ANALIZA VELIKIH OMREŽIJ S PROGRAMOM PAJEK

Vladimir Batagelj
FMF, Oddelek za matematiko, Univerza v Ljubljani
Andrej Mrvar
Fakulteta za družbene vede, Univerza v Ljubljani

Povzetek

Pri svojem delu se večkrat srečamo z velikimi omrežji, kjer gre število točk in povezav v tisoče, npr.: rodovniki, diagrami poteka programskih sistemov, organske molekule, računalniška omrežja, transportna omrežja, vodovodna in električna omrežja, referenčna omrežja, družboslovna omrežja, itd. Obvladovanje velikih omrežij pomeni tako časovno, kot tudi prostorsko zahteven problem. Večina standardnih algoritmov za analizo omrežij ima visoke časovne zahtevnosti in so zato neprimerni za analizo velikih omrežij. V sestavku so predstavljeni pristopi k analizi in predstavitvam tovrstnih omrežij. Pristopi so podprti s programom Pajek. Prikazanih je tudi nekaj tipičnih primerov uporabe.

Abstract

Large networks, having thousands of vertices and lines, can be found in many different areas, e. g: genealogies, flow graphs of programs, molecules, computer networks, transportation networks, social networks, intra/inter organisational networks... Many standard network algorithms are very time and space consuming and therefore unsuitable for analysis of such networks. In the article we present some approaches to analysis and visualisation of large networks implemented in the program Pajek. Some typical examples are also given.

■■■

1. Uvod

Pajek je programski paket, za okolje Windows (32 bit), ki omogoča analizo in prikaz velikih omrežij (omrežij z več tisoč točkami). Program je prosto dostopen na naslovu:

`http://vlado.fmf.uni-lj.si/pub/networks/pajek/`

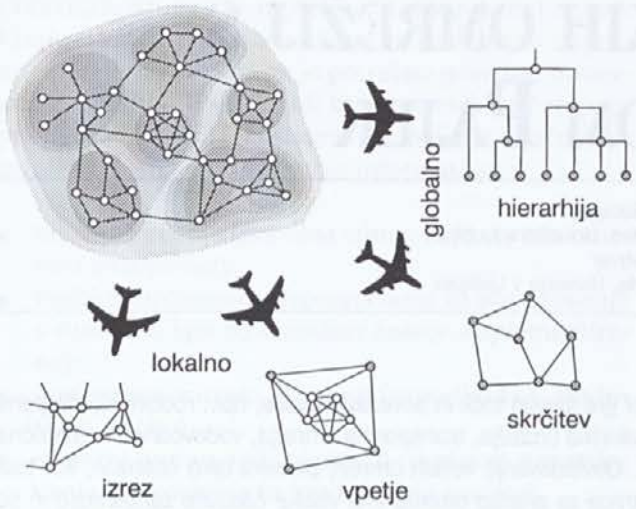
Velika omrežja lahko najdemo na številnih področjih. Največkrat pridemo do njih avtomatično, z uporabo računalnikov, iz različnih podatkovnih virov, ki so že v elektronski obliki. Primeri:

- veliki rodovniki (rodovniki z nekaj 10.000 [25] osebami), omrežje mentorstev pri izdelavi doktorskih disertacij s področja teoretičnega računalništva (1.882 oseb [36]);
- omrežja, dobljena iz slovarjev in drugih besedil (povezave med 52.652 angleškimi besedami glede na zameno/vrivanje/brisanje posameznih črk [24]);
- transportna omrežja (povezave med 332 ameriškimi letališči [37]);
- velike molekule (molekule z nekaj tisoč atomi, npr. DNA [31]);
- komunikacijska omrežja: povezave med stranmi ali strežniki na Internetu, uporaba debatnih skupin (Usenet) [34], telefonski klici [22];
- diagrami poteka programskih sistemov [16];
- bibliografije, referenčna omrežja [9,7], omrežje Erdösevih soavtorjev (5.822 soavtorjev [20]), ...

Velikih omrežij ne moremo učinkovito analizirati z uporabo standardnih programov za analizo omrežij, ki povečini temeljijo na matrični predstavitvi omrežja in so zato omejeni na omrežja z nekaj deset ali sto točkami.

Glavni cilji pri zasnovi programa **Pajek** so bili:

- podpreti abstrakcijo s postopno razčlenitvijo velikega omrežja na več manjših omrežij, ki jih lahko nadalje analiziramo z uporabo običajnih metod;
- ponuditi uporabniku močna orodja za prikaz omrežij;
- vgraditi večje število učinkovitih algoritmov za analizo velikih omrežij.



Slika 1: Cilji pri zasnovi programa **Pajek.**

Kot je prikazano na sliki 1, lahko abstrakcijo podpremo na naslednje načine: poiščemo skupine (komponente, soseščine 'osrednjih' točk, jedra, ...) v omrežju; izrežemo in ločeno prikažemo točke, ki pripadajo posameznim skupinam (podroben lokalni pogled); skrčimo skupine v točke in prikažemo povezave med njimi (globalni pogled).

2. Učinkoviti algoritmi za analizo velikih omrežij

Časovna T(n) in prostorska *S(n)* zahtevnost algoritma nam povesta, koliko časa in prostora potrebujemo za njegovo izvedbo na nalogah velikosti *n* (v našem primeru — število točk ali povezav v omrežju). V večini velikih omrežij je število povezav m istega velikostnega reda kot število točk — $O(n)$ ali največ $O(n \log n)$. Taka omrežja imenujemo redka omrežja. V nadaljevanju bomo zato predpostavili, da analiziramo velika a redka omrežja.

Glede na vse večje pomnilniške zmogljivosti današnjih računalnikov prostorska zahtevnost za redka omrežja ni več kritična. Problem rešimo z ustreznimi podatkovnimi strukturami za notranjo predstavitev omrežja. V programu Pajek je bila uporabljena predstavitev omrežja z dvojno povezanimi seznami.

Časovna zahtevnost pa ostaja še vedno velik problem, saj tudi veliko hitrejši računalniki ne pomagajo veliko. V teoriji algoritmov veljajo problemi z algoritmi s polinomsko časovno zahtevnostjo za pohlevne — lahko rešljive. Toda, v primeru zelo velikih n, so lahko dejansko prezahtevni že algoritmi s časovno zahtevnostjo $O(n^2)$, kar je razvidno iz tabele 1.
Zato ima večina algoritmov, ki so vključeni v program Pajek, podkvadratične časovne zahtevnosti: $O(n)$, $O(n \log n)$, $O(n \sqrt{n})$, ali pa je njihova uporaba omejena samo na manjše množice izbranih točk.

3. Podatkovne strukture

Izvedbe algoritmov v programu Pajek so trenutno oprte na šest podatkovnih struktur:

- omrežje — glavna struktura (točke in povezave);
- permutacija — preureditev točk;
- vektor — vrednosti (lastnosti) točk;
- skupina — podmnožica točk (npr. en razred iz razbitja);
- razbitje — pove za vsako točko, v katero skupino spada;
- hierarhija — hierarhična razvrstitev skupin in točk omrežja.

Prava moč celotnega sistema Pajek se skriva v številnih prehodih med temi strukturami.

Poleg svojih vhodnih formatov podpira Pajek še več drugih formatov: UCINET DL [32]; Vega [33]; GED [21], rodoslovne podatke lahko predelamo v dve vrsti rodovnikov: navadne in parne rodovnike [25, 17, 18, 12]; in nekaj kemijskih formatov: BS (Ball and Stick), MAC (Mac Molecule) in MOL (MDL MOLfile).

Prav tako lahko omrežja shranjujemo v internih in različnih drugih izhodnih formatih (UCINET DL, Vega, BS in MOL). Eno od možnih uporab programa Pajek je torej tudi pretvorba med različnimi formati za opis omrežja.

4. Algoritmi

V tekoči različici so v program Pajek vključeni naslednji učinkoviti algoritmi [1, 24, 14, 15];

- *razbitja:* po stopnjah, po globinah, jedra, p-klike, centri;

Tabela 1: Časovne zahtevnosti algoritmov (Pentium/64M/90MHz).

	T(n)	1.000	10.000	100,000	1,000.000
Shuffle	$O(n)$	0,00 s	0,015 s	0,17 s	2,22 s
Quick Sort	$O(n \log n)$	0,00 s	0,00 s	0,40 s	5,14 s
Heap Sort	$O(n \log n)$	0,00 s	0,06 s	0,98 s	14,35 s
Insertion Sort	$O(n^2)$	0,07 s	7,50 s	12,5 min	20,83 h
XY	$O(n^3)$	0,10 s	1,67 min	1,16 dni	3,17 let

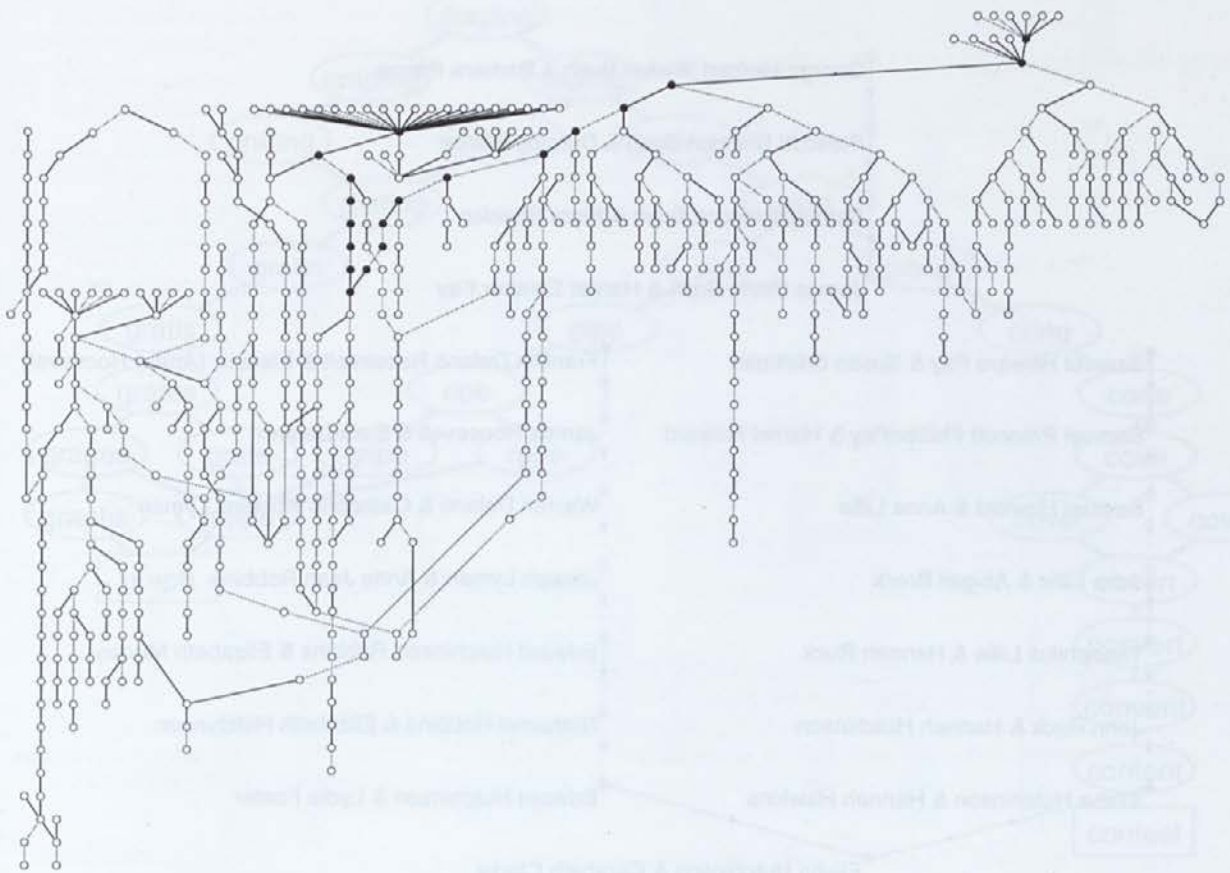
- *dvojiške operacije:* unija, presek, razlika;
- *komponente:* krepke, šibke, dvopovezane, simetrične [1];
- *dekompozicije:* simetrično-aciklična;
- *poti:* najkrajše poti, vse poti med izbranima točkama [11];
- *pretoki:* največji pretok med izbranima točkama [15];
- *glavne poti:* metoda Paths Count in metoda SPLC [2];
- *soseščine:* k-sosedi;
- *CPM* (metoda kritične poti);
- *izrez* pod-omrežja;
- *skrčitve* skupin v omrežju (posplošeni bločni modeli) [4];
- *preurejanja*: topološko urejanje, Richardsovo oštevilčenje, oštevilčenje glede na iskanje v globino oz. širino;
- *kleščenje:* dreves, vmesnih točk, po stopnjah;
- *poenostavitve in transformacije:* brisanje zank in večkratnih povezav, razusmerjanje ...;

Pogosto uporabljena zaporedja osnovnih operacij lahko definiramo kot makro, ki ga poženemo kot en sam ukaz. Z uporabo makrojev lahko sistem prilagodimo posebnim skupinam uporabnikov (analiza rodovnikov, kemijske uporabe, ...).

V program Pajek so vključeni tudi nekateri algoritmi namenjeni reševanju posebnih problemov: npr. algoritmi za preverjanje ali je program napisan po pravilih strukturiranega programiranja [16]; simulacija Petrijevih mrež [13]; iskanje zanimivih vzorcev v molekulah ali rodovnikih.

Poseben poudarek je dan avtomatičnemu določanju prikazov omrežij [35]. V sistem so vključeni številni tovrstni algoritmi: energijska risanja (Kamada-Kawai [10] in Fruchterman-Reingold [6]), risanja z uporabo lastnih vektorjev (Lanczosev algoritem [3, 5]), nivojska risanja (rodovnikov in drugih acikličnih struktur).

Ti algoritmi so bili izpopolnjeni in prilagojeni, tako da omogočajo dodatne učinke, npr.: risanja z omejitvami (optimizacija le izbranega dela omrežja, določitev izbranih točk za nepremične, uporaba podobnosti in različnosti oziroma vrednosti na povezavah), prostorski prikazi. Pajek nudi tudi orodja za ročno risanje omrežij.



Slika 2: Rodovnik dveh ameriških predsednikov.

Dobljene prikaze lahko pretvorimo v številne izhodne formate, ki si jih lahko nadalje ogledujemo s posebnimi pregledovalniki za ravninske in prostorske prikaze: (Encapsulated PostScript — GSView [28]; VRML — CosmoPlayer [27]; MDLMOL — Rasmol [31], Chime [26]; Kinemages — Mage [29]).

5. Primeri

Na sliki 2 je prikazana največja povezana komponenta (parni graf) v rodovniku ameriških predsednikov [19]. Najkrajša sorodstvena vez med Franklinom D. Roosveltom in Georgeom H. W. Bushem, dobljena z uporabo makroja Path, je narisana na sliki 3.

Slika 4 prikazuje najkrajše poti med besedama graph in drawing ter drawing in contest.

Slika 5 prikazuje prvonagrajeni graf s tekmovanja v risanju grafov Graph Drawing Contest 1998 v Motrealu [23].

Na sliki 6 je prikazan posnetek pogleda na prostorski prikaz grafa v prikazovalniku CosmoPlayer [27].

6. Nadaljnji razvoj

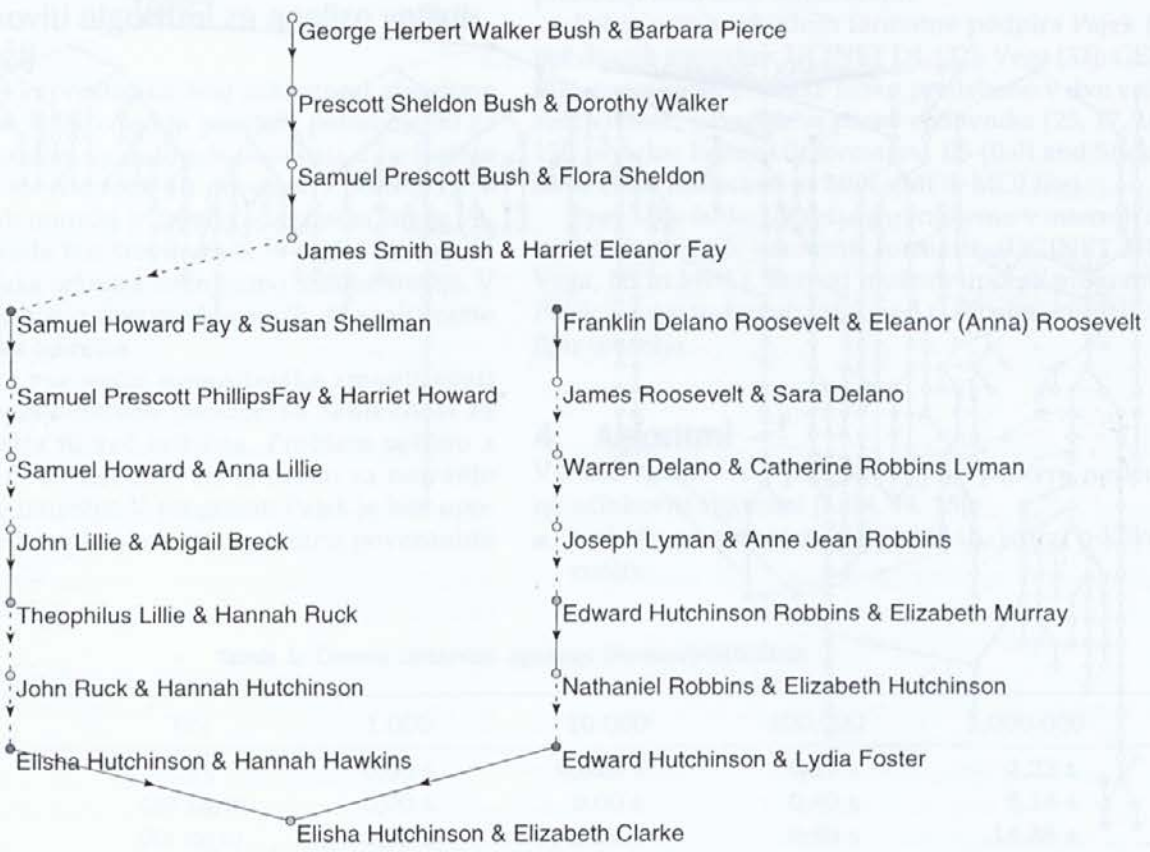
Program Pajek je še v razvoju. Zadnja izdaja je vedno na voljo na njegovi predstavitveni strani. V bližnji prihodnosti nameravamo v sistem vključiti naslednje algoritme: različne metode združevanj in dekompozicij, nekatere statistike (štetja triad), animacije in predstavitve zaporedja omrežij, uvedba krmilnih stavkov v sistemu makrojev, ravninska risanja ...

Literatura

[1] AHO AHO, A. V., HOPCROFT, J. E., ULLMAN, J. D. (1976): The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading, MA.

[2] BATAGELJ, V. (1994): An Efficient Algorithm for Citation Networks Analysis. Presented at EASST'94, Budapest, Hungary, August 28-31, 1994.

[3] DATTA, B. N. (1995): Numerical Linear Algebra and Applications. BrooksĐ&Cole Publishing Company, Pacific Grove.



Slika 3: Najkrajša pot med Franklinom D. Roosveltom in Georgeom H. W. Bushem.

[4] DOREIAN, P., BATAGELJ, V., FERLIGOJ, A. (1994):
Partitioning Networks Based on Generalized Concepts of Equivalence.
Journal of Mathematical Sociology, šÐbf 19, 1, 1-27.

[5] GOLUB, G. H., van LOAN, C. F. (1996):
Matrix Computations. The John Hopkins University Press, Baltimore.

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

[7] HUMMON, N. P., DOREIAN, P. (1989):
Connectivity in a Citation Network: The Development of DNA Theory.
Social Networks, Ðtextbfš11, 39-63.

[8] HUMMON, N. P., DOREIAN, P. (1990):
Computational Methods for Social Network Analysis.
Social Networks, Ðtextbfš12, 273-288.

[9] HUMMON, N. P., DOREIAN, P., FREEMAN, L. C. (1990):
Analyzing the Structure of the Centrality-Productivity Literature Created Between 1948 and 1979.
Knowledge: Creation, Diffusion, Utilization, Vol. Ðtextbfš11, June, No. 4, 459-480.

[10] KAMADA, T., KAWAI, S. (1989):
An Algorithm for Drawing General Undirected Graphs.
Information Processing Letters, šÐbf 31, 7-15.

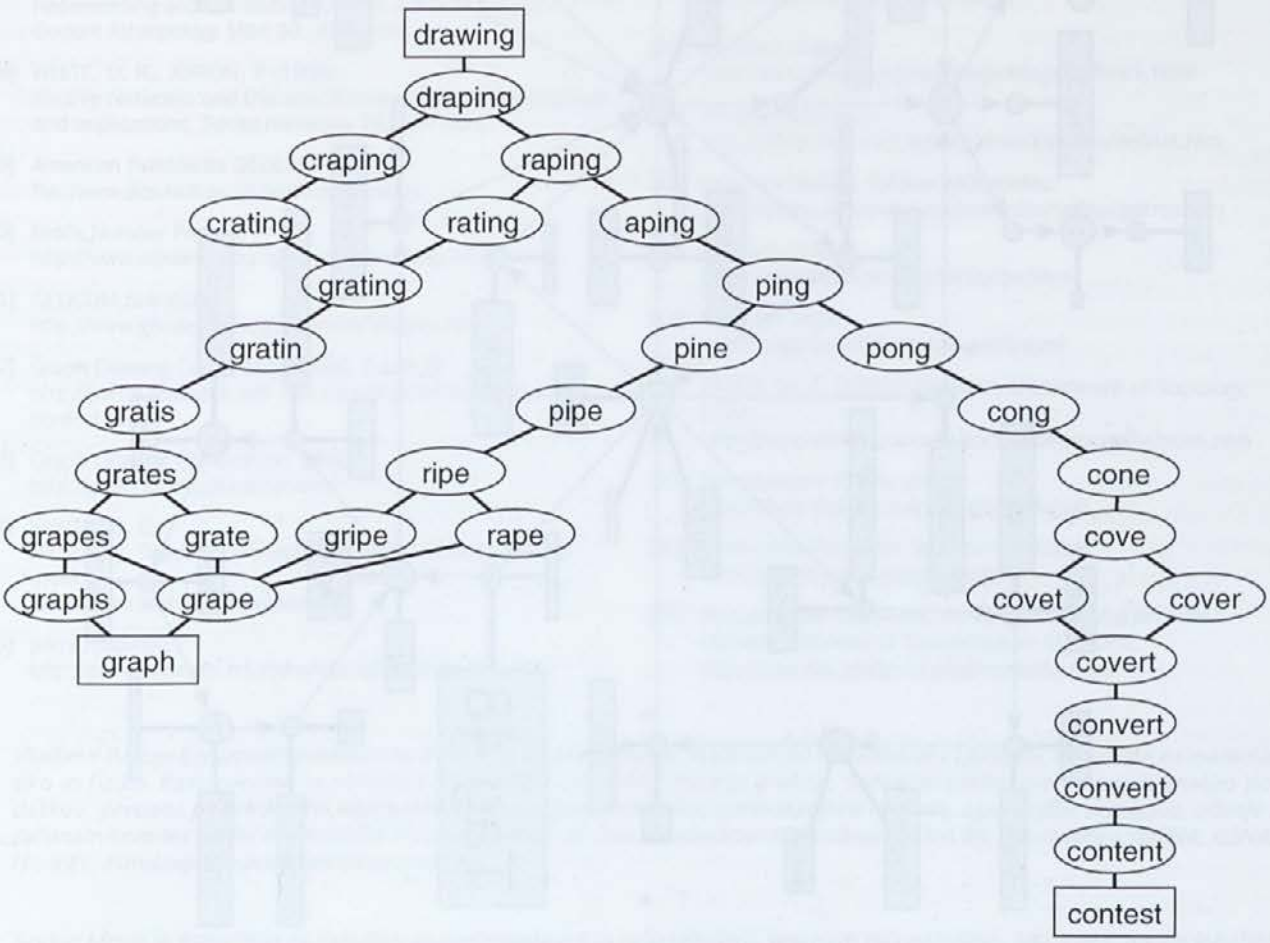
[11] KNUTH, D. E. (1993):
The Stanford GraphBase.
Stanford University, ACM Press, New York.

[12] 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.

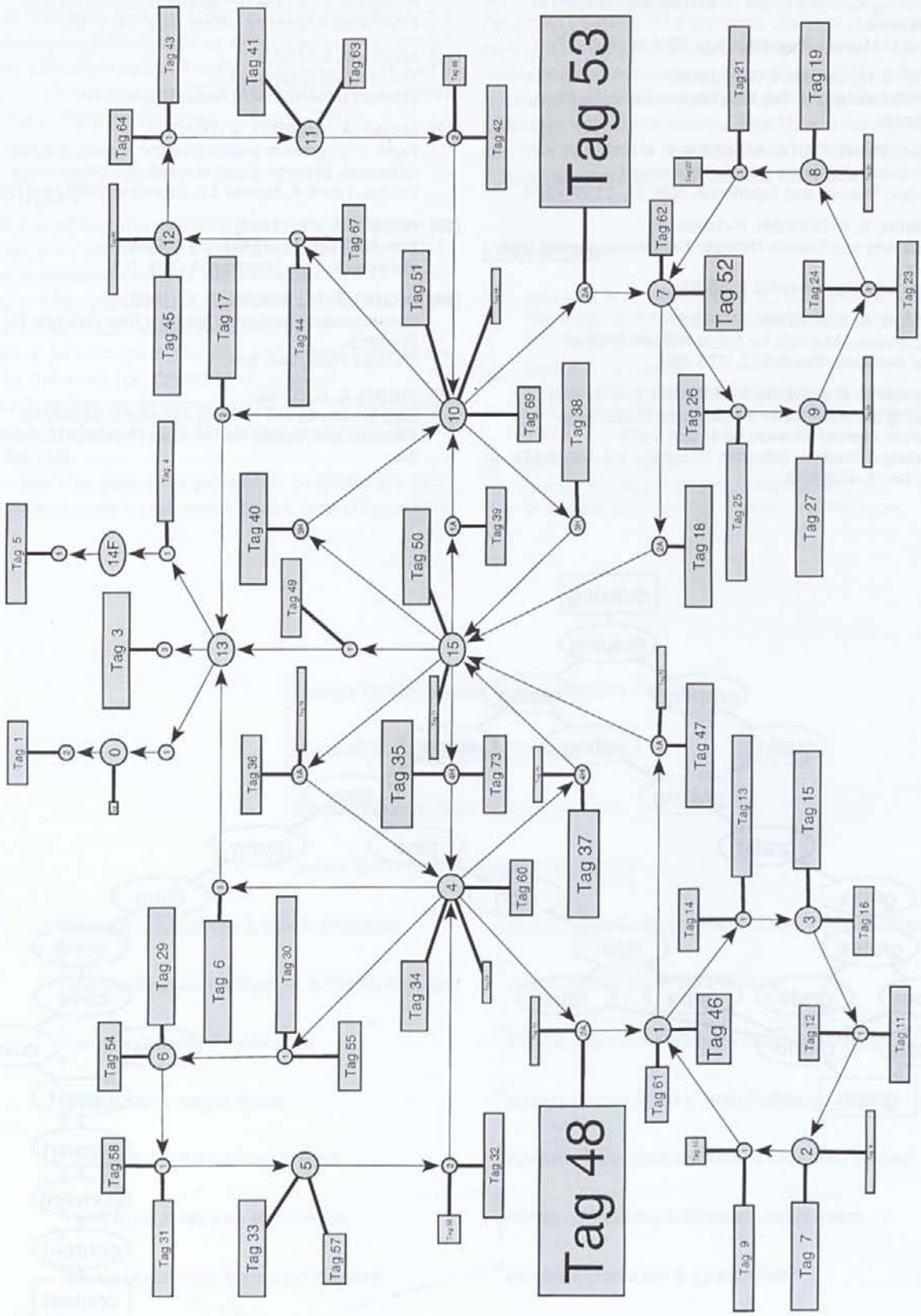
[13] PETERSON, J. L. (1981):
Petri Net Theory and Modeling of Systems.
Prentice-Hall, Inc., Englewood Cliffs, N.J.

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

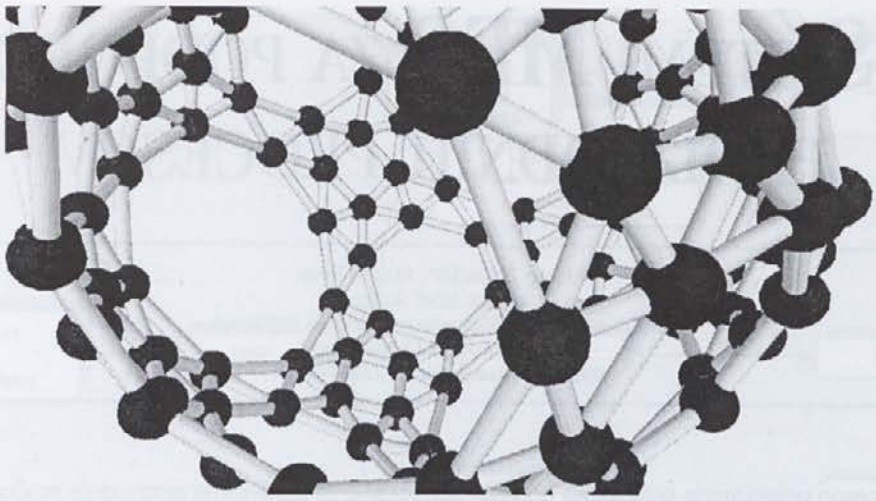
[15] TARJAN, R. E. (1983):
Data Structures and Network Algorithms. Society for Industrial and Applied Mathematics Philadelphia, Pennsylvania.



Slika 4: Najkrajše poti med besedami.



Slika 5: Prvonagrajeni graf s tekmovanja v risanju grafov GD'98.



Slika 6: Posnetek prostorskega grafa.

[16] WATSON, A. H., MCCABE, T. J. (1996): Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric. Computer Systems Laboratory, National Institute of Standards and Technology Special Publication 500-235, Gaithersburg, MD 20899-0001.

[17] WHITE, D. R., JORION, P. (1992): Representing and Computing Kinship: A New Approach. Current Athropology šĐbf 33, 454-462.

[18] WHITE, D. R., JORION, P. (1996): Kinship Networks and Discrete Structure Theory: Applications and Implications. Social Networks 18, 267-314.

[19] American Presidents GEDCOM file: ftp://www.dcs.hull.ac.uk/public/genealogy/

[20] Erdös Number Project: http://www.oakland.edu/čgrossman/erdoshp.html

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

[22] Graph Drawing Competition 1996, Graph B: http://portal.research.bell-labs.com/orgs/ssr/people/north/contest.html

[23] Graph Drawing Competition 1998: http://gd98.cs.mcgill.ca/contest/

[24] KNUTH, D. E.: Dictionary. Stanford University, Computer Science Department: ftp://labrea.stanford.edu/pub/dict/

[25] parni rodovniki: http://eclectic.ss.uci.edu/čdrwhite/pgraph/p-graphs.html

[26] Plug-in Chime: http://www.mdli.com/download/chimedown.html

[27] Plug-in Cosmo Player: http://cosmosoftware.com/

[28] Program GSView: ftp://ftp.cs.wisc.edu/pub/ghost/rjl/

[29] Program Mage: http://www.prosci.org/Kinemage/MageSoftware.html

[30] Program MODEL2: http://vlado.fmf.uni-lj.si/pub/networks/stran/default.htm

[31] Program RasMol (RASter MOLecules): http://klaatu.oit.umass.edu/microbio/rasmol/getras.htm

[32] Program Ucinet: http://eclectic.ss.uci.edu/člin/order.html

[33] Program Vega: http://vega.ijp.si/Htmldoc/Vega03.html

[34] SMITH, M. A. (1996): NetScan, Department of Sociology, UCLA: http://www.sscnet.ucla.edu/soc/csoc/netscan/netscan.htm

[35] Tekmovanja v risanju grafov: http://vlado.fmf.uni-lj.si/pub/gd/gd95.htm

[36] Theoretical Computer Science Genealogy: http://sigact.acm.org/genealogy/

[37] Transportation Networks, National Transportation Atlas Database, Bureau of Transportation Statistics: http://www.bts.gov/gis/ntatlas/networks.html

◆

Vladimir Batagelj je izredni profesor za diskretno in računalniško matematiko na Univerzi v Ljubljani, Fakulteta za matematiko in fiziko. Raziskovalno se ukvarja z diskretno matematiko (teorija grafov), optimizacijskimi metodami in analizo podatkov; predava pa predmete: diskretne strukture, kombinatorika, optimizacijske metode, operacijske raziskave, učenje z računalnikom ter razvoj matematike in računalništva. Je član več strokovnih združenj (DMFA RS, Informatika, INSNA, CSNA, ISI, IEEE, EuroLogo) in programskega sveta Ro.

◆

Andrej Mrvar je diplomiral na Fakulteti za elektrotehniko in računalništvo, program računalništvo, leta 1992. Na isti fakulteti je leta 1995 zaključil magistrski študij. Na Fakulteti za računalništvo in informatiko trenutno pripravlja doktorsko disertacijo z naslovom Analiza in prikaz velikih omrežij. Zaposlen je kot asistent za računalništvo in statistiko na Fakulteti za družbene vede.