MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Zuzana Šimečková

Varianty Eberhardovy věty

Informatický ústav Univerzity Karlovy

Vedoucí bakalářské práce: doc. Mgr. Robert Šámal, Ph.D.
Studijní program: Informatika
Studijní obor: IOI

Praha 2018

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V .... dne .... Podpis autora

Poděkování.

Název práce: Varianty Eberhardovy věty

Autor: Zuzana Šimečková

Ústav: Informatický ústav Univerzity Karlovy

Vedoucí bakalářské práce: doc. Mgr. Robert Šámal, Ph.D., Informatický ústav Univerzity Karlovy

Abstrakt: Abstrakt.

Klíčová slova: klíčová slova

Title: Eberhard-Like Theorems

Author: Zuzana Šimečková

Institute: Computer Science Institute of Charles University

Supervisor: doc. Mgr. Robert Šámal, Ph.D., Computer Science Institute of Charles University

Abstract: Abstract.

Keywords: key words

Obsah

Úvod

Zkusíme-li upravovat Eulerův vzorec pro rovinné grafy $|V| - |E| + |S| = 2$, můžeme pro kubické grafy dospět do tvaru

$$\sum_{k\geq 3} (6-k)p_k = 12, \tag{1}$$

kde p_k značí počet k-hranných stěn grafu. Výraz (1) je nutnou podmínkou pro rovinnost kubického grafu. Pozoruhodně, počet šestihranných stěn v této podmínce nehraje žádnou roli. Toho si všiml Victor Eberhard, který v roce 1891 formuloval a dokázal, že pokud můžeme volit p_6, dokážeme najít graf zadaný počtem ostatních stěn, který je rovinný [1].

Věta 1 (Eberhardova věta)**.** *Pro každou posloupnost* $(p_k \mid 3 \leq k \neq 6)$ *kladných celých čísel, splňující* (1) *existuje taková hodnota* p_6*, že existuje rovinný 3-regulární 3-souvislý graf, který má právě* p_k *k-hranných stěn pro každé* $k \in \mathbb{N}, k \geq 3$.

Na větu navázali další a dnes je známá celá řada jejích obměn. Fisher dokázal silnější verzi Eberhardovy věty, kde počet potřebných stěn velikosti 6 shora omezil počtem ostatních stěn chtěného grafu [2]. Grünbaum představuje vlastní stručnější důkaz Eberhardovy věty a shrnuje výsledky podobného typu [3].

TODO protáhnout jestli bude třeba

DeVos a kol. [4] představuje obdobu Eberhardovy věty, kde místo p_6 můžeme volit p_5 a p_7. Článek také ukazuje, že konstrukce důkazu, kterou autoři použili, by mohla jít uplatnit i pro další velikosti stěn, které mohou splnit (1). V tomto článku nejprve zavedeme potřebnou terminologii a představíme zmiňovanou konstrukci důkazu, která pro dokončení potřebuje najít pomocné grafy. Cílem práce bylo tyto grafy získat. V závěrečných kapitolách navrhneme program, který bude potřebné grafy hledat, a nakonec představíme, pro které velikosti stěn jsme grafy získali a tedy dokončili důkaz dalších variant Eberhardovy věty.

1. Pojmy a definice

Dokažme nejprve, jak z Eulerova vzorce získáme (1). Pro každý kubický graf $G = (V,E)$ platí $3|V| = 2|E|$. Obecně platí, že počet stěn s můžeme přepsat následovně $s = \sum_{k\geq 3} p_k$ a pro rovinné grafy platí i $\sum_{k\geq 3} k \cdot p_k = 2|E|$, protože pokud pro každou stěnu započteme každou její hranu, získáme právě $2|E|$. Úpravou a dosazením do vzorce získáme požadovaný výraz:

$$\begin{aligned} |V| - |E| + |S| &= 2 \\ \frac{1}{3}|E| + s &= 2 \\ \frac{1}{6}\sum_{k\geq 3} k \cdot p_k - \sum_{k\geq 3} p_k &= 2 \\ \sum_{k\geq 3} (6-k)p_k &= 12 \end{aligned}$$

Čímž je důkaz dokončen.

Zaveďme nyní základní pojmy. Stěně rovinného grafu, která se skládá z k hran, budeme říkat jednoduše **k-úhelník**. Pokud máme posloupnost $(p_k) = (p_3,p_4,...)$, která bude představovat počty k-úhelníků v grafu, definujme $P = \{k \mid p_k \neq 0\}$ množinu jejích stěn.

Klasifikujme posloupnosti $(p_k) = (p_3,p_4,...)$ podle jejich vlastností.

Definice 1 (Neutrální posloupnost)**.** *Posloupnost $(p_k) = (p_3,p_4,...)$ nezáporných celých čísel je neutrální, pokud $\sum_{k\geq 3} (6-k)p_k = 0$.*

Definice 2 (Přípustná posloupnost)**.** *Posloupnost $(p_k) = (p_3,p_4,...)$ nezáporných celých čísel je přípustná, pokud $\sum_{k\geq 3} (6-k)p_k = 6$.*

Definice 3 (Přijatelná posloupnost)**.** *Posloupnost $(p_k) = (p_3,p_4,...)$ nezáporných celých čísel je přijatelná, pokud existuje rovinný kubický graf, který má právě p_k k-úhelníků.*

Všimněme si, že původní Eberhardova věta, a i mnoho jejích variant, byla formulována pro 3-spojité grafy, tedy přesněji pro jednoduché konvexní 3-polytopy. Bijekci mezi těmito dvěma strukturami dokázal až o několik desítek let později Steinitz.

Pro ujištění pojmů uveďme následující jednoduché vlastnosti přijatelných a neutrálních posloupností, které využijeme později.

Tvrzení 2. *Pro každou neutrální posloupnost q platí $\exists a,b \in Q : a < 6 \land b > 6$.*

Součet dvou neutrálních posloupností je neutrální posloupnost. Součet neutrální a přípustné posloupnosti je přípustná posloupnost.

Pro každou posloupnost $(p_k) = (p_3,p_4,...)$ existuje neutrální posloupnost q, že $p + qn$ je přijatelná pro nějaké $n \in \mathbb{N}$.

Pro ukázání první vlastnosti si stačí uvědomit, že chceme výraz $\sum_{k\geq 3} (6-k)p_k = 0$, kde pro $k < 6$ je sčítaný člen kladný, zato pro $k > 6$ je sčítaný člen záporný. Druhá vlastnost plyne z definice a distributivity násobení. Třetí tvrzení pro $q = (q_6)$ je slabší verze (1).

2. Konstrukce důkazu

Představme hypotézu, kterou se snažíme ověřit.

Hypotéza 3. *Mějme přípustnou posloupnost* $p = (p_k|3 \leq k \neq 6)$ *a neutrální posloupnost* $q = (q_k \mid 3 \leq k \neq 6)$, *pak existuje takové přirozené* n, *že* $p + nq$ *je přijatelná.*

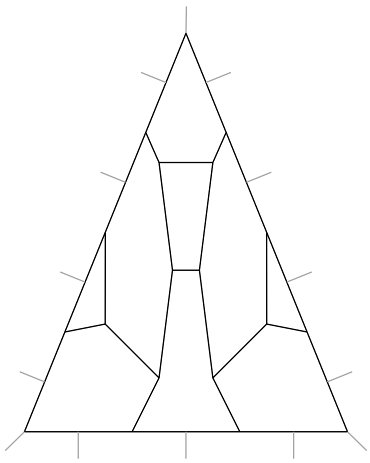
TODO zmínit výjimku pro torus.

V článku [4] autoři naznačují konstrukci důkazu, za předpokladu, že existují nějaké pomocné grafy. V další kapitole představíme způsob, jak takové grafy hledat. Teď se zaměříme na důkaz samotný, respektive nejprve představíme graf, který v důkazu pomáhal již Eberhardovi.

2.1 Triarky a operace s nimi

Definice 4 (Triark)**.** *Triark je takový rovinný graf* T, *že vrcholy jeho vnější stěny tvoří cyklus* C, *každý vnitřní vrchol (tj. vrcholy* $T - C$*) má v* T *stupeň právě 3 a v* C *jsou tři navzájem různé vrcholy* x, y, z *stupně 2 -* ***rohy****, že vrcholy každé ze tří cest v C, které vzniknou odstraněním rohů z C, mají střídavě stupeň 2 a 3, počínaje i konče stupněm 2.*

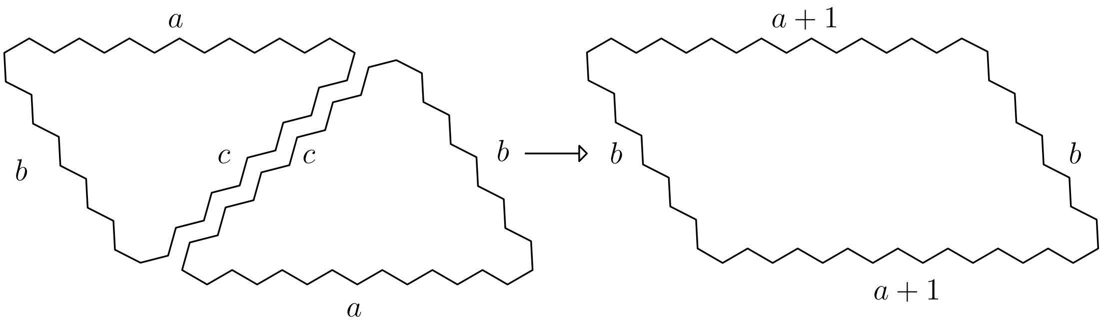
Strana triarku je každá z výše zmíněných cest v C, ke které na oba konce připojíme i příslušný roh. **Délka strany** triarku odpovídá počtu jejích vnitřních vrcholů stupně 2 v T. O triarku se stranami délky a, b, c mluvíme jako o (a, b, c,)-**triarku**. Poznamenejme, že na pořadí stran v názvu nezáleží (odpovídají rotacím). Později využijeme ještě dalšího značení. M-**triark** má vnitřní strany pouze velikostí z M. U triarku přejímáme některé termíny používané pro trojúhelníky a jejich význam bude vždy intuitivní.



Obrázek 2.1: Ukázka (4,4,3){4,7}-triarku.

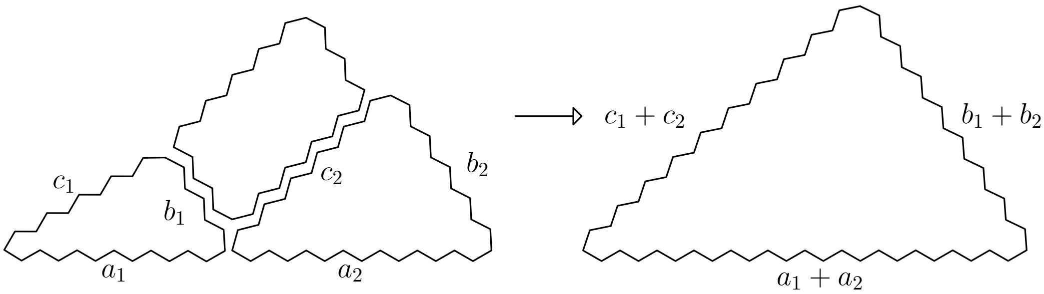
Zmiňme velmi užitečnou vlastnost triarků: pokud máme dva triarky, oba mající stranu stejné délky, můžeme je za tuto stranu slepit a získáme opět graf, jehož vnitřní vrcholy mají stupeň 3. (Při slepovaní se ztotožní vždy vrchol stupně

2 s vrcholem stupně 3.) Pokud slepíme triarky tak, že protější strany výsledného grafu budou stejně dlouhé (tedy speciálně při slepení dvou stejný triarků), budeme podle jeho tvaru mluvit o **rovnoběžníku**. Rovnoběžníky jdou navíc slepovat podobně jako triarky a můžeme tím každý již existující rovnoběžník zvětšit na libovolný rozměr, který je násobkem jeho původních rozměrů.



Obrázek 2.2: Spojení dvou triarků za vzniku rovnoběžníku.

Mohli bychom ale chtít spojovat triarky tak, aby výsledkem byl opět triark. Mějme (a_1, b_1, c_1)-triark a (a_2, b_2, c_2)-triark a vhodný rovnoběžník. Slepením, jako na obrázku 2.3, vznikne ($a_1 + a_2$, $b_1 + b_2$, $c_1 + c_2$)-triark.



Obrázek 2.3: Spojení triarků spolu s rovnoběžníkem za vzniku triarku.

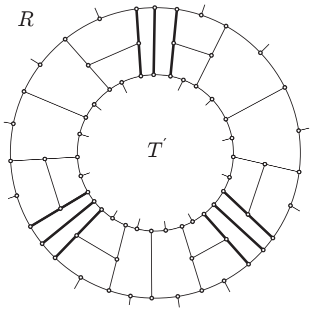
A závěrem budeme chtít spojit dva triarky tak, aby výsledkem byl kubický graf (tedy aby nevznikly žádné nové stěny s vrcholy stupně dva). Graf, který má tuto funkci označíme za **prstenec**. Jde o souvislý graf, ve kterém můžeme vyznačit dva disjunktní cykly tak, že stupně vrcholů každého cyklu jsou opačné než stupně příslušných triarků, které chceme spojit. Kde opačně znamená záměna dvoj- a tří- vaznosti vrcholů. Způsob spojení odpovídá přesně zavedení definice, ale je znázorněno i na obrázku 2.4.

TODO připravit vlastní obrázky (které víc odpovídají popiskům) pro ukázku slepování.

2.2 Důkaz za předpokladu existence pomocných grafů

Převeďme hypotézu ve větu.

Věta 4. *Mějme přípustnou posloupnost $p = (p_k | 3 \leq k \neq 6)$ a neutrální posloupnost $q = (q_k \mid 3 \leq k \neq 6)$ a následující grafy pro nějaké přirozené k:*



Obrázek 2.4: Spojení triarků prstencem za vzniku kubického grafu.

(i) *(k, k, k),Q-triark;*

(ii) *(k, k, $k-1$),Q-triark;*

(iii) *rovnoramenný $Q \cup \{1\}$-triark, délky jehož stejných stran jsou dělitelné k a který obsahuje právě jednu stěnu velikosti l pro každé nenulové p_l v p, této stěně říkejme* ***jádro*** *triarku;*

(iv) *prstenec, který dokáže spojit dva stejně velké, rovnostranné triarky.*

Pak existuje takové přirozené n, že $p + nq$ je přijatelná.

Myšlenka důkazu pak není příliš složitá: každou stěnu ze zadané posloupnosti p zabalíme do triarku (iii), připravíme si pomocné lepící a zkrášlující prvky (ii), díky kterým získáme jediný velký rovnostranný triark. K němu zkonstruujeme ještě jeden se stejně dlouhými (i) a pomocným prstencem je spojíme v kýžený graf (iv). Všechny tyto pomocné objekty jsou totiž (kromě jader) jen ze stěn, které jsou v zadané neutrální posloupnosti, lepicí operace zachovávají kubičnost uvnitř grafu a prstenec pak spojí dva triarky v hledaný kubický graf.

Důkaz. Nejprve slepíme dva $(k, k, k-1)$-triarky za stěnu délky k a získáme rovnoběžník se všemi stranami délky k. A díky slepování můžeme získat i libovolný rovnoběžník o rozměrech mk, lk pro m, l přirozená. Poté postupně spojujeme jednotlivé jádrové triarky za pomoci příslušného lichoběžníku, dokud nezískáme jediný triark, který obsahuje všechny stěny z P.

Zkusme vzniklý triark upravit na rovnostranný. Při slepování triarků se velikosti výsledných stran rovnají součtům původním. Pokud tedy slepujeme s $(k, k, k-1)$-triarkem, zmenšíme vždy tu stěnu, na kterou připadne rozměr $k-1$, vůči ostatním. Takže pokud budeme opakovat lepení výsledného triarku, v každém kroku se součet rozdílů mezi stranami zmenší a tedy nutně získáme rovnostranný triark. Modifikujme ho stejnou operací, aby zůstal rovnostranný, ale navíc délka jeho stran byla násobkem k a označme výsledný triark T_1.

Slepováním (k,k,k)-triarků z (i) spolu s k,k rovnoběžníky získáme druhý triark T_2 o stejném rozměru jako T_1. Kdybychom celý problém řešili na tóru, stačilo by T_1 a T_2 spojit do rovnoběžníku a sjednotit odpovídající strany. Na kouli místo toho použijeme prstenec z (iv). Tím získáme graf, který je kubický, rovinný a obsahuje požadované stěny.

TODO Možná poznamenat, že tím v podstatě je dokázána i verze věty, kde poslední řádek zní Pak existuje nekonečně mnoho takových přirozených n, že $p + nq$ je přijatelná.

3. Řešítko

Abychom mohli dokončit důkaz některých instancí hypotézy 3, potřebujeme získat požadované stavební bloky. Nabízí se naprogramovat řešítko, které bude umět alespoň některé typy hledaných grafů najít. Hlavním cílem této práce bylo takový program připravit a pomocí něj získat lepší představu o potenciálu uvedené konstrukce důkazu.

3.1 Algoritmus

Program na vstupu očekává zadání vnější stěny: každý vrchol je zastoupen jedním bitem, který určuje, zda má být ve výsledném grafu stupně 2 nebo 3. Navíc očekává seznam velikostí stěn, které má využít. Na výstupu informuje, zda se mu daný graf podařilo najít (říkejme **vyplnit**), a umožní jej exportovat.

Postup hledání původně imitoval lidské pokusy o řešení problému: nakreslit si vnější stěnu, zkusit spojit nějaké dva vrcholy řetízkem vhodné délky (aby nově uzavřená stěna byla z neutrální posloupnosti) a dokud je místo na papíře, spojovat. Pak si překreslit nejvnitřnější, zatím neuzavřenou stěnu (budeme mluvit o **hranici**), ta se stane „vnější stěnou“ na novém papíře a pokračovat. V situaci, kdy nelze dál nic spojit, nebo je jasné, že graf nemůže vyhovovat parametrům, vrátit se podle uvážení zpět. Viz obrázek TODO odkaz.

Kdybychom chtěli znát jen ano/ne odpověď, jestli graf existuje, nebylo by vůbec třeba si pamatovat celý rozpracovaný graf, stačilo-by pracovat s hranicemi, které navíc stačí reprezentovat jako binární číslo. Výsledkem by pak mohla být jen posloupnost hranic, kterými se prošlo před uzavřením grafu, nebo samotné „ano/ne“. Překvapivě obtížné je pak z této posloupnosti nestrojově získat skutečný graf, proto program nabízí i možnost graf dodatečně rekonstruovat podle prošlých stavů.

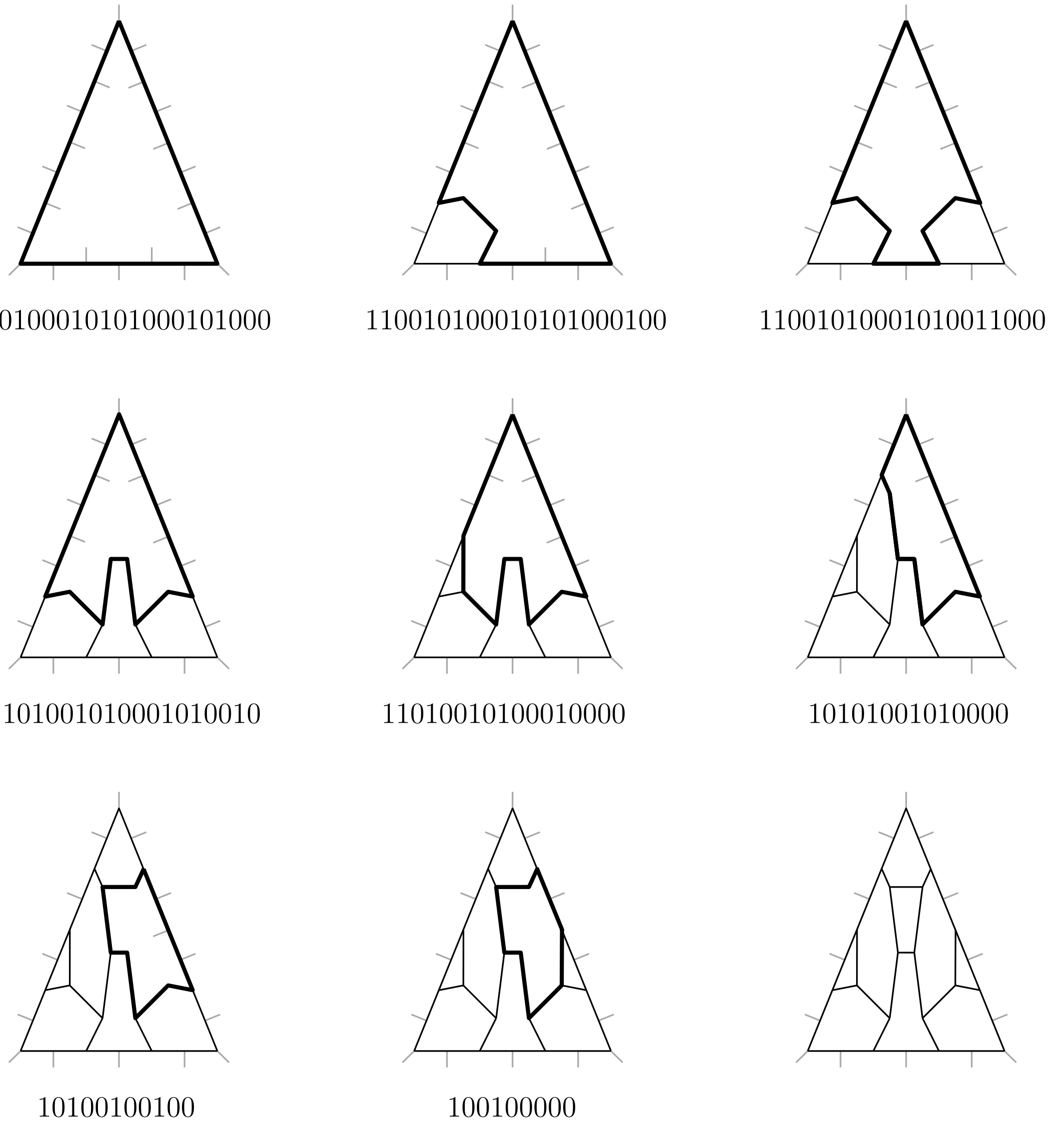
V tento okamžik je jasné, že problém je vlastně prohledávání v binárních řetězcích (které reprezentují hranice). Je proto vhodné zmínit, podle jakého kritéria se program rozhoduje, kterým směrem hledat dále. Implementace vždy upřednostňuje ke zpracování již nalezený řetězec nejmenší délky, a pro něj najde všechny další sousedy.

Aby toto prohledávání fungovalo dobře, je třeba trochu zkomplikovat reprezentaci hranice. Hlavním požadavkem bude identita mezi reprezentací (jediným binárním řetězcem) a všemi hranicemi (tedy cykly, na kterých vyznačené vrcholy ještě vyžadují dalšího souseda), které jsou pro algoritmus izomorfní - tedy všechny rotace a převrácení hranice.

Definice 5 (Hranice a její reprezentace)**.** *Cyklus C a množinu $I \subseteq C(V)$ zveme hranicí H. Množina I jsou právě ty vrcholy, které ve výsledném vyplnění musí mít dalšího souseda. Pokud $i = \emptyset$, mluvíme o hranici přímo jako o stěně.*

Binární řetězec (číslo) reprezentující hranici H získáme následně: každý vrchol H označíme buď znakem 1 (jako I v "in"podle orientace pomyslené hrany) nebo 0 (jako 0 v "out"). Zaznamenejme pak všechny řetězce, které získáme čtením od každého vrcholu pro i proti směru hodinových ručiček. Ten z nich, který z nich *má největší hodnotu (pokud řetězec chápeme jako binární číslo), je reprezentací hranice. Pokud $i = \emptyset$, pak přidejme speciální znak a zapamatujme počet vrcholů.*

Pro lepší představu přikládáme posloupnost hranic s vizualizací na obrázku 3.1, která řeší (4,4,3){4,7}-triark.



Obrázek 3.1: Možný postup vyplnění (4,4,3){4,7}-triarku. Hodnota pod grafem vždy odpovídá reprezentaci aktuální, tučně zvýrazněné, hranice.

Pro jistotu poznamenejme, že pokud program hledaný graf nenašel, může, ale nemusí to znamenat, že neexistuje.

Podle předchozí kapitoly pro dokončení důkazu pro konkrétní dvojici p a q a nějaké přirozené k potřebujeme tyto čtyři typy grafů:

(i) (k, k, k),Q-triark;

(ii) $(k, k, k-1)$,Q-triark;

(iii) (mk,mk,x),$Q \cup \{1\}$-triark, kde je právě jedna stěna velikosti l;

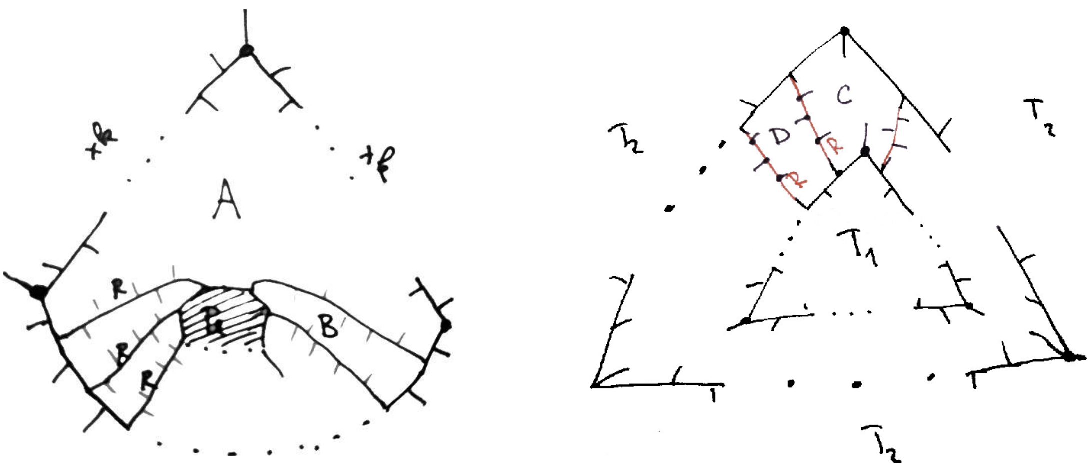
(iv) prstenec, který dokáže spojit dva stejně velké, rovnostranné triarky.

Pro dané k získáme grafy (i) a (ii) z řešítka hned. Pro zbylé je nutné pomoci si konstrukcí, která se ukázala jako úspěšná pro některé neutrální sekvence. O výsledcích získaných z programu píšeme v další kapitole. TODO odkaz?

Na graf (iii) se neumíme zeptat přímo, protože potřebujeme v grafu mít právě jednu stěnu délky p_l. Spojme proto stěnu s vnější stěnou triarku ručně a ptejme se na výplň vzniklých oblastí A a B. Ke spojení použijeme l kopií řetízku R, každý napojíme na jeden z vrcholů jádra a druhé konce spojíme s „in“ vrcholy základny triarku. V řetízku navíc fixujeme, ve které straně od něj budou mít které jeho vrcholy třetího souseda (znázorněno šedě v obrázku TODO odkaz). Konstrukce je obecná pro všechny přípustné hodnoty l, tedy není závislá na volbě p. Navíc není (při dobré volně R) vynucená ani příliš velká stěna, takže konstrukce může být obecná i pro všechny q (protože q je neutrální, tedy v ní musí být nenulová hodnota na pozici reprezentující stěnu velikosti alespoň 6, tvrzení (2).

Podobnou konstrukci tvoříme i pro graf typu (iv). V tomto případě za pomoci řetízků spojujeme odpovídající vrcholy rovnostranných, stejně velkých triarků T_1 a T_2. Vzniknou dva typy oblastí - C při rozích triarku a D mezi odpovídajícími kusy stran triarků. Poznamenejme, že na obrázku je T_2 nakreslen "naruby"TODO lepší slovo?.

Tyto konstrukce jsou v řešítku implementovány. Výsledný program tedy na vstupu očekává seznam velikostí stěn, které mohou tvořit neutrální posloupnost. Pokud pro daný seznam stěn existuje více neutrálních posloupností, využije program pro každý pomocný graf libovolnou z nich.



Obrázek 3.2: Vlevo konstrukce grafu typu (iii), konkrétně $(xk,xk,l+1)$-triarku s jádrem velikosti l, vpravo konstrukce grafu typu (iv).

3.2 Uživatelská dokumentace

4. Výsledky

Díky programu popsanému v předchozí kapitole bylo možné zkusit dokončit důkaz věty (3) pro některé neutrální posloupnosti q (na volbě posloupnosti p nezáleží, protože konstrukce pro jádrový triark nezávisí na velikosti jádra).

Vzhledem k výpočetním omezením programu jsme zkusili dokončit důkaz věty pouze pro takové neutrální posloupnosti q, že $Q = r, s$ a navíc $r < s < 18$. Seznam posloupností, pro které program nalezl potřebné grafy je v tabulce 4.1. Pokud bychom se omezili na rozsah $r < s < 14$, pak program grafy nalezl právě tehdy, když r a s jsou nesoudělná čísla.

-	7	8	9	10	11	12	13	14	15	16	17
3	•	•		•	•		•	•		•	•
4	•		•		•		•		•		
5	•	•	•		•	•	•				

Obrázek 4.1: Úplný výčet dvojic stran, pro které se podařilo dokončit důkaz věty (3).

TODO označneí

Všechny potřebné grafy pro doložení důkazu jsou k práci přiloženy. TODO přiložit

Přirozenou snahou při zkoumání nalezených grafů je grafy zobrazit, aby byly pro člověka dobře čitelné. V další sekci se tomuto tématu krátce věnujeme.

4.1 Kreslení

TODO přesunout Při hledání triarků byla vynaložena netriviální snaha pro nalezení způsobu, jak výsledný graf dobře zobrazit - od rovinného grafu by se dalo čekat, že nepůjde o příliš složitý úkol. Po špatné zkušenosti s dostupnými možnostmi jako GraphViz nebo funkcemi v SageMath, došlo na implementaci kreslícího algoritmu založeného na Tuttově kreslení TODO zmínit odkaz a dál o kreslení nemluvit? Nebo popsat?

Závěr

Seznam použité literatury

[1] V. Eberhard. Zur morphologie der polyeder. *Leipzig: B. G. Teubner, 1891*, 1891.

[2] J. C. Fisher. An existence theorem for simple convex polyhedra. *Discrete Math. 7*, pages 75–97, 1974.

[3] B. Grünbaum. *Convex Polytopes*. Second Edition. Springer-Verlag, New York, 2003.

[4] M. DeVos, A. Georgakopoulos, B. Mohar, and R. Šámal. An eberhard-like theorem for pentagons and heptagons. *TODO*, 2009.

Seznam obrázků

Seznam tabulek

Seznam použitých zkratek

A. Přílohy

A.1 První příloha