

Algoritam za pronalaženje preseka n kružnica metodom brišuće prave

Anja Bukurov

Opis problema

Dato je n kružnica u ravni i potrebno je odrediti njihove preseke. Dve kružnice se seku ukoliko je rastojanje između njihovih centara manje od zbira poluprečnika, dodiruju se ukoliko je rastojanje između centara jednako zbiru poluprečnika, dok u ostalim slučajevima nema presečnih tačaka.

Ulaz: skup od n kružnica predstavljenih centrom i poluprečnikom

Izlaz: skup tačaka koje predstavljaju preseke kružnica

Naivno rešenje problema

Naivni algoritam redom proverava za svaku kružnicu da li ima preseka sa preostalim kružnicama i, ako ih ima, određuje ih i dodaje u skup presečnih tačaka (*intersectionsNaive*).

```
for(unsigned i=0; i<_circles.size(); i++) {  
    for(unsigned j=0; j<_circles.size(); j++) {  
        _circles[i].intersections(_circles[j], _intersectionsNaive);  
    }  
}
```

Algoritam zasnovan na metodi brišuće prave

Poboljšanje naivnog algoritma leži u tome da se ne porede sve kružnice sa svakom nego da se izaberu one sa kojima zaista postoji šansa da ima preseka. Održava se skup tačaka događaja (*events*) i skup krugova koje trenutno preseca prava (*status*).

Tačke događaja:

1. **Gornja tačka kruga** – prva tačka na kružnici na koju prava nailazi. U tom trenutku se proverava da li ima preseka sa ostalim kružnicama iz statusa. Zatim se i ta kružnica dodaje u status.

```
for(Circle c : _status) {  
    p.circle().intersections(c, _intersections);  
}  
_status.insert(p.circle());  
}
```

2. **Donja tačka kruga** – poslednja tačka na krugu koju će prava dodirnuti. U ovom trenutku kružnica se izbacuje iz statusa i svi njeni preseki sa ostalim kružnicama su već određeni i sačuvani u skupu presečnih tačaka (*intersections*) .

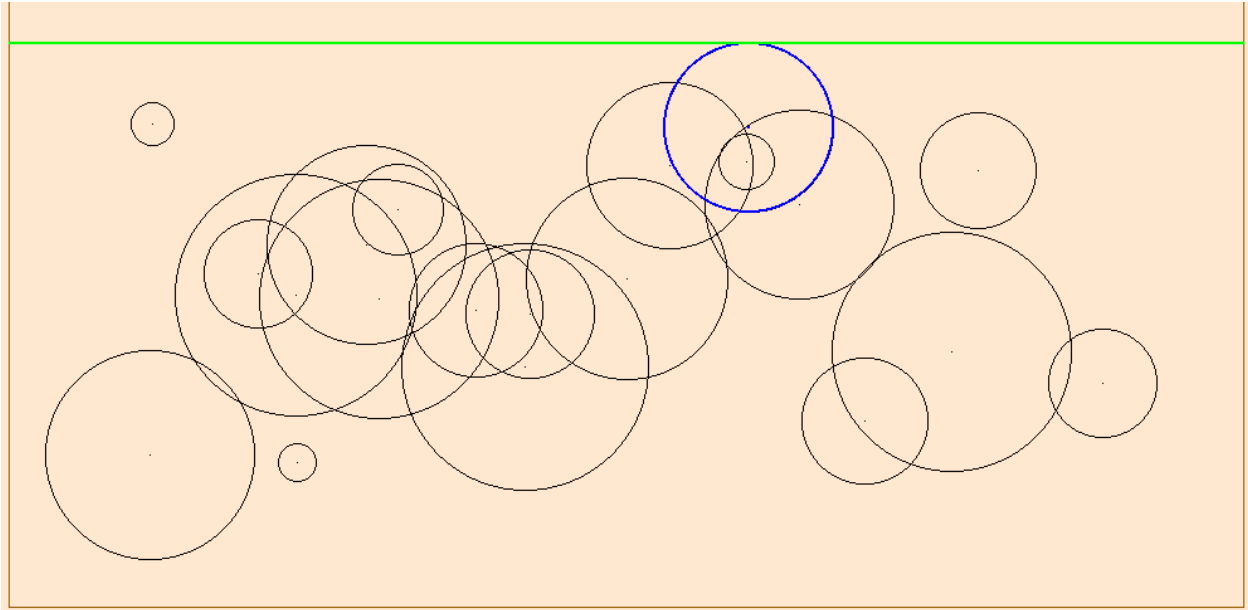
```
_status.erase(p.circle());
```

Tačke događaja čuvaju se u skupu koji se uređuje rastuće prema y koordinati tačke. Status je takođe skup koji se uređuje prema x koordinati najlevlje tačke kruga.

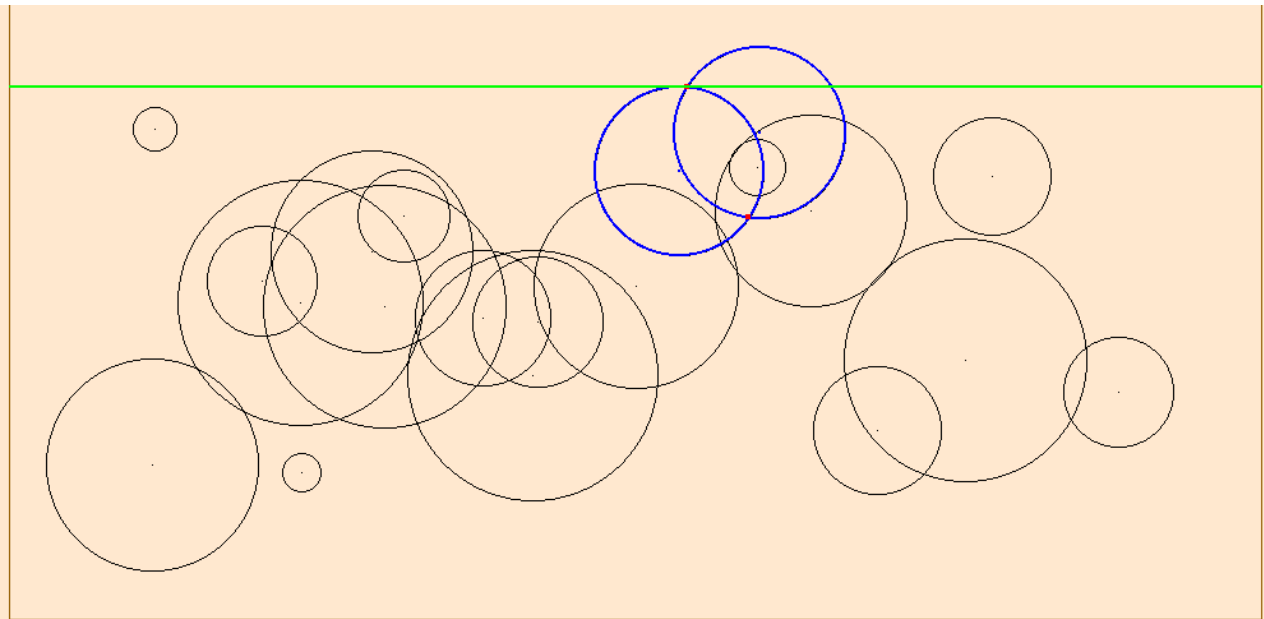
Algoritam se izvršava dok se ne obrade sve tačke događaja.

Vizuelizacija algoritma

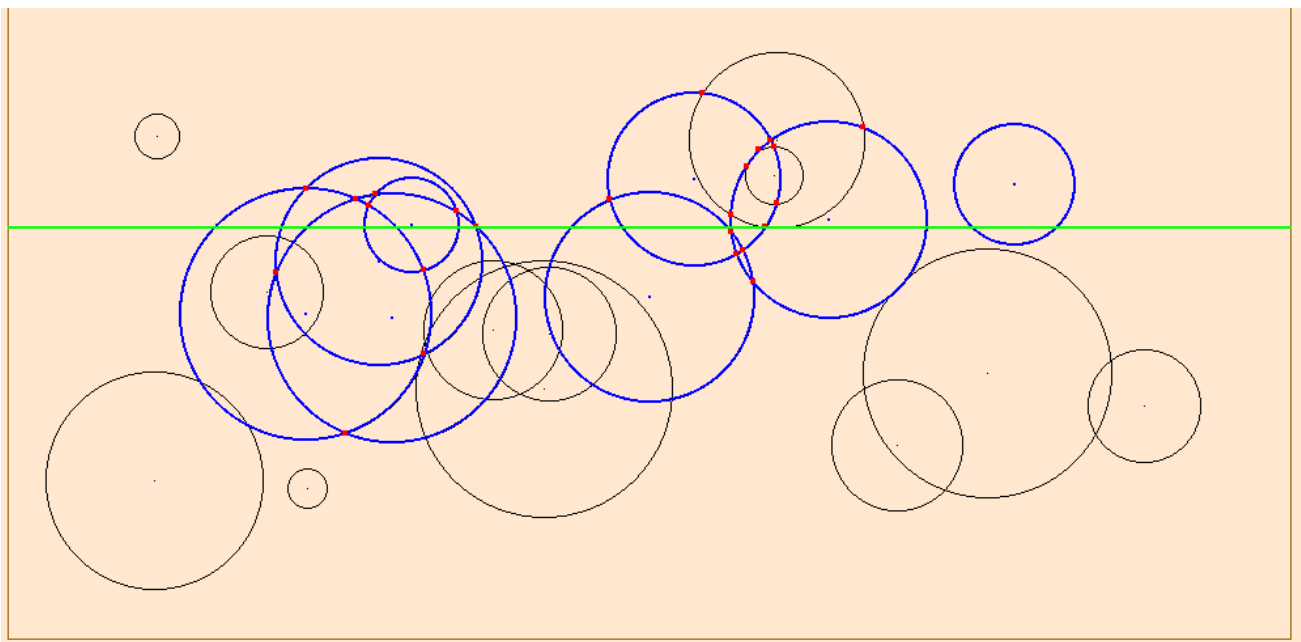
Na prvoj slici vidimo situaciju kada se nailazi na prvu tačku događaja. U tom trenutku, status je prazan pa se kružnica samo dodaje u status i boji u plavo.



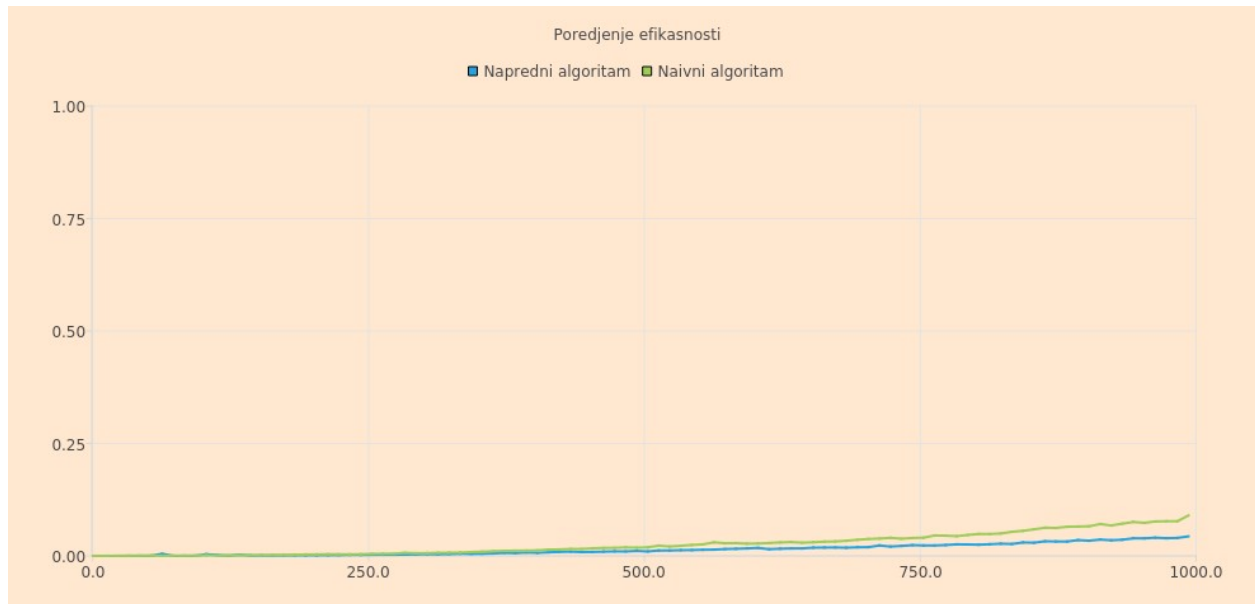
Na drugoj slici vidimo trenutak kada se nailazi na novu kružnicu, tj. još jednu tačku događaja „gornja tačka“. Pošto status nije prazan, ona se poredi sa kružnicama iz statusa i računaju se preseči. Nakon toga, kružnica je dodata u status i obojena plavom bojom.



Na trećoj slici prikazana je situacija kada se nailazi na tačku događaja „donja tačka“. U tom trenutku kružnica kojoj odgovara tačka izbačena je iz statusa i više nije obojena. Za nju su korektno određeni preseki sa svim kružnicama.



Poredjenje efikasnosti naivnog i naprednog algoritma



alg. / dim. ulaza	10	100	1000	10000	50000
naivni	0.00002	0.001308	0.083659	9.96476	242.665
optimalni	0.000045	0.000814	0.048207	5.04817	146.628

Testiranje ispravnosti algoritma

Naziv testa	Opis testa	Ulaz	Očekivani izlaz
noCircle	Proverava se da li algoritam radi ispravno kada nema kružnica čije preseke treba da traži.	[]	[]
oneCircle	Proverava se da li algoritam radi kada ima samo jednu kružnicu.	Niz dimenzije 1	[]

firstRandomTest	Proverava se da li algoritam ispravno radi na slučajno generisanom skupu kružnica.	Niz dimenzije 20	Poklapanje rezultata naivnog i naprednog algoritma
secondRandomTest	Proverava se da li algoritam ispravno radi na slučajno generisanom skupu kružnica.	Niz dimenzije 50	Poklapanje rezultata naivnog i naprednog algoritma
thirdRandomTest	Proverava se da li algoritam ispravno radi na slučajno generisanom skupu kružnica.	Niz dimenzije 500	Poklapanje rezultata naivnog i naprednog algoritma
bigRandomTest	Proverava se da li algoritam ispravno radi na slučajno generisanom skupu kružnica.	Niz dimenzije 5000	Poklapanje rezultata naivnog i naprednog algoritma
fixed	Proverava se da li algoritam radi za skup unapred određenih kružnica.	[[140, 150, 100], [190, 297, 98], [256, 189, 150], [350, 187, 180], [450, 350, 60]]	Poklapanje rezultata naivnog i naprednog algoritma
overlap	Proverava se da li algoritam radi ispravno za skup kružnica u kom su neke jednake.	[[350, 250, 100], [400, 250, 100], [400, 250, 100], [350, 250, 100]]	Poklapanje rezultata naivnog i naprednog algoritma [[375, 153], [375, 346]]

Poboljšanje naprednog algoritma

Algoritam zasnovan na brišućoj pravoj, koji je opisan u jednom od prethodnih poglavlja, nije optimalno rešenje, ali definitivno jeste pomak od naivnog.

Kako bismo mogli da ga poboljšamo? Posmatrajmo kvadrat u koji je upisana kružnica. Dužina stranice takvog kvadrata jednaka je prečniku kružnice.

Dakle, problem pronalaženja preseka kružnica mogao bi da se svede na problem pronalaženja preseka kvadrata. Klasa Circle mogla bi se proširiti za jedno polje tipa QRect u kom bi se čuvao kvadrat koji odgovara toj kružnici. Koordinate temena takvog kvadrata lako

se određuju na osnovu koordinata centra kružnice i njenog poluprečnika.

Ideja je da se za kvadrate proverava da li se seku, i ako se seku onda se određuje presek odgovarajućih kružnica. Provera da li se kvadrati seku može se izvesti takođe pomoću brišuće prave. Tačke događaja su gornje i donje teme kvadrata (ili odgovarajuće kružnice). U statusu se čuva projekcija kvadrata na pravu, odnosno jedna duž. Kada se naiđe na novi kvadrat u statusu se traže kvadrati sa kojima postoji presek.

Status više nije običan niz tačaka događa već stablo pretrage duži. Svaki čvor čuva kvadrat (odnosno duž koja predstavlja projekciju kvadrata – za svaku duž čuva se levi i desni kraj) koje brišuća prava seče kao i maksimalno desno teme u podstablu. Čvorovi su uređeni prema levom kraju duži. Dodatak ovakvoj strukturi je kružnica koja odgovara kvadratu.

Kako se sada određuju preseci? Kada se naiđe na novu kružnicu (ili kvadrat), u status se dodaje novi čvor koji sadrži duž i kružnica i ažurira se maksimum desnog kraja u podstablu. Za takav čvor tražimo sa kojima ima preseka u statusu. Kada se nađe presek duži treba izračunati i preseke odgovarajućih kružnica. Na ovaj način smanjuje se broj kružnica koje se porede.