

Matematika III – 11. přednáška

Stromy a kostry, rovinné grafy

Michal Bulant

Masarykova univerzita
Fakulta informatiky

28. 11. 2012

Obsah přednášky

Doporučené zdroje

- Martin Panák, Jan Slovák, Drsná matematika, e-text.
- *Předmětové záložky v IS MU*

- Martin Panák, Jan Slovák, Drsná matematika, e-text.
- *Předmětové záložky v IS MU*
- Jiří Matoušek, Jaroslav Nešetřil, Kapitoly z diskrétní matematiky, Univerzita Karlova v Praze, Karolinum, Praha, 2000, 377 s.
- Petr Hliněný, Teorie grafů, studijní materiály, <https://is.muni.cz/auth/el/1433/podzim2012/MA010/index.qwarp>

- Martin Panák, Jan Slovák, Drsná matematika, e-text.
- *Předmětové záložky v IS MU*
- Jiří Matoušek, Jaroslav Nešetřil, Kapitoly z diskrétní matematiky, Univerzita Karlova v Praze, Karolinum, Praha, 2000, 377 s.
- Petr Hliněný, Teorie grafů, studijní materiály, <https://is.muni.cz/auth/el/1433/podzim2012/MA010/index.qwarp>
- Donald E. Knuth, The Stanford GraphBase, ACM, New York, 1993
(<http://www-cs-faculty.stanford.edu/~knuth/sgb.html>).

- Martin Panák, Jan Slovák, Drsná matematika, e-text.
- *Předmětové záložky v IS MU*
- Jiří Matoušek, Jaroslav Nešetřil, Kapitoly z diskrétní matematiky, Univerzita Karlova v Praze, Karolinum, Praha, 2000, 377 s.
- Petr Hliněný, Teorie grafů, studijní materiály, <https://is.muni.cz/auth/el/1433/podzim2012/MA010/index.qwarp>
- Donald E. Knuth, The Stanford GraphBase, ACM, New York, 1993
(<http://www-cs-faculty.stanford.edu/~knuth/sgb.html>).

Plán přednášky

Stromy jako datové struktury

Stromy se často používají jako (acyklické) datové struktury, v praxi stromy procházíme v určitém pořadí vrcholů – na rozdíl od grafů k jednoznačnému určení nějakého uspořádání vrcholů stačí vybrat jeden vrchol – *kořen* (root) v_r .

Ve stromu není žádná kružnice, proto volba jednoho vrcholu v_r zadává orientaci všech hran.

Stromy jako datové struktury

Stromy se často používají jako (acyklické) datové struktury, v praxi stromy procházíme v určitém pořadí vrcholů – na rozdíl od grafů k jednoznačnému určení nějakého uspořádání vrcholů stačí vybrat jeden vrchol – *kořen* (root) v_r .

Ve stromu není žádná kružnice, proto volba jednoho vrcholu v_r zadává orientaci všech hran.

Po výběru **kořenu** se začíná graf více podobat skutečnému stromu v přírodě. Stromy s jedním vybraným kořenem nazýváme **kořenové stromy**.

Stromy jako datové struktury

Stromy se často používají jako (acyklické) datové struktury, v praxi stromy procházíme v určitém pořadí vrcholů – na rozdíl od grafů k jednoznačnému určení nějakého uspořádání vrcholů stačí vybrat jeden vrchol – *kořen* (root) v_r .

Ve stromu není žádná kružnice, proto volba jednoho vrcholu v_r zadává orientaci všech hran.

Po výběru **kořenu** se začíná graf více podobat skutečnému stromu v přírodě. Stromy s jedním vybraným kořenem nazýváme **kořenové stromy**.

Definice

V kořenovém stromu $T = (V, E)$ je vrchol w **následník vrcholu** v a naopak v je předchůdce vrcholu w právě tehdy, když existuje cesta z kořene stromu do w která prochází v a $v \neq w$. **Přímý následník** a **přímý předchůdce** vrcholu jsou pak následníci a předchůdci přímo spojení hranou. Mluvíme také o **synech** a **otcích** (patrně v narážce na genealogické stromy).

Definice

Binární stromy jsou speciálním případem kořenového stromu, kdy každý otec má nejvýše dva následníky (někdy se ale pod stejným označením binární strom předpokládá, že všechny vrcholy kromě listů mají právě dva následníky).

Definice

Binární stromy jsou speciálním případem kořenového stromu, kdy každý otec má nejvýše dva následníky (někdy se ale pod stejným označením binární strom předpokládá, že všechny vrcholy kromě listů mají právě dva následníky).

Často jsou vrcholy stromu spojeny s klíči v nějaké úplně uspořádané množině (např. reálná čísla) a slouží k hledání vrcholu s daným klíčem.

Je realizováno jako hledání cesty od kořene stromu a v každém vrcholu se podle velikosti rozhodujeme, do kterého ze synů budeme pokračovat (resp. zastavíme hledání, pokud jsme již ve hledaném vrcholu). Abychom mohli tuto cestu jednoznačně krok po kroku určovat, požadujeme aby jeden syn společně se všemi jeho následníky měli menší klíče než druhý syn a všichni jeho následníci.

Pro efektivní vyhledávání užíváme **vyvážené binární stromy**, ve kterých se délky cest z kořene do listů liší maximálně o jedničku.

Pro efektivní vyhledávání užíváme **vyvážené binární stromy**, ve kterých se délky cest z kořene do listů liší maximálně o jedničku. Nejdále od vyváženého stromu na n vrcholech je tedy cesta P_n (která formálně může být považována za binární strom), zatímco dokonale vyvážený strom, kde kromě listů má každý otec právě dva syny, je možné sestavit pouze pro hodnoty $n = 2^k - 1$, $k = 1, 2, \dots$

Pro efektivní vyhledávání užíváme **vyvážené binární stromy**, ve kterých se délky cest z kořene do listů liší maximálně o jedničku. Nejdále od vyváženého stromu na n vrcholech je tedy cesta P_n (která formálně může být považována za binární strom), zatímco dokonale vyvážený strom, kde kromě listů má každý otec právě dva syny, je možné sestavit pouze pro hodnoty $n = 2^k - 1$, $k = 1, 2, \dots$

Ve vyvážených stromech dohledání vrcholu podle klíče bude vždy vyžadovat pouze $O(\log_2 n)$ kroků. Hovoříme v této souvislosti také často o **binárních vyhledávacích stromech**.

Pro efektivní vyhledávání užíváme **vyvážené binární stromy**, ve kterých se délky cest z kořene do listů liší maximálně o jedničku. Nejdále od vyváženého stromu na n vrcholech je tedy cesta P_n (která formálně může být považována za binární strom), zatímco dokonale vyvážený strom, kde kromě listů má každý otec právě dva syny, je možné sestavit pouze pro hodnoty $n = 2^k - 1$, $k = 1, 2, \dots$

Ve vyvážených stromech dohledání vrcholu podle klíče bude vždy vyžadovat pouze $O(\log_2 n)$ kroků. Hovoříme v této souvislosti také často o **binárních vyhledávacích stromech**.

Jako cvičení si rozvažte, jak lze účinně vykonávat základní operace s grafy (přidávání a odebírání vrcholů se zadanými klíči, včetně vyvážení) nad binárními vyhledávacími stromy.

Izomorfismus stromů

Již dříve jsme si říkali, že rozhodnout o izomorfismu dvou obecných grafů je velmi obtížný problém. U stromů je naštěstí, jak si ukážeme, situace podstatně jednodušší.

Izomorfismus stromů

Již dříve jsme si říkali, že rozhodnout o izomorfismu dvou obecných grafů je velmi obtížný problém. U stromů je naštěstí, jak si ukážeme, situace podstatně jednodušší. Pro popis všech možných izomorfismů (kořenových) stromů je užitečné kromě vztahů otec–syn ještě užitečné mít syny uspořádaný v pořadí (třeba v představě odleva doprava nebo podle postupného růstu atd.).

Izomorfismus stromů

Již dříve jsme si říkali, že rozhodnout o izomorfismu dvou obecných grafů je velmi obtížný problém. U stromů je naštěstí, jak si ukážeme, situace podstatně jednodušší. Pro popis všech možných izomorfismů (kořenových) stromů je užitečné kromě vztahů otec–syn ještě užitečné mít syny uspořádaný v pořadí (třeba v představě odleva doprava nebo podle postupného růstu atd.).

Definice

Pěstěný strom $T = (V, E, v_r, \nu)$ je kořenový strom společně s částečným uspořádáním ν na hranách takovým, že srovnatelné jsou vždy právě hrany směřující od jednoho otce k synům.

Izomorfismem kořenových stromů $T = (V, E, v_r)$ a $T' = (V', E', v'_r)$ rozumíme takový izomorfismus grafů $\varphi : T \rightarrow T'$, který převádí v_r na v'_r .

Pro pěstěné stromy navíc požadujeme aby zobrazení hran zachovávalo částečná uspořádání ν a ν' .

Pro pěstěné stromy $T = (V, E, v_r, \nu)$ zavedeme jejich (jak uvidíme) jednoznačný popis pomocí slov z nul a jedniček. Obrazně si můžeme představit, že strom kreslíme a každý přírůstek naznačíme dvěma tahy, které si označíme 0 (dolů) a 1 (nahoru). Začneme od listů (příp. mimo kořene), kterým všem přiřadíme slovo 01. Celý strom pak budeme popisovat zřetězováním částí slov tak, že má-li otec v syny uspořádány jako posloupnost v_1, \dots, v_ℓ , a jsou-li již jednotliví synové označeni slovy W_1, \dots, W_ℓ , pak pro otce použijeme slovo

$$0W_1 \dots W_\ell 1.$$

Hovoříme o **kódu pěstěného stromu**.

Pro pěstěné stromy $T = (V, E, v_r, \nu)$ zavedeme jejich (jak uvidíme) jednoznačný popis pomocí slov z nul a jedniček. Obrazně si můžeme představit, že strom kreslíme a každý přírůstek naznačíme dvěma tahy, které si označíme 0 (dolů) a 1 (nahoru). Začneme od listů (příp. mimo kořene), kterým všem přiřadíme slovo 01. Celý strom pak budeme popisovat zřetězováním částí slov tak, že má-li otec v syny uspořádány jako posloupnost v_1, \dots, v_ℓ , a jsou-li již jednotliví synové označeni slovy W_1, \dots, W_ℓ , pak pro otce použijeme slovo

$$0W_1 \dots W_\ell 1.$$

Hovoříme o **kódu pěstěného stromu**.

Skutečně, kreslením cest dolů a nahoru získáme skutečně původní pěstěný strom s jednou hranou směřující shora do kořene navíc.

Věta

Dva pěstěné stromy jsou izomorfní právě, když mají stejný kód.

Důkaz.

Z konstrukce je zřejmé, že izomorfní stromy budou mít stejný kód, zbývá tedy pouze dokázat, že neizomorfní stromy vedou na různé kódy, jinými slovy, že z daného kódu jednoznačně zrekonstruujeme výchozí pěstovaný strom.

Dokážeme to indukcí podle délky kódu (tj. počtu nul a jedniček) tak, že využijeme jednoznačné kódy pro všechny podstromy vzniklé odejmutím kořene.

Věta

Dva pěstěné stromy jsou izomorfní právě, když mají stejný kód.

Důkaz.

Z konstrukce je zřejmé, že izomorfní stromy budou mít stejný kód, zbývá tedy pouze dokázat, že neizomorfní stromy vedou na různé kódy, jinými slovy, že z daného kódu jednoznačně zrekonstruujeme výchozí pěstovaný strom.

Dokážeme to indukcí podle délky kódu (tj. počtu nul a jedniček) tak, že využijeme jednoznačné kódy pro všechny podstromy vzniklé odejmutím kořene.

Pro nejkratší kód 01 je situace snadná.

Věta

Dva pěstěné stromy jsou izomorfní právě, když mají stejný kód.

Důkaz.

Z konstrukce je zřejmé, že izomorfní stromy budou mít stejný kód, zbývá tedy pouze dokázat, že neizomorfní stromy vedou na různé kódy, jinými slovy, že z daného kódu jednoznačně zrekonstruujeme výchozí pěstovaný strom.

Dokážeme to indukcí podle délky kódu (tj. počtu nul a jedniček) tak, že využijeme jednoznačné kódy pro všechny podstromy vzniklé odejmutím kořene.

Pro nejkratší kód 01 je situace snadná. V indukčním kroku je dán kód délky $2(n+1)$ tvaru $0A1$, přičemž $A = A_1A_2 \dots A_t$ je zřetězením několika kódů pěstovaných stromů. Část A_1 je tvořena nejkratším prefixem A , který má stejný počet 0 a 1, dále A_2 , atd. Podle indukčního předpokladu každé A_i jednoznačně odpovídá pěstěnému stromu, z čehož zřejmě dostáváme jediný pěstěný strom odpovídající kódu $0A1$. □

Nyní převedeme testování izomorfismu kořenových stromů, resp. stromů na testování pěstěných stromů.

Nyní převedeme testování izomorfismu kořenových stromů, resp. stromů na testování pěstěných stromů.

U kořenových stromů lze využít kódy, pokud se podaří určit pořadí jejich synů jednoznačně až na izomorfismus. Na pořadí synů ovšem nezáleží právě tehdy, když jsou podgrafy určené jejich následníky izomorfní.

Nyní převedeme testování izomorfismu kořenových stromů, resp. stromů na testování pěstěných stromů.

U kořenových stromů lze využít kódy, pokud se podaří určit pořadí jejich synů jednoznačně až na izomorfismus. Na pořadí synů ovšem nezáleží právě tehdy, když jsou podgrafy určené jejich následníky izomorfní.

Využijeme proto obdobu (rekurzivní) konstrukce kódu pro pěstěné stromy – budeme postupovat obdobně s využitím lexikografického (slovníkového) uspořádaní synů podle jejich kódů. Kořenový strom budeme tedy popisovat zřetězováním částí slov tak, že má-li otec v syny již označeny kódy W_1, \dots, W_ℓ , pak pro otce použijeme slovo

$$0W_1 \dots W_\ell 1$$

kde pořadí W_1, \dots, W_ℓ je zvoleno tak aby $W_1 \leq W_2 \leq \dots \leq W_\ell$.

Pokud není určen kořen ve stromě, můžeme jej určit tak, aby byl „přibližně uprostřed stromu“.

Pokud není určen kořen ve stromě, můžeme jej určit tak, aby byl „přibližně uprostřed stromu“.

Každému vrcholu stromu T přiřadíme hodnotou $ex_G(v)$ tzv. **výstřednosti** (excentricity), kterou definujeme pro každý vrchol v jako maximální vzdálenost z v do jiného vrcholu w v T .

Tvrzení

Bud' $C(T)$ množina vrcholů stromu T , jejichž výstřednost nabývá minimální hodnoty ($C(T)$ se nazývá střed/centrum grafu, minimální hodnota pak poloměr grafu). Pak $C(T)$ má jeden vrchol nebo dva vrcholy spojené hranou v T .

Pokud není určen kořen ve stromě, můžeme jej určit tak, aby byl „přibližně uprostřed stromu“.

Každému vrcholu stromu T přiřadíme hodnotou $ex_G(v)$ tzv. **výstřednosti** (excentricity), kterou definujeme pro každý vrchol v jako maximální vzdálenost z v do jiného vrcholu w v T .

Tvrzení

Bud' $C(T)$ množina vrcholů stromu T , jejichž výstřednost nabývá minimální hodnoty ($C(T)$ se nazývá střed/centrum grafu, minimální hodnota pak poloměr grafu). Pak $C(T)$ má jeden vrchol nebo dva vrcholy spojené hranou v T .

Důkaz.

Snadno indukcí s využitím triviálního faktu, že nejvzdálenějším vrcholem od každého vrcholu v je nutně list. Centrum T tedy splývá s centrem stromu T' , který vznikne z T vypuštěním listů a příslušných hran. □

Libovolnému stromu přiřadíme jednoznačný kód, až na izomorfismus takto: Pokud je v centru T jediný vrchol, použijeme jej jako kořen; v opačném případě vytvoříme stejným způsobem kód pro dva stromy vzniklé z T odebráním hrany (bez vrcholů) spojující vrcholy x_1, x_2 v $C(T)$, tyto kódy lexikograficky porovnáme a za kód stromu T prohlásíme kód kořenového stromu (T, x) , kde x je ten z vrcholů, jehož komponenta měla lexikograficky menší kód.

Libovolnému stromu přiřadíme jednoznačný kód, až na izomorfismus takto: Pokud je v centru T jediný vrchol, použijeme jej jako kořen; v opačném případě vytvoříme stejným způsobem kód pro dva stromy vzniklé z T odebráním hrany (bez vrcholů) spojující vrcholy x_1, x_2 v $C(T)$, tyto kódy lexikograficky porovnáme a za kód stromu T prohlásíme kód kořenového stromu (T, x) , kde x je ten z vrcholů, jehož komponenta měla lexikograficky menší kód.

Věta

Dva stromy T a T' jsou izomorfní právě, když mají společný kód.

Libovolnému stromu přiřadíme jednoznačný kód, až na izomorfismus takto: Pokud je v centru T jediný vrchol, použijeme jej jako kořen; v opačném případě vytvoříme stejným způsobem kód pro dva stromy vzniklé z T odebráním hrany (bez vrcholů) spojující vrcholy x_1, x_2 v $C(T)$, tyto kódy lexikograficky porovnáme a za kód stromu T prohlásíme kód kořenového stromu (T, x) , kde x je ten z vrcholů, jehož komponenta měla lexikograficky menší kód.

Věta

Dva stromy T a T' jsou izomorfní právě, když mají společný kód.

Poznámka

Z uvedených úvah lze snadno nahlédnout, že algoritmus na testování izomorfismu stromů lze implementovat v lineárním čase vzhledem k počtu vrcholů.

Plán přednášky

Velice často se setkáváme s grafy, které jsou nakresleny v rovině. To znamená, že každý vrchol grafu je ztotožněn s nějakým bodem v rovině a hrany mezi vrcholy v a w odpovídají spojitým křivkám $c : [0, 1] \rightarrow \mathbb{R}^2$ spojujícím vrcholy $c(0) = v$ a $c(1) = w$. Pokud navíc platí, že se jednotlivé dvojice hran protínají nejvýše v koncových vrcholech, pak hovoříme o **rovinném grafu** G .

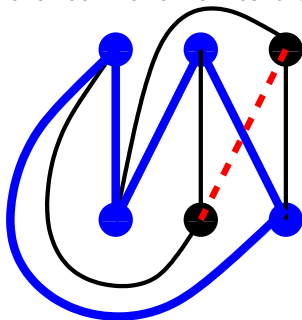
Velice často se setkáváme s grafy, které jsou nakresleny v rovině. To znamená, že každý vrchol grafu je ztotožněn s nějakým bodem v rovině a hrany mezi vrcholy v a w odpovídají spojitým křivkám $c : [0, 1] \rightarrow \mathbb{R}^2$ spojujícím vrcholy $c(0) = v$ a $c(1) = w$. Pokud navíc platí, že se jednotlivé dvojice hran protínají nejvýše v koncových vrcholech, pak hovoříme o **rovinném grafu** G . Otázka, jestli daný graf připouští realizaci (nakreslení) jako rovinný graf, vyvstává velice často v aplikacích.

Jednoduchý příklad je následující:

Tři dodavatelé vody, elektřiny a plynu mají každý své jedno přípojné místo v blízkosti tří rodinných domků. Chtějí je všichni napojit tak, aby se jejich sítě nekřížily (třeba se jim nechce kopat příliš hluboko. . .). Je to možné zvládnout? Odpověď zní „není“.

Jednoduchý příklad je následující:

Tři dodavatelé vody, elektřiny a plynu mají každý své jedno přípojná místa v blízkosti tří rodinných domků. Chtějí je všichni napojit tak, aby se jejich sítě nekřížily (třeba se jim nechce kopat příliš hluboko. . .). Je to možné zvládnout? Odpověď zní „není“. Jde o bipartitní úplný graf $K_{3,3}$, kde tři vrcholy představují přípojná místa, další tři pak domky. Hrany jsou linie sítí. Všechny hrany umíme zvládnout, jedna poslední ale už nejde, viz obrázek na kterém neumíme čárkovanou hranu nakreslit bez křížení:



Obecně se dá ukázat tzv. Kuratowského věta:

Věta

Graf G je rovinný právě tehdy když žádný jeho podgraf není izomorfní dělení grafu $K_{3,3}$ nebo grafu K_5 .

Obecně se dá ukázat tzv. Kuratowského věta:

Věta

Graf G je rovinný právě tehdy když žádný jeho podgraf není izomorfní dělení grafu $K_{3,3}$ nebo grafu K_5 .

Jedna implikace je zřejmá – dělením rovinného grafu vzniká vždy opět rovinný graf a jestliže podgraf nelze v rovině nakreslit bez křížení, totéž musí platit i pro celý graf G . Opačný směr důkazu je naopak velice složitý a nebudeme se jím zde zabývat.

Obecně se dá ukázat tzv. Kuratowského věta:

Věta

Graf G je rovinný právě tehdy když žádný jeho podgraf není izomorfní dělení grafu $K_{3,3}$ nebo grafu K_5 .

Jedna implikace je zřejmá – dělením rovinného grafu vzniká vždy opět rovinný graf a jestliže podgraf nelze v rovině nakreslit bez křížení, totéž musí platit i pro celý graf G . Opačný směr důkazu je naopak velice složitý a nebudeme se jím zde zabývat.

Problematicke rovinných grafů je věnováno ve výzkumu a aplikacích hodně pozornosti, my se zde omezíme pouze na vybrané ilustrace. Zmiňme alespoň naokraj, že existují algoritmy, které testují rovinnost grafu na n vrcholech v čase $O(n)$, což určitě nejde přímou aplikací Kuratowského věty.

Uvažme (konečný) rovinný graf G , včetně jeho nakreslení v \mathbb{R}^2 a necht' $\mathbb{R}^2 \setminus G$ je množina všech bodů $x \in \mathbb{R}^2$, které nepatří žádné hraně, ani nejsou vrcholem. Množina $\mathbb{R}^2 \setminus G$ se rozpadne na disjunktní souvislé podmnožiny S_i , kterým říkáme **stěny rovinného grafu** G . Jedna stěna je výjimečná – ta jejíž doplněk obsahuje všechny vrcholy grafu. Budeme jí říkat neohraničená stěna S_0 . Množinu všech stěn budeme označovat $S = \{S_0, S_1, \dots, S_k\}$ a rovinný graf $G = (V, E, S)$.

Jako příklad si můžeme rozebrat stromy. Každý strom je zjevně rovinný graf, jak je vidět například z možnosti realizovat jej postupným přidáváním listů k jedinému vrcholu. Samozřejmě také můžeme použít Kuratowského větu – když není v G žádná kružnice, nemůže obsahovat jakékoliv dělení grafů $K_{3,3}$ nebo K_5 . Protože strom G neobsahuje žádnou kružnici, dostáváme pouze jedinou stěnu S_0 a to tu neohrazenou. Protože víme, jaký je rozdíl mezi počty vrcholů a hran pro všechny stromy, dostáváme vztah

$$|V| - |E| + |S| = 2.$$

Vztah mezi počty hran, stěn a vrcholů lze odvodit pro všechny rovinné grafy. Jde o tzv. Eulerův vztah. Všimněme si, že z něho zejména vyplývá, že počet stěn v rovinném grafu nezávisí na způsobu, jaké jeho rovinné nakreslení vybereme.

Věta

Nechť $G = (V, E, S)$ je souvislý rovinný graf. Pak platí

$$|V| - |E| + |S| = 2.$$

Vztah mezi počty hran, stěn a vrcholů lze odvodit pro všechny rovinné grafy. Jde o tzv. Eulerův vztah. Všimněme si, že z něho zejména vyplývá, že počet stěn v rovinném grafu nezávisí na způsobu, jaké jeho rovinné nakreslení vybereme.

Věta

Nechť $G = (V, E, S)$ je souvislý rovinný graf. Pak platí

$$|V| - |E| + |S| = 2.$$

Důkaz.

Indukcí podle počtu hran. □

Rovinné grafy si můžeme dobře představit jako namalované na povrchu koule místo v rovině. Sféra vznikne z roviny tak, že přidáme jeden bod „v nekonečnu“. Opět můžeme stejným způsobem hovořit o stěnách a pro takovýto graf pak jsou všechny jeho stěny rovnocenné (i stěna S_0 je ohraničená).

Rovinné grafy si můžeme dobře představit jako namalované na povrchu koule místo v rovině. Sféra vznikne z roviny tak, že přidáme jeden bod „v nekonečnu“. Opět můžeme stejným způsobem hovořit o stěnách a pro takovýto graf pak jsou všechny jeho stěny rovnocenné (i stěna S_0 je ohraničená).

Naopak, každý konvexní mnohostěn $P \subset \mathbb{R}^3$ si můžeme představit jako graf nakreslený na povrchu koule. Vypuštěním jednoho bodu uvnitř jedné ze stěn (ta stane neohraničenou stěnou S_0) pak obdržíme rovinný graf jako výše.

Rovinné grafy, které vzniknou z konvexních mnohostěnů, jsou zjevně 2–souvislé, protože každé dva vrcholy v konvexním mnohostěnu leží na společné kružnici. Navíc v nich platí, že každá stěna kromě S_0 je vnitřkem nějaké kružnice a S_0 je vnějškem nějaké kružnice. Názorné se zdá i to, že ve skutečnosti budou grafy vznikající z konvexních mnohostěnů 3–souvislé.

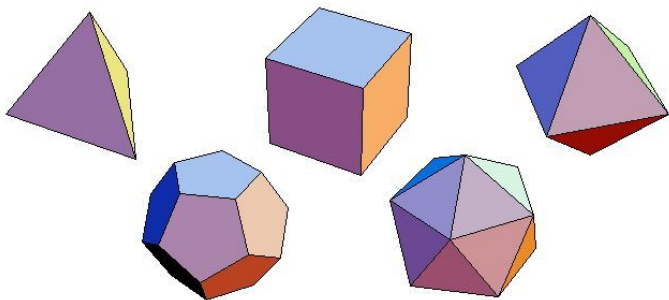
Rovinné grafy, které vzniknou z konvexních mnohostěnů, jsou zjevně 2-souvislé, protože každé dva vrcholy v konvexním mnohostěnu leží na společné kružnici. Navíc v nich platí, že každá stěna kromě S_0 je vnitřkem nějaké kružnice a S_0 je vnějškem nějaké kružnice. Názorné se zdá i to, že ve skutečnosti budou grafy vznikající z konvexních mnohostěnů 3-souvislé. Ve skutečnosti platí dosti náročná Steinitzova věta:

Věta

Libovolný vrcholově 3-souvislý rovinný graf G vzniká z konvexního mnohostěnu v \mathbb{R}^3 .

Jako ilustraci kombinatorické práce s grafy odvodíme klasifikaci tzv. pravidelných mnohostěnů, tj. mnohostěnů poskládaných ze stejných pravidelných mnohoúhelníků tak, že se jich v každém vrcholu dotýká stejný počet. Již v dobách antického myslitele Platóna se vědělo, že jich je pouze pět:

Jako ilustraci kombinatorické práce s grafy odvodíme klasifikaci tzv. pravidelných mnohostěnů, tj. mnohostěnů poskládaných ze stejných pravidelných mnohoúhelníků tak, že se jich v každém vrcholu dotýká stejný počet. Již v dobách antického myslitele Platóna se vědělo, že jich je pouze pět:



Přeložíme si požadavek pravidelnosti do vlastností příslušného grafu: chceme aby každý vrchol měl stejný stupeň $d \geq 3$ a zároveň aby na hranici každé stěny byl stejný počet $k \geq 3$ vrcholů. Označme n počet vrcholů, e počet hran a s počet stěn. Máme k dispozici jednak vztah provazující stupně vrcholů s počtem hran:

$$dn = 2e$$

Přeložíme si požadavek pravidelnosti do vlastností příslušného grafu: chceme aby každý vrchol měl stejný stupeň $d \geq 3$ a zároveň aby na hranici každé stěny byl stejný počet $k \geq 3$ vrcholů. Označme n počet vrcholů, e počet hran a s počet stěn. Máme k dispozici jednak vztah provazující stupně vrcholů s počtem hran:

$$dn = 2e$$

a podobně počítáme počet hran, které ohraničují jednotlivé stěny, a bereme v úvahu, že každá je hranicí dvou stěn, tj.

$$2e = ks.$$

Eulerův vztah pak říká

$$2 = n - e + s = \frac{2e}{d} - e + \frac{2e}{k}.$$

Úpravou odtud dostáváme pro naše známé d a k vztah

$$\frac{1}{d} + \frac{1}{k} = \frac{1}{2} + \frac{1}{e}.$$

Protože e a n musí být přirozená čísla (tj. zejména je $\frac{1}{e} > 0$) a minimum pro d i k je 3, dostáváme přímou diskusí všech možností tento výčet:

d	k	n	e	s
3	3	4	6	4
3	4	8	12	6
4	3	6	12	8
3	5	20	30	12
5	3	12	30	20

Tabulka zadává všechny možnosti. Ve skutečnosti ale také všechny odpovídající pravidelné mnohostěny existují - již jsme je viděli.

Maximální počet hran

Věta

Nechť (V, E, S) je rovinný graf s aspoň třemi vrcholy. Pak

$$|E| \leq 3|V| - 6.$$

Rovnost přitom nastává pro maximální rovinný graf, tj. rovinný graf, k němuž nejde při zachování rovinnosti přidat žádnou hranu. Pokud navíc uvažovaný graf neobsahuje trojúhelník (tj. K_3 jako podgraf), platí dokonce $|E| \leq 2|V| - 4$.

Maximální počet hran

Věta

Nechť (V, E, S) je rovinný graf s aspoň třemi vrcholy. Pak

$$|E| \leq 3|V| - 6.$$

Rovnost přitom nastává pro maximální rovinný graf, tj. rovinný graf, k němuž nejde při zachování rovinnosti přidat žádnou hranu. Pokud navíc uvažovaný graf neobsahuje trojúhelník (tj. K_3 jako podgraf), platí dokonce $|E| \leq 2|V| - 4$.

Důkaz.

Maximální rovinný graf má všechny stěny ohraničené kružnicí délky 3, z čehož plyne $3|S| = 2|E|$ a odtud již pomocí Eulerova vztahu dostáváme první tvrzení. Podobně v druhé části. □

Důsledek

- K_5 není rovinný;

Důsledek

- K_5 není rovinný;
- $K_{3,3}$ není rovinný;

Důsledek

- K_5 není rovinný;
- $K_{3,3}$ není rovinný;
- každý rovinný graf obsahuje alespoň jeden vrchol stupně nejvýše 5;

Důsledek

- K_5 není rovinný;
- $K_{3,3}$ není rovinný;
- každý rovinný graf obsahuje alespoň jeden vrchol stupně nejvýše 5;
- každý rovinný graf bez trojúhelníků obsahuje alespoň jeden vrchol stupně nejvýše 3.

Problém čtyř barev

Jedním z nejznámějších kombinatorických problémů je otázka:

Je možné každou mapu obarvit 4 barvami?

Tento problém sice na první pohled vypadá ryze geometricky, ale dá se přeformulovat do kombinatorické podoby.

Definice

Mapou nazýváme souvislý rovinný multigraf bez mostů. Normální mapou pak mapu, jejíž všechny vrcholy jsou stupně 3. Obarvení mapy je funkce, která každé stěně mapy přiřadí číslo (barvu).

Problém čtyř barev byl rozřešen teprve po více než sto letech bádání – mnoho matematiků na prezentovaný důkaz stále pohlíží s despektem, protože je založen na prověření velkého množství případů pomocí počítače. Elementárními kombinatorickými prostředky je možné alespoň dokázat možnost obarvení normálních map pěti barvami – viz literatura.

Věta (Appel, Haken (1976))

Každou normální mapu je možné obarvit pomocí čtyř barev.

Plán přednášky

V praktických aplikacích často zadává graf všechny možnosti propojení mezi objekty, příkladem může být třeba silniční nebo vodovodní nebo elektrická síť. Pokud nám stačí zajistit propojitelnost každých dvou vrcholů při minimálním počtu hran, hledáme vlastně v grafu G faktor T , který je stromem.

V praktických aplikacích často zadává graf všechny možnosti propojení mezi objekty, příkladem může být třeba silniční nebo vodovodní nebo elektrická síť. Pokud nám stačí zajistit propojitelnost každých dvou vrcholů při minimálním počtu hran, hledáme vlastně v grafu G faktor T , který je stromem.

Definice

Libovolný strom $T = (V, E')$ v grafu $G = (V, E)$, $E' \subseteq E$ se nazývá **kostra** (*spanning tree*) grafu G (tj. faktor grafu, který neobsahuje kružnice).

V praktických aplikacích často zadává graf všechny možnosti propojení mezi objekty, příkladem může být třeba silniční nebo vodovodní nebo elektrická síť. Pokud nám stačí zajistit propojitelnost každých dvou vrcholů při minimálním počtu hran, hledáme vlastně v grafu G faktor T , který je stromem.

Definice

Libovolný strom $T = (V, E')$ v grafu $G = (V, E)$, $E' \subseteq E$ se nazývá **kostra** (*spanning tree*) grafu G (tj. faktor grafu, který neobsahuje kružnice).

Evidentně může kostra v grafu existovat pouze pokud je graf G souvislý. Místo formálního důkazu, že platí i opak uvedeme přímo algoritmus, jak kostru grafu sestavit.

Počet koster grafu K_n

Věta (Cayleyho formule)

Pro $n \geq 2$ je počet koster $\kappa(K_n)$ na K_n (tj. počet stromů na daných n vrcholech) roven n^{n-2} .

Poznámka

Počet koster je významný pojem používaný v mnoha aplikacích. Např. v elektrotechnice, při hypotetickém předpokladu jednotkového odporu mezi každými dvěma vrcholy spojenými hranou, naměříme mezi 2 vrcholy spojenými hranou (vodičem) odpor, který je roven počtu koster obsahujících tuto hranu lomeno celkový počet koster v grafu

Důkaz: Není znám žádný přímočarý způsob, jak dokázat platnost této jednoduché formule, lze ji ale dokázat mnoha různými způsoby (např. pomocí skóre, kódování koster, determinantů, či počítání *povykosů* – viz [MN]).

Důkaz: Není znám žádný přímočarý způsob, jak dokázat platnost této jednoduché formule, lze ji ale dokázat mnoha různými způsoby (např. pomocí skóre, kódování koster, determinantů, či počítání *povykosů* – viz [MN]).

Spočítáme dvěma způsoby *povykosy* (povykos = postup výroby kořenového stromu). Povykos je definován jako trojice (T, r, ν) , kde T je strom na n vrcholech, r jeho kořen a ν očíslování hran, neboli bijekce $\nu : E(T) \rightarrow \{1, 2, \dots, n-1\}$ (začínáme s prázdnou množinou hran a postupně přidáváme hrany v pořadí podle ν).

Důkaz: Není znám žádný přímočarý způsob, jak dokázat platnost této jednoduché formule, lze ji ale dokázat mnoha různými způsoby (např. pomocí skóre, kódování koster, determinantů, či počítání *povykosů* – viz [MN]).

Spočítáme dvěma způsoby *povykosy* (povykos = postup výroby kořenového stromu). Povykos je definován jako trojice (T, r, ν) , kde T je strom na n vrcholech, r jeho kořen a ν očíslování hran, neboli bijekce $\nu : E(T) \rightarrow \{1, 2, \dots, n-1\}$ (začínáme s prázdnou množinou hran a postupně přidáváme hrany v pořadí podle ν).

Pro každý strom T můžeme kořen r zvolit n způsoby a očíslování hran $(n-1)!$ způsoby, proto je počet povykosů $n(n-1)! \cdot \kappa(K_n)$.

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šípkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šipky v $n - 1$ krocích.

1. šipka: $n(n - 1)$ možností.

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šípkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šípky v $n - 1$ krocích.

1. šípka: $n(n - 1)$ možností.

další šípka:

- nesmí vytvořit kružnici;

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šípkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šípky v $n - 1$ krocích.

1. šípka: $n(n - 1)$ možností.

další šípka:

- nesmí vytvořit kružnici;
- na konci musí z každého vrcholu kromě jediného vycházet nějaká šípka, proto musí každá nová šípka vycházet z vrcholu, z něhož ještě žádná nevychází.

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šípkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šípky v $n - 1$ krocích.

1. šípka: $n(n - 1)$ možností.

další šípka:

- nesmí vytvořit kružnici;
- na konci musí z každého vrcholu kromě jediného vycházet nějaká šípka, proto musí každá nová šípka vycházet z vrcholu, z něhož ještě žádná nevychází.

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šipkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šipky v $n - 1$ krocích.

1. šipka: $n(n - 1)$ možností.

další šipka:

- nesmí vytvořit kružnici;
- na konci musí z každého vrcholu kromě jediného vycházet nějaká šipka, proto musí každá nová šipka vycházet z vrcholu, z něhož ještě žádná nevychází.

V každé komponentě již vytvořeného grafu je právě jeden vrchol, z něhož nevychází šipka. Šipka číslo $k + 1$ musí vést do některého vrcholu a vycházet z kořene některé z ostatních komponent – $n(n - k - 1)$ možností.

Důkaz – pokr.:

Druhý způsob: kořenový strom budeme uvažovat jako orientovaný strom se šipkami směřujícími ke kořeni. Začneme s prázdným grafem a budeme přidávat šipky v $n - 1$ krocích.

1. šipka: $n(n - 1)$ možností.

další šipka:

- nesmí vytvořit kružnici;
- na konci musí z každého vrcholu kromě jediného vycházet nějaká šipka, proto musí každá nová šipka vycházet z vrcholu, z něhož ještě žádná nevychází.

V každé komponentě již vytvořeného grafu je právě jeden vrchol, z něhož nevychází šipka. Šipka číslo $k + 1$ musí vést do některého vrcholu a vycházet z kořene některé z ostatních komponent – $n(n - k - 1)$ možností. Celkem máme

$$\prod_{k=0}^{n-2} n(n - k - 1) = (n - 1)! \cdot n^{n-1}$$

způsobů a porovnáním s prvním výpočtem dostaneme tvrzení.