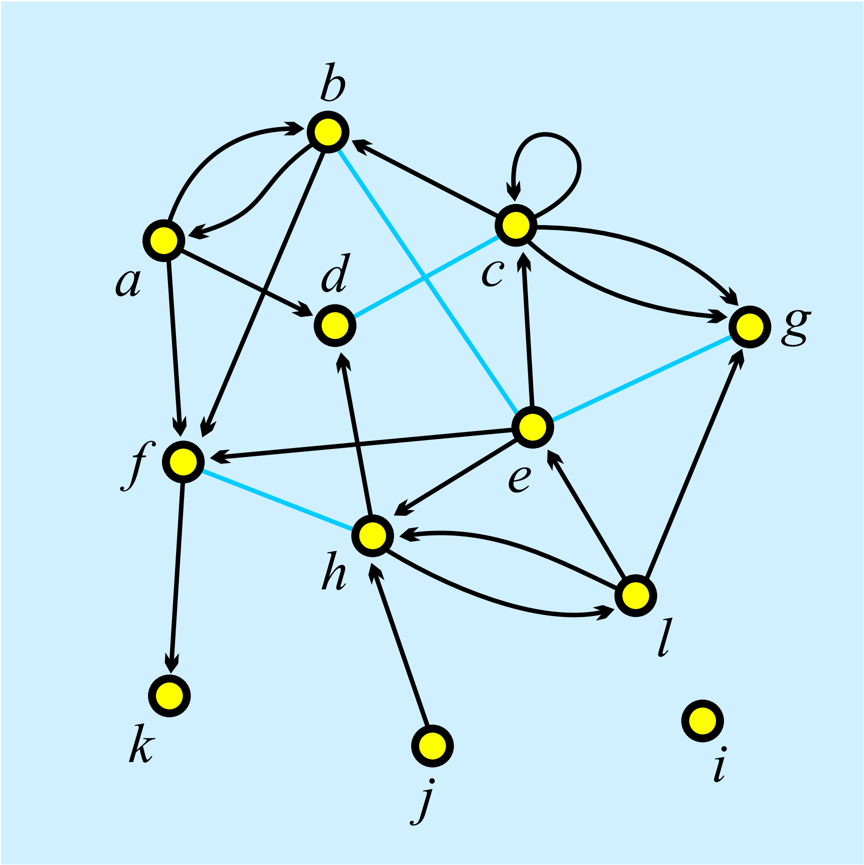


Network Analysis

Description of Networks

Vladimir Batagelj

University of Ljubljana

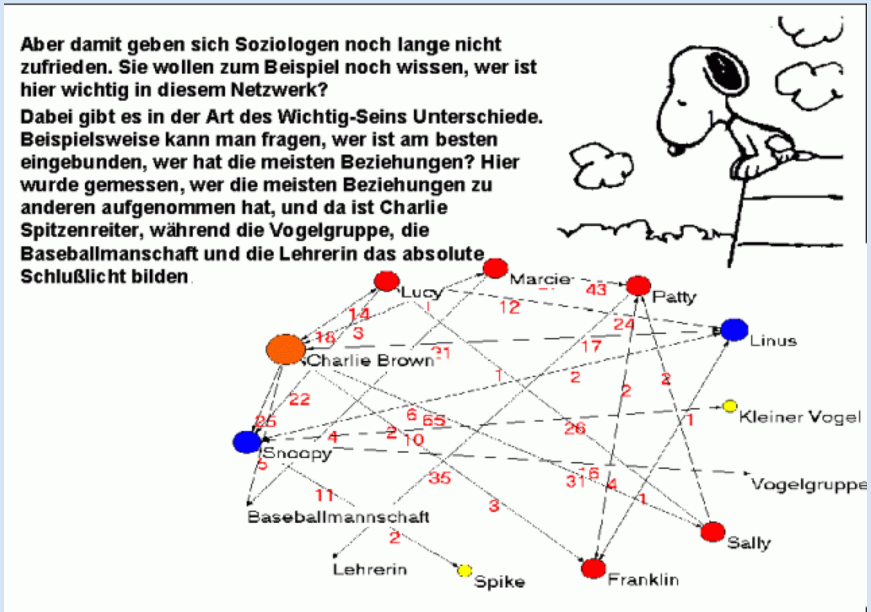
ECPR Summer School, July 30 – August 16, 2008

Faculty of Social Sciences, University of Ljubljana

version: August 5, 2008 / 00 : 10

Outline

Networks



Alexandra Schuler/ Marion Laging-Glaser:
Analyse von Snoopy Comics

A *network* is based on two sets – set of *vertices* (nodes), that represent the selected *units*, and set of *lines* (links), that represent *ties* between units. They determine a *graph*. A line can be *directed* – an *arc*, or *undirected* – an *edge*.

Additional data about vertices or lines can be known – their *properties* (attributes). For example: name/label, type, value, . . .

Network = Graph + Data

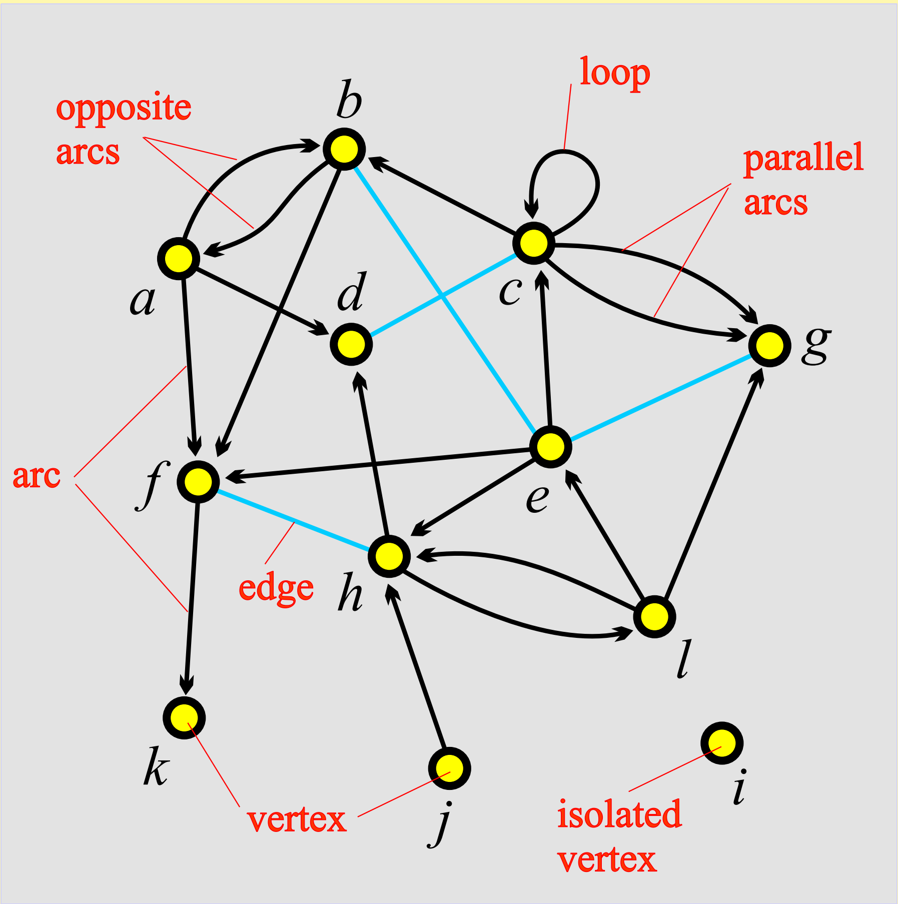
The data can be measured or computed.

Networks / Formally

A *network* $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$ consists of:

- a *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{L})$, where $\mathcal{V}$ is the set of vertices, $\mathcal{A}$ is the set of arcs, $\mathcal{E}$ is the set of edges, and $\mathcal{L} = \mathcal{E} \cup \mathcal{A}$ is the set of lines.
  $n = |\mathcal{V}|$, $m = |\mathcal{L}|$
- $\mathcal{P}$ *vertex value functions* / properties: $p : \mathcal{V} \to A$
- $\mathcal{W}$ *line value functions* / weights: $w : \mathcal{L} \to B$

Graph



unit, actor – vertex, node

tie, link – line, edge, arc

arc = directed line, (a, d)

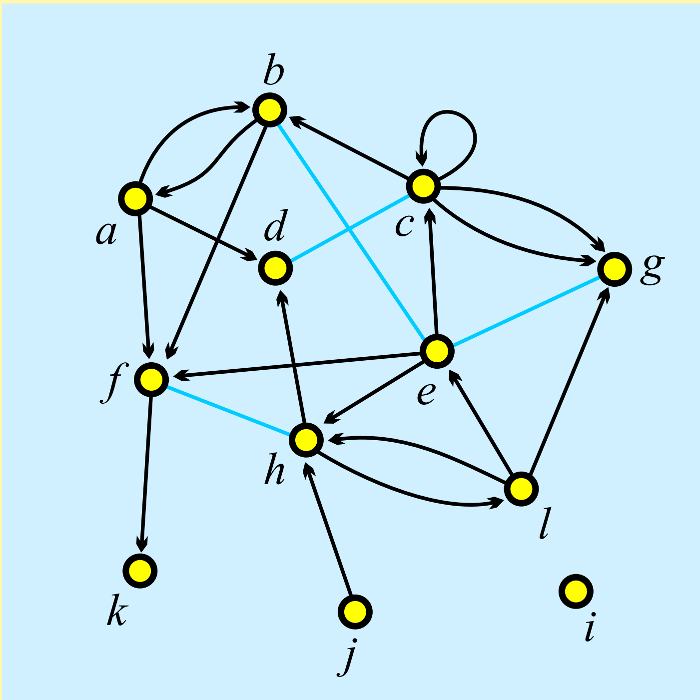
a is the *initial* vertex,

d is the *terminal* vertex.

edge = undirected line, $(c: d)$

c and d are *end* vertices.

Graph / Sets – NET



$$\mathcal{V} = \{a, b, c, d, e, f, g, h, i, j, k, l\}$$

$$\mathcal{A} = \{(a, b), (a, d), (a, f), (b, a), (b, f), (c, b), (c, c), (c, g), (c, g), (e, c), (e, f), (e, h), (f, k), (h, d), (h, l), (j, h), (l, e), (l, g), (l, h)\}$$

$$\mathcal{E} = \{(b\colon e), (c\colon d), (e\colon g), (f\colon h)\}$$

$$\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{E})$$

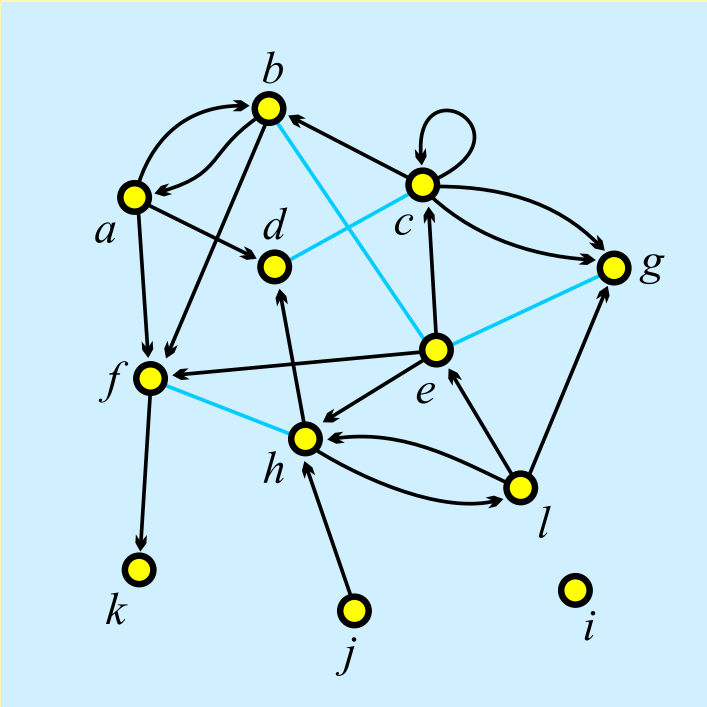
$$\mathcal{L} = \mathcal{A} \cup \mathcal{E}$$

$\mathcal{A} = \emptyset$ – *undirected* graph; $\mathcal{E} = \emptyset$ – *directed* graph.

Pajek: local: `GraphSet`; `TinaSet`;

WWW: `GraphSet` / `net`; `TinaSet` / `net`, picture `picture`.

Graph / Neighbors – NET



$N_A(a) = \{b, d, f\}$
$N_A(b) = \{a, f\}$
$N_A(c) = \{b, c, g, g\}$
$N_A(e) = \{c, f, h\}$
$N_A(f) = \{k\}$
$N_A(h) = \{d, l\}$
$N_A(j) = \{h\}$
$N_A(l) = \{e, g, h\}$

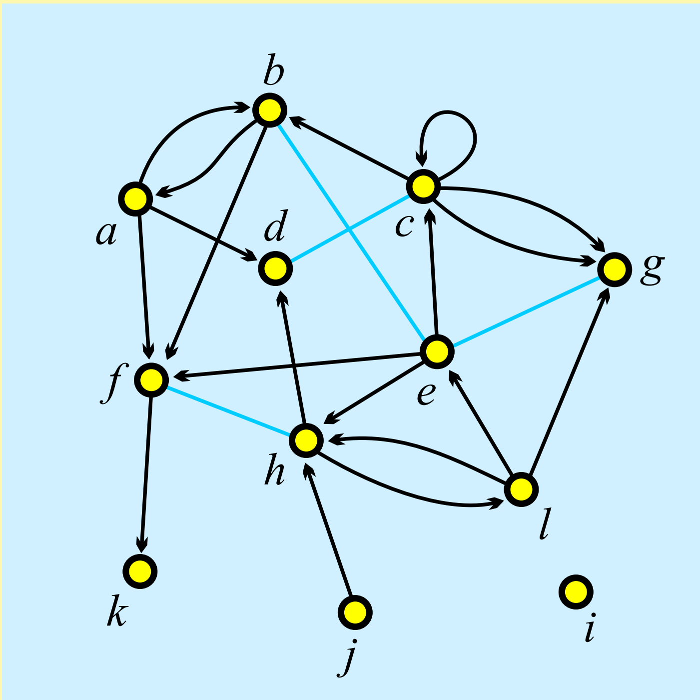
$N_E(e) = \{b, g\}$
$N_E(c) = \{d\}$
$N_E(f) = \{h\}$

Pajek: local: `GraphList`; `TinaList`;
WWW: `GraphList` / `net`; `TinaList` / `net`.

$$N(v) = N_A(v) \cup N_E(v), \qquad \text{also} \quad N_{out}(v),\ N_{in}(v)$$

Star in v, $S(v)$ is the set of all lines with v as their initial vertex.

Graph / Matrix – MAT



	a	b	c	d	e	f	g	h	i	j	k	l
a	0	1	0	1	0	1	0	0	0	0	0	0
b	1	0	0	0	1	1	0	0	0	0	0	0
c	0	1	1	1	0	0	2	0	0	0	0	0
d	0	0	1	0	0	0	0	0	0	0	0	0
e	0	1	1	0	0	1	1	1	0	0	0	0
f	0	0	0	0	0	0	0	1	0	0	1	0
g	0	0	0	0	1	0	0	0	0	0	0	0
h	0	0	0	1	0	1	0	0	0	0	0	1
i	0	0	0	0	0	0	0	0	0	0	0	0
j	0	0	0	0	0	0	0	1	0	0	0	0
k	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	1	0	1	1	0	0	0	0

Pajek: local: `GraphMat`; `TinaMat`, picture `picture`;

WWW: `GraphMat` / `net`; `TinaMat` / `net`, `paj`.

Graph G is *simple* if in the corresponding matrix all entries are 0 or 1.

Vertex Properties / CLU, VEC, PER

All three types of files have the same structure:

`*vertices` n	n is the number of vertices
v_1	vertex 1 has value v_1
$\ldots$	
v_n	

CLUstering – partition of vertices – *nominal* or *ordinal* data about vertices
$v_i \in \mathbb{N}$: vertex i belongs to the cluster v_i;

VECtor – *numeric* data about vertices
$v_i \in \mathbb{R}$: the property has value v_i on vertex i;

PERmutation – *ordering* of vertices
$v_i \in \mathbb{N}$: vertex i is at the v_i-th position.

When collecting the network data consider to provide as much properties as possible.

Example: Wolfe Monkey Data

inter.net	inter.net	sex.clu	age.vec	rank.per
*Vertices 20	1 6 5	*vertices 20	*vertices 20	*vertices 20
1 "m01"	1 7 9	1	15	1
2 "m02"	1 8 7	1	10	2
3 "m03"	1 9 4	1	10	3
4 "m04"	1 10 3	1	8	4
5 "m05"	1 11 3	1	7	5
6 "f06"	1 12 7	2	15	10
7 "f07"	1 13 3	2	5	11
8 "f08"	1 14 2	2	11	6
9 "f09"	1 15 5	2	8	12
10 "f10"	1 16 1	2	9	9
11 "f11"	1 17 4	2	16	7
12 "f12"	1 18 1	2	10	8
13 "f13"	2 3 5	2	14	18
14 "f14"	2 4 1	2	5	19
15 "f15"	2 5 3	2	7	20
16 "f16"	2 6 1	2	11	13
17 "f17"	2 7 4	2	7	14
18 "f18"	2 8 2	2	5	15
19 "f19"	2 9 6	2	15	16
20 "f20"	2 10 2	2	4	17
*Edges	2 11 5			
1 2 2	2 12 4			
1 3 10	2 13 3			
1 4 4	2 14 2			
	...			

Important notes: 0 is not allowed as vertex number. **Pajek** doesn't support Unix text files – lines should be ended with CR LF.

Pajek's Project File / `PAJ`

All types of data can be combined into a single file – Pajek's *project* file *file*`.paj`.

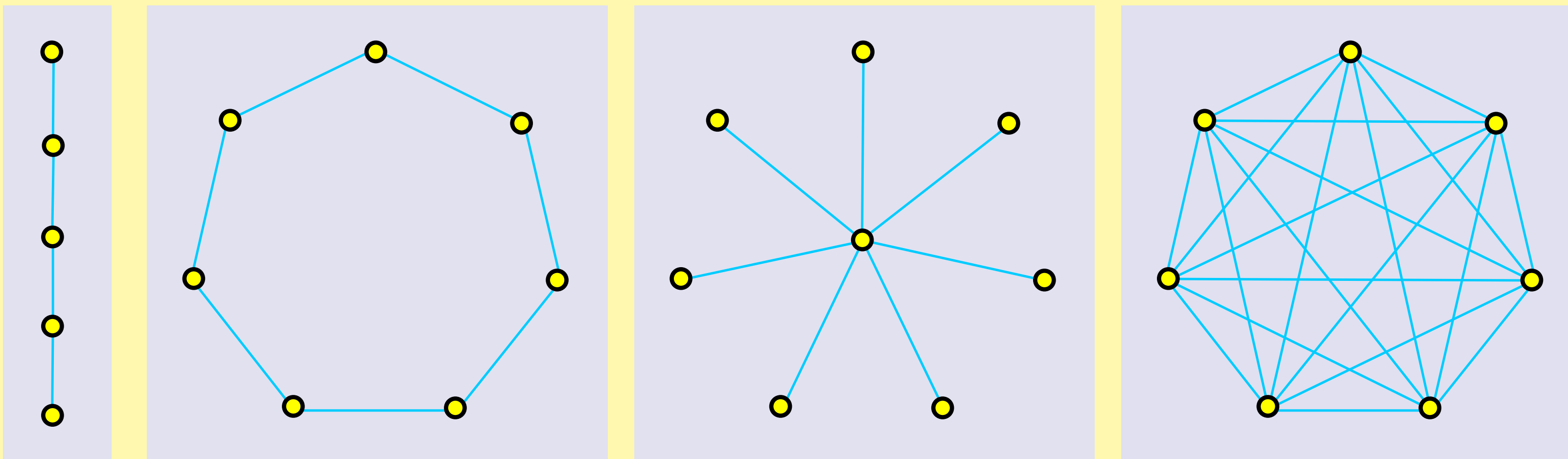
The easiest way to do this is:

- read all data files in Pajek,
- compute some additional data,
- delete (dispose) some data,
- save all as a project file with `File/Project file/Save`.

Next time you can restore everything with a single `File/Project file/Read`.

Wolfe network as Pajek's project file (PDF/paj).

Special graphs – path, cycle, star, complete



Graphs: *path* P_5, *cycle* C_7, *star* S_8 in *complete graph* K_7.

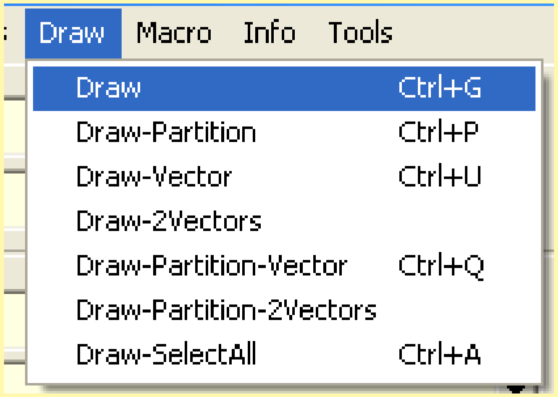
Representations of properties

Properties of vertices $\mathcal{P}$ and lines $\mathcal{W}$ can be measured in different scales: numerical, ordinal and nominal. They can be *input* as data or *computed* from the network.

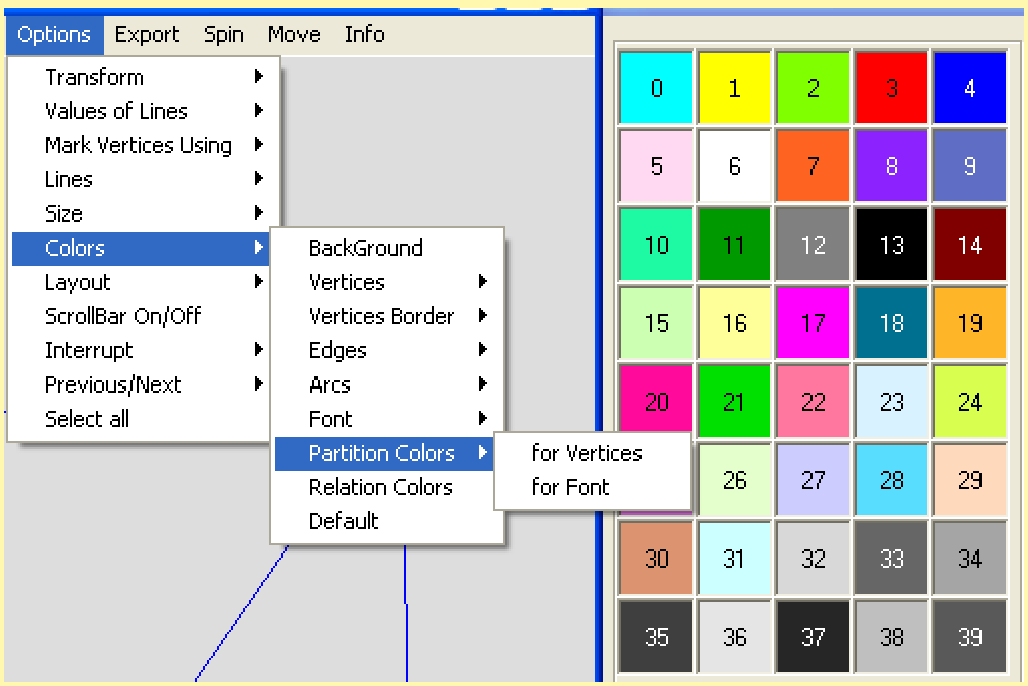
In **Pajek** numerical properties of vertices are represented by *vector*s, nominal properties by *partitions* or as *label*s of vertices. Numerical property can be displayed as *size* (width and height) of vertex (figure), as its *coordinate*; and a nominal property as *color* or *shape* of the figure, or as a vertex *label* (content, size and color).

We can assign in **Pajek** numerical values to links. They can be displayed as *value*, *thickness* or *grey level*. Nominal vales can be assigned as *label*, *color* or *line pattern* (see Pajek manual, section ***4.3***).

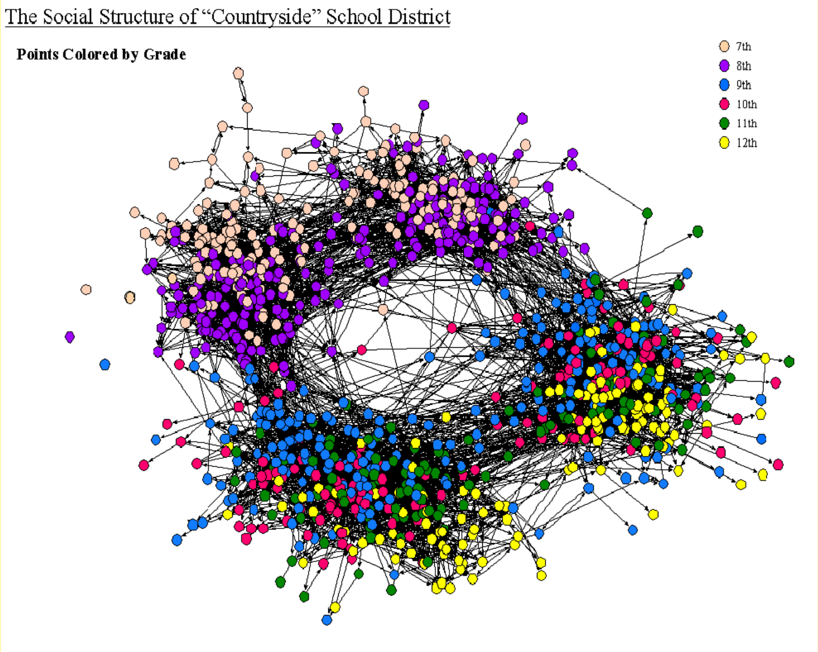
Some related operations

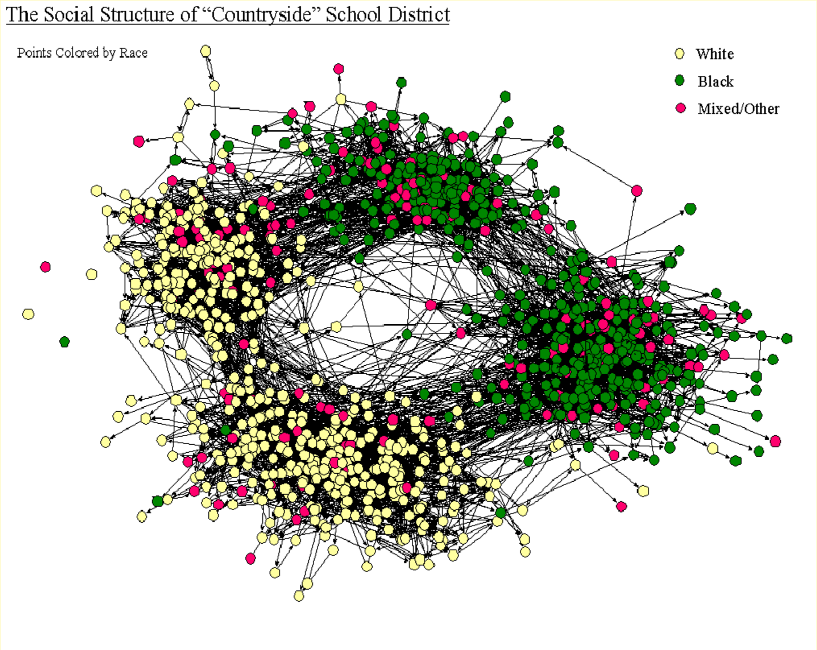



```
Operations/Vector/Put Coordinate
Net/Vector/Get Coordinate
[Draw] Options
[Draw] Layout/Energy/Kamada-Kawai
[Draw] Export/2D/EPS-PS
```



Display of properties – school (Moody)





Types of networks

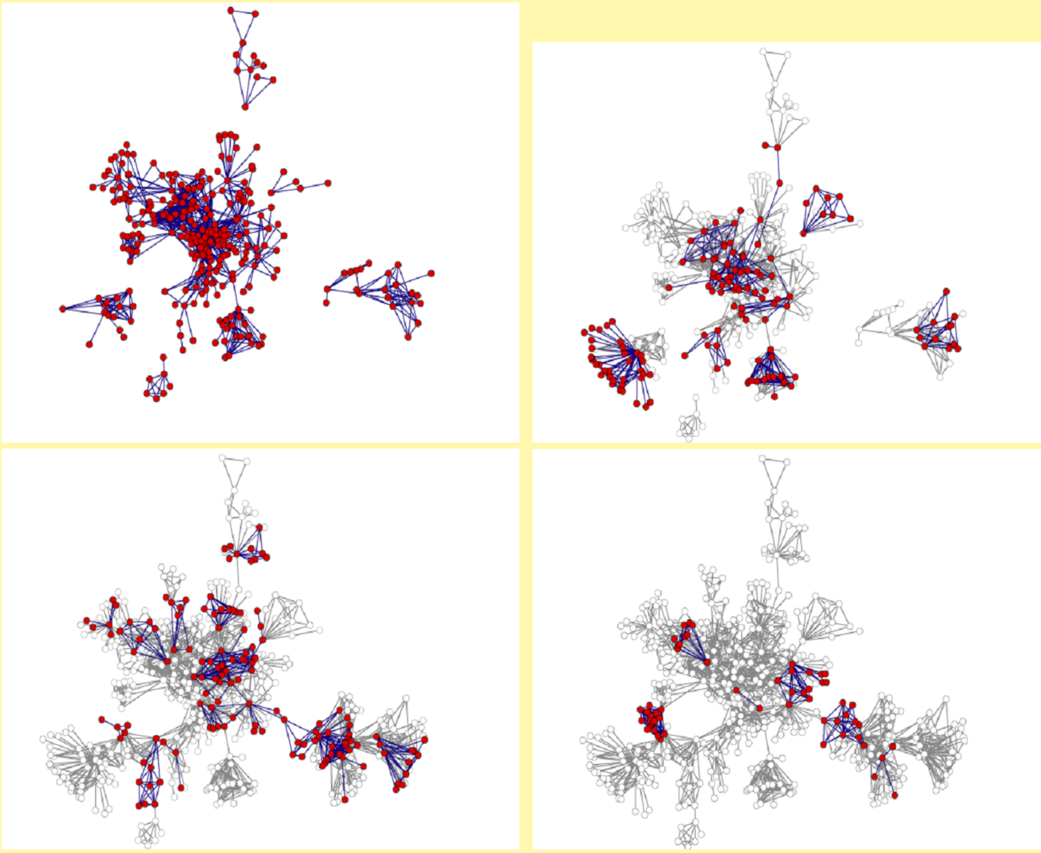
Besides ordinary (directed, undirected, mixed) networks some extended types of networks are also used:

- *2-mode networks*, bipartite (valued) graphs – networks between two disjoint sets of vertices.
- *multi-relational networks*.
- *temporal networks*, dynamic graphs – networks changing over time.
- specialized networks: representation of genealogies as *p-graphs*; *Petri's nets*, ...

The network (input) file formats should provide means to express all these types of networks. All interesting data should be recorded (respecting privacy).

Temporal networks

In a *temporal network* the presence/activity of vertex/line can change through time. **Pajek** supports two types of descriptions of temporal networks based on *presence* and on *events*.



Moody:
Drug users in Colorado Springs, 5 years

Temporal network

Temporal network

$$\mathcal{N}_T = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W}, T)$$

is obtained if the *time* T is attached to an ordinary network. T is a set of *time points* $t \in T$.

In temporal network vertices $v \in \mathcal{V}$ and lines $l \in \mathcal{L}$ are not necessarily present or active in all time points. If a line $l(u, v)$ is active in time point t then also its endpoints u and v should be active in time t.

We will denote the network consisting of lines and vertices active in time $t \in T$ by $\mathcal{N}(t)$ and call it the *time slice* in time point t. To get time slices in **Pajek** use

`Net / Transform / Generate in time`

Temporal networks – presence

```
*Vertices 3
1 "a" [5-10,12-14]
2 "b" [1-3,7]
3 "e" [4-*]
*Edges
1 2 1 [7]
1 3 1 [6-8]
```

Time.net.

Vertex a is present in time points 5, 6, 7, 8, 9, 10 and 12, 13, 14.

Edge $(1 : 3)$ is present in time points 6, 7, 8.

$\star$ means 'infinity'.

A line is present, if both its end-vertices are present.

Temporal networks – events

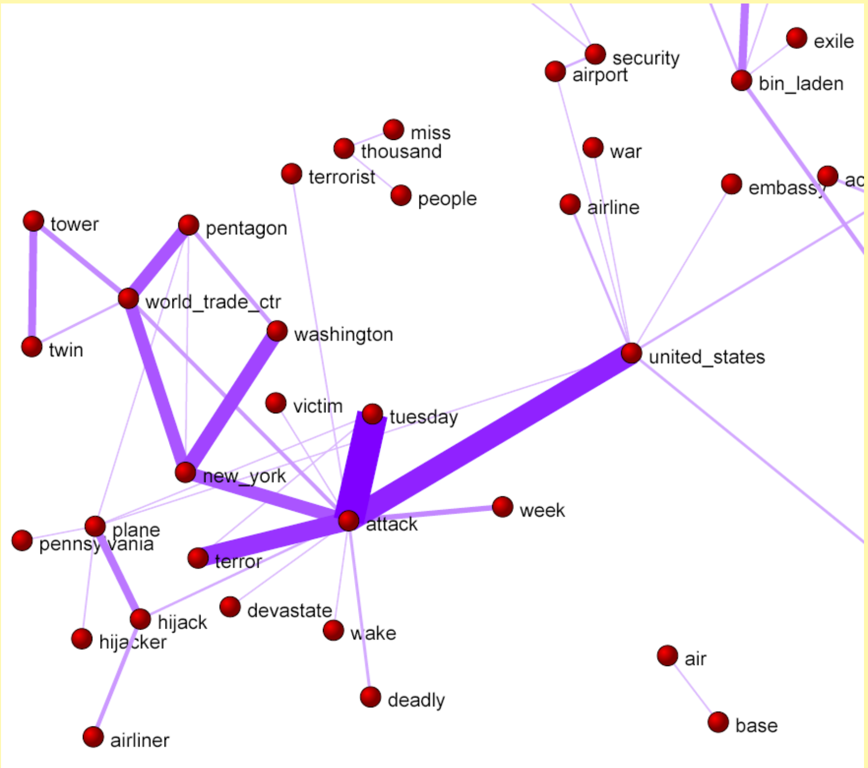
Event	Explanation
`TI` t	initial events – following events happen when time point t starts
`TE` t	end events – following events happen when time point t is finished
`AV` $v\,n\,s$	add vertex v with label n and properties s
`HV` v	hide vertex v
`SV` v	show vertex v
`DV` v	delete vertex v
`AA` $u\,v\,s$	add arc (u,v) with properties s
`HA` $u\,v$	hide arc (u,v)
`SA` $u\,v$	show arc (u,v)
`DA` $u\,v$	delete arc (u,v)
`AE` $u\,v\,s$	add edge $(u{:}v)$ with properties s
`HE` $u\,v$	hide edge $(u{:}v)$
`SE` $u\,v$	show edge $(u{:}v)$
`DE` $u\,v$	delete edge $(u{:}v)$
`CV` $v\,s$	change property of vertex v to s
`CA` $u\,v\,s$	change property of arc (u,v) to s
`CE` $u\,v\,s$	change property of edge $(u{:}v)$ to s
`CT` $u\,v$	change (un)directedness of line (u,v)
`CD` $u\,v$	change direction of arc (u,v)
`PE` $u\,v\,s$	replace pair of arcs (u,v) and (v,u) by single edge $(u{:}v)$ with properties s
`AP` $u\,v\,s$	add pair of arcs (u,v) and (v,u) with properties s
`DP` $u\,v$	delete pair of arcs (u,v) and (v,u)
`EP` $u\,v\,s$	replace edge $(u{:}v)$ by pair of arcs (u,v) and (v,u) with properties s

s can be empty.
In case of parallel lines $:k$ denotes the k-th line – `HE:3 14 37` hides the third edge connecting vertices `14` and `37`.

```
*Vertices 3
*Events
TI 1
AV 2 "b"
TE 3
HV 2
TI 4
AV 3 "e"
TI 5
AV 1 "a"
TI 6
AE 1 3 1
TI 7
SV 2
AE 1 2 1
TE 7
DE 1 2
DV 2
TE 8
DE 1 3
TE 10
HV 1
TI 12
SV 1
TE 14
DV 1
```

`Time.tim`. `Friends.tim`. `File/Time Events Network`

Temporal networks / September 11



Steve Corman with collaborators from Arizona State University transformed, using his Centering Resonance Analysis (*CRA*), daily Reuters news (66 days) about September 11th into a temporal network of words coappearance.

Pictures in SVG: *66 days*.

Multiple networks

Multiple or *multi-relational* networks on the same set of vertices were implemented in **Pajek** only recently (November 2004). Examples of such networks are: Transportation system in a city (stations, lines); WordNet (words, semantic relations: synonymy, antonymy, hyponymy, meronymy,…), KEDS networks (states, relations between states: Visit, Ask information, Warn, Expel person, …), …

. . . Multiple networks

The relation can be assigned to a line as follows:

- add to a keyword for description of lines (`*arcs`, `*edges`, `*arcslist`, `*edgeslist`, `*matrix`) the number of relation followed by its name:

  ```
  *arcslist :3 "sent a letter to"
  ```

  All lines controlled by this keyword belong to the specified relation. (Sampson, SampsonL)

- Any line controlled by `*arcs` or `*edges` can be assigned to selected relation by starting its description by the number of this relation.

  ```
  3:  47 14 5
  ```

  Line with endpoints 47 and 14 and weight 5 belongs to relation 3.

KEDS / Gulf

Multi-relational temporal network – KEDS/WEIS

```
% Recoded by WEISmonths, Sun Nov 28 21:57:00 2004
% from http://www.ku.edu/~keds/data.dir/balk.html
*vertices 325
1 "AFG" [1-*]
2 "AFR" [1-*]
3 "ALB" [1-*]
4 "ALBMED" [1-*]
5 "ALG" [1-*]
 ...
318 "YUGGOV" [1-*]
319 "YUGMAC" [1-*]
320 "YUGMED" [1-*]
321 "YUGMTN" [1-*]
322 "YUGSER" [1-*]
323 "ZAI" [1-*]
324 "ZAM" [1-*]
325 "ZIM" [1-*]
*arcs :0 "*** ABANDONED"
*arcs :10 "YIELD"
*arcs :11 "SURRENDER"
*arcs :12 "RETREAT"
 ...
*arcs :223 "MIL ENGAGEMENT"
*arcs :224 "RIOT"
*arcs :225 "ASSASSINATE TORTURE"
*arcs
224: 314 153 1 [4]          890402  YUG     KSV     224  (RIOT)  RIOT-TORN
212: 314 83 1 [4]           890404  YUG     ETHALB  212  (ARREST PERSON) ALB ETHNIC JAILED IN YUG
224: 3 83 1 [4]             890407  ALB     ETHALB  224  (RIOT)  RIOTS
123: 83 153 1 [4]           890408  ETHALB  KSV     123  (INVESTIGATE)   PROBING
 ...
42: 105 63 1 [175]          030731  GER     CYP     042  (ENDORSE)       GAVE SUPPORT
212: 295 35 1 [175]         030731  UNWCT   BOSSER  212  (ARREST PERSON) SENTENCED TO PRISON
43: 306 87 1 [175]          030731  VAT     EUR     043  (RALLY) RALLIED
13: 295 35 1 [175]          030731  UNWCT   BOSSER  013  (RETRACT)       CLEARED
121: 295 22 1 [175]         030731  UNWCT   BAL     121  (CRITICIZE)     CHARGES
122: 246 295 1 [175]        030731  SER     UNWCT   122  (DENIGRATE)     TESTIFIED
121: 35 295 1 [175]         030731  BOSSER  UNWCT   121  (CRITICIZE)     ACCUSED
```

Kansas Event Data System *KEDS*

Two-mode networks

In a *two-mode* network $\mathcal{N} = (\mathcal{U}, \mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$ the set of vertices consists of two disjoint sets of vertices $\mathcal{U}$ and $\mathcal{V}$, and all the lines from $\mathcal{L}$ have one end-vertex in $\mathcal{U}$ and the other in $\mathcal{V}$. Often also a *weight* $w : \mathcal{L} \to \mathbb{R} \in \mathcal{W}$ is given; if not, we assume $w(u, v) = 1$ for all $(u, v) \in \mathcal{L}$.

A two-mode network can also be described by a rectangular matrix $\mathbf{A} = [a_{uv}]_{\mathcal{U}\times\mathcal{V}}$.

$$a_{uv} = \begin{cases} w_{uv} & (u, v) \in \mathcal{L} \\ 0 & \text{otherwise} \end{cases}$$

Examples: (persons, societies, years of membership), (buyers/consumers, goods, quantity), (parlamentarians, problems, positive vote), (persons, journals, reading).

A two-mode network is announced by `*vertices` $n\ n_{\mathcal{U}}$.

Authors and works.

Deep South

Classical example of two-mode network are Southern women (Davis 1941).
Davis.paj. Freeman's overview.

Names of Participants of Group I	Code Numbers and Dates of Social Events Reported in *Old City Herald*													
	(1) 6/27	(2) 3/2	(3) 4/12	(4) 9/26	(5) 2/25	(6) 5/19	(7) 3/15	(8) 9/16	(9) 4/8	(10) 6/10	(11) 2/23	(12) 4/7	(13) 11/21	(14) 8/3
1. Mrs. Evelyn Jefferson	×	×	×	×	×	×		×	×					
2. Miss Laura Mandeville	×	×	×		×	×	×	×						
3. Miss Theresa Anderson		×	×	×	×	×	×	×	×					
4. Miss Brenda Rogers	×		×	×	×	×	×	×						
5. Miss Charlotte McDowd			×	×	×		×							
6. Miss Frances Anderson			×		×	×		×						
7. Miss Eleanor Nye					×	×	×	×						
8. Miss Pearl Oglethorpe						×		×	×					
9. Miss Ruth DeSand					×		×	×	×					
10. Miss Verne Sanderson							×	×	×			×		
11. Miss Myra Liddell								×	×	×		×		
12. Miss Katherine Rogers								×	×	×		×	×	×
13. Mrs. Sylvia Avondale							×	×	×	×		×	×	×
14. Mrs. Nora Fayette						×	×		×	×	×	×	×	×
15. Mrs. Helen Lloyd							×	×		×	×	×		
16. Mrs. Dorothy Murchison								×	×					
17. Mrs. Olivia Carleton									×		×			
18. Mrs. Flora Price									×		×			