

# Teorie grafů

16. a 17. přednáška z LGR

# Obsah

- 1 **Stromy**
  - Charakterizace stromů
  - Počet stromů
  
- 2 **Minimální kostra grafu**
  - Algoritmus na minimální kostru
  - Borůvkův - Kruskalův algoritmu
  - Jarníkův - Primův algoritmus

# Stromy

## Definice

**Strom** je souvislý graf neobsahující kružnici (jako svůj podgraf).

## Poznámka

Každý strom je obyčejným grafem, neboť smyčky či paralelní hrany by znamenaly přítomnost kružnice.

# Stromy

## Tvrzení

Každý strom s  $n \geq 2$  vrcholy má alespoň dva vrcholy stupně jedna (tzv. listy).

## Tvrzení

Každý graf bez kružnic s  $n \geq 2$  vrcholy a  $m \geq 1$  hranami má alespoň dva vrcholy stupně jedna.

## Věta (Eulerův vzorec)

Každý strom  $T = (V, E)$  o  $n$  vrcholech má  $n - 1$  hran, tj.  $|E| = |V| - 1$ .

# Stromy

## Věta

Nechť  $G$  je graf o  $n$  vrcholech. Následující tvrzení jsou ekvivalentní:

- 1  $G$  je strom.
- 2  $G$  je souvislý a má  $n - 1$  hran.
- 3  $G$  je bez kružnic a má  $n - 1$  hran.

## Tvrzení

Má-li graf  $n \geq 2$  vrcholů a  $n - 1$  hran a nemá-li izolované vrcholy (tj. nemá vrchol stupně nula), pak v něm existují alespoň dva vrcholy stupně jedna.

# Stromy

## Věta

Nechť  $G$  je graf. Následující tvrzení jsou ekvivalentní:

- 1  $G$  je strom.
- 2  $G$  je souvislý a odebráním libovolné hrany přestane být souvislý ( $G$  je minimální souvislý).
- 3  $G$  je bez kružnic a přidáme-li ke grafu libovolnou hranu, uzavřeme přesně jednu kružnici ( $G$  je maximální bez kružnic).
- 4 Mezi každými dvěma jeho vrcholy vede právě jedna cesta.

Poznámka: Hrana v grafu  $G$  se nazývá *most*, pokud její odebrání zvětší počet komponent souvislosti grafu. Tvrzení 2 říká, že ve stromu je každá hrana mostem.

# Stromy

## Tvrzení

Počet stromů na  $n$  vrcholech je  $n^{n-2}$ .

## Tvrzení

Nechť  $d_1 \geq d_2 \geq \dots \geq d_n > 0$ . Pak existuje strom se skóre  $(d_1, d_2, \dots, d_n)$ , právě když  $\sum_{i=1}^n d_i = 2(n-1)$ .

Důkazy těchto tvrzení používají Prüferovy kódy a zájemci je mohou nalézt v knize: Matoušek, Nešetřil: Kapitoly z diskretní matematiky.

# Stromy

Je aspoň tolik neizomorfních stromů na  $n$  vrcholech, kolik je možných stromových skóre, tj. nerostoucích  $n$ -tic  $(d_1, d_2, \dots, d_n)$  kladných čísel, v nichž  $\sum_{i=1}^n d_i = 2(n-1)$ .  
(Podmínka  $d_n = d_{n-1} = 1$  z předchozího plyne automaticky).  
Přitom může existovat více neizomorfních stromů se stejným skóre.

Označme počet neizomorfních stromů na  $n$  vrcholech jako  $t_n$ .  
Počet neizomorfních stromů rychle roste:

$n$	3	4	5	6	...	10	...	20
$t_n$	1	2	3	6	...	106	...	832 065



# Stromy

## Definice

Graf, který neobsahuje kružnice, se nazývá *les*.

Poznámka: Komponenty souvislosti lesa jsou tedy stromy.

## Tvrzení

Les o  $n$  vrcholech a  $k$  komponentách souvislosti má  $n - k$  hran.

# Minimální kostra grafu

## Problém minimální kostry

Otakar Borůvka řešil v souvislosti s elektrifikací Jižní Moravy tzv. problém minimální kostry grafu. Cílem bylo propojit vesnice a města tak, aby náklady na elektrické vedení byly co nejmenší. Borůvka publikoval v roce 1926 dva algoritmy na hledání minimální kostry, z nichž představíme jen jeden. Dále uvedeme algoritmus od Vojtěcha Jarníka z roku 1930.

Algoritmy jsou známé pod jmény amerických matematiků jako Kruskalův algoritmus (1956) a Primův algoritmus (1957).

# Minimální kostra grafu

## Definice

Nechť  $G$  je souvislý graf. Faktor grafu  $G$ , který je stromem, se nazývá *kostra grafu*.

## Tvrzení

Každý souvislý graf má aspoň jednu kostru.

V angličtině se kostra grafu nazývá "spanning tree" = *vepsaný strom*. Analogicky, "spanning forest" = *vepsaný les* je faktor (obecně nesouvislého) grafu  $G$ , který je lesem.

# Minimální kostra grafu

## Definice

Nechť  $G = (V, E)$  je souvislý graf, který má ohodnocené hrany, tj. je dáno zobrazení  $c: E \rightarrow \mathbb{R}^+ : c(e) = \text{cena hrany } e$ .

**Minimální kostra** grafu  $G$  je taková kostra  $K = (V, L)$ , která má nejmenší součet cen hran  $\sum_{e \in L} c(e)$  mezi všemi kostrami grafu  $G$ .

## Tvrzení

Každý souvislý ohodnocený graf má minimální kostru (nemusí však být jediná).

# Minimální kostra grafu

## Obecný algoritmus na minimální kostru

Vstup: Souvislý graf  $G = (V, E)$  s ohodnocením hran  $c: E \rightarrow \mathbb{R}^+$

Výstup: Hrany minimální kostry  $K = (V, L)$

Myšlenka algoritmu: Začne s jednovrcholovými komponentami budoucí kostry  $K$  a postupně přidává hrany tak, že vždy spojí nějaké dvě komponenty takovou hranou, která je (aspoň) pro jednu z nich tou nejlevnější hranou, která z ní trčí ven.

# Minimální kostra grafu

## Obecný algoritmus

(inicializace)

- $\mathcal{S} \leftarrow \{\{v\}, v \in V\}$  množina komponent souvislosti podgrafu  $K$
- $L \leftarrow \emptyset$

(přidávání hran)

- while  $|\mathcal{S}| > 1$  do
  - vyber hranu  $e \in E$ , která spojuje dvě různé komponenty  $S, S' \in \mathcal{S}$  a aspoň pro jednu z nich je tou nejlevnější hranou, která z ní trčí
  - $L \leftarrow L \cup \{e\}$
  - do  $\mathcal{S}$  dej  $S \cup S'$  místo  $S$  a  $S'$  (spoj komponenty  $S$  a  $S'$ )
  - enddo
- output  $L$

# Minimální kostra grafu

## Tvrzení

Nechť  $G = (V, E)$  je ohodnocený souvislý graf a  $K = (V, L)$  je jeho kostra.  $K$  je minimální kostra, právě když pro každou hranu  $e \in E \setminus L$  platí:  $e$  je tou nejdražší hranou na kružnici, která vznikne, když hranu  $e$  přidáme ke kostře  $K$ .

## Tvrzení

Nechť  $G = (V, E)$  je ohodnocený souvislý graf a  $K = (V, L)$  je jeho kostra.  $K$  je minimální kostra, právě když pro každou hranu  $e \in L$  platí:  $e$  je tou nejlevnější hranou mezi množinami vrcholů dvou komponent souvislosti, na něž se kostra rozpadne, když hranu  $e$  z kostry vyhodíme.

# Minimální kostra grafu

## Obecný algoritmus - korektnost

- Terminace - invariant =  $|\mathcal{S}|$  = počet komponent souvislosti podgrafu  $K$ .  
(Ten po každém opakování cyklu klesne o jedna, lze použít též for-cyklus "for  $i \leftarrow 1$  to  $n - 1$  do ...", kde  $n = |V|$ .)
- Parciální korektnost - invariant = "Existuje minimální kostra  $K'$  taková, že  $L \subseteq E(K')$ ."  
(Lze dokázat indukcí z předchozích tvrzení.)  
Jelikož algoritmus zastaví, když  $|L| = n - 1 = |E(K')|$ , tak  $K = K'$  je minimální kostra.



# Minimální kostra grafu

Ukážeme nyní dva speciální případy obecného algoritmu.

## Borůvkův - Kruskalův algoritmus

Vstup: Ohodnocený (ne nutně souvislý) graf  $G = (V, E)$

Výstup: Hrany minimální kostry  $K = (V, L)$

Myšlenka algoritmu: Bere hrany do kostry od nejlevnějších, pokud nevytvoří kružnici (tj. pokud oba vrcholy hrany neleží ve stejné komponentě souvislosti). Jedná se tedy o "hladový algoritmus".

Pokud přidáváme celkově nejlevnější mezikomponentovou hranu, pak je to jistě (dokonce pro obě komponenty) ta nejlevnější hrana trčící z komponenty ven, jde o speciální případ obecného algoritmu.

# Minimální kostra grafu

## Borůvkův - Kruskalův algoritmus

(inicializace)

- $\mathcal{S} \leftarrow \{\{v\}, v \in V\}$
- $L \leftarrow \emptyset, i \leftarrow 1$

(přidávání hran)

- seřad' hrany podle ceny:  $e_1, \dots, e_m$ , kde  $c(e_i) \leq c(e_j)$  pro  $i < j$
- while  $|\mathcal{S}| > 1$  and  $i \leq m$  do
  - if  $e_i = \{u, v\}$  má vrcholy v různých komponentách, tj.  $u \in S, v \in S', S \neq S'$ , then
    - $L \leftarrow L \cup \{e_i\}$
    - do  $\mathcal{S}$  dej  $S \cup S'$  místo  $S$  a  $S'$     endif
  - $i \leftarrow i + 1$     enddo
- output  $L$

# Minimální kostra grafu

## Poznámka

Algoritmus je napsán tak, aby zastavil, i když na vstupu bude nesouvislý ohodnocený graf. V tom případě zastaví po probrání všech hran a najde minimální vepsaný les.

## Poznámka

Algoritmus má ještě daleko do konkrétní implementace. Otázkou je, jak bude datově representován soubor množin  $\mathcal{S}$  (komponent souvislosti). Až pak bude možné odhadnout čas potřebný pro

- vyhledání, do které komponenty patří vrchol  $v$  -  $FIND(v)$ ,
- sjednocení dvou komponent -  $UNION(S, S')$ .

# Minimální kostra grafu

## Borůvkův - Kruskalův algoritmus - časová náročnost

Komponenty souvislosti podgrafu  $K$  jsou zadané svými vrcholy, množina komponent  $\mathcal{S}$  tvoří rozklad na množině  $V$  (= systém po dvou disjunktních množin, jejichž sjednocením je celá množina  $V$ ).

Možná datová struktura:  $\mathcal{S}$  může být pole délky  $n = |V|$ , kde  $\mathcal{S}(v)$  je jméno komponenty obsahující vrchol  $v$ .

$FIND(v)$  trvá čas  $O(1)$ ,  $UNION(S, S')$  čas  $O(\min\{|S|, |S'|\})$ .

Celkový čas pro přidávání hran do kostry je  $O(m + n \log(n))$ .

Nejdelší fází je tedy setřídění hran dle cen, které vyžaduje čas  $O(m \log(m)) = O(m \log(n))$ .

# Minimální kostra grafu

## Jarníkův - Primův algoritmus

Vstup: Ohodnocený (ne nutně souvislý) graf  $G = (V, E)$

Výstup: Hrany minimální kostry  $K = (V, L)$

Myšlenka algoritmu: Zvětšuje stále první komponentu souvislosti o nejlevnější hranu, která z ní trčí, resp. o druhý vrchol té hrany. Stačí tedy pamatovat si pouze vrcholy první komponenty.

# Minimální kostra grafu

## Jarníkův - Primův algoritmus

(inicializace)

- vyber vrchol  $v \in V$ ,  $S \leftarrow \{v\}$
- $L \leftarrow \emptyset$

(přidávání hran)

- while  $S \neq V$  and trčí nějaká hrana z  $S$  do
  - vyber nejlevnější hranu  $e = \{u, w\}$  takovou, že  $u \in S$ ,  $w \notin S'$
  - $L \leftarrow L \cup \{e\}$
  - $S \leftarrow S \cup \{w\}$     enddo
- output  $L$

# Minimální kostra grafu

## Poznámka

Algoritmus je opět napsán tak, aby zastavil, i když na vstupu bude nesouvislý ohodnocený graf. V tom případě najde minimální kostru první komponenty souvislosti (té, v níž je vybráný vrchol  $v$ ).

## Jarníkův - Primův algoritmus - časová náročnost

Pokud si budeme držet v paměti pole těch nejlevnějších hran z komponenty  $S$  do vrcholů mimo  $S$ , bude časová náročnost  $O(n^2)$ .

# Minimální kostra grafu

## Příklad

Ohodnocený graf  $G$  s vrcholy  $V = \{1, 2, 3, 4, 5\}$  má hrany zadané maticí cen:

$$\begin{pmatrix} - & 3 & 1 & 2 & - \\ 3 & - & 3 & 6 & 5 \\ 1 & 3 & - & 1 & 4 \\ 2 & 6 & 1 & - & - \\ - & 5 & 4 & - & - \end{pmatrix}$$

Jarníkův - Primův algoritmus nalezne minimální kostru  $K = (V, L)$  s hranami  $\{1, 2\}, \{1, 3\}, \{3, 4\}, \{3, 5\}$  o celkové ceně  $c(K) = 9$ .



# Minimální kostra grafu

## Příklad

Borůvkův - Kruskalův algoritmus, při tomto setřídění hran dle ceny:  $\{1, 3\}$ ,  $\{3, 4\}$ ,  $\{1, 4\}$ ,  $\{2, 3\}$ ,  $\{1, 2\}$ ,  $\{3, 5\}$ ,  $\{2, 5\}$ ,  $\{2, 4\}$ , nalezne minimální kostru  $K' = (V, L')$  s hranami  $\{1, 3\}$ ,  $\{2, 3\}$ ,  $\{3, 4\}$ ,  $\{3, 5\}$  o celkové ceně  $c(K') = 9$ .

Náš graf  $G$  má tedy aspoň dvě minimální kostry.

# Stromy a minimální kostry

## Literatura

- J. Demel: Grafy a jejich aplikace, Academia, 2015.
- J. Matoušek, J. Nešetřil: Kapitoly z diskrétní matematiky, Nakladatelství Karolinum, 2000.
- M. Dostál: Cvičení k přednášce LGR (najdete v nich důkazy některých tvrzení z přednášky a mnoho dalších příkladů).