Realizacija matrice rastojanja

Andreja Ilić
Prirodno Matematički Fakultet, Niš
e-mail: andrejko.ilic@gmail.com

April, 2011

1 Uvod

Problem realizacije datog konačnog metričkog prostora je, kao i većina problema iz teorije grafova, relativno mlad optimizacioni problem. Problem su uveli matematičari Hakimi i Yau 1964. godine, [2]. Neformalno problem se sastoji u nalaženju što boljeg predstavljanje konačnog metričkog prostora preko podgrafa traženog grafa.

Definicija 1.1 *Uredjeni par (M, d), pri čemu je M neprazan skup a funckija $d : M \times M \to R$ definisana kao:*

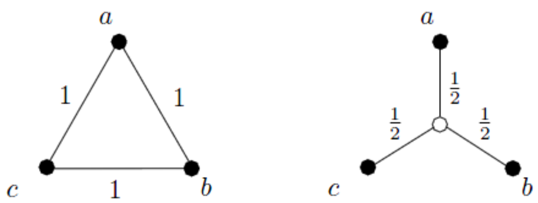
- $d(x, y) \geq 0$ *za svako* $x, y \in M$
- $d(x, x) = 0$ *za svako* $x \in M$
- $d(x, y) = d(y, x)$ *za svako* $x, y \in M$ *(simetričnost)*
- $d(x, y) + d(y, z) \leq d(x, z)$ *za svako* $x, y, z \in M$ *(nejednakost trougla)*

naziva se ***metrički prostor****. Ukoliko je $|M|$ konačan broj, za prostor kažemo da je i* ***konačan****.*

Definicija 1.2 *Težinski graf $G(V, E)$ je* ***realizacija konačnog metričkog prostora*** (M, d) *ako je $M \subset V$ i važi da je $d(x, y) = d^G(x, y)$ za svako $x, y \in M$. Sa $d^G(x, y)$ je označeno minimalno rastojanje izmedju čvorova označenih ovim elementima skupa M u grafu G.*

Čvorove iz skupa M nazivamo **osnovnim** (spoljašnjim), dok za čvorove iz skupa $V \setminus M$ kažemo da su **dodatni** (unutrašnji).

$$\begin{array}{cc} & \begin{array}{ccc} a & b & c \end{array} \\ \begin{array}{c} a \\ b \\ c \end{array} & \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$



Slika 1: *Primer matrice rastojanja, njene realizacije i optimalne realizacije.*

Definicija 1.3 *Za realizaciji $G(V,E)$ konačnog metričkog prostora (M,d) kažemo da je* ***optimalna*** *ukoliko ima najmanju težinu, gde težina grafa predstavlja sumu težina njegovih ivica.*

Problem optimalne realizacije je NP težak ([4], [5]). Mnoge klase grafova su napadnute, medjutim jedino klase stabala i unicikličnih grafova su u potpunosti okarakterisane. Složenost algoritama za ispitivanje ove vrste optimalne realizacije su kvadratne.

U ovom radu ćemo izneti heuristiku za opšti problem optimalne realizacije i izneti greške i nedostatke rada [1]. Testiranje ćemo pored korpusa iz ovog rada vršiti i na slučajno generisanom korpusu.

2 Poznati rezultati

U ovom delu su izneti poznatiji rezultati iz optimalne realizacije matrice rastojanja. Neke od njih ćemo koristiti u samom algoritmu. Bez gubljenja opštosti nadalje ćemo konačni metrički prostor (M,d) označavati i kao matricu rastojanja elemenata skupa M, odnosno $d(v,u) = d_{vu}$. Na početku iznet je prelep dokaz postojanja optimalne realizacije koji govori da je i sam problem dobro definisan.

Teorema 2.1 (Andreas Dress) *Svaki konačni metrički prostor ima optimalnu realizaciju.*

Dokaz: Dokažimo prvo da optimalna realizavija ima konačan broj čvorova. Posmatrajmo realizaciju $G = (V,E)$ prostora (X,d), pri čemu je $|V| = n$. Graf G možemo posmatrati kao uniju $\binom{n}{2}$ najkraćih puteva. Za svaki od puteva, puteve izmdju preostalim elemenata iz M možemo izabrati na takav način da je njihov presek sa posmatranim putem interval. Intervalni presek dva puta odredjuje najviše dva čvora na posmatranom putu. Broj čvorova na posmatranom putu nije veći od n^2. Dakle, broj čvorova u optimalnoj realizacije je ograničen sa n^4.

Težina ivice je ograničena nulom sa donje strane i rastojanjem $d(x,y)$ sa gornje strane. Kako imamo da važi

$$d(x,v_1) + d(x,v_2) + \ldots + d(v_m,y) \geq d(x,y)$$

za proizvoljan niz (v_i) čvorova iz V, dobijamo da skup W svih mogućih težina ivica u grafu čini konveksan poliedar. Naravno za gornju nejednakost postoji izbor čvorova za koji važi jednakost. Suma težina svih ivica $f : R^{|E|} \to R$ je neprekidna funkcija. Kako je skup W na kojem mi posmatramo preslikavanje kompaktan, dobijamo da je i njegova slika kompaktan skup. Kao takav sadrži tačku u kojoj se dostiže infimum funkije težine grafa. $\square$

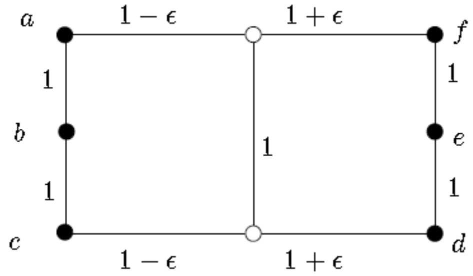
Optimalna realizacija ne mora da buti jedinstvena. Iako je ovo bila Dress-ova hipoteza, kontra primer je dat na slici 2. U tom primeru čak postoji neprebrojivo mnogo optimalnih realizacija.

Teorema 2.2 (Hakimi i Yau) *Optimalna realizacija ne sadrži trougao kao podgraf.*

Dokaz ove teoreme se svodi na konstrukciju prikazanu na slici 1.

Teorema 2.3 (Hakimi i Yau) *Ukoliko matrica rastojanja ima realizaciju u obliku stabla tada je ona jedinstvena i optimalna.*

$$\begin{array}{cc} & \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \end{array} & \begin{bmatrix} 0 & 1 & 2 & 3 & 3 & 2 \\ 1 & 0 & 1 & 3 & 4 & 3 \\ 2 & 1 & 0 & 2 & 3 & 3 \\ 3 & 3 & 2 & 0 & 1 & 2 \\ 3 & 4 & 3 & 1 & 0 & 1 \\ 2 & 3 & 3 & 2 & 1 & 0 \end{bmatrix} \end{array}$$



Slika 2: *Primer beskonačnog broja optimalnih realizacija.*

3 Algoritam

Kao što je napomenuto, u radu je analiziran heuristički pristup generalnom problemu optimalne realizacije. Na početku je iznet uopšteni pseudo kod algoritma. Svaki korak je detaljno analiziran u daljem delu.

Algoritam: Generalni opis algoritma

Input: Matrica rastojanja D reda n
Output: Graf G koji predstavlja realizaciju matrice D

```
Sažimanje i izbacivanje listova;
Greeady inicijalizacija;
Izbacivanje trouglova;
Reorganizacija podstruktura;
Izbacivanje nepotrebnih ivica;
Nadogradnja;
```

3.1 Sažimanje i izbacivanje listova

Ovo je samo jedan od mogućih predprocesiranja matrice rastojanja. Ideja je da se detekcijom listova, čvorova koji su u svakoj optimalnoj realizaciji listovi, uklone čime bi se dobila matrica manjeg reda. Na kraju bi se ovi čvorovi samo dodali realizaciji manje matrice.

Pretpostavimo da je čvor v list i da je njegov jedini sused čvor v. U matrici rastojanja vrste ovih čvorova se razlikuju za konstantu vrednost koja je jednaka upravo rastojanju izmedju ova dva čvora (osim u elementima (v, u) i (u, v)). Ovo nas navodi na konstrukciju

$$D_i(a) = D - aE_i$$

gde je E_i kvadratna matrica reda n definisana kao

$$(E_i)_{(j,k)} = \begin{cases} 1, j = i \neq k \\ 1, j \neq i = k \\ 0, \text{ inače} \end{cases}$$

U ideji smo imali da se za a uzima vrednost $d(u, v)$ i ona je bila najveća koju smo mogli da oduzmemo od čvora v. Manje vrednosti ne bi imale smisla iz prostog razloga što bi samo dodavale unutrašnje čvorove na ivicama grafa. Bez gubljenja opštosti možemo pretpostaviti da svi unutrašnji čvorovi imaju stepen veći ili jednak od 2. Sličnu ideju možemo uopštiti.

Teorema 3.1 (Hakimi i Yau) *$D_i(a)$ je matrica rastojanja ako i samo ako je $a \leq \frac{1}{2}(d_{pi}+d_{ir}-d_{pr})$ za svako $p, r \neq i$.*

Označimo sa $a_0(i) = \min\{(d_{pi} + d_{ir} - d_{pr})|p, r \neq i\}$. Novodobijena matrica rastojanja $D_i(a)$ naziva se **sažimanje** (eng. compaction). Pretpostavimo da je $G_i(a)$ realizacija matrice $D_i(a)$ pri čemu čvor v odgovara vrsti i. Označimo sa G graf dobijen od grafa $G_i(a)$ dodavanjem lista u koji je povezan sa v ivicom težine a. Tada imamo da je graf G realizacija matrice D pri čemu vrsti i odgovara čvor u.

Teorema 3.2 *Ako je $0 \leq a \leq a_0(i)$ i ako je $G_i(a)$ optimalna realizacija matrice $D_i(a)$, tada je G dobijen iz grafa $G_i(a)$ optimalna realizacija matrice D.*

Ukoliko se oduzimanjem vrednosti $a_0(i)$ elementima vrste i dobija jednakost vrsti i i j, tada se jednostavnim izbacivanjem vrste j dobija matrice manjeg reda. Slično gornjem postupku, dodavanjem novog lista dobija se realizacija matrice D.

U [1] autori su implementirali ovaj algoritam u složenosti $O(n^4)$. Ovde ćemo izneti implementaciju u složenosti $O(n^3)$. Na početku inicijalizujemo vrednosti $A = (a_0(i))_{i=1}^{n}$. U svakom koraku ispitujemo postojanje pozitivnog elementa niza A. Ukoliko takav element ne postoji, graf realizacije ne sadrži list. U suprotnom označimo indeks vrste sa v. Oduzimanjem $a_0(v)$ od elemenata vrste i kolone v dobijamo novu matricu rastojanja D_v. Moguća su dva slučaja:

- Novodobijena vrsta v je jednaka vrsti u. Tada vrstu v izbacujemo iz matrice kao i element sa indeksom v iz niza A. Nakon generisanja realizacije matrice D_v dodajemo novi list v kojeg povezujemo sa čvorom u.
- Novodobijena vrsta v je različita od svih ostalih vrsti. Element $A[v]$ postavljamo na 0. Nakon generisanja realizacije matrice D_v dodajemo novi list v' kojeg povezujemo sa čvorom v.

Vidimo da nismo ponovo inicijalizovali elemente niza A. Ovo imamo pravo da uradimo zbog:

$$d_v(u,v)+d_v(j,u)-d_v(v,j) = d(u,v)+a(v)+d(j,u)-d_v(v,j)-a(v) = d(u,v)+d(j,u)-d(v,j)$$

Složenost iznete implementacije je $O(n^3)$ što je ubrzanje za red veličine u odnosu na rad [1].

3.2 Greeady inicijalizacija

Kompletan graf svakako predstavlja realizaciju matrice D. Naravno i u slučaju kada ne dodajemo nove dodatne čvorove ovo ne predstavlja optimalnu realizaciju. Medjutim optimalnu realizaciju bez dodatnih čvorova možemo dobiti sledećim gramzivim algoritmom:

- Inicijalizujemo graf $G = (V, \{\})$ i matricu rastojanja d^G grafa G.
- Sortiramo elemente matrice D u neopadajućem redosledu u niz *edges*
- Za svaku vrednost d_{vu} iz niza *edges* ispitujemo da li je trenutna vrednost $d^G(v,u)$ jednaka d_{vu}. Ukoliko jeste, prelazimo na sledeći element niza *edges*. Inače, ubacujemo ivicu (v,u) u graf G i dodeljujemo joj težinu d_{uv}. Nakon ubacivanja ivice vršimo ponovnu inicijalizaciju vrednosti matrice d^G.

Algoritam: Pseudo kod algoritma sažimanja

Input: Matrica rastojanja D reda n

Output: Nova matrica rastojanja D' i stek operacija koje treba izvršiti nakon realizacije matrice D'

```
inicijaliacija niza A ;
S = {};
while postoji pozitivni element a[v] niza A do
    umanji v-tu vrstu i kolonu matrice D za vrednost a[v];
    u je indeks vrste jednake vrsti v;
    if u ≠ −1 then
        izbaci vrstu v iz matrice;
        izbaci element a[v] iz niza A;
        ubaci operaciju (v, u, a[v]) u stek operacija S;
    else
        a[v] = 0;
        ubaci operaciju (v, a[v]) u stek operacija S;
    end
end
return (D, S);
```

Najveći problem u ovom pristupu je reinicijaliacija matrice d^G pri ubacivanje nove ivice u graf. Označimo novu ivicu sa (v, u). Udaljednosti od čvorova v i u do svih ostalih čvorova možemo računati kao

$$d^G[v][w] = \min\{d^G[v][w], d_{vu} + d^G[u][w]\}$$

Pored ovog pristupa, zbog donje složenosti, rastojanja od čvorova u i v se mogu računati i Dijkstrinim algoritmom. Nakon ponovne inicijalizacije rastojanja od čvorova v i u, ostale elemente matrice rastojanja inicijalizujemo u konstantnom vremenu kao:

$$d^G[a][b] = \min\{d^G[a][b], d^G[a][v] + d_{vu} + d^G[v][b], d^G[a][u] + d_{uv} + d^G[v][b]\}$$

Složenost reinicijalizacije matrice d^G je kvadratna. Maksimalni broj ivica koje možemo ubaciti u graf ovim postupkom je upravo maksimalni broj ivica - $\binom{n}{2}$. Konačna složenost algoritma je $O(n^2 \log n + n^4)$. U radu [1] je ovde napravljena greška i rečeno je da je složenost ovog dela algoritma samo $O(n^2 \log n)$.

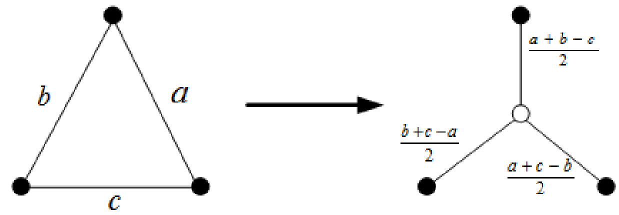
Ukoliko je ishod ovog algoritma stablo ili unciklični graf, on ujedno predstavlja i optimanlno rešenje. Dakle nakog gramzivog algoritma u tom slučaju možemo prekinuti i vratiti optimalni rezultat.

Za testiranje ovog pristupa može se koristiti korpus problema *Calls*, IPSC 2004.

3.3 Izbacivanje trouglova

Ovaj deo algoritma je najlakši kako za implementaciju tako i za konstrukciju. Naime sa slike 1. smo videli kako izgleda optimalna realizacija matrice rastojanja reda 3. U opštem slučaju optimalnu realizaciju trogula dobijamo konstrukcijom sa slike 3.

Kako optimalna realizacija ne sadrži trouglove, za svaku trojku čvorova iz grafa dobijenog gramzivim algoritmom ćemo primeniti gornju konstrukciju ukoliko ona dovodi do manje težine grafa. Za ovo nije potrebno da posmatrana trojka čvorova (a, b, c) čini trougao. Možemo posmatrati i puteve $a - b - c$ za koje važi da je $d_{ac} < d_{ab} + d_{bc}$. Složenost ovog dela je očigledno jednaka $O(n^3)$. Takodje ovo je prvi deo u kome se dodaju novi čvorovi u realizaciji.



Slika 3: *Konstrukcija optimalne realizacije trougla.*

3.4 Izbacivanje nepotrebnih ivica

Predzadnji korak algoritma izbacuje nepotrebne ivice. Ukoliko se izbacivanjem neke ivice dobija ista matrice rastojanja grafa, posmatrana ivica je nepotrebna. Njenim izbacivanjem dobijamo optimalniju realizaciju. Pokušajmo da nadjemo gornju granicu broja ivica u grafu realizacije. U radu [1] broj ivica je ograničen sa n^6 što je svakako pregruba granica. Naime, broj ivica se od početnog grafa povećava jedino u koraku izvacivanja "trouglova" koji predstavljaju put dužine dva. U tom slučaju broj ivica se povećava za 1. Kako u konstrukciji koja dodaje novu ivicu učestvuju dve ivice iz početnog grafa, dobijamo da broj ivica nakon izbacivanja trouglova ne može biti veći od $E + \frac{E}{2}$. Najgori mogući slučaj dobijamo za matricu rastojanja kompletnog grafa. Ovim se broj ivica ograničava sa n^2 što je za 4 reda veličine manje nego u radu [1].

Pri svakom izbacivanju ivice ponovna inicijalizacija matrice rastojanja je neophodna u celosti. U algoritmu je implementiran FloydWarshall-ov algoritam.

3.5 Nadogradnja

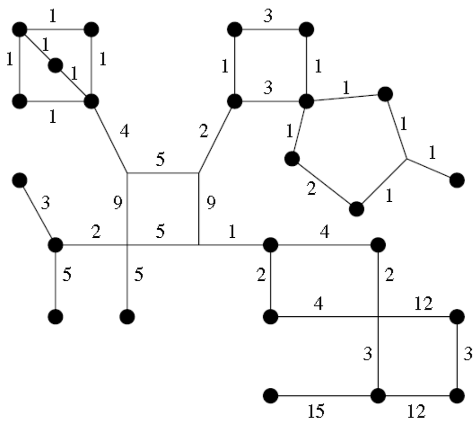
Ovaj korak predstavlja ponovno dodavanje listova dobijenih iz prvog dela - sažimanja. Dodavanje novog lista trenutnoj realizaciji je konstantne složenosti. Ukupna složenost ovog koraka je $O(n)$.

4 Testiranje

Nedostatak numeričkih podataka i korpusa za ovaj problem otežava uporedjivanje algoritma sa postojećim pristupima. Implementacija našeg algoritma je testirana na korpusu iz rada [1]. Dobijeni rezultati se poklapaju sa onima datim u radu (svi osim dva su optimalni). Za dati korpus su poznate optimalne vrednosti matrica rastojanja.

Matrica rastojanja	Težina realizacije	Optimalna realizacija
2pentagons	12	11
2squares	9	9
4triangles	46.5	46.5
cube2	14	14
h	5	5
hak64	11	11
imr64	132	128
polygon6	6	6
star4	4	4
star5	5	5
star6	6	6
triangle	4.5	4.5

Tabela 1. Numerički rezultati na korpusu matrica rastojanja iz rada [1]



Slika 4: *Optimalna realizacija imr64 primera matrice rastojanja.*

Uporedjivanjem vremena naše implementacije i implementacije u radu [1] dobija se ubrzanje za red veličine na korpusu iz rada kao i na slučajno generisanom korpusu.

Kao što je napomenuto, u svrhu testiranja kreiran je slučajan korpus matrica rastojanja. Matrice rastojanja su kreirane na sledeći način:

- Za date parametre $(nMax, p)$ kreiran je slučajan graf Erdos-Renyi metodom i izabrana najveća komponenta povezanosti dobijenog grafa. Označimo ovu komponentu sa G i broj čvorova sa n.
- Na slučajan način je izabrana veličina podgrafa $m \leq n$ a zatim m slučajnih čvorova iz grafa G.
- Matricu rastojanja dobijamo kao matricu rastojanja indukovanog podgrafa G', reda m, definisanog gornjim čvorovima.
- Rezultat našeg algoritma je uporedjivan sa težinom grafa G, koji predstavlja realizaciju dobijene matrice rastojanja (numerička uporedjivanja su prikazana u Tabeli 2).

Parametri $(nMax, p)$ Erdos-Renyi modela	Težina realizacije	Realizacija slučajnog grafa
$(50, 0.3)$	120	367
$(100, 0.4)$	79.5	1957
$(50, 0.1)$	113	122
$(100, 0.1)$	212.5	484
$(100, 0.05)$	329	**289**

Tabela 2. Numerički rezultati na slučajno generisanom korpusu matrica rastojanja

5 Zaključak

U radu je analiziran algoritam iz [1] i izneta su poboljšanja u svakom od koraka algoritma. Izvršeno je testiranje na novom slučajno generisanom korpusu za koje se znaju gornja granica optimalne realizacije i u većini primera dobijena je bolja realizacija. Testiranjem je zaključeno da heuristika dobro radi za sve vrednosti parametra osim iz segmenta $[0.025, 0.075]$.

Dalji rad bi mogao da obuhvati dodavanje novih koraka u algoritmu, pre svega u prvom koraku gde bi mogla da se vrši detekcija artikulacionih tačaka i mostova matrice rastojanja (čvorova koji su u svakoj optimalnoj realizaciji artikulacione tačke). Takodje moglo bi se pristupiti modifikaciji algoritma za socijalne mreže jer je i sama primena realizacije vezana za neke socijalne fenomene.

Reference

[1] S.C. Varone, A constructive algorithm for realizing a distance matrix, European Journal of Operational Research, 174 (2006), 102-111.

[2] S.L. Hakimi, S.S. Yau, Distance matrix of a graph and its realizability, Quarterly of Applied Mathematics, 22 (1964), 305/317.

[3] V. Bagatelj, T. Pisanski, An algorithm for tree-realizability of distance matrices, International Journal of Computer Mathematics, 34(1990), 171–176.

[4] I. Althfer, On optimal realization of finite metric spaces by graphs, Discrete and Computational Geometry, 3 (1988), 103/122.

[5] P. Winkler, The complexity of metric realisation, SIAM Journal on Discrete Mathematics, 1 (4) (1988), 552/559.

[6] J.M.S. Simoes-Pereira, A note on optimal and suboptimal digraph realizations of quasidistance matrices, SIAM Journal on Algebraic Discrete Methods, 5 (1984), 117/132.

[7] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, *Introduction to Algorithms*, The MIT Press (2001)

[8] Udi Manber, *Introduction to Algorithms*, Addison-Wesley, Boston (1989)

[9] Robert Sedgewick, *Algorithms*, Addison-Wesleys (1984)