

UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET



Lazar Stanojević

ALGORITMI ZA PRONALAŽENJE
RAZLIČITIH TIPOVA ARTIKULACIONIH
TAČAKA I MOSTOVA U USMERENIM
GRAFOVIMA

master rad

Beograd, 2026.

Mentor:

prof. dr Vesna MARINKOVIĆ, vanredni profesor
Univerzitet u Beogradu, Matematički fakultet

Članovi komisije:

prof. dr Filip MARIĆ, redovni profesor
Univerzitet u Beogradu, Matematički fakultet

dr Danijela SIMIĆ, docent
Univerzitet u Beogradu, Matematički fakultet

Datum odbrane: _____

Naslov master rada: Algoritmi za pronalaženje različitih tipova artikulacionih tačaka i mostova u usmerenim grafovima

Rezime: U ovom radu razmatraju se algoritmi za pronalaženje različitih tipova artikulacionih tačaka i mostova u usmerenim grafovima. Osnovna motivacija rada je analiza elemenata usmerenog grafa čijim se uklanjanjem narušava neka vrsta povezanosti. U radu se najpre obrađuju jake artikulacione tačke i jaki mostovi, odnosno čvorovi i grane čijim se uklanjanjem povećava broj jakih komponenti povezanosti grafa. Posebno se ističe veza ovih pojmova sa dominatorima u grafovima toka, na osnovu koje se dobijaju efikasni algoritmi za njihovo pronalaženje. Zatim se razmatraju b-artikulacione tačke i b-mostovi, koji se definišu u kontekstu jako bipovezanih grafova i dodatno uzimaju u obzir bipovezanost odgovarajućeg neusmerenog grafa. Na kraju su predstavljeni i algoritmi za pronalaženje $s - t$ artikulacionih tačaka i $s - t$ mostova, gde se posmatraju čvorovi i grane neophodni za postojanje puta između dva izdvojena čvora. Za opisane algoritme prikazana je osnovna ideja, pseudokod i analiza vremenske složenosti. Rad sadrži i deo namenjen eksperimentalnoj evaluaciji, u kome se mogu potvrditi teorijski rezultati vremena izvršavanja algoritama.

Ključne reči: usmereni grafovi, jake artikulacione tačke, jaki mostovi, dominatori, jako bipovezani grafovi, b-artikulacione tačke, b-mostovi, $s - t$ artikulacione tačke, $s - t$ mostovi

Sadržaj

1	Uvod	1
2	Jake artikulacione tačke i jaki mostovi	4
2.1	Osnovni pojmovi	4
2.2	Jake artikulacione tačke, jaki mostovi i dominatori	8
2.3	Algoritam za nalaženje jakih mostova	11
2.4	Algoritam za nalaženje jakih artikulacionih tačaka	13
2.5	Složenost	15
3	b-artikulacione tačke i b-mostovi	16
3.1	Osnovni pojmovi	16
3.2	b-mostovi	19
3.3	b-artikulacione tačke	24
4	$s - t$ artikulacione tačke i mostovi	26
4.1	Osnovni pojmovi	26
4.2	Algoritam za pronalaženje $s - t$ mostova	29
4.3	Algoritam za pronalaženje $s - t$ artikulacionih tačaka	33
4.4	Analiza složenosti	36
5	Implementacija i evaluacija algoritama	37
5.1	Evaluacija algoritma za pronalaženje jakih artikulacionih tačaka i jakih mostova	38
5.2	Evaluacija algoritma za pronalaženje b-artikulacionih tačaka i b-mostova	40
5.3	Evaluacija algoritma za pronalaženje $s - t$ artikulacionih tačaka i $s - t$ mostova	43
6	Zaključak i mogući pravci daljeg rada	46

Glava 1

Uvod

Grafovi predstavljaju jedan od osnovnih matematičkih modela za opisivanje odnosa između objekata. U zavisnosti od prirode problema, grane grafa mogu biti neusmerene ili usmerene. Usmereni grafovi se prirodno javljaju u brojnim primenama, kao što su komunikacione mreže, tokovi podataka, mreže puteva, zavisnosti između zadataka i različiti sistemi u kojima odnos između dva objekta nije nužno simetričan. Zbog toga je analiza povezanosti usmerenih grafova važan problem, kako sa teorijske, tako i sa praktične strane.

Jedno od važnih pitanja u teoriji grafova jeste koliko je dati graf otporan na uklanjanje pojedinačnih čvorova ili grana. U neusmerenim grafovima dobro su poznati pojmovi artikulacionih tačaka i mostova. Artikulaciona tačka je čvor čijim se uklanjanjem povećava broj komponenti povezanosti grafa. Slično, most je grana čijim se uklanjanjem povećava broj komponenti povezanosti grafa. U usmerenim grafovima situacija je složenija, jer je potrebno voditi računa o smeru grana i o tome da li se posmatra obična dostižnost, jaka povezanost ili neka dodatna osobina grafa. Iz tog razloga postoji više različitih uopštenja ovih pojmova.

Tema ovog rada su algoritmi za pronalaženje različitih tipova artikulacionih tačaka i mostova u usmerenim grafovima. Posmatraju se sledeće tri grupe problema:

1. Jake artikulacione tačke i jaki mostovi.

Ovi pojmovi definisani su preko narušavanja jakih komponenti povezanosti [9]. Kod njih se posmatra kako se struktura jako povezanog grafa menja nakon uklanjanja jednog čvora ili jedne grane

Ovi algoritmi nalaze svoju primenu u komunikacionim i transportnim mrežama, gde artikulacione tačke mogu predstavljati kritične stanice, rutere ili

čvorišta, a mostovi kritične usmerene veze. Njihovo otkrivanje pomaže u pronalaženju slabih tačaka sistema i u projektovanju mreža koje ostaju funkcionalne i nakon otkaza pojedinačnih elemenata.

2. B-artikulacione tačke i b-mostovi.

B-artikulacione tačke i b-mostovi se javljaju u kontekstu jako bipovezanih grafova [11]. Ovi pojmovi dodatno uzimaju u obzir bipovezanost odgovarajućeg neusmerenog grafa

Slično kao jake artikulacione tačke i jaki mostovi, i ovi algoritmi nalaze svoju primenu u komunikacionim i transportnim mrežama.

3. $s - t$ artikulacione tačke i $s - t$ mostovi.

Ovu grupu čine $s - t$ artikulacione tačke i $s - t$ mostovi, gde se za dva izdvojena čvora s i t ispituje koji čvorovi i grane su neophodni za postojanje puta od s do t [4].

Za razliku od prethodne dve grupe problema, ovde cilj nije očuvanje povezanosti celog grafa, već očuvanje mogućnosti da se iz čvora s stigne do čvora t . Ovi algoritmi su korisni pri analizi pouzdanosti sistema, planiranju alternativnih ruta i otkrivanju uskih grla u mrežama, jer omogućavaju identifikovanje elemenata od kojih direktno zavisi dostižnost između zadatog izvora i odredišta.

U radu se najpre uvode potrebni osnovni pojmovi i svojstva za svaku od navedenih grupa problema. Posebna pažnja posvećena je vezi između jakih artikulacionih tačaka, jakih mostova i dominatora u grafovima toka. Ta veza omogućava da se problem pronalaženja kritičnih čvorova i grana u jako povezanim grafovima svede na problem računanja dominatora [12].

Nakon toga se razmatraju b-artikulacione tačke i b-mostovi, pri čemu se koriste prethodno uvedeni algoritmi za pronalaženje jakih artikulacionih tačaka i jakih mostova, kao i osobine jako bipovezanih komponenti. Na kraju se obrađuju $s - t$ artikulacione tačke i $s - t$ mostovi, za koje se prikazuju algoritmi zasnovani na transformaciji grafa i pretrazi koja se zaustavlja upravo u traženim čvorovima ili granama.

Za svaki od prikazanih algoritama razmatraju se njegova osnovna ideja, pseudokod i vremenska složenost. Pored teorijskog opisa, u radu je sprovedena i eksperimentalna evaluacija algoritama. U tom delu se za svaku grupu algoritama opisuju načini generisanja test-grafova i prikazuju rezultati merenja vremena izvršavanja

za različite veličine ulaza. Na taj način se teorijska analiza dopunjuje praktičnim prikazom ponašanja algoritama nad nasumično generisanim grafovima.

Cilj rada je da se na jednom mestu predstavie i uporede srodni algoritamski problemi koji se odnose na kritične čvorove i grane u usmerenim grafovima. Iako se svi posmatrani pojmovi zasnivaju na istoj osnovnoj ideji, odnosno na tome da uklanjanje nekog elementa narušava određenu vrstu povezanosti, algoritmi za njihovo pronalaženje koriste različite tehnike. Zbog toga rad daje pregled više pristupa, od dominatorskih stabala i transponovanja grafa, preko bipovezanih komponenti, do specifičnih transformacija grafa za $s - t$ probleme.

Glava 2

Jake artikulacione tačke i jaki mostovi

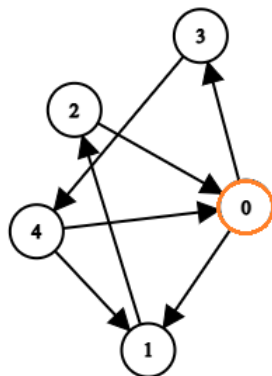
U ovom poglavlju upoznaćemo se sa pojmovima jakih artikulacionih tačaka i mostova, nakon čega ćemo predstaviti efikasan algoritam za njihovo pronalaženje. Koristićemo standardne termine teorije grafova, bez njihovog posebnog uvođenja. Broj čvorova grafa označavaćemo sa n , a broj grana grafa sa m , ukoliko nije eksplicitno naznačeno drugačije.

2.1 Osnovni pojmovi

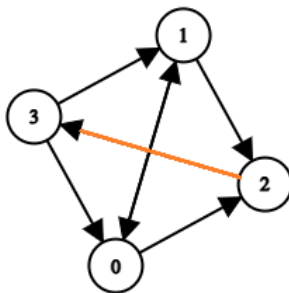
Neka je $G = (V, E)$ usmeren graf, sa n čvorova i m grana. *Put* u grafu G je niz čvorova $v_1, v_2, \dots, v_k$ takvih da grana $(v_i, v_{i+1}) \in E, \forall i \in \{1, 2, \dots, k-1\}$. Usmereni graf G je *jako povezan*, ukoliko postoji put između svaka dva čvora u grafu. *Jake komponente povezanosti* grafa G su njegovi maksimalni jako povezani podgrafovi [2].

Definicija 1. *Čvor grafa G je jaka artikulaciona tačka, ukoliko nakon njegovog uklanjanja, kao i svih njemu susednih grana, dolazi do povećanja broja jakih komponenti povezanosti grafa G [9].*

Definicija 2. *Grana grafa G je jak most, ukoliko nakon njenog uklanjanja dolazi do povećanja broja jakih komponenti povezanosti grafa G [9].*



Slika 2.1: Jako povezan graf u kome je čvor 0 jaka artikulaciona tačka.



Slika 2.2: Jako povezan graf u kome je grana (2, 3) jak most.

Primer 1. Na slici 2.1 prikazan je usmeren graf koji je jako povezan jer se iz svakog čvora može doći do svakog drugog. Kada uklonimo čvor 0 broj komponenti jake povezanosti se povećava na četiri, pa je čvor 0 jaka artikulaciona tačka u grafu.

Primer 2. Na slici 2.2 prikazan je usmeren graf koji je jako povezan jer se iz svakog čvora može doći do svakog drugog. Kada uklonimo granu (2, 3) broj komponenti jake povezanosti se povećava na tri, pa je grana (2, 3) jak most u grafu.

U nastavku teksta pokazaćemo neka osnovna svojstva jakih artikulacionih tačaka i jakih mostova, uz pretpostavku da radimo sa jako povezanim grafom. Ovo možemo da pretpostavimo bez umanjenja opštosti, zato što bi u slučaju da graf ima više jakih komponenti povezanosti, skupovi jakih mostova i artikulacionih tačaka bili unija odgovarajućih skupova, po svim komponentama povezanosti.

Radi jednostavnosti zapisa, uvodimo dodatnu notaciju, uz pomoć skupovne razlike. Neka je $G = (V, E)$ usmeren graf, i neka su $S \subseteq E$, i $X \subseteq V$, podskupovi skupa

grana i skupa čvorova, respektivno. Označavamo sa $G \setminus S$, odnosno $G \setminus X$, graf koji se dobija uklanjanjem svih grana skupa S , odnosno graf koji se dobija uklanjanjem svih čvorova skupa X , kao i svih grana susednih tim čvorovima.

Lema 1. *Neka je $G = (V, E)$ jako povezan graf, i neka je $v \in V$ čvor grafa G . Čvor v je jaka artikulaciona tačka u G ako i samo ako postoje čvorovi x i y iz G , takvi da $x \neq v$ i $y \neq v$, i pritom svi putevi od x do y vode preko v [9].*

Dokaz. Pretpostavimo najpre da je v jaka artikulaciona tačka. To znači da njenim uklanjanjem iz grafa, G prestaje da bude jako povezan, odnosno postoje čvorovi x i y , različiti od v , između kojih ne postoji put. Kako je pre uklanjanja v , postojao put između njih, jer je G bio jako povezan, sledi da je svaki put od x do y vodio preko v . Kako bismo dokazali suprotnu implikaciju, pretpostavimo da svaki put od x do y , vodi preko v . Ovo znači da uklanjanjem čvora v , ne postoji nijedan put od x do y , odnosno G prestaje da bude jako povezan, pa sledi da je v jaka artikulaciona tačka. $\square$

Lema 2. *Neka je $G = (V, E)$ jako povezan graf, i neka je $(u, v) \in E$ grana grafa G . Grana (u, v) je jak most u G ako i samo ako postoje čvorovi x i y iz G , takvi da svi putevi od x do y sadrže granu (u, v) [9].*

Dokaz. Analogno prethodnoj lemi, važi da ukoliko je (u, v) jak most, njenim uklanjanjem postoje dva čvora x i y , tako da više ne postoji put od x do y , što znači da je svaki put od x do y sadržao granu (u, v) . U suprotnom, ukoliko svaki put od x do y vodi preko (u, v) , znači da se njenim uklanjanjem povećava broj jakih komponenti povezanosti, odnosno (u, v) je jak most. $\square$

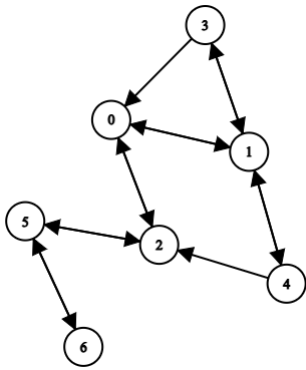
Definicija 3. *Neka je $G = (V, E)$ jako povezan graf. Za granu (u, v) kažemo da je redundantna, ukoliko postoji alternativni put od u do v , koji je ne sadrži. U suprotnom za granu (u, v) kažemo da nije redundantna [9].*

Posledica 1. *Grana (u, v) je jak most ako i samo ako nije redundantna.*

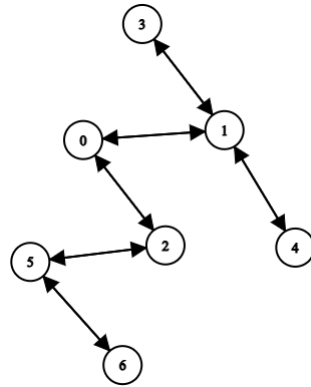
Primer 3. *Posmatrajmo graf na slici 2.1 i moguće usmerene puteve od čvora 4 do čvora 3. Jedan mogući put sastoji se od grana $(4, 0)$ i $(0, 3)$, a drugi od grana $(4, 1)$, $(1, 2)$, $(2, 0)$ i $(0, 3)$. Na osnovu toga vidimo da su neke od redundantnih grana u ovom grafu $(4, 0)$, $(4, 1)$, $(1, 2)$ i $(2, 0)$ jer postoje alternativni putevi koji ih ne sadrže. Nasuprot tome, jedina ulazna grana u čvor 3 je grana $(0, 3)$ iz čega sledi da svi putevi od čvora 4 do čvora 3 moraju da je sadrže, pa ona nije redundantna.*

Usmeren graf sa n čvorova može da ima najviše n jakih artikulacionih tačaka. Ukoliko se, na primer, graf sastoji od jednog usmerenog ciklusa, onda je svaka tačka jaka artikulaciona tačka, i dostigli smo ovu granicu.

Da bismo odredili gornju granicu za broj jakih mostova, potreban nam je malo drugačiji pristup. Neka je $G = (V, E)$ jako povezan graf. Fiksirajmo proizvoljan čvor v . Označimo sa $T^+(v)$ usmereno razapinjuće stablo grafa G , sa korenom u v , i svim granama usmerenim od čvora v . Analogno tome, definišemo i stablo $T^-(v)$, u kome su sve grane usmerene ka v .



Slika 2.3: Graf G .



Slika 2.4: Graf G' .

Lema 3. Neka je v proizvoljan čvor grafa G . Graf $G'(v) = T^+(v) \cup T^-(v)$ sadrži najviše $2n - 2$ grane i sadrži sve jake mostove grafa G [9].

Dokaz. Radi jednostavnosti zapisa graf $G'(v)$ označićemo samo sa G' . Graf G' se sastoji od dva razapinjuća stabla, pri čemu se svako od njih sastoji od $n - 1$ grane. Ovim je određena gornja granica za broj grana grafa G' , budući da je skup grana grafa G' unija skupova grana ova dva stabla. Potrebno je još pokazati da G' sadrži i sve jake mostove grafa G . Za svaka dva para čvorova x i y iz V postoji usmeren put od x do y u G' jer $T^-(v)$ sadrži put od x do v , a $T^+(v)$ sadrži put od v do y . Dakle, G' je jako povezan podgraf grafa G , i sve grane koje nisu u G' su redundantne, i ne mogu biti jaki mostovi. $\square$

Posledica 2. Graf G može imati najviše $2n - 2$ jakih mostova.

Za konstrukciju grafa G' možemo koristiti bilo algoritam pretrage u dubinu bilo algoritam pretrage u širinu, od kojih su oba linearne vremenske složenosti iz čega sledi da je vreme potrebno za izračunavanje grafa $G'(v)$ jednako $\mathcal{O}(m + n)$.

Prethodna lema nam daje jednostavan algoritam za nalaženje svih jakih mostova u grafu G , složenosti $\mathcal{O}(n \cdot (m + n))$. Prvo izračunamo graf G' , a pošto on sadrži sve jake mostove, za svaku granu grafa G' testiramo da li je ona jak most, odnosno da li se njenim uklanjanjem povećava broj jakih komponenti povezanosti. Vremenska složenost ovog pristupa sledi iz toga što je maksimalan broj grana grafa G' reda $\mathcal{O}(n)$, a jake komponente povezanosti možemo pronaći u vremenskoj složenosti $\mathcal{O}(m + n)$ koristeći Tarjanov algoritam [13].

Primer 4. Na slici 2.4 prikazan je graf $G'(v) = T^+(v) \cup T^-(v)$ koji odgovara grafu sa slike 2.3. Za čvor v uzet je čvor 1.

2.2 Jake artikulacione tačke, jaki mostovi i dominatori

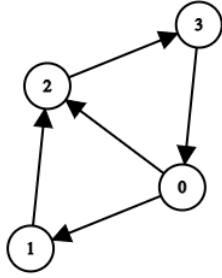
Efikasan algoritam za pronalaženje jakih artikulacionih tačaka i mostova se zasniva na njihovoj vezi sa dominatorima u grafovima toka (eng. *flowgraphs*). Graf toka $G(s) = (V, E, s)$ je usmeren graf sa početnim čvorom $s \in V$, takvim da je svaki čvor iz V dostižan iz s . Na sledeći način možemo definisati relaciju dominacije između čvorova: u dominira čvorom v , ako svaki put od s do v vodi preko čvora u . Za u kažemo i da je *dominator* čvora v . Sa $dom(v)$ označavamo skup svih dominatora čvora v . Jasno je da je $dom(s) = \{s\}$, kao i da $\forall v \neq s \{s, v\} \subseteq dom(v)$. Za s i v kažemo da su trivijalni dominatori čvora v . Relacija dominacije je tranzitivna i njenu tranzitivnu redukciju¹ zovemo *dominatorsko stablo* (eng. *dominator tree*) i označavamo sa $DT(s)$. Koren ovog stabla je čvor s . Iz dominatorskog stabla možemo da zaključimo da čvor u dominira čvorom v ako i samo ako je on njegov predak u $DT(s)$. Kažemo i da je u *neposredni dominator* (eng. *immediate dominator*) čvora v ako u dominira nad v , i svaki drugi netrivialni dominator čvora v dominira čvorom u . Ukoliko v ima bar jednog dominatora, onda ima i jedinstvenog neposrednog dominatora, koji je njegov roditelj u dominatorskom stablu [12].

Primer 5. Na slici 2.5 prikazan je graf toka $G(0)$.

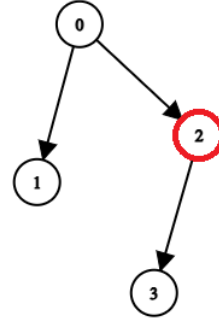
Primer 6. Na slici 2.6 prikazano je dominatorsko stablo grafa toka sa slike 2.5. Čvor 2 dominira čvorom 3 jer svi putevi do njega vode preko čvora 2, a kako je po

¹Tranzitivna redukcija je operacija nad grafovima, koja za cilj ima da se za dati usmereni graf konstruiše usmereni graf sa istim skupom čvorova i što manjim skupom grana, a da se pritom ne promeni relacija dostižnosti

definiciji netrivijskih dominatora različit od 0 i 3, on je i netrivijski dominator. Ukoliko za neka dva čvora u i v postoji usmeren put između čvora u i čvora v u dominatorskom stablu, to znači da je čvor u dominator čvora v .



Slika 2.5: Graf toka $G(0)$.



Slika 2.6: Dominatorsko stablo grafa toka, sa posebno označenim netrivijskim dominatorima.

Ako je $G = (V, E)$ jako povezan graf, onda za svaki čvor $s \in V$ važi da je $G(s)$ graf toka, pošto su u G svi čvorovi međusobno dostižni. Iz ovoga zaključujemo da jako povezan graf G sa n čvorova ima n različitih grafova toka. Sledeće leme nam daju bitne veze između jakih artikulacionih tačaka i dominatora, kao i jakih mostova i dominatora, koje će biti potrebne za izvođenje algoritma.

Lema 4. *Neka je $G = (V, E)$ jako povezan graf, i s proizvoljni čvor iz V . Neka je $G(s) = (V, E, s)$ graf toka sa početnim čvorom s . Ako je čvor u netrivijski dominator čvora v u grafu toka $G(s)$, onda za čvor u kažemo da je on jaka artikulaciona tačka [9].*

Dokaz. Pošto je u netrivijski dominator čvora v , znači $u \neq v$ i $u \neq s$, i svaki put od s do v mora da sadrži u . Sledi da graf $G \setminus \{u\}$ nije jako povezan, pa u mora biti jaka artikulaciona tačka grafa G . $\square$

Lema 5. *Neka je $G = (V, E)$ jako povezan graf. Ako je čvor u jaka artikulaciona tačka u G , onda mora postojati čvor s , takav da je u netrivijski dominator nekog čvora v u grafu toka $G(s) = (V, E, s)$ [9].*

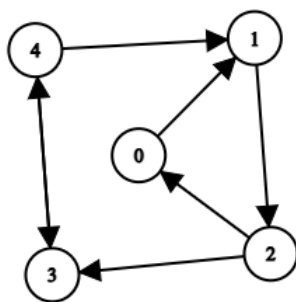
Dokaz. Na osnovu prethodne leme, ukoliko je u jaka artikulaciona tačka u G , onda moraju postojati dva čvora, v i s , $v \neq u$ i $s \neq u$, takvi da svi putevi od s do v

vode preko u . Odavde sledi da u mora biti netrivialni dominator od v u grafu toka $G(s)$. $\square$

Na osnovu prethodne dve leme sledi da je za nalaženje jakih artikulacionih tačaka dovoljno odrediti sve netrivialne dominatore u grafu toka $G(s)$, $\forall s \in V$. Međutim, kao što je ranije napomenuto broj grafova toka je jednak broju čvorova grafa G , što znači da bismo za svaki od njih morali da tražimo dominatore. Kako bismo dobili efikasan algoritam, potrebno je da detaljnije razmotrimo vezu između dominatora i jakih artikulacionih tačaka.

Pre toga, razmotrićemo slična svojstva za jake mostove. Kažemo da je (u, v) grana dominator (eng. *edge dominator*) čvora w , ukoliko svaki put od s do w sadrži granu (u, v) . Za (u, v) kažemo da je neposredna grana dominator (eng. *immediate edge dominator*) čvora w , ako dominira nad w , i svaka druga grana koja dominira nad w , dominira i nad čvorom u . U slučaju kada čvor ima bar jednu granu dominatora važi da on ima jedinstvenu neposrednu granu dominatora. U nastavku navodimo dve leme koje su analogne lemapa za čvorove dominatore.

Primer 7. Na slici 2.7 prikazan je graf toka $G(0)$, pri čemu je grana $(2, 3)$ neposredna grana dominator čvora 3. Grana $(2, 3)$ je neposredna grana dominator zato što čvorom 3 dominiraju još i grane $(0, 1)$ i $(1, 2)$, ali one takođe dominiraju i čvorom 2.



Slika 2.7: Graf toka $G(0)$. Grana $(2, 3)$ je neposredna grana dominator čvora 3.

Lema 6. Neka je $G = (V, E)$ jako povezan graf, i s proizvoljni čvor iz V . Neka je $G(s) = (V, E, s)$ graf toka sa početnim čvorom s . Ako je (u, v) grana dominator u $G(s)$, onda je (u, v) jak most u G [9].

Lema 7. *Neka je $G = (V, E)$ jako povezan graf. Ako je (u, v) jak most u G , onda mora postojati čvor s , takav da je (u, v) grana dominator u grafu toka $G(s) = (V, E, s)$ [9].*

Kao i malopre, ove dve leme nam omogućavaju pronalazak jakih mostova u grafu, međutim, nisu dovoljne za konstrukciju efikasnog algoritma za nalaženje jakih mostova.

2.3 Algoritam za nalaženje jakih mostova

Kako bi dobili efikasan algoritam za nalaženje svih mostova u grafu potrebno je da uvedemo pojam *transponovanog grafa* (eng. *reversed graph*). Neka je $G = (V, E)$ usmeren graf, njegov transponovani graf u oznaci $G^R = (V, E^R)$ je graf sa istim skupom čvorova i granama suprotnog usmerenja, odnosno $\forall (u, v) \in E, (v, u) \in E^R$.

Lema 8. *Neka je $G = (V, E)$ jako povezan graf, i G^R njegov transponovan graf. Onda je i G^R jako povezan. Grana (u, v) je jak most u G ako i samo ako je grana (v, u) jak most u G^R [9].*

Neka je G jako povezan graf, $s \in V$ proizvoljan čvor, i neka je $G(s)$ graf toka sa početnim čvorom s . Označimo sa $DE(s)$ skup grana dominatora u $G(s)$. Analogno, neka je G^R transponovani graf grafa G , $G^R(s)$ graf toka transponovanog grafa sa početkom u čvoru s , i $DE^R(s)$ skup grana dominatora u $G^R(s)$.

Teorema 1. *Neka je $G = (V, E)$ jako povezan graf, i $s \in V$ proizvoljan čvor grafa. Grana (u, v) je jak most u G ako i samo ako $(u, v) \in DE(s)$ ili $(v, u) \in DE^R(s)$ [9].*

Dokaz. Dokažimo najpre da ako je (u, v) jak most u G , onda moramo imati (u, v) u $DE(s)$ ili (v, u) u $DE^R(s)$. Kako je grana (u, v) jak most, znači da se njenim uklanjanjem povećava broj jakih komponenti povezanosti. Odavde sledi da postoji čvor $w \neq s$, koji je nakon uklanjanja grane (u, v) u različitoj komponenti povezanosti od s . Kako je G jako povezan graf u njemu postoje putevi i od čvora s do čvora w , i od čvora w do čvora s , pa je prethodno tvrđenje moguće ako nakon uklanjanja grane prestaje da postoji put od čvora s do čvora w ili prestaje da postoji put od čvora w do čvora s . U prvom slučaju, svi putevi od s do w moraju sadržati granu (u, v) , što je ekvivalentno tome da je ova grana dominator čvora w u $G(s)$, odnosno da pripada $DE(s)$. Ukoliko je pak ispunjen drugi uslov, svi putevi od w do s sadrže granu (u, v) , odnosno (v, u) je grana dominator čvora w u $G^R(s)$, što znači da pripada $DE^R(s)$.

Dakle, ukoliko je grana (u, v) jak most u G , onda ili ona pripada $DE(s)$ ili grana (v, u) pripada $DE^R(s)$.

Suprotan smer je direktna posledica prethodnih lema. Ukoliko $(u, v) \in DE(s)$, onda je ona grana dominator u $G(s)$, te na osnovu Leme 6 sledi da mora biti jak most u G . Slično, ukoliko $(v, u) \in DE^R(s)$, onda je (v, u) jak most u G^R , pa sledi da je (u, v) jak most u G , na osnovu Leme 8. $\square$

Na osnovu prethodnih tvrđenja dolazi se do algoritma 1 za određivanje jakih mostova u jako povezanom grafu, čiji su autori Giuseppe F. Italiano, Luigi Laura i Federico Santaroni [9]. Skupove grana dominatora izračunaćemo pomoću algoritma koji su predložili autori Lengauer i Tarjan, složenosti $\mathcal{O}(m \cdot \alpha(m, n))$ [12]².

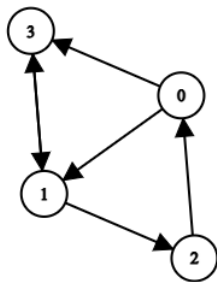
Algoritam 1 Jaki mostovi

Ulaz: Jako povezan graf $G = (V, E)$, sa n čvorova i m grana

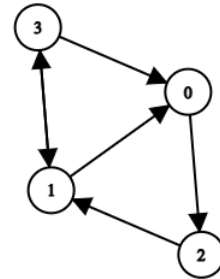
Izlaz: Jaki mostovi grafa G

- 1: Izaberi proizvoljni čvor $s \in V$
 - 2: Izračunaj skup grana dominatora $DE(s)$ u grafu toka $G(s) = (V, E, s)$
 - 3: Izračunaj transponovani graf $G^R = (V, E^R)$
 - 4: Izračunaj skup grana dominatora $DE^R(s)$ u grafu toka $G^R(s) = (V, E^R, s)$
 - 5: **return** unija grana iz $DE(s)$ i grana suprotnog usmerenja iz $DE^R(s)$
-

Sva prethodna razmatranja važe u slučaju kada je graf jako povezan. Ukoliko bi bilo potrebno da nađemo jake mostove za običan usmereni graf, bilo bi potrebno da pokrenemo algoritam za svaku jako povezanu komponentu grafa, i vratimo uniju svih pronađenih jakih mostova.



Slika 2.8: Jako povezan graf G



Slika 2.9: Transponovani graf G^R grafa G

² α je inverzna Akermanova funkcija, koja raste toliko sporo da je za sve praktične ulaze praktično konstanta ≤ 4

Primer 8. Na slici 2.8 prikazan je jako povezan graf G na kome ćemo prikazati izvršavanje algoritma za pronalaženje jakih mostova u grafu. Prvi korak je da izaberemo proizvoljan čvor i neka je to recimo $s = 0$. Potom računamo grane dominatore $DE(0)$ u grafu toka $G(0)$. Posmatramo koje su nam grane neophodne da bi se iz čvora 0 stiglo do ostalih čvorova. Zaključujemo da se do čvora 2 može stići samo preko grane $(1, 2)$, dok do ostalih čvorova postoje alternativni putevi. Odavde sledi da je $DE(0) = \{(1, 2)\}$. Okrećemo usmerenje grana kako bismo dobili graf G^R (slika 2.9), i potom ponavljamo postupak računanja dominatora. Slično kao i malopre zaključujemo da je $DE^R(0) = \{(0, 2), (2, 1), (1, 3)\}$ zbog toga što u grafu $G^R(0)$ svi putevi od čvora 0 do čvorova 2, 1 i 3 vode preko ovih grana. Na kraju računamo uniju grana iz $DE(0)$ i grana suprotnog usmerenja iz $DE^R(0)$ i zaključujemo da je skup jakih mostova u grafu G jednak $\{(0, 2), (1, 2), (3, 1)\}$.

2.4 Algoritam za nalaženje jakih artikulacionih tačaka

Razmotrimo sada kako na sličan način možemo da dođemo do algoritma za određivanje jakih artikulacionih tačaka u grafu.

Lema 9. Neka je $G = (V, E)$ jako povezan graf, i G^R njegov transponovan graf. Čvor u je jaka artikulaciona tačka u G ako i samo ako je u jaka artikulaciona tačka u G^R [9].

Teorema 2. Neka je $G = (V, E)$ jako povezan graf i $s \in V$ proizvoljan čvor. Čvor u je jaka artikulaciona tačka u G ako i samo ako $u \in D(s) \cup D^R(s)$ [9].

Dokaz. Slično dokazu za jake mostove, prvo ćemo dokazati smer koji tvrdi da ako je u jaka artikulaciona tačka u G , i $u \neq s$, onda mora da važi da $u \in D(s) \cup D^R(s)$. Kako je čvor u jaka artikulaciona tačka, njegovim uklanjanjem se povećava broj jakih komponenti povezanosti. Stoga postoji čvor $w \neq s$, koji je nakon uklanjanja u različitoj komponenti povezanosti od s . To je moguće ako postoji put od s do w u G , a ne postoji put nakon uklanjanja čvora, ili postoji put od w do s u G , a ne postoji nakon uklanjanja čvora. U prvom slučaju, svi putevi od s do w moraju sadržati čvor u , što je ekvivalentno tome da je ovaj čvor dominator čvora w u $G(s)$, odnosno da pripada $D(s)$. Ukoliko važi drugi slučaj, svi putevi od w do s sadrže čvor u , odnosno u je dominator čvora w u $G^R(s)$, što znači da pripada $D^R(s)$. Dakle, ukoliko je čvor

u jaka artikulaciona tačka u G , onda on pripada skupu $D(s) \cup D^R(s)$.

Ponovo se na osnovu prethodnih lema lako izvodi i suprotan smer tvrđenja. Ukoliko $u \in D(s)$, onda je on dominator u $G(s)$, pa mora biti jaka artikulaciona tačka u G . Slično, ukoliko $u \in D^R(s)$, onda je u jaka artikulaciona tačka u G^R , pa sledi da je u i jaka artikulaciona tačka u G . $\square$

Na osnovu prethodne teoreme sledi algoritam 2, čiji su autori takođe Giuseppe F. Italiano, Luigi Laura i Federico Santaroni [9]. Kao i ranije, za pronalaženje skupova dominatora koristimo algoritam autora Lengauera i Tarjana [12].

Algoritam 2 Jake artikulacione tačke

Ulaz: Jako povezan graf $G = (V, E)$, sa n čvorova i m grana

Izlaz: Jake artikulacione tačke grafa G

- 1: $S = \emptyset$
 - 2: Izaberi proizvoljni čvor $s \in V$
 - 3: **if** s jaka artikulaciona tačka **then**
 - 4: dodaj s u S
 - 5: **end if**
 - 6: Izračunaj skup netrivialnih dominatora $D(s)$ u grafu toka $G(s) = (V, E, s)$
 - 7: Izračunaj transponovani graf $G^R = (V, E^R)$
 - 8: Izračunaj skup netrivialnih dominatora $D^R(s)$ u grafu toka $G^R(s) = (V, E^R, s)$
 - 9: **return** $D(s) \cup D^R(s) \cup S$
-

Primer 9. Na slici 2.8 prikazan je jako povezan graf G na kome ćemo prikazati izvršavanje algoritma za pronalaženje jakih mostova u grafu. Prvi korak je da izaberemo proizvoljan čvor i neka je to recimo $s = 0$. Proveravamo da li je čvor 0 jaka artikulaciona tačka i zaključujemo da jeste jer nakon njegovog uklanjanja čvor 2 više nema izlaznih grana, pa ne postoji usmereni put od čvora 2 do ostalih čvorova. Dodajemo čvor 0 u S . Računamo skup netrivialnih dominatora $D(0)$ u grafu toka $G(0)$. Do čvorova 1 i 3 možemo stići na dva načina, ali do čvora 2 svi putevi vode kroz čvor 1, odakle sledi da je $D(0) = \{1\}$. Okrećemo usmerenje grana kako bismo dobili graf G^R (slika 2.9), i potom ponavljamo postupak računanja dominatora. U grafu toka $G^R(0)$ čvor 1 je neophodan da bismo stigli do čvora 3, dok je čvor 2 neophodan kako bismo stigli do svih ostalih čvorova. Odavde sledi da je $D^R(0) = \{1, 2\}$. Na kraju vraćamo uniju ova tri skupa i zaključujemo da je skup jakih artikulacionih tačaka jednak $\{0, 1, 2\}$.

2.5 Složenost

Razmatrajući date pseudokodove algoritama, zaključuje se da složenost algoritma za traženje jakih mostova i algoritma za traženje jakih artikulacionih tačaka zavisi od složenosti faze traženja dominatora. To je zbog toga što se transponovani graf može konstruisati u vremenu $\mathcal{O}(m)$, pri čemu je m broj grana u grafu, a pronalaženje traženih unija je reda veličine broja elemenata, pri čemu smo granice i za artikulacione tačke i mostove već diskutovali. Najefikasniji poznati algoritam za pronalaženje dominatora je linearne vremenske složenosti u odnosu na veličinu grafa, odnosno složenosti $\mathcal{O}(m + n)$ [1] [3]. Mi ćemo jednostavnosti radi, za potrebe implementacije algoritama za pronalaženje jakih artikulacionih tačaka i jakih mostova, koristiti algoritam koji su predložili autori Lengauer i Tarjan, složenosti $\mathcal{O}(m \cdot \alpha(m, n))$ [12]. Samim tim i algoritmi za određivanje jakih mostova i jakih artikulacionih tačaka biće složenosti $\mathcal{O}(m \cdot \alpha(m, n))$.

Glava 3

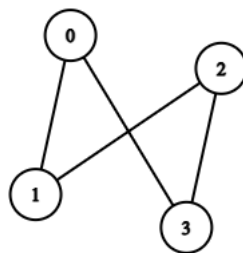
b-artikulacione tačke i b-mostovi

Prethodno obrađeni algoritmi i termini poslužiće nam kao dobra osnova za uvođenje nove klase artikulacionih tačaka i mostova. U ovom poglavlju, uvešćemo pojmove jako bipovezanih grafova (eng. *strongly biconnected graph*), zatim b-artikulacionih tačaka i b-mostova koji se u njima javljaju, kao i algoritme za njihovo pronalaženje.

3.1 Osnovni pojmovi

Definicija 4. *Neusmeren graf G je bipovezan ukoliko je povezan i ne sadrži nijednu artikulacionu tačku, odnosno uklanjanjem bilo kog čvora iz G on ostaje povezan [10].*

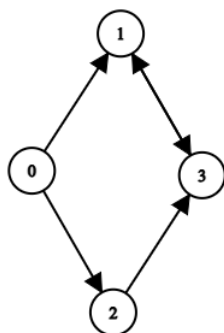
Na slici 3.1 prikazan je jedan bipovezan graf.



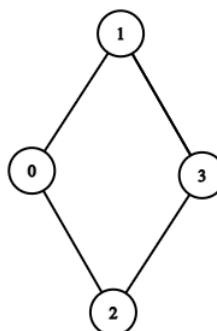
Slika 3.1: Bipovezan neusmereni graf.

Definicija 5. Neka je $G = (V, E)$ usmeren graf. Za neusmeren graf $G' = (V, E')$ sa istim skupom čvorova i skupom grana dobijenim kada se granama iz G ukine usmerenje kažemo da je odgovarajući neusmeren graf usmerenog grafa G .

Na slici 3.3 prikazan je odgovarajući neusmeren graf usmerenog grafa G ilustriranog na slici 3.2.



Slika 3.2: Usmeren graf G .



Slika 3.3: Neusmeren graf G' koji odgovara grafu sa slike 3.2.

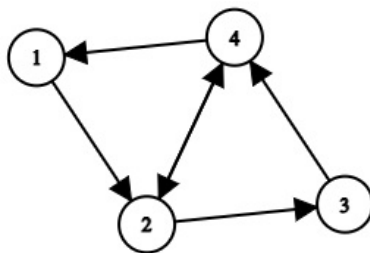
Prethodne definicije, kao i termin jake povezanosti grafa nam pomažu pri uvođenju termina *jako bipovezanih grafova*, nad kojima se definišu pojmovi *b-artikulacionih tačaka* i *b-mostova*.

Definicija 6. Neka je $G = (V, E)$ usmeren graf. Za G kažemo da je jako bipovezan ukoliko je on jako povezan i odgovarajući neusmeren graf grafa G je bipovezan [10].

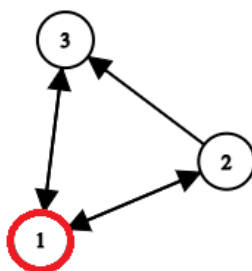
Primer jako bipovezanog grafa prikazan je na slici 3.4.

Sada kada smo uveli potrebne termine, možemo definisati pojmove b-artikulacionih tačaka i b-mostova.

Definicija 7. Neka je $G = (V, E)$ jako bipovezan graf. Čvor v grafa G je b-artikulaciona tačka ukoliko njegovim uklanjanjem graf G prestaje da bude jako bipovezan [11].



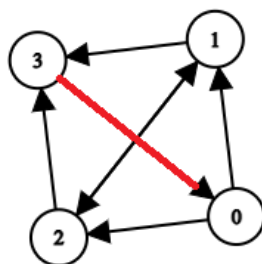
Slika 3.4: Jako bipovezan graf.



Slika 3.5: Jako bipovezan graf. Čvor 1 je b-artikulaciona tačka.

Definicija 8. Neka je $G = (V, E)$ jako bipovezan graf. Grana e grafa G je b-most ukoliko njenim uklanjanjem G prestaje da bude jako bipovezan [11].

Primer 10. Posmatrajmo graf na slici 3.5. Nakon uklanjanja čvora 1 i svih njemu susednih grana graf prestaje da bude jako povezan jer više ne postoji usmereni put od čvora 3 do čvora 2 iz čega sledi da je čvor 1 b-artikulaciona tačka.



Slika 3.6: Jako bipovezan graf. Grana $(0, 3)$ je b-most.

Primer 11. Posmatrajmo graf na slici 3.6. Nakon uklanjanja grane $(3, 0)$ ne postoji više nijedna izlazna grana iz čvora 3. Zaključujemo da više ne postoji usmereni put od čvora 3 do bilo kog drugog čvora u grafu pa graf nije jako povezan, sledi da je grana $(3, 0)$ b-most.

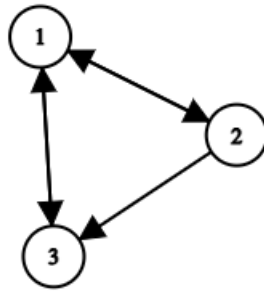
3.2 b-mostovi

U ovom poglavlju ćemo razmotriti na koji način je moguće pronaći sve b-mostove u datom bipovezanom grafu. Za potrebe toga, prezentovaćemo potrebne leme.

Lema 10. *Neka je $G = (V, E)$ jako bipovezan graf i neka je grana e jak most u G . Onda je e i b-most u G [11].*

Dokaz. Kako je grana e jak most, a graf G je jako bipovezan pa samim tim i jako povezan, sledi da uklanjanjem grane e iz grafa G on prestaje da bude jako povezan, što znači da nije ni jako bipovezan, iz čega sledi da je e b-most. $\square$

Primetimo da suprotno ne mora da važi: ukoliko je grana e b-most u G , ne mora da znači da je e i jak most u G .



Slika 3.7: Jako bipovezan graf. Grana $(2, 3)$ je b-most ali nije jak most.

Primer 12. *Posmatrajmo jako bipovezan graf na slici 3.7. Nakon uklanjanja grane $(2, 3)$ i dalje postoji usmereni put između svaka dva čvora, pa ona nije jak most. Sa druge strane, nakon uklanjanja ove grane čvor 1 postaje artikulaciona tačka u odgovarajućem neusmerenom grafu, jer njegovim uklanjanjem čvorovi 2 i 3 prestaju da budu povezani. Ovim zaključujemo da odgovarajući neusmeren graf prestaje da bude bipovezan, pa grana $(2, 3)$ jeste b-most a nije istovremeno i jak most.*

Definicija 9. Jako bipovezana komponenta *jako povezanog grafa* $G = (V, E)$ je maksimalan podskup čvorova $U \subseteq V$ takav da je graf indukovani skupom čvorova U jako bipovezan.

Neka je G jako bipovezan graf. Naredna lema nam daje mehanizam kojim je moguće smanjiti broj jako bipovezanih komponenti u jako povezanom podgrafu od G .

Lema 11. *Neka je $G_s = (V, E_s)$ podgraf jako bipovezanog grafa $G = (V, E)$ takav da je G_s jako povezan sa n jako bipovezanih komponenti. Neka je (u, w) grana iz $E \setminus E_s$ takva da u i w pripadaju različitim jakim bipovezanim komponentama C_1^B i C_2^B grafa G_s i $u, w \notin C_1^B \cap C_2^B$. Tada graf $(V, E \cup \{(u, w)\})$ sadrži najviše $n - 1$ jako bipovezanih komponenti [11].*

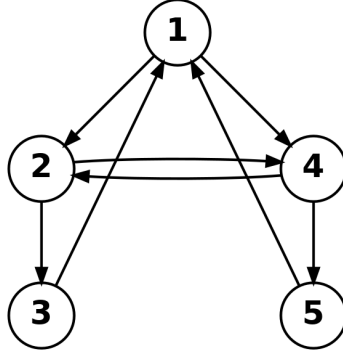
Dokaz. Budući da je graf G_s jako povezan, sledi da postoji put između čvorova w i u u G_s . Grana (u, w) formira ciklus, iz čega sledi da su u i w u istoj jako bipovezanoj komponenti u grafu $(V, E \cup \{(u, w)\})$. $\square$

Lema 12. *Za proizvoljnu granu $e \in E \setminus E_y$ važi da nije b-most [11].*

Dokaz. Poznato je da jako povezan podgraf jako povezanog grafa može da se nađe pronalaženjem razapinjućeg stabla u G i u transponovanom grafu G_r [6]. Skup grana E_y se inicijalizuje na uniju grana dva razapinjuća stabla, a potom se proširuje granom (u, w) koja spaja dve različite komponente jake bipovezanosti, sve dok graf $G_y = (V, E_y)$ ne postane jako bipovezan. Kako se broj komponenti jake bipovezanosti nakon dodavanja grane (u, w) , na osnovu prethodne leme, smanjuje bar za jedan, znači da će graf G_y postati jako bipovezan. Iz ovoga sledi da grane iz $E \setminus E_y$ nisu b-mostovi, budući da njihovo uklanjanje ne narušava jaku bipovezanost. $\square$

Algoritam za pronalaženje b-mostova

Na osnovu prethodnih lema, možemo definisati algoritam 3 za pronalaženje b-mostova, koji je 2020. godine predstavio autor Raed Jaber [11]. Osnovna ideja algoritma je da prvo izračunamo skup jakih mostova u grafu, a potom za proizvoljan čvor grafa izračunamo dva razapinjuća stabla sa korenom u tom čvoru, prvo u početnom grafu a drugo u transponovanom grafu. Nakon toga kreiramo novi graf sa istim skupom čvorova, dok za skup grana uzimamo uniju grana iz oba razapinjuća stabla. U novi graf dodajemo grane koje smanjuju broj jako bipovezanih komponenti sve dok on ne postane jako bipovezan. Više reči o tome kako možemo naći jako bipovezane komponente grafa biće u poglavlju analiza složenosti algoritma. Za svaku granu u novom grafu koja nije jak most u početnom grafu proveravamo da li se njenim uklanjanjem narušava bipovezanost neusmerenog grafa koji odgovara početnom grafu. Kao rezultat vraćamo uniju jakih mostova i grana koje narušavaju bipovezanost neusmerenog grafa.



Slika 3.8: Jako bipovezan graf.

Primer 13. Posmatrajmo usmereni graf $G = (V, E)$ prikazan na slici 3.8. Odgovarajući neusmereni graf sadrži grane $(1, 2), (2, 3), (3, 1), (1, 4), (4, 5), (5, 1), (2, 4)$. Ovaj graf je bipovezan, dok je početni usmereni graf jako povezan. Prema tome, graf G je jako bipovezan. Koristeći algoritam 1 nalazimo jake mostove u grafu, sledi da je skup $U = \{(2, 3), (3, 1), (4, 5), (5, 1)\}$.

Izaberimo čvor $y = 1$. Najpre se postavlja $E_y = \emptyset$. Jedno razapinjuće stablo sa korenom u čvoru 1 u grafu G može sadržati grane $T = \{(1, 2), (2, 3), (1, 4), (4, 5)\}$. U transponovanom grafu G_r može se izabrati razapinjuće stablo čije grane, nakon preslikavanja nazad na originalni graf, odgovaraju skupu $\{(3, 1), (2, 3), (5, 1), (4, 5)\}$. Prema tome, $E_y = \{(1, 2), (2, 3), (3, 1), (1, 4), (4, 5), (5, 1)\}$. Podgraf $G_y = (V, E_y)$ jeste jako povezan ali nije jako bipovezan. U odgovarajućem neusmerenom grafu čvor 1 predstavlja artikulacionu tačku.

Jako bipovezane komponente trenutnog podgrafa mogu se zapisati kao $C_1^B = \{1, 2, 3\}$ i $C_2^B = \{1, 4, 5\}$. Njihov presek je $C_1^B \cap C_2^B = \{1\}$. Zbog toga se u prvom koraku while petlje skupu E_y dodaje grana $(2, 4)$, nakon čega se broj jako bipovezanih komponenti smanjuje i graf postaje jako bipovezan.

Nakon završetka petlje postavlja se $B = U$ pa važi $B = \{(2, 3), (3, 1), (4, 5), (5, 1)\}$. Potrebno je proveriti da li uklanjanjem neke grane skupa $E_y \setminus B$ odgovarajući neusmeren graf grafa G prestaje da bude bipovezan. Ovaj skup je jednak $\{(1, 2), (1, 4), (2, 4)\}$. Kako odgovarajući neusmeren graf grafa G ostaje bipovezan nakon uklanjanja bilo koje od ove tri grane, sledi da je skup b-mostova u grafu G isti kao skup jakih mostova u grafu G , pa je $B = \{(2, 3), (3, 1), (4, 5), (5, 1)\}$ konačan rezultat.

Algoritam 3 b-mostovi

Ulaz: Jako bipovezan graf $G = (V, E)$, sa n čvorova i m grana

Izlaz: B , pri čemu je B skup b-mostova u G

```

1:  $U = \emptyset$ 
2: Izračunaj skup jakih mostova u  $G$  i smesti u  $U$ 
3: Izaberi proizvoljan čvor  $y \in V$ 
4:  $E_y = \emptyset$ 
5: Izračunaj razapinjuće stablo  $T$  sa korenom u  $y$  u  $G$ 
6: Dodaj sve grane iz  $T$  u  $E_y$ 
7: Izračunaj transponovani graf  $G_r$ 
8: Izračunaj razapinjuće stablo  $T_y$  sa korenom u  $y$  u  $G_r$ 
9: Dodaj sve grane iz  $T_y$  u  $E_y$ 
10: while  $G_y = (V, E_y)$  nije jako bipovezan do
11:   Pronađi jako bipovezane komponente u  $G_y$ 
12:   Pronađi granu  $(u, w)$  takvu da  $u$  i  $w$  pripadaju različitim jakim bipovezanim
       komponentama  $C_1^B, C_2^B$  grafa  $G_s$  i  $u, w \notin C_1^B \cap C_2^B$ 
13:   Dodaj granu  $(u, w)$  u  $E_y$ 
14: end while
15:  $B = U$ 
16: for  $(u, w) \in E_y \setminus B$  do
17:   Izračunaj odgovarajući neusmeren graf  $G'$  grafa  $G$ 
18:   if  $G' \setminus \{(u, w)\}$  nije bipovezan then
19:      $B = B \cup (u, w)$ 
20:   end if
21: end for
22: return  $B$ 

```

Analiza složenosti algoritma

Preostaje nam da vidimo kako možemo da realizujemo sve navedene korake iz algoritma, kao i koja je složenost ovog algoritma.

Jake mostove u grafu možemo naći u vremenu $\mathcal{O}(m + n)$, ali kao što je ranije diskutovano, mi smo koristili algoritam vremenske složenosti $\mathcal{O}(m \cdot \alpha(m, n))$, tako da se koraci 1 – 4 algoritma izvršavaju u vremenu $\mathcal{O}(m \cdot \alpha(m, n))$. Razapinjuće stablo jako bipovezanog grafa možemo konstruisati u linearnom vremenu, koristeći DFS algoritam [5]. Transponovani graf se takođe može odrediti u linearnoj vremenskoj složenosti, iz čega sledi da je vreme izvršavanja koraka 5 – 9 jednako $\mathcal{O}(m + n)$.

Prema Lemi 11 broj koraka **while** petlje je najviše $n - 1$. Ostaje da vidimo kako možemo da odredimo jako bipovezane komponente grafa.

Teorema 3. *Neka je $G = (V, E)$ jako povezan graf. Jako bipovezane komponente u*

G su oni podgrafovi od G koji su indukovani skupom čvorova $V(B)$, pri čemu su B bipovezane komponente odgovarajućeg neusmerenog grafa od G [14].

Ovom teoremom smo problem pronalaženja jako bipovezanih komponenti u jako povezanom grafu sveli na problem pronalaženja bipovezanih komponenti u odgovarajućem neusmerenom grafu. Preostaje nam da vidimo kako možemo naći bipovezane komponente u neusmerenom grafu. Bipovezane komponente u neusmerenom grafu možemo odrediti korišćenjem Tarjanovog DFS algoritma, čija je vremenska složenost linearna po veličini grafa [9], pa je samim tim i ukupna vremenska složenost određivanja jako bipovezanih komponenti u jako povezanom grafu takođe linearna. Za pronalaženje grane čiji čvorovi pripadaju različitim jako bipovezanim komponentama potrebno je da prođemo skupom grana, i za oba njena čvora proverimo kojim komponentama pripadaju. Kako broj grana može biti reda $\mathcal{O}(m)$, a u najgorem slučaju čvorovi mogu da pripadaju $n - 1$ jako bipovezanoj komponenti, vremenska složenost koraka 12 je $\mathcal{O}(m \cdot n)$. Ukupna vremenska složenost koraka 11 – 13 je stoga jednaka $\mathcal{O}(m \cdot n)$, a kako je broj iteracija petlje reda $\mathcal{O}(n)$, ukupno vreme izvršavanja petlje je $\mathcal{O}(m \cdot n^2)$.

U `for` petlji, imamo proveru da li je graf dobijen uklanjanjem grane jako bipovezan, a kako prolazimo skupom grana u kojem se ne nalaze jaki mostovi, potrebno je samo da proverimo da li uklanjanje grane narušava bipovezanost u odgovarajućem neusmerenom grafu, što se svodi na proveru da li dobijeni graf ima artikulacionih tačaka. Ovo možemo utvrditi Tarjanovim algoritmom, u linearnoj vremenskoj složenosti. Oдавde sledi da je ukupno vreme izvršavanja `for` petlje reda $\mathcal{O}(m \cdot (n + m))$. Iz ovoga jasno vidimo da `while` petlja odnosi najviše vremena, prema tome ukupna vremenska složenost algoritma je $\mathcal{O}(m \cdot n^2)$.

Napomenimo da autor algoritma u svom radu navodi da je ukupna složenost algoritma $\mathcal{O}(m \cdot n)$ [11], pri čemu nije eksplicitno objašnjeno kako se ta vremenska složenost postiže. Navedeno je da je vremenska složenost tela `while` petlje $\mathcal{O}(m)$, što postavlja izazov pronalaženja efikasnijeg načina za određivanje grane čiji čvorovi pripadaju različitim jako povezanim komponentama, obzirom na to da ovaj korak narušava vremensku složenost izvršavanja našeg algoritma.

3.3 b-artikulacione tačke

Pronalaženje b-artikulacionih tačaka je jednostavniji problem nego pronalaženje b-mostova. Navedimo lemu koja će nam biti potrebna za definisanje algoritma.

Lema 13. *Neka je $G = (V, E)$ jako bipovezan graf, i neka je čvor v jaka artikulaciona tačka u G . Onda je v i b-artikulaciona tačka u G .*

Dokaz. Kako je čvor v jaka artikulaciona tačka u G , njegovim uklanjanjem graf G prestaje da bude jako povezan, pa samim tim i jako bipovezan, iz čega sledi da je čvor v i b-artikulaciona tačka u G . $\square$

Dakle, nakon što pronađemo skup jakih artikulacionih tačaka, koje narušavaju jaku povezanost grafa pa samim tim i jaku bipovezanost, preostaje da za ostale čvorove proverimo da li se njihovim uklanjanjem narušava drugo svojstvo jake bipovezanosti, odnosno bipovezanost odgovarajućeg neusmerenog grafa. Iz ovoga direktno proizilazi sledeći algoritam.

Algoritam za pronalaženje b-artikulacionih tačaka

Ovaj algoritam je direktna posledica Leme 13, kao i definicije b-artikulacione tačke. Osnovna ideja algoritma je da prvo izračunamo njegove jake artikulacione tačke, a potom da za ostale čvorove proverimo da li se njihovim uklanjanjem narušava bipovezanost neusmerenog grafa koji odgovara usmerenom grafu nad kojim vršimo algoritam. Konačni rezultat je unija jakih artikulacionih tačaka i čvorova koji su uklanjanjem narušili bipovezanost odgovarajućeg neusmerenog grafa. Algoritam 4 prikazuje odgovarajući pseudokod. U daljem tekstu sa $G \setminus u$ označavamo graf koji se dobija uklanjanjem čvora u kao i svih njegovih incidentnih grana iz grafa G .

Primer 14. *Posmatrajmo ponovo jako bipovezan usmereni graf $G = (V, E)$ prikazan na slici 3.8. Kao što je već pokazano, odgovarajući neusmereni graf G' sadrži grane $(1, 2), (2, 3), (3, 1), (1, 4), (4, 5), (5, 1), (2, 4)$ i bipovezan je, dok je sam usmereni graf G jako povezan. Prema tome, graf G je jako bipovezan.*

Najpre, koristeći algoritam 2 nalazimo skup jakih artikulacionih tačaka grafa G . Prema tome na početku je $B = \{1, 2, 4\}$. Zatim se u for petlji razmatraju preostali čvorovi $V \setminus B = \{3, 5\}$. Za svaki od njih se konstruiše odgovarajući neusmeren graf G' i ispituje se da li $G' \setminus u$ ostaje bipovezan.

Algoritam 4 b-artikulacione tačke

Ulaz: Jako bipovezan graf $G = (V, E)$, sa n čvorova i m grana

Izlaz: B , pri čemu je B skup b-artikulacionih tačaka u G

```

1: Izračunaj skup jakih artikulacionih tačaka u  $G$  i smesti u  $B$ 
2: for  $u \in V \setminus B$  do
3:   Izračunaj odgovarajući neusmeren graf  $G'$  grafa  $G$ 
4:   if  $G' \setminus u$  nije bipovezan then
5:      $B = B \cup u$ 
6:   end if
7: end for
8: return  $B$ 

```

Za $u = 3$ dobija se neusmeren graf sa granama $(1, 2), (1, 4), (4, 5), (5, 1), (2, 4)$ nad čvorovima $\{1, 2, 4, 5\}$. Ovaj graf nema artikulacionih tačaka, pa je bipovezan, stoga čvor 3 nije b-artikulaciona tačka.

Za $u = 5$ dobija se neusmeren graf sa granama $(1, 2), (2, 3), (3, 1), (1, 4), (2, 4)$ nad čvorovima $\{1, 2, 3, 4\}$. I ovaj graf je bipovezan, pa ni čvor 5 nije b-artikulaciona tačka.

Kako uklanjanjem nijednog čvora iz $V \setminus B$ odgovarajući neusmeren graf ne prestaje da bude bipovezan, skup B se ne menja. Konačan rezultat je skup b-artikulacionih tačaka $B = \{1, 2, 4\}$, koji se u ovom primeru poklapa sa skupom jakih artikulacionih tačaka grafa G .

Analiza složenosti algoritma

Kao što smo više puta diskutovali, jake artikulacione tačke možemo najbrže naći u linearnom vremenu, ali kako smo u implementaciji koristili ranije opisani algoritam, vremenska složenost koraka broj 1 biće $\mathcal{O}(m \cdot \alpha(m, n))$. Broj koraka **for** petlje je $n - j$, pri čemu je j broj jakih artikulacionih tačaka u G . Odgovarajući neusmeren graf G' grafa G se može naći u linearnom vremenu. Ostaje samo da vidimo kako možemo da proverimo da li je neusmeren graf bipovezan nakon uklanjanja čvora u . Ovo se svodi na proveru da li graf ima artikulacionih tačaka, a kako smo ranije diskutovali, to možemo proveriti u linearnoj složenosti, koristeći Tarjanov algoritam [13]. Zaključujemo da najviše vremena u algoritmu oduzima **for** petlja, čija je vremenska složenost $\mathcal{O}((n - j) \cdot (m + n))$, gde je j broj jakih artikulacionih tačaka. U najgorem slučaju vremenska složenost je jednaka $\mathcal{O}(n \cdot (m + n))$.

Glava 4

$s - t$ artikulacione tačke i mostovi

Poslednji u nizu tipova artikulacionih tačaka i mostova koje obrađujemo u ovom radu je $s - t$ tip [4]. Nakon jakih, a zatim i b-artikulacionih tačaka i mostova, u ovom poglavlju videćemo kako možemo da definišemo artikulacione tačke i mostove na malo drugačiji način od uobičajenog.

4.1 Osnovni pojmovi

Naredni tip artikulacionih tačaka i mostova parametrizovaćemo pomoću čvorova koji prestaju da budu povezani nakon njihovog uklanjanja. Za potrebe definisanja potrebnih termina, formulišaćemo termine dominatora na malo drugačiji način, u kontekstu artikulacionih tačaka i mostova.

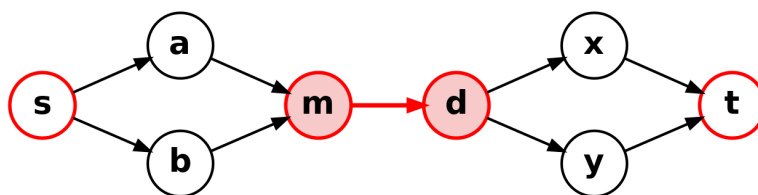
Definicija 10. *Neka je $G = (V, E)$ usmereni graf i neka je s neki njegov čvor. Za čvor $v \in V$ kažemo da je s artikulaciona tačka, ukoliko postoji čvor t koji prestaje da bude dostižan iz s nakon njegovog uklanjanja [4].*

Definicija 11. *Neka je $G = (V, E)$ usmereni graf i neka je s neki njegov čvor. Za grenu $e \in E$ kažemo da je s most, ukoliko postoji čvor t koji prestaje da bude dostižan iz s nakon njenog uklanjanja [4].*

Primetimo da ove dve definicije zapravo opisuju čvorove i grane dominatore. Sada kada smo upoznati sa ovakvim načinom parametrizacije artikulacionih tačaka i mostova, možemo da definišemo $s - t$ artikulacione tačke i mostove. U nastavku ćemo pretpostavljati da je čvor t dostižan iz čvora s u inicijalnom grafu.

Definicija 12. Neka je $G = (V, E)$ usmereni graf i neka su s i t neki njegovi čvorovi. Za čvor $v \in V$ kažemo da je $s - t$ artikulaciona tačka, ukoliko njegovim uklanjanjem ne postoji usmereni put od s do t [4].

Definicija 13. Neka je $G = (V, E)$ usmereni graf i neka su s i t neki njegovi čvorovi. Za granu $e \in E$ kažemo da je $s - t$ most, ukoliko njenim uklanjanjem ne postoji usmereni put od s do t [4].



Slika 4.1: Usmeren graf. Čvorovi m i d su $s - t$ artikulacione tačke. Grana (m, d) je $s - t$ most.

Primer 15. Na slici 4.4 prikazan je usmeren graf sa izdvojenim čvorovima s i t . Čvorovi m i d su $s - t$ artikulacione tačke jer njihovim uklanjanjem prestaje da postoji usmereni put od čvora s do čvora t . Grana (m, d) je $s - t$ most jer njenim uklanjanjem takođe prestaje da postoji usmereni put od čvora s do čvora t .

Radi jednostavnosti zapisa, uvodimo sledeće pojmove.

Definicija 14. Neka je $G = (V, E)$ usmereni graf i neka je $e = (u, v) \in E$ jedna njegova grana. Za čvor v grane e kažemo da je glava grane e , u oznaci $glava(e)$, a za čvor u da je njen rep, u oznaci $rep(e)$.

Razmotrimo sada neka svojstva $s - t$ mostova. Neka je $B = \{b_1, b_2, \dots, b_{|B|}\}$ skup $s - t$ mostova u G za fiksirane čvorove s i t . Po definiciji, $\forall b_i \in B$ ne postoji put od s do t u $G \setminus b_i$, i svi $s - t$ mostovi u B se pojavljuju na svakom putu od s do t u G . Takođe, $s - t$ mostovi iz B su posećeni u istom redosledu na svakoj $s - t$ putanji, što sledi iz dostižnosti u $G \setminus b_i, b_i \in B$. Dakle B možemo posmatrati kao niz $s - t$ mostova uređenih prema njihovom redosledu posećivanja na bilo kojoj $s - t$ putanji. Ovakav niz mostova određuje delove grafa koji su dostižni iz s u $G \setminus b_i$, pri čemu su delovi sve veći kako i raste.

Definicija 15. Neka je $G = (V, E)$ usmereni graf, i neka su s i t dva njegova čvora i B skup $s - t$ mostova. Delovi grafa koji su dostižni iz s u $G \setminus b_i$ postaju sve veći kako

raste i , pri čemu $b_i \in B$. Na osnovu ovoga delimo graf na komponente indukovane mostovima i označavamo ih sa $C = \{C_1, C_2, \dots, C_{|B|+1}\}$, pri čemu C_i označava deo grafa koji je dostižan iz s u $G \setminus b_i$, ali nije bio dostižan u $G \setminus b_{i-1}$ [4].

Na slici 4.2 su prikazane komponente indukovane mostovima. Kako su ove komponente razdvojene $s - t$ mostovima, svaka $s - t$ putanja ulazi u C_i u jedinstvenom čvoru. Pošto most b_{i-1} razdvaja komponente C_{i-1} i C_i , ulazni čvor u komponentu C_i jednak je $glava(b_{i-1})$, odnosno s za C_1 . Slično, $s - t$ putanja napušta C_i u jedinstvenom čvoru, koji je jednak $rep(b_i)$, odnosno t za C_{i+1} .

Analogno ovome, i skup artikulacionih tačaka $A = \{a_1, a_2, \dots, a_{|A|}\}$ prati fiksirani redosled na svakoj $s - t$ putanji.

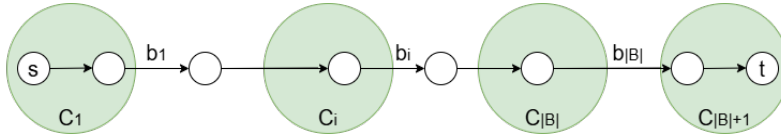
Definicija 16. Neka je $G = (V, E)$ usmereni graf, i neka su s i t dva njegova čvora i A skup $s - t$ artikulacionih tačaka. Delovi grafa koji su dostižni iz s u $G \setminus a_i$ postaju sve veći kako raste i , pri čemu $a_i \in A$. Na osnovu ovoga delimo graf na komponente indukovane artikulacionim tačkama i označavamo ih sa $C = \{C_1, C_2, \dots, C_{|A|+1}\}$, pri čemu C_i označava deo grafa koji je dostižan iz s u $G \setminus a_i$, ali nije bio dostižan u $G \setminus a_{i-1}$ [4].

Na slici 4.3 su prikazane komponente indukovane artikulacionim tačkama. Izlaz iz komponente C_i predstavlja artikulaciona tačka a_i , ukoliko ona postoji, odnosno čvor t u suprotnom. Slično, ulaz u komponentu C_i predstavlja artikulaciona tačka a_{i-1} , ukoliko ona postoji, odnosno čvor s u suprotnom.

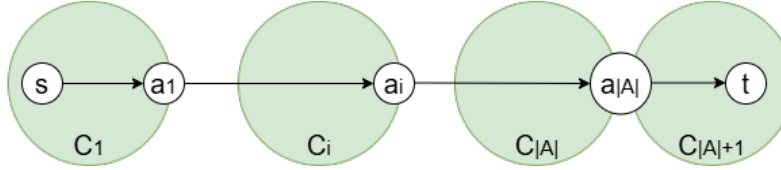
Primer 16. Radi lakšeg razumevanja komponenata indukovanih $s - t$ artikulacionim tačkama i $s - t$ mostovima, na slici 4.4 prikazan je konkretan graf G koji će služiti kao primer za pojašnjenje kako izgledaju pomenute komponente.

Sa slike vidimo da svaka putanja od čvora s do čvora t u grafu G mora da prođe kroz čvorove p , x , y i q . Iz definicije $s - t$ artikulacione tačke sledi da su ova četiri čvora $s - t$ artikulacione tačke. Pošto imamo četiri $s - t$ artikulacione tačke, one određuju pet komponenti na sledeći način: $C_1 = \{s, p\}$, $C_2 = \{p, a, b, x\}$, $C_3 = \{x, c, d, y\}$, $C_4 = \{y, e, f, q\}$ i $C_5 = \{q, t\}$. Primetimo da svaka $s - t$ artikulaciona tačka pripada dvema komponentama, i predstavlja granicu između njih.

Slično, grane koje se nalaze na svakoj putanji od čvora s do čvora t su (s, p) i (q, t) , iz čega sledi da postoje dva $s - t$ mosta u grafu G . Ova dva $s - t$ mosta indukuju tri komponente, i to na sledeći način: $C_1 = \{s\}$, $C_2 = \{p, a, b, x, c, d, y, e, f, q\}$

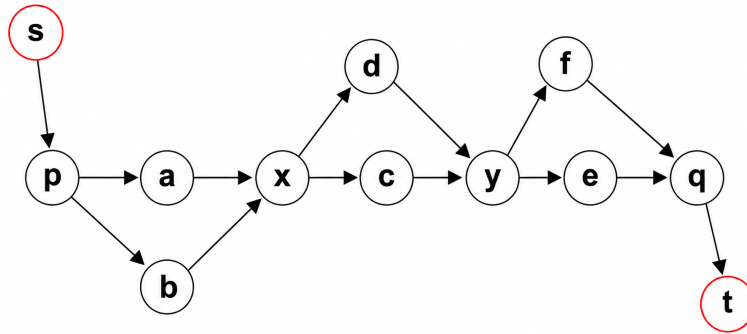


Slika 4.2: Komponente indukovane mostovima $b_1, b_2, \dots, b_{|B|}$.



Slika 4.3: Komponente indukovane artikulacionim tačkama $a_1, a_2, \dots, a_{|A|}$.

i $C_3 = \{t\}$. Primetimo da ovde komponente nemaju zajedničke čvorove, zbog toga što različiti krajevi $s - t$ mosta koji razdvaja komponente pripadaju različitim komponentama.



Slika 4.4: Graf G koji sadrži $s - t$ artikulacione tačke i $s - t$ mostove za fiksirane čvorove s i t

4.2 Algoritam za pronalaženje $s - t$ mostova

Skup $s - t$ mostova u grafu možemo pronaći na sledeći način: pokrenućemo pretragu od čvora s do čvora t izbegavajući proizvoljnu $s - t$ putanju. Pretraga biva prekinuta u $s - t$ mostovima. Kako pretraga napreduje otkrivamo niz mostova B , u

redosledu u kome se oni nalaze na izabranoj $s - t$ putanji. Kako bi olakšali pronalaženje $s - t$ mostova graf ćemo transformisati.

Razmotrimo prethodni algoritam: on prvo pronalazi proizvoljnu $s - t$ putanju P , a potom izvršava pretragu od s do t izbegavajući grane sa putanje P . Cilj nam je da transformišemo graf tako da pretraga biva prekinuta samo u $s - t$ mostovima. Ovo možemo postići jer se svi mostovi nalaze na putanji P . Pretpostavimo da je pretraga stala pre nego što je došla do čvora t : kako bi došla do čvora t ona dalje mora da nastavi nekom granom sa putanje P . Ova grana je prvi $s - t$ most jer smo njome morali da prođemo kako bismo iz čvora s stigli do čvora t i označavamo je sa b_1 . Pretragu onda možemo nastaviti iz čvora $glava(b_1)$, dok se naredni put ne zaustavimo ponovo i pronađemo sledeći most, ukoliko on postoji, i sve tako dok ne stignemo to t .

Transformacija grafa. Prva pretpostavka mogla bi biti da je dovoljno ukloniti grane koje pripadaju proizvoljnoj $s - t$ putanji, odnosno grane za koje je prethodno navedeno da će biti izbegavane tokom pretrage. Međutim, samo izbegavanje grana sa takve putanje ne garantuje da će se pretraga zaustaviti isključivo na $s - t$ mostu. Naime, kad pretraga dođe do čvora v iz P , to znači da nijedna grana koja prethodi v u P nije $s - t$ most. Neka čvor u prethodi čvoru v u P , on može biti iskorišćen kako bi se od njega stiglo do nekog čvora w , koji je dalji od v u P . Iz ovoga sledi da moramo obezbediti da su svi čvorovi koji prethode čvoru v posećeni, kako bismo mogli da tvrdimo da je pretraga prekinuta baš u $s - t$ mostu. Zato ćemo, umesto da uklonimo grane sa putanje P , obrnuti njihovo usmerenje.

Označimo sa P^{-1} putanju dobijenu od putanje P , okretanjem usmerenja grana.

Razmotrimo sada detaljnije sam algoritam. Nakon što izaberemo proizvoljnu $s - t$ putanju P , transformišimo graf na ranije opisan način, obrtanjem usmerenja grana sa putanje P . Osim što ćemo kako pretraga napreduje računati niz $s - t$ mostova B , računaćemo i niz $comp$, kako bismo pratili kojoj komponenti pripada koji čvor, pri čemu je $comp[v] = i$ ukoliko čvor v pripada komponenti C_i . Indikator da čvor v nije posećen biće kada je $comp[v] = 0$. Pri pretrazi koristimo red Q , koji na početku sadrži čvor s , odnosno ulaz u komponentu C_1 . Pretraga uklanja i posećuje čvorove iz Q , dodavajući sve njihove neposećene susede u Q , dok red ne bude prazan. Svaki čvor koji je posećen u ovoj iteraciji je dodeljen komponenti C_i , pri čemu je i redni broj iteracije. Ukoliko t nije posećen u ovom procesu, pronalazimo čvor y kao poslednji posećeni čvor na putanji P , odnosno izlaz iz komponente C_i . Izlaznu

granu iz čvora y , recimo (y, z) , pri čemu z pripada putanji P , identifikujemo kao $s - t$ most i dodajemo u B . Potom pretragu nastavljamo iz čvora z dodajući ga u Q . Ukoliko je čvor t ipak bio posećen nakon jedne ovakve iteracije prekidamo algoritam.

Algoritam 5 prikazuje pseudokod ovog algoritma, čiji su autori Massimo Cairo, Shahbaz Khan, Romeo Rizzi, Sebastian Schmidt, Alexandru I. Tomescu i Elia C. Zironde [4].

Algoritam 5 $s - t$ mostovi

Ulaz: Graf $G = (V, E)$, $s, t \in V$

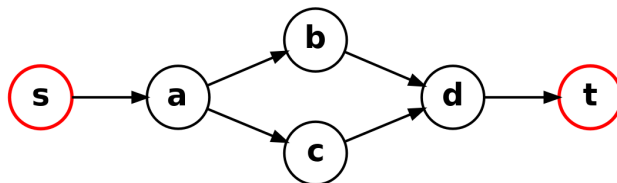
Izlaz: B , pri čemu je B niz $s - t$ mostova

```

1: Izračunaj proizvoljnu putanju  $P$  od  $s$  do  $t$  u  $G$ 
2: Transformiši graf  $G = (G \setminus P) \cup P^{-1}$ 
3: for all  $v \in V$  do
4:    $comp[v] = 0$ 
5: end for
6:  $i = 1$ 
7: while  $comp[t] = 0$  do
8:   if  $i = 1$  then
9:     Dodaj  $s$  u  $Q$ 
10:     $comp[s] = 1$ 
11:   else
12:     Pronađi poslednji čvor  $y$  na  $P$  tako da je  $comp[y] \neq 0$ 
13:     Čvor  $z$  postaje sledbenik čvora  $y$  na  $P$ 
14:     Dodaj granu  $(y, z)$  u  $B$ 
15:     Dodaj  $z$  u  $Q$ 
16:      $comp[z] = i$ 
17:   end if
18:   while  $Q \neq \emptyset$  do
19:     Uzmi čvor  $u$  iz  $Q$ 
20:     for all  $(u, v) \in E$  za koje važi  $comp[v] = 0$  do
21:       Dodaj  $v$  u  $Q$ 
22:        $comp[v] = i$ 
23:     end for
24:   end while
25:    $i = i + 1$ 
26: end while
27: return  $B$ 

```

Primer 17. Posmatrajmo usmereni graf $G = (V, E)$ prikazan na slici 4.5, sa izvor-nim čvorom s i ciljnim čvorom t . Postoje dva puta od s do t : $s \rightarrow a \rightarrow b \rightarrow d \rightarrow t$ i



Slika 4.5: Usmeren graf sa izvornim čvorom s i ciljnim čvorom t .

$s \rightarrow a \rightarrow c \rightarrow d \rightarrow t$.

Primenimo algoritam 5. Najpre biramo proizvoljan put od s do t , na primer $P : s \rightarrow a \rightarrow b \rightarrow d \rightarrow t$, čije su grane $(s, a), (a, b), (b, d), (d, t)$. Zatim transformišemo graf tako što uklonimo grane puta P i dodamo njihove obrnute grane, čime dobijamo $G' = (G \setminus P) \cup P^{-1}$ sa granama

$$\{(a, c), (c, d), (a, s), (b, a), (d, b), (t, d)\}.$$

Na početku je $\text{comp}[v] = 0$ za svaki čvor v i $i = 1$.

Iteracija $i = 1$. Pošto je $i = 1$, u Q se dodaje čvor s i postavlja se $\text{comp}[s] = 1$. U grafu G' čvor s nema izlaznih grana, pa se pretraga završava. Nakon prve iteracije važi $\text{comp}[s] = 1$, dok je comp svih ostalih čvorova jednak 0. Kako je $\text{comp}[t] = 0$, uvećavamo i na 2.

Iteracija $i = 2$. Poslednji čvor na putu P za koji važi $\text{comp}[y] \neq 0$ jeste $y = s$. Njegov sledbenik na P je $z = a$, pa se grana (s, a) dodaje u niz B . Postavlja se $\text{comp}[a] = 2$ i čvor a se dodaje u Q . Pretragom iz a u grafu G' dostižemo čvorove c , d i b (preko grana $a \rightarrow c$, $c \rightarrow d$, $d \rightarrow b$), pa oni dobijaju $\text{comp} = 2$. Čvor t i dalje nije dostignut, jer u G' nijedna grana ne ulazi u t . Kako je $\text{comp}[t] = 0$, uvećavamo i na 3.

Iteracija $i = 3$. Sada čvorovi s, a, b, d imaju $\text{comp} \neq 0$. Poslednji čvor na putu P za koji važi $\text{comp}[y] \neq 0$ jeste $y = d$ (iako b ima nenula vrednost, čvor d se na putu P nalazi bliže čvoru t). Njegov sledbenik na P je $z = t$, pa se grana (d, t) dodaje u niz B i postavlja se $\text{comp}[t] = 3$. Time je uslov spoljne **while** petlje narušen i algoritam se zaustavlja.

Konačni rezultat je niz $s - t$ mostova $B = \{(s, a), (d, t)\}$. Zaista, obe grane leže na svakom putu od s do t , dok grane $(a, b), (a, c), (b, d), (c, d)$ to nisu, jer za svaku od njih postoji alternativni put koji je zaobilazi.

4.3 Algoritam za pronalaženje $s - t$ artikulacionih tačaka

Algoritam za pronalaženje $s - t$ artikulacionih tačaka možemo definisati na sličan način kao prethodni algoritam. Pošto i ovde želimo da se pretraga zaustavi samo u $s - t$ artikulacionim tačkama, kako bismo olakšali postupak njihovog pronalaženja polazni graf ćemo na pogodan način transformisati.

Transformacija grafa. Analogno transformaciji kod mostova, želimo da obezbedimo da su pre nego što se pretraga zaustavi u $s - t$ artikulacionoj tački svi prethodni čvorovi na proizvoljnoj $s - t$ putanji P posećeni. Zbog ovoga, kao i ranije, dodajemo grane obrnutog usmerenja sa putanje P . Kako bismo obezbedili da pretraga bude prekinuta u $s - t$ artikulacionoj tački koristićemo niz *out* kao indikator da li je grananje dozvoljeno iz proizvoljnog čvora.

Kao i ranije, prvo pronalazimo proizvoljnu $s - t$ putanju P , potom transformišemo graf i inicijalizujemo niz *comp* koji nam služi za računanje kojoj komponenti čvorovi pripadaju. Niz *out* za koji smo rekli da će služiti kao indikator da li je grananje dozvoljeno iz datog čvora, inicijalizujemo na vrednost 0 za sve čvorove sa putanje P , izuzev s i t . Iz ostalih čvorova je grananje dozvoljeno, tako da za njih niz *out* inicijalizujemo na vrednost 1. Po istom principu kao i kod mostova, koristimo red Q za pretragu grafa. Kada je pretraga prekinuta pronalazimo poslednji posećeni čvor na P , proglašavamo ga za artikulacionu tačku, i nastavljamo pretragu iz njega, pri čemu moramo da proglasimo da je grananje iz njega dozvoljeno. Kada obrađujemo čvor koji se nalazio na vrhu reda Q , recimo u , njegove susede obrađujemo samo ukoliko je grananje iz čvora u dozvoljeno, ili ukoliko grana (u, v) pripada P^{-1} , pri čemu je v sused od u . Kada je neki od ovih uslova zadovoljen, proveravamo da li čvor v nije već posećen. Takođe proveravamo da li grana (u, v) pripada putanji P^{-1} , pri čemu mora da važi i da je grananje iz čvora v zabranjeno. Ovaj uslov označava da indikator grananja čvora v nije promenjen, odnosno da čvor v nije obrađen. Ukoliko je neki od ovih uslova ispunjen onda možemo da posetimo i obradimo čvor v . Dodajemo čvor v u red Q , a potom određujemo komponentu kojoj pripada. Ukoliko smo se kretali povratnom granom sa putanje P , moramo da dozvolimo i grananje iz čvora v .

Algoritam 6 prikazuje pseudokod ovog algoritma, istih autora kao i algoritam za

pronalaženje $s - t$ mostova [4].

Primer 18. Posmatrajmo ponovo graf $G = (V, E)$ sa slike 4.5, sa izvornim čvorom s i ciljnim čvorom t . Kao i ranije, postoje dva puta od čvora s do čvora t : $s \rightarrow a \rightarrow b \rightarrow d \rightarrow t$ i $s \rightarrow a \rightarrow c \rightarrow d \rightarrow t$.

Primenimo algoritam 6. Biramo isti put $P : s \rightarrow a \rightarrow b \rightarrow d \rightarrow t$. Za razliku od algoritma za $s - t$ mostove, ovde se grane puta ne uklanjaju, već se grafu samo dodaju obrnute grane puta, pa dobijamo $G \cup P^{-1}$ sa dodatnim granama $(a, s), (b, a), (d, b), (t, d)$.

Na početku je $\text{comp}[v] = 0$ za svako v . Vrednost out postavlja se na 0 za unutrašnje čvorove puta P , odnosno $\text{out}[a] = \text{out}[b] = \text{out}[d] = 0$, a na 1 za ostale čvorove: $\text{out}[s] = \text{out}[c] = \text{out}[t] = 1$. Postavlja se $i = 1$. Podsetimo se, čvor u koristi svoje polazne grane tek kada je $\text{out}[u] = 1$; dok je $\text{out}[u] = 0$, iz njega se može ići jedino unazad, po granama iz P^{-1} .

Iteracija $i = 1$. Dodaje se s u Q i $\text{comp}[s] = 1$. Iz s se prelazi u a (grana (s, a)), pa je $\text{comp}[a] = 1$. Čvor a je unutrašnji čvor puta, pa je $\text{out}[a] = 0$; njegova jedina upotrebljiva grana je obrnuta grana (a, s) , koja vodi u već posećen čvor s . Pretraga se zaustavlja. Kako je $\text{comp}[t] = 0$, prelazi se na $i = 2$.

Iteracija $i = 2$. Poslednji čvor na P za koji važi $\text{comp} \neq 0$ jeste $y = a$. On se dodaje u A , postavlja se $\text{out}[a] = 1$, $\text{comp}[a] = 2$ i a se dodaje u Q . Pošto je sada $\text{out}[a] = 1$, iz a se koriste sve grane, pa se dostižu b i c (grane $(a, b), (a, c)$), uz $\text{comp}[b] = \text{comp}[c] = 2$. Iz c , za koje je $\text{out}[c] = 1$, prelazi se u d , pa je $\text{comp}[d] = 2$. Iz d je $\text{out}[d] = 0$, ali obrnuta grana $(d, b) \in P^{-1}$ dozvoljava prelazak u b ; kako je odgovarajuća grana (b, d) upravo grana puta P , postavlja se $\text{out}[b] = 1$. Time se ne dostižu novi čvorovi. Čvor t i dalje nije dostignut, jer je $\text{out}[t] = 0$, a grana (d, t) nije obrnuta grana puta, pa se iz d ne može iskoristiti. Prelazi se na $i = 3$.

Iteracija $i = 3$. Poslednji čvor na P za koji važi $\text{comp} \neq 0$ sada je $y = d$. On se dodaje u A , postavlja se $\text{out}[d] = 1$ i $\text{comp}[d] = 3$. Sada grana (d, t) postaje upotrebljiva, pa se dostiže t i $\text{comp}[t] = 3$. Time je uslov spoljne **while** petlje narušen i algoritam se zaustavlja.

Konačni rezultat je niz $s - t$ artikulacionih tačaka $A = \{a, d\}$. Zaista, svaki put od s do t prolazi kroz čvorove a i d , dok se čvorovi b i c mogu zaobići, pa oni nisu $s - t$ artikulacione tačke.

Algoritam 6 $s - t$ artikulacione tačke

Ulaz: Graf $G = (V, E)$, $s, t \in V$

Izlaz: A , pri čemu je A niz $s - t$ artikulacionih tačaka

```

1: Izračunaj proizvoljnu putanju  $P$  od  $s$  do  $t$  u  $G$ 
2: Transformiši graf  $G = G \cup P^{-1}$ 
3: for all  $v \in V$  do
4:    $comp[v] = 0$ 
5: end for
6: for all  $x \in V$  do
7:   if  $x \in P \setminus \{s, t\}$  then
8:      $out[x] = 0$ 
9:   else
10:     $out[x] = 1$ 
11:   end if
12: end for
13:  $i = 1$ 
14: while  $comp[t] = 0$  do
15:   if  $i = 1$  then
16:     Dodaj  $s$  u  $Q$ ;  $comp[s] = 1$ 
17:   else
18:     Pronađi poslednji čvor  $y$  na  $P$  tako da je  $comp[y] \neq 0$ 
19:     Dodaj  $y$  u  $A$ 
20:      $out[y] = 1$ 
21:     Dodaj  $y$  u  $Q$ 
22:      $comp[y] = i$ 
23:   end if
24:   while  $Q \neq \emptyset$  do
25:     Uzmi čvor  $u$  iz  $Q$ 
26:     for all  $(u, v) \in E$  za koje važi  $out[u] = 1$  ili  $(u, v) \in P^{-1}$  do
27:       if  $comp[v] = 0$  ili (  $(u, v) \in P^{-1}$  i  $out[v] = 0$  ) then
28:         Dodaj  $v$  u  $Q$ 
29:          $comp[v] = i$ 
30:       if  $(v, u) \in P$  then
31:          $out[v] = 1$ 
32:       end if
33:     end if
34:   end for
35:   end while
36:    $i = i + 1$ 
37: end while
38: return  $A$ 

```

4.4 Analiza složenosti

Algoritmi za određivanje $s - t$ mostova i $s - t$ artikulacionih tačaka u usmerenim grafovima se sastoje od tri celine, koje moramo da nezavisno analiziramo kako bismo odredili složenost algoritma. Prva je pretraga grafa kako bi se pronašla proizvoljna $s - t$ putanja P . Ovo je moguće izvesti uz pomoć algoritma pretrage grafa u širinu, tako da je složenost ovog dela algoritma $\mathcal{O}(m + n)$, pri čemu je n broj čvorova a m grana u grafu. Za pretragu unapred koja biva prekidana u $s - t$ mostovima (artikulacionim tačkama) i posećuje samo neposećene čvorove, koristićemo takođe pretragu u širinu, tako da je i složenost ovog dela algoritma linearna u funkciji veličine grafa. Treći deo algoritma je pronalaženje poslednjeg posećenog čvora na putanji P , kada se pretraga prekine u $s - t$ mostu (artikulacionoj tački). Pošto se ova pretraga odvija unutar petlje, naizgled može delovati da će ovo pokvariti složenost algoritma, međutim u svakoj iteraciji petlje potraga za poslednjim posećenim čvorom nastavlja tamo gde je stala u prethodnoj iteraciji, tako da se putanjom P prođe ukupno jednom tokom celog izvršavanja petlje. Kako su ostale operacije konstante složenosti, sledi da je ukupna složenost oba razmatrana algoritma $\mathcal{O}(m + n)$.

Glava 5

Implementacija i evaluacija algoritama

Implementacija algoritama rađena je u programskom jeziku C++. Izvorni kod algoritma za pronalaženje jakih artikulacionih tačaka sadrži oko 200 linija koda, dok je kod algoritma za pronalaženje jakih mostova broj linija koda oko 250. Za implementaciju algoritama za pronalaženje b-artikulacionih tačaka i b-mostova upotrebljeno je oko 70 i 250 linija koda, respektivno. Izvorni kod algoritama za pronalaženje $s - t$ artikulacionih tačaka i mostova se sastoji od po 100 linija koda, za oba algoritma. Izvorni kod svih algoritama može se naći na sledećem github repozitorijumu: <https://github.com/lazars01/MasterRad>.

U ovom poglavlju prikazana je i eksperimentalna evaluacija algoritama opisanih u prethodnim poglavljima. Za svaki algoritam posebno je opisan način generisanja nasumičnih grafova nad kojima je testiranje izvršeno, zatim su dati rezultati merenja vremena izvršavanja za različite veličine grafova, grafički prikaz dobijenih rezultata, kao i poređenje sa teorijskim rezultatima.

U svim eksperimentima vreme izvršavanja meri se za samu algoritamsku obradu grafa, bez vremena potrebnog za generisanje ulaznih podataka i ispis rezultata. Za svaku od posmatranih veličina grafa generisano je 100 test instanci i kao konačan rezultat prikazano prosečno vreme izvršavanja. Konfiguracija računara na kojoj su vršeni eksperimenti sastoji se od procesora Intel(R) Core(TM) i5-3320M CPU sa 2.60GHz, 16 GB RAM memorije i integrisane grafičke kartice. Za iscrtavanje grafika korišćeni su ugrađeni latex alati.

5.1 Evaluacija algoritma za pronalaženje jakih artikulacionih tačaka i jakih mostova

Generisanje nasumičnih grafova

Da bismo se osigurali da je nasumični graf koji generišemo jako povezan, dodajemo mu usmereni ciklus. Ciklus dodajemo tako što generišemo jednu nasumičnu permutaciju čvorova grafa, pa potom dodajemo usmerenu granu između susednih čvorova formirajući ciklus. Nakon generisanja početne strukture koja obezbeđuje jaku povezanost grafa, preostale grane dodaju se nasumičnim izborom uređenih parova čvorova, sa proizvoljno malom verovatnoćom.

Za potrebe evaluacije ovog algoritma generisani su nasumični jako povezani usmereni grafovi. Svaki test primer određen je brojem čvorova n . Pošto se tokom generisanja test primera formira ciklus, a potom grane nasumično dodaju, broj grana nije fiksiran. Zbog toga smo veličinu grafa parametrizovali brojem čvorova i prosečnim brojem grana u tom klasteru test primera.

Rezultati merenja

Tabela 5.1 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje jakih artikulacionih tačaka, za različite veličine ulaznih grafova.

Broj čvorova	Prosečan broj grana	Prosečno vreme izvršavanja (ms)
10	17.69	0.462
20	56.59	0.959
50	291.46	4.121
75	625.43	6.066
100	1077.51	7.998
125	1660.53	9.427
150	2367.07	12.028
175	3211.22	14.750
200	4159.34	19.026

Tabela 5.1: Vreme izvršavanja algoritma za pronalaženje jakih artikulacionih tačaka

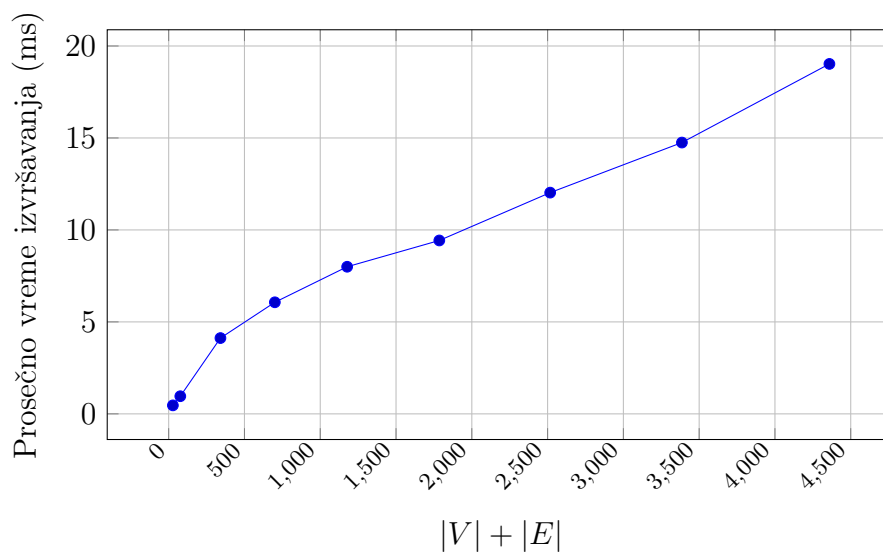
Tabela 5.2 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje jakih mostova, za različite veličine ulaznih grafova.

Broj čvorova	Prosečan broj grana	Prosečno vreme izvršavanja (ms)
10	18	0.538
20	56.93	1.356
50	288.36	5.545
75	621.27	10.382
100	1080.18	17.577
125	1663.6	25.948
150	2376.77	37.872
175	3195	48.442
200	4151.29	65.032

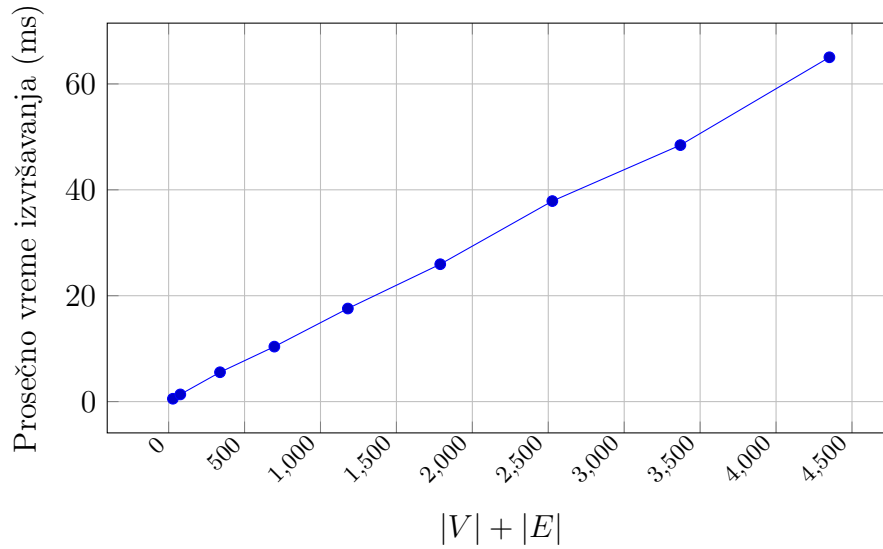
Tabela 5.2: Vreme izvršavanja algoritma za pronalaženje jakih mostova

Grafički prikaz rezultata

Na grafiku 5.1 prikazana je zavisnost vremena izvršavanja algoritma za pronalaženje jakih artikulacionih tačaka u odnosu na veličinu grafa, dok je na grafiku 5.2 isto to prikazano za algoritam za pronalaženje jakih mostova.



Slika 5.1: Vreme izvršavanja algoritma za pronalaženje jakih artikulacionih tačaka



Slika 5.2: Vreme izvršavanja algoritma za pronalaženje jakih mostova

Tokom analize vremenske složenosti algoritama za pronalaženja jakih artikulacionih tačaka i jakih mostova zaključili smo da ona zavisi od vremenske složenosti algoritma za pronalaženje dominatora. Kako smo ranije naveli, u implementaciji smo koristili Lengauer Tarjanov algoritam za pronalaženje dominatora, vremenske složenosti $\mathcal{O}(m \cdot \alpha(m, n))$. Otuda sledi da je i vremenska složenost ovih algoritama $\mathcal{O}(m \cdot \alpha(m, n))$, odnosno praktično linearna po veličini grafa. Analizirajući ova dva grafika zaključujemo da prikazane funkcije podsećaju na linearnu funkciju, što potvrđuje rezultate teorijske analize algoritama.

5.2 Evaluacija algoritma za pronalaženje b-artikulacionih tačaka i b-mostova

Generisanje nasumičnih grafova

Za evaluaciju ovog algoritma generisani su nasumični jako povezani usmereni grafovi, čiji je odgovarajući neusmeren graf bipovezan. Njihovo generisanje radićemo na isti način kao kod jako povezanih grafova, zato što ciklus kroz svih n čvorova grafa obezbeđuje i bipovezanost. Svaki test primer određen je brojem čvorova n . Pošto se tokom generisanja test primera formira ciklus, a potom grane nasumično dodaju, broj grana nije fiksiran. Zbog toga smo veličinu grafa parametrizovali brojem čvorova

i prosečnim brojem grana u tom klasteru test primera.

Rezultati merenja

Tabela 5.3 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje b-artikulacionih tačaka za različite veličine ulaznih grafova.

Broj čvorova	Prosečan broj grana	Prosečno vreme izvršavanja (ms)
10	17.75	0.464
20	55.32	1.513
50	293.55	12.808
75	620.63	11.138
100	1079.2	15.516
125	1675.09	22.748
150	2370.09	35.018
175	3205.85	48.374
200	4160.45	60.260

Tabela 5.3: Vreme izvršavanja algoritma za pronalaženje b-artikulacionih tačaka

Tabela 5.4 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje b-mostova za različite veličine ulaznih grafova.

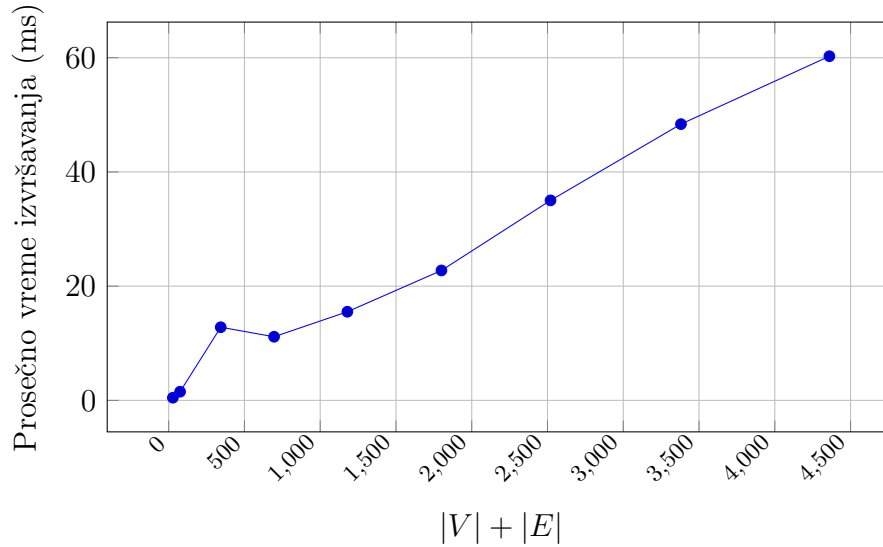
Broj čvorova	Prosečan broj grana	Prosečno vreme izvršavanja (ms)
10	18.21	1.548
20	55.91	7.496
50	286.25	84.505
75	623.96	244.061
100	1076.11	502.761
125	1662.56	924.304
150	2371.82	1583.46
175	3206.76	2389.52
200	4166.08	3553.17

Tabela 5.4: Vreme izvršavanja algoritma za pronalaženje b-mostova

Grafički prikaz rezultata

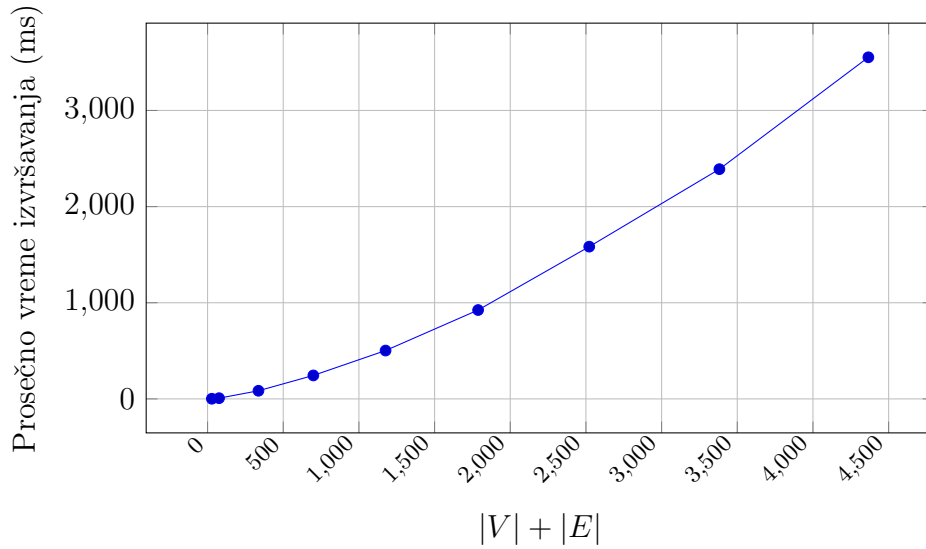
Na grafiku 5.3 prikazana je zavisnost vremena izvršavanja algoritma za pronalaženje b-artikulacionih tačaka u odnosu na veličinu grafa, dok je na grafiku 5.4 isto

to prikazano za algoritam za pronalaženje b-mostova.



Slika 5.3: Vreme izvršavanja algoritma za pronalaženje b-artikulacionih tačaka

Kako je vremenska složenost algoritma za pronalaženje b-artikulacionih tačaka $\mathcal{O}((n - j) \cdot (m + n))$, pri čemu je n broj čvorova, m broj grana a j broj jakih artikulacionih tačaka, zaključujemo da ona opada kako se povećava broj b-artikulacionih tačaka. Sa grafika vidimo da funkcija prvo raste brzo, a potom usporava, što potvrđuje teorijsku analizu jer je teže održati bipovezanost kako graf raste, pa samim tim ima i više b-artikulacionih tačaka.



Slika 5.4: Vreme izvršavanja algoritma za pronalaženje b-mostova

Na slici 5.4 vidimo da funkcija koja prikazuje zavisnost vremena izvršavanja algoritma od veličine grafa podseća na kvadratnu funkciju. Vremenska složenost algoritma za pronalaženje b-mostova je $\mathcal{O}(m \cdot n^2)$. Kako je u gustim grafovima broj grana m reda $\mathcal{O}(n^2)$, a generisani grafovi imaju znatno više grana nego čvorova, sledi da praktični rezultati potvrđuju teorijsku analizu.

5.3 Evaluacija algoritma za pronalaženje $s - t$ artikulacionih tačaka i $s - t$ mostova

Generisanje nasumičnih grafova

Za potrebe evaluacije algoritama za pronalaženje $s - t$ artikulacionih tačaka i $s - t$ mostova generisani su nasumični usmereni grafovi i na proizvoljan način birani čvorovi s i t . Prilikom generisanja obezbeđuje se da u grafu postoji bar jedan usmeren put od čvora s do čvora t , budući da se algoritam oslanja na postojanje takve putanje. Postojanje puta od čvora s do čvora t obezbeđujemo dodavanjem usmerenog puta koji prolazi kroz sve čvorove, nasumičnim redosledom, a polazi od čvora s i završava u čvoru t .

Rezultati merenja

Tabela 5.5 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje $s - t$ artikulacionih tačaka, za različite veličine ulaznih grafova.

Broj čvorova	Broj grana	Prosečno vreme izvršavanja (ms)
10	60	0.184
20	240	0.435
50	700	1.024
75	1000	1.368
100	1500	1.812
125	1800	2.147
150	2200	2.336
175	2500	2.610
200	3000	2.948

Tabela 5.5: Vreme izvršavanja algoritma za pronalaženje $s - t$ artikulacionih tačaka

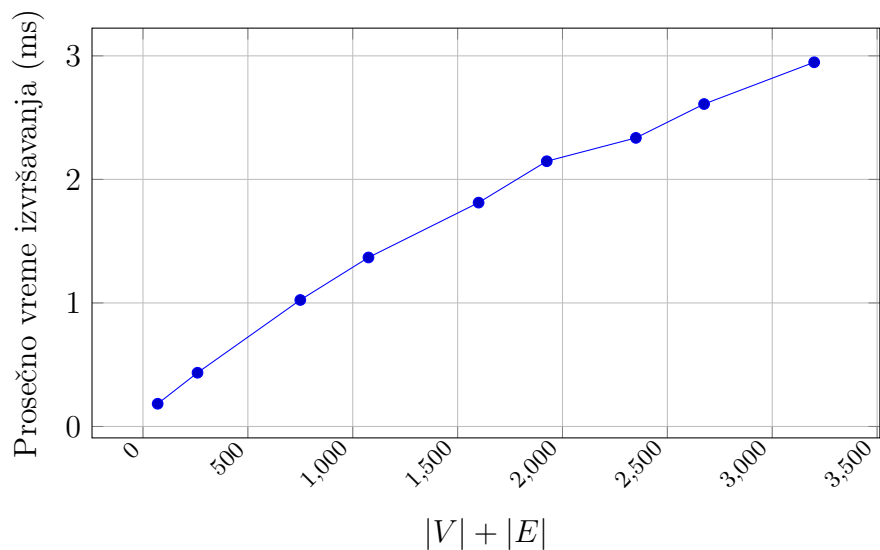
Tabela 5.6 prikazuje rezultate merenja vremena izvršavanja algoritma za pronalaženje $s - t$ mostova, za različite veličine ulaznih grafova.

Broj čvorova	Broj grana	Prosečno vreme izvršavanja (ms)
10	60	0.047
20	240	0.064
50	700	0.121
75	1000	0.173
100	1500	0.243
125	1800	0.279
150	2200	0.333
175	2500	0.385
200	3000	0.428

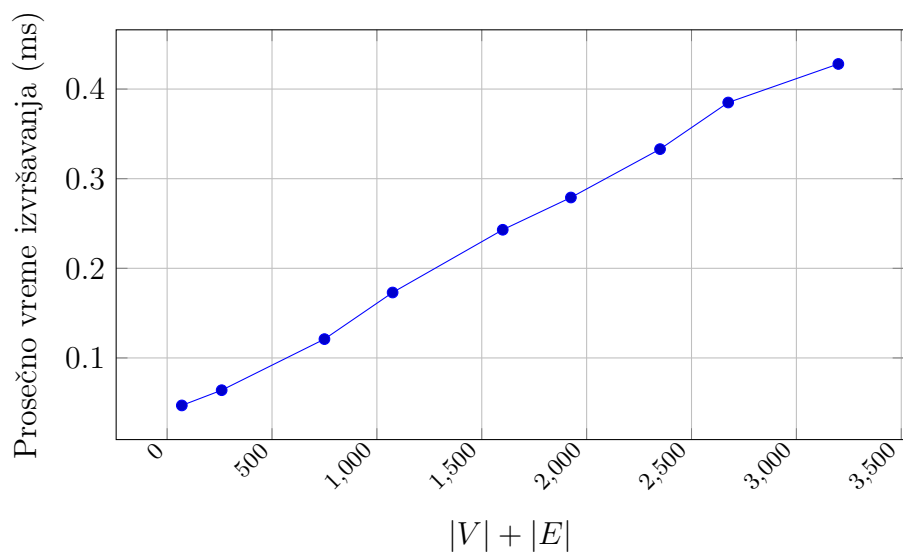
Tabela 5.6: Vreme izvršavanja algoritma za pronalaženje $s - t$ mostova

Grafički prikaz rezultata

Na grafiku 5.5 prikazana je zavisnost vremena izvršavanja algoritma za pronalaženje $s - t$ artikulacionih tačaka u odnosu na veličinu grafa, dok je na grafiku 5.6 isto to prikazano za algoritam za pronalaženje $s - t$ mostova.



Slika 5.5: Vreme izvršavanja algoritma za pronalaženje $s - t$ artikulacionih tačaka



Slika 5.6: Vreme izvršavanja algoritma za pronalaženje $s - t$ mostova

Sa grafika zaključujemo da zavisnost vremena izvršavanja u odnosu na veličinu grafa za oba algoritma možemo modelovati linearnom funkcijom, što je u skladu sa teorijskim rezultatima za vremensku složenost ova dva algoritma, koja je $\mathcal{O}(m + n)$, pri čemu je m broj grana a n broj čvorova u grafu.

Glava 6

Zaključak i mogući pravci daljeg rada

U ovom radu prikazani su algoritmi za pronalaženje različitih tipova kritičnih čvorova i grana u usmerenim grafovima u pogledu održanja broja komponenti povezanosti. Posmatrani su jaki mostovi i jake artikulacione tačke, b-mostovi i b-artikulacione tačke, kao i $s - t$ mostovi i $s - t$ artikulacione tačke. Iako su svi ovi pojmovi motivisani istom osnovnom idejom, odnosno ispitivanjem šta se dešava sa usmerenim grafom nakon uklanjanja jednog čvora ili jedne grane, njihova precizna definicija zavisi od vrste povezanosti koja se posmatra.

Za jake artikulacione tačke i jake mostove istaknuta je veza sa dominatorima u grafovima toka. Ta veza je značajna jer omogućava da se problem analize jako povezanog usmerenog grafa svede na poznate algoritamske tehnike za računanje dominatora. Kod b-artikulacionih tačaka i b-mostova dodatno se posmatra struktura pratećeg neusmerenog grafa, pa se pored jake povezanosti uzima u obzir i bipovezanost. Na kraju, kod $s - t$ artikulacionih tačaka i $s - t$ mostova pažnja je usmerena na očuvanje dostižnosti između dva izdvojena čvora, što predstavlja drugačiji, lokalniji pogled na problem.

Kroz rad je pokazano da se za srodne probleme često koriste različite ideje: dominatorska stabla, transponovani graf, bipovezane komponente i posebne transformacije grafa. Zbog toga poređenje ovih algoritama nije korisno samo sa stanovišta njihove vremenske složenosti, već i sa stanovišta razumevanja koje osobine grafa svaki od njih koristi. Eksperimentalna evaluacija dodatno omogućava da se teorijska analiza dopuni praktičnim ponašanjem algoritama na grafovima različitih veličina. Takođe, ona je važna i kako bi potvrdila teorijsku analizu.

Kao pravac daljeg rada mogu se izdvojiti implementacija i detaljnije testiranje algoritama nad većim skupovima grafova, kao i poređenje rezultata na nasumično generisanim i realnim mrežama. Takođe bi bilo korisno razmotriti mogućnosti optimizacije implementacije, naročito za guste grafove i grafove sa velikim brojem čvorova. Dodatno, mogle bi se razmatrati i dodatne vrste artikulacionih tačaka i mostova u usmerenim grafovima, poput 2-jakih mostova, k -mostova, dominatorskih mostova i slično [7] [8].

Bibliografija

- [1] Stephen Alstrup, Dov Harel, Peter W. Lauridsen, and Mikkell Thorup. Dominators in linear time. *SIAM Journal on Computing*, 28(6):2117–2132, 1999.
- [2] Jørgen Bang-Jensen and Gregory Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer Monographs in Mathematics. Springer-Verlag, London, 2 edition, 2009.
- [3] Adam L. Buchsbaum, Loukas Georgiadis, Haim Kaplan, Anne Rogers, Robert E. Tarjan, and Jeffery R. Westbrook. Linear-time algorithms for dominators and other path-evaluation problems. *SIAM Journal on Computing*, 38(4):1533–1573, 2008.
- [4] Massimo Cairo, Shahbaz Khan, Romeo Rizzi, Sebastian Schmidt, Alexandru I. Tomescu, and Elia C. Zirondelli. A simplified algorithm computing all s-t bridges and articulation points. *Discrete Applied Mathematics*, 305:103–108, 2021.
- [5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, Cambridge, MA, 3 edition, 2009.
- [6] J. JaJa G. N. Frederickson. Approximation Algorithms for Several Graph Augmentation Problems. *SIAM Journal on Computing*, 2(10):270–283, 1981.
- [7] Loukas Georgiadis, Giuseppe F. Italiano, Luigi Laura, and Nikos Parotsidis. 2-edge connectivity in directed graphs. *ACM Transactions on Algorithms*, 13(1):9:1–9:24, 2016.
- [8] Loukas Georgiadis, Giuseppe F. Italiano, Luigi Laura, and Nikos Parotsidis. 2-vertex connectivity in directed graphs. *Information and Computation*, 261:248–264, 2018.

- [9] Giuseppe F. Italiano, Luigi Laura, and Federico Santaroni. Finding strong bridges and strong articulation points in linear time. *Theoretical Computer Science*, 447:74–84, 2012.
- [10] Raed Jaber. On computing the 2-vertex-connected components of directed graphs. *Discrete Applied Mathematics*, 204:164–172, 2016.
- [11] Raed Jaber. b-articulation points and b-bridges in strongly biconnected directed graphs, 2020.
- [12] Thomas Lengauer and Robert Endre Tarjan. A fast algorithm for finding dominators in a flowgraph. *ACM Transactions on Programming Languages and Systems*, 1(1):121–141, July 1979.
- [13] Robert Endre Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.
- [14] Stephane Grumbach Zhilin Wu. Feasibility of motion planning on acyclic and strongly connected directed graphs. *Discrete Applied Mathematics*, 158(9):1017–1028, 2010.