' \in \mathcal{O}(C)$ také, že $c(C') < c(C)$ a polož $C := C'$. Goto Krok 2.*

♣



Cyklus C a niekoľko prvkov jeho okolia.

Nebezpečenstvo algoritmu prehľadávania okolí:

Algoritmus uviazne v takom zlom riešení, v okolí ktorého niet lepšieho riešenia.

Riešenie:

Viacnásobné spustenie algoritmu s rôznymi štartovacími riešeniami.