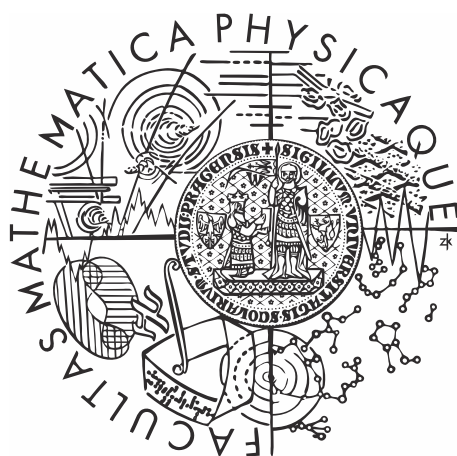


Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Anna Bernáthová

Kreslení grafů s podmínkami na velikosti stěn

Vedoucí bakalářské práce: prof. RNDr. Jan Kratochvíl, CSc.

Studijní program: Informatika, Obecná informatika

2009

Děkuji svému vedoucímu, profesoru Kratochvílovi, za zajímavé téma práce a všechny rady, kterými mi ukazoval cestu vpřed. Dále děkuji Martině Šimůnkové za korektury a nalezení díry v důkazu. Jitce Novotné děkuji za to, že mi pomáhala hledat dál, když už jsem viděla všude jen samé trojúhelníky. Nakonec děkuji Martinovi Marešovi za pomoc se sázením, sháněním článků, hledáním chyb . . . a zejména za poskytování nekonečného množství podpory, bez které bych práci jistě nestihla dopsat včas.

Prohlašuji, že jsem svou bakalářskou práci napsala samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze
dne 26. července 2009

Anna Bernáthová

Obsah

Obsah	3
1. Úvod	5
2. Nakreslení se stejně velkými stěnami	7
2.1. Hledání nakreslení se stejně velkými stěnami	7
2.2. Gröbnerovy báze a soustavy polynomiálních rovnic	9
2.3. Výpočet nakreslení se stejně velkými stěnami	10
3. Grafy s předepsanými obsahy stěn	13
3.1. Grafy, které lze nakreslit s libovolnými obsahy stěn	13
3.2. Hypotéza o triangulacích s minimálním stupněm větším než tři	14
3.3. Vyvrácení hypotézy	18
4. Závěr	25
A. Programy na CD	27
A.1. Program na generování nakreslení	27
A.2. Triangulace do deseti vrcholů	28
A.3. Graf z obrázku 6	29
B. Seznam literatury	31

Název práce: Kreslení grafů s podmínkami na velikosti stěn

Autor: Anna Bernáthová

Katedra: Katedra aplikované matematiky

Vedoucí bakalářské práce: prof. RNDr. Jan Kratochvíl, CSc.

e-mail vedoucího: Jan.Kratochvil@mff.cuni.cz

Abstrakt: V této práci se věnujeme úsečkovým nakreslením rovinných grafů s předepsanými obsahy stěn. Pomocí genetického algoritmu hledáme grafy, které pravděpodobně nelze nakreslit se všemi stěnami stejně velkými. Pro jeden takový graf to dokážeme vyřešením soustavy polynomiálních rovnic více proměnných. Dále popisujeme grafy, které lze nakreslit s libovolně předepsanými obsahy. Zkoušíme též ověřit hypotézu, že všechny triangulace s minimálním stupněm větším než tři nemají nakreslení pro libovolně předepsané obsahy stěn a zkoumáme všechny takové triangulace s počtem vrcholů nejvýše deset – zkoušíme náhodně generovat předepsané obsahy a ověřovat pro ně existenci nakreslení. Nakonec hypotézu vyvrátíme.

Klíčová slova: úsečková rovinná nakreslení, omezené velikosti stěn, grafy se stejně velkými obsahy stěn

Title: Graph drawing with constrained face sizes

Author: Anna Bernáthová

Department: Department of Applied Mathematics

Supervisor: prof. RNDr. Jan Kratochvíl, CSc.

Supervisor's e-mail address: Jan.Kratochvil@mff.cuni.cz

Abstract: In this thesis, we are working with straight-line drawings of planar graphs with prescribed face areas. We employ a genetic algorithm to help with searching for graphs that likely cannot be drawn in a way that all their faces have equal areas. Then we pick one of these graphs and after solving a system of polynomial equations in several variables, we formally prove that the graph cannot be drawn in such a way. We continue by describing graphs that can be drawn with arbitrary prescribed face areas. We also try to verify a conjecture that all the planar triangulations with minimum degree greater than three cannot be drawn with arbitrary prescribed face areas. We investigate all such triangulations with ten vertices or less – we try to set areas of their faces at random and we verify that a drawing with such areas really exists. We disprove this conjecture at the end.

Keywords: straight-line drawings, constrained face sizes, equiareal graphs

1. Úvod

V následujícím textu se budeme zabývat kreslením rovinných grafů tak, aby jejich vnitřní stěny měly předepsané obsahy. Budeme přitom pracovat s konečnými grafy bez smyček a násobných hran. Nebudeme definovat základní pojmy jako např. graf, cesta nebo souvislost. Definice, které používáme, se shodují například s těmi z knížky Kapitoly z diskretní matematiky [MN07].

V následujících odstavcích si nadefinujeme základní pojmy týkající se kreslení grafů a uvedeme znění několika důležitých tvrzení.

1.0.1. Definice. *Oblouk* je podmnožina euklidovské roviny ve tvaru $\varphi[[0, 1]]$, kde $\varphi : [0, 1] \rightarrow \mathbb{R}^2$ je nějaké spojitě prosté zobrazení uzavřeného intervalu $[0, 1]$ do roviny. Body $\varphi(0)$ a $\varphi(1)$ nazýváme *koncovými body* oblouku. *Topologická kružnice* je oblouk, jehož koncové body splývají.

1.0.2. Definice. *Nakreslení grafu* je dvojice zobrazení, která vrcholům grafu přiřazuje navzájem různé body roviny (*obrazy vrcholů*) a každé hraně $\{u, v\}$ oblouk (*obraz hrany*), jehož koncovými body jsou obrazy vrcholů u a v .

Nakreslení je *rovinné*, pokud v průniku obrazů dvou různých hran leží pouze obrazy vrcholů společných oběma hranám a každý obraz vrcholu ležící v nějakém obrazu hrany je jeho koncovým bodem.

Uzávěrům oblastí, na které dělí rovinu sjednocení obrazů vrcholů a hran, budeme říkat *stěny*. Nakreslení má právě jednu stěnu, které není omezená, tu budeme nazývat *vnější stěna*. Ostatní jsou *vnitřní stěny*.

Vrcholům a hranám, jejichž obrazy leží na (topologické) hranici stěny, budeme říkat její *hraniční vrcholy* a *hrany* a příslušný podgraf budeme nazývat (kombinatorickou) *hranicí stěny*.

Hranice stěny může být obecně docela složitá (uvažme například nakreslení stromu), ale jak ukáže následující věta, pro dostatečně propojené grafy platí, že je každá stěna ohraničena kružnicí.

1.0.3. Definice. Graf je (vrcholově) *k-souvislý*, pokud má alespoň $k + 1$ vrcholů a pokud je souvislý po odebrání libovolných méně než k vrcholů a všech hran, které z nich vedou.

1.0.4. Věta. V každém rovinném nakreslení 2-souvislého grafu je každá stěna ohraničena kružnicí.

Důkaz. Viz Diestel [Die06] strana 89, tvrzení 4.2.6. ◻

1.0.5. Definice. Graf nazýváme *rovinným grafem*, pokud má alespoň jedno rovinné nakreslení.

1.0.6. Definice. Jsou-li hranice stěn kružnice, můžeme každé stěně přiřadit vrcholy cyklicky uspořádané tak, jak leží na kružnici. Množině těchto cyklických posloupností budeme říkat *kombinatorická struktura* nakreslení.

1.0.7. Věta. Všechna nakreslení 3-souvislého rovinného grafu mají (až na obrácení pořadí vrcholů ve všech stěnách a určení toho, která stěna je vnější) stejnou kombinatorickou strukturu.

Důkaz. Viz Diestel [Die06] strana 96, věta 4.3.2. ◻

1.0.8. Definice. Rovinný graf nazveme *maximálním*, pokud přestane být rovinným grafem přidáním libovolné hrany. Každý maximální rovinný graf, který má aspoň čtyři vrcholy, je 3-souvislý a jeho stěny jsou ohraničeny trojúhelníky. Takový graf nazýváme *triangulace*.

1.0.9. Věta. (*Eulerova formule*)

Nechť má souvislý graf n vrcholů, m hran a l stěn. Potom $n - m + l = 2$.

Důkaz. Viz Diestel [Die06] strana 91, věta 4.2.9. ▽

1.0.10. Věta. Každá triangulace na n vrcholech má $3n - 6$ hran.

Důkaz. Důkaz je jednoduchým důsledkem Eulerovy formule, viz Diestel [Die06] strana 91, důsledek 4.2.10.

Přestože existence rovinného nakreslení vypadá na první pohled jako topologický problém, plyne pouze z kombinatorických vlastností grafu:

1.0.11. Věta. (*Kuratowského věta*)

Graf je rovinný právě tehdy, když neobsahuje jako minor K_5 nebo $K_{3,3}$.

Důkaz. Viz Diestel [Die06] strana 101, věta 4.4.6. ▽

1.0.12. Definice. *Úsečkové nakreslení grafu* je takové nakreslení, v kterém jsou všechny obrazy hran úsečky.

1.0.13. Věta. Každý rovinný graf o n vrcholech má úsečkové nakreslení v mřížce o velikosti $n \times n$.

Důkaz. Viz Schnyderův článek [Sch90]. ▽

Jsou známy algoritmy, které dokáží rozhodnout, jestli je graf rovinný, v lineárním čase. První takový algoritmus ukázal Tarjan v [TH74], od té doby byly objeveny i jednodušší algoritmy, jeden takový je popsán v [BM04]. Existuje i mnoho specializovaných algoritmů na kreslení rovinných grafů s dalšími přidanými požadavky. Jako příklad uvedeme úsečková nakreslení rovinných grafů do mřížky (tak, aby obrazy vrcholů měly celočíselné souřadnice). O tomto a velkém množství dalších algoritmů se lze dočíst např. v knize [DBETT99].

My se budeme zabývat problémem, jak k danému grafu a kombinatorické struktuře nakreslení najít takové nakreslení, aby jeho stěny byly konvexní mnohoúhelníky s předepsanými obsahy. Budeme studovat pouze triangulace, protože ani u nich zatím není známo, které grafy s předepsanými podmínkami nakreslit lze, a které ne. Pokud v dalším textu budeme mluvit o grafu, vždy budeme mít na mysli triangulaci. Zmíníme-li se o nakreslení, vždy budeme uvažovat o rovinném úsečkovém nakreslení.

2. Nakreslení se stejně velkými stěnami

Je známo, že existují takové grafy (viz Thomassenův článek [Tho92]), které nelze nakreslit do roviny tak, aby měly všechny jejich vnitřní stěny stejně velké obsahy. V této kapitole se budeme zabývat tím, jak takové grafy hledat. Nejprve se pokusíme problém, jak ověřit, má-li graf žádané nakreslení, řešit pomocí genetického algoritmu. Potom výsledky tohoto hledání podpoříme výpočtem.

2.1. Hledání nakreslení se stejně velkými stěnami

Máme-li graf, o kterém si myslíme, že nemá nakreslení se stejně velkými stěnami, můžeme to zkusit dokázat. K tomu se ale dostaneme až v závěru této kapitoly. Nejdříve nějaký takový graf musíme najít.

K odhadu, jestli má graf požadované nakreslení nebo ne, použijeme heuristiku založenou na genetickém algoritmu. Ten dostane na vstup graf a nějaké jeho nakreslení. Bude si udržovat populaci řádově tisíců nakreslení, která nechá náhodně mutovat. Neperspektivní nakreslení zahodí a nahradí je novými, vzniklými zkřížením nějaké náhodné dvojice jiných nakreslení.

Jádrem algoritmu je funkce *Ohodnot*, která pro dané nakreslení grafu vrátí jeho hodnocení – reálné číslo, které klesá k nule s tím, jak se velikosti stěn nakreslení k sobě blíží.

2.1.1. Algoritmus. (*Ohodnot*)

Vstup: Nakreslení grafu G

1. $avg \leftarrow$ průměrný obsah stěny G
2. $s \leftarrow 0$
3. Pro každou stěnu s_i grafu G :
4. $s \leftarrow s + (Obsah(s_i) - avg)^2$

Výstup: s/avg^2

Výběr této hodnotící funkce není podložen žádnou teorií, to, že se chová slušně, vyplynulo z experimentů. Nyní se můžeme podívat na celý algoritmus. Během celého výpočtu budeme pracovat s polem p , v kterém budeme uchovávat všech z nakreslení, s kterými pracujeme, a jejich ohodnocení.

2.1.2. Algoritmus. (*Najdi nakreslení s přibližně stejně velkými stěnami*)

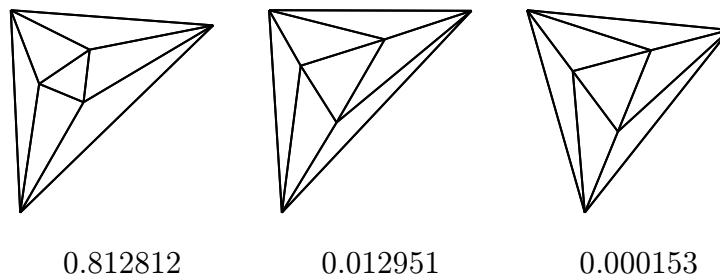
Vstup: Graf G , jeho nakreslení a kombinatorická struktura, cílové hodnocení, které má nakreslení splňovat

1. Vytvoř pole $p = (p_1, \dots, p_z)$ tak, aby obsahovalo z kopií vstupního nakreslení.
2. Opakuj:
3. Pro každou položku pole p : $Ohodnot(p_i)$
4. Setříd p podle hodnocení nakreslení.
5. Pokud je hodnocení p_1 lepší než cílové hodnocení, vrať p_1 jako výsledek.
6. Pro každou položku p_i v nejhůře hodnocené pětině pole p :

7. Vyber dvě náhodná (lepší s vyšší pravděpodobností) nakreslení z p .
8. Zprůměruj souřadnice všech vrcholů, ulož do p_i .
9. Dokud nezískáme takové nakreslení, které neporušuje rovinnost, opakuj předchozí dva kroky.
10. Pro každou položku pole p :
11. Vyber náhodný počet vrcholů a posuň jejich souřadnice o náhodnou vzdálenost.
12. Dokud nezískáme takové nakreslení, které neporušuje rovinnost, opakuj předchozí krok.

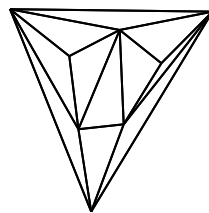
Výstup: Nakreslení s požadovaným nebo lepším hodnocením

Pro grafy, které mají požadované nakreslení, náš algoritmus obvykle generuje nakreslení ohodnocená přibližně 0.0001. Na následujících obrázcích osmistěnu můžeme nahlédnout, jak se s klesajícím hodnocením nakreslení přibližují obsahy stěn.



Obr. 1: Různá nakreslení osmistěny s klesajícím ohodnocením

Grafům, které se se stejně velkými stěnami nakreslit nedají, vygeneruje algoritmus nakreslení s hodnocením kolem 0.01, jeden takový si můžeme prohlédnout na následujícím obrázku.



Obr. 2: Graf bez požadovaného nakreslení. Přiblížení s hodnocením 0.014002.

Nyní tedy máme k dispozici nástroj, který nám pomůže odhadnout, jestli je pro daný graf lepší požadované nakreslení zkusit hledat nebo raději dokazovat, že neexistuje. Při řešení této úlohy se neobejdeme bez soustav polynomiálních rovnic více proměnných, v další sekci si proto stručně projdeme, jak je řešit. V závěrečné sekci konečně získáme graf, který není možné nakreslit se stejně velkými stěnami a dokážeme to o něm.

Nakreslení grafů, která algoritmus generuje, vypadají sice celkem pěkně, ale je na nich znát, že optimalizuje jen pro velikosti ploch, a ne i pro velikosti úhlů. Pokud bychom jej chtěli používat v praxi na generování esteticky přijatelných nakreslení, museli bychom do hodnotící funkce zapracovat ještě preferenci nakreslení s ostrými úhly.

2.2. Gröbnerovy báze a soustavy polynomiálních rovnic

V dalším textu se budeme zabývat tím, jak pomocí Gröbnerovýchází řešení soustavy polynomiálních rovnic. Ukážeme si jen to nejdůležitější, podrobnosti, včetně Buchbergerova algoritmu pro jejich hledání, lze najít v Heckově článku [Hec97]. Samotné hledáníází a řešení soustav rovnic potom přenecháme implementaci tohoto algoritmu v programu Mathematica.

Máme-li polynom ve více proměnných, potřebujeme si nejprve na jeho členech nadefinovat uspořádání, které budeme značit \prec . Možností, jak to udělat, je více, je třeba jen zajistit, aby toto uspořádání mělo následující vlastnosti:

- $1 \prec t$ pro každý nekonstantní člen $t \neq 1$,
- $s \prec t \Rightarrow s \cdot u \prec t \cdot u$ pro všechny členy s, t, u .

Často se používá lexikografické uspořádání.

2.2.1. Definice. Pro každý nenulový polynom p definujeme *počáteční člen* $lt(p)$ jako maximální člen mezi všemi členy p .

2.2.2. Definice. Máme-li nenulové polynomy p, \tilde{p}, q , řekneme, že p se *redukuje na \tilde{p} modulo q* a značíme $p \rightarrow_q \tilde{p}$, pokud v p existuje takový člen t , který je dělitelný počátečním členem q a $\tilde{p} = p - \frac{t}{u(q)} \cdot q$.

Nechť je $Q = \{q_1, q_2, \dots, q_n\}$ množina polynomů. Polynom p se *redukuje na \tilde{p} modulo Q* , pokud v Q existuje takový polynom q_i , pro který $p \rightarrow_{q_i} \tilde{p}$. *Normální forma vzhledem ke Q* je polynom, který získáme po konečném počtu redukcí, a který neobsahuje žádný člen dělitelný některým z počátečních členů polynomů z Q . Normální forma nemusí být jednoznačně určena.

2.2.3. Definice. Konečná množina polynomů Q je *Gröbnerova báze*, pokud je pro každé p normální forma p modulo Q jednoznačně určena.

2.2.4. Věta. Pro každou množinu polynomů $\{p_1, \dots, p_l\}$ a každé uspořádání \prec existuje Gröbnerova báze $\{q_1, \dots, q_m\}$ taková, že soustava rovnic

$$p_1(x_1, \dots, x_n) = \dots = p_l(x_1, \dots, x_n) = 0$$

má stejná řešení jako soustava

$$q_1(x_1, \dots, x_n) = \dots = q_m(x_1, \dots, x_n) = 0.$$

2.2.5. Příklad. Soustava rovnic

$$\begin{aligned} x^3 + x^2 + y + z &= 0 \\ y^2 + z^3 + x + 2 &= 0 \\ x^2 + z + 2 &= 0 \end{aligned}$$

má při použití lexikografického uspořádání Gröbnerovu bázi

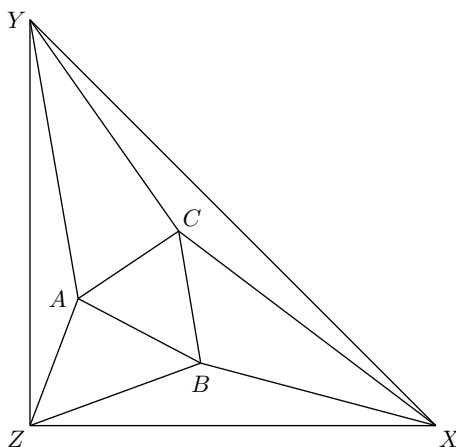
$$\{166 + 273z + 272z^2 + 160z^3 + 36z^4, \\ -142 + 43y - 160z - 104z^2 - 18z^3, \\ 138 + 43x + 124z + 158z^2 + 72z^3\}.$$

Je zřejmé, že najít řešení soustavy polynomů z Gröbnerovy báze je mnohem snazší než snažit se jej získat z původní soustavy, protože první polynom báze obsahuje pouze proměnnou z , druhý jen y a z a až třetí všechny, takže si můžeme snadno spočítat z z první rovnice, potom y z druhé a nakonec x z třetí.

2.3. Výpočet nakreslení se stejně velkými stěnami

V sekci 2.1 jsme zjistili, že existenci nakreslení osmistěnu se stejně velkými stěnami můžeme očekávat. Zkusíme tedy takové nakreslení najít.

Potom se poohlédneme po nějakém předpisu pro velikost stěn takovém, že nakreslení nebude existovat. Když se nám to podaří, zkusíme přidat do osmistěnu nějaké další vrcholy tak, abychom dostali graf, který nelze nakreslit se stejnými velikostmi stěn.



Obr. 3: Osmistěn

Budeme hledat souřadnice bodů A, B, C, X, Y, Z (viz obrázek). Bez újmy na obecnosti si pro ně můžeme předepsat nějaké dodatečné omezující podmínky. Posunutí či otočení trojúhelníku nezmění jeho obsah, regulární lineární zobrazení změní obsahy všech trojúhelníků stejně: vynásobí je hodnotou determinantu zobrazení.

Některý z bodů nakreslení si tedy můžeme položit do počátku, nechť je to bod Z . Otočit si nakreslení můžeme rovněž libovolně, bod X proto může ležet na ose x . Potom si můžeme vybrat takové zvětšení ve směru osy x , které bod X zobrazí na souřadnice $[1, 0]$, a takové zkosení, které zachovává osu x (a tedy již zvolené polohy bodů X a Z) a bod Y zobrazí na osu y . Změněné velikosti obsahů kompenzujeme zvětšením v y -ovém směru. (Kdyby nás zajímaly jen vzájemné poměry obsahů, a ne konkrétní velikosti, mohli bychom si zafixovat i bod Y .)

Nyní nám zbývá najít celkem sedm souřadnic a máme předepsáno sedm obsahů stěn. Sestavíme si tedy soustavu sedmi rovnic o sedmi neznámých a pokusíme se ji vyřešit. Obsahy budeme počítat pomocí determinantů. Máme-li trojúhelník s vrcholy $[0, 0]$, $[x_1, x_2]$ a $[y_1, y_2]$, jeho orientovaný obsah dostaneme jako

$$S = \frac{1}{2} \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix}.$$

Takto tedy vypadají naše rovnice:

$$\begin{aligned} a_1 y_2 &= 1 & (\triangle AZY) \\ -c_1(a_2 - c_2) - (a_1 - c_1)(y_2 - c_2) &= 1 & (\triangle AYC) \\ (a_2 - b_2)(c_1 - b_1) - (a_1 - b_1)(c_2 - b_2) &= 1 & (\triangle ACB) \\ a_2 b_1 - a_1 b_2 &= 1 & (\triangle ABZ) \\ b_2(c_1 - 1) - (b_1 - 1)c_2 &= 1 & (\triangle BCX) \\ b_2 &= 1 & (\triangle BXZ) \\ -(c_1 - 1)y_2 - c_2 &= 1 & (\triangle CYX) \end{aligned}$$

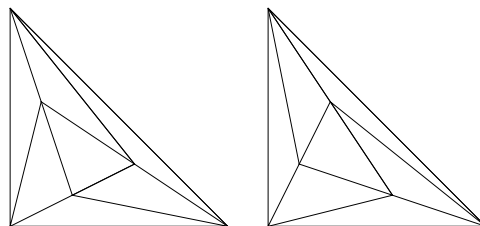
Jejich Gröbnerova báze:

$$\begin{aligned} &\{8 - 6c_2 + c_2^2, \\ &\quad -6 + 7c_1 + c_2, \\ &\quad -6 + a_2 + c_2, \\ &\quad -1 + 7a_1, \\ &\quad -7 + y_2, \\ &\quad -1 + b_2, \\ &\quad 7b_1 - c_2\} \end{aligned}$$

Řešení:

$$a_1 = \frac{1}{7}, a_2 = 4, b_1 = \frac{2}{7}, b_2 = 1, c_1 = \frac{4}{7}, c_2 = 2, y_2 = 7$$

$$a_1 = \frac{1}{7}, a_2 = 2, b_1 = \frac{4}{7}, b_2 = 1, c_1 = \frac{2}{7}, c_2 = 4, y_2 = 7$$



Obr. 4: Dvě nakreslení osmistěnu se stejně velkými stěnami

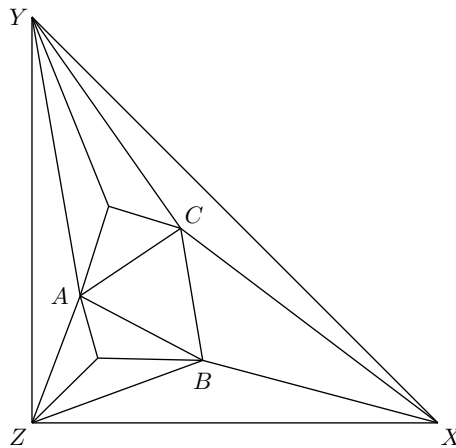
Nyní se pokusíme najít nějaký graf, který nemá nakreslení se stejnými obsahy stěn. Vyjdeme z osmistěny a najdeme takové ohodnocení obsahů všech jeho stěn, aby výše uvedená soustava rovnic neměla řešení. Budeme se přitom snažit, aby velikosti těchto předepsaných obsahů byla lichá celá čísla, protože potom můžeme snadno přidat další vrcholy tak, aby se nám stěny s větším obsahem než 1 rozpadly na stěny s jednotkovým obsahem.

Jedno takové ohodnocení přiřazuje stěnám AYC a ABZ velikost 3 a ostatním velikost 1. Gröbnerova báze této soustavy rovnic je

$$\begin{aligned} &\{36 - 20c_2 + 3c_2^2, \\ &\quad - 10 + 11c_1 + c_2, \\ &\quad - 30 + 4a_2 + 3c_2, \\ &\quad - 1 + 11a_1, \\ &\quad - 11 + y_2, \\ &\quad - 1 + b_2, \\ &\quad - 10 + 33b_1 - 3c_2\} \end{aligned}$$

Již z toho, že diskriminant prvního polynomu je záporný, je zřejmé, že soustava rovnic nemá reálné řešení.

Pokud tedy přidáme ke grafu další dva vrcholy, jejich obrazy umístíme do těchto stěn a k vrcholům, jejichž obrazy jsou body A, Y, C a A, B, Z je připojíme hranami, získáme hledaný graf.



Obr. 5: Hledaný graf

3. Grafy s předepsanými obsahy stěn

V první sekci této kapitoly ukážeme, že některé grafy mají nakreslení pro jakékoli předepsané obsahy stěn. Potom se naopak budeme věnovat grafům, které tuto podmínku nesplňují. Budeme pracovat s hypotézou, že mezi ně patří triangulace, které mají všechny stupně větší než tři. Hypotézu nakonec vyvrátíme.

3.1. Grafy, které lze nakreslit s libovolnými obsahy stěn

V Thomassenově článku [Tho92] je dokázáno, že každý kubický rovinný graf má úsečkové nakreslení pro libovolně předepsané obsahy stěn. Tento důkaz je veden poměrně složitým rozborem případů, v kterém je nejprve pro každý případ nalezeno nějaké nekorektní nakreslení (takové, které vrcholy grafu může zobrazit na obrazy jiných vrcholů nebo na obraz hrany, na které vrchol neleží) a potom se ukáže, že existence tohoto nakreslení zaručuje i existenci korektního nakreslení, v kterém jsou všechny předepsané obsahy zachovány.

Další zajímavé triangulace, které je možné nakreslit s libovolnými předepsanými obsahy stěn, lze zkonstruovat takto:

3.1.1. Definice. *Rekurzivní triangulace* je taková triangulace, kterou lze zkonstruovat následujícím způsobem: Na začátku vezmeme trojúhelník, a potom opakujeme následující postup: do libovolné vnitřní (trojúhelníkové) stěny grafu umístíme vrchol, který připojíme hranami ke všem třem vrcholům na hranici původní stěny.

3.1.2. Věta. Nakreslení každé rekurzivní triangulace lze zkonstruovat pro libovolně předepsané obsahy stěn.

Důkaz. Budeme postupovat indukcí podle délky konstrukce triangulace. Postupně zkonstruujeme trojúhelníky tak, aby byl jejich obsah roven součtu předepsaných obsahů stěn, které v trojúhelníku leží.

Trojúhelník jistě můžeme nakreslit tak, aby byl jeho obsah součtem všech předepsaných obsahů stěn.

Nyní mějme graf, který můžeme nakreslit tak, aby jeho stěny měly libovolný obsah. Do jedné z jeho trojúhelníkových stěn (nazvěme její vrcholy X , Y a Z) přidejme vrchol P a spojme jej hranami s vrcholy X , Y , Z . Předepsané obsahy nově vzniklých stěn označme S_{XYP} , S_{YZP} , S_{ZXP} . Již víme, že body X , Y , Z můžeme nakreslit tak, aby měl trojúhelník XYZ obsah $S_{XYP} + S_{YZP} + S_{ZXP}$. Protože obsah trojúhelníka je roven polovině délky součinu nějaké jeho strany a výšky na tuto stranu, víme, že bod P musí ležet na nějaké přímce p , procházející trojúhelníkem XYZ , rovnoběžné s XY , a také na přímce q , rovnoběžné s YZ . Tyto přímky se v trojúhelníku protínají. Umístíme-li bod do jejich průsečíku, je splněn obsah S_{XYP} i S_{YZP} . Splnění obsahu S_{ZXP} potom již plyne z toho, že trojúhelník XYZ má obsah $S_{XYP} + S_{YZP} + S_{ZXP}$. ∇

3.2. Hypotéza o triangulacích s minimálním stupněm větším než tři

V této sekci zkusíme popsat nějaké grafy, které s libovolně předepsanými obsahy stěn nakreslit nelze.

3.2.1. Hypotéza. (Kratochvíl [Kra09])

Triangulace s minimálním stupněm větším než tři nelze nakreslit s libovolně předepsanými obsahy stěn.

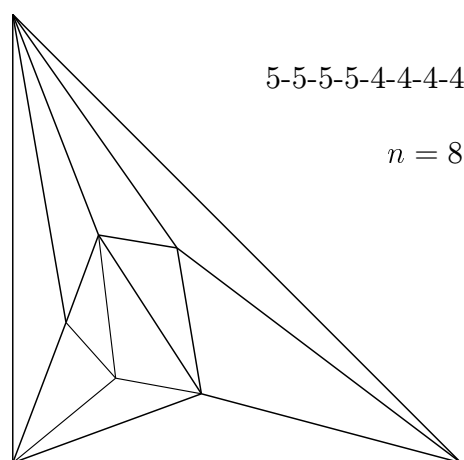
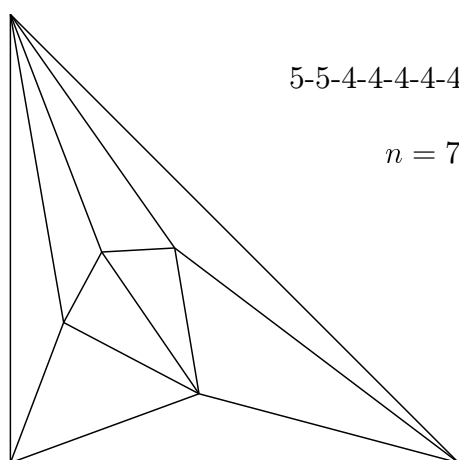
Nyní se pokusíme ověřit platnost hypotézy pro všechny triangulace s minimálním stupněm alespoň čtyři, které nemají víc než deset vrcholů. Samotnému hledání těchto grafů se věnovat nebudeme a k jejich vygenerování použijeme program `plantri`, který je popsán v článku od autorů programu [BM07].

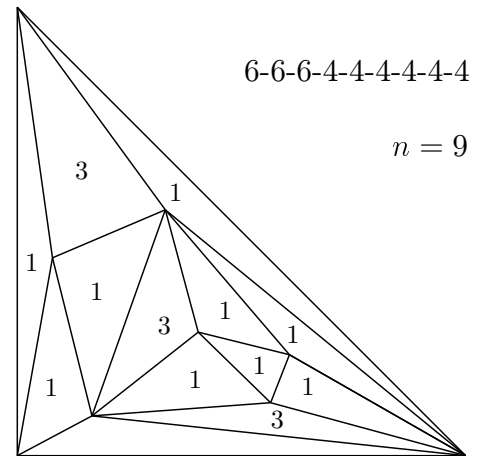
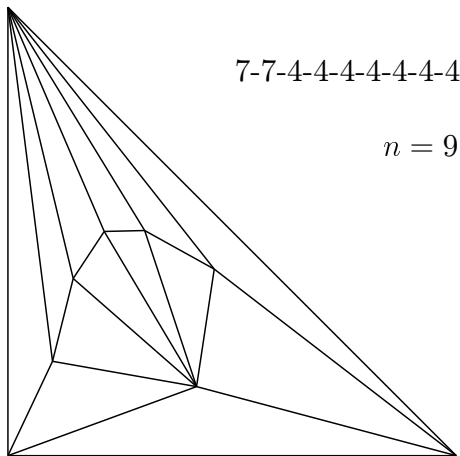
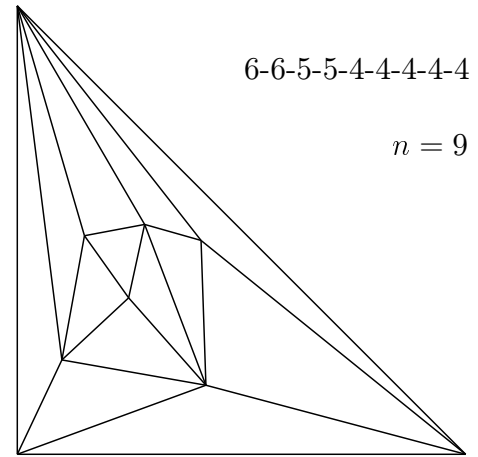
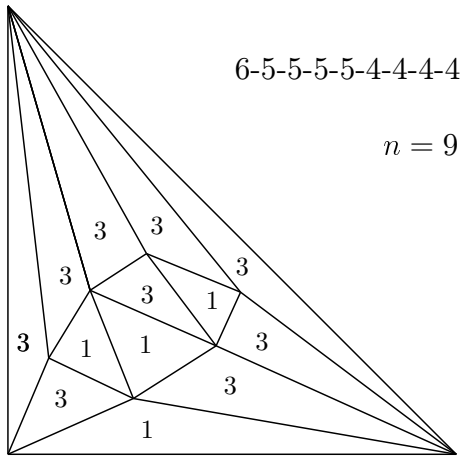
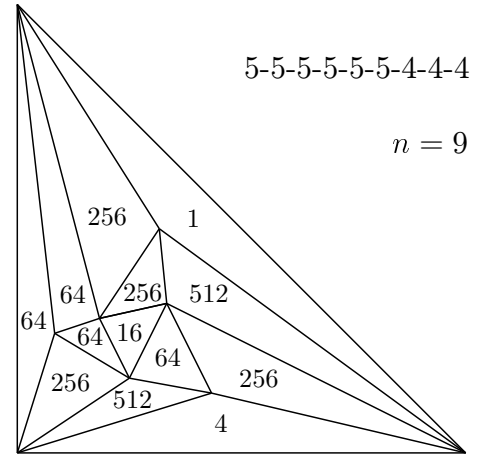
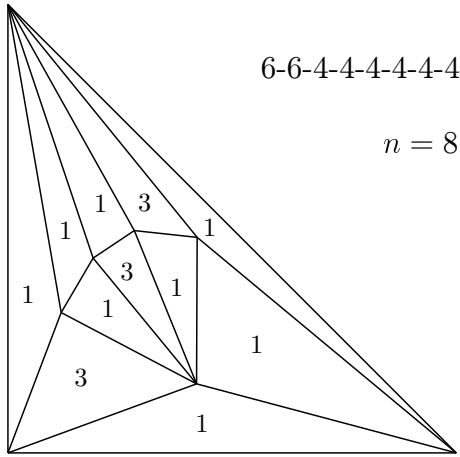
Z výstupu tohoto programu vygenerujeme soustavy rovnic pro Mathematicu (budeme je sestavovat stejným způsobem, jako rovnice pro osmistěn v předchozí kapitole) a pokusíme se těmito rovnicím hledat pravé strany tak, aby neměly žádné reálné řešení. U větších grafů se bohužel budeme muset spolehnout pouze na numerické řešení těchto soustav, protože nalezené Gröbnerovy báze jsou tak složité, že by Mathematica jejich symbolické řešení nenašla v rozumném čase.

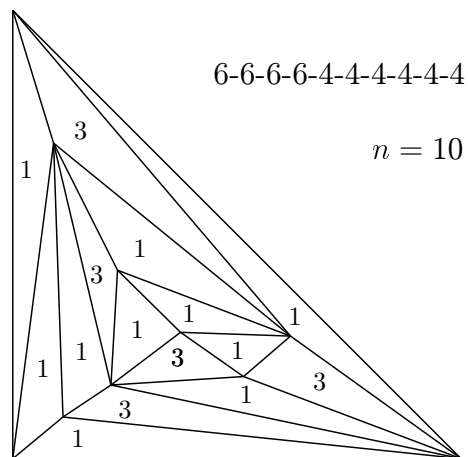
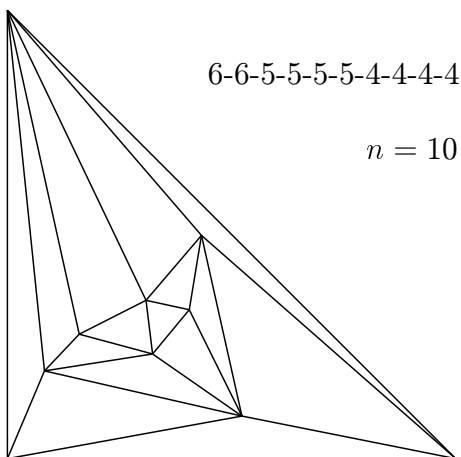
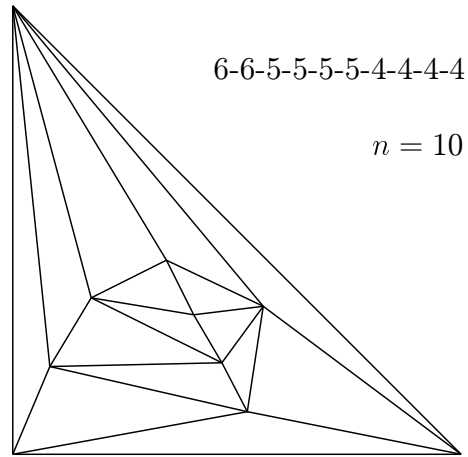
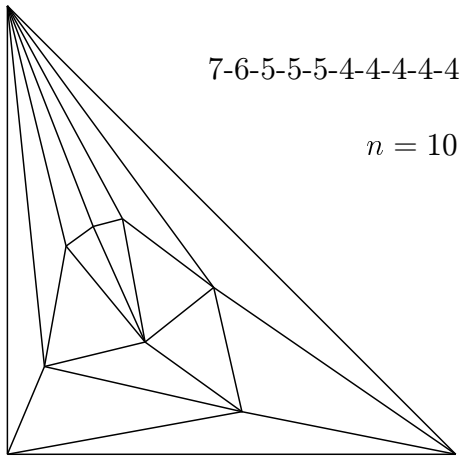
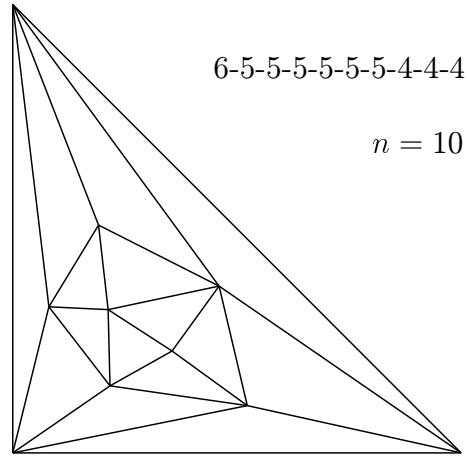
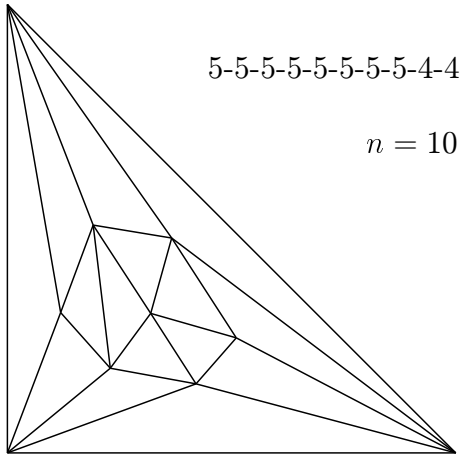
Samotné hledání pravých stran většinou ponecháme náhodě. U některých grafů lze jednoduše nahlédnout, že pokud některé obsahy předepíšeme „malé“ (dáme jim hodnotu 1) a jiné „velké“ (ukáže se, že si vystačíme s hodnotou 3), nakreslení zřejmě existovat nebude, což pak snadno ověříme. U těch, u kterých intuice selhává, prostě vyzkoušíme velké množství náhodných velikostí a budeme doufat, že hledané ohodnocení dříve či později najdeme.

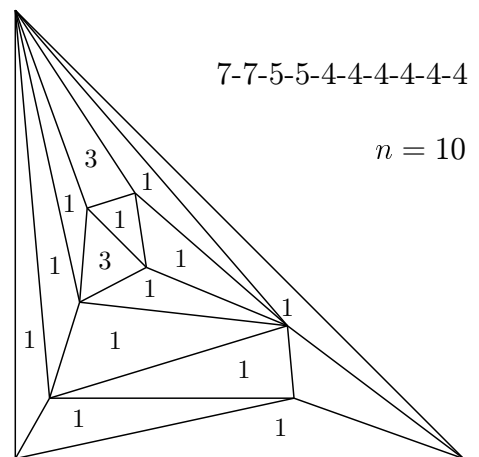
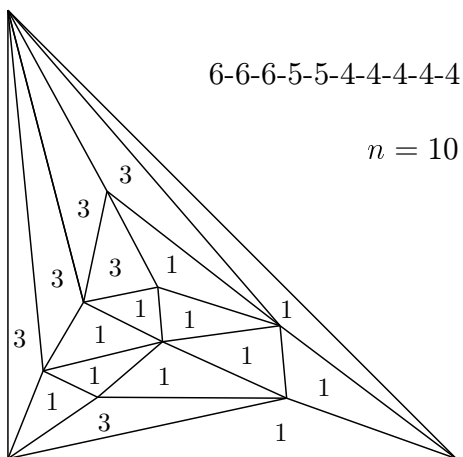
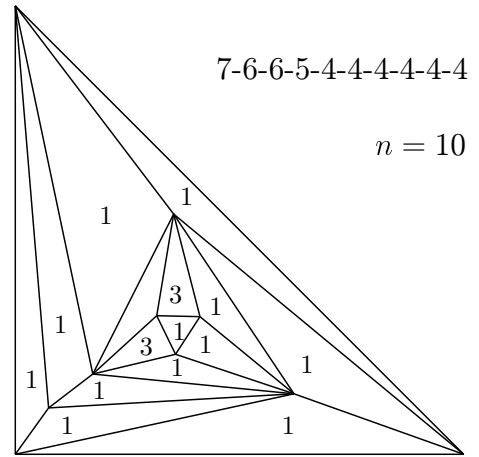
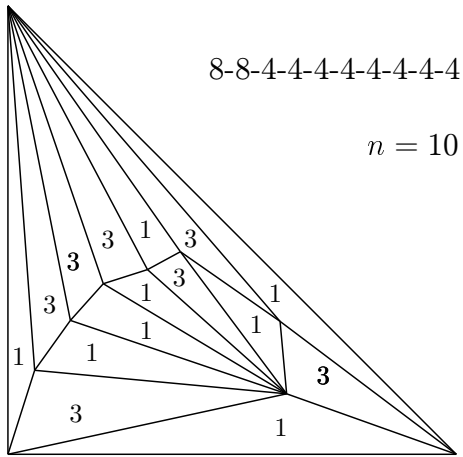
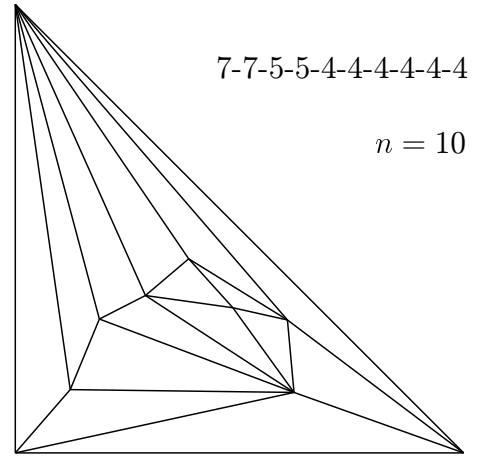
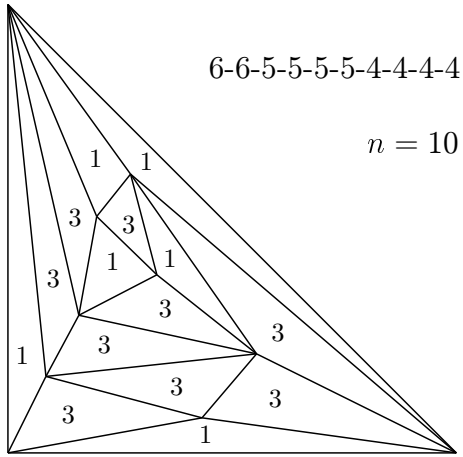
Pokud předepisujeme všem stěnám jen „malé“ a „velké“ obsahy, lze u menších grafů vyzkoušet všechny možnosti. Již u triangulace na sedmi vrcholech (a u mnoha dalších) nám to ale nepomůže. Příliš si nepomůžeme ani přidáním dalších hodnot – vypadá to, že hledané ohodnocení buď najdeme velmi rychle, a nebo ani po několikaletém výpočtu.

Na následujících obrázcích si můžeme prohlédnout všechny zkoumané triangulace, kromě osmistěnu, kterým jsme se zabývali již dříve. Pokud jejich stěnám umíme přiřadit takové obsahy, aby nakreslení neexistovalo, jsou tyto obsahy v obrázcích uvedeny. U obrázků jsou též uvedeny počty vrcholů grafů a jejich skóre.





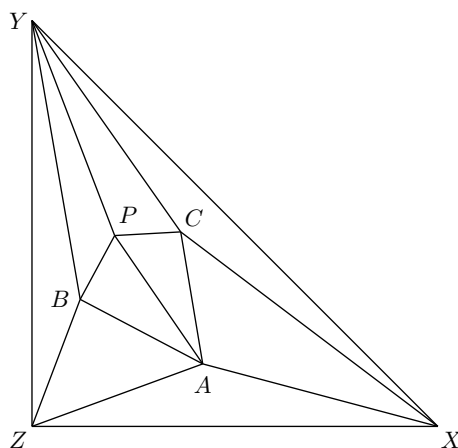




3.3. Vyvrácení hypotézy

Nyní se pořádněji podíváme na nejmenší triangulaci, pro kterou se nám nepodařilo předepsat obsahy stěn tak, aby nešla nakreslit. Najdeme ji na obrázku 6.

Budeme volit souřadnice bodů A, B, C, P, X, Y, Z tak, abychom postupně splnili všechny požadavky na předepsané obsahy stěn nakreslení dané devíti parametry $S_{ABZ}, S_{APB}, S_{AXC}, S_{AZX}, S_{BYZ}, S_{CXY}, S_{BPY}, S_{PCY}, S_{ACP}$. Polohy bodů budou obvykle ihned určeny jednoznačně, jedinou výjimku tvoří x -ová souřadnice bodu A . Tu si označíme jako neznámou a a nakonec dokážeme, že její hodnotu můžeme zvolit tak, aby všechny trojúhelníky měly požadovaný obsah.



Obr. 6: Zkoumaná triangulace

Body X, Y, Z : Jak už jsme ukázali v 2.3, můžeme bez újmy na obecnosti přidat některá dodatečná omezení. Zadané obsahy lze pomocí lineárního regulárního zobrazení změnit a zaručit přitom, že pokud existuje nějaké nakreslení se změněnými obsahy, bude existovat i nakreslení s obsahy původními. Body X, Y, Z si tedy zafixujeme na souřadnicích $[1, 0], [0, 1], [0, 0]$, součet všech předepsaných obsahů potom bude obsah trojúhelníka XYZ , tedy $1/2$.

Obsah trojúhelníka AZX , bod A : Souřadnice x bodu A (budeme ji značit a_x) označíme jako neznámou a . Protože obsah trojúhelníka je polovina součinu délky libovolné jeho strany a výšky na tu stranu a strana ZX má délku 1, je a_y rovna $2S_{AZX}$.

Obsah trojúhelníků BYZ, ABZ , bod B : Souřadnice b_x je z téhož důvodu rovna $2S_{BYZ}$. Souřadnici b_y získáme vyjádřením z rovnice

$$\begin{vmatrix} a_x & a_y \\ b_x & b_y \end{vmatrix} = 2S_{ABZ},$$

je tedy rovna $2(S_{ABZ} + S_{AZX}S_{BYZ})/a$.

Obsah trojúhelníků AXC, CXY , bod C : Poloha bodu C závisí na obsazích S_{AXC} a S_{CXY} , které nám určují jeho vzdálenost od přímek XY a AX . Bod C totiž leží na přímce rovnoběžné s XY , od které má vzdálenost $2S/|XY|$. Stejně můžeme vyjádřit i jeho vzdálenost od přímky AX .

Protože podobný výpočet budeme opakovat, zkusíme řešit problém obecně. Nejprve si připravíme výraz, který vyjadřuje hodnotu y -ové souřadnice bodu ležícího na přímce pomocí souřadnice x -ové a souřadnic dvou bodů, které určují polohu této přímky. Potom najdeme výraz, který určuje přímku k původní přímce rovnoběžnou, se vzdáleností určenou daným obsahem trojúhelníka. Pokud se nám podaří takto vyjádřit souřadnice bodu pomocí dvou různých přímek, stačí již jen vyřešit soustavu dvou rovnic.

Máme-li přímkou, která prochází nějakými body K a L a t je x -ová souřadnice nějakého bodu na ní, můžeme y -ovou souřadnici vyjádřit jako $(t - k_x)\frac{l_y - k_y}{l_x - k_x} + k_y$. (Nejprve posuneme t do počátku, vynásobíme jej směrnici a posuneme v ose y .) Pokud chceme najít přímkou takovou, aby pro každý bod M , který na ní leží, měl trojúhelník KLM obsah S , musíme k y -ové souřadnici přičíst ještě $2S/(k_x - l_x)$ (a dbát přitom na správnou orientaci obsahu, abychom neporušili rovinnost nakreslení). Proč zrovna tolik? Uvažujme následovně:

Nechť přímkou $K'L'$ vznikne posunutím přímky KL v y -ovém směru o nějakou vzdálenost d . Obě přímky sdílejí směrový vektor $\mathbf{s} = (l_x - k_x, l_y - k_y)$ a normálový vektor $\mathbf{n} = (l_y - k_y, k_x - l_x)$. Vzdálenost $K'L'$ od KL bude rovna velikosti kolmého průmětu vektoru $(0, d)$ do směru normály. Tu můžeme vyjádřit jako skalární součin $(0, d) \cdot \mathbf{n} / \|\mathbf{n}\| = d \cdot (k_x - l_x) / \|\mathbf{n}\|$. Obsah trojúhelníka KLM získáme jako součin této vzdálenosti s hodnotou $\frac{1}{2}|KL| = \frac{1}{2}\|\mathbf{s}\|$. Nyní stačí zvolit d tak, aby obsah byl roven požadovanému S . Musí tedy platit $\frac{1}{2}\|\mathbf{s}\| \cdot d \cdot (k_x - l_x) / \|\mathbf{n}\| = S$. Protože vektory \mathbf{s} a \mathbf{n} mají stejné normy, rovnici zjednodušíme na $d \cdot (k_x - l_x) = 2S$.

Zbývá ještě rozhodnout se o směru posunutí – my si vybereme ten, který odpovídá obrázku 6. Je sice možné, že existují nějaká další řešení, i když se rozhodneme pro posunutí opačným směrem, nám ale stačí ukázat existenci jediného. Ukážeme tedy, že řešení odpovídající obrázku existuje vždy.

Pokud do výrazu

$$(t - k_x)\frac{l_y - k_y}{l_x - k_x} + k_y + \frac{2S}{k_x - l_x}$$

za souřadnice bodů K, L dosadíme souřadnice bodů X, Y a položíme jej rovný témuž výrazu s dosazenými souřadnicemi A, X , získáme následující rovnici:

$$1 - t - 2S_{CXY} = \frac{2S_{AXC}}{1 - a} + 2S_{AZX} - \frac{2(-a + t)S_{AZX}}{1 - a}.$$

Z té již snadno vyjádříme t , což je x -ová souřadnice bodu C , a posléze dopočítáme y -ovou. Bod C má tedy souřadnice

$$\left[\frac{-1 + a + 2S_{AXC} + 2S_{AZX} + 2S_{CXY} - 2aS_{CXY}}{-1 + a + 2S_{AZX}}, 1 - 2S_{CXY} - c_x \right].$$

Obsahy trojúhelníků BPY , ABP , bod P : Stejným způsobem vyjádříme i souřadnice bodu P pomocí obsahů trojúhelníků BPY a APB . Výpočet ani jejich hodnotu zde ale nevedeme, protože se složitostí výpočtu značně narůstá i jeho délka.

Obsahy trojúhelníků PCY , ACP : Nyní nám zbývá sestavit rovnici pro a a ukázat, že její řešení vždy existuje. Splníme obsah trojúhelníka PCY :

$$2S_{PCY} = p_x(c_y - 1) - c_x(p_y - 1),$$

tuto rovnici po dosazení všech souřadnic můžeme vidět níže.

Zbývá nám již jen S_{ACP} . Tento zbývající obsah je ale nyní již splněn automaticky, protože jej můžeme vyjádřit pomocí těch ostatních. Z volby poloh bodů X , Y , Z totiž víme, že součet všech obsahů je $1/2$.

Získaná rovnice pro a je bohužel poměrně dlouhá, po úpravě vypadá takto:

$$\begin{aligned} & a^3(1 - 2S_{BPY} - 2S_{BYZ} - 2S_{PCY} - 2S_{CXY}) \\ & + a^2(-4S_{ABZ} - 2S_{APB} + 2S_{AXC} + 2S_{AZX} + 2S_{BPY} - 4S_{AXC}S_{BPY} - 8S_{AZX}S_{BPY} \\ & \quad + 4S_{ABZ}S_{BYZ} + 4S_{APB}S_{BYZ} - 4S_{AXC}S_{BYZ} - 8S_{AZX}S_{BYZ} + 4S_{BPY}S_{BYZ} \\ & \quad + 4S_{BYZ}^2 + 2S_{PCY} + 4S_{ABZ}S_{PCY} - 4S_{AZX}S_{PCY} + 4S_{BYZ}S_{PCY} + 2S_{CXY} \\ & \quad + 8S_{ABZ}S_{CXY} + 4S_{APB}S_{CXY} + 4S_{BYZ}S_{CXY} - 1) \\ & + a(4S_{ABZ} + 4S_{ABZ}^2 + 2S_{APB} + 4S_{ABZ}S_{APB} - 8S_{ABZ}S_{AXC} - 4S_{APB}S_{AXC} \\ & \quad - 8S_{ABZ}S_{AZX} - 4S_{APB}S_{AZX} + 4S_{ABZ}S_{BPY} + 4S_{AZX}S_{BPY} \\ & \quad - 8S_{AZX}S_{BPY} + 2S_{BYZ} - 4S_{APB}S_{BYZ} - 4S_{AXC}S_{BYZ} + 8S_{ABZ}S_{AXC}S_{BYZ} \\ & \quad + 8S_{APB}S_{AXC}S_{BYZ} + 16S_{ABZ}S_{AZX}S_{BYZ} + 16S_{APB}S_{AZX}S_{BYZ} \\ & \quad - 8S_{AXC}S_{AZX}S_{BYZ} - 8S_{AZX}^2S_{BYZ} - 4S_{BPY}S_{BYZ} + 8S_{AXC}S_{BPY}S_{BYZ} \\ & \quad + 16S_{AZX}S_{BPY}S_{BYZ} - 4S_{BYZ}^2 + 8S_{AXC}S_{BYZ}^2 + 16S_{AZX}S_{BYZ}^2 \\ & \quad - 4S_{ABZ}S_{PCY} + 8S_{ABZ}S_{AZX}S_{PCY} - 4S_{BYZ}S_{PCY} + 8S_{AZX}S_{BYZ}S_{PCY} \\ & \quad - 8S_{ABZ}S_{CXY} - 8S_{ABZ}^2S_{CXY} - 4S_{APB}S_{CXY} - 8S_{ABZ}S_{APB}S_{CXY} \\ & \quad - 8S_{ABZ}S_{BPY}S_{CXY} - 8S_{AZX}S_{BPY}S_{CXY} - 4S_{BYZ}S_{CXY} - 8S_{ABZ}S_{BYZ}S_{CXY} \\ & \quad - 8S_{AZX}S_{BYZ}S_{CXY} - 8S_{AXC}S_{AZX}S_{BPY}) \\ & + 8S_{ABZ}^2S_{CXY} + 8S_{ABZ}S_{APB}S_{CXY} + 8S_{ABZ}S_{BPY}S_{CXY} + 8S_{ABZ}S_{BYZ}S_{CXY} \\ & \quad + 16S_{ABZ}S_{AZX}S_{BYZ}S_{CXY} + 16S_{APB}S_{AZX}S_{BYZ}S_{CXY} \\ & \quad + 16S_{AZX}S_{BPY}S_{BYZ}S_{CXY} + 16S_{AZX}S_{BYZ}^2S_{CXY} - 4S_{ABZ}^2 - 4S_{ABZ}S_{APB} \\ & \quad + 8S_{ABZ}^2S_{AZX} + 8S_{ABZ}S_{APB}S_{AZX} - 4S_{ABZ}S_{BPY} + 8S_{ABZ}S_{AXC}S_{BPY} \\ & \quad + 8S_{ABZ}S_{AZX}S_{BPY} - 4S_{ABZ}S_{BYZ} + 8S_{ABZ}S_{AXC}S_{BYZ} - 8S_{APB}S_{AZX}S_{BYZ} \\ & \quad + 16S_{ABZ}S_{AXC}S_{AZX}S_{BYZ} + 16S_{APB}S_{AXC}S_{AZX}S_{BYZ} + 16S_{ABZ}S_{AZX}^2S_{BYZ} \\ & \quad + 16S_{APB}S_{AZX}^2S_{BYZ} - 8S_{AZX}S_{BPY}S_{BYZ} + 16S_{AXC}S_{AZX}S_{BPY}S_{BYZ} \\ & \quad + 16S_{AZX}^2S_{BPY}S_{BYZ} - 8S_{AZX}S_{BYZ}^2 + 16S_{AXC}S_{AZX}S_{BYZ}^2 + 16S_{AZX}^2S_{BYZ}^2 \\ & \quad + 8S_{ABZ}^2S_{AXC} + 8S_{ABZ}S_{APB}S_{AXC} \\ & = 0. \end{aligned}$$

Ukážeme, že tato rovnice má pro jakoukoliv volbu parametrů alespoň jeden kořen v intervalu $[0, 1]$. Dosadíme-li za a nulu a upravíme, dostaneme výraz

$$4(S_{ABZ} + S_{APB} + S_{BPY} + S_{BYZ})(S_{ABZ} + 2S_{AZX}S_{BYZ})(-1 + 2S_{AXC} + 2S_{AZX} + 2S_{CXY}),$$

jehož první dvě závorky jsou vždy kladné a poslední je vždy záporná, protože součet všech obsahů je $1/2$. Polynom je tedy v nule záporný.

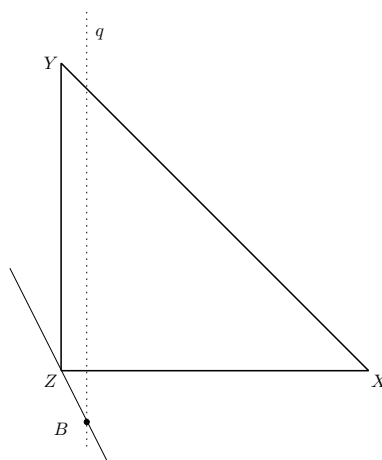
Pokud za a dosadíme jedničku, dostaneme podstatně ošklivější polynom:

$$\begin{aligned}
& S_{AXC} - 8S_{ABZ}S_{AXC} + 8S_{ABZ}^2S_{AXC} - 4S_{APB}S_{AXC} + 8S_{ABZ}S_{APB}S_{AXC} + 2S_{AZX} \\
& - 8S_{ABZ}S_{AZX} + 8S_{ABZ}^2S_{AZX} - 4S_{APB}S_{AZX} + 8S_{ABZ}S_{APB}S_{AZX} - 4S_{AXC}S_{BPY} \\
& + 8S_{ABZ}S_{AXC}S_{BPY} - 4S_{AZX}S_{BPY} + 8S_{ABZ}S_{AZX}S_{BPY} - 8S_{AXC}S_{AZX}S_{BPY} \\
& - 8S_{AZX}^2S_{BPY} - 8S_{AXC}S_{BYZ} + 16S_{ABZ}S_{AXC}S_{BYZ} + 8S_{APB}S_{AXC}S_{BYZ} \\
& - 8S_{AZX}S_{BYZ} + 16S_{ABZ}S_{AZX}S_{BYZ} + 8S_{APB}S_{AZX}S_{BYZ} \\
& - 8S_{AXC}S_{AZX}S_{BYZ} + 16S_{ABZ}S_{AXC}S_{AZX}S_{BYZ} + 16S_{APB}S_{AXC}S_{AZX}S_{BYZ} \\
& - 8S_{AZX}^2S_{BYZ} + 16S_{ABZ}S_{AZX}^2S_{BYZ} + 16S_{APB}S_{AZX}^2S_{BYZ} + 8S_{AXC}S_{BPY}S_{BYZ} \\
& + 8S_{AZX}S_{BPY}S_{BYZ} + 16S_{AXC}S_{AZX}S_{BPY}S_{BYZ} + 16S_{AZX}^2S_{BPY}S_{BYZ} \\
& + 8S_{AXC}S_{BYZ}^2 + 8S_{AZX}S_{BYZ}^2 + 16S_{AXC}S_{AZX}S_{BYZ}^2 + 16S_{AZX}^2S_{BYZ}^2 \\
& - 4S_{AZX}S_{PCY} + 8S_{ABZ}S_{AZX}S_{PCY} + 8S_{AZX}S_{BYZ}S_{PCY} - 8S_{AZX}S_{BPY}S_{CXY} \\
& - 8S_{AZX}S_{BYZ}S_{CXY} + 16S_{ABZ}S_{AZX}S_{BYZ}S_{CXY} + 16S_{APB}S_{AZX}S_{BYZ}S_{CXY} \\
& + 16S_{AZX}S_{BPY}S_{BYZ}S_{CXY} + 16S_{AZX}S_{BYZ}^2S_{CXY}.
\end{aligned}$$

Hledat minimum tohoto polynomu je bohužel příliš náročné i pro Mathematicu, poslední krok důkazu tedy budeme muset opřít jen o numerický výpočet pomocí funkce `NMinimize`. Omezíme-li všechny parametry tak, aby ležely v intervalu $[0, 1/2]$ a jejich součet byl neostře menší než $1/2$, spočítá Mathematica minimum 0 pro $S_{CXY} = 0.5$ a ostatní obsahy nulové.

Protože polynom je spojitá funkce, máme-li interval, na jehož jednom okraji je záporná a na druhém kladná, musí v něm někde nabývat hodnoty 0. Rovnice má tedy v intervalu $[0, 1]$ vždy alespoň jedno řešení.

Zbývá ještě dokázat, že nalezené řešení má vždy geometrický smysl. Využijeme toho, že obsahy trojúhelníků jsou orientované, a ukážeme, že pokud řešení smysl nemá, vždy je alespoň jeden trojúhelník špatně orientován.

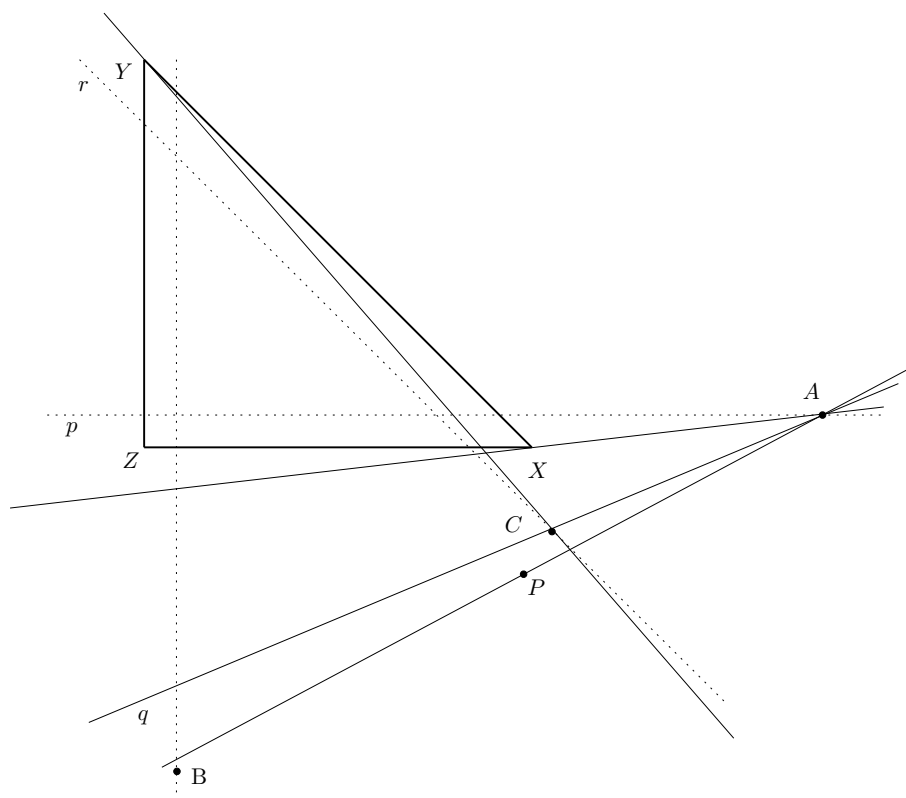


Obr. 7: B nemůže ležet vpravo od přímky ZX

Nejprve rozebereme případy, kdy body A , B nebo C opustí trojúhelník XYZ . Bod A leží na přímce p rovnoběžné se ZX ve směru vlevo od ní a zároveň vpravo od přímky ZY (hovoříme-li o směrech vlevo a vpravo od přímky KL , chápeme přímku jako orientovanou z bodu K do bodu L). Bod B se nachází na přímce q rovnoběžné se ZY , vpravo od ní. Bod C je na přímce r rovnoběžné s XY , vlevo od ní.

Nejdříve ukážeme (viz obrázek 7), že bod B nemůže ležet vpravo od přímky ZX . Pokud tomu tak je, máme-li zachovat orientaci trojúhelníku ABZ , musí bod A ležet vlevo od přímky BZ . To by ale znamenalo, že bod A leží také vlevo od přímky ZY , což je spor s předpokladem, že leží vpravo (a nebo vpravo od ZX).

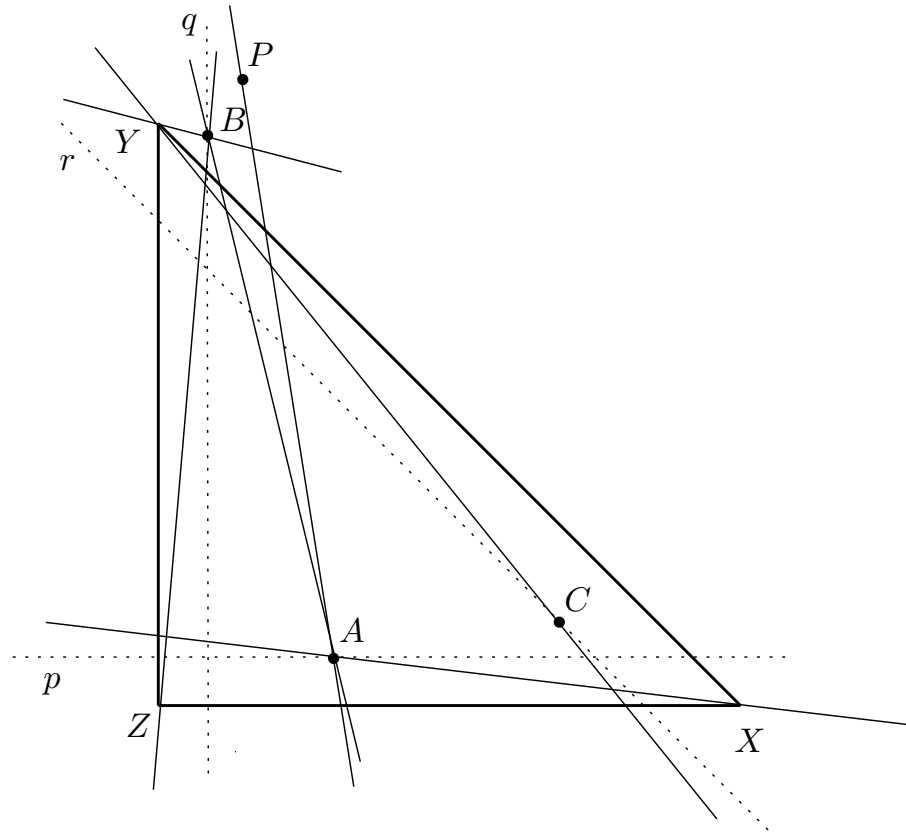
Nyní předvedeme (viz obrázek 8), že bod A nemůže ležet vpravo od přímky XY . Předpokládejme, že tomu tak je. Potom bod C , máme-li splnit orientaci trojúhelníka AXC , musí ležet vpravo od přímky XA (tedy i vpravo od přímky ZX a mimo trojúhelník XYZ). Z orientace trojúhelníka ACP potom plyne, že bod P leží vlevo od přímky AC , z orientace PCY potom, že P leží vlevo od přímky CY . Protože podle orientace trojúhelníka APB bod B leží vpravo od přímky PA , musí bod B nutně ležet též vpravo od přímky ZX . To ale, jak už jsme ukázali, není možné.



Obr. 8: A nemůže ležet vpravo od přímky ZX

Ukázat, že bod C nemůže ležet vpravo od přímky ZX , nyní bude jednoduché. Stačí si všimnout, že máme-li splnit orientaci trojúhelníka AXC , bod A musí ležet vpravo od přímky XY , což, jak již víme, nemůže.

Bod B nemůže ležet vpravo od přímky XY (viz obrázek 9). Předpokládejme, že je tomu naopak. Podle orientace ABZ bude bod A ležet vpravo od přímky ZB . Orientace APB a BPY vynutí polohu bodu P vpravo od přímky AB a zároveň vlevo od přímky YB . Bod C se potom nachází vpravo od přímky AP (podle ACP) a vpravo od XA (podle AXC). Bod C přitom nemůže ležet vlevo od ZY , protože tím by vynutil polohu bodu A vpravo od XY . Bod P se potom tedy nachází vpravo od CY , ačkoliv se měl podle orientace PCY nacházet vlevo.

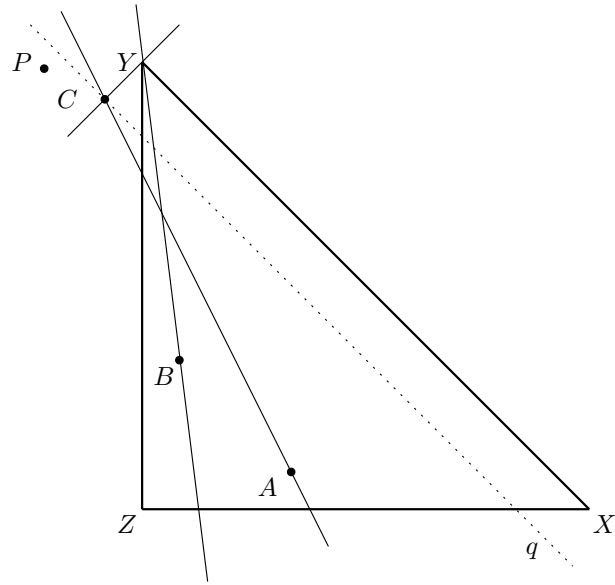


Obr. 9: B nemůže ležet vpravo od přímky XY

Bod C nemůže ležet ani vlevo od přímky ZY (viz obrázek 10). Pokud tomu tak je, podle orientace APC bod P leží vlevo od AC a podle PCY vlevo od CY . To ovšem znamená, že leží i vlevo od přímky BY , ačkoliv mu orientace PBY předepisuje, že má ležet vpravo od ní.

Pokud žádný z bodů A , B , C neopustí trojúhelník XYZ a výsledek přesto neodpovídá požadovanému rovinnému nakreslení, ukázat, že vždy není dodržena orientace alespoň jednoho z trojúhelníků, je již jednoduché.

Dokázali jsme tedy, že pro daný graf existuje rovinné nakreslení s libovolnými předepsanými obsahy stěn.



Obr. 10: C nemůže ležet vlevo od přímky ZY

4. Závěr

V této práci jsme se snažili charakterizovat rovinné grafy, které lze a nebo naopak nelze nakreslit s libovolně předepsanými obsahy stěn. Nejprve jsme ukázali, že existuje rovinný graf, který není možné nakreslit tak, aby všechny jeho stěny měly stejný obsah. Potom jsme ukázali třídu grafů, kterou lze nakreslit se všemi předepsanými obsahy stěn. Nakonec jsme se pokoušeli ověřit hypotézu, že všechny triangulace s minimálním stupněm větším než tři není možné nakreslit s libovolně předepsanými obsahy stěn. Tuto hypotézu jsme vyvrátili.

A. Programy na CD

Na přiloženém CD se nachází program na generování nakreslení s (téměř) stejnými obsahy genetickým algoritmem a sada skriptů, pomocí které jsme získali informace o triangulacích do deseti vrcholů. Protože jde spíš jen o pomocné nástroje než o skutečný software, nebyla při jejich vytváření prioritou pohodlnost použití.

A.1. Program na generování nakreslení

V adresáři `ouce` se nachází program, pomocí kterého jsme se snažili odhadnout existenci nakreslení jednotlivých grafů. Algoritmus byl již popsán v 2.1 a jeho podrobné vnitřní fungování dokumentují četné komentáře ve zdrojovém kódu, popíšeme si tedy jen, jak jej používat.

Program očekává na standardním vstupu nějaké nakreslení grafu a jeho kombinatorickou strukturu v následující formě:

Nejprve čte seznam vrcholů grafu, při čemž každý vrchol reprezentován jedním řádkem po sobě jdoucích čísel oddělených mezerami. Nejprve je uvedena x -ová a y -ová souřadnice obrazu vrcholu, potom následuje seznam jeho sousedů reprezentovaných jako čísla odpovídající číslům řádků vstupu, číslovány jsou ovšem od nuly.

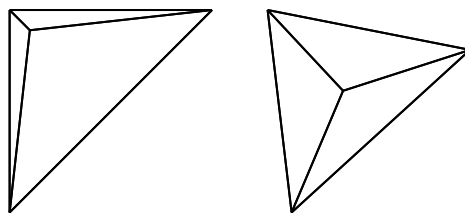
Následuje prázdný řádek a za ním kombinatorická struktura grafu v podobě seznamu všech trojúhelníků. Každý řádek se skládá ze tří čísel odpovídajících číslům vrcholů na hranici stěny, oddělených mezerami.

Takto by mohly vypadat vstupní údaje pro čtyřstěn (nakreslení viz obrázek 11):

```
0 0 1 2 3
0 10 0 2 3
10 0 0 1 3
1 1 0 1 2

0 1 3
0 2 3
1 2 3
```

Tento a nějaké další testovací vstupy se nacházejí v podadresáři `ouce/input`.



Obr. 11: Čtyřstěn, vstupní nakreslení a jeden z výstupů pro uvedený vstup.

Za svého běhu program generuje obrázky ve formátu PDF, v kterých je sto nejlepších nakreslení dané generace a textové soubory s jejich hodnoceními a dalšími informacemi. Četnost generování těchto souborů, množství generací, po které program poběží, a také hodnotu random seed (číslo, kterým je inicializován generátor pseudonáhodných čísel), je možno ovlivnit argumenty zadávanými z příkazové řádky.

První argument, který program čte, je počet generací, po které má program běžet. Pokud zadáme hodnotu 0, běží program tak dlouho, dokud jej neukončíme sami. Druhý argument říká, kolik generací se má přeskakovat při generování výstupu – sice tak získáme méně informací o jeho běhu, ale zase běží mnohem rychleji. Nastavíme-li tuto hodnotu na 0, bude program vypisovat údaje o nové generaci tehdy, pokud se v ní nachází nakreslení s lepším ohodnocením, než jakékoliv, které jsme zatím vygenerovali. Třetí argument je hodnota random seed.

Při spuštění bez argumentů program běží po tisíc generací, informace o každé ukládá do souboru a používá seed načtený ze zařízení `/dev/urandom`. Typické použití, při kterém necháme program běžet tak dlouho, dokud se děje něco zajímavého, a zabýváme se jen nejlepšími nakresleními, je spustit program s argumenty `0 0`.

Pokud chceme nechat program kreslit jen jeden velký obrázek místo stovky malých, můžeme tak učinit ve zdrojovém kódu nastavením hodnoty `MAX_SHEEP_DRAW` na 1. Maximální počet vrcholů grafu, s kterým si program poradí, je 100. Toto omezení bychom sice mohli odstranit, ale pro velké grafy by program stejně běžel tak dlouho (a generoval tak nepřehledné obrázky), že to nemá význam.

Pro instalaci programu je třeba mít nainstalován Linux (nebo jiný operační systém se zařízením `/dev/urandom`), kompilátor jazyka C, program `make` a knihovnu Cairo včetně hlavičkových souborů.

A.2. Triangulace do deseti vrcholů

Sada skriptů přiložená na CD v adresáři `scripts` zpracovává výstup z programu `plantri` (volaný jako `plantri -m4 -a N`, kde N nabývalo hodnot 6–10) na textové soubory, které slouží jako vstup programu Mathematica, který pro jejich používání potřebujeme mít nainstalovaný. Kromě toho taky potřebujeme funkční instalaci `perlu`.

Výstup tohoto programu se nachází v souboru `triangulations`. Na každém řádku je popsán graf, nejprve jeho počet vrcholů, potom pro jeho jednotlivé vrcholy seznam sousedů po směru hodinových ručiček. Tím je dána kombinatorická struktura grafu. Pokud výstup z `plantri` předáme na standardní vstup skriptu `triangles`, dostaneme pro každý graf jeden textový soubor se soustavou rovnic, po jejichž vyřešení známe souřadnice obrazů vrcholů v nakreslení se stejně velkými obsahy stěn pro Mathematicu. Podle nalezených řešení nám pak Mathematica nakreslí i příslušná nakreslení. Textové soubory vygenerované pro naše triangulace se nacházejí v podadresáři `equations`.

Soubor `oracle` obsahuje podobný skript, který čte stejný vstup, ale tentokrát generuje textové soubory se vstupem pro Mathematicu tak, aby nám odpovídala jen hodnotami `True` nebo `False` podle toho, jestli nějaké nakreslení pro dané ohodnocení stěn existuje nebo ne. Soubory vygenerované tímto skriptem se nacházejí v podadresáři `guess`.

Konečně, dvojice skriptů `mighty_oracle` a `mightier_oracle` čte na standardním vstupu některý z výstupů skriptu `oracle`, generuje náhodná ohodnocení pravic stran a volá Mathematicu tak dlouho, dokud nenarazí na nějaké ohodnocení, pro které neexistuje nakreslení, a to pak vypíše. Skripty se liší jen tím, že první přiřazuje pravým stranám pouze hodnoty 1 nebo 100, kdežto druhý hodnoty $1, 2, 4, \dots, 2^n$, kde n je počet stěn grafu.

A.3. Graf z obrázku 6

Při dokazování, že graf na obrázku 6 má nakreslení pro všechny předepsané obsahy stěn, jsme v programu Mathematica vytvořili vizualizaci, která nám umožňuje předepsat nějaké obsahy a zkoumat, jak se při posouvání bodu A po přímce, na které musí ležet, mění poloha bodů B , C a přímek, na jejichž společném průsečíku musí ležet bod P .

Tato vizualizace se nachází na CD v souboru `vizualizace.nb` a po otevření v Mathematice se s ní zachází následovně: Nejprve nastavíme libovolné hodnoty proměnným r_{ABZ} až r_{APJ} , potom necháme výpočet vyhodnotit (například stisknutím kláves `Shift` a `Enter`). Ve spodní části dokumentu se objeví obrázek s posuvníkem, jehož posouváním pohybujeme x -ovou souřadnicí bodu A v intervalu $[0, 1]$. Můžeme sledovat, jak se mění polohy bodů a přímek. Pokud se v pravé části obrázku nacházejí dvě jedničky, je čtyřúhelník $ABCY$ konvexní, jinak nikoliv.

B. Seznam literatury

- [BM04] J. Boyer a W. Myrvold. On the cutting edge: Simplified $O(n)$ planarity by edge addition. *Journal of Graph Algorithms and Applications*, 8(3):241–273, 2004.
- [BM07] G. Brinkmann a B. McKay. Fast generation of planar graphs. *Communications in Mathematical and in Computer Chemistry/MATCH*, 58(2):323–357. Citeseer, 2007.
- [DBETT99] G. Di Battista, P. Eades, R. Tamassia, a I. G. Tollis. *Graph Drawing*. Prentice Hall, 1999.
- [Die06] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, third edition, 2006.
- [Hec97] A. Heck. Bird’s-eye view of Grobner bases. *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment*, 389(1-2):16–21. Elsevier, 1997.
- [Kra09] J. Kratochvíl. Osobní komunikace, 2009.
- [MN07] J. Matoušek a J. Nešetřil. *Kapitoly z diskrétní matematiky*. Karolinum, 2007.
- [Sch90] W. Schnyder. Embedding planar graphs on the grid. In *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*, pages 138–148. Society for Industrial and Applied Mathematics Philadelphia, PA, USA, 1990.
- [TH74] R. E. Tarjan a J. Hopcroft. Efficient planarity testing. *Journal of the Association for Computing Machinery*, 21(4):549–568, 1974.
- [Tho92] C. Thomassen. Plane cubic graphs with prescribed face areas. *Combinatorics, Probability and Computing*, 1(04):371–381. Cambridge Univ Press, 1992.

