Melkmanov algoritam za pronalaženje konveksnog omotača prostog poligona

Opis i implementacija algoritma u okviru kursa
Geometrijski algoritmi
Matematički fakultet

Viktor Novaković, 1063/2022

Novembar 2023.

Sažetak

U ovom radu se opisuje Melkmanov algoritam za pronalaženje konveksnog omotača prostog poligona i način na koji je implementiran. Algoritam je implementiran u okviru kursa Geometrijski algoritmi, na master studijama Matematičkog fakulteta.

Sadržaj

1 Opis problema

Problem konveksnog omotača je osnovni geometrijski problem u kom se traži najmanji konveksni poligon koji potpuno obuhvata skup tačaka u ravni. Ovaj konveksni omotač se može zamisliti kao elastična traka koja se stavlja oko skupa tačaka, čime se formira najmanji konveksan oblik koji obuhvata sve tačke.

Melkmanov algoritam [1] se bavi generisanjem konveksnog omotača nad prostim poligonom. Poligon predstavlja ravni geometrijski oblik sastavljen od niza temena (tačaka) i ivica koje ih povezuju. Svako teme poligona definisano je svojim koordinatama u ravni. Ono što razlikuje prosti poligon od opšteg poligona jeste to što su njegove ivice međusobno disjunktne i ne seku se, osim na svojim krajevima.

Melkmanov algoritam je poznat po svojoj efikasnosti u generisanju konveksnih omotača nad prostim poligonima. Za razliku od naivnih algoritama za rešavanje istog problema, koji često imaju kubnu složenost i brže rastu sa brojem tačaka, Melkmanov algoritam se odlikuje linearnom složenošću.

Važno je napomenuti da Melkmanov algoritam donosi osetno poboljšanje u performansama u odnosu na naivne pristupe generisanju konveksnih omotača, čineći ga značajnim alatom u oblasti geometrijskih algoritama.

2 Opis algoritma

Melkmanov algoritam se zasniva na ideji održavanja deka (dvostranog reda) koji se proširuje kako bi pratio konveksni omotač. Algoritam funkcioniše u više koraka, gde se temena poligona procesuiraju jedno po jedno.

U svakom koraku, algoritam proverava da li trenutno razmatrano teme pripada tekućem konveksnom omotaču:

- Ukoliko ne pripada, algoritam nastavlja dalje
- Ukoliko pripada, teme se dodaje u obe strane deka i prethodna temena koja više nisu deo omotača se izbacuju iz deka, jer su postala „unutrašnja" u odnosu na trenutno analiziranu tačku. Ova tehnika održava konveksnost omotača i omogućava algoritmu da se efikasno kreće kroz temena poligona.

Ovaj proces se ponavlja sve dok se ne obrade sva temena poligona, nakon čega se dobija konveksni omotač. Algoritam uvek poseduje odgovor za podatke koje je do tog trenutka obrađivao, bez potrebe za dodatnom obradom. Ova osobina je naročito korisna kada su podaci preveliki da bi stali u memoriju ili kada pate od kašnjenja mreže.

3 Implementacija algoritma

3.1 Ulazni podaci

Algoritam pretpostavlja da je na ulazu dobio uredjenu listu temena prostog poligona. Za generisanje nasumičnog prostog poligona, postupak uključuje generisanje slučajnog skupa tačaka, koji se zatim sortira prema uglovima koje te tačke formiraju sa tačkom koja predstavlja prosečnu vrednost svih tačaka unutar tog skupa.

3.2 Invarijanta petlje

Algoritam koristi dek (dvostrani red) za rešavanje problema, jer su pristup i modifikacija efikasni sa obe strane. Dek sadrži tekući konveksni omotač tačaka za koje znamo.

Jedna karakteristika konveksnih poligona (kao omotača koji pokušavamo da pronađemo) je da ako se krećemo duž temena u jednom smeru, pravimo samo desna skretanja. Ako se krećemo u suprotnom smeru, pravimo samo leva skretanja.

Dek koristi ovu karakteristiku kako bi održavao invarijantu. Ako čitamo dek udesno, pravimo samo desna skretanja. Ako čitamo unazad, pravimo samo leva skretanja. Još jedna invarijanta koju održavamo je da je poslednje dodato teme i na početku i na kraju deka.

3.3 Inicijalizacija deka

Na početku je potrebno da se u dek unesu prva tri temena poligona. Pošto želimo da održimo invarijantu, potrebno je da se ova temena unesu u određenom poretku. Proverava se u koju stranu se skreće kada se kreće temenima u početnom poretku. Ako se skreće desno, invarijanta je ispunjena i temena se dodaju u istom poretku. Ako se skreće levo, menjaju se mesta prvom i drugom temenu. Treće teme se dodaje na početak deka jer je poslednje teme koje je dodato.

3.4 Korak petlje

Pri razmatranju narednog temena potrebna su nam neka od temena tekućeg omotača: prvo (koje je istovremeno i poslednje), drugo i pretposlednje. Razmatramo naredno teme poligona i prvo proveravamo da li pripada tekućem omotaču. Ako se duž putanja *prvo teme - drugo teme - novo teme* i *pretposlednje teme - poslednje teme - novo teme* u oba slučaja skreće desno, iz toga saznajemo da je teme unutar tekućeg omotača pa ne može da bude deo rešenja, u tom slučaju nastavljamo dalje.

Ako se u bilo kojoj od putanja skreće levo to znači da se novo teme dodaje na početak i kraj deka. Ako je u prvoj putanji levo skretanje, to znači da je potrebno da se sa početka nakon novog temena uklanjaju temena sve dok se ne postigne desno skretanje i ispuni invarijanta.

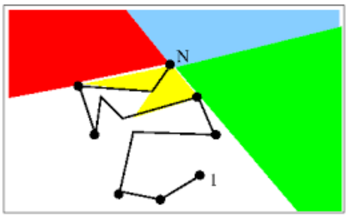
Ako je u drugoj putanji levo skretanje, slično prvom slučaju, sa kraja se pre novog temena uklanjaju temena sve dok se ne postigne desno skretanje. Moguć je slučaj da obe putanje skreću levo, tada se i sa početka i sa kraja uklanjaju temena dok se ne ispuni invarijanta.

Ilustracija ova četiri slučaja je na slici 1, gde je prikazan Melkmanov algoritam za tekući poligon, i obojene oblasti predstavljaju moguće slučajeve u zavisnosti od toga gde se nalazi naredno teme.

Oblast obojena u žuto predstavlja slučaj kada je novo teme ne pripada omotaču. Oblasti obojene u zeleno i crveno se odnose na slučajeve kada jedna od putanja skreće levo, a plava oblast je slučaj kada obe putanje skreću levo.

3.5 Složenost i korektnost

Melkmanov algoritam postiže linearnu složenost po broju temena u poligonu. Ovo je lako dokazati jer svaki od temena se dodaje na dek najviše dva puta i uklanja sa deka najviše dva puta.



Slika 1: Ilustracija Melkmanovog algoritma

Korektnost algoritma je dokazana u Melkmanovom radu [1]. Koriste se osobine konveksnog omotača H poligona P:

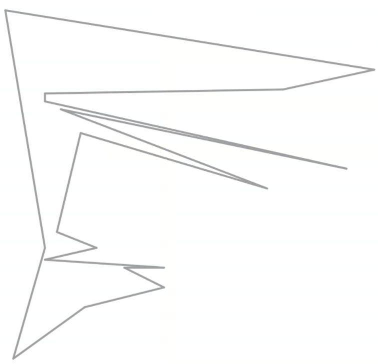
1. H je konveksan
2. P je sadržan u H
3. Skup temena iz H je podskup skupa temena iz P

Poslednja osobina je unapred ugrađena u algoritam. Za prve dve dovoljno je da se dokaže hipoteza da je dek D, predstavljen kao poligon, konveksan i da u sebi sadrži sva do sada viđena temena poligona P.

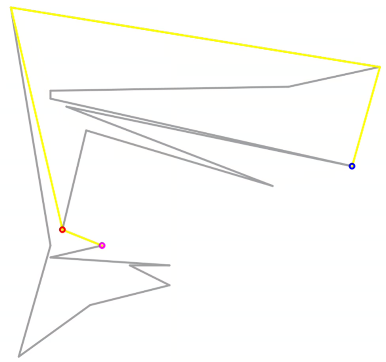
4 Zaključak i rezultati

Ovim radom predstavljen je Melkmanov algoritam za pronalaženje konveksnog omotača prostog poligona. Kroz detaljan opis algoritma i njegove implementacije, stekli smo uvid u ključne korake ovog efikasnog pristupa. Algoritam se ističe svojom linearnom složenošću u odnosu na broj temena poligona, što ga čini odgovarajućim izborom za brzu obradu i efikasno upravljanje velikim skupovima tačaka.

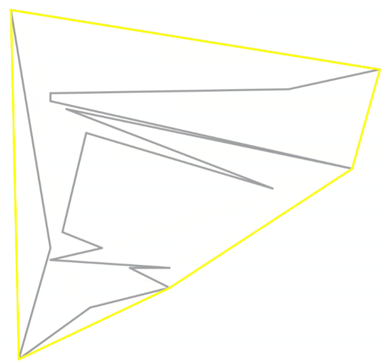
Na narednim slikama možete videti prikaz procesa i rezultate primene algoritma na dat prost poligon. Učitan poligon na slici 2, tekući omotač tokom izvršavanja na slici 3, i omotač poligona nakon izvršavanja algoritma na slici 4.



Slika 2: Učitan poligon



Slika 3: Tekući omotač tokom izvršavanja algoritma



Slika 4: Omotač poligona nakon izvršavanja algoritma

Literatura

[1] Avraham A. Melkman. On-line construction of the convex hull of a simple polyline. 1987.