

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

FARBENIA KUBICKÝCH GRAFOV
STEINEROVSKÝMI KONFIGURÁCIAMI
BAKALÁRSKA PRÁCA

2017
FILIP MATUŠÁK

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

FARBENIA KUBICKÝCH GRAFOV
STEINEROVSKÝMI KONFIGURÁCIAMI
BAKALÁRSKA PRÁCA

Študijný program: Informatika
Študijný odbor: 2508 Informatika
Školiace pracovisko: Katedra informatiky
Školiteľ: prof. RNDr. Martin Škoviera, PhD.

Bratislava, 2017
Filip Matušák



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Filip Matušák
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Farbenia kubických grafov Steinerovskými konfiguráciami
Colouring of cubic graphs with Steiner configurations

Cieľ: Cieľom práce bude študovať farbenia kubických grafov pomocou vybraných konfigurácií v Steinerovských systémoch trojíc. Budú sa analyzovať najdôležitejšie vlastnosti takých farbení a vytvoria sa aj potrebné výpočtové prostriedky na hľadanie požadovaných farbení. Pomocou týchto prostriedkov sa budú analyzovať niektoré triedy snarkov.

Literatúra: Aktuálna časopisecká literatúra a poznámky vedúceho.

Kľúčové slová: Steinerovský systém trojíc, farbenie, snark, programový systém na vyhľadávanie farbení

Vedúci: prof. RNDr. Martin Škoviera, PhD.
Katedra: FMFI.KI - Katedra informatiky
Vedúci katedry: prof. RNDr. Martin Škoviera, PhD.

Spôsob sprístupnenia elektronickej verzie práce:
bez obmedzenia

Dátum zadania: 25.10.2016

Dátum schválenia: 26.10.2016

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

Pod'akovanie: Chcel by som sa pod'akovať prof. RNDr. Martinovi Škovierovi, PhD. za vedenie práce a cenné poznámky. Tiež chcem pod'akovať rodine a priateľom za podporu.

Abstrakt

Táto práca je zameraná na konštrukciu snarkov, ktoré sa nedajú pokryť štyrmi perfektnými páreniami. Využívame fakt, že pokrytie snarku štyrmi perfektnými páreniami je ekvivalentné farbeniu snarku konkrétnou konfiguráciou 10 bodov a 6 priamok v 3-rozmernom projektívnom priestore nad 2-prvkovým poľom. Nainplementovali sme program, ktorý testuje zafarbiteľnosť snarkov touto konfiguráciou a tiež farebné vlastnosti častí grafov. Pomocou vlastností, ktoré sme odhalili, sme potom konštruovali snarky, ktoré sa nedajú pokryť štyrmi perfektnými páreniami.

Kľúčové slová: snark, farbenie, Steinerovský systém trojíc

Abstract

This thesis is focused on the construction of snarks whose edge-set cannot be covered by four perfect matchings. We use the fact that covering graphs by four perfect matchings is equivalent to coloring snark with a configuration of 10 points and 6 straight lines in the 3-dimensional projective space over the field of two elements. We have developed a software that tests colorability of snarks with configurations and properties of colored graph components. We have constructed new snarks whose edge-set cannot be covered by four perfect matchings by using properties that we have discovered.

Keywords: snark, coloring, Steiner configuration

Obsah

Úvod	1
1 Snarky - základné pojmy a vlastnosti	4
1.1 Snarky	4
1.2 Pokrytie perfektnými páreniami	5
2 Steinerovské systémy trojíc	6
2.1 Systémy trojíc	6
2.2 Farbenie steinerovskými konfiguráciami	6
2.3 Steinerovské konfigurácie a 1-faktory	7
2.4 Multipóly	9
2.5 Prechodové relácie	9
3 Známe konštrukcie	11
3.1 Konštrukcia "veterný mlyn"(windmill)	11
3.2 Konštrukcia "stromovitých"(treelike) snarkov	12
4 Výsledky práce	14
4.1 Dekolineátor	14
4.2 Komponenty I a B	16
4.3 Halinov dipól	20
4.4 Nedofarbitelné komponenty	20
4.5 Snark s vysokou nepárnosťou	21
4.6 Halinov dipól ako substitúcia	22
4.7 Bipartitné vložky	23
4.8 Priamky a antipriamky	27
5 Snark Master	29
5.1 Funkcionalita a použitie Snark Mastra	29
5.2 Scala	31
5.3 Akka	31
5.4 Implementácia	32

<i>OBSAH</i>	vii
5.5 Kompilácia a spustenie	34
Záver	36
Prílohy	40

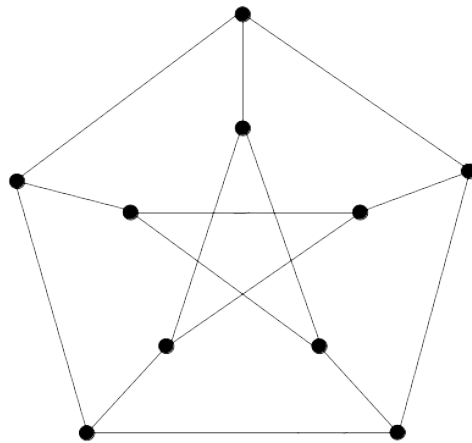
Zoznam obrázkov

1	Petersenov graf	1
2	Graf WM_{34}	2
2.1	Fanova rovina	7
2.2	Tetraedrálna konfigurácia	8
3.1	Konštrukcia windmill	11
3.2	Konštrukcia komponentu F	12
3.3	Konštrukcia WM_{34}	12
4.1	Dekolineátor z Petersenovho grafu	16
4.2	Komponent I	16
4.3	Rozšírenie Petersenovho grafu pomocou XIX	17
4.4	Operácia \odot -súčinu na $(2,2,1)$ -póloch	18
4.5	Halinov dipól	20
4.6	Komponent HXHX	21
4.7	Graf HXHX	22
4.8	Cesta dĺžky 1 a Halinov dipól	23
4.9	Nahradenie hrany Halinovým dipólom	24
4.10	Heawoodov graf	24
4.11	Pappusov graf	24
4.12	WM_{34} s bipartitnou vložkou	26
4.13	Spojenie 3-pólov zo snarku s pokrývacím indexom aspoň 5 a bipartitného grafu	27
4.14	Spojenie dvoch 3-pólov zo snarkov s pokrývacím indexom aspoň 5 a komponentu I	28
5.1	Označenie farieb v T-konfigurácii	31
5.2	Hierarchia actorov	33

Úvod

Cieľom tejto bakalárskej práce je pomôcť pri skúmaní vlastností kubických grafov nazývaných snarky. Ich základnou charakteristikou je, že sú to 3-regulárne bezmostové grafy, ktoré sa nedajú hranovo zafarbiť tromi farbami. Podľa Vizingovej vety je teda ich chromatický index 4.

Hoci sa ich skúmaniu venuje viacero tímov po celom svete, stále je táto trieda grafov veľmi málo preskúmaná a pochopená. K pochopeniu snarkov nám pomáha analýza rôznych vlastností a invariantov. Mnohé zo skúmaných snarkových invariantov rôznym spôsobom merajú, nakoľko je daný snark ďaleko od hranovo 3-zafarbiteľného grafu. Jedným z týchto invariantov je počet perfektných párení (1-faktorov) potrebných na pokrytie všetkých hrán. Túto hodnotu pre graf G budeme volať *pokrývací index* a označovať $\tau(G)$. Existencia pokrytia štyrmi perfektnými páreniami súvisí s farbením konkrétnou konfiguráciou 10 bodov a 6 priamok v 3-rozmernom projektívnom priestore nad 2-prvkovým poľom, pracovne nazvanou tetraedrálna konfigurácia. Jej body a priamky totiž ohraničujú štvorsten (tetraéder). Podobným konfiguráciám sa venujeme v osobitnej kapitole.

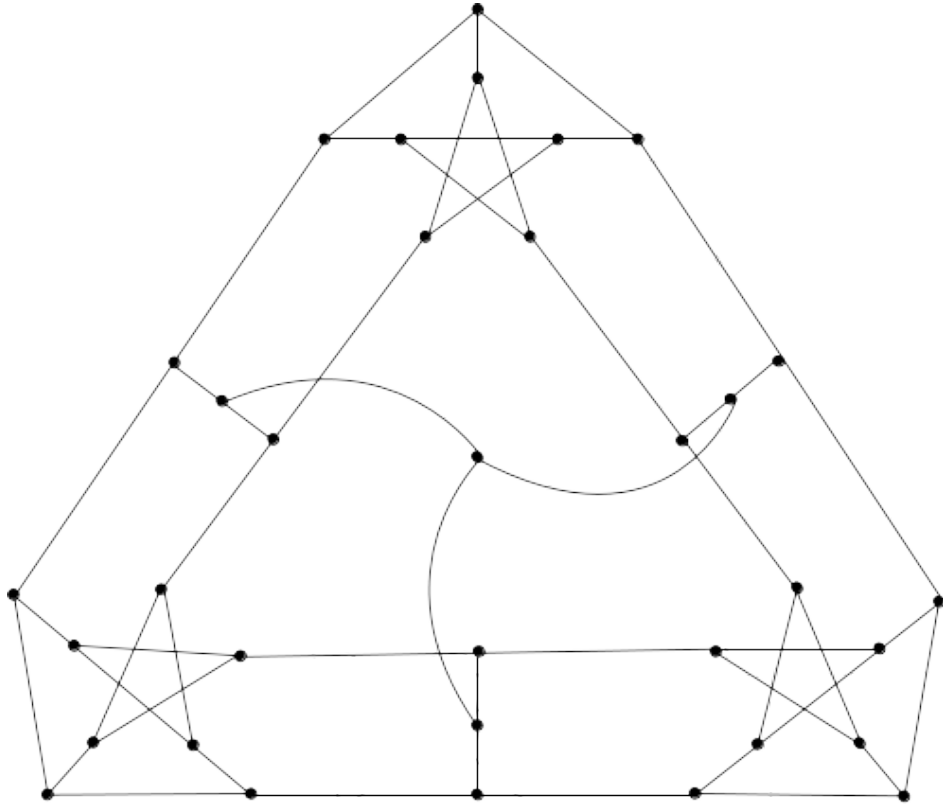


Obr. 1: Petersenov graf

Veľká väčšina malých snarkov má pokrývací index 4, preto sú zaujímavé najmä tie, ktoré sa nedajú pokryť štyrmi 1-faktormi. Počítačové generovanie a testovanie ukázalo, že do 36 vrcholov existujú len dva netriviálne snarky s pokrývacím indexom aspoň 5 [3]. Jeden je Petersenov graf (obrázok 1) a druhý je graf na 34 vrchoch s názvom

WM_{34} na obrázku 2. Časť tejto práce je zameraná na počítačovú podporu skúmania snarkov s pokrývacím indexom väčším ako 4. Skúmame ich vlastnosti a pomocou nich konštruujeme ďalšie snarky s pokrývacím indexom aspoň 5. S ich použitím sa nám podarilo vytvoriť niekoľko nekonečných sérií takých snarkov. Ale cieľom je aj nájsť, čo najviac malých snarkov s pokrývacím indexom aspoň 5.

Jednou z motivácií štúdia snarkov sú ťažké a dlho otvorené hypotézy. Snarky sú kandidátmi na protipríklady pre tieto hypotézy. Prvou je *Cycle double cover* hypotéza [8], podľa ktorej sa každý bezmostový graf dá pokryť cyklami tak, aby každá hrana bola v dvoch cykloch. Ďalšia je *Five Flow* hypotéza. Táto hypotéza hovorí, že každý bezmostový graf má nikde nulový 5-tok. *Fulkersonova* hypotéza hovorí, že v každom kubickom bezmostovom grafe existuje šesť takých perfektných párení, že každá hrana patrí do práve dvoch. Podľa *Bergeovej* hypotézy sa hrany každého kubického bezmostového grafu dajú pokryť piatimi 1-faktormi. Neskôr sa ukázalo, že posledné dve spomenuté hypotézy sú ekvivalentné [11].



Obr. 2: Graf WM_{34}

Na testovanie vlastností skonštruovaných snarkov sme naimplementovali program Snark Master. Umožňuje nám testovať zafarbenie grafov rôznymi steinerovskými konfiguráciami a tiež prechodové relácie, o ktorých budeme písať neskôr. Tento program je napísaný v jazyku Scala a na testovanie využíva knižnicu riešiacu SAT problém. Vzhľadom na to, že ide o riešenie hrubou silou, testovanie aj relatívne malých snarkov (do 60

vrcholov) môže trvať hodiny, a to aj napriek viacerým optimalizáciám pre konkrétne konfigurácie.

Výsledkom práce sú nami skonštruované snarky s pokrývacím indexom aspoň 5 a ich nekonečné série. Ďalej sú to čiastočné komponenty snarkov, ktoré majú zaujímavé vlastnosti vzhľadom na ich použitie v konštrukcii nových snarkov. K týmto snarkom a komponentom sú aj ilustrácie a tiež ich zápis v nami používanom formáte grafov. Na záver je popísaný program Snark Master.

Kapitola 1

Snarky - základné pojmy a vlastnosti

Základné pojmy z teórie grafov sú ľahko dostupné v učebniciach a monografiách. My vychádzame z knihy R. Diestela [4]. Pokiaľ ide o snarky, zatiaľ nie sú vôbec knižne spracované. Existuje niekoľko prehľadových článkov, zväčšia staršieho dáta [15]. Pôvodných vedeckých štúdií na tému snarkov je však veľa a ich počet v ostatnom čase začal prudko rásť. V tejto kapitole uvedieme základné charakteristiky snarkov a niektoré poznatky o nich. Sformulujeme tiež otázky, ktorým sa v práci budeme venovať.

1.1 Snarky

Budeme sa zaoberať len jednoduchými grafmi, teda bez násobných hrán a bez slučiek. Kubický graf nazývame snark, ak je bezmostový a jeho chromatický index je 4. Podľa Vizingovej vety chromatický index každého kubického grafu je 3 alebo 4. Aby sme vylúčili triviálne prípady, kladieme na snark spravidla aj podmienku, aby cyklická súvislosť bola aspoň 4 a aby mal obvod väčší alebo rovný 5. To znamená, že neobsahuje cyklus dĺžky menej ako 5. Snark, ktorý nespĺňa posledné dve podmienky budeme nazývať triviálny.

Prvý objavený snark bol Petersenov graf (1898). Je zobrazený na obrázku 1. Až do polovice 70-tych rokov minulého storočia boli známe len tri netriviálne snarky. Prvé nekonečné triedy objavili v tom čase G. M. Adelson Velski a V. K. Titov [2] a R. Isaacs [7]. Hoci aj v súčasnosti sa výskumu snarkov venuje viacero tímov po celom svete, snarky nie sú dostatočne preskúmané a pochopené. Preto je v tejto oblasti stále veľa nezodpovedaných otázok a hypotéz. Zatiaľ nemáme dostatok znalostí o štruktúre snarkov, aby sme ich mohli zodpovedať.

V súčasnosti k výskumu snarkov významne pomáha aj výpočtová technika. Pomocou nej sme schopní generovať veľké množstvo snarkov a testovať ich vlastnosti. Ale ani s takým veľkým množstvom snarkov sa nepodarilo vyvrátiť niektoré hypotézy. Treba poznamenať, že z množstva kubických grafov na danom počte vrcholov je len

málo snarkov. Vieme, že ich podiel s rastúcim počtom vrcholov ide k nule. Aj preto sú pre nás zaujímavé.

1.2 Pokrytie perfektnými páreniami

Perfektné párenie alebo *1-faktor* grafu $G(V, E)$ je podgraf $H(V', E')$ grafu G , pre ktorý platí $V = V'$ a každý vrchol podgrafu H je stupňa 1. Nie každý graf nutne má 1-faktor. Petersen však dokázal, že každý bezmostový kubický graf má 1-faktor [13]. Neskôr bolo dokázané, že každá hrana bezmostového kubického grafu je v nejakom 1-faktore. Z toho vieme, že pre každý graf G existuje množina 1-faktorov, ktorými sa dá pokryť celý graf G . Preto zavádzame pojem *pokrývací index* a označujeme ho $\tau(G)$. Pokrývací index hovorí, koľko 1-faktorov potrebujeme na pokrytie grafu. Nie je známa žiadna konečná hranica, ktorá by platila pre všetky snarky.

Ak na pokrytie stačia tri 1-faktory, tieto 1-faktory musia byť disjunktné, a preto môžeme hranám patriacim do prvého 1-faktoru dať farbu 1, druhému 2 a tretiemu 3, a tým dostaneme korektné 3-farbenie. Ale to sa pri snarkoch nemôže stať, pretože tie sa nedajú zafarbiť tromi farbami. Pre snarky teda platí, že ich pokrývací index je aspoň 4. Väčšina snarkov, ktoré poznáme, sa dá pokryť štyrmi 1-faktormi. Ďalší fakt je, že nepoznáme snarky, na ktoré by päť 1-faktorov nestačilo. Hovorí o tom Bergeova hypotéza [11], podľa ktorej na pokrytie každého kubického bezmostového grafu stačí päť 1-faktorov. V tejto práci nás zaujímajú práve tie, na ktoré štyri 1-faktory nestačia. Snažíme sa vytvoriť čo najväčšie množstvo takých snarkov.

Pomocou počítačového generovania a testovania sa ukázalo, že do 36 vrcholov existujú len dva také snarky. Jeden je Petersenov graf a ďalší je snark WM_{34} na 34 vrcholov. Prvá konštrukcia, ktorá vysvetľuje pokrývací index 5 pre graf WM_{34} , je z článku [5]. Druhú konštrukciu opisujú autori M. Abreu, T. Kaiser, D. Labbate a G. Mazzuoccolo v článku Treelike snarks [1]. Obe tieto konštrukcie popisujú nekonečnú triedu snarkov, ktoré sa nedajú pokryť štyrmi 1-faktormi a pomocou nich sa dá skonštruovať aj WM_{34} . Ale sú aj iné postupy konštrukcie, ktorými ho vieme vytvoriť.

Kapitola 2

Steinerovské systémy trojíc

V tejto kapitole sa budeme venovať steinerovským systémom trojíc a steinerovským konfiguráciám. Popíšeme, ako pomocou nich vieme farbiť grafy a aký majú súvis s tokmi. To všetko nám pomôže pri skúmaní snarkov s pokrývacím indexom aspoň 5.

2.1 Systémy trojíc

Steinerovský systém trojíc S je dvojica (P, B) kde P je n -prvková množina, ktorej prvky sa nazývajú body a B je množina 3-prvkových podmnožín množiny P . Trojica v systéme sa tiež nazýva priamka alebo blok. Musí tiež platiť, že každá dvojica bodov sa nachádza spolu práve v jednom bloku. Bloky podľa definície steinerovského systému majú spoločný maximálne jeden bod.

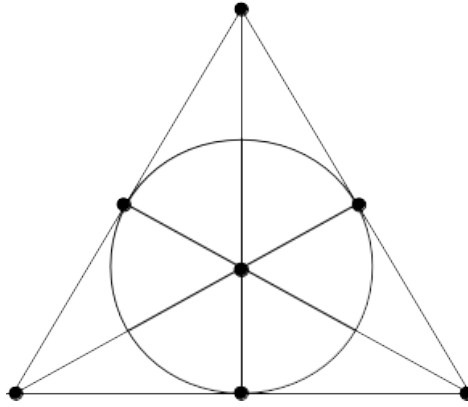
Najmenší steinerovský systém je triviálny systém, ktorý má len tri body a jeden blok. Najmenší netriviálny systém je *Fanova rovina*, ktorá obsahuje sedem bodov a sedem priamok. Je znázornená na obrázku 2.1. Je známe, že steinerovský systém na n vrcholoch existuje práve vtedy, keď $n \equiv 1$ alebo $3 \pmod{6}$ [6].

Čiastočný steinerovský systém je opäť dvojica (P', B') pričom $P' \subseteq P$ a $B' \subseteq B$ pre nejaký systém (P, B) . Navyše musí platiť, že neobsahuje izolované body a každá dvojica bodov je najviac v jednom bloku. Čiastočné steinerovské systémy nazývame tiež *steinerovskými konfiguráciami*.

Homomorfizmus zo steinerovskej konfigurácie S do T je zobrazenie bodov konfigurácie S do bodov konfigurácie T také, že obrazy bodov, ktoré v S tvorili blok, musia tiež tvoriť blok v T .

2.2 Farbenie steinerovskými konfiguráciami

Pomocou steinerovských konfigurácií môžeme farbiť kubické grafy. Dovoľené farby vo farbení sú body v steinerovskej konfigurácii. Podmienkou farbenia je, že farby použité



Obr. 2.1: Fanova rovina

vo farbení na hranách incidentných s nejakým vrcholom musia v steinerovskej konfigurácii tvoriť blok. Z definície steinerovskej konfigurácie vyplýva, že dve zafarbené hrany incidentné s jedným vrcholom implikujú farbu tretej hrany. Ak sa kubický graf dá hranovo zafarbiť nejakým systémom S , tak hovoríme, že má S -farbenie. Napríklad farbenie triviálnym systémom je rovnaké ako obyčajné farbenie tromi farbami. Snarky teda nemajú farbenie triviálnym steinerovským systémom.

Za vrcholy do steinerovského systému trojíc môžeme zobrať napríklad aj prvky grupy. Zoberieme za vrcholy prvky $\mathbb{Z}_2^2 - \{0\}$, systém bude obsahovať jeden blok so všetkými vrcholmi. Tento systém má tieto dve vlastnosti:

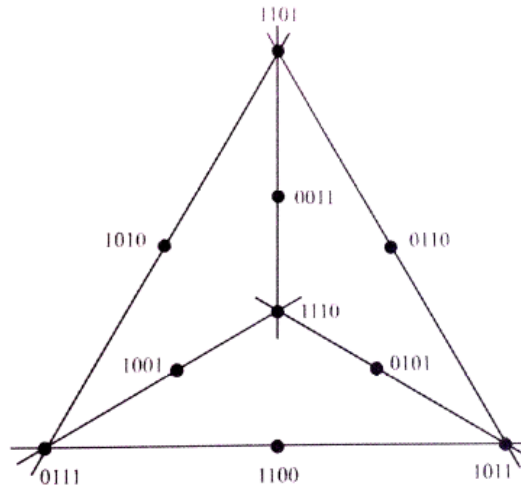
1. hodnota vrcholu je sama sebe opačnou hodnotou
2. súčet hodnôt vrcholov v jednom bloku je 0

Vďaka týmto vlastnostiam sa na farbenie takýmto steinerovským systémom trojíc môžeme pozeráť ako na nikde nulový tok [10]. Prvá vlastnosť hovorí, že ak jedným smerom v hrane tečie hodnota X , tak druhou stranou tiež tečie X . To nám ponúka nový pohľad na problém. Pri konštruovaní snarkov z menších komponentov môžeme skúmať ich tokové vlastnosti. Vďaka nim môžeme odvodiť vlastnosti skonštruovaného snarku. Teda pri vhodnom ohodnotení bodov steinerovskej konfigurácie je farbenie aj tokom.

2.3 Steinerovské konfigurácie a 1-faktory

Teraz sa pozrieme na to, akým spôsobom súvisí pokrytie snarku štyrmi 1-faktormi so steinerovskými konfiguráciami a teda aj s farbením a tokmi.

Majme štyri 1-faktory A, B, C, D . Majme snark s pokrývacím indexom 4 a pokryme ho 1-faktormi A, B, C, D . Potom pre hrany incidentné s jedným vrcholom platí, že dve hrany patria každá do jedného 1-faktora a jedna hrana patrí do dvoch 1-faktorov. Nech farba hrany je množina 1-faktorov, do ktorých hrana patrí. Pre farby v okolí jedného



Obr. 2.2: Tetraedrálna konfigurácia

vrcholu platí, že každý 1-faktor sa tam nachádza práve raz, buď samostatne, alebo vo dvojici s iným 1-faktorom. Teda povolené trojice farieb v okolí vrcholu sú (A, B, CD) , (A, C, BD) , (A, D, BC) , (B, C, AD) , (B, D, AC) , (C, D, AB) . Farby vyjadríme pomocou prvkov $\mathbb{Z}_2^4 - \{0\}$. $A = 0111$, $B = 1011$, $C = 1101$ a $D = 1110$. Farba, ktorá je kombináciou dvoch 1-faktorov bude ich súčet v $\mathbb{Z}_2^4 - \{0\}$, napr $AB = 1100$. Takto definované farbenie môžeme reprezentovať ako farbenie pomocou steinerovskej konfigurácie z obrázku 2.2. Táto konfigurácia je pre našu ďalšiu prácu dôležitá. Budeme ju volať tetraedrálna konfigurácia a označovať T . Ak je ňou snark zafarbiteľný, potom hovoríme, že má T -farbenie. Všimnime si, že T -farbenie je zároveň aj tokom v grupe \mathbb{Z}_2^4 , pričom všetky hodnoty sú v T . Je to vďaka tomu, že súčet bodov v jednom bloku je 0.

Trojicu rôznych stredových bodov jedného trojuholníka T -konfigurácie nazývame *antipriamka*. Trojicu rôznych stredových bodov, ktoré nie sú v jednom trojuholníku T -konfigurácie nazývame *kvázipriamka*.

Jediný spôsob, ako vyjadriť 0 ako súčet troch rôznych bodov z T -konfigurácie je, že tieto tri body tvoria buď priamku, alebo antipriamku v T -konfigurácii.

Platí nasledujúce tvrdenie.

Veta [12]: Bezmostový kubický graf sa dá pokryť štyrmi 1-faktormi práve vtedy, keď má T -farbenie.

Vďaka tejto vete vieme, že testovať, či sa graf dá pokryť štyrmi 1-faktormi je to isté, ako testovať, či je zafarbiteľný tetraedrálou konfiguráciou.

Hammingova váha bodu T -konfigurácie je počet jeho jednotiek. V T -konfigurácii sú dva typy bodov. Body s Hammingovou váhou 3 sú krajné body blokov. Nazývame ich *rohové body*. Potom sú to body s Hammingovou váhou 2. Tieto body sú v strede bloku a nazývame ich *stredové body*.

Pre tetraedrálnu konfiguráciu ďalej platí, že neexistuje homomorfizmus z T do triviálnej konfigurácie s tromi bodmi a jedným blokom [12]. Označme T' konfiguráciu, ktorá vznikne, keď z T odstránime bod $p \in T$, bloky, v ktorých je p a potom všetky izolované body. Potom z T' existuje homomorfizmus do triviálnej konfigurácie. To znamená, že z T' -farbenia grafu vieme získať aj 3-zafarbenie. To sa pri snarkoch nemôže stať. Preto pri každom T -farbení snarku je použitý každý bod tetraedrálnej konfigurácie.

2.4 Multipóly

Multipól je štruktúra podobná grafu. Líši sa tým, že môže obsahovať hrany, ktorých konce nemusia byť incidentné s vrcholmi. Ak je iba jeden koniec hrany incidentný s vrcholom, tak hranu nazývame *visiaca hrana* alebo tiež *polhrana*. Ak hrana nie je incidentná so žiadnym vrcholom, potom hranu nazývame *izolovaná hrana*. Visiace hrany multipólu M budeme označovať $S(M)$. Ak $|S(M)| = 0$ potom M je graf. Ak $|S(M)| = n$, potom M nazývame n -pól.

Z visiacich hrán môžeme robiť skupiny, ktoré nazývame *konektory*. Konektory nemusia pokryť celé $S(M)$. Jedna visiaca hrana môže byť maximálne v jednom konektore. Ak polhrana nie je v žiadnom konektore, tak ju nazývame *reziduálna polhrana*.

Dve visiace hrany môžeme spájať tak, že ich odstránime a pridáme hranu medzi vrcholy, s ktorými boli visiace hrany incidentné. Tiež môžeme spájať konektory S_1 a S_2 rovnakej veľkosti tak, že i -tu polhranu z S_1 spojíme s i -tou polhranou z S_2 predchádzajúcim spôsobom.

2.5 Prechodové relácie

Pri zafarbení komponentu tetraedrálou konfiguráciou musí byť súčet farieb na visiacich hranách 0, čo vyplýva z toho, že T -farbenie je tok. Na konektore veľkosti 2 budú niektoré dva vrcholy z tetraedrálnej konfigurácie. Podľa toho, aká je ich vzájomná poloha v tetraedrálnej konfigurácii, ich môžeme rozdeliť do nasledujúcich tried:

- **nula** - je buď **rohový bod**, teda dvakrát ten istý bod, ktorý je v prieniku viacerých blokov alebo **stredový bod**, čo je dvakrát ten istý bod, ktorý je v strede bloku
- **úsečka** - dva rohové body jednej priamky
- **polpriamka** - rohový a stredový bod jednej priamky
- **uhol** - dva stredové body, pričom bloky, v ktorých sú, majú spoločný bod
- **os** - dva stredové body, pričom bloky, v ktorých sú, nemajú spoločný bod

- výška - krajný a stredový bod, ktoré nie sú v jednom bloku

Pri tomto delení nám nezáleží na poradí vrámci jedného komponentu. Delenie nám pomáha popisovať, aké farbenia tetraedrálou konfiguráciou dovoľuje daný komponent. Zaujímá nás, čo môžeme dostať na konektoroch. Teda ak komponent má k konektorov, potom chceme vedieť všetky k -tice z vyššie definovaných tried, ktoré môžu byť na konektoroch pri T -zafarbení. Túto vlastnosť nazveme prechodová relácia.

Kapitola 3

Známe konštrukcie

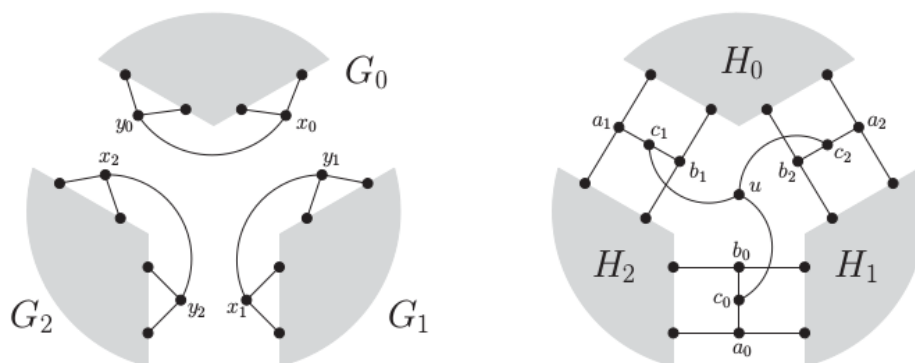
V tejto kapitole popíšeme dve známe konštrukcie nekonečných tried snarkov s pokrývacím indexom 5. Obe konštrukcie sa dajú použiť na konštrukciu WM_{34} . Neskôr však popíšem náš spôsob konštrukcie tohoto grafu.

3.1 Konštrukcia "veterný mlyn"(windmill)

Snark WM_{34} bol objavený pomocou počítačového generovania. Prvý matematický dôkaz, prečo má pokrývací index 5 bol pomocou tejto konštrukcie [5].

Postup konštrukcie je nasledovný. Pre $i \in \{0, 1, 2\}$ zoberme snark G_i , pričom $\tau(G_i) \geq 5$, ktorým má hranu $x_i y_i$. Nech x_i^0 a x_i^1 sú susedia x_i a y_i^0 a y_i^1 sú susedia y_i . Nech H_i pre $i \in \{0, 1, 2\}$ je graf, ktorý vznikne odstránením vrcholov x_i a y_i . Vytvoríme nový vrchol u a ďalšie vrcholy a_i, b_i, c_i také, že a_i susedí s x_{i+1}^0 a y_{i-1}^0 , ďalej b_i susedí s x_{i+1}^1 a y_{i-1}^1 a c_i susedí s a_i, b_i a novým vrcholom u . Konštrukcia je znázornená na obrázku 3.1. V článku je ďalej dokázané, že touto konštrukciou dostaneme snark s pokrývacím indexom aspoň 5.

Ak zoberieme $G_0 = G_1 = G_2 =$ Peresenov graf, potom dostávame WM_{34} .



Obr. 3.1: Konštrukcia windmill

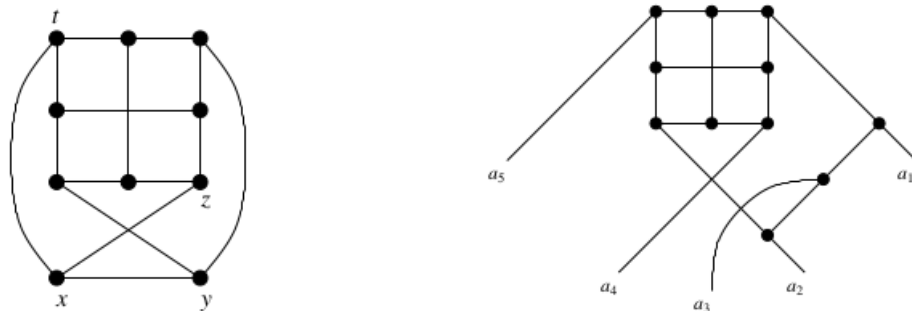
3.2 Konštrukcia "stromovitých" (treelike) snarkov

Teraz popíšeme konštrukciu od autorov M. Abreu, T. Kaiser, D. Labbate a G. Mazzucolo [1].

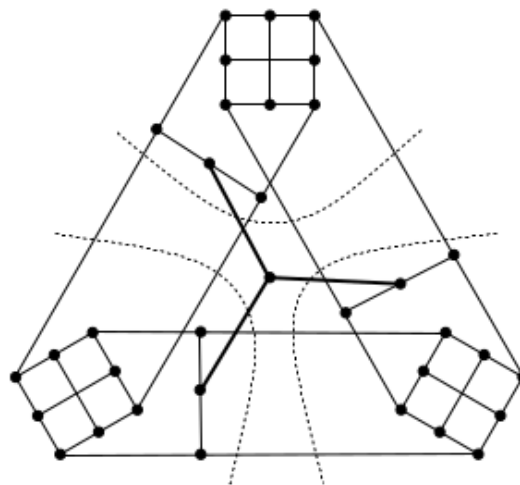
Na túto konštrukciu potrebujeme komponent F , čo je komponent, ktorý vznikne z Petersenovho grafu nasledovnými operáciami.

- Z Petersenovho grafu najprv odstránime vrchol x . Visiace hrany, ktoré ostanú, označme a_3, a_4, a_5 a ich susedov y, z, t .
- Na dvoch hranách incidentných s y pridáme po jednom novom vrchole.
- K novým vrcholom pridáme polpriamky a_1, a_2 .

Postup je znázornený na obrázku 3.2. Komponent F má päť visiacich hrán. Z toho sú dva konektory a jedna reziduálna hrana.



Obr. 3.2: Konštrukcia komponentu F



Obr. 3.3: Konštrukcia WM_{34} ako stromovitého snarku

Ďalej zdefinujeme *Halinov graf*. Skladá sa z planárneho stromu bez vrcholov stupňa 2 a z cyklu cez jeho listy. Zoberme kubický Halinov graf. Listy z cyklu nahradíme komponentami F tak, že polhrana a_3 bude pripojená ku stromu z Halinovho grafu. Na obrázku 3.3 je príklad takejto konštrukcie pre Halinov graf pozostávajúceho zo stromu s jedným vrcholom a z troch komponentov F . To, čo dostaneme je opäť graf WM_{34} . Autori článku ďalej dokazujú, že takouto konštrukciou vznikajú snarky s pokrývacím indexom aspoň 5.

Existuje aj všeobecnejšia konštrukcia tohoto typu [12]. Časti tejto konštrukcie budú detailnejšie popísané v ďalšej kapitole. V tejto konštrukcii sú použité tri druhy komponentov. Prvý je *dekolineátor*, čo je (2,2)-pól, ktorý vznikne odobraním cesty dĺžky 1 zo snarku s pokrývacím indexom aspoň 5. Druhý je (2,2,1)-pól B , ktorý vznikne odstránením cesty dĺžky 2 z grafu $K_{3,3}$. Spojením dekolineátora a B dostávame komponent F' , čo je tiež (2,2,1)-pól. Teraz zoberieme Halinov graf a rovnako ako v predchádzajúcej konštrukcii nahradíme jeho cyklus komponentmi F' . Takéto snarky nazývame *halinovské snarky*. Halinovské snarky majú pokrývací index aspoň 5.

Kapitola 4

Výsledky práce

V tejto kapitole popíšem výsledky a zistenia, ku ktorým sme dospeli pri testovaní a konštruovaní snarkov a tiež postupy, ktoré sme využili. Teoretické poznatky sme čerpali z práce [12].

4.1 Dekolineátor

Zoberme si ľubovoľný kubický (2,2)-pól a pozrime sa na jeho prechodovú reláciu pri farbení T -konfiguráciou. Označme farby na jednom konektore x_1, x_2 a na druhom y_1, y_2 . Z tokových vlastností vieme, že $x_1 + x_2 + y_1 + y_2 = 0$. Chceme teraz vedieť, aké prechodové relácie sú vôbec prípustné.

- Ak $x_1 + x_2 = 0$, znamená to, že na konektore je buď rohový, alebo stredový bod. Vtedy ale aj $y_1 + y_2 = 0$. Teda aj na druhom konektore bude rohový, alebo stredový bod. (2,2)-pól zachováva prechod **nula** \rightarrow **nula**.
- Ak vstupuje **os**, potom $x_1 + x_2 = 1111$ v \mathbb{Z}_2^4 . Jediná možnosť ako dostať 1111 na druhom konektore ako súčet dvoch farieb je opäť **os**. (2,2)-pól zachováva aj prechod **os** \rightarrow **os**.
- Ak x_1, x_2 tvorí polpriamku, potom $x_1 + x_2 = c$, kde c je rohový bod a (x_1, x_2, c) tvorí priamku. Aby boli zachované tokové podmienky, potom musí platiť aj $y_1 + y_2 = c$ potom ale $y_1 + y_2 + c = 0$. Keďže c je rohový bod, potom y_1, y_2 je polpriamka. (2,2)-pól zachováva aj prechod **polpriamka** \rightarrow **polpriamka**.
- Nech teraz x_1, x_2 je výška. Potom jeden z x_1, x_2 je rohový bod c_1 a druhý je stredom priamky c_2, c_3 . Body c_1, c_2, c_3 tvoria v tetraedrálnej konfigurácii trojuholník, v ktorom je výška x_1, x_2 . Potom $x_1 + x_2 = c_1 + c_2 + c_3 = y_1 + y_2$ a teda y_1, y_2 musí byť nektorá výška v trojuholníku c_1, c_2, c_3 .

- Ak x_1, x_2 je uhol, potom sú to dva stredové body blokov, ktoré majú spoločný rohový bod. Nech tento uhol leží v trojuholníku c_1, c_2, c_3 a c_1 je spomínaný spoločný vrchol. Potom $x_1 + x_2 = (c_1 + c_2) + (c_1 + c_3) = c_2 + c_3 = y_1 + y_2$. Teda y_1 a y_2 dávajú v súčte stredový bod. To môže nastať len vtedy, keď y_1, y_2 je úsečka z rovnakej priamky ako c_2, c_3 alebo uhol. Môže byť ten istý alebo iný, ktorý v súčte dáva bod $c_2 + c_3$. V (2,2)-póle môže nastať uhol \rightarrow uhol a aj uhol \rightarrow úsečka.
- Ak x_1, x_2 je úsečka, potom $x_1 + x_2$ je stredový bod. Preto tak isto ako v predchádzajúcom prípade y_1, y_2 môže byť úsečka alebo uhol. Teda môže nastať prechod úsečka \rightarrow uhol alebo úsečka \rightarrow úsečka.

Toto sú všetky možné prechody cez (2,2)-pól:

nula \rightarrow nula

os \rightarrow os

polpriamka \rightarrow polpriamka

výška \rightarrow výška

uhol \rightarrow uhol

uhol \rightarrow úsečka

úsečka \rightarrow uhol

úsečka \rightarrow úsečka

Túto prechodovú reláciu označíme Ω . Je to maximálna prechodová relácia, ktorú môže (2,2)-pól mať. Nie všetky prechody však musia nastať. Vnútoraná štruktúra (2,2)-pólu môže niektoré, možno aj všetky prechody vylúčiť.

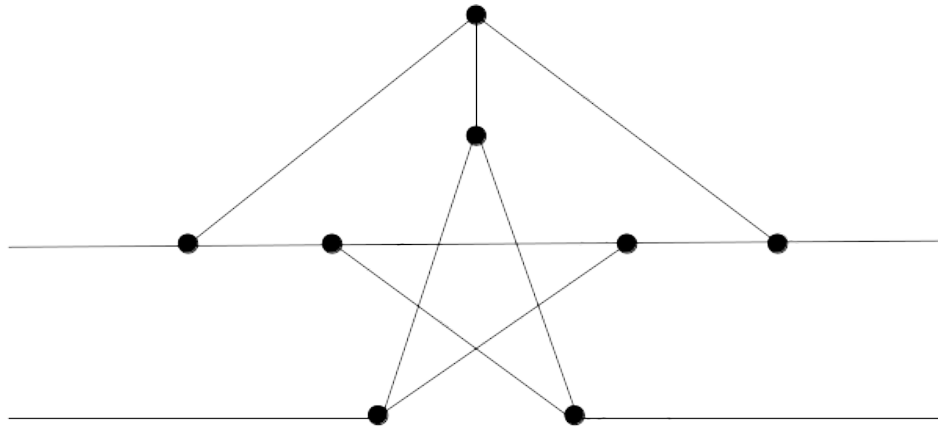
Zoberme snark, ktorý sa nedá zafarbiť tetraedrálou konfiguráciou. Po odstránení cesty dĺžky 1 dostávame (2,2)-pól. Takýto (2,2)-pól nazveme *dekolineátor*. Môžeme ho označiť X . Vyššie sme popísali, aké prechody môže mať. Vďaka tomu, že vznikol z T -nezafarbiteľného grafu, vieme o prechodoch viac.

Vieme, že nemôže mať prechod polpriamka \rightarrow polpriamka. Keby taký prechod mal, potom na oboch konektoroch by v súčte bola rovnaká farba. Túto farbu by sme mohli použiť v pôvodnom grafe na odstránenú cestu a mali by sme korektné T -zafarbenie. To ale nie je možné.

Podobne je to aj s prechodom úsečka \rightarrow úsečka. Keby dekolineátor mal takýto prechod, potom by na oboch konektoroch bol v súčte ten istý stredový bod. Z takejto situácie by sme opäť vedeli nájsť T -zafarbenie pôvodného grafu, čo je spor.

Z tejto menšej množiny prechodov môžeme pozorovať, že pri každom prechode $(x_1, x_2) \rightarrow (y_1, y_2)$ aspoň jedna z dvojíc $(x_1, x_2), (y_1, y_2)$ neleží na jednej priamke. Preto názov dekolineátor.

Na obrázku 4.1 je príklad dekolineátora, ktorý vznikne z Petersenovho grafu.



Obr. 4.1: Dekolineátor z Petersenovho grafu

4.2 Komponenty I a B

Ďalší komponent, s ktorým budeme pracovať je $(2,2)$ -pól I z obrázku 4.2. Komponent I vzniká odstránením cesty dĺžky 1 z grafu K_4 . Tento komponent má prechodovú reláciu:

nula \rightarrow nula

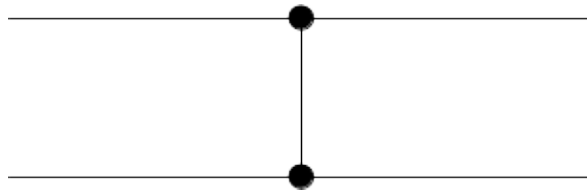
výška \rightarrow výška

polpriamka \rightarrow polpriamka

uhol \rightarrow úsečka

úsečka \rightarrow uhol

úsečka \rightarrow úsečka

Obr. 4.2: Komponent I

Pri farbení komponentu I tetraedrálnoú konfiguráciou je pri každom vrchole jedna hrana farby s Hammingovou váhou 2 a dve hrany s váhou farby 3. Dôležité pozorovanie je, že ak na jednom konektore I sú farby s Hammingovou váhou 3 potom na druhom konektore sú buď obe farby váhy 2 alebo 3, v závislosti od farby stredovej hrany. Opačne ak na jednom konektore sú dve farby váhy 2, potom na druhom musia byť dve farby váhy 3, pretože pri každom vrchole môže byť maximálne jedna farba váhy 2. Preto sa uhol zobrazí na úsečku. Túto vlastnosť neskôr využijeme pri konštrukcii.

Komponenty môžeme spájať za sebou pomocou ich konektorov. Skúsme sa pozrieť na prechodovú reláciu komponentu, ktorý vznikne spojením X s I a to s ďalším X (skrátene XIX). Zistíme, že prechodová relácia obsahuje prechody:

nula \rightarrow nula

výška \rightarrow výška

uhol \rightarrow uhol

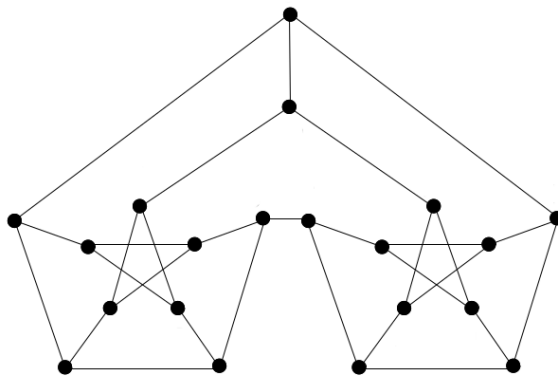
uhol \rightarrow úsečka

úsečka \rightarrow uhol

Je to teda opäť dekolineátor. Táto prechodová relácia je podmnožinou prechodovej relácie dekolineátora X . Chýba tu prechod $os \rightarrow os$, pretože komponent I nemá tento prechod. Priamky, na ktorých ležia vrcholy tvoriace os , nemajú žiaden spoločný bod. Preto by sa stredová hrana v komponente I nedala dofarbiť.

Toto nám umožňuje nahrádzať dekolineátor dvoma dekolineátormi a komponentom I . Nech Z je petersenovský dekolineátor. Potom nahradením dekolineátora Y v T-nezafarbiteľnom snarku dekolineátorom YIZ dostaneme nový T-nezafarbiteľný snark, ktorý má o 10 vrcholov viac.

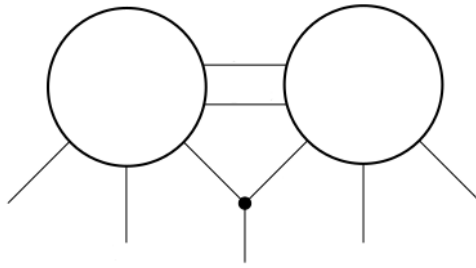
Keďže nás zaujímajú malé snarky, tak môžeme napríklad v Petersenovom grafe nahradiť petersenovský dekolineátor X dekolineátorom XIX . Dostaneme graf z obrázku 4.3. Problémom je, že tento graf má trojrez. Ako zvýšiť súvislosť si ukážeme neskôr.



Obr. 4.3: Rozšírenie Petersenovho grafu pomocou XIX

Komponent B vznikne z I pridaním vrcholu na stredovú hranu a pridaním polhrany z neho, alebo z grafu $K_{3,3}$ odstránením cesty dĺžky 2. B je $(2,2,1)$ -pól, má dva konektory a jednu visiacu hranu, ktorú nazývame reziduálna.

Operácia \odot -súčinu na $(2,2,1)$ -póloch funguje tak, že spojí konektor jedného komponentu s konektorom druhého komponentu a pridá nový vrchol, do ktorého pôjdu reziduálne hrany a nová polhrana. Operácia je znázornená na obrázku 4.4.

Obr. 4.4: Operácia \odot -súčinu na $(2,2,1)$ -póloch

B komponent povoľuje viac prechodov cez konektory, konkrétne týchto 18 prechodov:

- výška \rightarrow výška
- výška \rightarrow úsečka
- výška \rightarrow polpriamka
- uhol \rightarrow rohový bod
- uhol \rightarrow úsečka
- os \rightarrow úsečka
- rohový bod \rightarrow uhol
- rohový bod \rightarrow polpriamka
- úsečka \rightarrow výška
- úsečka \rightarrow uhol
- úsečka \rightarrow os
- úsečka \rightarrow polpriamka
- úsečka \rightarrow stredový bod
- polpriamka \rightarrow výška
- polpriamka \rightarrow rohový bod
- polpriamka \rightarrow úsečka
- polpriamka \rightarrow polpriamka
- stredový bod \rightarrow úsečka

Dôležité je, že každý prechod vynucuje nejakú farbu na reziduálnej hrane. Pri operácii \odot práve reziduálna hrana často vyvolá spor vo farbení. Môžeme si všimnúť, že komponent, ktorý vznikne spojením dekolineátora X a komponentu B je ekvivalentný

komponentu F z konštrukcie *treelike* snarkov. Komponent XB má tiež bohatú prechodovú reláciu. Povoľuje až 19 prechodov:

výška \rightarrow výška
 výška \rightarrow úsečka
 výška \rightarrow polpriamka
 uhol \rightarrow výška
 uhol \rightarrow uhol
 uhol \rightarrow os
 uhol \rightarrow rohový bod
 uhol \rightarrow úsečka
 uhol \rightarrow polpriamka
 uhol \rightarrow stredový bod
 os \rightarrow úsečka
 rohový bod \rightarrow uhol
 rohový bod \rightarrow úsečka
 rohový bod \rightarrow polpriamka
 úsečka \rightarrow rohový bod
 úsečka \rightarrow úsečka
 polpriamka \rightarrow uhol
 polpriamka \rightarrow úsečka
 polpriamka \rightarrow polpriamka

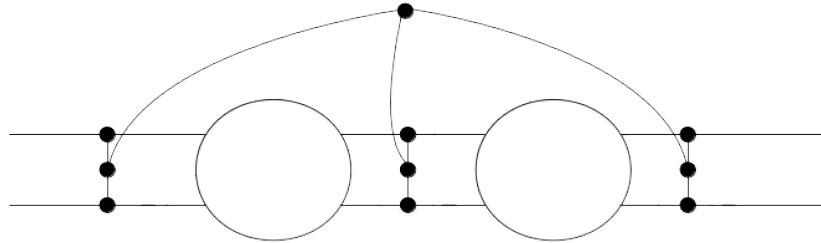
Ale komponent $(XB) \odot (XB)$ povoľuje len 7 prechodov:

výška \rightarrow úsečka
 uhol \rightarrow výška
 uhol \rightarrow úsečka
 uhol \rightarrow polpriamka
 rohový bod \rightarrow polpriamka
 úsečka \rightarrow polpriamka
 polpriamka \rightarrow polpriamka

Je to vďaka reziduálnej hrane.

4.3 Halinov dipól

Vpredchádzajúcej kapitole sme videli, že hoci komponent B má veľa prechodov, jeho kombinovaním s dekolineátormi sa prechodová relácia rýchlo redukuje. To uvidíme aj v ďalšom komponente. Pripojme k už spomínanému komponentu $(XB) \odot (XB)$ ďalšie B spôsobom ako je na obrázku 4.5. Tento komponent nazývame Halinov dipól.



Obr. 4.5: Halinov dipól

Halinov dipól ma len dva možné prechody. Sú to:

úsečka \rightarrow úsečka

polpriamka \rightarrow polpriamka

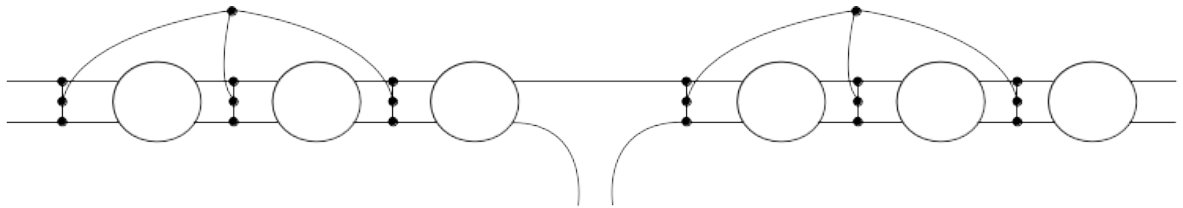
Môžeme sa pokúsiť tieto prechody ešte zredukovať. Už sme videli komponent, ktorý nám môže pomôcť. Je to dekolineátor. Ten totiž v prechodovej relácii nemá žiaden prechod s objektom *polpriamka*. Skúsme teda zapojiť dekolineátor X za Halinov dipól. Označme tento komponent Z a skúmajme jeho prechodovú reláciu. Tento komponent nie je symetrický. Ak do Halinovho dipólu vstupuje *polpriamka*, tak z neho aj vystupuje. Ak je na vstupe dekolineátora *polpriamka*, dekolineátor ju nevie dofarbiť, preto komponent Z nemá prechod z objektu *polpriamka*. Keď do Halinovho dipólu vstupuje *úsečka*, tak z neho aj *úsečka* vychádza. Dekolneátor X má prechod *úsečky* \rightarrow *uhol*. Iné vstupy pre Halinov dipól nemá zmysel rozoberať. Preto celý komponent Z má iba jeden jediný prechod, ktorým je *úsečka* \rightarrow *uhol*. Keby sme uzavreli komponent Z tak, že jeden konektor spojíme s druhým, získame snark, ktorý nemá T-zafarbenie. Je to práve vďaka sporu na konektoroch. Keď na konštrukciu komponentu Z ako dekolneátory použijeme dekolneátory z Petersenovho grafu, dostaneme už známy graf, WM_{34} . Pomocou skúmania prechodových relácií komponentov teda vieme vysvetliť nepokryteľnosť grafu WM_{34} štyrmi 1-faktormi.

4.4 Nedofarbitelné komponenty

Prechodovú reláciu komponentu Z vieme využiť aj inak. Vieme, že pri každom zafarbení komponentu Z musí byť na jednom konektore *uhol*. Je na konektore dipólu, ktorý sme pripojili ku Halinovmu dipólu. Mohli by sme k nemu pripojiť komponent,

ktorý nemá prechod z objektu uh_1 . Takým komponentom je napríklad opäť Halinov dipól. Spojme teda tieto dva komponenty do jedného komponentu, ktorý označíme HXH podľa komponentov, z ktorých vznikol ($Z = HX$). Vieme, že medzi HX a H nastáva konflikt v prechodovej relácii. Preto je tento komponent nedofarbitelný. Keď pripojíme k tomuto komponentu ľubovoľný komponent, tak nebude mať T-zafarbenie. Samozrejme nezaujímajú nás všetky doplnenia komponentu ale len tie, ktoré spĺňajú podmienky zaujímavých snarkov, napríklad, že majú dostatočnú súvislosť.

Komponent HXH má 2 konektory. Môžeme však vyrobiť o trochu väčší nezafarbitelný komponent s tromi konektormi, ktorý neobsahuje HXH ako podgraf. Na to si musíme všimnúť Hammingovu váhu farieb v prechodovej relácii komponentu HX . Na výstupe môže mať komponent HX len uh_1 . Objekt uh_1 je tvorený stredovými bodmi priamok tetraedrálnej konfigurácie. Tie majú Hammingovu váhu 2. Naopak, na vstupe môže mať len objekt úsečka, čo sú vrcholy s Hammingovou váhou 3. Môžeme spraviť nasledujúcu konštrukciu. Zoberme dva komponenty HX . Spojme jednu hranu z výstupu HX s hranou vstupu druhého HX . Konštrukcia je znázornená na obrázku 4.6.



Obr. 4.6: Komponent HXX

Dostaneme nový konektor pozostávajúci z jednej hrany výstupu HX a jednej zo vstupu druhého HX . Tento komponent je nezafarbitelný, pretože druhý komponent HX dostáva na vstupe farbu s Hammingovou váhou 2 a tú nevie spracovať. Tento krát však máme viac možností ako komponent doplniť.

Takto by sme vedeli vytvoriť ďalšie komponenty, ktoré sa nedajú dofarbiť pomocou tetraedrálnej konfigurácie. Toto boli najmenšie, ktoré sme našli. Každý takýto komponent je základom pre nekonečnú triedu snarkov s pokrývacím indexom aspoň 5.

4.5 Snark s vysokou nepárnosťou

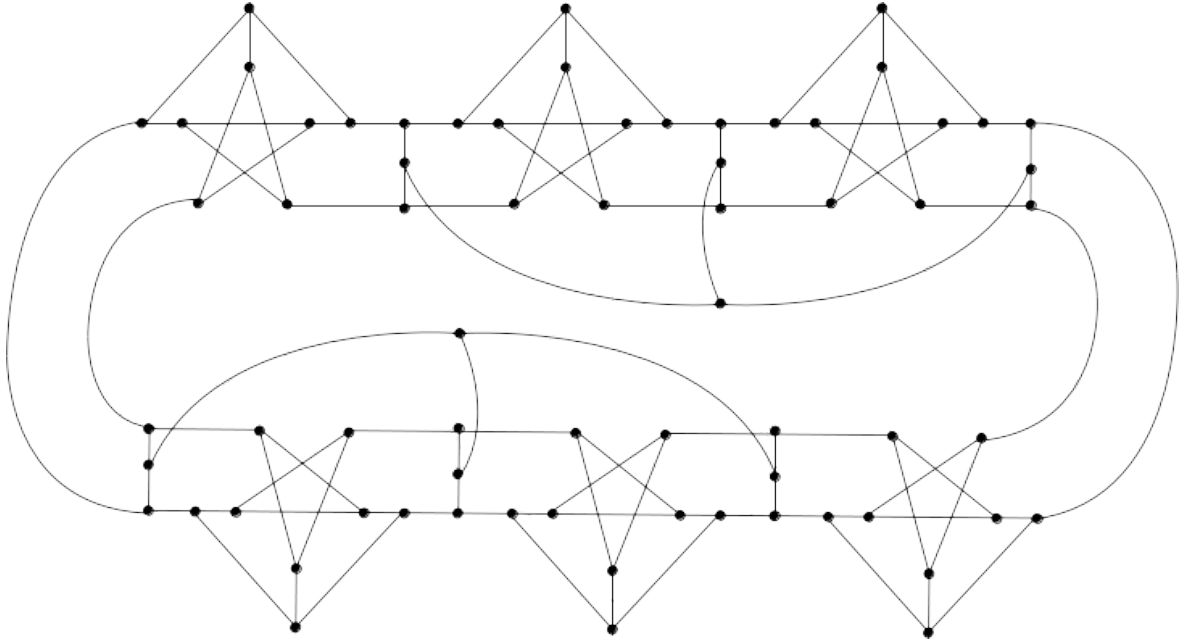
V tejto časti čerpáme teoretické poznatky z práce [9].

Nepárnosť je minimálny počet nepárnych kružníc v 2-faktore v bezmostovom kubickom grafe G , označujeme $\omega(G)$. Keďže každý kubický graf má párny počet vrcholov, potom aj nepárnosť musí byť párne číslo.

Rezistencia grafu G je minimálny počet vrcholov, ktoré treba z grafu G odstrániť, aby bol graf 3-zafarbitelný. Označujeme $\rho(G)$.

Obidva invarianty sú, podobne ako pokrývací index, mierami vzdialenosti snarku od 3-zafarbiteľného grafu. Vieme, že $\rho(G) \leq \omega(G)$. Nepoznáme však vzťah medzi $\tau(G)$ a $\rho(G)$ alebo $\omega(G)$.

Na obrázku 4.7 je snark, ktorý vznikne uzatvorením nedofarbiteľného komponentu $HXHX$ z predchádzajúcej časti. Tento snark je prvý známy snark s $\omega(G) > 2$ a $\tau(G) > 4$.



Obr. 4.7: Graf HXHX

4.6 Halinov dipól ako substitúcia

V predchádzajúcich častiach kapitoly sme prechodovú reláciu Halinovho dipólu využili na vytvorenie sporu. Teraz budeme hľadať podobnosť s iným komponentom, aby sme ich navzájom mohli substituovať.

V prechodovej relácii halonovho dipólu sú prechody:

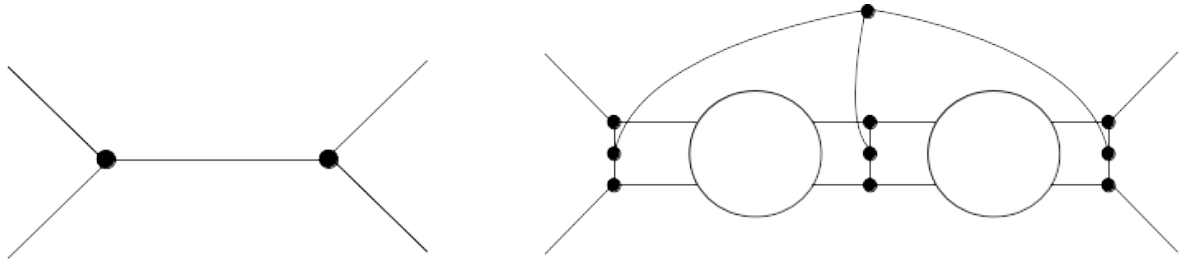
úsečka \rightarrow úsečka

polpriamka \rightarrow polpriamka

V oboch prípadoch je na každom konektore maximálne jedna farba s Hammingovou váhou 2. Pri prechode úsečka \rightarrow úsečka je súčet farieb na jednom konektore rovný súčtu farieb na druhom konektore. Je to stredový vrchol priamky. Môžeme si to predstaviť aj tak, že celý Halinov dipól okrem konektorov má ako farbu tento stredový vrchol. Podobne pri prechode z polpriamky to bude jeden rohový vrchol.

Keď skúmame prechodovú reláciu cesty dĺžky 1 (alebo hrany s jej koncovými vrcholmi) obr. 4.8, zistíme že je rovnaká ako prechodová relácia Halinovho dipólu. Tento

poznatok nám dáva nové možnosti na konštrukciu snarkov s pokrývacím indexom aspoň 5.



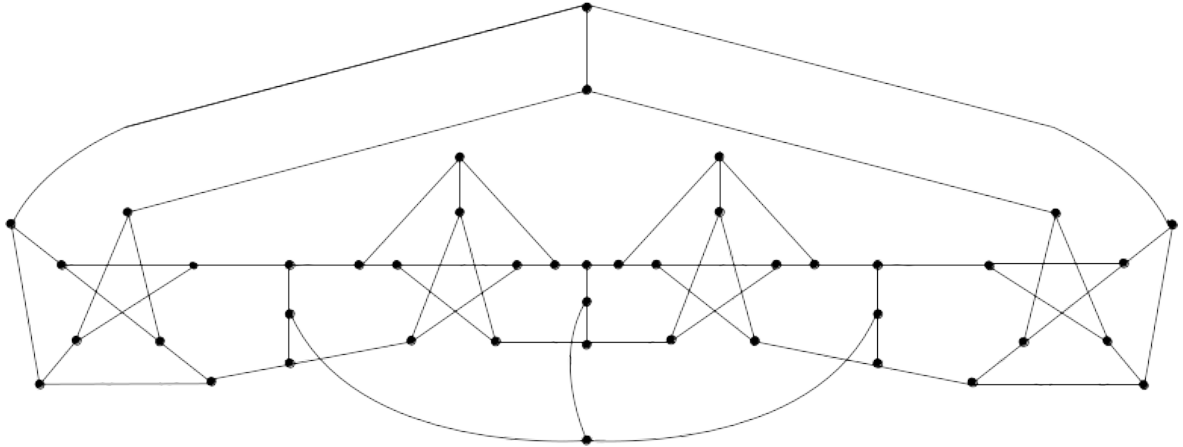
Obr. 4.8: Cesta dĺžky 1 a Halinov dipól

Ak zoberieme ľubovoľnú cestu dĺžky 1 v ľubovolnom snarku s pokrývacím indexom aspoň 5, tak aj graf, ktorý vznikne nahradením tejto cesty Halinovým dipólom bude mať pokrývací index aspoň 5. Halinov dipól nemusí mať presne tie isté prechody ako cesta dĺžky 1. Môže mať aj menej, ak v konštrukcii Halinovho dipólu použijeme dekolineátory, ktoré niektoré prechody ešte zredukujú. Keď chceme konštruovať malé snarky, tak môžeme zobrať Halinov dipól s pretersenovskými dekolineátormi a nahradiť ním cestu v Petersenovom grafe. Táto operácia zobrebie 2 vrcholy z Petersenovho grafu a pridá 26 vrcholov. Celkom 34 vrcholov. Dostávame opäť WM_{34} . Dôvod nezafarbitelnosti WM_{34} je tentokrát prenesený z Petersenovho grafu substitúciou, ktorá túto vlastnosť zachováva.

Substitúcia Halinovým dipólom má aj inú dobrú vlastnosť. Keďže nahradíme jednu hranu niečím súvislým, tak súvislosť grafu neklesne, ale môže stúpať. Takýmto posilnením hrán, ktoré sú v malých rezoch môžeme vyrobiť súvislejší graf. Môžeme sa vrátiť k už spomínanému grafu z obrázku 4.3. Tento graf má dva netriviálne trojrezy. Oba majú jednu spoločnú hranu. Túto hranu spolu s jej vrcholmi môžeme nahradiť Halinovým dipólom. Dostávame 44 vrcholový snark s pokrývacím indexom aspoň 5 a bez trojrezov 4.9.

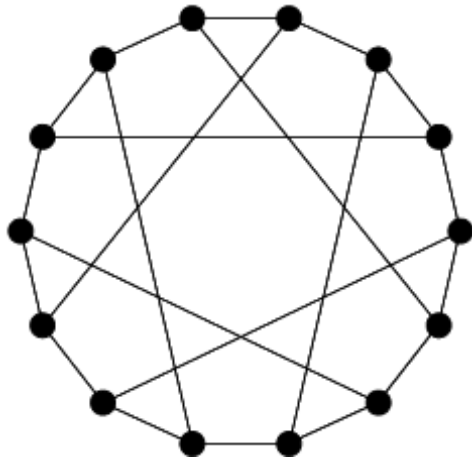
4.7 Bipartitné vložky

Už sme videli komponenty, ktoré menia prechodovú reláciu a tiež komponenty, ktoré majú rovnaké prechodové relácie. Teraz budeme hľadať komponenty, ktoré zachovávajú geometrické typy dvojíc. S týmto cieľom sme skonštruovali viacero malých komponentov, ktorých prechodové relácie sme potom otestovali. Najskôr sa nám podarilo nájsť (2,2)-póly na 10 vrchoch so zachovávajúcou prechodovou reláciou. Tie však obsahovali cyklus dĺžky 4. Snarky, ktoré majú cyklus dĺžky 4 nie sú pre nás príliš zaujímavé. Nakoniec sa nám podarilo nájsť (2,2)-pól na 12 vrchoch, ktorý neobsahoval cyklus dĺžky 4 a mal konštantnú prechodovú reláciu. Teda ak na vstupe bola úsečka, tak aj

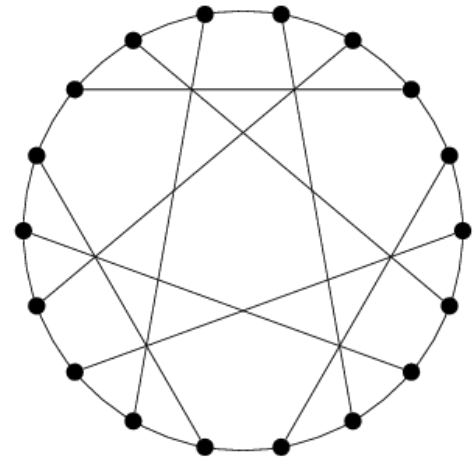


Obr. 4.9: Nahradenie hrany Halinovým dipólom

na výstupe bola úsečka. A podobne to platilo pre všetky triedy dvojíc vrcholov tetraedrálnej konfigurácie. Chceli sme vedieť, ako takýto komponent vzniká. Ukázalo sa, že obsahuje veľa cyklov dĺžky 6. Existuje graf na 14 vrcholoch, ktorý tiež obsahuje takéto cykly. Volá sa Heawoodov graf 4.10. Zistili sme, že náš 12 vrcholový komponent vznikne odstránením cesty dĺžky 1 z Heawoodovho grafu. Potom sme túto vlastnosť otestovali aj na Pappusovom grafe 4.11. Tento graf je štruktúrou podobný Heawoodovmu grafu. Má 18 vrcholov. Komponent, $(2,2)$ -pól, ktorý vznikne odstránením cesty dĺžky 1 z Pappusovho grafu sme otestovali na prechodovú reláciu. Tento $(2,2)$ -pól má tiež zachovávajúcu prechodovú reláciu.



Obr. 4.10: Heawoodov graf



Obr. 4.11: Pappusov graf

Otázka je koľko takýchto grafov existuje. Heawoodov aj Pappusov graf sú kubické, bipartitné grafy. Preto má zmysel pozrieť sa na všetky také grafy. Podarilo sa dokázať, že $(2,2)$ -pól, ktorý vznikne z bipartitného kubického grafu odstránením cesty dĺžky 1 bude mať zachovávajúcu prechodovú reláciu [12].

V dôkaze sa najprv ukáže, že ak z bipartitného kubického grafu odstránime vrchol

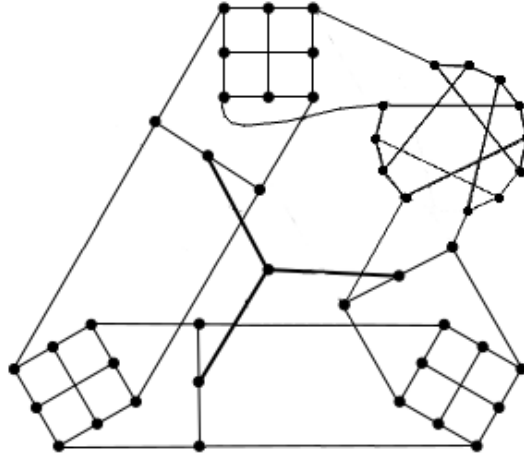
v , tak v každom T farbení zvyšného 3-pólu bude na visiach hranách priamka v tetraedrálnej konfigurácii. Potom sa ukáže, že vďaka tomu po odstránení nejakého suseda vrcholu v vznikne $(2,2)$ -pól so zachovávajúcou prechodovou reláciou.

Pretože tieto komponenty vznikajú z bipartitných grafov a môžeme ich ľubovoľný počet krát vkladať do snarkov, nazvali sme ich spoločne *bipartitné vložky*. Ako sme už spomínali, niektoré grafy obsahujú cyklus dĺžky 4. To nám nie vždy prekáža, pretože cestu dĺžky 1 môžeme vybrať tak, aby vzniknutý $(2,2)$ -pól neobsahoval cyklus dĺžky 4. Problém je, keď takých cyklov je v grafe viac a nedajú sa všetky zrušiť odstránením jednej cesty dĺžky 1. Pre nájdenie najmenšieho vyhovujúceho bipartitného grafu sme stiahli z databázy grafov (<https://hog.grinvin.org/>) malé bipartitné kubické grafy. Spracovali sme ich pomocou grafového programu Sage (<http://www.sagemath.org/>). Hľadali sme v nich grafy, z ktorých sa dá odstrániť cesta dĺžky 1 tak, aby nezostal v grafe žiaden cyklus dĺžky 4. Najmenší taký graf je spomínaný Heawoodov graf na 14 vrcholoch. Potom ďalšie sú na 16, 18 a 20 vrcholoch.

Bipartitné vložky nám môžu pomôcť pri konštrukcii snarkov s pokrývacím indexom aspoň 5. Zoberme snark s pokrývacím indexom aspoň 5. Nech v tomto snarku existujú 2 hrany, ktorých rozrezaním vznikne $(2,2)$ -pól, ktorého prechodová relácia spôsobuje pokrývací index aspoň 5 snarku. To znamená, že po spätnom spojení týchto dvoch hrán dostaneme na tomto mieste spor v prechodovej relácii. Potom na toto miesto môžeme zapojiť za sebou ľubovoľne veľa bipartitných vložiek a vzniknutý graf bude mať tiež pokrývací index aspoň 5. Keď nás zaujímajú najmenšie snarky s pokrývacím indexom aspoň 5, tak môžeme zobrať WM_{34} a bipartitnú vložku z Heawoodovho grafu. V časti 4.3 sme ukázali, že komponent Z má iba prechod úsečka $\rightarrow uho1$. Môžeme za tento komponent zapojiť bipartitnú vložku a uzavrieť celý tento komponent ako na obrázku 4.12. Dostaneme $34 + 12 = 46$ vrcholov.

V prílohe je súbor, ktorý obsahuje kubické bipartitné grafy s obvodom aspoň 6 od 14 do 20 vrcholov. Keďže majú dostatočný obvod, môžu byť použité na vytvorenie bipartitných vložiek.

Bipartitné vložky môžeme naozaj vložiť len na miesto, kde chceme zachovať prechodovú reláciu. Inak pokrývací index snarku nemusí ostať 5. Zoberme napríklad Petersenov graf. Rozrežeme niektoré dve hrany vo vzdialenosti 1 a vložme tam bipartitnú vložku z Heawoodovho grafu. Dostávame graf na $10 + 12 = 22$ vrcholov. Tento graf bude mať stále dostatočujúcu súvislosť. Nemôže však mať pokrývací index aspoň 5, pretože už vieme že do 34 vrcholov existujú iba dva také snarky, Petersenov graf a WM_{34} . Prečo teda tento 22 vrcholový graf nemá pokrývací index aspoň 5? Dokonca tento graf ani nie je snark, pretože sa dá zafarbiť tromi farbami. Problém je, že nazafarbiteľnosť Petersenovho grafu tetraedrálou konfiguráciou teraz nevieme vysvetliť pomocou prechodovej relácie, tak ako sme ju zdefinovali. Problém je už to, že pri prechodovej relácii nás nezaujíma poradie farieb na konektore. Keď rozpojíme dve hrany v Petersenovom

Obr. 4.12: WM_{34} s bipartitnou vložkou

grafe, tak $(2,2)$ -pól, ktorý vznikne má takúto prechodovú reláciu:

výška \rightarrow výška

uhol \rightarrow uhol

uhol \rightarrow úsečka

nula \rightarrow nula

úsečka \rightarrow uhol

úsečka \rightarrow úsečka

polpriamka \rightarrow polpriamka

Môžeme nájsť prechody napríklad $\text{polpriamka} \rightarrow \text{polpriamka}$. Mohlo by sa nám zdať, že keď spojíme rozpojené hrany späť, tak s týmto prechodom je Petersenov graf T-zafarbiteľný. To však nie je pravda. Síce existoval prechod $\text{polpriamka} \rightarrow \text{polpriamka}$, ale na konektoroch boli buď rôzne polpriamky, alebo ak boli rovnaké, tak farby na konektore boli v rôznom poradí.

Iná situácia nastáva, keď rozpojíme v Petersenovom grafe hrany, ktoré majú spoločný vrchol. Takýto $(2,2)$ -pól má v prechodovej relácii len prechod $\text{úsečka} \rightarrow \text{uhol}$. Keď hrany späť spojíme tak máme spor a preto nebude Petersenov graf T-zafarbiteľný. Na toto miesto môžeme dať bipartitnú vložku. Tiež dostaneme 22 vrcholový snark. Problémom je 3-rez. Ten musíme odstrániť substitúciou cesty dĺžky 1 Halinovým dipólom. Tým to spôsobom vieme skonštruovať 46-vrcholový snark, ale nie menší.

Iný pohľad na Petersenov graf je, že sa skladá z petersenovského dekolineátora a cesty dĺžky 1. Cesta dĺžky 1 má prechody:

úsečka \rightarrow úsečka

polpriamka \rightarrow polpriamka

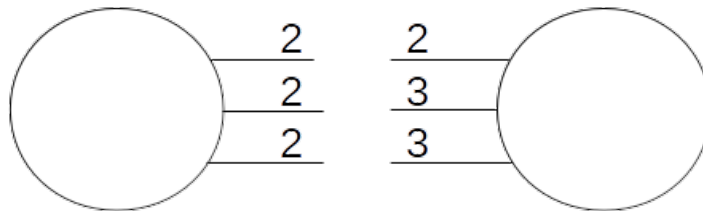
Petersenovský dekolineátor tieto prechody nemá. Preto ich spojením dostávame T-nezafarbiteľný snark. Medzi spojenie konektorov dekolineátora a cesty dĺžky 1 môžeme vkladať bipartitné vložky a tak vyrábať snarky s pokrývacím indexom aspoň 5. Vyrobíme však aj trojrezy, ktorých sa potom musíme zbaviť.

4.8 Priamky a antipriamky

V tejto kapitole ukážeme ďalšiu techniku ako konštruovať snarky s pokrývacím indexom aspoň 5. V predchádzajúcich kapitolách sme spájali (2,2)-póly. Teraz budeme spájať 3-póly. Tu už nebudeme skúmať prechodovú reláciu, ale bude nás zaujímať Hammingova váha farieb na visiach hranách a tiež, či tvoria priamku alebo antipriamku v tetraedrálnej konfigurácii.

Už sme videli, že farby na visiach hranách 3-pólu, ktorý vznikne odstránením vrchola z bipartitného kubického grafu, pre každé T-farbenie tvoria priamku v tetraedrálnej konfigurácii. To sú dva rohové vrcholy a jeden stredový vrchol z tetraedrálnej konfigurácie. Hammingove váhy farieb na visiach hranách sú 3, 2, 3.

Keď zoberieme snark s pokrývacím indexom aspoň 5 a odstránime vrchol, tak dostaneme 3-pól, ktorý pri každom T-farbení má na visiach hranách farby tvoriace antipriamku. Je to preto, lebo pri farbení 3-pólu tetraedrálou konfiguráciou musí byť súčet farieb na visiach hranách rovný 0, pretože je to tok. To znamená, že farby tvoria priamku alebo antipriamku v tetraedrálnej konfigurácii. Keby tvorili priamku, tak potom pôvodný snark, z ktorého 3-pól vznikol by sa dal dofarbiť tetraedrálou konfiguráciou. To ale nie je možné, pretože tento snark mal pokrývací index aspoň 5. Preto na visiach hranách môže byť len antipriamka. Hammingove váhy farieb na visiach hranách sú preto 2, 2, 2.

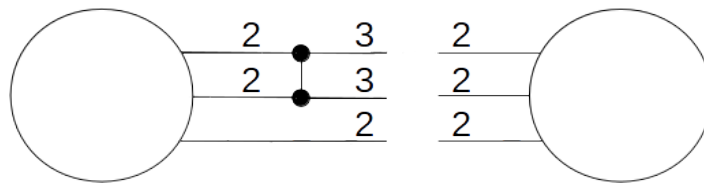


Obr. 4.13: Spojenie 3-pólov zo snarku s pokrývacím indexom aspoň 5 a bipartitného grafu

Spojením 3-pólu z kubického bipartitného grafu a 3-pólu zo snarku s pokrývacím indexom aspoň 5 dostávame T-nezafarbiteľný snark uvedený na obrázku 4.13. Je to vďaka rôznym Hammingovým váham, ktoré jednotlivé 3-póly vedú dofarbiť. Jediný problém je, že tento snark má trojrez. Toho sa vieme zbaviť pomocou Halinovho dipólu.

Môžeme sa vrátiť ku komponentu I z časti 4.2. Tento komponent má len 2 vrcholy a 2 konektory. Komponent I má takú vlastnosť, že ak na jednom konektore sú farby s Hammingovou váhou 2, tak potom na druhom konektore sú farby váhy 3.

Zoberme spomínaný 3-pól zo snarku s pokrývacím indexom aspoň 5. Pripojme na dve jeho visiace hrany komponent I . Dostávame nový 3-pól, ktorý pri kažom T-farbení má na visiacich hranách farby váh 2, 3, 3. Zoberme ďalší 3-pól skonštruovaný zo snarku s pokrývacím indexom aspoň 5 odstránením jedného vrchola. Spojením týchto dvoch 3-pólov dostávame opäť T-nezafarbiteľný snark z obrázku 4.14. Je to preto, lebo tieto 3-póly majú na visiacich hranách farby rôznych Hammingových váh. Jeden 2, 2, 2 a druhý 2, 3, 3. Opäť sa potrebujeme vysporiadať s trojrezom.



Obr. 4.14: Spojenie dvoch 3-pólov zo snarkov s pokrývacím indexom aspoň 5 a komponentu I

Aby sme odstánili trojrez, tak stačí nahradiť spodnú hranu z trojrezu z obrázku aj s príslušnými vrcholmi Halinovým dipólom. Ak oba 3-póly vzniknú z Petersenovho grafu, tak dostávame už spomínaný graf z obrázku 4.3.

Kapitola 5

Snark Master

Snark Master je program, ktorý nám pomáha skúmať pokrývaci index snarkov pomocou steinerovských konfigurácií. Je napísaný v jazyku Scala. Pre grafické používateľské rozhranie využíva Java Swing. Na testovanie vlastností grafov využíva knižnicu CafeSat. Je to knižnica riešiaci SAT problém napísaná v jazyku Scala. Snark Master poskytuje funkcionality pre testovanie zafarbiteľnosti grafu vybranou steinerovskou konfiguráciou a tiež testovanie prechodových relácií komponentov grafu.

V nasledujúcich častiach popíšeme súčasti programu, ako sa program používa a s akými dátami vie pracovať.

5.1 Funkcionalita a použitie Snark Mastra

Okno aplikácie Snark Master obsahuje 3 časti. Najprv sú to dve záložky s názvom "Coloring" a "Transition". Potom jedno textové pole, do ktorého Snark Master vypisuje výsledky všetkých akcií vyvolaných užívateľom.

Záložka "Coloring" umožňuje testovať zafarbiteľnosť grafov steinerovskými konfiguráciami. Užívateľ najskôr musí vybrať súbor, ktorý obsahuje grafy na testovanie. Program vie spracovať len súbory v takzvanom bratislavskom formáte pre kubické grafy.

Súbor v tomto formáte môže obsahovať komentáre. Sú to riadky, ktoré začínajú symbolom "{" a končia symbolom "}". Komentáre môžu byť v ľubovoľných riadkoch. Teraz popíšeme riadky okrem komentárov. V prvom riadku je počet grafov v súbore. Pre každý graf je v súbore $m + 2$ riadkov, kde m je počet vrcholov grafu. Prvý riadok je poradové číslo grafu v súbore, číslované od 1. V ďalšom riadku je počet vrcholov grafu m . Nasleduje m riadkov. V i -tom riadku pre $0 \leq i < m$ sú čísla vrcholov susediacich s vrcholom i oddelené medzerou. Vrcholy sú číslované od 0.

Ak sa programu nepodarí spracovať grafy zo súboru, tak vypíše chybu. Program tiež testuje, či každý vrchol je stupňa 3, tým sa predchádza chybám pri ručnom zapisovaní

grafu do bratislavského formátu. Ak táto podmienka nie je splnená pre nejaký vrchol, tak program vypíše číslo grafu, číslo vrcholu a počet susedov tohto vrcholu. Ak sa podarilo grafy spracovať, tak sa vypíše správa o úspešnom načítaní grafov a názov súboru sa objaví v kolonke "Graphs".

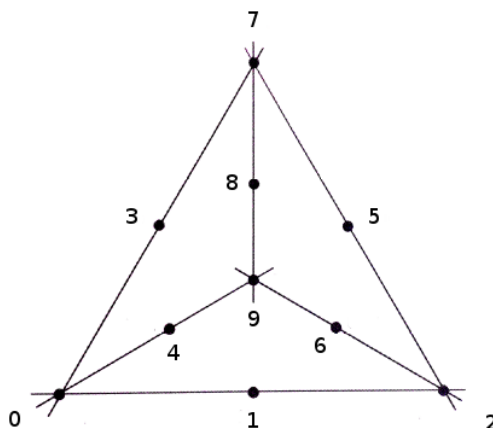
Pred testom je potrebné vybrať steinerovskú konfiguráciu, ktorou zafarbitelnosť chceme testovať. Užívateľ má na výber z troch konfigurácií. Tetraedrálna konfigurácia (obrázok 2.2), rozšírená tetraedrálna konfigurácia vznikne z T -konfigurácie, pridaním 4 antipriamok, a posledná je triviálna konfigurácia, pomocou ktorej je možné testovať, či sa graf dá hranovo zafarbiť tromi farbami.

Potom je možné spustiť test. Ak nie je vybraný žiaden graf, tak sa vypíše chyba. Inak sa postupne na všetkých grafoch z vybratého súboru spustí test na zafarbitelnosť vybranou konfiguráciou. Program vypíše začiatok testovania aj výsledok testu pre každý graf. Ak program pre graf nájde zafarbenie, tak pre každú hranu grafu vypíše bod z konfigurácie, ktorým program hranu ofarbil.

Záložka "*Transition*" je určená pre testovanie prechodových relácií. Tu sa netestujú celé grafy, ale len komponenty grafov. Pre zápis komponentov máme ďalší formát. Súbor môže obsahovať také isté komentáre ako pri bratislavskom formáte. Súbor popisuje však len jeden komponent. Komponent je graf z visacími hranami. Preto potrebujeme pri zápise komponentu vytvoriť fiktívne nové vrcholy, pre každú visiacu hranu. Visaca hrana bude hrana medzi vrcholom komponentu a fiktívnym vrcholom. V prvom riadku je počet vrcholov n komponentu (aj s fiktívnymi). Nasleduje n riadkov, ktoré rovnako ako v bratislavskom formáte popisujú susedov v grafe. V tomto formáte sú zapísané aj konektory, na ktorých chceme sledovať prechodovú reláciu. Preto na ďalšom riadku je číslo m počet konektorov. Potom pre každý konektor je riadok s dvoma fiktívnymi vrcholmi, ktorých incidentné hrany tvoria konektor.

Program testuje len prechodové relácie pri zafarbení komponentu tetraedrálnou konfiguráciou. Po spustení testu program nájde všetky zafarbenia. Po skončení vypíše dva výsledky. Prvým sú všetky možnosti zafarbenia visiacich hrán. Pre každú možnosť jeden riadok. Na prvý riadok ešte pred tým vypíše fiktívne vrcholy v poradí, v akom bude vypisovať ich zafarbenia. Táto časť výstupu môže byť veľmi veľká. Druhá časť je jednoduchšia. Program zoberie všetky možné zafarbenia konektorov podľa vstupného súboru. Pre každé zafarbenie konektoru zistí, aké geometrické typy tvoria farby v konfigurácii, to znamená triedy z konca kapitoly o steinerovských systémoch. Ak má komponent m konektorov, tak program vypíše všetky m -tice tried dvojíc vrcholov.

Pre farbenie tetraedrálnou konfiguráciou používa Snark Master bloky s číslami farieb ako na obrázku 5.1.



Obr. 5.1: Označenie farieb v T-konfigurácii

5.2 Scala

Scala je programovací jazyk, ktorý vznikol v roku 2003 na uvierzite v Lausanne. Je to silno typovaný jazyk, ktorý spája objektovo orientované programovanie s funkcionálnym programovaním. Kompiluje sa do bajt kódu a beží na JVM. Kompilátor Scaly obsahuje aj časť Java kompilátoru. Java a Scala triedy sa môžu miešať, odkazovať na seba a používať svoje rozhrania.

Názov Scala vznikol zo zpojenia *Scalable Language*. V Scale sa dajú písať silné jednoriadkové skripty ale aj veľké systémy. Je to objektovo orientovaný jazyk. Dokonca viac ako Java. Neexistujú static metódy ani metódy s návratovým typom void. Všetko v Scale je objekt a všetky operácie sú volania metód. Viac tradičných dizajnových vzorov je zakomponovaných v jazyku. Tiež podporuje implicitné triedy a funkcie.

Je to tiež funkcionálny jazyk. Funkcia je tiež objekt. Dajú sa ľahko spájať. Funkcie môžu brať ďalšie funkcie ako parametre. Scala má efektívne nemenné (immutable) dátové štruktúry. Má tiež aj implemencie dátových štruktúr, ktoré môžu meniť stav, ale preferované sú nemenné.

Táto kombinácia objektového a funkcionálneho programovanie pomáha vývoju distribuovaných a paralelných systémov. Príkladom ja Apache Spark (<http://spark.apache.org/>).

5.3 Akka

Akka je knižnica napísaná v jazyku Scala, ktorá obsahuje nástroje pre jednoduché písanie distribuovaných a paralelných systémov. Je to open-source projekt. Je základom viacerých moderných frameworkov.

Základný a najsilnejší nástroj je *actor model*. Umožňuje vytvárať actorov, čo sú triedy, ktoré bežia vo svojom vlastnom vlákne. Actory bežia paralelne, nezávisle od

seba.

Iným objektom nie je možné pristupovať k funkciám alebo atribútom actora. Jediná možnosť, ako komunikovať s actorom je pomocou správ. Je to podobný spôsob komunikácie ako medzi procesmi v operačných systémoch. Každý actor má schránku na správy. Actor musí implementovať funkciu *receive*, pomocou ktorej sa spracovávajú správy so schránky. *Pattern matching* je konštrukcia v jazyku Scala, pomocou ktorej sa implementuje funkcia *receive*. Je to náhrada *switch-u* z Javy, *pattern matching* v Scale však ponúka väčšiu flexibilitu. V konštrukcii sú výrazy a k nim aj príkazy, ktoré sa majú vykonať ak daný výraz vyhovuje porovávanému objektu. Výrazy môžu obsahovať konštanty, stromové konštrukcie tried, typové testy a rôzne iné.

Actory tvoria hierarchiu. Na začiatku sa v actor systéme vytvorí hierarchia troch základných actorov. Z nich jeden je koreňový actor pre všetkých actorov vytvorených používateľom a druhý pre actorov potrebných pre actor systém. Actor, ktorý vytvorí actora sa stáva jeho rodičom v hierarchii. Ak je actor vytvorený mimo nášho actora, tak rodičom mu je koreňový actor pre používateľských actorov.

Akka sa riadi heslom "Let it crash". Týka sa to hlavne veľkých systémov, ktoré sú spustené nepretržite dlhú dobu. Ak v nejakom komponente nastane chyba, je lepšie konkrétne časti reštartovať, ako udržiavať v zlom stave. S týmto nám pomáha hierarchická štruktúra actorov. Rodič monitoruje svojich synov a môžeme zdefinovať stratégiu, akým spôsobom vyriešiť chyby z jeho synov. Jedným zo spôsobov je reštartovať syna.

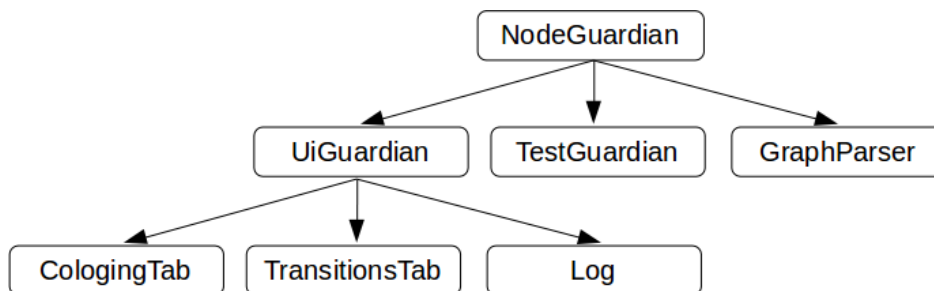
5.4 Implementácia

Implementáciu Snark Mastra môžeme rozdeliť do dvoch častí. Prvá je implementácia používateľského rozhrania. To je implementované pomocou Java Swing. Na jeho vytvorenie sme použili nástroj, pomocou ktorého sme mohli jednoducho rozmiestniť komponenty, ktoré potrebujeme a nástroj vygeneroval zdrojový kód. Všetok kód potrebný na vytvorenie komponentov je v triede *MainForm*. Okrem toho sme potrebovali do vygenerovaného kódu pridať, ktoré *radioButton-y* tvoria jednu skupinu.

Druhá časť sú všetky riadiace a testovacie prvky. Tie sú naimplementované pomocou actorov. Hierarchia týchto actorov je znázornená na obrázku 5.2. Actory komunikujú pomocou správ a väčšiu cez svojho rodiča. Teraz popíšem úlohy jednotlivých actorov.

NodeGuardian actor je prvý actor, ktorý Snark Master po vytvorení actor systému vytvorí. Po vytvorení vytvorí ďalších troch actorov, *uiGuardian*, *testGuardian* a *graphParser*. Slúži tiež ako sprostredkovateľ komunikácie medzi týmito tromi actormi. To znamená, že prijíma správy a podľa typu správy rozhodne, komu ju prepošle.

UIGuardian je actor zodpovedný za používateľské rozhranie. Najprv vytvorí používateľské rozhranie pomocou vytvorenia inštancie triedy *MainForm* spomínanej vyššie.



Obr. 5.2: Hierarchia actorov

Potom vytvorí ďalšie 3 actory pre jednotlivé komponenty. Slúži na komunikáciu medzi nimi a tiež s ostatnými actormi cez nodeGuardiana.

Log actor spravuje textové pole určené pre logy. Keď dostane správu, pozrie sa na jej typ a podľa toho vyberie akým spôsobom zapíše správu. Zapisuje jednoduché informácie o chode programu, výnimky a výsledky všetkých spustených testov.

ColoringTab actor sleduje komponenty na záložke *Coloring* a reaguje na vstup od užívateľa. Keď používateľ vyberie súbor s grafmi, pošle ho svojmu rodičovi ako požiadavku na spracovanie grafov. Ak sa vráti správa o úspešnom spracovaní súboru, tak sa zobrazí názov prečítaného súboru. Ak užívateľ chce spustiť test zafarbiteľnosti, actor sa pozrie ktorú konfiguráciu používateľ vybral a pošle rodičovi požiadavku na otestovanie grafu spolu s grafom a konfiguráciou.

TransitionsTab actor sleduje komponenty na záložke *Transitions*. Používateľ má možnosť vybrať súbor s komponentom. Actor potom pošle požiadavku na spracovanie súboru. Používateľ ma na výber dva testy. Prvý vyhľadá všetky prechodové relácie načítaného komponentu a vypíše ich. Druhý testuje, či komponent má aspoň jeden možný prechod. Tento test je užitočný pri veľkých komponentoch, ktoré boli skonštruované so zámerom nájsť komponent, ktorý sa nedá zafarbiť. Pri spúšťaní testu actor najskôr overí, či užívateľ vybral komponent a potom pošle požiadavku na testovanie so všetkými parametrami.

GraphParser je actor, ktorý dostáva požiadavky na spracovanie súborov. V požiadavke je aj formát súboru. Zatiaľ však používame len bratislavský formát a pre komponenty máme už opisovaný formát. Formáty sú naprogramované ako singletony s funkciou *parse(file)*. V prípade, že pri spracovávaní súboru nastala chyba, napríklad pre zlý formát, tak actor pošle späť chybu.

TestGuardian actor prijíma požiadavky na spustenie testov. V požiadavke je aj to, ktorý test sa má vykonať. Actor ho spustí a po jeho dokončení vráti späť výsledok testu.

Prvý test je test na zafarbiteľnosť steinerovským systémom. Pri všetkých testoch sa využíva knižnica pre riešenie SAT problému. Na začiatku testu sa vytvoria premenné pre každú hranu a farbu steinerovského systému, ktorých platnosť znamená, že

hrana je zafarbená danou farbou. Následne sa vygenerujú podmienky, ktoré popisujú farbenie steinerovským systémom grafu. ich splniteľnosť znamená zafarbiteľnosť grafu steinerovskou konfiguráciou. Zoberie sa jedno možné ohodnotenie premenných. Z nich sa zistí, aké sú farby hrán v danom farbení a vráti sa ako výsledok.

Druhý test je zisťovanie prechodových relácií komponentu. Tak ako aj v predchádzajúcom teste sa využíva transformácia na SAT. Tento test je však komplikovanejší. Pre každé možné zafarbenie visiaticich hrán sa vyrobí menší test. Pre tento menší test sa pridávajú aj podmienky, ktoré vynucujú konkrétne zafarbenie na visiaticich hranách. Menšie testy nebežia za sebou, ale keďže sú nezávislé, tak môžu bežať paralelne. Všetky sa spustia akoby naraz, ale iba toľko úloh môže bežať paralelne, koľko má počítač dostupných jadier procesora. Každá úloha zapíše svoj výsledok do jemu priradeného miesta v poli. Po skončení všetkých úloh sa zoberú tie prechodové relácie, ktoré sú splniteľné v danom grafe a transformujú sa na výsledok testu.

Posledný test je na zistenie, či komponent má vôbec nejaké možné prechody. To sa robí s využitím prvého testu. Komponent s virtuálnymi vrcholmi je síce graf, ale nie kubický. Preto musí byť drobná zmena v podmienkach. Ak sa komponent s virtuálnymi vrcholmi dá zafarbiť, potom má nejakú prechodovú reláciu.

Toľko ku implementácii Snark Mastra. Prvotná idea bola vytvoriť distribuovaný systém, ktorý bude na klastri počítačov testovať vlastnosti zafarbiteľnosti väčších snarkov. Naprogramovali sme jednoduché aplikácie, ktoré sa dajú nasadiť na klaster, jedna na hlavný počítač a druhá na ostatné. Tieto aplikácie vytvorili spolu jeden actor systém a vedeli medzi sebou jednoducho komunikovať. Hlavný počítač vie prijímať malé, viac neparalelizovateľné úlohy. Tie posielal ostatným počítačom, ktoré ich vykonávali a výsledok posielali späť hlavnému počítaču. Chýba tu viac vecí, ako napríklad rozdelenie výpočtu hruhou silou na menšie časti, komunikácia s klastrom, monitoring a ďalšie. Neskôr sa však cieľ nášho výskumu obrátil na hľadanie malých snarkov s pokrývacím indexom. Preto sme sa rozhodli pre omnoho jednoduchšiu aplikáciu, ktorá nám bude ponúkať postačujúcu funkcionality.

5.5 Kompilácia a spustenie

Program Snark Master je napísaný v jazyku Scala verzia 2.11, preto potrebuje aj jeho kompilátor. Snark Master však využíva viacero knižníc. Na udržiavanie závislostí sme použili program Gradle. Gradle je nástroj, ktorým sa dá celý projekt konfigurovať. Preto pre kompiláciu a vytvorenie spustiteľného .jar archívu odporúčam použiť Gradle. Stačí spustiť pomocou neho úlohu *buildJar*, ktorá je spolu s ostatnou konfiguráciou definovaná v súbore *build.gradle*. V prílohe je aj skompilovaný projekt. Pre spustenie .jar súboru je potrebné mať nainštalovanú Javu (napríklad JRE SE 8u112).

Po spustení sa otvorí okno s ponukou testov popísaných vyššie.

Záver

Cieľom tejto práce bolo konštruovanie snarkov s pokrývacím indexom aspoň 5. Vieme, že táto vlastnosť súvisí s farbením tetraedrálnej konfigurácie.

Naimplementovali sme program Snark Master, ktorý nám pomáhal skúmať snarky. Pomocou neho vieme testovať, či snark je zafarbiteľný tromi konfiguráciami. Prvá je triviálne trojzafarbenie, ďalšia je tetraedrálnej a nakoniec rozšírená tetraedrálnej konfigurácia. Snark Master tiež vie testovať prechodovú reláciu komponentov.

Prechodovú reláciu komponentu vieme ohraničiť rozborom prípadov a tokovými podmienkami. Tým získame možné prechody cez komponent. Vnútornej štruktúra komponentu však môže vylúčiť viac prechodov. Na to sme využívali Snark Master. S poznaním prechodových relácií komponentov vieme konštruovať snarky s pokrývacím indexom aspoň 5. Existujú už triedy snarkov s pokrývacím indexom aspoň 5. Ukázali sme však, že ich vieme rozšíriť. Napríklad pomocou substitúcií častí grafov väčšími. Tu sme využívali nahradenie dekolineátora dvoma dekolineátormi a komponentom I alebo nahradením cesty dĺžky 1 Halinovým dipólom. Dopracovali sme sa ku bipartitným vložkám, ktoré zachovávajú prechodovú reláciu. Vďaka tomu vieme do každého snarku s pokrývacím indexom aspoň 5, ktorý má na niektorej dvojici hrán spor v prechodovej relácii, vkladať ľubovoľný počet bipartitných vložiek. Týmto konštrukciami vieme vyrábať veľké množstvo malých snarkov s pokrývacím indexom aspoň 5.

Pomocou prechodových relácií sa nám podarilo tiež vysvetliť pre niektoré snarky, prečo majú pokrývací index aspoň 5. Napríklad dekompozíciou Petersonovho grafu na dekolineátor a cestu dĺžky 1 sme našli spor v ich prechodovej relácii.

Motivácia štúdia snarkov bola nájsť protipríklady viacerým hypotézam. Hľadanie snarkov je však ťažké. Zistiť, či sa kubický bezmostový graf dá pokryť štyrmi perfektnými páreniami je NP-úplný problém [5]. Konštruovať snarky s pokrývacím indexom aspoň 5 je ľahšie. Napríklad pomocou spomínaných konštrukcií. Vieme, že do 34 vrcholov existujú 2 netriviálne snarky s pokrývacím indexom aspoň 5. Pomocou spomínaných konštrukcií vieme na každom párnom počte vrcholov od 44 skonštruovať taký snark.

Snark na 44 vrchoch vieme skonštruovať pomocou nahradenia dekolineátora X pomocou XIX v grafe WM_{34} . Pridaním 12, 14, 16 alebo 18-vrcholovej bipartitnej vložky do grafu WM_{34} dostaneme snarky na 46, 48, 50 a 52 vrchoch. Snarky na 54, 56, a 58 dostaneme nahradenia dekolineátora X pomocou XIX v už spomenutých

grafoch. Snarky na 60 vrcholoch a viac už vieme skonštruovať pomocou nezafarbitelného komponentu na 60 vrcholoch. Stále však nevieme, či existujú netriviálne snarky s pokrývacím indexom aspoň 5 na rcholoch medzi 34 a 44.

Ďalšie možnosti pre pokračovanie v tejto práci je vytvorenie generátora, ktorý pomocou popísaných konštrukcií vygeneruje databázu snarkov s pokrývacím indexom aspoň 5 do nejakého rozumného počtu vrcholov. Týchto snarkov bude veľa. Už len pri konštrukcii pomocou nedofarbitelného komponentu na 60 vrcholoch máme obrovské množstvo možných konštrukcií.

Čo sa týka programu Snark Master, je tu tiež priestor na vylepšovanie, či už rýchlosti alebo funkcionality. Napríklad sa môžu pridať nové steinerovské konfigurácie na testovanie, alebo možnosť nahrať pred testom súbor s popísanou konfiguráciou. Ďalšie možnosti sú návrh skúmania prechodových funkcií na väčších konektoroch ako veľkosti 2. Tiež je možnosť pridať výstup do súboru alebo výber formátu výstupu.

Literatúra

- [1] M. Abreu, T. Kaiser, D. Labbate, and G. Mazzuoccolo. Treelike snarks. *The electronic journal of combinatorics*, 23(3), 2016.
- [2] G.M. Adelson-Velskij and A. Titov. On 4-chromatic cubic graphs. *Voprosy Kibernet.(Moscow)*, 1:5–14, 1974.
- [3] G. Brinkmann, J. Goedgebeur, J. Hägglund, and K. Markström. Generation and properties of snarks. *Journal of Combinatorial Theory, Series B*, 103(4):468–488, 2013.
- [4] R. Diestel. *Graph theory {graduate texts in mathematics; 173}*. Springer-Verlag Berlin and Heidelberg GmbH & amp, 2000.
- [5] L. Esperet and G. Mazzuoccolo. On cubic bridgeless graphs whose edge-set cannot be covered by four perfect matchings. *Journal of Graph Theory*, 77(2):144–157, 2014.
- [6] F. Holroyd and M. Škoviera. Colouring of cubic graphs by steiner triple systems. *Journal of Combinatorial Theory, Series B*, 91(1):57–66, 2004.
- [7] R. Isaacs. Infinite families of nontrivial trivalent graphs which are not tait colorable. *The American Mathematical Monthly*, 82(3):221–239, 1975.
- [8] F. Jaeger. A survey of the cycle double cover conjecture. *North-Holland Mathematics Studies*, 115:1–12, 1985.
- [9] R. Lukořka, E. Máčajová, J. Mazák, and M. Škoviera. Small snarks with large oddness. *The electronic journal of combinatorics*, 22(1), 2015.
- [10] E. Máčajová, O. Pangrác, A. Raspaud, J. Sereni, M. Škoviera, et al. Projective, affine, and abelian colorings of cubic graphs. *European Journal of Combinatorics*, 30(1):53–69, 2009.
- [11] G. Mazzuoccolo. The equivalence of two conjectures of berge and fulkerson. *Journal of Graph Theory*, 68(2):125–128, 2011.

- [12] E. Máčajová and M. Škoviera. Snarks that cannot be covered with four perfect matching (v štádiu prípravy).
- [13] J. Petersen. Die theorie der regulären graphs. *Acta Mathematica*, 15(1):193–220, 1891.
- [14] R. Roestenburg, R. Bakker, and R. Williams. *Akka in action*. Manning Publications Co., 2015.
- [15] J. Watkins. Snarks. *Annals of the New York Academy of Sciences*, 576(1):606–622, 1989.

Prílohy

CD obsahujúce

- zdrojový kód aplikácie Snark Master
- spustiteľnú verziu programu Snark Master
- kódy niektorých grafov s pokrývacím indexom aspoň 5 v bratislavskom formáte
- kódy niektorých skúmaných komponentov a ich prechodové relácie
- bipartitné kubické grafy z obvodom aspoň 6 do 20 vrcholov