ANALIZA RODOVNIKOV S PROGRAMOM PAJEK

Andrej Mrvar
Fakulteta za družbene vede
Univerza v Ljubljani

Vladimir Batagelj
FMF, Oddelek za matematiko
Univerza v Ljubljani

Povzetek

Z grafi lahko predstavimo veliko sistemov z zelo različnih področij. Eno od zanimivih področij, kjer se pojavlja večje število obsežnih grafov so rodovniki (genealogije).
Za izmenjavo rodoslovnih podatkov se uporablja oblika zapisa **GEDCOM***, ki jo prepozna tudi programski paket* **Pajek***. Rodoslovne podatke lahko predelamo v dve vrsti grafov: navadne in parne rodovnike.*
V prispevku so predstavljene uporabe programa **Pajek** *pri analizi rodovnikov, kot so iskanje najkrajše sorodstvene zveze med osebama, iskanje najbližjih prednikov in potomcev izbrane osebe, iskanje zanimivih vzorcev (porok med bližnjimi sorodniki), . . . Poseben poudarek je dan postopkom risanja rodovnikov.*

1. OBLIKA ZAPISA GEDCOM

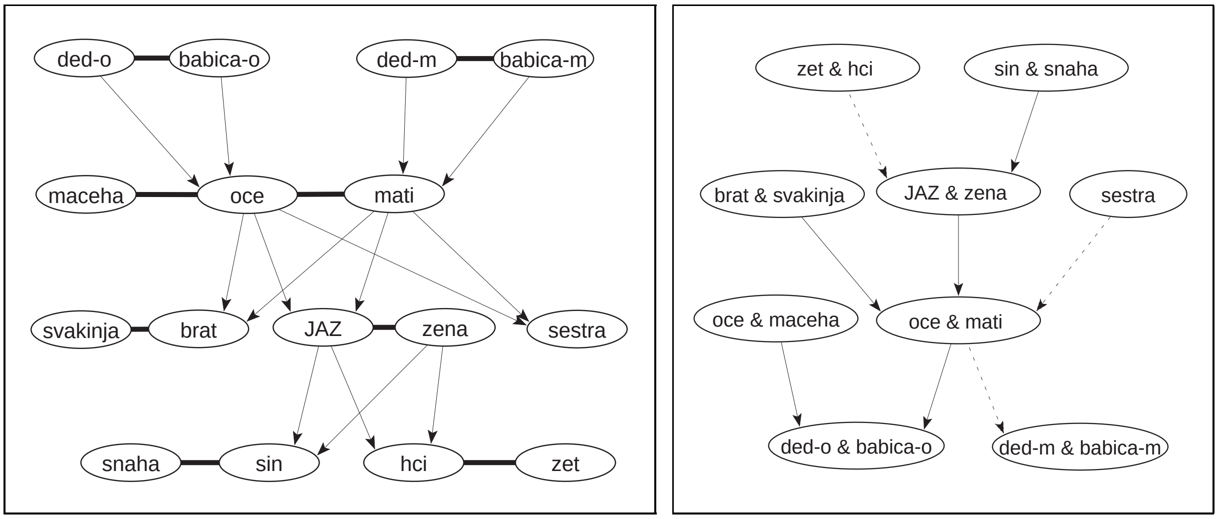
Eno od področij, kjer se pojavlja večje število obsežnih grafov, so rodovniki (genealogije). Veliko rodovnikov je že dostopnih v računalniški obliki [7]. Na omrežju so povezani v poizvedbeni sistem **GENDEX** [8], ki vključuje milijone oseb. Zelo dejavno je tudi slovensko rodoslovno društvo [11].

Za izmenjavo rodoslovnih podatkov se uporablja oblika zapisa **GEDCOM** [9]. V rodoslovju obstaja precej različnih programov, ki uporabljajo različne načine shranjevanja podatkov, zato so se dogovorili za skupno obliko **GEDCOM**, ki naj bi jo prepoznali vsi programi. V datoteki **GEDCOM** so zapisani podatki o osebah (ime, spol, rojstni podatki, poroke, dogodki, posnetki,. . .) in družinah (oče, mati in otroci). Za enolično identifikacijo oseb in družin so vpeljane posebne oznake, tako da ne pride do dvoumnosti.

2. RODOVNIKI

Obliko **GEDCOM** prepozna tudi program **Pajek** [10] in jo predela v graf. Podrobnosti o uporabi programa v rodoslovne namene se opisane v lanski decembrski številki rodoslovnega glasila *Drevesa* [4].

Pravzaprav lahko rodoslovne podatke predelamo v dve vrsti grafov – *rodovnikov*:



Slika 1: **Predstavitev rodovnika v navadni obliki (levo) in v parni obliki (desno)**.

V *navadni* obliki je vsaka oseba predstavljena s točko grafa, poroka je označena z neusmerjeno povezavo med točkama, usmerjene povezave pa vodijo od staršev do otrok. Usmerjeni podgraf je acikličen. Glej primer na levi strani slike 1.

V *parni* obliki (**`p-graph`**) so točke grafa lahko posamezniki ali pari. Več o tej obliki najdemo na predstavitveni strani dr. D. R. Whitea z Univerze v Kaliforniji (Irvine) [6]. Če neka oseba še ni poročena, je v grafu predstavljena s svojo točko; če pa je poročena, je v grafu predstavljena skupaj s svojim zakoncem s skupno točko. V tej obliki imamo samo usmerjene povezave, ki vodijo od otrok do staršev. Ker so točke lahko tudi pari, moramo posebej označiti ali se povezava nanaša na moža ali na ženo. Povezave, ki se nanašajo na moža (kažejo na moževe starše) so označene z neprekinjeno črto, povezave, ki se nanašajo na ženo (kažejo na ženine starše) pa s prekinjeno (pikčasto) črto. Če je neka oseba večkrat poročena, se pojavi v toliko točkah, kolikor je porok. Glej primer na desni strani slike 1.

V izvirni parni obliki kažejo puščice od otrok proti staršem. Za razumevanje je morda bolje obrniti smeri puščic, zato je v nadaljnjih primerih uporabljena parna oblika z obrnjenimi puščicami.

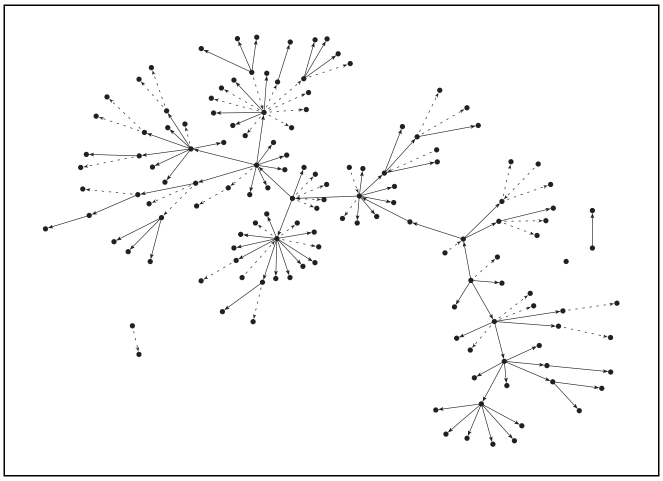
2.1. Primerjava navadne in parne oblike rodovnika

V nadaljevanju se bo pokazalo, da je parna oblika v več pogledih ugodnejša od navadne oblike, čeprav je (vsaj v začetku) težja za razumevanje. Dve prednosti opazimo takoj:

Ker določena točka lahko predstavlja dve osebi (zakonca), je število točk v parni obliki majše kot v navadni obliki; prav tako je v grafu manj povezav.

V parni obliki nimamo več neusmerjenih povezav, graf je acikličen. Za aciklične grafe obstaja kar nekaj algoritmov, ki so dovolj hitri tudi pri zelo velikih grafih (npr. razbitje po globinah, metoda kritične poti in še nekateri, ki bodo opisani v nadaljevanju).

Rodovnik je *pravilen*, če ima vsakdo največ enega očeta in eno mater. Rodovniki so *redki* grafi, saj je število povezav linearno povezano s številom točk. To lahko pokažemo z nekaj neenakostmi:



Slika 2: **Parni rodovnik narisan s Fruchterman-Reingold-ovim modelom**.

V *pravilnem navadnem* rodovniku velja za usmerjeni del:

$$|A| = \sum_{v \in V} d_{in}(v) \leq 2|V|$$

kjer so A usmerjene povezave, V točke, $d_{in}(v)$ pa vhodna stopnja točke v (število v točko vstopajočih povezav). Zgornja neenakost izhaja iz neenakosti $d_{in}(v) \leq 2$.

Ker je večina poročena le enkrat, nekateri pa nikoli, velja praviloma za neusmerjeni del $|E| \leq \frac{1}{2}|V|$, kjer so E neusmerjene povezave. Torej je število vseh povezav v navadnem rodovniku $|L| = |A| + |E| \leq \frac{5}{2}|V|$. V dejanskih rodovnikih $|L|$ le malo presega $|V|$.

Parni rodovniki so skoraj drevesa – odstopanja od drevesa pomenijo poroke med sorodniki. Če z $|V_p|$ označimo število točk v parnih grafih, z n_{vec} pa število večkratnih porok, dobimo

$$|V_p| \approx (|V| - 2|E| + n_{vec}) + |E| = |V| - |E| + n_{vec}$$

Člen v oklepaju je ocena števila samskih, drugi člen pa števila parov. Torej $|V_p| \geq |V| - |E|$. Velja tudi

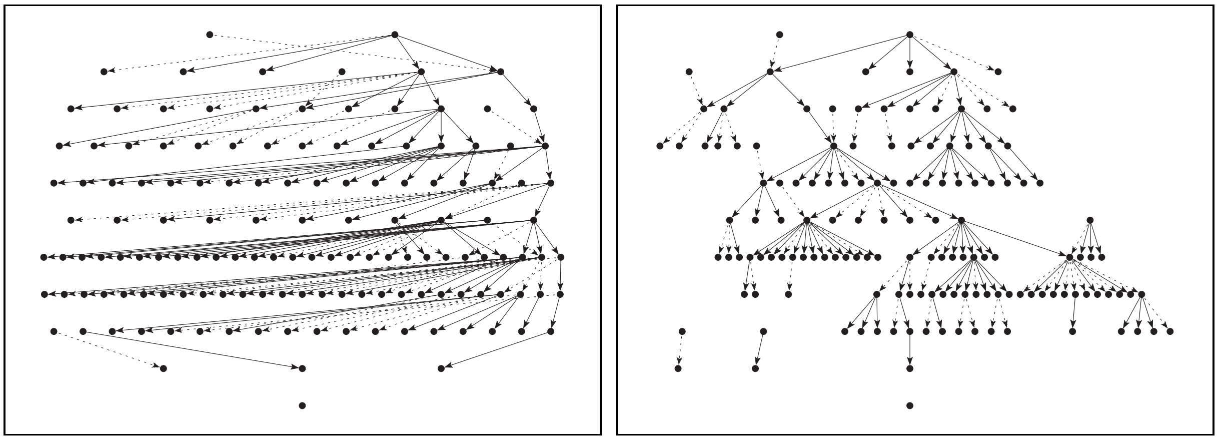
$$|A_p| = \sum_{v \in V_p} d_{out}(v) \leq 2|V_p|$$

$d_{out}(v)$ je izhodna stopnja točke v (število iz točke izstopajočih povezav).

3. RISANJE RODOVNIKA

Za risanje rodovnikov se je smiselno odločiti, če graf ni prevelik (največ 200 do 300 točk). Sicer je pred tem smiselno opraviti še kakšno analizo in na njeni osnovi izločiti in narisati samo izbrani podgraf.

Za risanje grafa lahko uporabimo kakega od splošnih postopkov – na primer energijska risanja (Kamada-Kawai [3] ali Fruchterman-Reingold [2]), vendar za to vrsto grafov ne dobimo preveč dobrih rezultatov. Vzemimo za primer rodovnik Ivana Cankarja. V njem se nahaja 179 posameznikov. Če rodovnik preberemo v parni obliki, dobimo graf s 140 točkami (89 posameznikov



Slika 3: **Rodovnik narisan po nivojih pred in po optimizaciji**.

in 51 parov) in 136 usmerjenimi povezavami. Slika 2 ga prikazuje narisanega s Fruchterman-Reingoldovim postopkom. Namen slike je prikazati samo rezultat postopka (razmestitev točk), zato točke niso označene.

Točke na sliki so sicer dokaj enakomerno razporejene po ravnini. Slaba stran slike pa je, da iz nje zelo težko razberemo generacije. Za risanje rodovnikov zato potrebujemo posebne algoritme.

3.1. Risanje rodovnikov po nivojih

Iz slike rodovnika bi radi razbrali generacije. Zato je prvi korak določitev generacij, ki jih predstavimo z nivoji, npr. vsi otroci neke družine naj bodo narisani na isti višini – koordinati y. Kot smo že omenili je rodovnik, če ga preberemo v parni obliki, acikličen graf, zato za določitev nivojev lahko uporabimo globinsko razbitje. Izkaže pa se, da to razbitje v primeru rodovnika ne da željenega rezultata. Tipičen primer, kjer je rezultat drugačen od željenega je npr.:

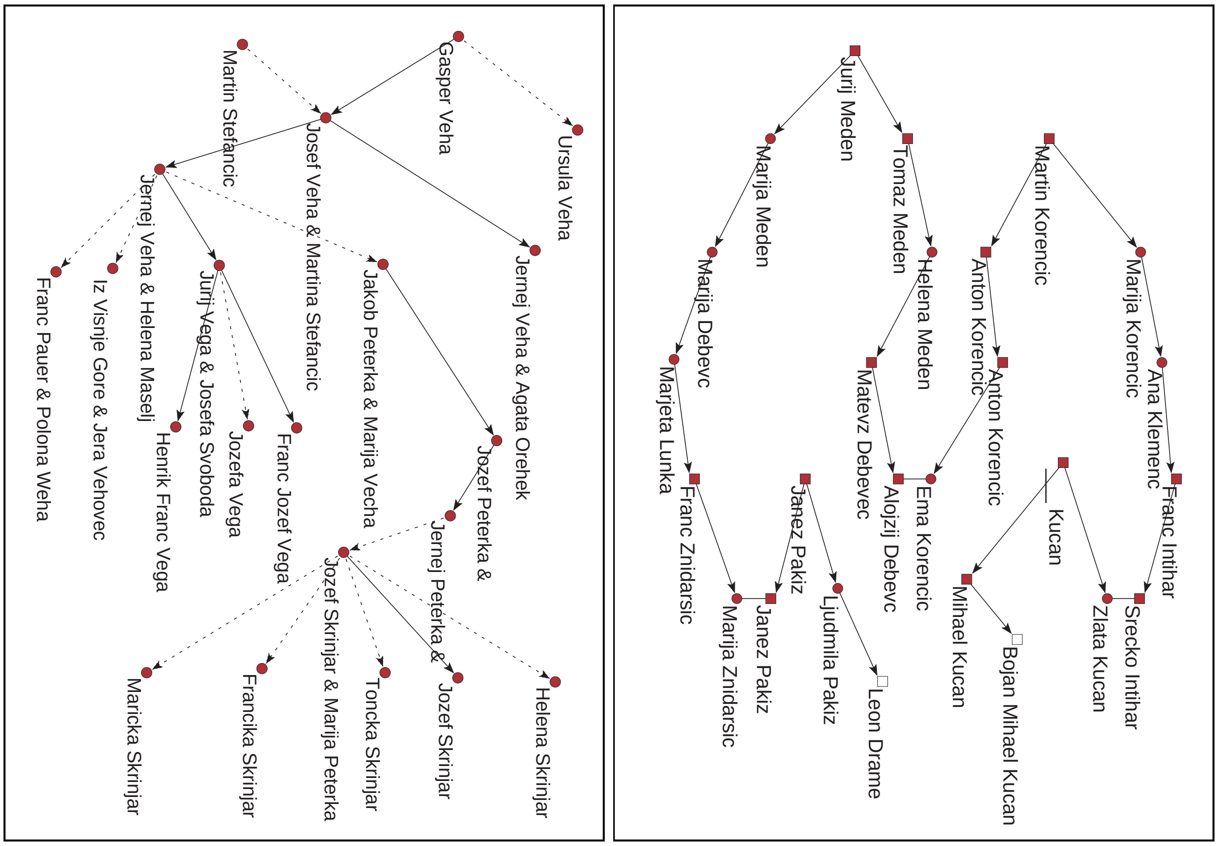
Nekje v rodovniku se pojavi oseba, za katero poznamo oba starša, sama oseba pa nima potomcev (v parni obliki to pomeni, da v točko ne vodi nobena povezava), zato se tej osebi dodeli najnižji nivo, ne glede na to na katerem nivoju so njeni starši (le ti so lahko na precej višjem nivoju). Na sliki grafa se povezava od staršev do tega otroka odrazi kot dolga povezava med nesosednimi nivoji, ki povzroči tudi veliko križanj povezav.

Pri risanju rodovnika bomo skušali čimbolj zadostiti zahtevi:

> Če za neko osebo poznamo vsaj enega od staršev, naj ima oseba nivo, ki je praviloma za ena nižji od nivoja staršev.

Postopek določanja nivojev je naslonjen na metodo kritične poti. Takšna določitev nivojev nam da v večini primerov dobre rezultate. Seveda pa lahko pride do povezav med nesosednjimi nivoji (*generacijski preskoki* – npr. vnuk se poroči prej kot eden od otrok).

Leva stran slike 3 prikazuje parni rodovnik Ivana Cankarja narisan na opisani način. Predno smo točke narisali po nivojih, smo jih še oštevilčili glede na sprehod v globino, s čimer se je precej zmanjšalo število križanj povezav. Vidi pa se, da bi se dalo dobljeno sliko še precej izboljšati. Motijo predvsem dolge povezave, ki so težko berljive in poleg tega povzročajo nepotrebna križanja.



Slika 4: **Parni rodovnik Jurija Vege (levo), najkrajša pot v `Drame.ged` (desno).**

3.1.1. Določanje začetne slike rodovnika

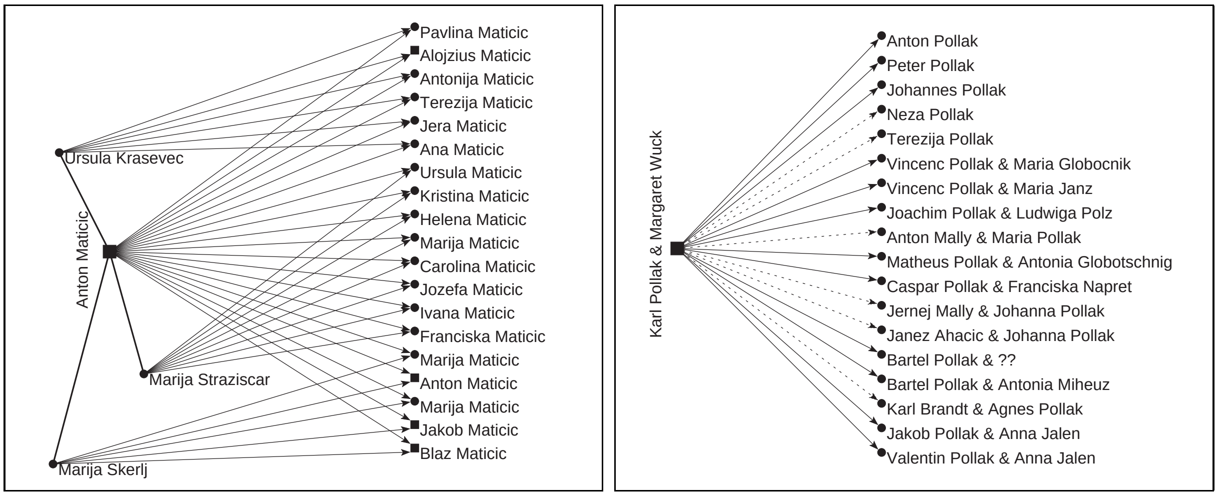
Na levi strani slike 3 so točke razporejene po nivojih glede na oštevilčenje, ki ga vrne sprehod v globino. Do začetne slike pa lahko pridemo tudi na drugačen način. Pomagamo si lahko z idejo *Richardsovega oštevilčenja – Negopy* [5, str. 165]. Rezultat tega algoritma je taka razporeditev točk po nivojih, da imajo točke, ki so povezane med sabo, čimbolj enake koordinate x. Postopek: Začnemo z neko razporeditvijo točk po nivojih. Za vsako točko izračunamo povprečno koordinato x njenih sosedov. Povprečje postane nova koordinata x dane točke. Postopek ponavljamo dokler se koordinate ne ustalijo. Po koncu postopka ima lahko več točk na nivoju enako (ali zelo podobno) koordinato x, zato točke še razmaknemo na željeno minimalno razdaljo.

Idejo lahko posplošimo tudi na risanje v prostoru – nivoji so v tem primeru ravnine na različnih višinah – koordinata z.

3.1.2. Optimizacija skupne dolžine povezav v rodovniku

Z dobljeno sliko na levi strani slike 3 še nismo bili zadovoljni. Slika se precej izboljša, če dobljeno sliko optimiziramo glede na skupno dolžino povezav s premeščanjem točk na istih nivojih.

Za optimizacijo uporabimo *postopek lokalne optimizacije*, kjer je kriterij skupna dolžina povezav



Slika 5: **Posameznik s skupno največ zakonskimi partnerji in otroki (navadna oblika, levo) in zakonski par z veliko (vendar ne največ) otroki (parna oblika, desno)**.

– točke premeščamo toliko časa, da je skupna dolžina povezav čimmanjša. Desna stran slike 3 prikazuje rezultat omenjene optimizacije, če za začetno razporeditev uporabimo tisto z leve strani slike.

Na levi strani slike 4 je na ta način narisan še parni rodovnik Jurija Vege.

4. ANALIZE RODOVNIKOV

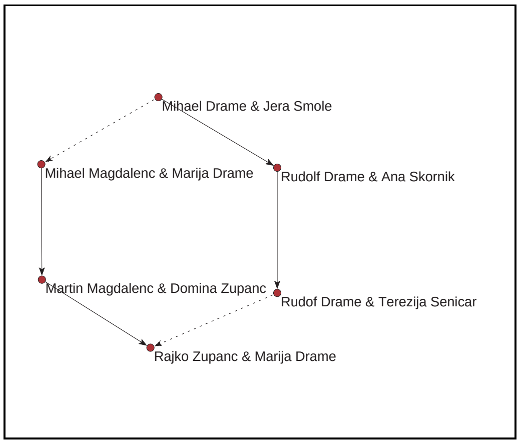
4.1. Iskanje najkrajše sorodstvene zveze med dvema osebama

Zanimivo vprašanje, ki si ga lahko zastavimo nad rodovnikom je tudi, ali sta si dve osebi v sorodu. Če sta si v sorodu, bi seveda radi našli najkrajšo sorodstveno zvezo med njima. Za to lahko uporabimo Dijkstrov algoritem [1] za iskanje najkrajše poti, le da moramo celoten postopek izpeljati v treh korakih: Najprej pretvorimo vse usmerjene povezave v neusmerjene, s čimer dovolimo algoritmu, da išče tudi v smeri nasprotni smeri puščic. Če tega ne bi storili, algoritem npr. ne bi našel sorodstvene vezi med bratoma. V tako dobljenem grafu poiščemo najkrajšo pot med točkama. Iz osnovnega (usmerjenega) grafa izločimo podgraf, ki je določen z množico točk, ki ležijo na najkrajši poti in pripadajočimi povezavami.

Za primer smo uporabili podatke z datoteke **`Drame.ged`** (rodovnik Leona Drameta), v kateri se nahajajo podatki o 29606 osebah. Datoteka je sestavljena iz 703 ločenih (šibkih) komponent, od katerih je ena zelo velika (z 19854 osebami), ostale pa so manjše. Če datoteko preberemo v parni obliki, dobimo graf z 22193 točkami (13254 posameznikov in 8939 parov) in 21862 povezavami. V navadni obliki pa imamo 29606 točk, 8256 neusmerjenih povezav (za nekatere pare nista poznana oba zakonca) in 41814 usmerjenih povezav.

Desna stran slike 4 prikazuje najkrajšo pot med Leonom Drametom in Bojanom Mihaelom Kučanom, po navadnem rodovniku.

št. točk v komp.	št. komp.
2	19474
3	1
4	11
6	3
7	1
11	1
12	1
144	1
1602	1



Slika 6: **Porazdelitev števila točk v dvopovezanih komponentah (levo) in primer poroke med sorodniki (desno)**.

4.2. Iskanje okolice izbrane osebe

Če imamo opravka z velikim rodovnikom, se je pred risanjem smiselno omejiti na neko okolico izbrane osebe. Zato se odločimo koliko generacij prednikov in naslednikov želimo videti in le-te izločimo iz rodovnika, ter po potrebi narišemo na opisani način.

4.3. Število neposrednih potomcev izbrane osebe

Tudi enostavno razbitje po stopnjah nam da v primeru rodovnikov zanimiv rezultat, če ga kombiniramo še z nekaterimi drugimi operacijami.

Vzemimo spet podatke **`Drame.ged`**. Če preberemo datoteko v navadi obliki, dobimo kot rezultat osebo z največ zakonskimi partnerji in otroki (skupno glede na vse poroke), kar je prikazano na levi strani slike 5.

Če datoteko preberemo v parni obliki, lahko dobimo kot rezultat zakonski par z največ otroki, vendar to vedno ne drži (kot ne drži npr. v primeru na sliki 5, desno). Problem se pojavi zato, ker v parni obliki štejemo vsakega otroka enkrat, če ni poročen, sicer pa tolikokrat kolikorkrat je poročen.

4.4. Iskanje zanimivih vzorcev

S programom **`Pajek`** lahko odkrijemo tudi zanimive vzorce v rodovnikih (poroke med sorodniki). Rodovnik najprej preberemo v parni obliki. V njem določimo dvopovezane komponente. Če se v rezultatu pojavijo samo komponente velikosti 2, je vse v redu. Večje komponente pa so bolj zanimive. Tabela na levi strani slike 6 prikazuje porazdelitev velikosti komponent v primeru podatkov **`Drame.ged`**.

Obhodi (smeri povezav zanemarimo) v dvopovezanih komponentah pomenijo poroke med sorodniki; lihi obhodi pomenijo tudi generacijske preskoke. Posebno 'nevarni' so kratki obhodi – poročajo se bližnji sorodniki. Desna stran slike 6 prikazuje enega od zanimivih vzorcev v datoteki **`Drame.ged`** (sodi obhod dolžine 6).

V tabeli na sliki 6 sicer lahko preberemo, da so v datoteki tri dvopovezane komponente velikosti šest, vendar to še ne pomeni, da v grafu ni mogoče najti še več šestkotnikov. Dvopovezane komponente namreč ne odkrijejo tistih, ki so 'vpeti' v večje komponente. S programom **`Pajek`** lahko v omenjeni datoteki odkrijemo 14 šestkotnikov, od katerih se dva ujemata tudi po spolu z vzorcem na sliki 6.

5. ZAKLJUČEK

Postopke analize rodovnikov še razvijamo in sproti vključujemo v program **`Pajek`**. Tekoče stanje in povezave na vire podatkov ter druge rodoslovne informacije lahko najdete na predstavitveni strani programa **`Pajek`** [10].

Viri

[1] DIJKSTRA, E. (1968): Cooperating Sequential Processes. In: F. Genuys (Editor), *Programming Languages*, New York: Academic Press.

[2] FRUCHTERMAN, T. M. J., REINGOLD, E. M. (1991): Graph Drawing by Force-Directed Placement. *Software, Practice and Experience* **21**, 1129-1164.

[3] KAMADA, T., KAWAI, S. (1989): An Algorithm for Drawing General Undirected graphs. *Inf. Proc. Letters* **31**, 7-15.

[4] MRVAR A., BATAGELJ V. (1997): *Pajek – program za analizo obsežnih omrežij. Uporaba v rodoslovju*. Drevesa. Bilten slovenskega rodoslovnega društva. Letnik 4, številka **12**, december 1997, 4-6.

[5] ROGERS, E. M., KINCAID, D. L. (1981): *Communication Networks, Toward a New Paradigm for Research*. The Free Press, New York.

[6] Parni grafi
`http://eclectic.ss.uci.edu/~drwhite/pgraph/p-graphs.html`

[7] Primeri rodovnikov
`http://vlado.fmf.uni-lj.si/pub/networks/doc/genealog.htm`

[8] GENDEX – WWW Genealogical Index
`http://www.gendex.com/gendex/`

[9] Standard GEDCOM
`http://www.gendex.com/gedcom55/55gcint.htm`

[10] Programski paket Pajek
`http://vlado.fmf.uni-lj.si/pub/networks/pajek/`

[11] Slovensko rodoslovno društvo
`http://genealogy.ijp.si/slovrd/rd.htm`