

Orákulum I

1 Abstrakt

Tato práce se zabývá nalezením maximální množiny nezávislých vrcholů obecného grafu G . K tomu máme k dispozici máme „orákulum“, kterému předložíme graf G a číslo k a ono nám odpoví „ano“ v případě že se v grafu nachází množina nezávislých vrcholů o velikosti k a „ne“ v opačném případě. Jde nám o to najít algoritmus jenž bude mít polynomiální složitost. V závěru je zdůvodněna správnost algoritmu.

2 Úvod

Zde bych přistoupil k vysvětlení několika potřebných pojmů.

Nezávislá množina vrcholů v grafu G je taková množina, jejíž žádné dva vrcholy nejsou spojeny hranou. Podgraf indukovaný touto množinou je tedy diskrétní. Maximální nezávislá množina je nezávislá množina, která je maximální vzhledem k inkluzi, tj. nezávislá množina, ke které již nelze přidat žádný vrchol, aniž by přestala být nezávislou.

Nejpočetnější nezávislá množina je nezávislá množina, která má největší počet prvků (mezi všemi nezávislými množinami v grafu G). Každá nejpočetnější množina je samozřejmě i maximální nezávislou množinou ale naopak to neplatí. Jak nejpočetnějších, tak i maximálních nezávislých množin může být v témže grafu několik.

Jako obecný graf chápu prostý neorientovaný graf bez smyček. Vrcholy grafu G značíme n . Jelikož ke splnění cíle využiji i barvení grafu tak tedy přidám i pár pojmů z barevnosti grafů. Obarvení vrcholů grafu je ohodnocení vrcholů hodnotami z množiny B (takzvanými barvami), a to takové, že žádné dva sousední vrcholy nejsou ohodnoceny (obarveny) stejnou barvou. Graf nazýváme r -barevným, jestliže existuje jeho obarvení r barvami. Barevnost grafu je nejmenší počet barev, který je potřeba k obarvení grafu. Barevnost grafu značíme $\chi(G)$. V obecném grafu G platí:

$$\chi(G) \leq \max\{d(x) \mid x \in V(G)\} + 1$$

3 Řešení

Jak jsem již uvedl výše k řešení využiji algoritmus pro obarvování grafu nejmenším možným počtem barev, který je klasickým příkladem backtrackingu

3.1 Algoritmus pro zjišťování barevnosti

Vstup: Graf G jehož vrcholy jsou očíslovány čísly $1, 2, \dots, n$. Pro každý vrchol je dána množina $P(x) = V(x) \cap \{1, 2, \dots, x-1\}$.

Pomocné proměnné: Právě zkoumané částečné obarvení budeme uchovávat v hodnotách $B(x)$. Hodnoty $BARVA(x)$ budou obsahovat nejlepší dosud nalezené obarvení. Proměnná $OMEZ$ bude obsahovat číslo barvy, kterou již nechceme použít ve snaze získat obarvení méně než $OMEZ$ barvami. S výjimkou počáteční fáze bude proměnná $OMEZ$ rovna počtu barev v nejlepší dosud nalezeném obarvení $BARVA$.

Metoda: Vzhledem k pevnému očíslování vrcholů můžeme každé obarvení pokládat za posloupnost barev. Tyto posloupnosti budeme systematicky prodlužovat a zkracovat v duchu backtrackingu.

Při postupu vpřed, tj. při prodlužování částečné posloupnosti barev, dáme následujícímu vrcholu x jako výchozí nejnižší barvu která (momentálně) není použita u vrcholů z množiny $P(x)$. Nižší barvu není možno použít (při stávajících barvách předchozích vrcholů), vyšší barvy budou zkoušeny v dalším průběhu backtrackingu (po provedení návratu).

Návraty v backtrackingu budeme provádět jednak při dosažení úplného obarvení, jednak v situaci, kdy některý vrchol y dostal barvu $OMEZ$ nebo vyšší. Přidělením této barvy jsme totiž dostaly částečné obarvení, jehož prodlužování již nemá smysl, neboť nemůžeme vylepšit nejlepší dosud známe obarvení $BARVA$.

Cílem návratu je snížit barvu ve vrcholu y . Poněvadž však nižší barvy v y jsou již vyzkoušeny je třeba změnit (a tedy zvýšit) barvu v některém předchozím vrcholu $x < y$. Aby to však mělo vliv na vrchol y , musí být $x \in P(y)$, proto provedeme prvý návrat až do vrcholu $x = \max(P(y))$ a v tomto vrcholu se pokusíme zvýšit barvu.

Nelze-li barvu ve vrcholu x zvýšit (byla použita barva OMEZ), provedeme další tentokrát již prostý návrat do vrcholu $x - 1$, $x - 2$, atd.

Barvu vrcholu 1 nezvyšujeme, nemá to smysl.

Z podobného důvodu nemá smysl barvit vrchol x barvou $b > x$.

Na počátku volíme $OMEZ = n + 1$, aby hodnota OMEZ nebránila nalezení pravého obarvení.

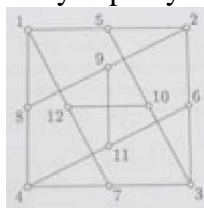
Algoritmus:

1. [Inicializace.] $x = 1$; $B(x) = 1$; $OMEZ = n + 1$.
2. [Postup vpřed – obarvení dalšího vrcholu.] Položíme $x = x + 1$. Jestliže $x > n$, pokračujeme podle kroku 3. V opačném případě položíme $B(x) = \text{minimum ze všech barev nevyskytujících se v množině } P(x)$. Jestliže $B(x) \geq OMEZ$, položíme $y = x$ a pokračujeme podle kroku č, jinak zopakujeme krok 2.
3. [Všechny hodnoty obarveny, počet barev je menší než OMEZ.] Pro všechny vrcholy x grafu G provedeme $BARVA(x) = B(x)$ a dále položíme $OMEZ = \text{maximum z hodnot } BARVA$. Označme y první vrchol, který má tuto barvu omez a pokračujeme krokem 4.
4. [Návrat – pokus o zvýšení barvy ve vrcholu y .] Označme x poslední vrchol z množiny $P(y)$ a pokračujme krokem 5.
5. [Pokračování návratu – pokus o zvýšení barvy ve vrcholu x .] Jestliže $x = 1$, výpočet končí. V opačném případě položíme $b = \text{minimum ze všech barev, které jsou větší než } B(x)$ a nevyskytují se v množině $P(x)$. Pokud $b < OMEZ$ a zároveň $b \leq$, položíme $B(x) = b$ a pokračujeme podle kroku 2, v opačném případě položíme $x = x - 1$ a zopakujeme krok 5

A když získáme takto obarvený graf, tak si seřadíme vrcholy jedné barvy podle počtu vrcholů sestupně, pak k první položce tohoto seznamu přičteme 1 a zeptáme se orákula zda existuje nezávislá množina o počtu vrcholů $n + 1$, v případě že ne udává zajisté první položka maximální nezávislou množinu grafu v případě že ne první položka udává nejpočetnější nezávislou množinu a byly bychom schopni zajisté najít i maximální nezávislou množinu.

3.2 Příklad

Činnost algoritmu předvedu na následujícím grafu. Postup výpočtu je patrný z následující tabulky, kde každý řádek zachycuje hodnoty B po vykonání kroku 4.



1	2	3	4	5	6	7	8	9	10	11	12	
1	1	1	1	2	2	2	2	3	3	4	4	(= obarvení 4 barvami)
1	1	1	1	2	2	2	3	2	3	3	4	
1	1	1	1	2	2	3	2	3	3	4		
1	1	1	1	2	2	3	3	2	3	3	2	(= obarvení 3 barvami)
1	1	1	2	2	2	3						
1	1	2	1	2	3							
1	2	1	1	3								
2												

Čtverečkem je zde vždy vyznačen vrchol y v němž chceme snížit barvu. Návraty v backtrackingu jsou vyznačeny šipkami a prvý vrchol, v němž se p návratu zvýšila barva je v kroužku.

Takže teď bychom si seřadily barvy jimiž jsou obarveny vrcholy podle počtu vrcholů sestupně. Vzali bychom barvu 1 která obarvila 4 vrcholy, k této 4 bychom přičetli 1 a zeptali se orákula jestli v grafu existuje nezávislá množina s 5 vrcholy. Orákulum nám odpoví „ne“ a tudíž víme že jsem našli maximální nezávislou množinu grafu G jež je 4 a tyto nezávislé množiny jsou zde 3 a každá z nich je v tomto případě maximální.

4 Závěr

Rychlost algoritmu velice záleží na očíslování vrcholů. Nejlepší je nejnižšími čísly očíslovat vrcholy s velkým stupněm a nejnižšími čísly očíslovat vrcholy s co největší klikou, také je vhodné očíslovat sousední vrcholy pokud možno sousedními čísly. Pro řešení je využito teorie vrcholového barvení grafu. Barvení grafu patří do NP-úplných grafů. Rychlost algoritmu tedy závisí na tom jak si očíslováme vrcholy grafu, případně jak jsou očíslovány.