

Univerza v Ljubljani
Fakulteta za matematiko in fiziko

Predstavitev 4-ciklov v snarkih

Domen Skornšek, Žan Luka Kolarič

Mentorja: Janoš Vidali, Riste Škrekovski

Predmet: Finančna matematika

Ljubljana, 2024

1 Uvodna beseda

Cilj najine naloge je bil ugotoviti, kako uvedba 4- ciklov vpliva na kromatični indeks snarkov. Uporabila sva metodo, ki je na 2 povezavi dodala 2 novi vozlišči in ju povezala med seboj, tako da je nastal 4-cikel. Več o algoritmih (programih) sva zapisala v nadaljevanju poročila.

2 Komentar končnih programov

Napisala sva dva programa, ki delujeta le enkratno in pa program, ki deluje samodejno glede na to, koliko iteracij mu podamo.

Prvi program je namenjen ročnemu vnašanju, na katerih povezavah si želiva nova vozlišča in potem med njih doda povezave, da nastane 4-cikel. Poleg tega tudi preveri, če povezave v grafu sploh obstajajo, da se vozlišča ne dodajajo na kakšnih neobstoječih povezavah. Na koncu nam vrne kromatični indeks novega grafa.

Drugi deluje na podoben način, le da program naključno izbere, na katere povezave bo dodal vozlišča in potem med njih doda povezave, da nastane 4-cikel. Na koncu nam vrne kromatični indeks novega grafa.

Funkcija *nakljucni_robovi_kromaticni_indeks* omogoča iterativno spreminjanje grafa (snarka) z dodajanjem 4-ciklov in spremljanje sprememb kromatičnega indeksa grafa. Poleg tega za vsako iteracijo beleži uporabljene robove, ki so bili vključeni v modifikacijo grafa.

Najprej funkcija ustvari kopijo podanega grafa z uporabo metode `copy()`. Ta korak zagotavlja, da originalni graf ostane nespremenjen, medtem ko se spremembe izvajajo na njegovi kopiji. Nato funkcija pripravi tabelo rezultatov, kjer shrani začetni kromatični indeks grafa.

Znotraj glavne zanke, ki se izvaja n -krat (število iteracij), funkcija naključno izbere dva različna robova iz grafa. Prvi rob se izbere neposredno z uporabo funkcije `random.choice`, medtem ko se drugi rob izbere izmed preostalih robov, pri čemer je poskrbljeno, da ni enak prvemu.

Za vsak izbran rob funkcija ustvari novi vozlišči in poveže obstoječe robove z novima vozliščema ter novi vozlišči med seboj, kar ustvari 4-cikel.

Po vsaki iteraciji funkcija izračuna kromatični indeks trenutnega grafa z metodo `chromatic_index()` in shrani rezultat v tabelo skupaj z informacijami o dveh uporabljenih robovih. Na koncu funkcija vrne tabelo rezultatov, ki vključuje kromatični indeks grafa po vsaki iteraciji ter seznam uporabljenih robov.

3 Snarki, ki sva jih vključila v nalogo

- celmins swart snark 1 (26 vozlišč)
- double star snark (30 vozlišč)
- flower snark j7 (28 vozlišč)
- goldberg snark 3 (24 vozlišč)
- goldberg snark 5 (40 vozlišč)
- loupekines snark 1 (22 vozlišč)
- loupekines snark 2 (22 vozlišč)

4 Ugotovitve

4.1 Ročno - ena iteracija

V tem primeu naju je zanimalo, ali enkratna uvedba cikla vpliva na kromatični indeks. Kot je razvidno iz testnega programa, ostane kromatični indeks ostane 4, ampak le v primeru, ko izbereva dve povezavi, ki nista zaporedni. Pri izbri dveh zaporednih povezav pa se indeks zviša za 1 (torej na 5), kar se intuitivno zdi smiselno, ker bo v ciklu potrebna ena barva več za barvanje grafa.

4.2 Naključno - ena iteracija

V tem primeu naju je zanimalo, ali enkratna uvedba cikla vpliva na kromatični indeks. Kot je razvidno iz testnega programa, ostane kromatični indeks 4, le na določenih grafih, na ostalih pa se poveča na 5. Tukaj sklepava, da se indeks ohrani le v primerih, ko vozlišči dodamo na dve nezaporedni povezavi. To bova lahko preverila z zadnjim programom, ki nama bo povedal tudi, na katere povezave dodamo vozlišči.

4.3 Naključno - i iteracij

V tem primeru sva na vsakem grafu izbedla 20 iteracij in spodaj v tabele zapisala kromatične indekse na vsakem koraku ter tudi na katerih povezavah so se dodajala nova vozlišča. Vodoravno so vse iteracije od 1 do 20, navpično pa so snarki.

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
celmins swart snark 1	4	4	6	7	7	7	7	7	7	7	8	9	10	10	10	10	10
double star snark	4	4	4	4	5	7	7	8	8	8	8	9	9	9	9	9	9
flower snark j7	4	4	4	5	6	6	8	8	8	8	8	8	9	9	9	10	10
goldberg snark 3	4	4	4	4	5	5	6	6	6	6	6	7	7	7	7	7	8
goldberg snark 5	4	5	5	5	5	5	5	5	5	5	6	6	6	6	6	6	6
loupekines snark 1	4	4	5	5	5	6	6	7	7	7	8	8	8	8	8	9	9
loupekines snark 2	4	5	5	5	5	6	6	6	7	7	7	7	7	9	10	11	11
Ptersenov graf	4	4	4	5	6	7	7	7	8	8	8	8	8	8	9	9	9

x	17	18	19	20
celmins swart snark 1	10	10	10	11
double star snark	9	9	9	9
flower snark j7	10	10	10	10
goldberg snark 3	8	8	8	8
goldberg snark 5	7	7	7	7
loupekines snark 1	9	9	9	9
loupekines snark 2	11	11	11	11
Ptersenov graf	9	9	9	10

V naslednji tabeli je prikazano, na kateri povezavi sta bili dodani vozlišči v 1., 2., 3. in 4. koraku. Tako bova ugotovila, ali sta bili vozlišči dodani na sosednji povezavi.

x	1	2
celmins swart snark 1	(11, 19), (16, 22)	(11, 19), ($'u_0', 19$)
double star snark	(6, 15), (21, 28)	(2, 26), (16, 27)
flower snark j7	(21, 23), (19, 25)	(4, 7), (0, 3)
goldberg snark 3	(4, 13), (15, 20)	(16, 23), (3, 22)
goldberg snark 5	(30, 34), (11, 20)	(23, 29), (7, 13)
loupekines snark 1	(11, 18), (5, 15)	(8, 16), (2, 18)
loupekines snark 2	(6, 7), (7, 13)	(5, 9), (9, 16)
Petersenov graf	(2, 7), (3, 4)	($'u_0', v_0'$), (6, 9)

x	3	4
celmins swart snark 1	$(11, 19), (7, 15)$	$(11, 22), (6, 14)$
double star snark	$(4, 8), (10, 12)$	$(21, 28), (9, 13)$
flower snark j7	$(3, 26), ('u1'_0, 21)$	$(21, 23), (13, 16)$
goldberg snark 3	$(6, 7), ('v1'_0, 'v2'_0)$	$(0, 3), ('u1'_1, 16)$
goldberg snark 5	$(6, 23), (5, 32)$	$('v2'_2, 32), (22, 24)$
loupekines snark 1	$(1, 16), (0, 3)$	$(4, 10), ('u1'_2, 'u2'_2)$
loupekines snark 2	$('u1'_1, 'u2'_1), (0, 3)$	$('v1'_2, 'v2'_2), ('u1'_0, 6)$
Petersenov graf	$('u1'_0, 'u2'_0), (5, 8)$	$('u1'_0, 'v1'_0), ('v2'_0, 4)$

UGOTOVITVE 1. *Pri snarkih, ki se jim kromatični indeks poveča iz 4 na 5, sta novi vozlišči nastali na povezavah, ki sta zaporedni. To bi lahko potrdilo začetno domnevo, ampak to bodo lahko potrdile šele iteracije.*

UGOTOVITVE 2. *Ko sva na snarkih uporabila algoritem za iteracije sva ugotovila naslednje stvari:*

1. če se novi vozlišči dodata na 2 zaporedni povezavi, se kromatični indeks poveča za 1
2. če se novi vozlišči dodata na eni izmed povezav, ki je bila že uporabljena v eni izmed prejšnjih iteracij se kromatični indeks poveča za 1
3. če se novi vozlišči dodata na kateri koli povezavi, ki je sosednja povezavi iz prejšnjih iteracij se kromatični indeks poveča za 1. (tj. vozlišče te povezave je že bilo uporabljeno)
4. novonastale povezave se lahko uporabilo, vendar ne povezava $(prvirob[0], u_1)$, $(prvirob[1], u_2)$, $(drugirob[0], v_1)$ in $(drugirob[1], v_2)$
5. pri snarku "goldberg snark 5" ni nekega pravila, kdaj se kromatični indeks poveča
6. kromatični indeks se z dodajanjem novih ciklov nikoli ne more znižati