

UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET



Jelena Marković

**ALGORITMI ZA PRONALAZENJE
KONVEKSNOG MNOGOUGLA
NAJMANJEG OBIMA KOJI PRESECA DATI
SKUP DUŽI**

master rad

Beograd, 2020.

Mentor:

dr Milan BANKOVIĆ, docent
Univerzitet u Beogradu, Matematički fakultet

Članovi komisije:

dr Miodrag ŽIVKOVIĆ, redovni profesor
Univerzitet u Beogradu, Matematički fakultet

dr Predrag JANIČIĆ, redovni profesor
Univerzitet u Beogradu, Matematički fakultet

Datum odbrane: _____

*Mojoj porodici,
hvala im na beskrajnom strpljenju*

Naslov master rada: Algoritmi za pronalaženje konveksnog mnogougla najmanjeg obima koji preseca dati skup duži

Rezime: U ovom radu razmatraju se algoritmi za pronalaženje mnogougla najmanjeg obima koji sa svakom od duži iz datog skupa ima neprazan presek. Za ovaj problem još uvek ne postoji polinomski algoritam, a ne zna se ni da li je problem NP-težak. U radu prikazujemo jedan od najznačajnijih egzaktnih algoritama eksponencijalne složenosti za ovaj problem, kao i nekoliko aproksimativnih algoritama. U novije vreme, u literaturi su opisana i dva algoritma polinomske složenosti ($O(n^5)$ i $O(n^4)$) koji tretiraju specijalni slučaj opisanog problema kada su duži međusobno disjunktne. Međutim, pomenuti algoritmi u literaturi nisu opisani dovoljno precizno i potpuno, tako da se njihova korektnost treba uzeti sa rezervom. U ovom radu razmatramo i ove algoritme, uz diskusiju o njihovim spornim i nedovoljno razjašnjenim delovima. Najzad, predlažemo izvesne dopune i poboljšanja ovih algoritama. Sa predloženim unapređenjima, ukupna asimptotska složenost algoritma je $O(n^3 \log n)$, što predstavlja poboljšanje u odnosu na aktuelne rezultate opisane u literaturi. Dokaz korektnosti predloženih dopuna i njihova programska implementacija ostaje pravac budućeg rada.

Ključne reči: računarska geometrija, konveksni mnogougao, konveksni omotač, najkraći putevi unutar mnogougla, minimalni obim

Sadržaj

1	Uvod	1
2	Egzaktni eksponencijalni algoritam	4
2.1	Računanje kritične sekvence	6
2.2	Najkraći put između susjednih finih ivičnih duži	16
2.3	Složenost algoritma u odsustvu problematičnih ivičnih duži	22
2.4	Najteži slučaj - problematične ivične duži	23
2.5	Implementacija	25
3	Aproksimativni algoritmi	28
3.1	Algoritam zasnovan na minimalnom razapinjućem krugu	28
3.2	Unapređeni aproksimativni algoritmi	36
4	Polinomski algoritmi za disjunktne duži	45
4.1	Algoritam složenosti $O(n^5)$	45
4.2	Algoritam složenosti $O(n^4)$	53
5	Zaključak	60
	Bibliografija	61

Glava 1

Uvod

Pažljivim razmatranjem mnogih problema poput TSP problema, problema pronalaženja minimalnog razapinjućeg stabla, problema pronalaženja konveksnog omotača i mnogih drugih, možemo uočiti da se njihovi ulazi mogu razmatrati kao skupovi tačaka u ravni. Iako je, u teoriji, rešenje ovakvih problema egzaktno određeno, u praksi tačnost dobijenog rešenja može zavistiti od mnogo faktora, kao što su preciznost našeg računara, ili preciznost sa kojom su zadati ulazni podaci. Pritom, imamo u vidu da su ulazni podaci često dobijeni kao rezultat nekih eksperimenata, pa samim tim prirodno sadrže izvesnu grešku. Zbog toga je često u ovakvim problemima potrebno definisati izvesno dozvoljeno odstupanje, a ono se formalno definiše pojmom neprecizne tačke.

Za tačku kazemo da je neprecizna, ako je zadata regijom koja je sadrži. U zavisnosti od potrebe, mogu se razmatrati regije različitih oblika, poput duži, kvadrata, krugova, itd. Sada se, na primer, problem pronalaženja konveksnog omotača neprecizno zadatih tačaka svodi na pronalaženje konveksnog mnogougla koji sa odgovarajućim regijama ima neprazan presek (često se kaže i da mnogougao ubada¹ ove regije). Kako ovaj mnogougao nije jedinstven, obično se razmatra pronalaženje takvog mnogougla koji maksimizuje ili minimizuje neki parametar, poput obima ili površine.

U ovom radu razmatramo problem pronalaženja konveksnog mnogougla najmanjeg obima koji ima neprazan presek sa svakom od duži iz zadatog skupa. Za ovaj problem u ovom trenutku još uvek nije poznato da li je, u opštem slučaju, rešiv

¹Pojam ubadanja se u literaturi takođe definiše na različite načine. Ponekad se zahteva da presek, osim što je neprazan, zadovoljava neke dodatne osobine. Na primer, ako su regije duži, možemo zahtevati da bar jedna od krajnjih tačaka duži pripada mnogouglu. U ovom radu se razmatra najopštiji slučaj, tj. zahteva se samo da presek bude neprazan.

u polinomskom vremenu, kao ni da li je NP-težak. Otuda su svi poznati egzaktni algoritmi za rešavanje ovog problema eksponencijalne složenosti u najgorem slučaju. Jedan takav algoritam razmatramo u ovom radu [2, 1]. Takođe, razmatramo i nekoliko aproksimativnih algoritama [1, 3] čija je složenost značajno manja u odnosu na egzaktnu, a kvalitet dobijenog rešenja (u smislu obima dobijenog mnogougla) nije mnogo lošiji, što ih čini primenljivijim u praksi.

Ono što je, s obzirom na najnovije rezultate [9, 10], izvesno, je da je u slučaju disjunktih duži problem rešiv u polinomskom vremenu. Nažalost, algoritmi izloženi u radovima [9, 10] nisu dovoljno precizno i detaljno opisani, a nedostaje i prototipska implementacija istih. U ovom radu ćemo, između ostalog, pokušati i da razjasnimo i preciziramo spomenute algoritme polinomske složenosti u delovima u kojima su oni ostali nejasni, a takođe ćemo predložiti i izvesna poboljšanja tih algoritama. Dokaz korektnosti predloženih poboljšanja, kao i njihova programska implementacija, ostaje predmet budućeg rada.

Glavni doprinosi ovog rada su:

- Dat je pregled nekoliko najznačajnijih algoritama za rešavanje navedenog problema. Iako su opisi algoritama i dokazi korektnosti u najvećoj meri preuzeti iz literature, u pojedinim delovima data su dodatna pojašnjenja i preciziranja, gde je bilo potrebno.
- U toku rada na ovoj tezi započeta je, i u najvećoj meri završena, implementacija egzaktnog algoritma eksponencijalne složenosti opisanog u radovima Rapaporta i Hasanzadeha [2, 1]. Jednom kada postane potpuno funkcionalna, ova implementacija će, koliko je autoru poznato, biti prva javno dostupna implementacija ovog algoritma.
- Algoritmi polinomske složenosti [9, 10] su razjašnjeni i dopunjeni u delovima koji su ostali nejasni u njihovim originalnim opisima, uz poboljšanje ukupne asimptotske složenosti.

Ostatak ovog rada ima sledeću strukturu. U glavi 2 razmatramo egzaktni eksponencijalni algoritam predložen od strane Rapaporta i Hasanzadeha [1] koji rešava opisani problem u opštem slučaju. U glavi 3 dajemo pregled važnijih aproksimativnih algoritama koji se zasnivaju na aproksimaciji traženog konveksnog mnogougla jednostavnijim geometrijskim oblicima. U glavi 4 razmatramo najnovije rezultate koji opisuju polinomske algoritme za slučaj disjunktih duži. Pomenute dopune i

poboljšanja ovih algoritama takođe razmatramo u glavi 4. U glavi 5 dajemo zaključke i razmatramo pravce daljeg rada.

Glava 2

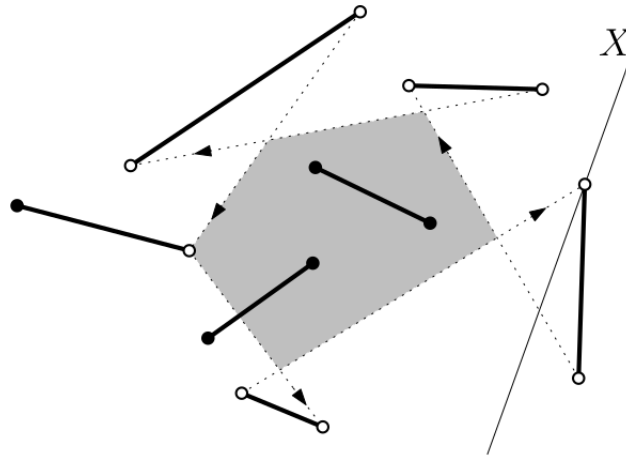
Egzaktni eksponencijalni algoritam

U ovoj glavi razmatramo egzaktni algoritam eksponencijalne složenosti koji je prvi put prikazan u Rapaportovom radu [2], a dodatno preciziran u radu Rapaporta i Hasanzadeha [1]. Ovaj algoritam ne pretpostavlja da su duži međusobno disjunktne, već samo da se ne preklapaju (potpuno ili delimično), kao i da se ne dodiruju u krajnjim tačkama. Zadatak je da konstruišemo konveksni mnogougao koji ima neprazan presek sa svakom od ovih duži i koji pritom od svih konveksnih mnogouglova koji to zadovoljavaju ima najmanji obim.

U nastavku teksta podrazumevamo istu notaciju kao u [1]. Za skup duži $\mathcal{S} = \{s_1, s_2, \dots, s_k\}$ i usmerenu pravu L kažemo da je prava L ekstremna u odnosu na ovaj skup ako postoji duž s_i takva da je ili cela na L ili je jedna njena krajnja tačka na L , a ostatak duži s_i se nalazi u desnoj poluravni¹ određenoj pravom L , dok za druge duži važi da svaka sa zatvorenom levom poluravni koja je određena sa L (u oznaci $H^l(L)$) ima neprazan presek. Krajnju tačku duži s_i koja leži na L zovemo kritična tačka. Duž koja ima jednu ili dve kritične tačke zovemo kritična duž. Ekstremna prava je kritična ekstremna prava ako sadrži barem dve kritične tačke koje pripadaju različitim dužima.

Na slici 2.1 beli kružići odgovaraju kritičnim tačkama. Prava X je ekstremna prava, a kritične ekstremne prave su nacrtane isprekidanim linijama. Ono što možemo da zapazimo je da postoji mnogougao određen presecima poluravni $H^l(L)$ (obojen sivo na slici 2.1). Jasno je da svaki konveksan mnogougao koji ima ne-

¹Desna poluravan može da se definiše pomoću vektora pravca: kako je prava L usmerena, postoji vektor \vec{v} koji određuje njen pravac i smer. Neka su tačke A i B na pravoj L takve da je vektor $\overrightarrow{AB} = \vec{v}$. Tada tačka C van prave L pripada otvorenoj desnoj (levoj) poluravni ako i samo ako je trojka tačaka (A, B, C) orijentisana u smeru kazaljke na satu (suprotno od kazaljke na satu). Zatvorena desna (leva) poluravan je unija otvorene desne (leve) poluravni i same prave L .



Slika 2.1: Kritične ekstremne prave (označene isprekidanim linijama) [1]

prazan presek sa svim datim dužima mora da sadrži ovaj mnogougao, pa i onaj sa najmanjim obimom. U suprotnom, postojala bi duž sa kojom bi imao prazan presek, a to nećemo. Ovaj mnogougao nazivamo siguran mnogougao. Duži koje imaju neprazan presek sa tim mnogougolom ne utiču na rezultat², pa se njihovim brisanjem, ne menja traženi konveksni mnogougao najmanjeg obima. Ove duži nazivaćemo nepotrebne duži.

Sada kada smo odbacili nepotrebne duži, ostaje nam da one koje su nam neophodne za konstrukciju rešenja podelimo u neke kategorije, na osnovu čega ćemo znati kako da se prema kojoj ponašamo.

Da bismo odredili u kojoj kategoriji je data duž s_i , razmatrajmo pravu L koja sadrži s_i i orijentišimo je tako da za levu stranu odaberemo onu gde imamo makar jednu duž. Sa CP_i označimo skup kritičnih tačaka duži u \mathcal{S} iz koga su izbačene kritične tačke koje se nalaze na s_i .

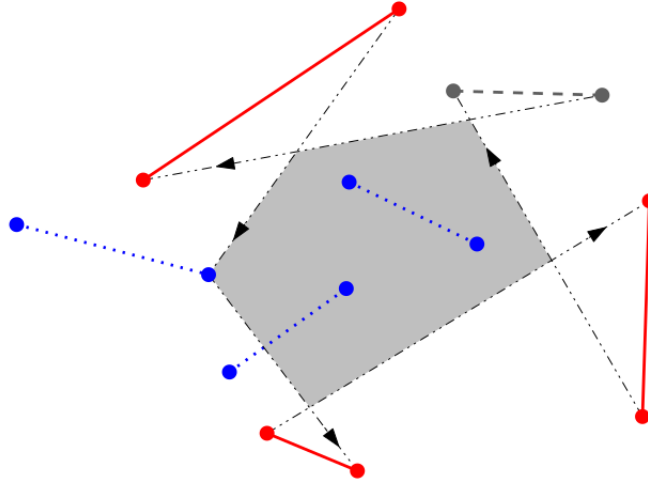
Ako se sve tačke iz CP_i nalaze u $H^l(L)$, onda duž s_i zovemo *fina ivična duž* (engl. *nice rim segment*). Ispostavlja se da su ovakve duži deo granice konveksnog mnogougla najmanjeg obima, odnosno da će se njegova ivica koja ima presek sa s_i „odbiti” od s_i . Drugim rečima, presek duži s_i sa mnogougolom biće jedna tačka duži s_i koja će ujedno biti teme mnogougla. Ovo teme nazivamo reflektujuće teme.

Ako nam se pak dogodi da imamo neku duž iz \mathcal{S} koja ima prazan presek sa $H^l(L)$, to će za našu posmatranu duž s_i značiti da ona ima barem jednu duž sa leve strane i barem jednu duž sa desne strane. Ove duži je potrebno spojiti granicom

²Baš kao što unutrašnje tačke konveksnog omotača ne utiču na konveksni omotač

konveksnog mnogougla i ta granica prolazi kroz s_i . Ovakve duži nazivamo *normalne duži* (engl. *normal line segments*).

Treći tip duži su *problematične ivične duži* (engl. *pesky rim segments*). Motivacija za ovakav naziv verovatno je proistekla iz toga što težina problema leži u ovim dužima. One su odgovorne za eksponencijalno vreme izvršavanja algoritma. Za problematičnu ivičnu duž s_i važi da svaka druga duž ima barem jednu tačku u $H^l(L)$, ali istovremeno, postoji tačka iz CP_i koja je sa desne strane s_i te s_i može da se ponaša i kao fina ivična duž, ali i kao normalna duž. Fine ivične i problematične ivične duži jednim imenom nazivamo *ivične duži*.



Slika 2.2: Tipovi duži: crvene - fine ivične, plave - normalne, crne - problematične ivične [1]

2.1 Računanje kritične sekvence

Podsetimo se još jednom šta je to ekstremna kritična prava. To je ekstremna prava koja prolazi kroz barem dve kritične tačke koje potiču sa različitih kritičnih duži. Neka je $\phi(L) \in [0, 2\pi)$ polarni ugao usmerene prave L . Kada uzmemo sve ekstremne kritične prave, stavimo ih u niz i uredimo rastuće po pomenutom uglu dobijamo kritičnu sekvencu. Označimo je sa $E(\mathcal{S})$, gde je \mathcal{S} skup duži za koji tražimo konveksni mnogougao najmanjeg obima koji sa svakom duži iz tog skupa ima neprazan presek.

U ovom poglavlju objasnićemo kako se može izračunati kritična sekvenca i u kojoj složenosti to možemo da uradimo. Algoritam potiče iz Rapaportovog rada [2], mada je originalna ideja prezentovana u Edelsbrunerovoj knjizi iz 1987. godine [6].³ Obrazložimo najpre zašto nam je ovako nešto uopšte potrebno.

Definicija 2.1.1 *Potporna prava za dati mnogougao je prava koja sa njim ima barem jednu zajedničku tačku i za koju važi da je mnogougao u potpunosti sadržan u jednoj od dve zatvorene poluravni određene tom pravom.*

Definicija 2.1.2 *Reflektujuće teme mnogougla \mathcal{P} u odnosu na duž s_i je teme mnogougla P za koje važi da je prava kroz s_i potporna prava mnogougla u tom temenu.*

Definicija 2.1.3 *Minimalna presecajuća poluravan za dati skup duži \mathcal{S} je poluravan za koju važi da ima neprazan presek sa svim dužima iz \mathcal{S} i da nijedna druga poluravan koja je njen podskup nema to svojstvo.*

Incidentnost sa reflektujućim temenom nije zagantovana za svaku duž. Samo ako je duž fina ivična znamo da na njoj postoji reflektujuće teme. Tu činjenicu dokazujemo kroz nekoliko narednih lema.

Lema 2.1.4 *Neka je P bilo koje teme mnogougla \mathcal{P} koji je konveksni mnogougao najmanjeg obima koji preseca skup duži \mathcal{S} . Tada je P incidentno sa granicom minimalne presecajuće poluravni od \mathcal{S} .⁴*

Dokaz. Pretpostavimo da P nije incidentno ni sa jednom granicom neke minimalne presecajuće poluravni. Posmatramo L , potpurnu pravu za mnogougao \mathcal{P} kroz tačku P . Odatle, postoji H minimalna presecajuća poluravan od \mathcal{S} za koju važi da je L potpuno unutar ili potpuno izvan nje.

Ako je L u unutrašnjosti od H , tada postoji duž s' iz \mathcal{S} , takva da su mnogougao \mathcal{P} i duž s' sa raznih strana prave L , a to nije moguće, jer \mathcal{P} preseca \mathcal{S} .

Neka L leži u spoljašnjosti od H . Ako teme P nije incidentno ni sa jednom duži od \mathcal{S} , možemo da nađemo pravu unutar $H^l(L)$ takvu da je njen presek sa mnogouglaom konveksan mnogougao koji i dalje preseca \mathcal{S} i ima manji obim. Sa druge strane, ako je teme P incidentno sa nekom duži s iz \mathcal{S} , tada s neće ležati na

³U ovoj knjizi izložene su dizajnerske tehnike konstruisanja algoritama i primer koji je bio jedan od vrlo važnih je onaj u kome treba ubosti skup duži pravom. Rappaport je iskoristio ovu ideju i prilagodio je problemu koji je razmatrao.

⁴Dokaz ove leme je preuzet iz [2].

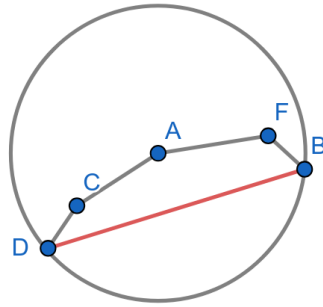
potpornoj pravoj kroz P , jer bi to bila kontradikcija sa pretpostavkom da P nije incidentno sa minimalnom presecajućom poluravni. Otuda, možemo da pomeramo granicu sve dok \mathcal{S} preseca unutrašnjost novoformiranog mnogougla, čime dobijamo mnogougao manjeg obima od \mathcal{P} što je kontradikcija.

Oдавде, sledi da je svako teme mnogougla \mathcal{P} incidentno sa granicom neke minimalne presecajuće poluravni. \square

Koristeći ovu lemu, možemo dokazati sledeće važno tvrđenje.⁵

Lema 2.1.5 *Neka važe uslovi iz prethodne leme. Tada je P incidentno sa nekom kritičnom duži.*

Dokaz. Pretpostavimo najpre da P nije incidentno ni sa jednom duži iz \mathcal{S} . To znači da možemo opisati krug sa središtem P sa dovoljno malim poluprečnikom tako da ne presečemo nijednu duž iz \mathcal{S} . Na slici 2.3 možemo videti kako zamenom mnogouglaone linije $DCAFB$ sa duži DB dobijamo mnogougao koji i dalje preseca \mathcal{S} (jer u krugu nema preseka sa \mathcal{S}), a koji pritom ima manji obim (na osnovu nejednakosti mnogougla koja će biti dokazana u daljem tekstu). Ovim smo dobili kontradikciju sa tim da je \mathcal{P} konveksni mnogougao koji preseca \mathcal{S} i ima najmanji obim.



Slika 2.3: Konstrukcija mnogougla manjeg obima koji preseca \mathcal{S}

Dakle, zaključili smo da mora postojati barem jedna duž iz \mathcal{S} koja je incidentna sa P . Neka ta duž nije kritična. Na osnovu prethodne leme, znamo da je P sigurno incidentno sa granicom minimalne presecajuće poluravni, što znači da svaka duž

⁵Dokaz ovog tvrđenja je preuzet iz [2], uz dodatna razjašnjenja i ilustraciju na slici 2.3 kao originalni doprinos ovog rada.

iz \mathcal{S} ima neprazan presek sa ovom poluravni. Samim tim i duži incidentne sa P . Kako P nije incidentno sa kritičnom duži, sve duži incidentne sa P su na onoj strani sa koje je mnogougao \mathcal{P} . Preciznije, nalaze se unutar mnogougla, jer kada bi bile vani, zbog toga što mnogougao ima neprazan presek sa svim dužima, ona koja je vani bi bila kritična duž za ekstremnu pravu koja prolazi kroz naše teme i ivicu mnogougla incidentnu sa njim. Kao i ranije, možemo konstruisati krug sa središtem P dovoljno malog poluprečnika da ispustimo druge duži iz \mathcal{S} koje nisu incidentne sa P . Slično, uzimanjem odgovarajuće tetive kao gore, dobijamo mnogougao koji i dalje ima neprazan presek sa svim dužima iz \mathcal{S} , ali manji obim, što je kontradikcija sa tim da je \mathcal{P} konveksni mnogougao koji preseca \mathcal{S} i ima najmanji obim.

Ovim smo pokazali da je svako teme mnogougla \mathcal{P} incidentno sa barem jednom kritičnom duži. \square

Ono što smo do sada zaključili je sledeće: svako teme mnogougla \mathcal{P} je incidentno sa kritičnom duži. Kako krajnje tačke kritičnih duži određuju kritičnu sekvencu $E(\mathcal{S})$, sledi da se korišćenjem ove sekvence mogu efikasno obići kritične duži, tj. duži koje sadrže temena traženog mnogougla. Kod duži kod kojih je samo jedno teme kritično, to teme predstavlja ujedno i teme mnogougla. U slučaju da duž ima dva kritična temena, tada ce ona sigurno sadržati reflektujuće teme mnogougla ukoliko je fina ivična, o čemu govori sledeća lema.⁶

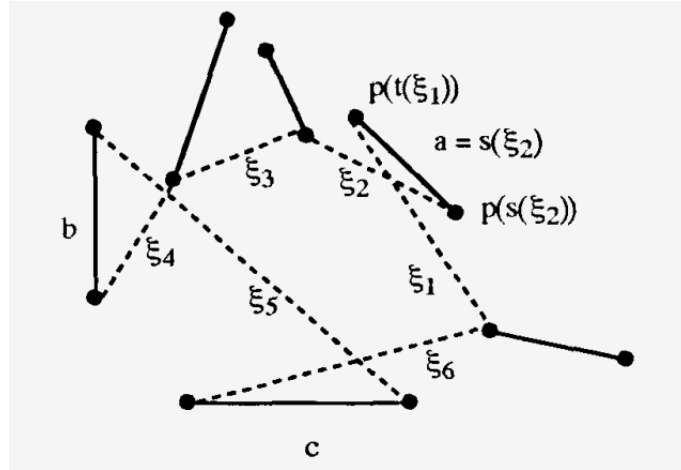
Lema 2.1.6 *Ako je duž s fina ivična, na njoj se sigurno nalazi reflektujuće teme.*

Dokaz. Iz definicije fine ivične duži, imamo da nam sve druge kritične tačke leže u $H^l(L(s))$. Ako su te kritične tačke jedine na svojim dužima, među njima su temena mnogougla. Dalje, za one kritične duži na kojima su obe tačke kritične, obe kritične tačke su u $H^l(L(s))$ te je svaka tačka sa njima incidentna sa $H^l(L(s))$, odatle sledi da su reflektujuća temena mnogougla \mathcal{P} takođe u $H^l(L(s))$. Ceo mnogougao \mathcal{P} je sa leve strane od $L(s)$ odatle sledi da je $L(s)$ potporna prava za \mathcal{P} , te je na s reflektujuće teme. \square

Napomenimo da, iako problematične ivične duži takođe imaju dva kritična temena, one mogu, ali ne moraju da sadrže reflektujuće teme mnogougla. Otuda se problematične ivične duži mogu ponašati dvojako i kao fine ivične duži i kao normalne duži, pa njihovo postojanje komplikuje sam postupak. Zato ćemo najpre

⁶Dokaz ove leme je preuzet iz [2].

pretpostaviti da nemamo problematične ivične duži u skupu duži \mathcal{S} za koji rešavamo problem. U tom slučaju, iz prethodnog razmatranja sledi da se možemo kretati od fine ivične do fine ivične duži, koristeći kritične prave iz kritične sekvence $E(\mathcal{S})$, tako da biramo na njima reflektujuća temena mnogougla i u međuvremenu uključujemo ili sečemo normalne duži koje se nalaze između njih.



Slika 2.4: Skup duži i njegova kritična sekvenca [2]

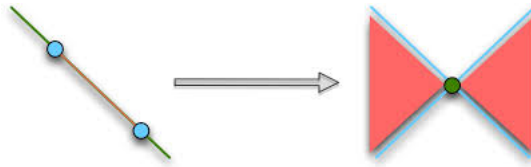
Način na koji se mogu odrediti ivične duži na osnovu kritične sekvence se može razumeti na intuitivnom nivou sa slike 2.4. Na slici 2.4 duži a , b i c su ivične. Možemo ih prepoznati po tome što su uzastopne prave iz kritične sekvence incidentne sa njima u različitim krajnjim tačkama. Ono što ostaje nejasno je kako za neku ivičnu duž odlučujemo da li je fina ivična ili problematična ivična. U radu Hasanzadeha [1] ovom pitanju nije posvećena nijedna rečenica, ali na osnovu teorije koja je izložena, može da se zaključi sledeće: trebalo bi formirati konveksni omotač kritičnih tačaka duži koje su odredile kritičnu sekvencu. Za duži koje su cele na granici konveksnog omotača važi da su fine ivične, jer su im sve kritične tačke sa iste strane. Za duži za koje ovo nije ispunjeno znamo da su problematične ivične. Duži koje imaju samo jedno kritično teme su normalne duži i one mogu sadržati nereflektujuća temena traženog mnogougla. Fine ivične duži sadrže reflektujuća temena, dok problematične ivične duži mogu sadržati bilo reflektujuće bilo nereflektujuće teme.

Sada kada znamo čemu nam služi kritična sekvenca, ostaje još da je odredimo. Dokažimo najpre sledeću lemu⁷ koja nam govori o broju kritičnih pravih.

Lema 2.1.7 *Postoji najviše $O(n)$ kritičnih ekstremnih pravih.*

⁷Dokaz ove leme je zasnovan na izlaganju datom u [2].

Dokaz. Tehnika koju ćemo u nastavku koristiti je dualna transformacija koja tačke slika u prave, i obratno.⁸ Ovakve transformacije mogu se javiti u raznim oblicima. Ovde koristimo dualnu transformaciju D datu sa $P = (\pi_1, \pi_2) \xrightarrow{D} y = \pi_1 x + \pi_2$ i $y = \pi_1 x + \pi_2 \xrightarrow{D} P = (\pi_1, \pi_2)$. Očigledno je da je D bijekcija i da ima svojstvo involutivnosti ($D^2 = I$). Duž se ovom transformacijom prevodi u dvostruki klin (slika 2.5). Slike krajnjih tačaka duži su dve prave koje se seku. Deo ovih pravih koji formira oblik " \vee " nazivamo gornjim polupravama, a deo koji formira oblik " \wedge " nazivamo donjim polupravama.



Slika 2.5: Dejstvo transformacije D na duž

Središnja tačka klina u dualnoj ravni predstavlja pravu kroz posmatranu duž s . Odredimo sada gde će biti proizvoljna prava koja preseca s . Neka su leva i desna krajnja tačka od s date sa $l(s)$ i $d(s)$, redom. Da bi prava p preseкла s ona mora ili da bude iznad $l(s)$ i ispod $d(s)$ ili ispod $l(s)$ i iznad $d(s)$.⁹ S obzirom na to da se transformacijom D čuva pripadnost i redosled, što se lako da zaključiti, slika naše prave p , $D(p)$, je ili ispod prave $D(l(s))$ i iznad prave $D(d(s))$ ili iznad prave $D(l(s))$ i ispod prave $D(d(s))$. Ako posmatramo sliku 2.5, prava p je u crvenoj oblasti.

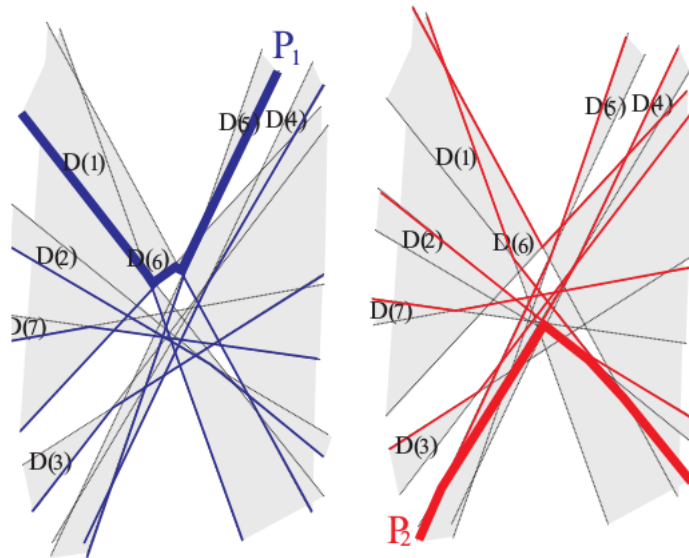
Osvrnimo se na sliku 2.6. Donju ovojnici gornjih polupravih formiramo tako što za svaku vrednost x -koordinata, odredimo tačku na nekoj od gornjih polupravih koja je niže od svih ostalih tačaka drugih gornjih polupravih sa istom x -koordinatom. Kada to uradimo za svako x , dobijamo poligonalni lanac P_1 koji je prikazan plavom bojom. Slično, gornju ovojnici donjih polupravih formiramo tako što za svaku vrednost x -koordinata, odredimo tačku na nekoj od donjih polupravih koja je više od svih ostalih tačaka drugih donjih polupravih sa istom x -koordinatom. Kada to uradimo za svako x , dobijamo poligonalni lanac P_2 koji je prikazan crvenom bojom.

Sve poluprave su delovi linearnih funkcija. Donja ovojnica gornjih polupravih je definisana formalno sa $\cup_{x \in \mathbb{R}} \min\{f_i(x) \mid f_i \text{ je gornja poluprava}\}$. Analogno definišemo i gornju ovojnici donjih polupravih $\cup_{x \in \mathbb{R}} \max\{f_i(x) \mid f_i \text{ je donja poluprava}\}$.

⁸Zapravo, ova transformacija je opštije data, slikamo tačke u hiperravni, ali pošto smo mi ovde u ravni, za nas je hiperravan prava.

⁹Tačka (p, q) je iznad (ispod) prave $y = kx + n$ akko je $kp + n < q (> q)$.

Vratimo se sada, nakon ove digresije, na kritične ekstremne prave. Zamislimo da imamo dve duži s i t i da posmatramo njihovu kritičnu ekstremnu pravu $L(s, t)$ koja je takva da joj je levo i gore isto¹⁰, s i t su joj desno, a sve ostale duži su u potpunosti u $H^l(L)$. Za svaku duž u iz \mathcal{S} koja je u $H^l(L)$ postoji prava p paralelna sa $L(s, t)$ koja je iznad $L(s, t)$ i seče u . U dualnoj ravni, $D(L(s, t))$ i $D(p)$ su tačke na istoj vertikalnoj pravoj i pritom je $D(p)$ iznad $D(L(s, t))$. Možemo da zaključimo da kritične prave koje seku \mathcal{S} u zatvorenoj donjoj poluravni, odgovaraju čvorovima na donjoj ovojnici gornjih polupravih od $D(\mathcal{S})$. Slično, kritična ekstremna prava koja preseca \mathcal{S} u zatvorenoj gornjoj poluravni odgovara čvorovima gornje ovojnice donjih polupravih od $D(\mathcal{S})$. Da bismo odredili koliko ekstremnih kritičnih pravih imamo, treba da odredimo koliko ovih čvorova imamo.

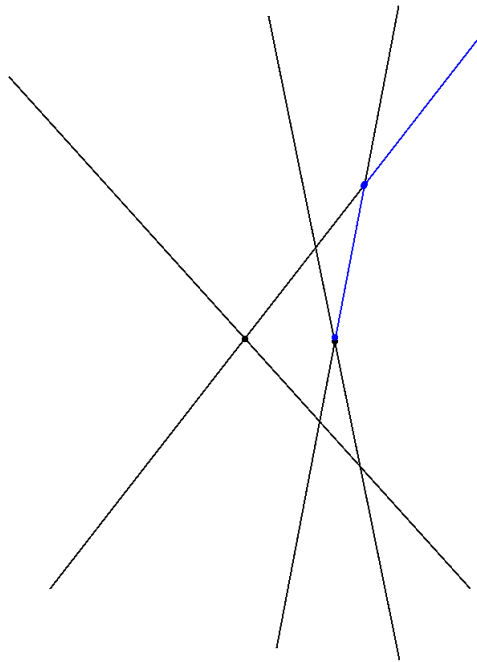


Slika 2.6: Donja ovojnica gornjih polupravih (levo) i gornja ovojnica donjih polupravih (desno) [16]

Ograničimo se na posmatranje na gornjim desnim polupravama, simetrično ćemo imati na levim. Primetimo da se donja ovojnica, isto kao i gornja sastoji od $n + 1$ uvala koje su ograničene s leve i desne strane sa centrima klinova. Za nas je uvala naziv za konveksni poligonalni lanac, čije su ivice na gornjim desnim polupravama uređene opadajuće po vrednosti nagiba. U principu, ivica na gornjoj desnoj polu-

¹⁰Tj. prava ima pozitivan koeficijent pravca

pravoj u jednoj uvali koja ima najmanji nagib je za nas poslednja ivica u toj uvali. Imamo najviše n poslednjih ivica. Pritom, najlevlja uvala ne sadrži gornje desne ivice. Takođe, primetimo da ne postoji gornja desna poluprava koja sadrži više od jedne ivice koja nije poslednja. Na slici¹¹ 2.7 imamo dva klina koji se seku. Plavom bojom označen je deo donje ovojnice gornjih polupravih koji se nalazi na gornjim desnim polupravama. Zamislimo da imamo ivicu e koja nije poslednja ivica i označimo njenu sledbenicu sa e^+ . Nije teško primetiti da je gornja desna poluprava na kojoj se e nalazi uvek iznad gornje desne poluprave gde se e^+ nalazi počevši od mesta preseka e i e^+ . Prema tome, gornja desna poluprava koja sadrži e se od mesta preseka e i e^+ nikada ne javlja kao deo ovojnice i prema tome, ne može da sadrži nijednu više ivicu. Takođe, primetimo da gornja desna poluprava koja sadrži poslednju, najdesniju ivicu ne sadrži ivicu koja nije poslednja. Odatle sledi da imamo najviše $n - 1$ ivica koje nisu poslednje na gornjim desnim polupravama donje ovojnice. Slično, simetrično radimo i za gornje leve poluprave. Ukupno imamo $4n - 2$ ivice. Kako smo zainteresovani za čvorove koji nisu centri klinova, sledi da možemo imati najviše $3n - 1$ čvorova. \square



Slika 2.7

¹¹Ova slika je originalni doprinos autora ovog rada, a dodata je u cilju dodatnog pojašnjenja dokaza.

Algoritam. U nastavku opisujemo algoritam za računanje kritične sekvence zasnovan na principu „podeli-pa-vladaj“:

- Ako je $|\mathcal{S}| \geq 3$, podelimo \mathcal{S} na dva podskupa približno istih kardinalnosti, A i B , tj. takvih da je $|A| - |B| \leq 1$
- Rekurzivno izračunamo kritičnu sekvencu za A i kritičnu sekvencu za B
- Spojimo te dve kritične sekvence $E(A)$ i $E(B)$ u jednu $E(\mathcal{S})$.

Pre nego što navedemo korake spajanja rešenja, uvedimo još malo notacije. Za kritičnu ekstremnu pravu ξ označimo sa $s(\xi)$ i $t(\xi)$ duži u \mathcal{S} takve da je ξ tangenta na obe, usmerena od $s(\xi)$ ka $t(\xi)$ i obe ove duži leže u zatvorenoj desnoj poluravni određenoj sa ξ , pri čemu ako se desi da je ξ tangenta na više od dve duži, ove dve duži su krajnje. Sa druge strane, ako su a i b dve različite duži, koristimo oznaku $L(a, b)$ da predstavimo pravu koja je tangenta na a i b , usmerena od a ka b , pri čemu a i b leže u desnoj zatvorenoj poluravni određenoj sa tom pravom. Odatle je $\xi = L(s(\xi), t(\xi))$.

Neka smo izračunali kritične sekvence skupova A i B i neka smo dobili $E(A)$ i $E(B)$. Spojimo ta dva u neki skup koji zovemo M pri čemu važi da su u M elementi sortirani po polarnom uglu.

Za proizvoljnu kritičnu ekstremnu pravu $\xi \in E(\mathcal{S})$ može da bude ispunjeno:

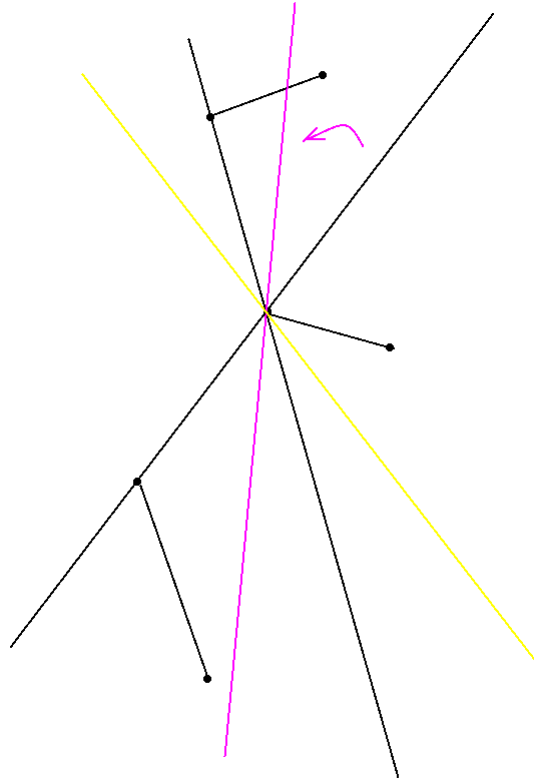
1. $\xi \in E(A)$
2. $\xi \in E(B)$
3. ξ je takvo da $s(\xi) \in A$ i $t(\xi) \in B$
4. ξ je takvo da $t(\xi) \in A$ i $s(\xi) \in B$

α će ostati kritična ekstremna prava i u $E(\mathcal{S})$ ako je α bila kritična ekstremna prava u A i ako je kritična ekstremna prava u B koja ima isti ugao $\phi(\alpha)$ sadržana u $H^l(\alpha)$. Algoritamski to proveravamo tako što za naše α nađemo dva uzastopna elementa kritične sekvence $E(B)$, zovimo ih β^- i β^+ . Označimo sa $b \in B$ duž $b = t(\beta^-) = s(\beta^+)$. Neka L_b označava usmerenu pravu sa polarnim uglom $\phi(\alpha)$ koja je tangenta na b . Ako je $H^l(\alpha)$ podskup od $H^l(L_b)$ izbrisati α iz M . Slučaj dva je simetričan.

Razmotrimo α^- i α^+ dva uzastopna elementa od $E(A)$. Neka je $a = t(\alpha^-) = s(\alpha^+)$ i neka je $b = t(\beta^-) = s(\beta^+)$, kao gore. Označimo sa $\theta(a)$ uglove u opsegu

između $\phi(\alpha^-)$ i $\phi(\alpha^+)$, a sa $\theta(\beta)$ uglove u opsegu između $\phi(\beta^-)$ i $\phi(\beta^+)$. Ako se dogodi da je $\theta(a) \cap \theta(b) \neq \emptyset$ i da za našu kritičnu ekstremnu pravu ξ iz slučaja tri (simetrično je za četiri), gde je $\xi = L(a, b)$ važi da je $\phi_{lo} \leq \phi(L(a, b)) \leq \phi_{hi}$ gde je $\phi_{lo} = \max\{\phi(\alpha^-), \phi(\beta^-)\}$ i $\phi_{hi} = \min\{\phi(\alpha^+), \phi(\beta^+)\}$ onda dodajemo $L(a, b)$ u M .

Objasnimo ovo potraživanje ugla na slici 2.8. Neka imamo dva uzastopna elementa u nekoj kritičnoj sekvenci, neka su to npr. α^- i α^+ iz $E(A)$. Pomeranjem prave kroz njihovu zajedničku duž tako da prava i dalje tangira tu duž, ako se nalazimo u opsegu između $\phi(\alpha^-)$ i $\phi(\alpha^+)$ ne menjamo veličinu probadajuće ravni, a svejedno sa levom poluravni imamo neprazan presek sa svim drugim dužima, tj. nismo ispustili nijednu duž. Ako izađemo iz datog opsega, kao što je to urađeno sa žutom pravom na slici¹² 2.8, vidimo da gubimo jednu duž, a samim tim onda ne možemo da imamo definiciju kritične prave.



Slika 2.8

Složenost ovog algoritma je $O(n \log n)$ što dobijamo iz jednačine: $T(n) = 2T(\frac{n}{2}) + O(n)$ gde je $T(n)$ broj koraka potreban da sračunamo kritičnu sekvencu za n duži.

¹²Ova slika je takođe doprinos autora ovog rada.

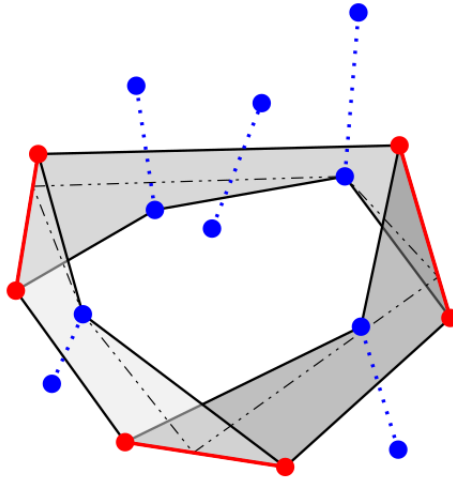
Način na koji smo računali vremensku složenost je tipičan za princip podeli-pa-zavladaj. Sabirak $O(n)$ je složenost koraka objedinjavanja rešenja problema i on je toliki jer u linearnom vremenu učešljavamo sekvencu ekstremnih kritičnih pravih, za koje smo pokazali da ih ima $O(n)$. Prostorna složenost je $O(n)$ jer hoćemo da sačuvamo kritične ekstremne prave.

2.2 Najkraći put između susednih finih ivičnih duži

Nakon što smo odredili kritičnu sekvencu $E(\mathcal{S})$, znamo na kojim dužima iz \mathcal{S} se nalaze reflektujuća temena traženog mnogougla najmanjeg obima \mathcal{P} , kao i poredak tih temena. Između svaka dva reflektujuća temena mnogougla \mathcal{P} koja odgovaraju susednim finim ivičnim dužima mogu postojati temena mnogougla \mathcal{P} koja pripadaju normalnim dužima koje se nalaze između odgovarajućih susednih finih ivičnih duži. Zbog toga je potrebno razmatrati puteve između dve susedne fine ivične duži koji presecaju normalne duži između njih.

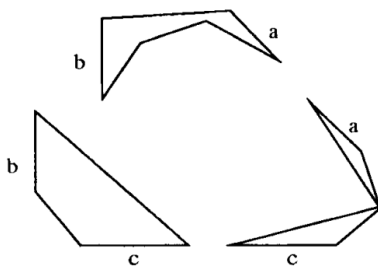
Definicija 2.2.1 *Izvodljivi put je put koji spaja dve susedne fine ivične duži i za koji važi da svaka normalna duž koja se nalazi između tih duži ima barem jednu tačku sa njegove leve strane. Izvodljivi mnogougao je prost mnogougao koji sadrži sve izvodljive puteve između dve susedne fine ivične duži.*

Pokazano je u prethodnom poglavlju da fina ivična duž mora da ima reflektujuće teme. Nakon što za svake dve fine ivične duži koje pripadaju uzastopnim pravama u kritičnoj sekvenci, konstruišemo izvodljivi mnogougao, zatvorili smo krug i taj skup mnogouglova zovemo izvodljivi ciklus (slika 2.9). Zahvaljujući notaciji koju smo ranije uveli, možemo jednostavno da opišemo konstrukciju izvodljivog mnogougla. Potrebno je da u kritičnoj sekvenci, nađemo sve parove uzastopnih finih ivičnih duži. Za uzastopne fine ivične duži $s(\xi_i)$ i $s(\xi_j)$, pravimo dva konveksna omotača. Prvi konveksni omotač gradimo nad skupom tačaka $T = \{p(t(\xi_{i-1})), p(s(\xi_i)), p(t(\xi_i)), p(s(\xi_{i+1})), \dots, p(s(\xi_{j-1})), p(t(\xi_{j-1})), p(s(\xi_j))\}$, a drugi nad skupom tačaka $T - \{p(t(\xi_{i-1})), p(s(\xi_j))\}$, gde je sa $p(s(\xi_i))$ označena tačka dodira kritične prave ξ_i i duži $s(\xi_i)$. Razlika ova dva konveksna omotača je izvodljiv mnogougao. Gornja i donja granica izvodljivog mnogougla, primetićemo, mogu i da se poklapaju na nekim mestima.

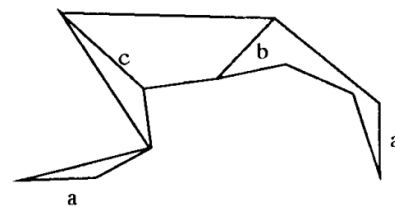


Slika 2.9: izvodljiv mnogougao sa prikazanim izvodljivim ciklusom [1]

Kakav god konveksan mnogougao unutar izvodljivog ciklusa, koji se reflektuje od finih ivičnih duži odaberemo, on je mnogougao koji zadovoljava uslov, tj. ima neprazan presek sa svim dužima iz \mathcal{S} . Ostaje samo da odaberemo konveksni mnogougao najmanjeg obima. Ispostavi se da je ovaj problem ekvivalentan traženju najkraćeg puta u prostom mnogouglu. Da bismo to uradili, tj. sveli jedan problem na drugi, moramo da „ispravimo” naš izvodljiv ciklus. Ovo ćemo uraditi tako što ga najpre posečemo po nekoj finoj ivičnoj duži, a potom ga koristeći ostale fine ivične duži, uz pomoć rotacije, translacije i refleksije razvijemo u prost mnogougao, kao što je to prikazano na slikama 2.10 i 2.11.



Slika 2.10: Primer izvodljivih mnogouglova za jedan skup duži; stranice označene istim slovom su identički jednake [2]



Slika 2.11: Transformacija u jedinstven prost mnogougao uz pomoć rotacije, refleksije i translacije [2]

Na osnovu diskusije u [2] možemo da pretpostavimo da uvek postoji ivica po kojoj možemo da posečemo izvodljiv ciklus tako da ravnanjem koje smo opisali

dobijemo prost mnogougao. Ostaje da se razreši još jedno važno pitanje: da li mnogougao mora da bude prost? S obzirom na način konstrukcije i na činjenicu da je svaki izvodljiv mnogougao nastao kao razlika dva konveksna omotača, jedino što može da se dogodi u izvodljivom mnogouglu da on ima samopresecanja jeste poklapanje gornje i donje granice (na slici 2.10 donji desni mnogougao). Međutim, kroz dalji opis algoritma, može se primetiti da to ne pravi nikakvu razliku, te možemo smatrati da radimo sa prostim mnogougloom.

Primetimo da su prva i poslednja ivica poravnatog mnogougla iste. Tačka od koje počinjemo i tačka u kojoj završavamo su takođe identične. Dakle, problem se sveo na to da pronađemo najkraći put kroz prost mnogougao od tačke na prvoj ivici do odgovarajuće tačke na poslednjoj ivici. U nastavku prikazujemo algoritam koji rešava ovaj problem. Prethodno, uvešćemo sledeću definiciju:

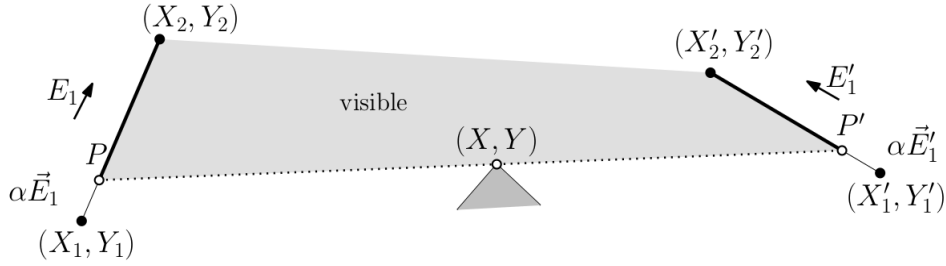
Definicija 2.2.2 *Prost mnogougao ćemo zvati neparan ako je za njegovu konstrukciju iskorišćen neparan broj izvodljivih mnogouglova. Analogno definišemo paran prost mnogougao.*

Motivacija za prethodnu definiciju leži u činjenici da ćemo na različite načine tražiti najkraći put kroz mnogougao u zavisnosti od toga da li je on paran ili neparan. Razlika je pre svega u orijentaciji prve i poslednje ivice, one koja je u stvari jedna ivica po kojoj smo rasekli ciklus. Kod neparnih su ove dve ivice suprotno orijentisane, a kod parnih imaju istu orijentaciju.

Vidljivost

Uspeli smo za sada da svedemo originalni problem na problem traženja najkraćeg puta u prostom mnogouglu, pri čemu su početna i krajnja tačka tog puta međusobno odgovarajuće tačke na dvema identičnim ivicama e_1 i e'_1 mnogougla, koje odgovaraju istoj finoj ivičnoj duži po kojoj smo rasekli ciklus. Uvedimo pojam vidljivosti u odnosu na dati mnogougao. Za dva dela prve i poslednje ivice mnogougla e_1 i e'_1 , p_1 i p'_1 kažemo da su obostrano vidljivi (nevidljivi) ako svaka tačka X sa p_1 vidi (ne vidi) svoju odgovarajuću tačku X' sa p'_1 . To znači je prava koja ih spaja unutar mnogougla. Sada je jasno da vidljivost u prostom mnogouglu mogu da naruše samo konkavna temena.

Za date duži e_1 i e'_1 i konkavno teme (X, Y) možemo da, uz pomoć malo osnovne geometrije, izračunamo koji deo ovih ivica je obostrano vidljiv.



Slika 2.12: Obostrano vidljivi delovi duži [1]

Pretpostavimo da prava koja prolazi kroz (X, Y) seče e_1 u $P(X_1, Y_1)$ i e'_1 u $P'(X'_1, Y'_1)$ (slika 2.12). Ono što želimo jeste da tačka P bude udaljena od prvog kraja duži e_1 onoliko koliko je P' udaljena od prvog kraja duži e'_1 .

Iskoristimo činjenicu da svaku tačku sa duži koja je određena tačkama (X_1, Y_1) i (X_2, Y_2) možemo na jedinstven način da predstavimo u sledećem obliku $(X_1 + \alpha \cdot (X_2 - X_1), Y_1 + \alpha \cdot (Y_2 - Y_1))$ gde je α neka konstanta između 0 i 1, uključujući i njih.

Dalje, možemo da iskoristimo činjenicu da je prava koja spaja (X, Y) i P ista kao prava koja spaja (X, Y) i P' . To što je u pitanju ista prava, na drugi način možemo izraziti kroz jednakost nagiba računatog sa jedne strane u odnosu na prve dve tačke, a sa druge strane u odnosu na druge dve tačke.

Kada sve ovo pojednostavimo, dobijamo kvadratnu jednačinu $A\alpha^2 + B\alpha + C = 0$, po parametru α čiji su koeficijenti:

$$A = (Y_2 - Y_1)(X'_2 - X'_1) - (Y'_2 - Y'_1)(X_2 - X_1)$$

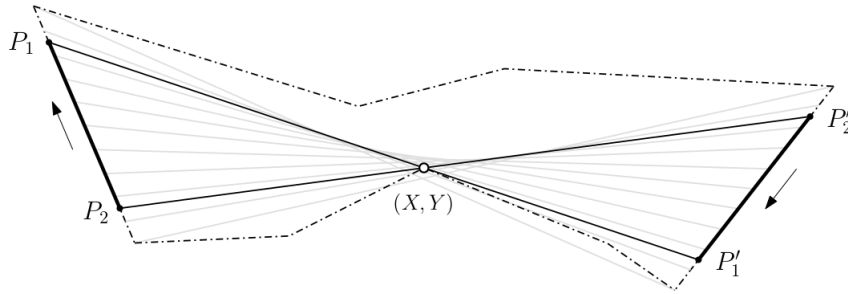
$$B = -(X'_2 - X'_1)(Y - Y_1) - (Y_2 - Y_1)(X - X'_1) + (X_2 - X_1)(Y - Y'_1) + (Y'_2 - Y'_1)(X - X_1)$$

$$C = (Y - Y_1)(X - X'_1) - (Y - Y'_1)(X - X_1)$$

Na osnovu diskriminante sistema, rešavamo ovu kvadratnu jednačinu sa restrikcijom da α mora biti iz $[0, 1]$. Ovim smo dobili efektivan način za određivanje obostrano vidljivih delova duži.

U slučaju da je u pitanju neparan mnogougao, imamo dva rešenja, dok u slučaju parnog mnogougla imamo samo jedno. Na slici 2.13 vidljiv deo nam je između P_1 i P_2 odnosno između P'_1 i P'_2 .

Ako imamo više konkavnih temena za svako pojedinačno određujemo obostrano nevidljiv deo, a krajnji nevidljivi deo je njihova unija. Imamo najviše dva nevidljiva i jedan vidljiv deo.



Slika 2.13: Slučaj neparnog mnogougla - dva rešenja za α [1]

Najkraći put u slučaju obostrano vidljivih delova

Bilo da je u pitanju neparan ili paran prost mnogougao, postoji efektivan način da odredimo najkraći put između obostrano vidljivih delova. Rastojanje između dve tačke minimizovano je u slučaju prave koja ih spaja. Ovde je olakšavajuća okolnost to što prava dolazi u obzir kao rešenje jer se s obzirom na vidljivost, nalazi unutar mnogougla. Odredimo kroz koje tačke treba provući traženu pravu.

Neka je (X, Y) tačka ivice e_1 kroz koju treba da prođe zadata prava i neka je (X', Y') druga tačka te prave, koja pripada ivici e'_1 . Tada iz pripadnosti ove dve tačke datim dužima sledi:

$$(X, Y) = (X_1 + c \cdot (X_2 - X_1), Y_1 + c \cdot (Y_2 - Y_1))$$

$$(X', Y') = (X'_1 + c' \cdot (X'_2 - X'_1), Y'_1 + c' \cdot (Y'_2 - Y'_1))$$

Na osnovu činjenice da su (X, Y) i (X', Y') odgovarajuće tačke sledi $c = c'$. Rastojanje između ovih tačaka, jednako je:

$$D = \sqrt{(X - X')^2 + (Y - Y')^2}$$

To rastojanje je funkcija jedne promenljive (po parametru C) koju treba minimizovati. Imamo standardni put u analizi koji sledimo. Tražimo izvod D po C , izjednačavamo ga sa nulom i tražimo rešenje po C . Pritom, mogu nam se dogoditi tri situacije:

- $c < 0$ - tada uzimamo $(X, Y) = (X_1, Y_1)$ i $(X', Y') = (X'_1, Y'_1)$
- $c > 1$ - tada uzimamo $(X, Y) = (X_2, Y_2)$ i $(X', Y') = (X'_2, Y'_2)$
- $0 \leq c \leq 1$ - tada uzimamo baš C koje smo dobili i za njega računamo odgovarajuće tačke

Interesantno je pomenuti da, ako računamo najkraći put između dva odgovarajuća intervala c_i na e_1 i d_i na e'_1 koji su potpuno vidljivi, najkraći put od c_i do d_i ili spaja dve krajnje tačke ili je normalan na simetralu ugla formiranog od pravih koje sadrže c_i i d_i .

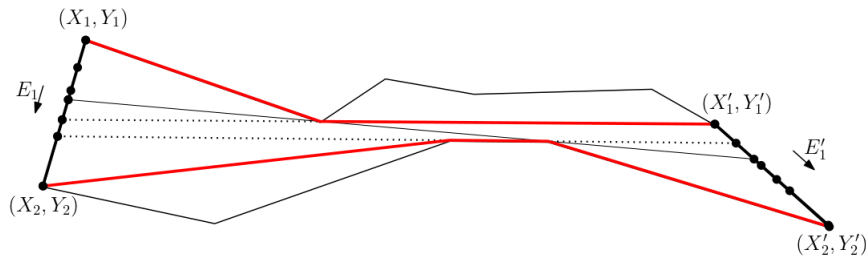
Najkraći put u slučaju obostrano nevidljivih delova

U ovom delu, nismo u mogućnosti da postupimo kao ranije. Naši putevi će se 'lomiti' u konkavnim temenima i neće biti prave. Uvešćemo pojam sidra.

Definicija 2.2.3 *Prvi čvor nakon X na najkraćem putu i poslednji čvor pre X' na najkraćem putu zovemo sidra.*

Može da se dogodi da neke tačke imaju isto sidro. Zapravo, nas će zanimati koji su to sve delovi ivica tako da sve tačke jednog dela imaju isto sidro. Ideja je da sprovedemo takvu podelu ivica e_1 i e'_1 .

Najpre, treba naći najkraće puteve između (X_1, Y_1) i (X'_1, Y'_1) sa jedne strane i (X_2, Y_2) i (X'_2, Y'_2) sa druge strane, u slučaju parnih mnogouglova (u slučaju neparnih je (X_1, Y_1) i (X'_2, Y'_2) sa jedne strane i (X_2, Y_2) i (X'_1, Y'_1) sa druge). Dalje, produžimo ivice ovih puteva, kao i tangente između njih (ako takve postoje) do preseka sa ivicama e_1 i e'_1 . Zapamtimo koje su to tačke preseka i nađemo im odgovarajuće. Podeok na ivici između svake dve takve duži je skup tačaka koje imaju isto sidro.

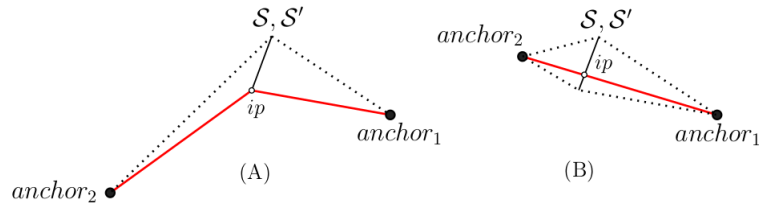


Slika 2.14: Podela E_1 na intervale čije tačke imaju isto sidra [1]

Za dati podeok s_i treba da pronađemo tačku X_i takvu da je najkraći put do njene odgovarajuće tačke X'_i na s'_i minimalne dužine. Neka je A_s sidro za s_i i neka je A'_s sidro za s'_i . Deo puta od A_s do A'_s je fiksna. Jedino na šta možemo da utičemo je izraz:

$$distance(X_i, A_s) + distance(X'_i, A'_s)$$

i njega želimo da minimizujemo. To možemo učiniti na sledeći način. Za dati podeok s_i na e_1 posmatramo njenu sliku s'_i na e'_1 i transliramo je i rotiramo dok ih ne poklopimo, a potom reflektujemo njeno sidro ukoliko je to potrebno, tako da bude na istoj strani kao sidro od s_i . Mogu da nam se dogode dva slučaja: ili duž koja spaja sidra seče s_i ili su one disjunktne, pa se sidra spajaju sa bližom krajnjom tačkom od s_i ne bi li se postigao najkraći put. Put dobijen na ovakav način može se sračunati u konstantnom vremenu [12].



Slika 2.15: Dva načina na koji sidra mogu biti raspoređena [1]

Razlikujemo levo i desno sidro. Za intervale koji su nastali podelom e_1 na način koji smo opisali ranije, definišemo levo sidro (to je prvi čvor na najkraćem putu od neke tačke posmatranog intervala do tačke (X'_1, Y'_1)) i slično desno sidro (to je prvi čvor na najkraćem putu od neke tačke posmatranog intervala do tačke (X'_2, Y'_2)). U slučaju da nam je interval koji posmatramo delimično vidljiv, a delimično nevidljiv, računamo najkraći put u vidljivom delu, pri čemu tu dodajemo poseban uslov, a to je da leva sidra budu sa jedne strane, a desna sa druge. Taj put će ući u konkurenciju sa putevima koje računamo praveći puteve sa levim, odnosno sa desnim sidrima. Nakon što za svaki podeok odredimo najkraći put minimalne dužine, među njima odaberemo najmanji i on je konačno rešenje.

2.3 Složenost algoritma u odsustvu problematičnih ivičnih duži

U ovom poglavlju razmatramo složenost opisanih postupaka u slučaju odsustva problematičnih ivičnih duži. Ono što je možda odavde ostalo nejasno je to kako nalazimo tačke koje nam dele ivicu e_1 (odnosno e'_1). Najkraći put koji spaja krajnju tačku (X_1, Y_1) ivice e_1 sa njom odgovarajućom krajnjom tačkom (X'_1, Y'_1) ivice e'_1 , je u stvari donji deo konveksnog omotača konkavnih tačaka prostog mnogougla

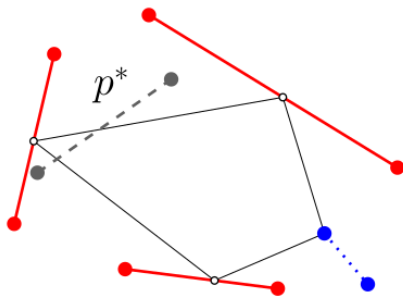
(slika 2.14), dok je najkraći put koji spaja krajnju tačku (X_2, Y_2) ivice e_1 sa njoj odgovarajućom krajnjom tačkom (X'_2, Y'_2) ivice e'_1 , je ustvari gornji deo konveksnog omotača konveksnih tačaka prostog mnogougla. Provera da li su temena mnogougla konkavna ili konveksna se vrši u vremenu $O(n)$, a formiranje konveksnog omotača u vremenu $O(n \log n)$. Povlačenje unutrašnjih tangenti između dva konveksna mnogougla može da se uradi u vremenu $O(n)$. Za svaki interval nalazimo na njemu tačku za koju dobijamo najkraći ciklus u konstantnom vremenu. Kako intervala imamo najviše n ovo možemo da uradimo u vremenu $O(n)$. Prolazak kroz niz i traženje minimuma takođe može da se uradi u vremenu $O(n)$. Način na koji su formirani delovi izvodljivog mnogougla, zahteva pronazalaženje dva konveksna omotača i traženje njihove razlike, što možemo da uradimo u vremenu $O(n \log n)$. Ispravljanje mnogougla, ali i kasnije inverzne transformacije kako bismo se vratili na početni problem, traje $O(n)$ vremena. Vremenska složenost formiranja kritične sekvence, bez koje ništa od ovoga ne bismo mogli da uradimo je takođe $O(n \log n)$. Ukupno vreme koje nam je potrebno za ovaj slučaj algoritma, kada nemamo problematične ivične duži je $O(n \log n)$.

2.4 Najteži slučaj - problematične ivične duži

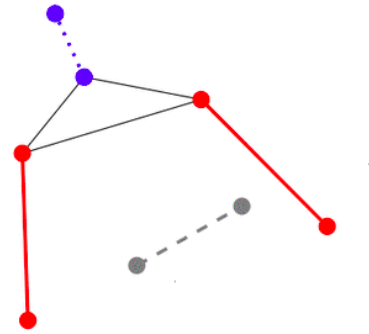
Do sada smo razmatrali situaciju u kojoj smo potpuno zanemarili postojanje problematičnih ivičnih duži. Kada njih vratimo u igru, postaće nam jasno zašto im je autor dao takav naziv. One su te koje nam donose eksponencijalnu vremensku složenost. Videćemo da za svaku problematičnu ivičnu duž imamo 4 kandidata koji mogu da je zamene, pri čemu oni nisu problematične ivične duži. Pretpostavimo da u skupu imamo k problematičnih ivičnih duži. U tom slučaju imamo 4^k različitih skupova duži za koje treba da računamo konveksni omotač najmanjeg obima. Složenost algoritma je stoga $O(4^k n \log n)$.

Najjednostavniji slučaj jeste da nam posmatrana problematična ivična duž p^* uopšte ne utiče na odabir konveksnog mnogougla najmanjeg obima. Ovo možemo da proverimo tako što ćemo da formiramo konveksni mnogougao najmanjeg obima za skup duži $\mathcal{S} - \{p^*\}$ i da proverimo da li smo možda slučajno obuhvatili i p^* . Na slici 2.16 se nalazi prikazana ta, po nas povoljna situacija, kada ne moramo da ispitujemo dalje. Međutim, ovo ne mora da bude slučaj, što je prikazano na slici 2.17.

Ono što dalje može da se dogodi jeste da se konveksni mnogougao najmanjeg

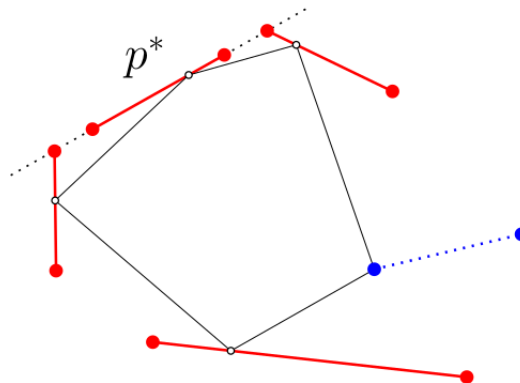


Slika 2.16: Konveksni omotač najmanjeg obima ne zavisi od p^* [1]



Slika 2.17: Konveksni omotač najmanjeg obima zavisi od p^* [1]

obima nalazi u potpunosti sa leve strane p^* . Tada možemo da p^* posmatramo kao finu ivičnu duž. To postizemo tako što provučemo pravu kroz p^* kolinearnu sa p^* i otklonimo sve one delove drugih duži koje su sa desne strane te prave. Primer za ovo imamo na slici 2.18.



Slika 2.18: Konveksni omotač najmanjeg obima za $\mathcal{S} - \{p^*\}$ je levo od p^* [1]

Moguća je još jedna situacija, da konveksni mnogougao najmanjeg obima nije u potpunosti sa leve strane p^* . U tom slučaju, on mora da prođe kroz p^* i p^* igra ulogu normalne duži. Konveksni mnogougao proći će kroz jednu od dve krajnje tačke od p^* , dakle imamo dva nova kandidata koja treba da uzmemo u obzir.

Uzimajući sve ovo u obzir, algoritam se sastoji iz sledećih koraka:

- Nađemo konveksni mnogougao najmanjeg obima za skup duži $\mathcal{S} - \{p^*\}$, gde je p^* posmatrana problematična ivična duž. Ako taj konveksni mnogougao ima neprazan presek sa p^* stajemo ovde, u suprotnom idemo na sledeći korak.

- Pretvorimo p^* u finu ivičnu duž i nađemo konveksni mnogougao najmanjeg obima za novi, izmenjeni skup duži
- Tumačimo p^* kao normalnu duž uzimajući njenu prvu krajnju tačku da bude teme konveksnog mnogougla najmanjeg obima koji preseca skup duži \mathcal{S}
- Tumačimo p^* kao normalnu duž uzimajući njenu drugu krajnju tačku da bude teme konveksnog mnogougla najmanjeg obima koji preseca skup duži \mathcal{S}

Krajnje rešenje dobijamo kada uzmemo minimum po tim pojedinačnim rešenjima. Vreme potrebno da proverimo da li duž p^* ima neprazan presek sa konveksnim mnogougлом najmanjeg obima za skup duži $\mathcal{S} - \{p^*\}$, je $O(n)$.¹³ Vreme potrebno da se odrežu delovi duži koji su sa desne strane prave koja sadrži p^* je $O(n)$. Vreme potrebno da nađemo konveksni mnogougao najmanjeg obima za skup duži koji ne sadrži u sebi problematične ivične duži je $O(n \log n)$. Ukupno, po jednoj problematičnoj ivičnoj duži za jedan korak trošimo najviše $O(n \log n)$ vremena, a za k takvih $O(4^k n \log n)$ vremena.

2.5 Implementacija

U toku rada na ovoj tezi, započet je rad na programskoj implementaciji algoritma opisanog u ovoj glavi. Koliko je autoru ovog rada poznato, ne postoji javno dostupna implementacija ovog algoritma¹⁴. Otuda je ova programska implementacija, iako još uvek ne potpuno funkcionalna, još jedan od značajnih doprinosa ovog rada.¹⁵

Implementacija se sastoji iz nekoliko modula koji su implementirani i testirani nezavisno i ostalo je da se oni objedine u funkcionalnu celinu. Programski jezik koji je izabran za realizaciju je C++. U nastavku opisujemo pojedinačne module i diskutujemo neke njihove najznačajnije karakteristike.

Modul za računanje kritične sekvence. Ispostavilo se da je ovo najteži modul za implementaciju. Težina pre svega leži u broju specijalnih slučajeva koji su

¹³Najpre, treba da proverimo da li su obe krajnje tačke unutar mnogougla. Složenost tog postupka je $O(n)$. Ako se ispostavi da duž p^* nije u pomenutom mnogouglu, onda za svaku ivicu treba da proverimo da li ona ima neprazan presek sa p^* . Vreme potrebno za to je $O(n)$.

¹⁴Rapaport je implementirao ovaj algoritam, ali implementacija nije javno dostupna.

¹⁵Implementacija se može preuzeti sa lokacije: <https://github.com/jecamark96/Minimum-perimeter-convex-polygon>.

se javljali prilikom testiranja. Osim glavne funkcije, postoji dosta pomoćnih funkcija koje se iz glavne funkcije pozivaju: funkcija za računanje kritične sekvence za dve i za tri duži (trivijalni slučajevi u algoritmu zasnovanom na principu „podeli-pa-zavладаj”), funkcija za određivanje sa koje strane prave određene dvema datim tačkama je posmatrana tačka, funkcija za računanje polarnog ugla prave, funkcija koja sortira ekstremne kritične prave na osnovu njihovog polarnog ugla, itd. Ukupan broj linija kôda u ovom modulu je 958.

Modul za određivanje finih ivičnih i problematičnih ivičnih duži. U Hasanzadehovom i Rapaportovom radu [1] nigde nije eksplicitno rečeno kako se za neku duž određuje da li je fina ivična ili problematična ivična. Na strani 9 smo opisali kako se ovo može uraditi, što je takođe jedan od originalnih doprinosa ovog rada. Do ovih zaključaka došli smo tokom implementacije ovog modula.

Modul za određivanje izvodljivog ciklusa. Izvodljivi ciklus računa se tako što se odrede svi izvodljivi mnogouglovi, prateći redosled određen kritičnom sekvencom. Svaki izvodljiv mnogougao računa se kao razlika dva konveksna omotača odgovarajućih krajnjih tačaka duži, kao što je to opisano u poglavlju 2.2.

Modul za ispravljanje izvodljivog ciklusa. Ovaj modul u sebi sadrži brojne funkcije koje služe za primenu geometrijskih transformacija: transliranje mnogougla za određeni vektor, refleksija mnogougla u odnosu na odabranu duž, rotacija mnogougla za određeni ugao. Glavna funkcija za ispravljanje izvodljivog ciklusa se svodi na uzastopnu primenu ovih transformacija. Ovaj modul ima 347 linija kôda.

Modul za određivanje vidljivih i nevidljivih delova u mnogouglu. U ovom modulu je implementiran postupak za određivanje vidljivih i nevidljivih delova opisan u odeljku 2.2. Ovaj modul ima 317 linija kôda.

Modul za računanje najkraćeg puta u izvodljivom mnogouglu. U poglavlju 2.3 izloženi su pojedini detalji koji u Hasanzadehovom i Rapaportovom radu [1] nisu dati eksplicitno. Do njih smo došli u toku implementacije ovog modula. Takođe, ono što nije bilo jasno precizirano je činjenica da za svaki podeok na ivici e_1 , ukoliko je on delom na vidljivom, a delom na nevidljivom delu, postoje i levo i desno sidro. Iz originalnog pristupa u Hasanzadehovom i Rapaportovom radu [1] moglo se zaključiti, bez napomene autora rada [12] odakle je pristup pozajmljen, da

postoji samo jedno sidro za svaki podeok na e_1 , što nije tačno. Ukupan broj linija kôda u ovom modulu je 934.

Modul za računanje konačnog rešenja. Ovaj modul trenutno nedostaje u implementaciji. U planu je da se on sastoji iz jedne rekurzivne funkcije. Ova funkcija bi kao argument prihvatala skup duži, a kao rezultat bi vraćala konveksni mnogougao najmanjeg obima koji ima neprazan presek sa svakom duži iz tog skupa. Pre poziva ove funkcije, najpre treba pozvati modul za računanje kritične sekvence, a potom modul za određivanje finih ivičnih i problematičnih ivičnih duži. U samoj funkciji bismo izabrali proizvoljnu problematičnu ivičnu duž $p^* \in \mathcal{S}$ i rekurzivno bismo pozvali tu istu funkciju za računanje konačnog rešenja za skup $\mathcal{S} \setminus \{p^*\}$. Potom bi sledila provera da li je dobijenim rešenjem obuhvaćeno i p^* . Ako jeste, ovde stajemo i dobijeni rezultat vraćamo kao konačno rešenje. U suprotnom, rekurzivno računamo rešenje u slučajevima kada je p^* zamenjeno nekom od svojih krajnjih tačaka, kao i kada se p^* posmatra kao fina ivična duž. Od ova tri rešenja biramo kao konačno ono koje ima najmanji obim. Detaljan opis ovog postupka dat je u poglavlju 2.4.

Glava 3

Aproksimativni algoritmi

U ovoj glavi izložemo nekoliko značajnih aproksimativnih algoritama za rešavanje problema kojim se bavimo u ovom radu. Aproksimativnost se ogleda u tome što se kao rezultat neće dobiti konveksni mnogougao najmanjeg mogućeg obima koji preseca sve duži datog skupa, već će obim biti nešto veći (uz garantovanje odgovarajuće gornje granice).

Značaj aproksimativnih algoritama je u tome što je njihova složenost značajno manja od složenosti egzaktnih algoritama, a sa druge strane, obim dobijenog mnogougla ne odstupa značajno od minimalnog obima. Otuda ovi algoritmi u praksi mogu biti primenljiviji od egzaktnih.

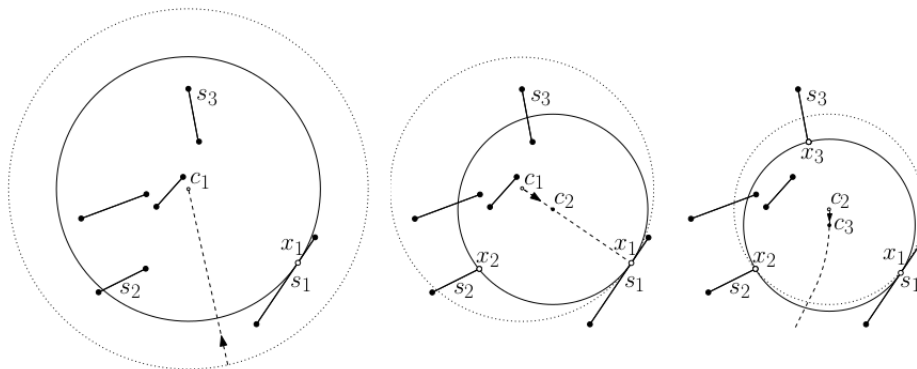
3.1 Algoritam zasnovan na minimalnom razapinjućem krugu

U ovom poglavlju prikazaćemo aproksimativni algoritam zasnovan na minimalnom razapinjućem krugu. Uvedimo najpre sledeću definiciju.

Definicija 3.1.1 *Minimalni razapinjući krug za skup duži \mathcal{S} je krug najmanjeg obima koji sadrži barem jednu tačku svake duži iz \mathcal{S} u svojoj unutrašnjosti ili na svojoj kružnoj liniji.*

Konstrukcija započinje tako što nađemo proizvoljan krug koji sadrži sve duži iz \mathcal{S} i označimo središte tog kruga sa C_1 . Dalje, možemo da smanjimo poluprečnik tog kruga, sve dok ne nađemo na duž koja sa krugom ima tačno jednu presečnu tačku i koju ćemo ispustiti ako nastavimo da smanjujemo poluprečnik. Označimo tu duž

sa s_1 , a tačku koja joj je zajednička sa našim krugom sa X_1 . Možemo pratiti šta se dešava na slici 3.1.



Slika 3.1: Konstrukcija minimalnog razapinjućeg kruga [1]

U narednom koraku, treba da pomeramo središte C_1 ka X_1 na pravouj koja ih spaja (tom prilikom poluprečnik nam je C_1X_1), sve dok ne dođemo u situaciju da smo dobili novu duž s_2 koja ima tačno jednu zajedničku tačku X_2 sa našim krugom i za koju važi da je ispuštamo ako nastavimo i dalje da radimo ovo pomeranje. Novo središte označavamo sa C_2 .

Ipak, sve ovo nije bilo dovoljno da nam odredi krug. Treba nam geometrijsko mesto tačaka koje su podjednako udaljene od s_1 i od s_2 . To geometrijsko mesto je kombinacija pravih i parabola.¹ Nastavimo da pomeramo C_2 duž ovog geometrijskog mesta tačaka u smeru s_1 i s_2 (napomenimo da tom prilikom mogu da nam se pomeraju i X_1 i X_2). Mogu da nam se dogode dve situacije. Jedna je da nam X_1X_2 postane prečnik kruga i time je krug potpuno određen. Druga je da nam ponovo zapreti opasnost da ispustimo neku duž, tj. da nam se pojavi duž s_3 koja ima tačno jednu zajedničku tačku sa našim krugom. U tom slučaju krug je potpuno određen sa tri tačke: X_1 , X_2 i tačkom koja mu je zajednička sa s_3 .

¹Kako u radu [1] nije preciznije opisano ovo geometrijsko mesto, ovde dajemo pojašnjenje kao originalni doprinos autora ovog rada. Parabola je geometrijsko mesto tačaka podjednako udaljenih od fiksirane tačke (fokusa) i fiksirane prave (direktrise). Mi tražimo geometrijsko mesto tačaka podjednako udaljenih od fiksirane tačke i fiksirane duži. U slučaju da se iz tačke može povući normala na pravu koja sadrži posmatranu duž tako da ona padne na tu duž, geometrijsko mesto je onaj deo parabole čiji je fokus ta tačka, a direktrisa prava na kojoj leži posmatrana duž, koji se nalazi između normala na duž povučeneh u njenim krajnjim tačkama. U suprotnom kao rezultat možemo dobiti pravu. Za svaku tačku prve duži i drugu duž tražimo posebno geometrijsko mesto tačaka podjednako udaljenih od posmatrane tačke i posmatrane duži. Unija ovih geometrijskih mesta je krajnje rešenje.

Prethodni postupak, iako jednostavan za razumevanje, nije pogodan za efikasnu implementaciju. Zbog toga u nastavku prikazujemo algoritam zasnovan na Voronojevim dijagramima najudaljenijih duži.

Definicija 3.1.2 *Voronojev dijagram najudaljenijih duži za skup duži $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ je podela ravni na oblasti R_1, R_2, \dots, R_n sa disjunktним unutrašnjostima, takva da se oblast R_i sastoji od tačaka ravni za koje je rastojanje od duži s_i veće od rastojanja od svih drugih duži skupa \mathcal{S} .*

U slučaju da nam je krug određen sa tri tačke postupamo na sledeći način. Neka je V_{ijk} čvor Voronojevog dijagrama najudaljenijih duži. Znamo da je to tačka koja je udaljena podjednako od duži s_i, s_j i s_k . Neka je to rastojanje d . Za bilo koju drugu duž iz skupa \mathcal{S} nad kojim je formiran Voronojev dijagram važi da je najkraće rastojanje V_{ijk} do te duži manje od d . Krug sa središtem V_{ijk} i poluprečnikom d je kandidat za minimalni razapinjući krug. Čvorova u Voronojevom dijagramu najudaljenijih duži imamo $O(n)$. Vreme potrebno da nađemo minimalni razapinjući krug sa najmanjim obimom među kandidatima je $O(n)$. Vreme potrebno da formiramo sam Voronojev dijagram je $O(n \log n)$, tako da je ukupno vreme ovog slučaja $O(n \log n)$.

Razmatrajmo sada krugove određene sa dve tačke. Srećom, možemo da razmatramo i ivice Voronojevog dijagrama najudaljenijih duži. Za datu ivicu e_{ij} važi da je proizvoljna tačka na njoj podjednako udaljena od duži s_i i s_j i bliža svim drugim dužima nego što je njima. Spojimo najkraćom mogućom duži s_i i s_j i razmatramo presek te duži sa e_{ij} . Ako presek ne postoji, nemamo kandidata. U suprotnom, kandidat je krug čije je središte presečna tačka i čiji je prečnik jednak dužini najkraće moguće duži između s_i i s_j . Ivica imamo $O(n)$, tako da u linearnom vremenu možemo da pronađemo krug sa najmanjim obimom.

Našli smo, za dati skup duži \mathcal{S} , razapinjući krug sa minimalnim obimom određen sa tri tačke i razapinjući krug sa minimalnim obimom određen sa dve tačke. Od ta dva, odaberemo onaj koji ima manji obim i on je konačno rešenje. Sada, kada znamo kako da to uradimo i znamo da nam za to treba $O(n \log n)$ vremena, ostaje nam da vidimo koliko je naša aproksimacija dobra.

Lema 3.1.3 *Za bilo koji konveksni mnogougao sa ivicama $E = \{v_1v_2, v_2v_3, \dots, v_nv_1\}$, $\forall e_i \in E, \text{len}(e_i) < \sum \text{len}(e_j), j \neq i$.²*

²Dokaz ove leme je preuzet iz [1].

Dokaz. Dokaz izvodimo indukcijom po broju temena. Ako imamo tri temena, znamo da važi nejednakost trougla ³. Baza indukcije je u redu.

Pretpostavimo da tvrđenje važi za n temena i dokažimo da važi i za $n + 1$ teme. Pošto je mnogougao konveksni, jedino što smo mogli da uradimo kad smo ubacili novo teme jeste da dodamo dve ivice v_nv_{n+1}, v_nv_1 . Iskoristimo dva puta pomenutu nejednakost trougla i imamo:

$$v_nv_{n+1} < v_nv_1 + v_1v_{n+1} < v_1v_{n+1} + \sum_{i=1}^{n-1} v_iv_{i+1}$$

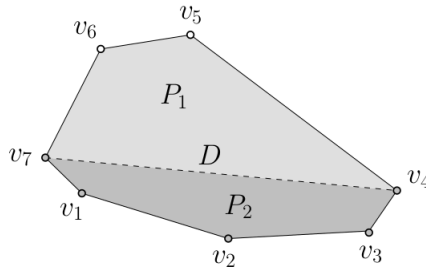
$$v_1v_{n+1} < v_1v_n + v_nv_{n+1} < v_nv_{n+1} + \sum_{i=1}^{n-1} v_iv_{i+1}$$

Odakle sledi da lema važi i kada imamo $n + 1$ temena. □

Dijametar $D(P)$ jednak je $\max\{d(a,b) \mid a,b \in P\}$, gde je $d(a,b)$ rastojanje između dve tačke skupa a i b . Važi sledeća lema:⁴

Lema 3.1.4 *Za bilo koji konveksni mnogougao \mathcal{P} sa dijametrom D , $2D < \text{obim}(\mathcal{P})$.*

Dokaz. Dijametar D deli \mathcal{P} na dva manja konveksna mnogougla. Naravno, može da se desi i da je dijametar baš ivica od \mathcal{P} , ali to ne remeti ovaj pristup.



Slika 3.2: Konveksni mnogougao \mathcal{P} sa dijametrom D [1]

U skladu sa nejednakošću mnogougla koju smo dokazali u prethodnoj lemi i na osnovu slike 3.2 imamo:

$$D < \text{obim}(\mathcal{P}_1) - D \Rightarrow 2D < \text{obim}(\mathcal{P}_1)$$

$$\text{obim}(\mathcal{P}_1) \leq \text{obim}(\mathcal{P})$$

³Radi potpunosti trebalo bi ubaciti ovde taj dokaz, ali mi se ovde nećemo baviti time.

⁴Dokaz ove leme je preuzet iz [1].

$$\Rightarrow 2D < \text{obim}(\mathcal{P})$$

□

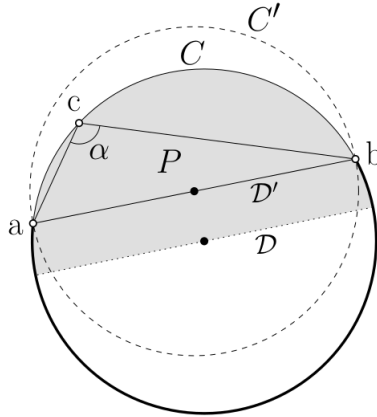
Definicija 3.1.5 *Minimalni razapinjući krug za dati konveksni mnogougao je krug najmanjeg obima za koji važi da su sva temena zadanog konveksnog mnogougla u unutrašnjosti ili na kružnoj liniji tog kruga.*

Može se pokazati da minimalni razapinjući krug za konveksni mnogougao mora da sadrži na kružnoj liniji barem dva temena tog mnogougla. Naime, ako ne bi sadržao nijedno teme, mi bismo mogli da smanjujemo obim tog kruga dok ne dođemo to toga da imamo jedno teme na kružnoj liniji, ali to je kontradikcija sa time da je naš krug minimalni razapinjući krug. Dalje, mogli bismo da pomeramo središte tog novodobijenog kruga ka jedinoj tački na kružnoj liniji, po pravoj koja ih spaja, dok ne dođemo do toga da imamo još jednu tačku na kružnoj liniji, čime ponovo dolazimo do kontradikcije. Dakle, došli smo do zaključka da moramo da imamo barem dve tačke na kružnoj liniji. Ako su one takve da je duž koja ih spaja prečnik kruga, tu stajemo, u suprotnom možemo još da suzimo krug koji smo dobili i time dobijemo i treću tačku na kružnoj liniji. Važi sledeća lema:⁵

Lema 3.1.6 *Središte minimalnog razapinjućeg kruga konveksnog mnogougla \mathcal{P} , nalazi se unutar \mathcal{P} .*

Dokaz. Pretpostavimo suprotno, neka je središte minimalnog razapinjućeg kruga \mathcal{C} van konveksnog mnogougla \mathcal{P} . Na osnovu prethodnog razmatranja, znamo da bar dva njegova temena moraju da leže na kružnoj liniji kruga \mathcal{C} . Pretpostavimo najpre da je \mathcal{P} trougao (primer je dat na slici 3.3). S obzirom na našu pretpostavku da \mathcal{P} ne uključuje središte, možemo da nacrtamo prečnik kruga za koji važi da je \mathcal{P} u celini sa jedne njegove strane. Pretpostavimo najpre da je teme C na kružnici. U tom slučaju znamo da je ugao $\alpha > \pi/2$, jer je luk koji mu odgovara veći od π . Ako nacrtamo novi krug \mathcal{C}' čiji će prečnik da bude \overline{AB} , dolazimo do zaključka da teme C trougla ABC mora da bude unutar \mathcal{C}' jer mu je odgovarajući ugao veći od $\pi/2$, što znači da \mathcal{C}' jeste razapinjući krug. Pritom, prečnik od \mathcal{C}' je manji od prečnika \mathcal{C} što je kontradikcija sa činjenicom da je \mathcal{C} minimalni razapinjući krug.

⁵Dokaz ove leme je preuzet iz [1], uz preciziranja na pojedinim mestima. Konkretno, detaljnije je razjašnjen slučaj kada mnogougao \mathcal{P} nije trougao.



Slika 3.3: Razapinjući krugovi trougla [1]

U slučaju da je teme C u unutrašnjosti kruga \mathcal{C} , situacija je još lakša, jer bi u tom slučaju teme C svakako bilo i u unutrašnjosti kruga \mathcal{C}' konstruisanog na prethodno opisan način.

Ako mnogougao \mathcal{P} ima više od tri temena, za temena koja su u unutrašnjosti kruga \mathcal{C} možemo da primenu prethodnu konstataciju. Za ona temena koja su na kružnici, na svako pojedinačno, bez posmatranja drugih temena, možemo da upotrebimo dokaz koji smo maločas izveli, jer je svaki od uglova koji formiraju A, B i proizvoljno teme tup. Uvek formiramo isti krug \mathcal{C}' sa središtem u središtu duži AB i poluprečnikom $\frac{AB}{2}$ i dokazujemo da su sva temena mnogougla \mathcal{P} u unutrašnjosti ovog kruga.

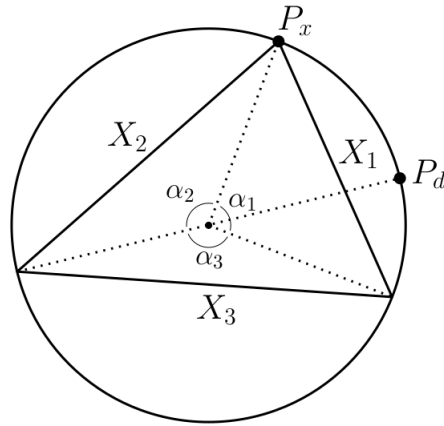
□

Teorema 3.1.7 *Neka je \mathcal{P}^* konveksni mnogougao najmanjeg obima koji seče sve duži iz datog skupa duži \mathcal{S} i neka je \mathcal{C}^* , krug prečnika D , minimalni razapinjući krug od \mathcal{S} . Tada važi $2D < \text{obim}(\mathcal{P}^*) < \pi D$.⁶*

Dokaz. Krug \mathcal{C}^* je konveksan oblik (između ostalog lako bismo mogli da nađemo konveksan mnogougao koji ga proizvoljno dobro aproksimira), pa $\text{obim}(\mathcal{P}^*)$ ne sme da bude veći od $\text{obim}(\mathcal{C}^*)$, jer ako bi to bilo ispunjeno, to bi protivrećilo činjenici da je \mathcal{P}^* rešenje našeg problema. Kako je D prečnik kruga \mathcal{C}^* , $\text{obim}(\mathcal{C}^*) = \pi D$, što nam sve zajedno daje $\text{obim}(\mathcal{P}^*) < \pi D$.

⁶Dokaz ove teoreme je u najvećoj meri preuzet iz [1], uz neka preciziranja na pojedinim mestima. Konkretno, upotreba sinusne teoreme za dokazivanje nejednakosti je originalni doprinos autora ovog rada.

Neka je $\mathcal{C}_{\mathcal{P}^*}$ sa prečnikom \mathbb{D} minimalni razapinjući krug za konveksni mnogougao \mathcal{P}^* . Ako su tačno dva temena od \mathcal{P}^* , A i B locirana na kružnici od $\mathcal{C}_{\mathcal{P}^*}$ tada je, na osnovu ranije opisane konstrukcije minimalnog razapinjućeg kruga za mnogougao, $\overline{AB} = \mathbb{D}$ prečnik kruga, samim tim i dijametar kruga, a time i dijametar od \mathcal{P}^* . Na osnovu leme 3.1.4 sledi da je $2\mathbb{D} < \text{obim}(\mathcal{P}^*)$. U suprotnom, ako su barem tri temena od \mathcal{P}^* na kružnici od $\mathcal{C}_{\mathcal{P}^*}$ na osnovu leme 3.1.6 sledi da je središte kruga $\mathcal{C}_{\mathcal{P}^*}$ u unutrašnjosti mnogougla \mathcal{P}^* i u unutrašnjosti trougla ABC .



Slika 3.4: Minimalni razapinjući krug trougla [1]

Na slici 3.4, neka je $0 < \alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \pi$. Ovi uglovi ne mogu da budu veći od π jer trougao ABC mora da uključuje središte svog minimalnog razapinjućeg kruga. Uglovi kod odgovarajućih temena su polovine ovih uglova (npr. ugao kod P_x je periferni ugao, koji odgovara centralnom uglu α_3 i iznosi $\alpha_3/2$). Ako upotrebimo sinusnu teoremu imaćemo da je $\frac{x_1}{\sin(\alpha_1/2)} = \frac{x_2}{\sin(\alpha_2/2)} = \frac{x_3}{\sin(\alpha_3/2)} = \mathbb{D}$, odatle imamo:

$$x_1 = \mathbb{D} \sin(\alpha_1/2) \qquad x_2 = \mathbb{D} \sin(\alpha_2/2)$$

Pomeranjem tačke P_x u smeru tačke P_d na kružnici, smanjujemo α_1 i x_1 i povećavamo α_2 i x_2 . Primitimo da se α_3 i x_3 ne menjaju. Ono što ostaje da se pokaže jeste da x_1 opada brže nego što x_2 raste, odnosno da obim stalno opada. Ako pogledamo izvode x_1 po α_1 i x_2 po α_2 , na osnovu njih ćemo moći da ocenimo rast i opadanje promenljivih x_1 i x_2 . Kako je $\alpha_1 + \alpha_2$ fiksno, povećavanje α_1 za jednu jedinicu mere, značiće smanjivanje α_2 za istu tu jedinicu mere.

$$\frac{\partial x_1}{\partial \alpha_1} = \frac{\mathbb{D}}{2} \cos\left(\frac{\alpha_1}{2}\right) \qquad \frac{\partial x_2}{\partial \alpha_2} = \frac{\mathbb{D}}{2} \cos\left(\frac{\alpha_2}{2}\right)$$

Kako važi $0 < \alpha_1 \leq \alpha_2 \leq \pi$ odatle sledi da je i $0 < \frac{\alpha_1}{2} \leq \frac{\alpha_2}{2} \leq \pi/2$. Kosinus je u prvom kvadrantu opadajuća funkcija te sledi da je: $\cos(\frac{\alpha_1}{2}) > \cos(\frac{\alpha_2}{2})$. Preostaje nam samo da ovu nejednakost pomnožimo sa \mathbb{D} i dobijamo da je opadanje od x_1 brže nego rast od x_2 .

Možemo da smanjujemo x_1 sve dok nam x_2 i x_3 ne postanu jednaki \mathbb{D} . U tom trenutku, obim nam je $2\mathbb{D}$. Obim originalnog trougla je sigurno veći od ovog, a takođe i obim \mathcal{P}^* . Otuda je $2\mathbb{D} < \text{obim}(\mathcal{P}^*)$.

Krug $\mathcal{C}_{\mathcal{P}^*}$ je razapinjući krug i za \mathcal{S} . Kako je krug \mathcal{C}^* minimalni razapinjući krug za \mathcal{S} sledi da je $D < \mathbb{D}$. Takođe, znamo da je $2\mathbb{D} < \text{obim}(\mathcal{P}^*)$, odatle sledi da je $2D < \text{obim}(\mathcal{P}^*)$.

Kada spojimo dve ključne nejednakosti koje smo izveli, imamo da je:

$$2D < \text{obim}(\mathcal{P}^*) < \pi D$$

□

Sumirajmo šta smo do sada uradili. Dakle, uspeali smo da u vremenu $O(n \log n)$ konstruišemo minimalni razapinjući krug za skup duži \mathcal{S} i da pokažemo da je njegov dijametar pomnožen konstantom iz intervala $(2, \pi)$ zapravo obim konveksnog mnogougla najmanjeg obima \mathcal{P}^* koji preseca skup duži \mathcal{S} . Sada nam ostaje da iz ovog minimalnog razapinjućeg kruga izvučemo konveksni omotač koji će biti naša aproksimacija rešenja datog problema.

Na osnovu definicije minimalnog razapinjućeg kruga za dati skup duži \mathcal{S} , znamo da za svaku duž iz tog skupa važi da je barem jedna njena tačka u unutrašnjosti ili na kružnoj liniji kruga koji smo ranije već označili sa \mathcal{C}^* . U skup koji ćemo nazvati \mathcal{S}' izdvojimo sa svake duži iz \mathcal{S} tačku koja je najbliža središtu kruga \mathcal{C}^* . I zaista, tačke iz \mathcal{S}' ne mogu biti na rastojanju od središta kruga \mathcal{C}^* većem od $D/2$.

Kada smo ovo uradili, u vremenu $O(n \log n)$ izračunamo konveksni omotač za tačke iz \mathcal{S}' . Nazovimo ga \mathcal{CH}' . Kako su sve tačke iz \mathcal{S}' uključene u \mathcal{C}^* i \mathcal{CH}' je uključen u \mathcal{C}^* . Jedno konveksno telo u ravni koje leži unutar drugog planarnog konveksnog tela ima manji obim od tog drugog konveksnog tela.

$$\text{obim}(\mathcal{CH}') < \text{obim}(\mathcal{C}^*) \Rightarrow \text{obim}(\mathcal{CH}') < \pi D$$

Neka je \mathcal{P}^* konveksni omotač najmanjeg obima za \mathcal{S} . Obim od \mathcal{CH}' ne može da bude manji od obima \mathcal{P}^* :

$$\text{obim}(\mathcal{P}^*) \leq \text{obim}(\mathcal{CH}')$$

Na osnovu granica koje smo ranije ustanovili:

$$2D < \text{obim}(\mathcal{P}^*) \leq \text{obim}(\mathcal{CH}') < \pi D$$

\mathcal{CH}' je traženo aproksimativno rešenje. Sve što je do sada urađeno, može da se objedini u sledeću teoremu:

Teorema 3.1.8 *Neka je \mathcal{P}^* konveksni mnogougao najmanjeg obima koji preseca skup duži \mathcal{S} . Tada u vremenu $O(n \log n)$ možemo odrediti konveksni mnogougao \mathcal{P} koji preseca \mathcal{S} i zadovoljava nejednakost:*

$$\text{obim}(\mathcal{P}) \leq \frac{\pi}{2} \cdot \text{obim}(\mathcal{P}^*)$$

3.2 Unapređeni aproksimativni algoritmi

Nakon Hasanzadeha i Rapaporta, pojavljuje se rad Dumitreskua i Jianga [3]. Oni u skladu sa preporukama svojih prethodnika koriste linearno programiranje. Naravno, za to im treba određeni vid diskretizacije problema, ali oni uspeavaju da procene kolika je greška time učinjena. Rad je objavljen 2011. godine i u njemu su predložena dva algoritma.

Prvi algoritam

U ovom odeljku opisujemo prvi od aproksimativnih algoritama dat u radu Dumitreskua i Jianga [3]. Najpre navedimo sledeću teoremu:

Teorema 3.2.1 *Za bilo koje $\varepsilon > 0$, $\frac{4}{\pi}(1 + \varepsilon)$ -aproksimativni algoritam za nalaženje presecajućeg mnogougla najmanjeg obima za dati skup od n duži u ravni, može biti izračunat rešavanjem $O(1/\varepsilon)$ linearnih programa, svaki sa $O(n)$ promenljivih i $O(n)$ ograničenja. Konkretno, 1.28-aproksimacija može da se izračuna rešavanjem konstantnog broja ovakvih linearnih programa.*

Teorema 3.2.1 je jedna od dve najbitnije teoreme rada [3]. Ovaj odeljak posvetićemo njenom razjašnjavanju i dokazu, po ugledu na [3]. Opišimo najpre sam algoritam. Dakle, ideja je da uzmemo nekoliko pravaca, tačnije njih $m = \lceil \frac{\pi}{4\varepsilon} \rceil$ i da za svaki pravac $\alpha_i = i \cdot 2\varepsilon, i = 0, 1, \dots, m - 1$ izračunamo presecajući pravougaonik \mathcal{R}_i najmanjeg obima takav da mu jedna ivica ima pravac α_i ($0 \leq \alpha_i \leq \pi/2$) dok je druga normalna na ovaj pravac. Tih $m = O(1/\varepsilon)$ minimalnih pravougaonika su

rešenja isto toliko linearnih programa. Za aproksimaciju uzimamo onaj koji ima najmanji obim.

U svakoj iteraciji, možemo da pretpostavimo da su ivice pravougaonika koji tražimo \mathcal{R}_i paralelne koordinatnim osama. Ako to nije slučaj, na osnovu ugla α_i možemo celu ravan zarotirati tako da ivice pravougaonika \mathcal{R}_i budu paralelne osama. Svaki pravougaonik je određen gornjom levom i donjom desnom tačkom. Pišemo $\mathcal{R}_i = [x_1, x_2] \times [y_1, y_2]$. To su nam četiri promenljive koje treba da otkrijemo. Neka je $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ naš skup duži koje posmatramo. Neka su krajnje tačke duži s_i , $U_i = (a_i, b_i)$ i $V_i = (c_i, d_i)$. Već smo koristili način da zapišemo geometrijski uslov da duž pripada tački. Za svaku duž iz našeg skupa duži mora da važi da je barem jedna njena tačka unutra ili na granici \mathcal{R}_i da bi on bio presecajući pravougaonik. Sada možemo ove nabrojane uslove da zapišemo na sledeći način:

$$\begin{aligned} &\text{minimizovati} && 2(x_2 - x_1) + 2(y_2 - y_1) \\ &\text{pri uslovima} && \begin{cases} x_1 \leq (1 - t_j)a_j + t_jc_j \leq x_2, & 1 \leq j \leq n, \\ y_1 \leq (1 - t_j)b_j + t_jd_j \leq y_2, & 1 \leq j \leq n, \\ 0 \leq t_j \leq 1, & 1 \leq j \leq n \end{cases} \end{aligned}$$

Za svaki pravac dobijamo minimalni presecajući pravougaonik čija se jedna ivica poklapa sa tim pravcem. Među ovakvim pravougaonicima, koji su rešenje linearnih programa, kao aproksimaciju krajnjeg rešenja našeg polaznog problema, uzimamo onaj koji ima najmanji obim.

U nastavku ovog odeljka dokazujemo nejednakosti koje će nam uspostaviti gornju granicu na ovaj način dobijene aproksimacije. Najpre navodimo sledeću definiciju:

Definicija 3.2.2 *Konveksno telo je kompaktni⁷ konveksni skup sa nepraznom unutrašnjosti.*

Lema 3.2.3 *Neka je \mathcal{R} pravougaonik najmanjeg obima koji sadrži konveksno telo \mathcal{B} . Tada važi:*

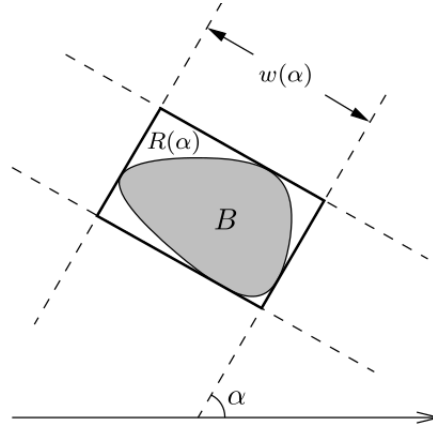
$$\text{obim}(\mathcal{R}) \leq \frac{4}{\pi} \text{obim}(\mathcal{B})$$

Dokaz. Izvodimo dokaz⁸ prateći sliku 3.5.

Za svaku orijentaciju α nademo posebno pravougaonik najmanjeg obima koji sadrži \mathcal{B} i ima tu orijentaciju. Označimo ga sa $\mathcal{R}(\alpha)$. Obim ovakvih pravougaonika

⁷Misli se na pojam kompaktnosti u matematičkoj analizi

⁸Dokaz ove leme je preuzet iz [3].



Slika 3.5: Pravougaonik $\mathcal{R}(\alpha)$ je presek dve trake širina $w(\alpha)$ i $w(\alpha + \pi/2)$ duž pravaca α i $\alpha + \pi/2$, redom [3]

je funkcija od α . Treba nam neprekidnost ove funkcije, da bi mogli da primenimo integral. Ona je kompozicija neprekidnih delova, imamo funkciju minimuma, potom zbirve normi, itd. Ako bismo želeli da ovo uradimo matematički precizno, trebalo bi to dokazati. Ovde pak samo to navodimo kao činjenicu. Ono što je dalje korišćeno iz aparata matematičke analize je integralna nejednakost, linearnost integrala i činjenica da je obim pozitivna funkcija.

Kako je od svih presecajućih pravougaonika, \mathcal{R} onaj sa najmanjim obimom, za bilo koje $\alpha \in [0, \pi/2)$ važi: $obim(\mathcal{R}) \leq obim(\mathcal{R}(\alpha))$, te odatle imamo:

$$\frac{\pi}{2} \cdot obim(\mathcal{R}) \leq \int_0^{\pi/2} obim(\mathcal{R}(\alpha)) d\alpha$$

Neka je $w(\alpha)$ minimalna širina paralelne trake duž pravca α da bi \mathcal{B} bilo sadržano u odgovarajućem pravougaoniku, određenom tom širinom. Primetimo da važi:

$$obim(\mathcal{R}(\alpha)) = 2w(\alpha) + 2w(\alpha + \pi/2)$$

Odatle je:

$$\int_0^{\pi/2} obim(\mathcal{R}(\alpha)) d\alpha = \int_0^{\pi/2} 2w(\alpha) d\alpha + \int_0^{\pi/2} 2w(\alpha + \pi/2) d\alpha$$

Ukoliko u drugom od sabiraka uvedemo smenu $\alpha + \pi/2 = u$, dobijamo da je $d\alpha = du$, a granice smo sveli na opseg od $\pi/2$ do π . Sledi:

$$\int_0^{\pi/2} obim(\mathcal{R}(\alpha)) d\alpha = 2 \int_0^{\pi} w(\alpha) d\alpha$$

Nakon što iskoristimo Košijevu teoremu koja nam izražava obim konveksnog tela iz knjige [8] imamo:

$$\int_0^\pi w(\alpha) d\alpha = \text{obim}(\mathcal{B})$$

Iz nejednakosti koje smo do sada izveli sledi:

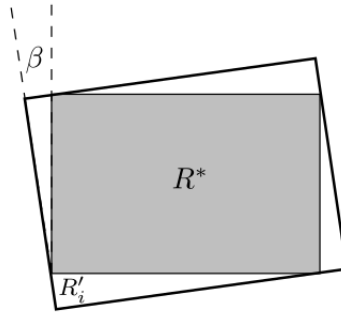
$$\frac{\pi}{2} \cdot \text{obim}(\mathcal{R}) \leq \int_0^{\pi/2} \text{obim}(\mathcal{R}(\alpha)) d\alpha = 2 \int_0^\pi w(\alpha) d\alpha = 2 \cdot \text{obim}(\mathcal{B})$$

odakle sledi tražena nejednakost. \square

Neka je \mathcal{R}^* minimalni presecajući pravougaonik za \mathcal{S} i neka je \mathcal{P}^* minimalni presecajući mnogougao za \mathcal{S} . Iz prethodne leme, sledi da je $\text{obim}(\mathcal{R}^*) \leq \frac{4}{\pi} \text{obim}(\mathcal{P}^*)$. Međutim, s obzirom da smo problem diskretizovali, naš algoritam ne određuje pravougaonik \mathcal{R}^* , već ga aproksimira jednim od pravougaonika iz m izabranih pravaca (onim koji ima najmanji obim). Zbog toga moramo da platimo određenu cenu, a sledeća lema nam govori i koliku.

Lema 3.2.4 *Za sve $i = 0, 1, \dots, m - 1$ važi da je $\text{obim}(\mathcal{R}_i) \leq \sqrt{2} \text{obim}(\mathcal{R}^*)$. Štaviše, postoji $i \in \{0, 1, \dots, m - 1\}$ tako da je $\text{obim}(\mathcal{R}_i) \leq (1 + \varepsilon) \text{obim}(\mathcal{R}^*)$.*

Dokaz. Izvodimo dokaz⁹ uz pomoć slike 3.6. Neka je β minimalna ugaona razlika



Slika 3.6: Greška prouzrokovana diskretizacijom leži u razlici uglova β između orijentacija \mathcal{R} i \mathcal{R}^* [3]

u orijentacijama \mathcal{R}_i i \mathcal{R}^* , $0 \leq \beta \leq \pi/4$. Konstruišemo oko \mathcal{R}^* pravougaonik \mathcal{R}'_i koji ga sadrži i koji je iste orijentacije kao \mathcal{R}_i . Svaka stranica novog pravougaonika

⁹Dokaz ove leme je preuzet iz [3].

\mathcal{R}'_i može da se izrazi kao proizvod $\cos \beta + \sin \beta$ i odgovarajuće stranice od \mathcal{R}^* koju dodiruje. Odatle sledi:

$$\text{obim}(\mathcal{R}'_i) = (\cos \beta + \sin \beta) \cdot \text{obim}(\mathcal{R}^*)$$

Od svih pravougaonika orijentacije α_i najmanji obim ima \mathcal{R}_i te imamo da je:

$$\text{obim}(\mathcal{R}_i) \leq \text{obim}(\mathcal{R}'_i) = (\cos \beta + \sin \beta) \cdot \text{obim}(\mathcal{R}^*) \leq \sqrt{2} \text{obim}(\mathcal{R}^*)$$

Kako se uzastopni pravci α_i razlikuju za najviše 2ε postoji i takav da je ugao β između orijentacija od \mathcal{R}_i i \mathcal{R}^* najviše ε . Za ovo i važi:

$$\text{obim}(\mathcal{R}_i) \leq (\cos \beta + \sin \beta) \cdot \text{obim}(\mathcal{R}^*) \leq (1 + \beta) \text{obim}(\mathcal{R}^*) \leq (1 + \varepsilon) \text{obim}(\mathcal{R}^*)$$

kao što je traženo. □

Iz svega ovoga sledi prva teorema ovog odeljka. Za dati pravougaonik \mathcal{R}_i koji algoritam vraća kao rešenje, ispunjeno je:

$$\text{obim}(\mathcal{R}_i) \leq \frac{4}{\pi} (1 + \varepsilon) \text{obim}(\mathcal{P}^*)$$

Drugi algoritam

Teorema 3.2.5 *Za bilo koje $\varepsilon > 0$, $(1 + \varepsilon)$ -aproksimativni algoritam za nalaženje presecajućeg mnogougla najmanjeg obima za dati skup n duži u ravni izvršava se u vremenu $O(1/\varepsilon) \text{poly}(n) + 2^{O((1/\varepsilon)^{2/3})} n$, gde je $\text{poly}(n)$ oznaka za polinomsku funkciju od n .*

Ova teorema je drugi važan rezultat rada [3]. Opišimo konstrukciju samog algoritma. Možemo algoritam prirodno podeliti na tri koraka. U prvom koraku treba da iskoristimo prvi algoritam, za $\varepsilon_1 = \frac{\varepsilon}{2+\varepsilon}$, čime dobijamo presecajući pravougaonik \mathcal{R} čiji je obim $(1 + \varepsilon_1)$ -aproksimacija minimalnog presecajućeg pravougaonika za \mathcal{S} . Neka je \mathcal{Q} kvadrat ivice $3 \cdot \text{obim}(\mathcal{R})$ koji je koncentričan sa \mathcal{R} i čije su ivice paralelne sa ivicama pravougaonika \mathcal{R} . Dužina ivice nije izabrana unapred, već posle računa, da bi se sve uklopilo.

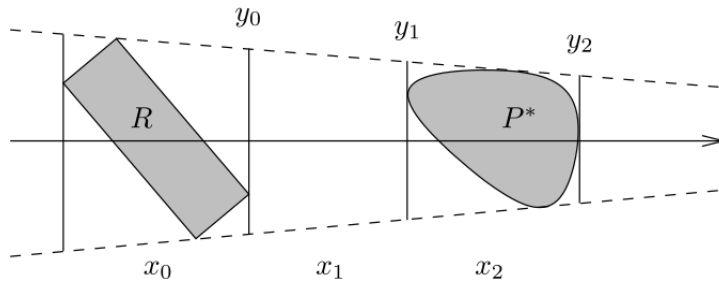
Drugi korak je da izaberemo $k = \lceil 48/\varepsilon \rceil$, a potom treba da podelimo \mathcal{Q} na mrežu $k \times k$ koju označimo sa \mathcal{Q}_δ ćelija stranice $\delta = 3 \text{obim}(\mathcal{R})/k$. Nakon toga, ideja je da nabrojimo sve konveksne mrežne poligone sa mrežnim čvorovima iz \mathcal{Q}_δ i da među njima izaberemo presecajući \mathcal{P}_δ sa najmanjim obimom.

U trećem koraku ostaje nam da između \mathcal{R} i \mathcal{P}_δ izaberemo onaj koji ima manji obim.

Označimo sa \mathcal{P}^* presecajući mnogougao najmanjeg obima za \mathcal{S} koji je na najmanjoj udaljenosti od \mathcal{R} koje smo izračunali u prvom koraku. Preciznije, od onih konveksnih mnogouglova koji su egzaktno rešenje, uzimamo za \mathcal{P}^* onaj koji je najbliže \mathcal{R} , gde je \mathcal{R} rešenje koje smo dobili primenjujući prvi aproksimativni algoritam u prvom koraku ovog drugog algoritma. Pritom, pod pojmom udaljenosti između dva kompaktna skupa \mathcal{A} i \mathcal{B} podrazumevamo $\min\{d(a, b) \mid a \in \mathcal{A}, b \in \mathcal{B}\}$. \mathcal{A} i \mathcal{B} se seku akko je njihova udaljenost 0. Pokažimo ispravnost algoritma. Za dokaz glavne teoreme, biće nam potrebna naredne dve leme:

Lema 3.2.6 *Ako je ispunjeno $\text{obim}(\mathcal{R}) \geq (1 + \varepsilon)\text{obim}(\mathcal{P}^*)$, tada je $\mathcal{P}^* \subseteq \mathcal{Q}$.*¹⁰

Dokaz. Pretpostavimo najpre da su \mathcal{R} i \mathcal{P}^* disjunktni i razmotrimo dve potporne prave, zajedničke za njih, koje ih dodiruju oba sa jedne strane. Bez gubitka opštosti, možemo pretpostaviti da su potporne prave simetrične u odnosu na x -osu. Bez gubitka opštosti možemo pretpostaviti da je $x_{\max}(\mathcal{R}) \leq x_{\max}(\mathcal{P}^*)$, pri čemu sa $x_{\max}(\mathcal{R})$ je označena najveća x -koordinata tačaka sa \mathcal{R} , slično i za $x_{\max}(\mathcal{P}^*)$. Pratimo sliku 3.7. Sada imamo dva slučaja da razmotrimo.



Slika 3.7: Isprekidane linije su spoljašnje zajedničke tangente za \mathcal{R} i \mathcal{P}^* . Podeoci x -ose određeni unutrašnjim pojedinačnim tangentama su redom x_0 , x_1 i x_2 . Veličine pojedinačnih unutrašnjih tangenti koje se nalaze između zajedničkih spoljašnjih tangenti su redom y_0 , y_1 i y_2 . Krajnju levu ne obeležavamo jer nam nije potrebna za naše nejednakosti. [3]

Slučaj 1: $x_{\max}(\mathcal{R}) < x_{\min}(\mathcal{P}^*)$

Ono što je važno da ovde zapazimo jeste da kako su i \mathcal{R} i \mathcal{P}^* presecajući mnogouglovi od \mathcal{S} , svaka duž iz \mathcal{S} ima barem jednu tačku u \mathcal{R} i barem jednu tačku u

¹⁰Dokaz ove leme preuzimamo iz [3].

\mathcal{P}^* . Na osnovu konveksnosti svake duži iz \mathcal{S} , bilo koja vertikalna duž između dve potporne prave, koja ima x -koordinatu koja je najmanje $x_{max}(\mathcal{R})$ i najviše $x_{min}(\mathcal{P}^*)$ seče sve ove duži, iz čega sledi da je ona degenerisani presecajući pravougaonik čiji je obim njena dvostruka dužina.

\mathcal{P}^* je pravougaonik najmanjeg obima koji je presecajući za \mathcal{S} i najbliži \mathcal{R} . Duž y_1 sa slike 3.7 je takođe (degenerisani) presecajući pravougaonik, te važi $obim(\mathcal{P}^*) \leq 2y_1$ (obim degenerisanog pravougaonika je dvostruka dužina te duži). Sa druge strane, $obim(\mathcal{P}^*) \geq 2y_1$ jer je upisano konveksno telo \mathcal{P}^* ili ova duž ili zatvorena konveksna linija upisana u četvorougao koji formiraju tangente, y_1 i y_2 . Iz toga sledi da mora da važi $y_0 > y_1$ jer bi u suprotnom bilo $2y_0 \leq 2y_1 \leq obim(\mathcal{P}^*)$ pa \mathcal{P}^* ne bi bio presecajući mnogougao za \mathcal{S} najmanjeg obima koji je najbliži \mathcal{R} , već bi to bio deformisani pravougaonik y_0 .

\mathcal{R} je $(1 + \varepsilon_1)$ -aproksimacija za presecajući pravougaonik od \mathcal{S} najmanjeg obima, te važi:

$$obim(\mathcal{R}) \leq (1 + \varepsilon_1)obim(\mathcal{P}^*) \leq (1 + \varepsilon_1)2y_1$$

Ovde smo iskoristimo nejednakost koju smo malopre izveli: $obim(\mathcal{P}^*) \leq 2y_1$.

Sa druge strane, ako iskoristimo pretpostavku koja nam je data u lemi, uz ono što je očigledno tačno, a to je $obim(\mathcal{R}) \geq 2y_0$ (odakle sledi da je $y_0 \leq (1 + \varepsilon_1)y_1$) imamo da je:

$$obim(\mathcal{P}^*) \leq \frac{1}{1 + \varepsilon}obim(\mathcal{R}) = \frac{1 - \varepsilon_1}{1 + \varepsilon_1}obim(\mathcal{R}) \leq (1 - \varepsilon_1)2y_1$$

Kako je $obim(\mathcal{P}^*) \geq 2y_2$, odatle je $y_2 \leq (1 - \varepsilon_1)y_1$.

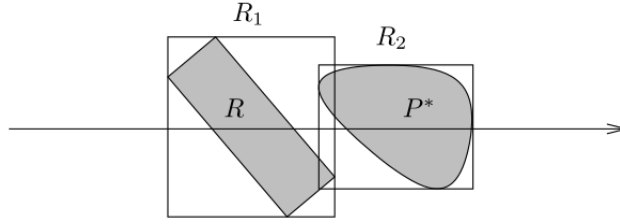
Neka je $x_0 = x_{max}(\mathcal{R}) - x_{min}(\mathcal{R})$, $x_1 = x_{min}(\mathcal{P}^*) - x_{max}(\mathcal{R})$, $x_2 = x_{max}(\mathcal{P}^*) - x_{min}(\mathcal{P}^*)$. Jasno je da važi: $x_0 \leq \frac{1}{2}obim(\mathcal{R})$. Na osnovu sličnosti trougla je:

$$\frac{x_1}{x_2} = \frac{y_1 - y_0}{y_1 - y_2} \leq \frac{(1 + \varepsilon_1)y_1 - y_1}{y_1 - (1 - \varepsilon_1)y_1} = 1$$

Odatle je $x_1 \leq x_2 \leq \frac{1}{2}obim(\mathcal{P}^*) \leq \frac{1}{2}obim(\mathcal{R})$.

Neka je \mathcal{R}' najmanji pravougaonik koji ima ose paralelne koordinatnim koji sadrži \mathcal{P}^* i koji je koncentričan sa \mathcal{R} . Njegova širina je tada najviše $x_0 + 2(x_1 + x_2) \leq \frac{5}{2}obim(\mathcal{R})$, a visina mu je najviše $y_1 < y_0 \leq \frac{1}{2}obim(\mathcal{R})$. Kako je $(\frac{5}{2})^2 + (\frac{1}{2})^2 \leq 3^2$ pravougaonik \mathcal{R}' sa ivicama paralelnim koordinatnim osama je sadržan u kvadratu \mathcal{Q} stranice $3obim(\mathcal{R})$ koji je koncentričan sa \mathcal{R} i paralelan sa \mathcal{R} (napomenimo da \mathcal{R} ne mora nužno da ima ivice paralelne koordinatnim osama). Odavde je $\mathcal{P}^* \subseteq \mathcal{Q}$.

Slučaj 2: $x_{max}(\mathcal{R}) \geq x_{min}(\mathcal{P}^*)$. Slučaj je ilustrovan slikom 3.8.



Slika 3.8: ilustrovan slučaj 2 [3]

Stavimo \mathcal{R} u najmanji pravougaonik čije su ivice paralelne koordinatnim osama i označimo taj pravougaonik sa \mathcal{R}_1 . Slično uradimo za \mathcal{P}^* i označimo taj pravougaonik sa \mathcal{R}_2 . Neka nam je \mathcal{L}_1 duža stranica od \mathcal{R}_1 i neka nam je \mathcal{L}_2 duža stranica od \mathcal{R}_2 . Iz uslova ovog slučaja jasno je da se \mathcal{R}_1 i \mathcal{R}_2 delimično preklapaju. \mathcal{R}_2 je sadržan u kvadratu \mathcal{Q}' čije su ivice paralelne koordinatnim osama, koji je koncentričan sa \mathcal{R}_1 (samim tim i sa \mathcal{R}) i koji ima stranicu dužine $l_1 + 2l_2$. Primetimo, kao što smo to i ranije činili, da je $l_1 \leq \frac{1}{2}obim(\mathcal{R})$, kao i da je $l_2 \leq \frac{1}{2}obim(\mathcal{P}^*)$, a time i $l_2 \leq \frac{1}{2}obim(\mathcal{R})$. Iz toga je $l_1 + 2l_2 \leq \frac{3}{2}obim(\mathcal{R})$. Kako je $\left(\frac{3}{2}\right)^2 + \left(\frac{3}{2}\right)^2 < 3^2$, kvadrat \mathcal{Q}' koji smo ranije pomenuli leži u kvadratu \mathcal{Q} koji ima ose paralelne sa \mathcal{R} , koncentričan je sa \mathcal{R} i ima dužinu stranice $3obim(\mathcal{R})$ (primetimo da ovaj kvadrat ne mora imati ivice paralelne osama, zato smo proveravali da li i iskošen kvadrat \mathcal{Q}' može da stane u njega). Oдавде, ponovo $\mathcal{P}^* \subseteq \mathcal{Q}$.

Slučaj kada \mathcal{R} i \mathcal{P}^* imaju presečnih tačaka svodimo na prethodni slučaj istom konstrukcijom. Ovim je dokaz završen. \square

Lema 3.2.7 *Neka je $\mathcal{P}^* \subseteq \mathcal{Q}$. Tada postoji konveksni mrežni poligon \mathcal{P}_δ sa mrežnim čvorovima iz \mathcal{Q}_δ tako da je $\mathcal{P}^* \subseteq \mathcal{P}_\delta$ i $obim(\mathcal{P}_\delta) \leq (1 + \varepsilon)obim(\mathcal{P}^*)$*

Dokaz. Dokaz¹¹ prati ilustraciju na slici 3.9. Neka se u skupu V nalaze čvorovi od \mathcal{P}^* . Formiramo skup C od ćelija mreže \mathcal{Q}_δ koji sadrže čvorovi iz V . Dalje, pomoćni skup D čine diskovi čiji su centar čvorovi iz V , a poluprečnik im je $\sqrt{2}\delta$ gde je δ dužina jedne ćelije u mreži \mathcal{Q}_δ i iznosi, kao što smo se na početku dogovorili, $3obim(\mathcal{R})/k = 3obim(\mathcal{R})/\lceil 48/\varepsilon \rceil \leq \frac{\varepsilon}{16}obim(\mathcal{R})$. Diskovi su tako izabrani da bi u njima stali kvadrati iz C . Odatle imamo $V \subseteq C \subseteq D$, a takođe i $conv(V) \subseteq conv(C) \subseteq conv(D)$, gde nam je $conv$ oznaka za konveksni omotač. Nije

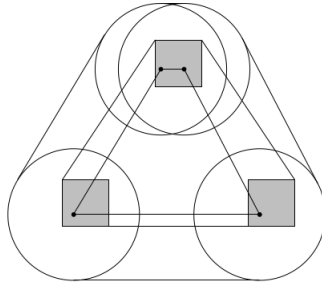
¹¹Dokaz ove leme je preuzet iz [3].

teško primetiti da ako jedno planarno konveksno telo sadrži drugo, manji je obim sadržanog tela. Sledi: $obim(conv(V)) \leq obim(conv(C)) \leq obim(conv(D))$. U skupu V su temena konveksnog mnogougla \mathcal{P}^* , pa je $conv(V) = \mathcal{P}^*$. Tvrđimo da možemo za \mathcal{P}_δ da uzmemo $conv(C)$. Najpre, vidimo da je ispunjeno $\mathcal{P}^* \subseteq \mathcal{P}_\delta$. Sa druge strane, $obim(\mathcal{P}_\delta) = obim(conv(C)) \leq obim(conv(D)) = obim(conv(V)) + 2\pi r = obim(\mathcal{P}^*) + 2\pi\sqrt{2}\delta$. Neka je \mathcal{R}^* presecajući pravougaonik najmanjeg obima od \mathcal{S} . Na osnovu lema koje smo ranije imali, znamo da je: $obim(\mathcal{R}) \leq \sqrt{2}obim(\mathcal{R}^*)$ i $obim(\mathcal{R}^*) \leq \frac{4}{\pi}obim(\mathcal{P}^*)$. Kada to sastavimo sa $\delta \leq \frac{\varepsilon}{16}obim(\mathcal{R})$, dobijemo:

$$2\pi\sqrt{2}\delta \leq 2\pi\sqrt{2} \cdot \frac{\varepsilon}{16} \cdot \sqrt{2} \cdot \frac{4}{\pi} \cdot obim(\mathcal{P}^*) = \varepsilon \cdot obim(\mathcal{P}^*)$$

$$obim(\mathcal{P}_\delta) \leq obim(\mathcal{P}^*) + 2\pi\sqrt{2}\delta \leq obim(\mathcal{P}^*) + \varepsilon \cdot obim(\mathcal{P}^*) = (1 + \varepsilon)obim(\mathcal{P}^*)$$

□



Slika 3.9: V (čvorovi), C (osenčeni kvadrati) i D (diskovi) [3]

Prodiskutujemo ukratko složenost drugog algoritma. Prvi korak se izvršava u vremenu $O(1/\varepsilon_1)poly(n) = O(1/\varepsilon_1)poly(n)$ gde je sa $poly(n)$ označen polinom po n . Ovo znamo jer prvi korak nije ništa drugo do pokretanje prvog algoritma koji smo opisali u ovom poglavlju. U drugom koraku svaki konveksni mnogougao čija temena odgovaraju čvorovima mreže dimenzija $k \times k$ ima $O(k^{\frac{2}{3}})$ temena [18, 19]. Takvih mnogouglova ima $2^{O(k^{\frac{2}{3}})}$ i svi se mogu nabrojati jer je dokaz iz rada Barana i Paha [17] konstruktivan. Računanje obima za svaki konveksni mnogougao čija su temena čvorovi mreže može se uraditi u vremenu $O(1/\varepsilon)$, a presek sa svakom duži se može naći u vremenu $O(n/\varepsilon)$. Ukupna vremenska složenost algoritma je stoga $O(1/\varepsilon)poly(n) + 2^{O((1/\varepsilon)^{\frac{2}{3}})}n$.

Glava 4

Polinomski algoritmi za disjunktne duži

U ovoj glavi biće opisana dva novija algoritma [9, 10] polinomske složenosti za rešavanje opisanog problema, uz uslov disjunktности duži, čija je složenost, prema tvrdnji autora, $O(n^5)$, odnosno $O(n^4)$. Pojedini delovi algoritma su nažalost vrlo oskudno opisani, tako da ostaje pitanje njihove korektnosti. U ovoj glavi razmatramo opštu strukturu ovih algoritama, diskutujemo pojedine delove koji su nedovoljno razjašnjeni i predlažemo nove pristupe u delovima algoritama koji su ostali nejasni. Napomenimo da su postupci koji su korišćeni u ovom radu kao zamena za nedostajuće ili nedovoljno jasne delove algoritama iz originalnih radova [9, 10] manje složenosti od onih koje autori tih radova navode. Otuda, uz dopune predložene u ovom radu dobijeni algoritmi postaju efikasniji od originalnih. Prema našem uverenju, predložene dopune dobro pokrivaju sve slučajeve koji se mogu javiti, ali u ovom radu tako nešto ne dokazujemo, već to ostavljamo za budući rad.

4.1 Algoritam složenosti $O(n^5)$

U ovom poglavlju razmatramo algoritam opisan u radu Jie i ostalih [9] iz 2017. godine u kome se razmatra varijanta našeg problema kod koje se pretpostavlja da su duži međusobno disjunktne. Strategija koja je korišćena zasniva se na kontrakciji velikog konveksnog omotača koji je konstruisan nad svim krajnjim tačkama zadatih duži u pravcu manjeg koji sadrži samo neophodne duži. Pokazano je da je algoritam koji je predstavljen vremenske složenosti $O(n^5)$.

Označimo rešenje našeg problema sa \mathcal{P}^* . To je konveksni mnogougao takav da

svaku duž iz \mathcal{S} ili sadrži u svojoj unutrašnjosti ili preseca granicom i pritom od svih mnogouglova koji zadovoljavaju taj uslov, ovaj ima najmanji obim. Neka je Ω konveksni omotač nad krajnjim tačkama duži iz \mathcal{S} i neka je Ψ donja granica za \mathcal{P}^* , tj. konveksni mnogougao koji je sigurno sadržan u \mathcal{P}^* . U prvoj fazi algoritma se iterativnim postupkom kontrakcije mnogougla Ω i širenja mnogougla Ψ dolazi do mnogougla \mathcal{P}_0 koji ima osobinu da su njegova temena na dužima iz \mathcal{S} koje sigurno preseca i granica mnogougla \mathcal{P}^* u istom redosledu. U drugoj fazi se za taj skup duži $\mathcal{F} \subseteq \mathcal{S}$ pronalazi mnogougao \mathcal{P} najmanjeg obima takav da njegova granica preseca ili dodiruje sve duži iz \mathcal{F} u istom redosledu kao i granica od \mathcal{P}_0 . U nastavku se sekvenca duži iz \mathcal{F} dopunjuje dužima iz $\mathcal{S} \setminus \mathcal{F}$ koje su eventualno ostale izvan mnogougla \mathcal{P} , uz ponavljanje postupka pronalaženja najkraćeg puta, čime se dobija mnogougao \mathcal{P}^* .

Postupak konstrukcije mnogouglova Ω i Ψ

Za konstrukciju konveksnog omotača Ω krajnjih tačaka duži iz \mathcal{S} možemo koristiti poznati Grahamov algoritam skeniranja [14] čija je složenost $O(n \log n)$. Ono što možemo da uočimo jeste da nema smisla sužavati ovaj omotač da bi dobili \mathcal{P}^* , ako \mathcal{P}^* nije sadržan u Ω . Na sreću imamo sledeću lemu:¹

Lema 4.1.1 Ω sadrži \mathcal{P}^* .

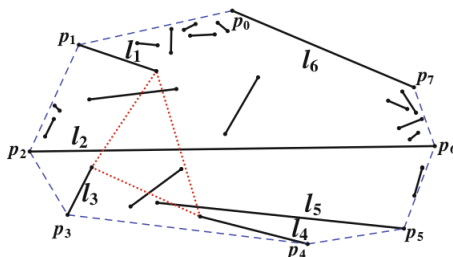
Dokaz. Ω sadrži sve duži u potpunosti. \mathcal{P}^* sa druge strane, sadrži barem po jednu tačku svake od duži (u unutrašnjosti ili na granici). Izaberimo za svaku duž po jednu njenu tačku na proizvoljan način. Najmanji konveksni mnogougao koji sadrži sve ove tačke će biti upravo njihov konveksni omotač, a on je u celini saržan u Ω , jer sve ove izabrane tačke pripadaju Ω . S obzirom da ovo važi za proizvoljno izabrane tačke, važiće i za \mathcal{P}^* jer će on biti konveksni omotač za neki odabir tačaka sa duži. \square

Za formiranje Ψ ćemo morati da se malo namučimo. Najpre, moramo da поделиmo skup duži iz \mathcal{S} koje imaju barem jednu zajedničku tačku sa granicom od Ω u tri podskupa: \mathcal{L}_0 koji sadrži duži koje imaju jednu krajnju tačku na granici od Ω , \mathcal{L}_1 koji sadrži duži koje imaju dve krajnje tačke na granici od Ω i \mathcal{L}_2 koji sadrži duži koje u potpunosti leže na granici od Ω . Ove skupove možemo formirati uporedo sa

¹Dokaz ove leme je preuzet iz [9].

formiranjem Ω . Za formiranje Ψ biće nam potrebne samo duži iz \mathcal{L}_0 , dok će nam duži iz \mathcal{L}_1 i \mathcal{L}_2 biti potrebne u narednim fazama algoritma.

Ψ konstruišemo kao konveksni omotač nad krajnjim tačkama duži iz \mathcal{L}_0 koje su u unutrašnjosti Ω . Ako u \mathcal{L}_0 imamo manje od dve duži, ne bavimo se konstrukcijom Ψ i jedina operacija koju primenjujemo je kontrakcija. U suprotnom nam ostaje da iz Ψ izbacimo ona temena za koje važi da duži koje im odgovaraju imaju deo u unutrašnjosti Ψ . Recimo, na slici 4.1 je to duž l_5 .



Slika 4.1: Primer konstrukcije mnogougla Ψ (označen crvenom bojom) za dati skup duži \mathcal{S} [9]

Ovako konstruisan mnogougao Ψ sigurno će biti sadržan u \mathcal{P}^* , o čemu govori sledeća lema.²

Lema 4.1.2 \mathcal{P}^* sadrži Ψ .

Dokaz. Pretpostavimo suprotno. Neka \mathcal{P}^* ne sadrži Ψ . U tom slučaju postoji duž iz \mathcal{L}_0 koja sa \mathcal{P}^* ima prazan presek, što je kontradikcija sa tim da je \mathcal{P}^* rešenje opisanog problema. \square

Prva faza algoritma

Težina problema koji razmatramo u ovom radu leži u tome što ne znamo na kojim dužima se nalaze temena kao ni njihov redosled u konveksnom mnogouglu najmanjeg obima. Cilj nam je da u narednim koracima postupno odredimo tu sekvencu duži. Krenimo najpre od sledeće leme.³

Lema 4.1.3 Granica od \mathcal{P}^* mora da preseče duži u \mathcal{L}_0 čije su krajnje tačke temena od Ψ .

²Ova lema nije formulisana ni dokazana u radu [9], već je originalni doprinos autora ovog rada.

³Dokaz ove leme je preuzet iz [9], uz izvesna preciziranja i korekcije.

Dokaz. \mathcal{P}^* mora da sadrži u sebi duži iz \mathcal{S} ili da ih seče svojom granicom. Duži iz \mathcal{L}_0 koje su izabrane zbog Ψ su podskup od \mathcal{S} što znači da i njih \mathcal{P}^* mora da sadrži u unutrašnjosti ili da ih seče granicom. Kada bi ih sadržao, granica od \mathcal{P}^* bi morala da se seče sa granicom od Ω ili bi krajnje tačke duži iz \mathcal{L}_0 koje ne pripadaju Ψ morale da budu temena \mathcal{P}^* . Prvi slučaj nije moguć jer Ω sadrži \mathcal{P}^* . U drugom slučaju bismo mogli da pomeranjem krajnje tačke proizvoljne duži iz \mathcal{L}_0 , koja je teme mnogougla \mathcal{P}^* , ka drugoj krajnjoj tački, dobijemo konveksni mnogougao manjeg obima koji preseca \mathcal{S} što je kontradikcija sa tim da je \mathcal{P}^* rešenje problema. \square

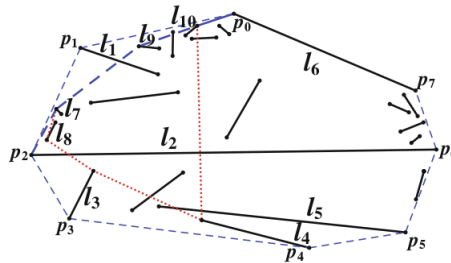
Ideja narednih koraka je da razmatrajući duži iz \mathcal{L}_0 i \mathcal{L}_1 proširimo dobijenu sekvencu duži dodatnim dužima za koje smo sigurni da ih granica od \mathcal{P}^* preseca. Ovo radimo tako što pronalazimo konveksni mnogougao \mathcal{P}_0 koji je rezultat iterativnog postupka sužavanja mnogougla Ω i širenja mnogougla Ψ . Ovaj mnogougao će imati osobinu da njegova temena leže na dužima iz \mathcal{S} koje granica od \mathcal{P}^* sigurno preseca i to u istom redosledu kao i granica od \mathcal{P}_0 . U nastavku opisujemo postupak konstrukcije mnogougla \mathcal{P}_0 .

Kao što je već rečeno, prilikom konstrukcije mnogougla \mathcal{P}_0 u obzir uzamemo duži iz \mathcal{L}_0 i \mathcal{L}_1 . Duži iz \mathcal{L}_2 ostaju na granici od Ω tokom čitavog iterativnog postupka i njih ne uzimamo u obzir u ovom razmatranju. Iako smo sve vreme govorili da širimo Ψ u cilju dobijanja \mathcal{P}_0 , to je samo iz tog razloga da bi znali kako da suzimo Ω . Kada u Ω više ne možemo da pomerimo nijedno teme, taj konveksni mnogougao je \mathcal{P}_0 . Na njegovog granici, očigledno i dalje leže duži iz \mathcal{L}_2 . Što se tiče duži koje nemaju nijednu zajedničku tačku sa granicom od Ω , koracima koji slede težimo da te duži zadržimo novoformiranim Ω , tj. ideja je da suzimo koliko god možemo Ω , ali da u tom procesu ne ispustimo nijednu duž iz \mathcal{S} .

U j -toj iteraciji algoritma imamo tekući spoljni mnogougao Ω_j i tekući unutrašnji mnogougao Ψ_j . Pritom, $\Omega_0 = \Omega$ i $\Psi_0 = \Psi$. Za svako teme P_i na granici od Ω_j označimo njegovog prethodnika sa P_{i-1} , a njegovog sledbenika sa P_{i+1} . Posmatrajmo duž $\overline{P_{i-1}P_{i+1}}$ i gledamo kakav ona ima položaj u odnosu na tekuće Ψ_j . Imamo tri mogućnosti:

- $\overline{P_{i-1}P_{i+1}}$ ne preseca Ψ_j i pritom važi da ako pomerimo P_i da bude u presečnoj tački $\overline{P_{i-1}P_{i+1}}$ i duži koja sadrži P_i nećemo ispustiti nijednu duž - u tom slučaju pomerimo P_i u tu odgovarajuću presečnu tačku i time smo suzili Ω_j , a Ψ_{j+1} nam je nepromenjeno Ψ_j .

- $\overline{P_{i-1}P_{i+1}}$ seče Ψ_j i pritom važi da ako P_i pomerimo u taj presek ne ispuštamo nijednu duž - sužavamo Ω_j tako što pomerimo granicu između P_{i-1} i P_{i+1} na odgovarajuću granicu od Ψ_j . Slično kao u prethodnom slučaju, Ψ_{j+1} nam je jednako Ψ_j .
- treći slučaj je bilo koji od ova dva prethodna koja smo naveli, osim što ne važi uslov da ne ispuštamo duži. Sa ovim se borimo tako što formiramo konveksni omotač koji sadrži sva temena mnogougla Ψ_j i krajeve ovih duži i dobijeni omotač nazovemo Ψ_{j+1} . Sada naprosto suzimo Ω_j tako što za granicu od Ω_j između P_{i-1} i P_{i+1} uzmemo odgovarajuću granicu na Ψ_{j+1} .



Slika 4.2: Primer kontrakcije

Na slici 4.2 nastaje slučaj tri koji je opisan iznad. Ako bismo umesto p_1 uzmeli presečnu tačku $\overline{p_0p_2}$ sa \mathcal{L}_1 , iz konveksnog omotača bi bili isključeni l_7, l_8, l_9, l_{10} . Zbog toga unutrašnji mnogougao Ψ_j proširujemo tako da sadrži i ove duži (to je sada Ψ_{j+1}), a onda Ω_j sužavamo na granicu sa Ψ_{j+1} i tako dobijamo Ω_{j+1} .

Kada dalje kontrakcije nisu moguće, prva faza algoritma je završena. Nazovimo dobijeni mnogougao \mathcal{P}_0 . Sva temena mnogougla \mathcal{P}_0 pripadaće nekim dužima iz \mathcal{S} . Radi lakšeg daljeg razmatranja, neka je sa \mathcal{F} označen skup duži iz \mathcal{S} koje sadrže temena mnogougla \mathcal{P}_0 .

Lema 4.1.4 *Važe sledeća svojstva:*

- Sve duži iz \mathcal{F} su presečene granicom mnogougla \mathcal{P}^* .
- Redosled presecanja duži iz \mathcal{F} od strane granice od \mathcal{P}^* je isti kao i kod \mathcal{P}_0 .
- Vremenska složenost procesa kontrakcije je najviše $O(n^2)$, gde je n broj duži u skupu \mathcal{S} .

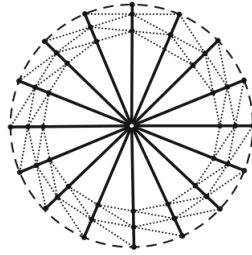
Dokaz. Tvrđenja koja smo naveli dokazujemo⁴ redom.

- Pretpostavimo da postoji neka duž iz skupa \mathcal{F} koja nije posećena granicom mnogougla \mathcal{P}^* . Da bi mnogougao \mathcal{P}^* zaista bio rešenje posmatranog problema, on mora da sadrži u svojoj unutrašnjosti tu duž. Zbog načina na koji je \mathcal{P}_0 konstruisan, on nije mogao da ispusti nijednu duž. Prema tome, \mathcal{P}_0 ili sadrži u svojoj unutrašnjosti posmatranu duž ili je seče granicom. Međutim, kako je ta duž iz skupa \mathcal{F} , znamo da je nemoguće da bude sadržana u \mathcal{P}_0 . Odatle sledi da se granica od \mathcal{P}_0 i granica od \mathcal{P}^* seku, što nam dalje dozvoljava da granicu od \mathcal{P}^* na tom delu izvan \mathcal{P}_0 svedemo na granicu od \mathcal{P}_0 . Time smanjujemo obim od \mathcal{P}^* što je kontradikcija sa tim da je \mathcal{P}^* traženo rešenje problema.
- \mathcal{P}_0 i \mathcal{P}^* su oba konstruisani sužavanjem Ω . Pritom duži su disjunktne. Iz prethodnog dela leme sledi da je redosled preseka duži iz \mathcal{F} sa granicama mnogouglova \mathcal{P}_0 i \mathcal{P}^* isti.
- Najgori slučaj u procesu kontrakcije je kada ne možemo da proširimo Ψ koji je na početku tačka i ostaje tačka, a Ω koji je veliki treba da skupimo do tačke. Na slici 4.3 vidimo kako taj primer izgleda. Slučaj koji će se ovde stalno ponavljati odgovara onome koji je opisan kao prvi u malopredšnjem razmatranju. Nećemo nikad da imamo presek sa tačkom jer naprosto imamo takvu konstrukciju. $\overline{P_{i-1}P_{i+1}}$ ne seče Ψ ni u jednom koraku, te je ovo onaj slučaj kada pomeramo P_i u presek te duži i one duži koja sadrži P_i . Dakle, mi se sporo krećemo ka središtu, odnosno ka Ψ (koji je ujedno i naš cilj \mathcal{P}^*) pomerajući P_i . U svakom procesu kontrakcije smanjujemo prečnik $\cos(2\pi/n)$ puta. Posle n^2 kontrakcija, prečnik je $|l| \times (\cos(2\pi/n))^{n^2}$. Ako razmatramo ovaj izraz kao funkciju po n i pustimo da n teži ka beskonačnosti, razvijanjem ove funkcije u red, dobićemo da nam je prečnik krajnjeg produkta (u beskonačnosti) 3.9×10^{-6} što je i više nego zadovoljavajuće. \square

Druga faza algoritma

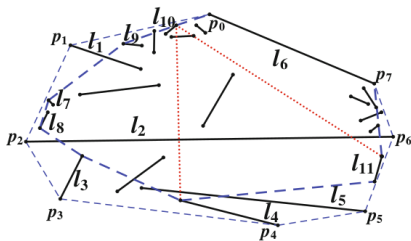
U drugoj fazi algoritma se najpre za dati skup duži \mathcal{F} koje određuju temena mnogougla \mathcal{P}_0 pronalazi najkraći zatvoreni put koji preseca sve duži iz \mathcal{F} u istom

⁴Dokaz u nastavku je preuzet iz [9].

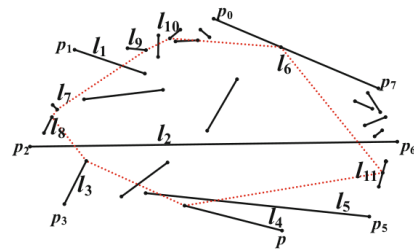


Slika 4.3: Najgori slučaj kontrakcije [9]

redosledu kao i granica od \mathcal{P}_0 (ovaj postupak će kasnije biti diskutovan detaljnije). Nakon što odredimo najkraći zatvoren put kroz duži iz \mathcal{F} u fiksiranom redosledu koji smo dobili, može da se desi da neke duži iz \mathcal{S} budu ispuštene zbog činjenice da sada ne pretpostavljamo da duži iz \mathcal{L}_2 uzimamo u celosti.



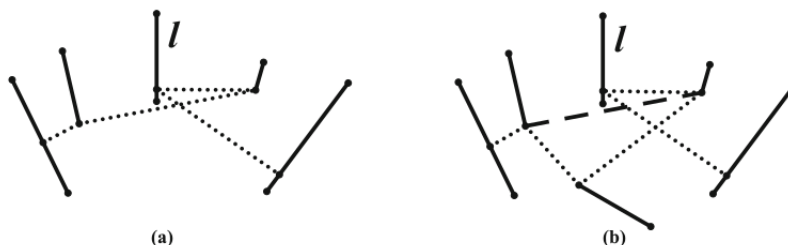
Slika 4.4: \mathcal{P}_0 (označen plavom isprekidanom linijom i segmentima l_{11} i l_6 [9])



Slika 4.5: Zatvoren put najmanje dužine kroz duži iz \mathcal{F} [9]

Redosled u kome su duži iz \mathcal{F} posećene od strane \mathcal{P}^* je fiksiran (na slici 4.4 to su duži $l_1, l_7, l_8, l_3, l_4, l_{11}, l_6, l_{10}, l_9, l_1$) te ostaje samo da odredimo redosled duži koje su ispuštene. Prvo treba utvrditi koje su to duži koje sadrže temena koja su odgovorna za to što su one ispuštene. Dalje, jednu po jednu, ubacujemo između duži koje sadrže ta temena, pazeći pritom da nam se ne naprave zaokreti na desno, jer onda imamo samopresecanje. Kada duž ispravno ubacimo na njeno mesto, ponovo računamo zatvoren najkraći put kroz duži koje su ranije bile prisutne u sračunatom poretku ali i kroz nju. Pretpostavimo da imamo m duži, r_1, r_2, \dots, r_m koje treba da ubacimo između dve duži \mathcal{L}_1 i \mathcal{L}_2 . Onda, za prvu računamo najkraći put jedanput, za drugu imamo dve mogućnosti, da je između \mathcal{L}_1 i r_1 ili između r_1 i \mathcal{L}_2 , za treću imamo tri mogućnosti, između \mathcal{L}_1 i r_1 , r_1 i r_2 ili između r_2 i \mathcal{L}_2 , itd. Ukupno, ovaj postupak se ponavlja $1 + 2 + 3 + \dots + m = m(m - 1)/2$ puta. Kako je $m < n$, postupak pronalaženja najkraćeg puta koji posećuje skup duži u datom redosledu

se ponavlja $O(n^2)$ puta. Autori rada [9] su za pronalazjenje najkraćeg puta koristili algoritam opisan u radu [15], čija je složenost, po navodima autora, $O(n^3)$.⁵ Otuda je složenost celog postupka $O(n^5)$.



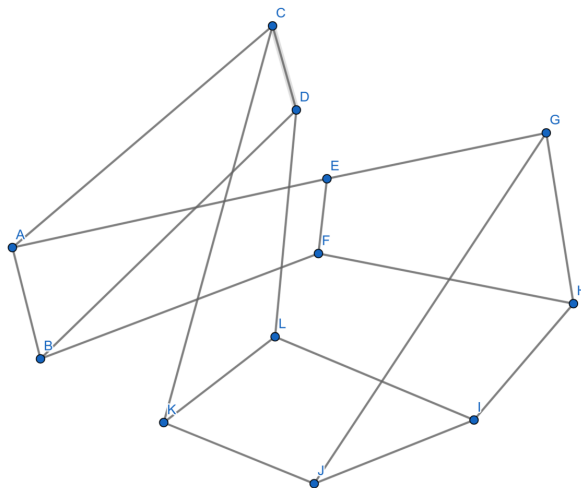
Slika 4.6: Zaokreti na desno, rezultat ubacivanja duži l na pogrešno mesto [9]

U ovom radu, umesto postupka opisanog u radu [15], za pronalazjenje najkraćeg puta koji posećuje dati skup duži u datom redosledu predlažemo drugačiji pristup, osmišljen po uzoru na algoritam koji su opisali Rapaport i Hasanzadeh [1], a koji je jedan od originalnih doprinosa ovog rada. Na slici 4.7 vidimo ciklus mnogouglova konstruisan na osnovu duži koje su nam date i njihovog redosleda koji je fiksiran (duži označene sivom bojom). Neka je redosled duži u sekvenci AB , EF , GH , IJ , KL , CD . Razmatramo svake tri uzastopne duži. Ukoliko su one takve da su krajnje tačke središnje od njih uzastopna temena konveksnog mnogougla formiranog nad krajnjim tačkama tri posmatrane duži, to je znak da na toj duži treba da napravimo „uvrtanje”. U suprotnom ih spojimo na uobičajen, intuitivan način. Nakon što smo konstruisali izvodljiv ciklus potrebno je da ga rasečemo po nekoj njegovoj ivici, a potom da ga ispravimo koristeći refleksije, translacije i rotacije. Detaljno objašnjenje ovog postupka, kao i pronalazjenja najkraćeg zatvorenog puta kroz ciklus mnogouglova, može se naći u [1], kao i u glavi 2 ovog rada. Složenost ovog postupka je $O(n \log n)$, čime se redukuje ukupna složenost čitavog algoritma, sa $O(n^5)$ na $O(n^3 \log n)$. Sa slike 4.7 možemo da primetimo da duž CD nije umetnuta na pravo mesto i da zato imamo samopresecanja na našem najkraćem putu.

Ostaje otvoreno pitanje da li opisani postupak pokriva sve moguće slučajeve različitih položaja duži u ravni. Konkretno, postavlja se pitanje da li se postupak na istovetan način može primeniti u slučaju postojanja problematičnih ivičnih duži

⁵Na žalost, rad [15] u kome je opisan algoritam za pronalazjenje najkraćeg puta koji posećuje dati skup duži u datom redosledu na koji se autori rada [9] pozivaju nije bio dostupan autoru ovog teksta. Otuda, nismo u mogućnosti da postupak rekonstruišemo u potpunosti, niti možemo da budemo sigurni u korektnost algoritma, kao ni u deklarisanu asimptotsku složenost.

i, ako je odgovor negativan, da li je postupak moguće proširiti tako da pokriva i takve slučajeve, a da složenost ostane polinomska. Ova pitanja ostavljamo za dalji rad.



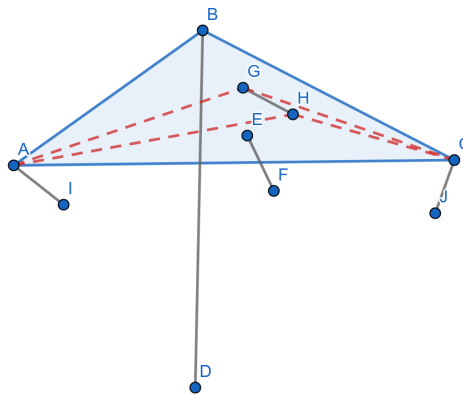
Slika 4.7: Formiranje ciklusa mnogouglova kroz koji tražimo najkraći put

4.2 Algoritam složenosti $O(n^4)$

U radu [10], koji se nadovezuje na prethodno opisani rad, postoji težnja da se proces kontrakcije i proces računanja najkraćeg puta nekako objedine. Takođe se polazi od konveksnog omotača svih krajnjih tačaka duži i razmatraju se razni slučajevi. Autori su naveli da se za rešavanje pojedinih slučajeva koristi *algoritam gumice* (engl. *rubberband algorithm*) detaljno izložen u radu [11]. Ovaj algoritam za dve tačke nekog (ne obavezno konveksnog) mnogougla pronalazi najkraći put između njih koji leži unutar tog mnogougla. Međutim, u radu [10] nedostaje objašnjenje na koji način je algoritam gumice primenjen u pojedinim slučajevima. Zato u ovom poglavlju, uz opis samog algoritma i pojedinačnih slučajeva koji se u njemu razmatraju dajemo i predlog kako bi algoritam gumice [11] mogao da se primeni u svakom od slučajeva.

Kao i ranije, najpre konstruišemo konveksni omotač Ω koji sadrži sve krajnje tačke iz \mathcal{S} . Nakon što smo konstruisali konveksni omotač svih krajnjih tačaka, obilazimo temena tog konveksnog omotača u smeru suprotnom skazaljke na satu (ili u smeru skazaljke na satu, svedjedno je). Za svako teme P_{i-1} posmatramo dva naredna

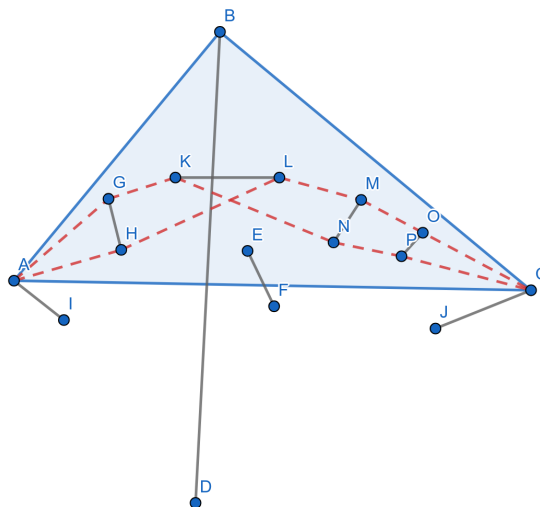
(neka su to P_i i P_{i+1}) i trougao $P_{i-1}P_iP_{i+1}$, pri čemu pokušavamo da smanjimo put od P_{i-1} do P_{i+1} a da on i dalje ostane konveksan i da nam se pri tome ne desi da ispustimo neku duž. Nekad može da se dogodi da je duž $P_{i-1}P_{i+1}$ rešenje. Problem ćemo rešiti razmatranjem nekoliko različitih slučajeva.



Slika 4.8: Deo konveksnog mnogougla na kome nastaje slučaj 1

Slučaj 1: Ne postoje duži koje nisu u potpunosti u trouglu $P_{i-1}P_iP_{i+1}$, a da pritom seku stranice $P_{i-1}P_i$ ili P_iP_{i+1} . To znači da duži mogu ili u celini biti sadržane u trouglu $P_{i-1}P_iP_{i+1}$, ili presecati njegovu stranicu $P_{i-1}P_{i+1}$. Pritom, primetimo da duži koje seku $P_{i-1}P_{i+1}$ ne treba posebno razmatrati, jer će one imati neprazan presek sa mnogougлом čak i ako odsečemo ceo trougao $P_{i-1}P_iP_{i+1}$. Možemo da imamo vrlo jednostavan primer, kao što je to onaj na slici 4.8, koji su i autori originalnog rada [10] iskoristili, ali na ovakvom primeru ne može mnogo toga da se zaključi, osim što je intuitivno jasno kako bi mnogougao unutar koga tražimo najkraći put od tačke C (koja ovde ima ulogu P_{i-1}) do tačke A (koja ovde ima ulogu P_{i+1}) trebalo da izgleda. Na slici 4.8 je to isprekidani crveni mnogougao. Ovaj mnogougao nazovimo dopustivi mnogougao. Sada je potrebno upotrebiti algoritam iz [11], koji za dve date tačke (u našem slučaju A i C) nekog prostog mnogougla određuje najkraći put između njih koji leži unutar pomenutog mnogougla. S obzirom da je u radu [10] ostalo nerazjašnjeno na koji način se formira dopustivi mnogougao, u ovom i narednim slučajevima naš glavni zadatak će biti upravo da osmislimo postupak za formiranje ovog mnogougla, što će biti jedan od glavnih doprinosa ovog rada. U slučaju 1 to možemo učiniti na sledeći način. Razmatrajmo situaciju koja je prikazana na slici 4.9.

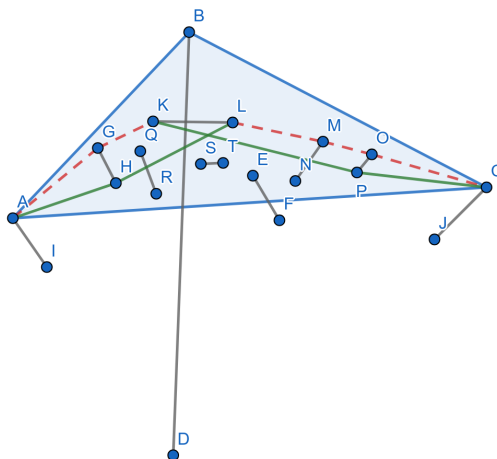
Iz primera sa slike 4.9, između ostalog, možemo zaključiti da gornja granica dopustivog mnogougla sigurno ne ide iznad gornje granice konveksnog omotača koji



Slika 4.9: Deo konveksnog mnogougla na kome nastaje slučaj 1

je formiran od krajnjih tačaka duži koje su unutar trougla ABC , kao i tačaka A i C . Dakle, gornja granica dopustivog mnogougla će se poklapati sa gornjom granicom ovog konveksnog omotača. Ukoliko je neka duž u celosti deo granice ovog omotača, tada je potrebno da u njoj dođe do „uvrtanja” kako bismo naterali put da prođe kroz tu duž. To „uvrtanje” rešavamo tako što razmotamo mnogougao refleksijama, rotacijama i translacijama, na način na koji je to opisano u glavi 2. „Uvrtanja” se dešavaju na dužima koje su potpuno u trouglu ABC i koje su deo gornjeg dela granice konveksnog omotača krajnjih tačaka duži koje su unutar trougla ABC (na slici 4.9 je to duž KL). Donja granica dopustivog mnogougla konstruiše se tako što za svaku duž odredimo krajnju tačku koja je bliža AC i te tačke spojimo redom na putu od A do C . Pritom, ako nađemo na duž koja cela leži na gornjoj granici formiranog mnogougla, tu napravimo uvrtanje. Na slici 4.9 uvrtanje se dešava na duži KL , H spajamo sa L , a K sa N , gde su H i N tačke sa donje granice koje su najbliže sa KL . Ovaj pristup deluje kao ispravno rešenje. Međutim, pretpostavimo da imamo još jednu duž u proizvoljnom položaju, na primer, barem još jednu u četvorouglu $KLMN$. Put može da prođe ispod nje i da je promaši. Posmatrajmo primer na slici 4.10.

Prestrogo bi bilo zahtevati za neku duž da put ide iznad nje, ili da je pogodi baš u nekoj krajnjoj tački. Na način koji je odabran na slici 4.10 dajemo mogućnost putu da prođe iznad nje, ali i da je pogodi u bilo kojoj tački. Ono što je sigurno, za ovakav način konstrukcije, je da nijedna duž neće biti ispuštena, a put biramo dovoljno



Slika 4.10: Deo konveksnog mnogougla na kome nastaje slučaj 1

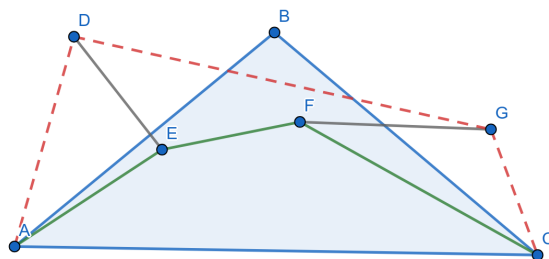
fleksibilno. Zelenom linijom je opisana donja granica dopustivog mnogougla. Na ovako formirani dopustivi mnogougao primenjujemo algoritam gumice za nalaženje najkraćeg puta od A do C . Zelenu liniju konstruišemo tako što najpre formiramo konveksni omotač tačaka A i C i krajnjih tačaka duži koje su u trouglu ABC , a koje su bliže AC , a potom izvršimo „uvrtanja” na mestima gde je to potrebno.

S obzirom na to da je gornja granica dopustivog mnogougla konveksna i put je konveksan. Zaista, pretpostavimo suprotno, da nam dobijeni put sadrži neku konkavnu uvalu. S obzirom na konveksnost gornje granice dopustivog mnogougla, imali bismo prostora da spojimo dva temena susedna sa vrhom uvala duži, koja je i dalje u mnogouglu. Dobili bismo kraći put što je kontradikcija sa pretpostavkom da je put koji smo izračunali najkraći. Ostaje još da razmotrimo složenost izvođenja opisanog postupka. Ono što mi radimo je pronalaženje najkraćeg puta od tačke A do tačke C koji je sadržan u dobijenom mnogouglu. Ovo, na osnovu rezultata iz rada [11] možemo uraditi u vremenu $O(n \log n)$.⁶ Takođe, napomenimo da je složenost formiranja dopustivog mnogougla $O(n \log n)$ na osnovu [14].

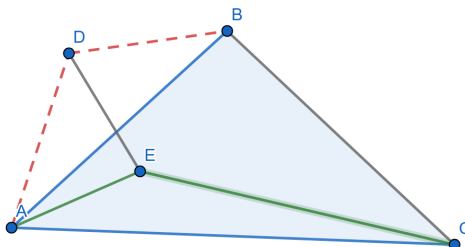
Slučaj 2: Postoje neke duži koje nisu potpuno unutar trougla ABC , već seku njegove spoljne ivice AB i BC ili se sa njima preklapaju. Autori rada naveli su da se može primeniti isti princip kao ranije, te stoga nastavljamo u tom pravcu. Na slici 4.11 imamo slučajeve kada su nam presečene stranice trougla ABC , dok na slici 4.12 imamo slučaj kada nam se jedna duž (BC) poklapa sa stranicom trougla. Za

⁶Autori rada u kome je algoritam originalno prikazan [10] su na ovom mestu računali $O(n^2)$, ali nije jasno zbog čega.

gornju granicu dopustivog mnogougla se uzima gornja granica konveksnog omotača krajnjih tačaka duži koje seku stranice AB i BC i tačaka A i C . Za donju granicu dopustivog mnogougla se uzima gornja granica konveksnog omotača krajnjih tačaka duži koje seku AB i BC koje su bliže AC , kao i tačaka A i C . Ako su krajnje tačke neke duži podjednako udaljene od AC uzimamo ih obe. Na mestima gde je to potrebno pravimo „uvrtanja”. U slučaju da nam je jedna od duži stranica trougla ABC , na primer BC (slika 4.12), pronalazimo put AEC kao najkraći (duž BC će se dalje tretirati u nekoj narednoj iteraciji kao slučaj 4 koji će kasnije biti objašnjen).



Slika 4.11: Deo konveksnog mnogougla na kome nastaje slučaj 2

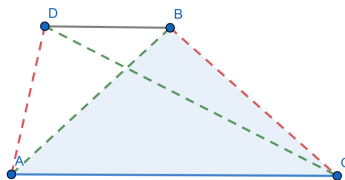


Slika 4.12: Deo konveksnog mnogougla na kome nastaje slučaj 2

Slučaj 3: Ovaj slučaj je prelak i zato za njega ne navodimo sliku. To je slučaj gde u trouglu ABC nema nijedne duži koja je potpuno u njegovoj unutrašnjosti i gde nemamo nijednu duž koja mu seče ili dodiruje spoljne ivice AB i BC . Jasno je da tada možemo da iskoristimo duž AC umesto puta koji se sastoji od AB i BC .

Slučaj 4: Postoji duž koja je vani, na način na koji vidimo na slici 4.13. Formiramo konveksni omotač od krajnjih tačaka te duži i tačaka A i C i njegovu gornju granicu uzimamo za gornju granicu dopustivog mnogougla. Donju granicu dopustivog mnogougla konstruišemo tako što napravimo „uvrtanje” na posmatranoj duži

van mnogougla (konkretno, na slici 4.13 A spojimo sa B , a C sa D). Ovim omogućavamo da se put odbije od duži BD , tj. da na duži BD imamo reflektujuće teme.



Slika 4.13: Deo konveksnog mnogougla na kome nastaje slučaj 4

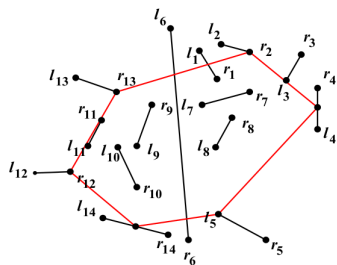
Gore opisani postupak funkcioniše pod pretpostavkom da su tačke P_{i-1} , P_i i P_{i+1} takve da mogu da formiraju trougao. U slučaju da ne mogu, treba fiksirati tačku P_{i-1} i tražiti P_{i+1} tako da može da se formira trougao. Ako obiđemo ceo krug, a nismo našli tu tačku, to je indikator da su sve duži kolinearne i pošto su disjunktne, nema nam druge nego da ih sve spojimo pravom, birajući one krajnje tačke na prvoj i poslednjoj duži koje nam daju najkraću dužinu duži. U suprotnom, kolinearne duži spajamo na ovaj način, a potom se bavimo tačkom koju smo našli. Proces traženja traje najviše $O(n)$.

Primetimo da smo u svim opisanim slučajevima umesto algoritma gumice mogli da za traženje najkraćeg puta kroz dopustivi mnogougao primenimo i postupak opisan u glavi o eksponencijalnom algoritmu, jer je problem sasvim sličan (ako fiksiramo tačke A i C , onda je malo jednostavniji, jer ne moramo da biramo „odgovarajuće tačke”).

Ponavljanjem ovih procesa, kroz kruženje oko mnogougla, u onom trenutku kada više ne možemo ništa da pomerimo dobija se traženi mnogougao najmanjeg obima P^* . Dokaz ove činjenice ostavljamo kao pravac budućeg rada.⁷ Ostaje još da odredimo složenost ovog postupka. Kao što je objašnjeno u prethodnom poglavlju, najgori slučaj je da ogroman konveksni omotač treba skupiti u tačku, a vremenska složenost potrebna da se postigne odgovarajuća preciznost je $O(n^2)$. Ukupna složenost algoritma je stoga $O(n^3 \log n)$. Složenost navedena u radu [10] je $O(n^4)$, ali

⁷U radu koji opisuje algoritam složenosti $O(n^4)$ [10] dokaz korektnosti je prilično diskutabilan i prepisan je iz prethodnog rada gde je opisan algoritam složenosti $O(n^5)$ [9]. U tom dokazu nisu uzete u obzir izmene koje su se dogodile u konstruisanju novog algoritma. Takođe iz tog dokaza se može zaključiti da je složenost samog algoritma $O(n^5)$, a ne $O(n^4)$ kako autori tvrde.

to je posledica toga što su oni računali da je složenost svakog pojedinačnog slučaja $O(n^2)$.



Slika 4.14: Primer iz rada [10]

Glava 5

Zaključak

U ovom radu razmatran je problem pronalaženja konveksnog mnogougla najmanjeg obima koji ima neprazan presek sa svakom od duži iz datog skupa. Rad daje pregled nekoliko postojećih algoritama, što egzaktnih, što aproksimativnih. I dalje nije poznato da li je u opštem slučaju moguće rešiti dati problem u polinomskom vremenu. U ovom radu razmatrani su aproksimativni pristupi zasnovani na minimalnom razapinjućem krugu, aproksimaciji traženog mnogougla pravougaonikom, kao i mnogougлом čija su temena čvorovi kvadratne mreže. Kada su u pitanju egzaktni algoritmi, u radu je prikazan algoritam predložen od strane Rapaporta i Hasanzadeha [1, 2], koji je u najvećoj meri i implementiran u toku rada na ovoj tezi.

U poslednjih nekoliko godina, pojavili su se radovi sa algoritmima polinomske složenosti koji su primenljivi na slučaj disjunktnih duži. Ti radovi pripadaju istim autorima. Međutim, problem sa njima je taj što, nažalost, nisu dovoljno jasno i precizno opisani, a nedostaje i prototipska implementacija. Kao svoj doprinos dali smo predlog kako bi mogli da se realizuju neki nejasni koraci. Pravac za dalji rad je implementacija i evaluacija naših dopuna zajedno sa kompletnim algoritmom, kao i dokaz korektnosti.

Bibliografija

- [1] Hassanzadeh, F., Rappaport, D.: *Approximation algorithms for finding a minimum perimeter polygon intersecting a set of line segments*, In: Proceedings of the 23rd International Workshop on Algorithms and Data Structures, LNCS, vol. 5664, pp. 363-374 (2009).
- [2] Rappaport, D.: *Minimum polygon transversals of line segments*, Int. J. Comput. Geom. Appl. 5, 243-265 (1995).
- [3] Dumitrescu, A., Jiang, M.: *Minimum-perimeter intersecting polygons*, Algorithmica 63(3) 602-615 (2012).
- [4] Goodrich, M.T., Snoeyink, J. *Stabbing parallel segments with a convex polygon*, Comput. Vis. Graph. Image Process. 49(2) 152-170 (1990).
- [5] Díaz-Báñez, J.M., Korman, M., Pérez-Lantero, P., Pilz, A., Seara, C., Silveira R.: *New results on stabbing segments with polygon*, P.G. Spirakis, M. J. Serna (Eds.), CIAC, in: Lect. Notes Comput. Sci., vol. 7878, Springer, pp. 146-157, (2013).
- [6] Edelsbrunner, H., *Algorithms in Combinatorial Geometry*, Springer-Verlag, (1987).
- [7] Shamos, M.I., Hoey, D., *Geometric intersection problems*, in Proceedings, 17th IEEE Symp. on Foundations of Computer Science, Houston, pp. 208-215, (1976).
- [8] Pach, J., Agarwal, P.K.: *Combinatorial Geometry*, Wiley, New York (1995).
- [9] Jia, Y., Jiang, B.: *The Minimum Perimeter Convex Hull of a Given Set of Disjoint Segments*, International Conference on Mechatronics and Intelligent Robotics. Springer, Cham, 308-318, (2017).

- [10] Li, N., Jiang, B., Li, N.: *An Efficient Algorithm for Finding the Minimum Perimeter Convex Hull of Disjoint Segments*, CSAE '18, October 22–24, Hohhot, China, (2018).
- [11] Li, F., Klette, R.: *Finding the Shortest Path Between Two Points in a Simple Polygon by Applying a Rubberband Algorithm*, Pacific Rim Conference on Advances in Image and Video Technology. Springer-Verlag, 280-291, (2006).
- [12] Czyzowicz, J., Egyed, P., Everett, H., Rappaport, D., Shemer, T., Souvaine, D., Toussaint, G., Urrutia, J.: *The Aquarium Keeper's Problem*, SODA 1991: Pages 459-464., (1991).
- [13] R. E. Burkard, G. Rote, E. Y. Yao, Z. L. Yu: *Shortest polygonal paths in space*, Computing, Volume 45, Pages 51-68, (1990).
- [14] R.L. Graham: *An efficient algorithm for determining the convex hull of a finite planar set*, Information Processing Letters, 1(4), 132–133, (1972).
- [15] L. Wang, B. Jiang, Q. Wei, D. He: *Research on the Algorithm for Euclidean Shortest Paths of Visiting Line Segments in the Plane*, ICIC Express Letters, 8(6), 1683-1688, (2014).
- [16] M. Claverol, D. Garijo, C. I. Grima, A. Márquez, C. Seara: *Stabbers of line segments in the plane*, Comput. Geom. 44(5), 303–318 (2011).
- [17] I., Bárány, J., Pach: *On the number of convex lattice polygons*, Comb. Probab. Comput. 1, 295–302 (1992).
- [18] G. E. Andrews: *An asymptotic expression for the number of solutions of a general class of Diophantine equations*. Trans. Am. Math. Soc. 99, 272–277 (1961).
- [19] G. E. Andrews: *A lower bound for the volume of strictly convex bodies with many boundary lattice points*, Trans. Am. Math. Soc. 106, 270–279 (1963).