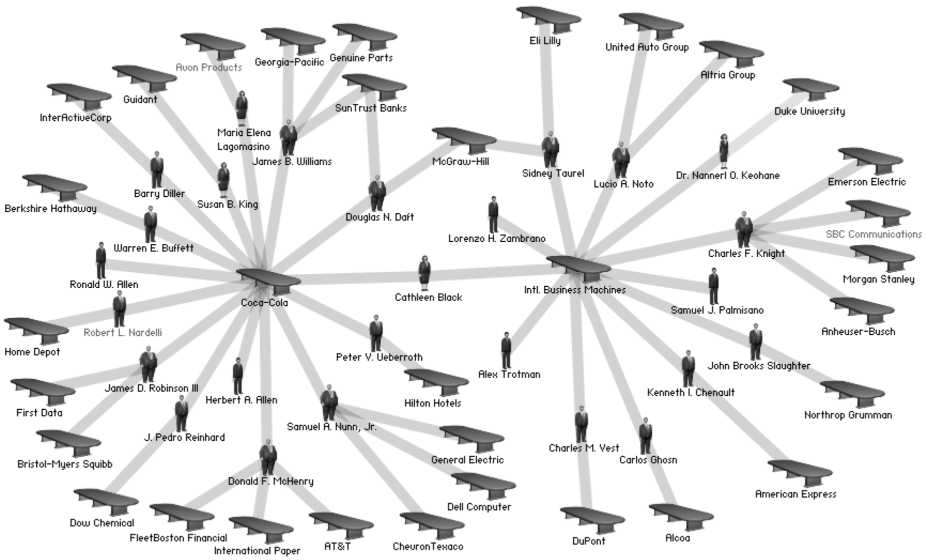


Josh On: They rule 2004

Network Analysis

Two mode Networks

Vladimir Batagelj

University of Ljubljana

ECPR Summer School, July 30 – August 16, 2008

Faculty of Social Sciences, University of Ljubljana

Outline

Analysis of two-mode networks

A *two-mode network* or *affiliation network* is a structure $\mathcal{N} = (\mathcal{U}, \mathcal{V}, \mathcal{A}, w)$, where $\mathcal{U}$ and $\mathcal{V}$ are disjoint sets of vertices, $\mathcal{A}$ is the set of arcs with the initial vertex in the set $\mathcal{U}$ and the terminal vertex in the set $\mathcal{V}$, and $w : \mathcal{A} \to \mathbb{R}$ is a weight. If no weight is defined we can assume a constant weight $w(u, v) = 1$ for all arcs $(u, v) \in \mathcal{A}$. The set $\mathcal{A}$ can be viewed also as a relation $A \subseteq \mathcal{U} \times \mathcal{V}$.

A two-mode network can be formally represented by rectangular matrix $\mathbf{W} = [w_{uv}]_{\mathcal{U} \times \mathcal{V}}$.

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

Approaches to two-mode network analysis

For direct analysis of two-mode networks we can use the *eigen-vector approach* – a two-mode variant of Kleinberg's hubs and authorities. The weight vector $(\mathbf{x}, \mathbf{y})$ on $\mathcal{U} \cup \mathcal{V}$ is determined by relations $\mathbf{y} = \mathbf{W}\mathbf{x}$ and $\mathbf{x} = \mathbf{W}^\mathsf{T}\mathbf{y}$.

Two new direct methods will be presented in this lecture: *two-mode cores* and *4-rings*. In the next lecture we shall also describe the *clustering* and *blockmodeling* in two-mode networks.

Internet Movie Database http://www.imdb.com/

12th Annual Graph Drawing Contest, 2005. The IMDB network is two-mode and has $1324748 = 428440 + 896308$ vertices and 3792390 arcs.

Two-mode cores

The subset of vertices $C \subseteq \mathcal{V}$ is a *(p,q)-core* in a two-mode network $\mathcal{N} = (\mathcal{V}_1, \mathcal{V}_2; \mathcal{L})$, $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$ iff

a. in the induced subnetwork $\mathcal{K} = (C_1, C_2; \mathcal{L}(C))$, $C_1 = C \cap \mathcal{V}_1$, $C_2 = C \cap \mathcal{V}_2$ it holds $\forall v \in C_1 : \deg_{\mathcal{K}}(v) \geq p$ and $\forall v \in C_2 : \deg_{\mathcal{K}}(v) \geq q$;

b. C is the maximal subset of $\mathcal{V}$ satisfying condition **a**.

Properties of two-mode cores:

- $C(0,0) = \mathcal{V}$
- $\mathcal{K}(p,q)$ is not always connected
- $(p_1 \leq p_2) \wedge (q_1 \leq q_2) \Rightarrow C(p_1,q_1) \subseteq C(p_2,q_2)$
- $\mathcal{C} = \{C(p,q) : p,q \in \mathbb{N}\}$. If all nonempty elements of $\mathcal{C}$ are different it is a lattice.

Algorithm for two-mode cores

To determine a (p, q)-core the procedure similar to the ordinary core procedure can be used:

repeat

remove from the first set all vertices of degree less than p,

and from the second set all vertices of degree less than q

until no vertex was deleted

It can be implemented to run in $O(m)$ time.

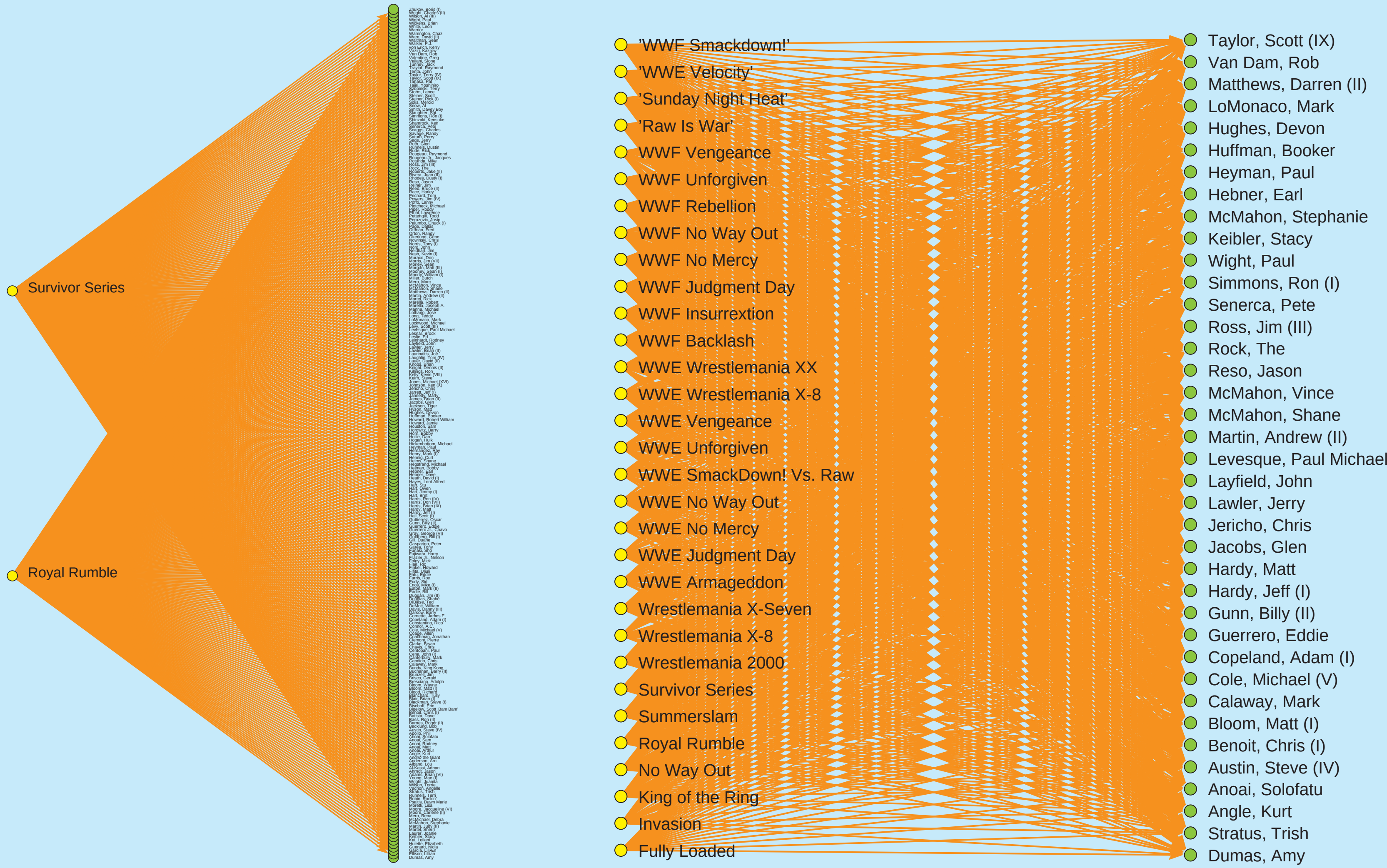
Interesting (p, q)-cores? Table of cores' characteristics $n_1 = |C_1(p, q)|$, $n_2 = |C_2(p, q)|$ and k – number of components in $\mathcal{K}(p, q)$:

- $n_1 + n_2 \leq$ selected threshold
- 'border line' in the (p, q)-table.

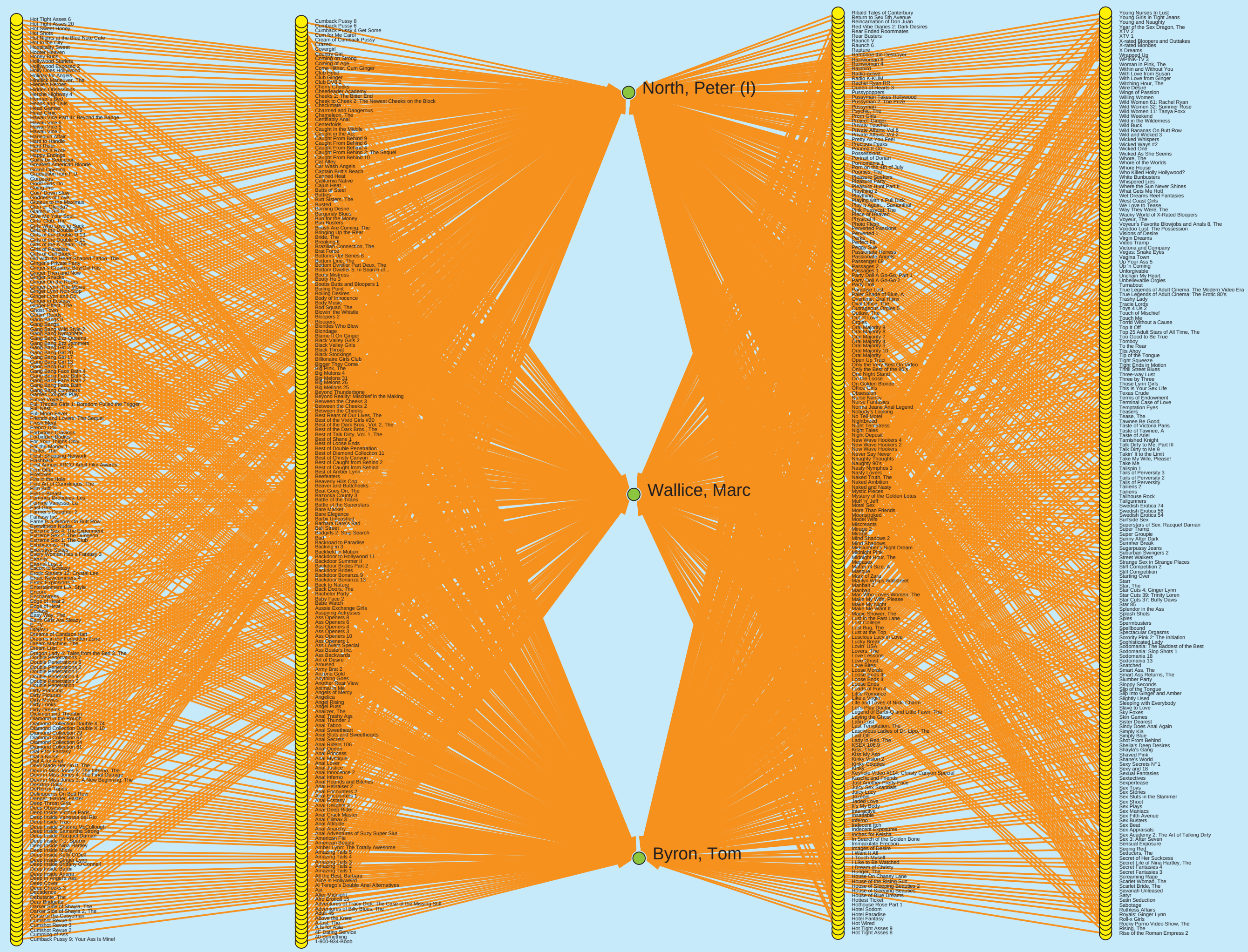
Table $(p, q : n_1, n_2)$ for Internet Movie Database

1	1590:	1590	1	16	39:	2173	678	44	14:	29	83
2	516:	788	3	17	35:	2791	995	46	13:	29	94
3	212:	1705	18	18	32:	2684	1080	49	12:	26	95
4	151:	4330	154	19	30:	2395	1063	52	11:	16	79
5	131:	4282	209	20	28:	2216	1087	56	10:	34	162
6	115:	3635	223	21	26:	1988	1087	62	9:	31	177
7	101:	3224	244	22	24:	1854	1153	66	8:	29	198
8	88:	2860	263	24	23:	34	39	72	7:	22	203
9	77:	3467	393	27	22:	31	38	96	6:	7	114
10	69:	3150	428	29	20:	35	52	119	5:	6	137
11	63:	2442	382	32	19:	34	57	141	4:	8	258
12	56:	2479	454	35	18:	33	61	186	3:	3	186
13	50:	3330	716	36	17:	33	65	247	2:	2	247
14	46:	2460	596	39	16:	29	70	1334	1:	1	1334
15	42:	2663	739	42	15:	28	76				

(247,2)-core and (27,22)-core



(2,516)-Hard core



IMDB cores / Pajek commands

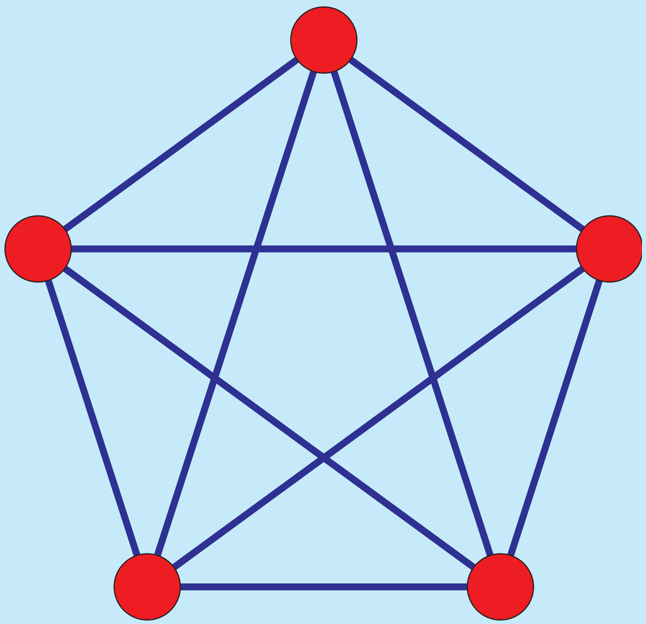
See How to deal with very large networks?

```
Options/Read-Write/Read-Save vertices labels [Off]
Read/Network [IMDB.net]  1:40
Info/Memory
Net/Partitions/Core/2-Mode Review
Net/Partitions/Core/2-Mode [27 22]
Info/Partition
Operations/Extract from Network/Partition [Yes 1]
Net/Partitions/2-Mode
Net/Transform/Add/Vertices Labels from File [IMDB.nam]
Draw/Draw-Partition
Layers/in y direction
Options/Transform/Rotate 2D [90]
```

k-rings

A *k-ring* is a simple closed chain of length k. Using k-rings we can define a weight of edges as

$w_k(e) = \#$ of different k-rings containing the edge $e \in \mathcal{E}$



Complete graph K_5

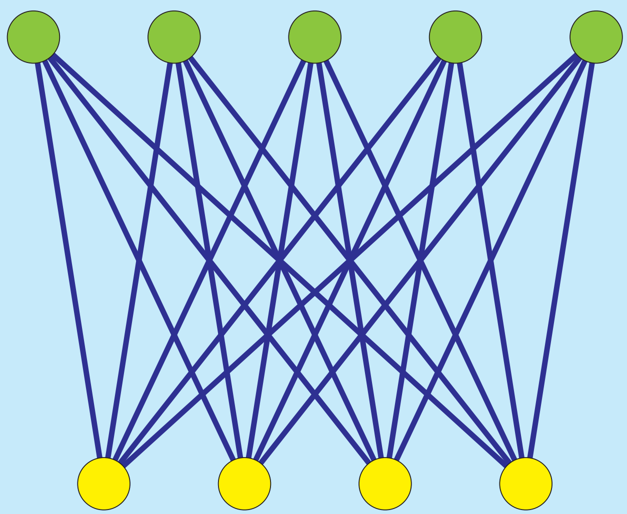
Since for each eadge e of a complete graph K_r, $r \geq k \geq 3$ we have $w_k(e) = (r-2)!/(r-k)!$ the edges belonging to cliques have large weights. Therefore these weights can be used to identify the dense parts of a network.

The k-rings can be efficiently determined only for small values of k – 3, 4, 5. The 3-rings (triangular) weights were implemented in **`Pajek`** in May 2002.

On the k-rings we can also base the notion of short cycle connectivity which provides us with another decomposition of networks.

4-rings and analysis of two-mode networks

In two-mode network there are no 3-rings. The densest substructures are complete bipartite subgraphs $K_{p,q}$. They contain many 4-rings.



There are

$$\binom{p}{2}\binom{q}{2} = \frac{1}{4}p(p-1)q(q-1)$$

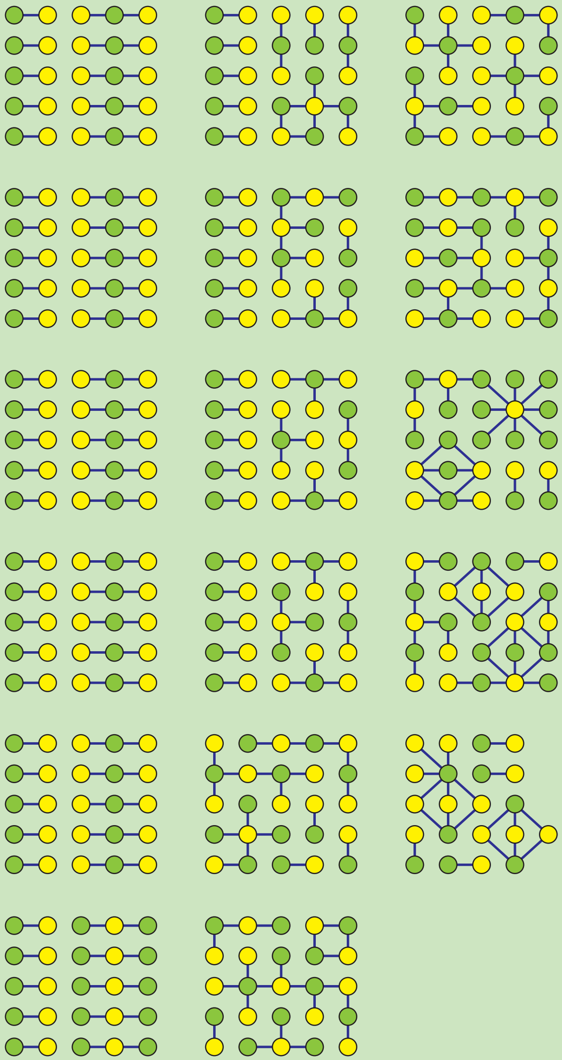
4-rings in $K_{p,q}$; and each of its edges e has weight

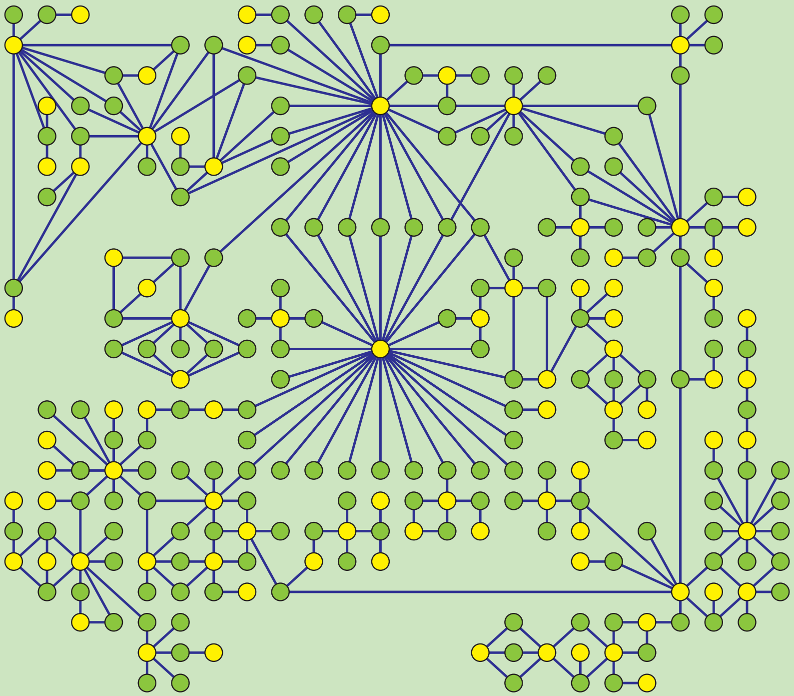
$$w_4(e) = (p-1)(q-1)$$

The 4-rings weights were implemented in **`Pajek`** in August 2005.

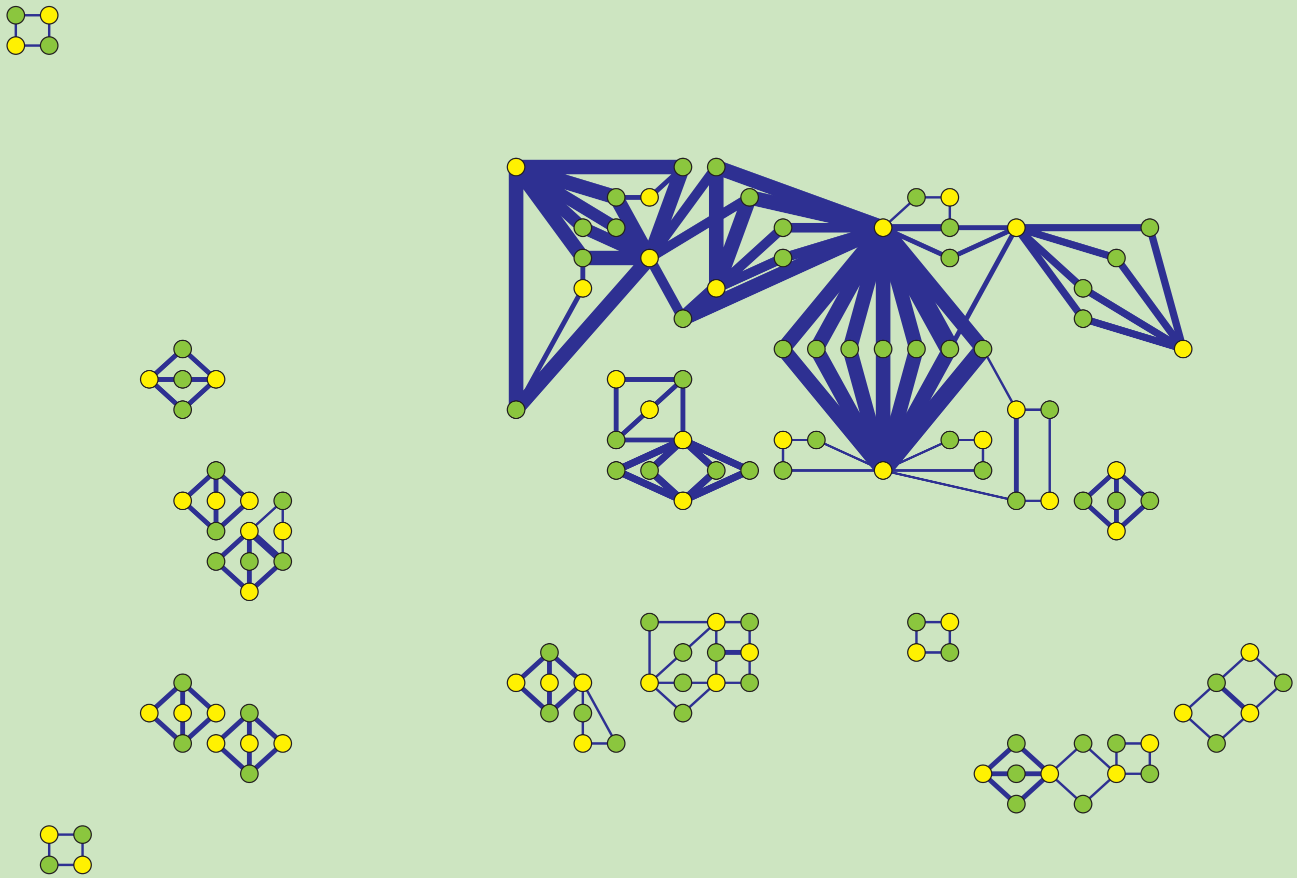
Example: Bibliography from W. Imrich, S. Klavžar: ***Product graphs: structure and recognition***, JohnWiley & Sons, New York, USA, 2000. (PDF), (net – two-mode 674×314 network).

Example: 4-rings in two-mode network

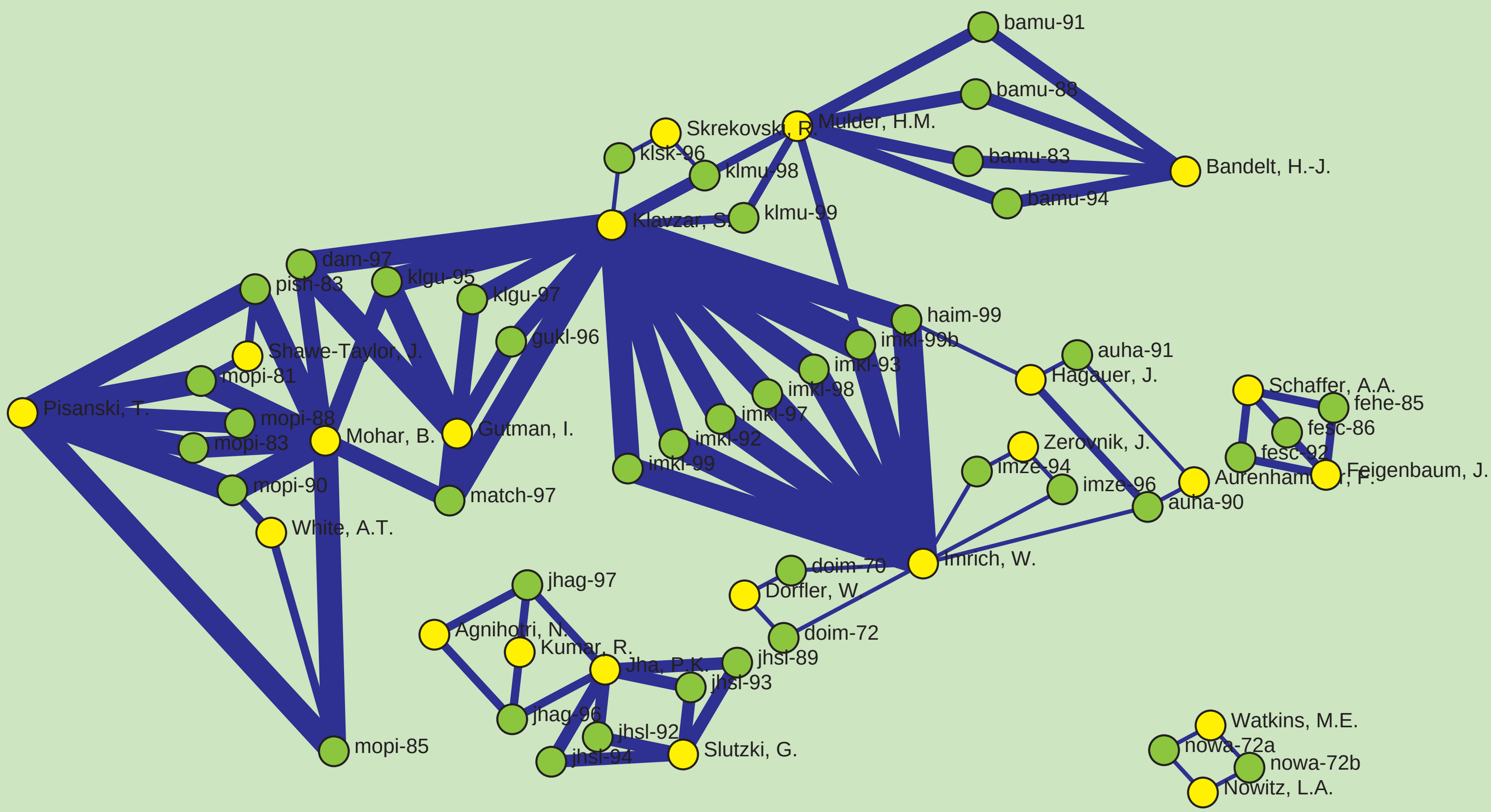




Example: 1-edge cut for w_4

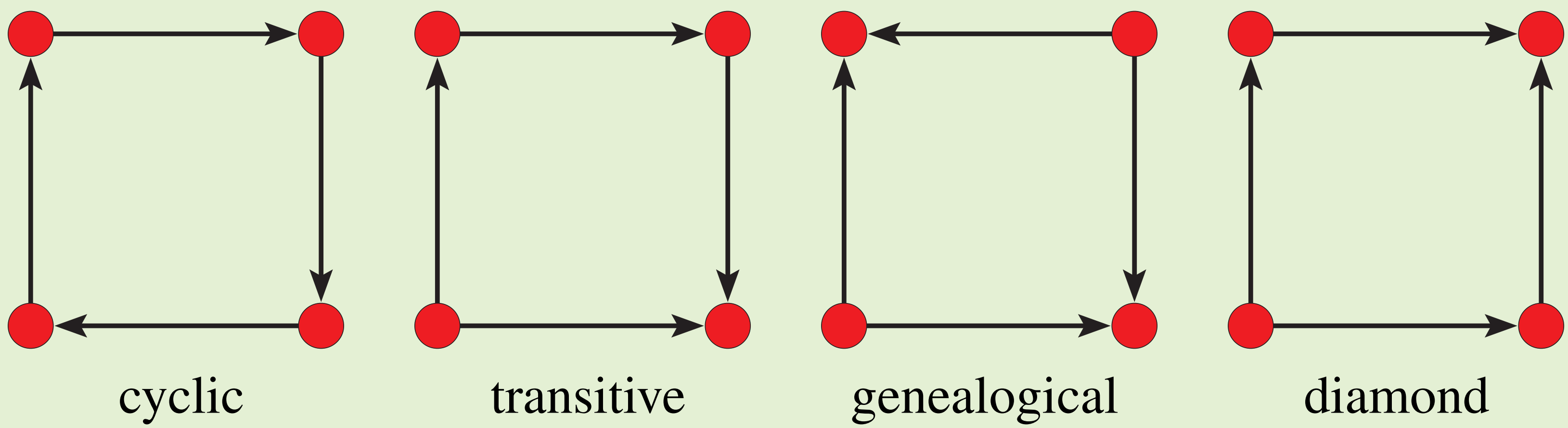


Example: labeled main part of 1-edge cut for w_4



Directed 4-rings

There are 4 types of directed 4-rings:



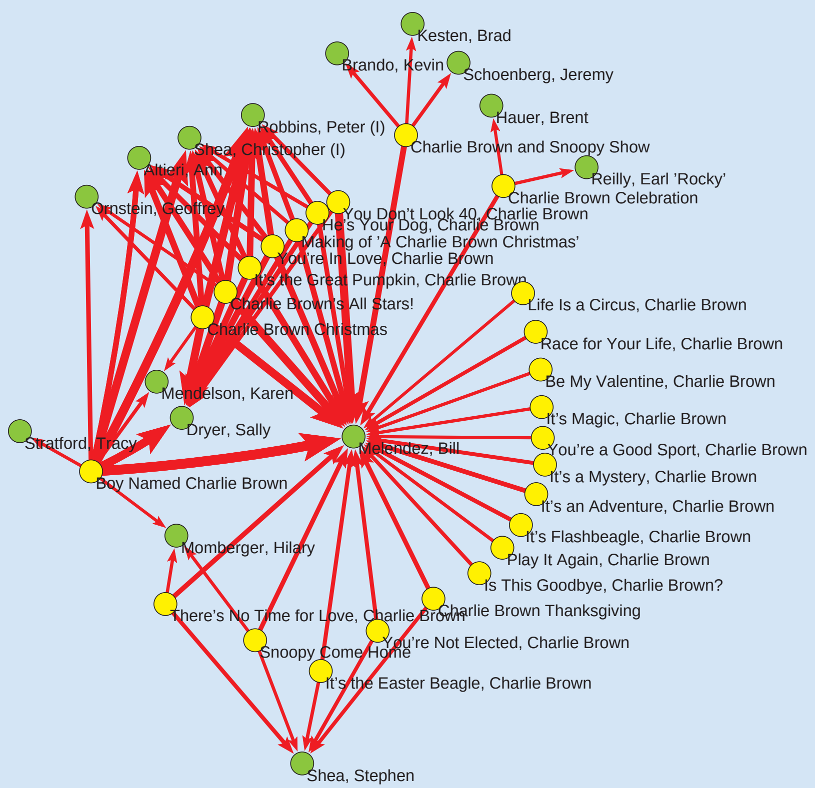
In the case of transitive rings **`Pajek`** provides a special weight counting on how many transitive rings the arc is a *shortcut*.

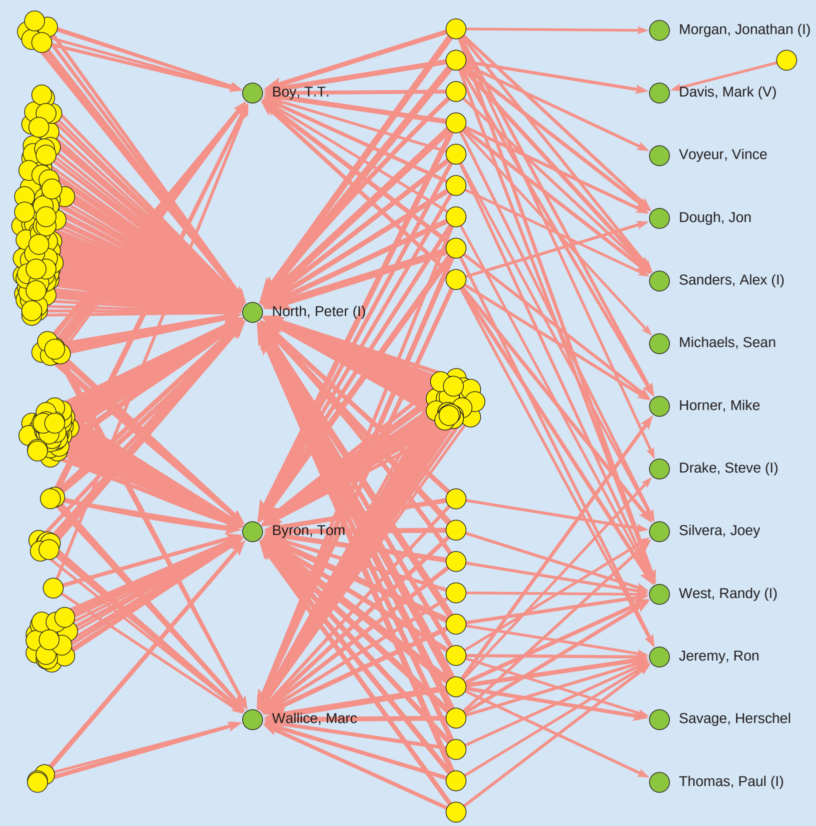
Simple line islands in IMDB for w_4

We obtained 12465 simple line islands on 56086 vertices. Here is their size distribution.

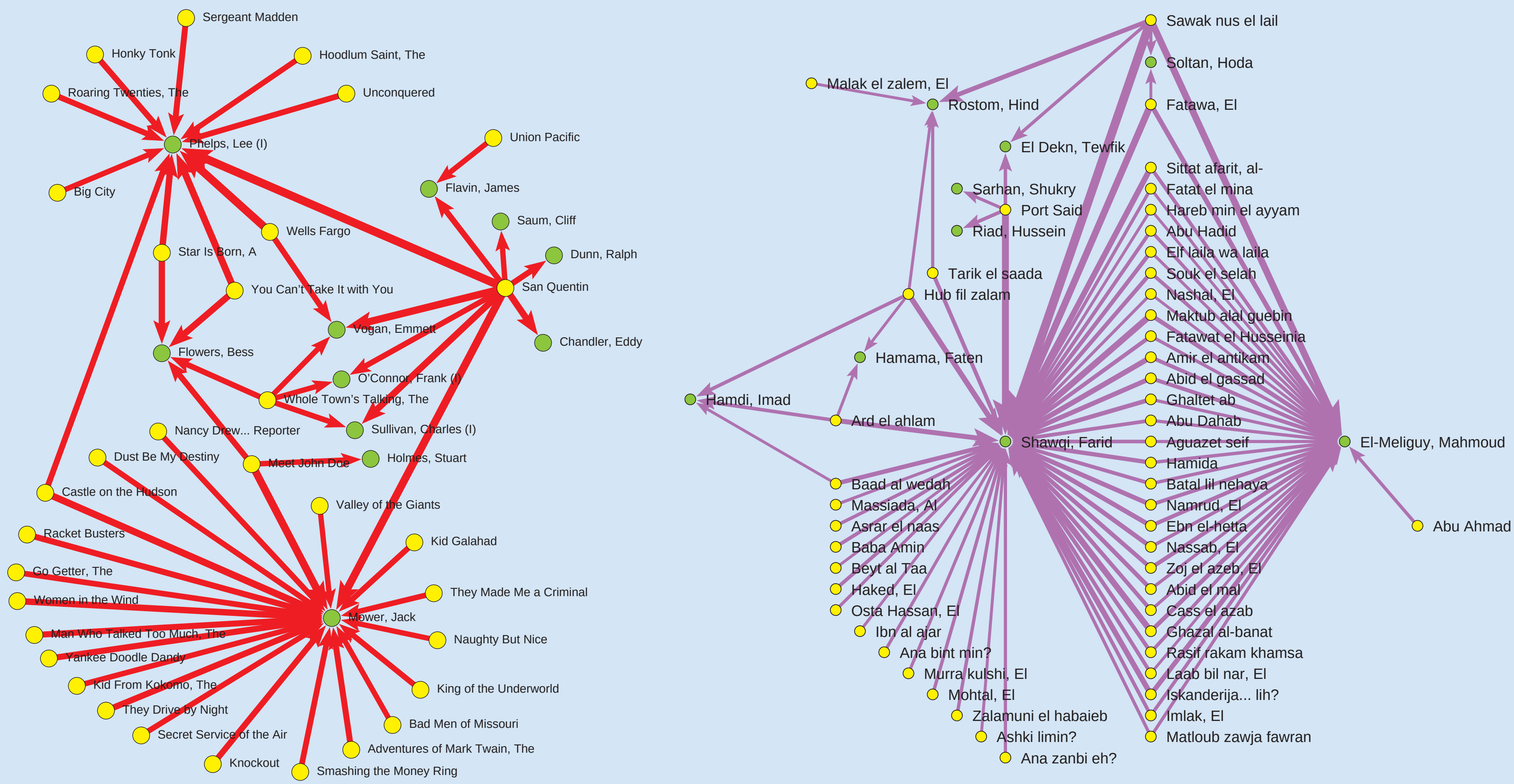
Size	Freq	Size	Freq	Size	Freq	Size	Freq
2	5512	20	19	38	4	59	2
3	1978	21	18	39	3	61	1
4	1639	22	15	40	2	64	1
5	968	23	9	42	2	67	1
6	666	24	13	43	3	70	1
7	394	25	12	45	3	73	1
8	257	26	6	46	4	76	1
9	209	27	6	47	5	82	1
10	148	28	5	48	1	86	1
11	118	29	6	49	2	106	1
12	87	30	3	50	2	122	1
13	55	31	6	51	1	135	1
14	62	32	5	52	2	144	1
15	46	33	3	53	1	163	1
16	39	34	1	54	2	269	1
17	27	35	5	55	1	301	1
18	28	36	4	57	1	332	2
19	29	37	7	58	1	673	1

Example: Islands for w_4 / Charlie Brown and Adult

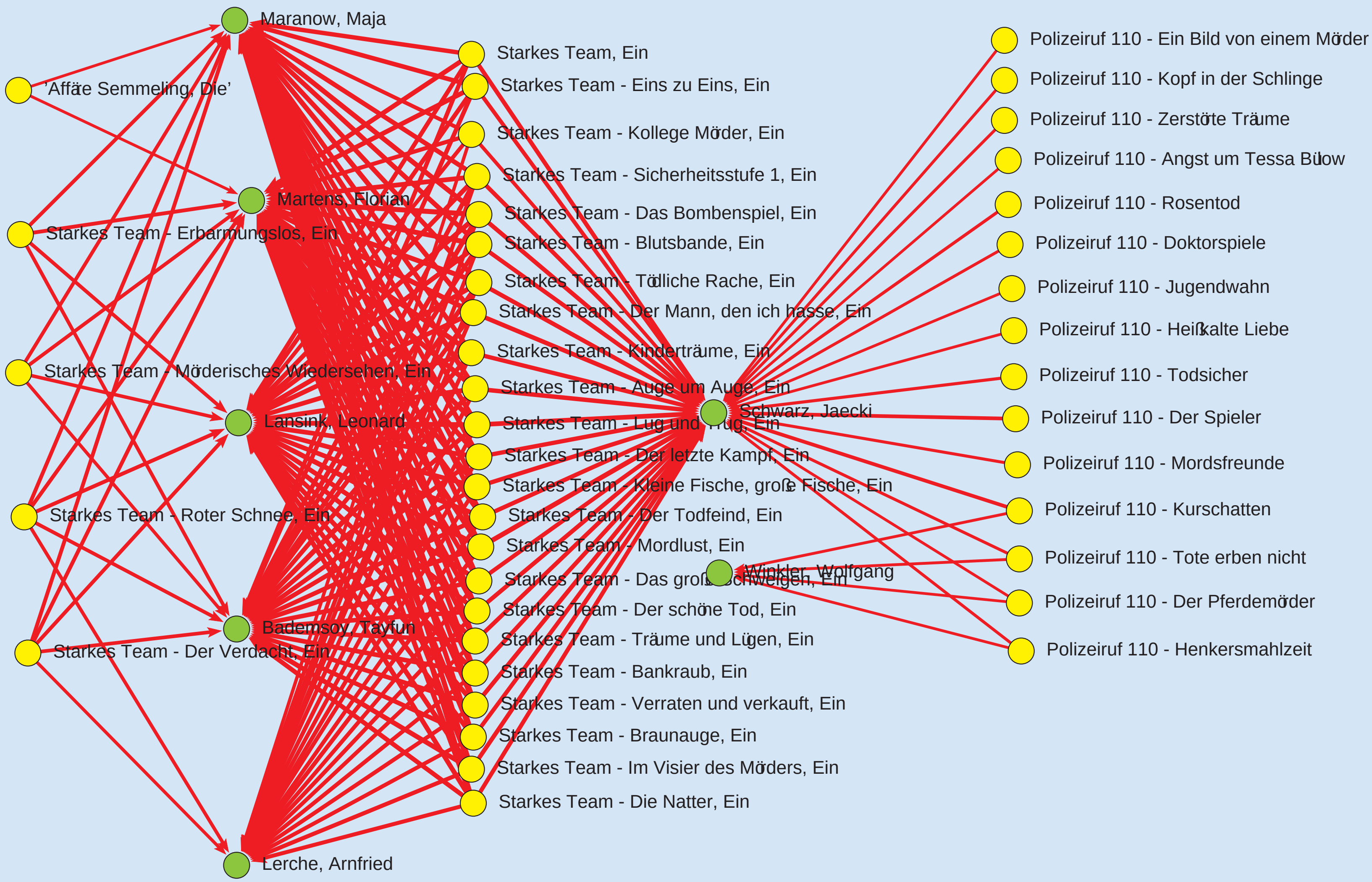




Example: Islands for w_4 / Mark Twain and Abid

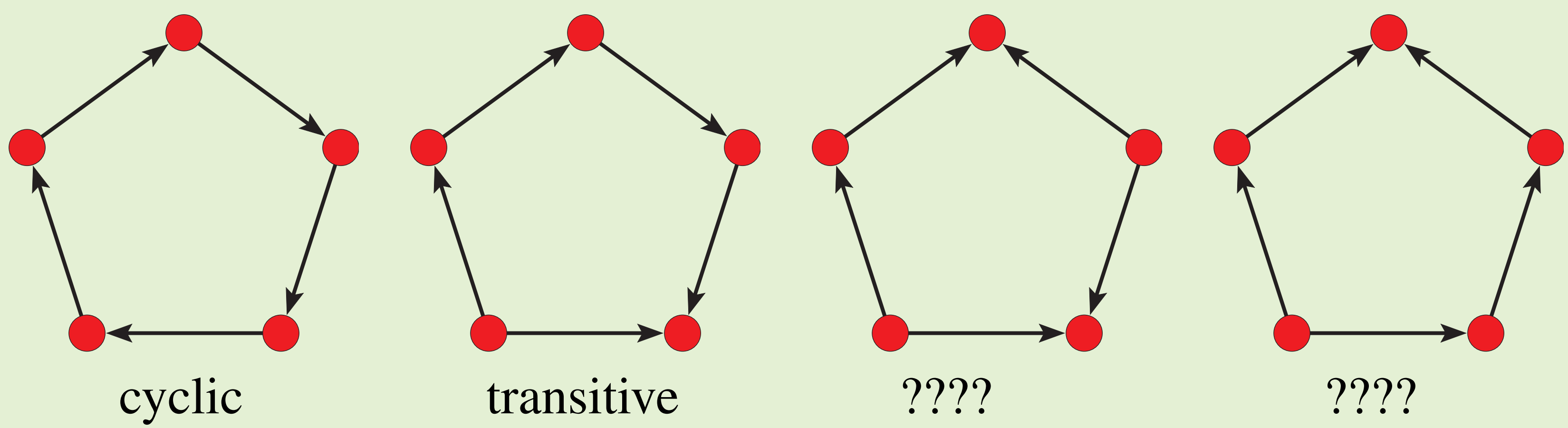


Example: Island for w_4 / Polizeiruf 110 and Starkes Team



5-rings

In the future we intend to implement in **Pajek** also weights w_5. Again there are only 4 types of directed 5-rings.



Multiplication of networks

To a simple two-mode *network* $\mathcal{N} = (\mathcal{I}, \mathcal{J}, \mathcal{E}, w)$; where $\mathcal{I}$ and $\mathcal{J}$ are sets of *vertices*, $\mathcal{E}$ is a set of *edges* linking $\mathcal{I}$ and $\mathcal{J}$, and $w : \mathcal{E} \to \mathbb{R}$ (or some other semiring) is a *weight*; we can assign a *network matrix* $\mathbf{W} = [w_{i,j}]$ with elements: $w_{i,j} = w(i,j)$ for $(i,j) \in \mathcal{E}$ and $w_{i,j} = 0$ otherwise.

Given a pair of compatible networks $\mathcal{N}_A = (\mathcal{I}, \mathcal{K}, \mathcal{E}_A, w_A)$ and $\mathcal{N}_B = (\mathcal{K}, \mathcal{J}, \mathcal{E}_B, w_B)$ with corresponding matrices $\mathbf{A}_{\mathcal{I}\times\mathcal{K}}$ and $\mathbf{B}_{\mathcal{K}\times\mathcal{J}}$ we call a *product of networks* $\mathcal{N}_A$ and $\mathcal{N}_B$ a network $\mathcal{N}_C = (\mathcal{I}, \mathcal{J}, \mathcal{E}_C, w_C)$, where $\mathcal{E}_C = \{(i,j) : i \in \mathcal{I}, j \in \mathcal{J}, c_{i,j} \neq 0\}$ and $w_C(i,j) = c_{i,j}$ for $(i,j) \in \mathcal{E}_C$. The product matrix $\mathbf{C} = [c_{i,j}]_{\mathcal{I}\times\mathcal{J}} = \mathbf{A} * \mathbf{B}$ is defined in the standard way

$$c_{i,j} = \sum_{k\in\mathcal{K}} a_{i,k} \cdot b_{k,j}$$

In the case when $\mathcal{I} = \mathcal{K} = \mathcal{J}$ we are dealing with ordinary one-mode networks (with square matrices).

Fast sparse matrix multiplication

The standard matrix multiplication has the complexity $O(|\mathcal{I}| \cdot |\mathcal{K}| \cdot |\mathcal{J}|)$ – it is too slow to be used for large networks. For sparse large networks we can multiply faster considering only nonzero elements:

for k **in** $\mathcal{K}$ **do**
    for i **in** $N_A(k)$ **do**
        for j **in** $N_B(k)$ **do**
            if $\exists c_{i,j}$ **then** $c_{i,j} := c_{i,j} + a_{i,k} * b_{k,j}$
            else new $c_{i,j} := a_{i,k} * b_{k,j}$

$N_A(k)$: neighbors of vertex k in network $\mathcal{N}_A$

$N_B(k)$: neighbors of vertex k in network $\mathcal{N}_B$

In general the multiplication of large sparse networks is a 'dangerous' operation since the result can 'explode' – it is not sparse.

Complexity of fast sparse matrix multiplication

Let $\mathbf{A}$ and $\mathbf{B}$ be matrices of networks $\mathcal{N}_A = (\mathcal{I}, \mathcal{K}, \mathcal{E}_A, w_A)$ and $\mathcal{N}_B = (\mathcal{K}, \mathcal{J}, \mathcal{E}_B, w_B)$.

Assume that the body of the loops can be computed in the constant time c. Then the complexity of product is

$$C = \sum_{k\in\mathcal{K}} \sum_{i\in N_A(k)} \sum_{j\in N_B(k)} c = c \cdot \sum_{k\in\mathcal{K}} \deg_A(k) \cdot \deg_B(k)$$

Let $\Delta_{\mathcal{K}}^{A} = \max_{k\in\mathcal{K}} \deg_A(k)$ and $\Delta_{\mathcal{K}}^{B} = \max_{k\in\mathcal{K}} \deg_B(k)$ and consider the well known equality

$$\sum_{k\in\mathcal{K}} \deg_A(k) = \sum_{i\in\mathcal{I}} \deg_A(i) = |\mathcal{E}_A|$$

We get $C \leq c \cdot \min(|\mathcal{E}_A| \cdot \Delta_{\mathcal{K}}^{B}, |\mathcal{E}_B| \cdot \Delta_{\mathcal{K}}^{A})$.

If at least one of the sparse networks $\mathcal{N}_A$ and $\mathcal{N}_B$ has small maximal degree on $\mathcal{K}$ then also the resulting product network $\mathcal{N}_C$ is sparse.

More detailed complexity analysis

Let $d_{min}(k) = \min(\deg_A(k), \deg_B(k))$, $\Delta_{min} = \max_{k\in\mathcal{K}} d_{min}(k)$, $d_{max}(k) = \max(\deg_A(k), \deg_B(k))$, $\mathcal{K}(d) = \{k \in \mathcal{K} : d_{max}(k) \geq d\}$, $d^* = \text{argmin}_d(|\mathcal{K}(d)| \leq d)$ and $\mathcal{K}^* = \mathcal{K}(d^*)$. Then $|\mathcal{K}^*| \leq d^*$ and

$$C = c \cdot \sum_{k\in\mathcal{K}} \deg_A(k) \cdot \deg_B(k) = c \cdot \sum_{k\in\mathcal{K}} d_{min}(k) \cdot d_{max}(k)$$

$$= c \cdot (\sum_{k\in\mathcal{K}^*} d_{min}(k) \cdot d_{max}(k) + \sum_{k\in\mathcal{K}\setminus\mathcal{K}^*} d_{min}(k) \cdot d_{max}(k))$$

$$\leq c \cdot (\Delta_{min} \cdot \sum_{k\in\mathcal{K}^*} d_{max}(k) + d^* \cdot \sum_{k\in\mathcal{K}\setminus\mathcal{K}^*} d_{min}(k))$$

$$\leq c \cdot d^* \cdot (\Delta_{min} \cdot \max(|I|, |J|) + \min(|\mathcal{E}_A|, |\mathcal{E}_B|))$$

If for the sparse networks $\mathcal{N}_A$ and $\mathcal{N}_B$ the quantities Δ_{min} and d^* are small then also the resulting product network $\mathcal{N}_C$ is sparse.

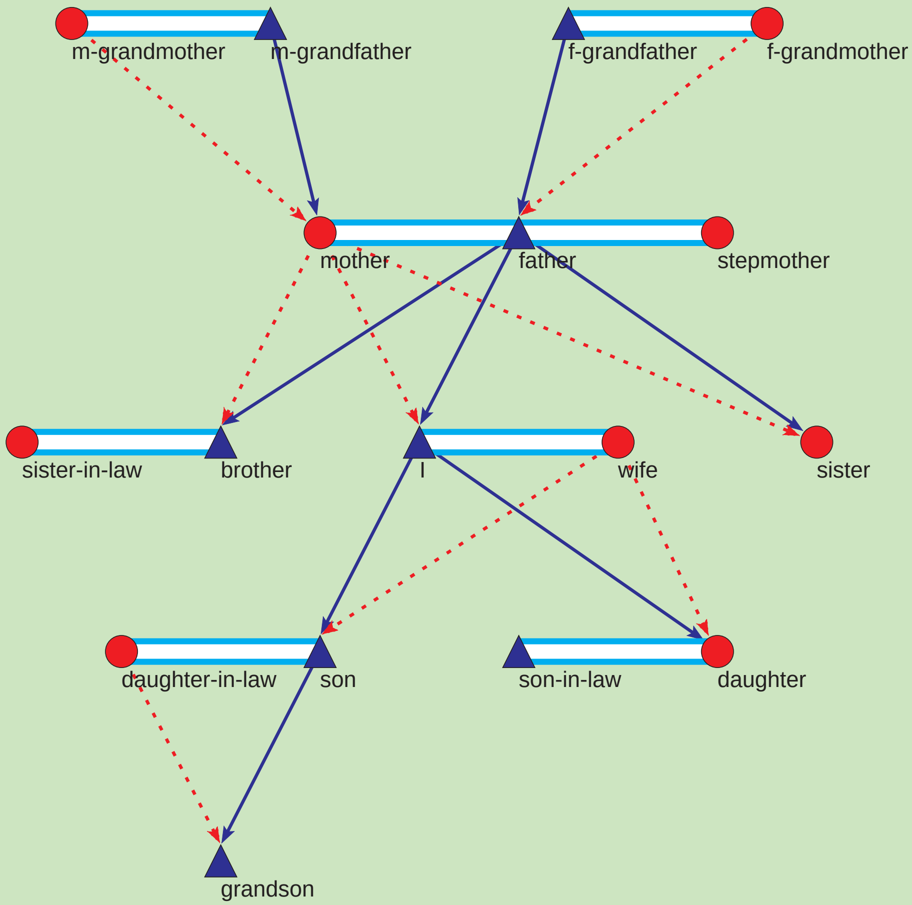
Example: Kinship relations

Anthropologists typically use a basic vocabulary of kin types to represent genealogical relationships. One common version of the vocabulary for basic relationships:

```
Kin Type    English Type
 P          Parent
   F        Father
   M        Mother
 C          Child
   D        Daughter
   S        Son
 G          Sibling
   Z        Sister
   B        Brother
 E          Spouse
   H        Husband
   W        Wife
```

The genealogies are usually described in GEDCOM format. Examples family, Bouchards. Paper

Ore graph



In Ore graph

every person is represented by a vertex,

marriages, relation

_ is a spouse of _ ,

are represented with edges,

and relations

_ is a mother of _

and

_ is a father of _

as arcs pointing from parents to their children.

Calculating kinship relations

Pajek generates three relations when reading genealogy as Ore graph:

F: _ ***is a father of*** _

M: _ ***is a mother of*** _

E: _ ***is a spouse of*** _

Additionally we must generate two binary diagonal matrices, to distinguish between male and female:

L: _ ***is a male*** _ / 1-male, 0-female

J: _ ***is a female*** _ / 1-female, 0-male

$$\mathbf{F} \cap \mathbf{M} = \emptyset, \qquad \mathbf{L} \cup \mathbf{J} \subseteq \mathbf{I}, \qquad \mathbf{L} \cap \mathbf{J} = \emptyset$$

Derived kinship relations

Other basic relations can be obtained using macros based on identities:

_ ***is a parent of*** _	P	$=$	$F \cup M$
_ ***is a child of*** _	C	$=$	P^T
_ ***is a son of*** _	S	$=$	$L * C$
_ ***is a daughter of*** _	D	$=$	$J * C$
_ ***is a husband of*** _	H	$=$	$L * E$
_ ***is a wife of*** _	W	$=$	$J * E$
_ ***is a sibling of*** _	G	$=$	$((F^T * F) \cap (M^T * M)) \setminus I$
_ ***is a brother of*** _	B	$=$	$L * G$
_ ***is a sister of*** _	Z	$=$	$J * G$
_ ***is an uncle of*** _	U	$=$	$B * P$
_ ***is an aunt of*** _	A	$=$	$Z * P$
_ ***is a semi-sibling of*** _	G_e	$=$	$(P^T * P) \setminus I$

and using them other relations can be determined

_ ***is a grand mother of*** _	M_2	$=$	$M * P$
_ ***is a niece of*** _	Ni	$=$	$D * G$

Relative sizes of kinship relations in genealogies

Kin Type	Turks	Ragusa	Loka	Silba	Royal
P-Parent	1.000	1.000	1.000	1.000	1.000
F-Father	0.514	0.532	0.504	0.519	0.540
M-Mother	0.486	0.468	0.496	0.481	0.460
C-Child	1.000	1.000	1.000	1.000	1.000
D-Daughter	0.431	0.384	0.480	0.469	0.427
S-Son	0.569	0.616	0.520	0.531	0.573
G-Sibling	1.250	0.943	1.019	0.811	0.767
Z-Sister	1.135	0.746	0.983	0.760	0.707
B-Brother	1.366	1.140	1.055	0.861	0.828
E-Spouse	0.205	0.215	0.208	0.230	0.306
H-Husband	0.205	0.215	0.208	0.230	0.306
W-Wife	0.205	0.215	0.208	0.230	0.306
U-Uncle	1.920	1.789	1.200	1.181	0.927
A-Aunt	1.750	1.143	1.190	1.097	0.798
Ge-Semi-sibling	1.473	1.155	1.128	0.932	0.905
n	1269	5999	47956	6427	3010
mE = Spouse	407	2002	14154	2217	1138
mA = Parent	1987	9315	68052	9627	3724

Two-mode network analysis by conversion to one-mode network

Often we transform a two-mode network $\mathcal{N} = (\mathcal{U}, \mathcal{V}, \mathcal{E}, w)$ into an ordinary (one-mode) network $\mathcal{N}_1 = (\mathcal{U}, \mathcal{E}_1, w_1)$ or/and $\mathcal{N}_2 = (\mathcal{V}, \mathcal{E}_2, w_2)$, where $\mathcal{E}_1$ and w_1 are determined by the matrix $\mathbf{W}^{(1)} = \mathbf{W}\mathbf{W}^T$, $w^{(1)}_{uv} = \sum_{z\in\mathcal{V}} w_{uz} \cdot w^T_{zv}$. Evidently $w^{(1)}_{uv} = w^{(1)}_{vu}$. There is an edge $(u : v) \in \mathcal{E}_1$ in $\mathcal{N}_1$ iff $N(u) \cap N(v) \neq \emptyset$. Its weight is $w_1(u, v) = w^{(1)}_{uv}$.

The network $\mathcal{N}_2$ is determined in a similar way by the matrix $\mathbf{W}^{(2)} = \mathbf{W}^T\mathbf{W}$.

The networks $\mathcal{N}_1$ and $\mathcal{N}_2$ are analyzed using standard methods.

Normalizations

The *normalization* approach was developed for quick inspection of (1-mode) networks obtained from two-mode networks – a kind of network based data-mining.

In networks obtained from large two-mode networks there are often huge differences in weights. Therefore it is not possible to compare the vertices according to the raw data. First we have to normalize the network to make the weights comparable.

There exist several ways how to do this. Some of them are presented in the following table. They can be used also on other networks.

In the case of networks without loops we define the diagonal weights for undirected networks as the sum of out-diagonal elements in the row (or column) $w_{vv} = \sum_u w_{vu}$ and for directed networks as some mean value of the row and column sum, for example $w_{vv} = \frac{1}{2}(\sum_u w_{vu} + \sum_u w_{uv})$. Usually we assume that the network does not contain any isolated vertex.

. . . Normalizations

$$\mathrm{Geo}_{uv} = \frac{w_{uv}}{\sqrt{w_{uu} w_{vv}}} \qquad \mathrm{GeoDeg}_{uv} = \frac{w_{uv}}{\sqrt{\deg_u \deg_v}}$$

$$\mathrm{Input}_{uv} = \frac{w_{uv}}{w_{vv}} \qquad \mathrm{Output}_{uv} = \frac{w_{uv}}{w_{uu}}$$

$$\mathrm{Min}_{uv} = \frac{w_{uv}}{\min(w_{uu}, w_{vv})} \qquad \mathrm{Max}_{uv} = \frac{w_{uv}}{\max(w_{uu}, w_{vv})}$$

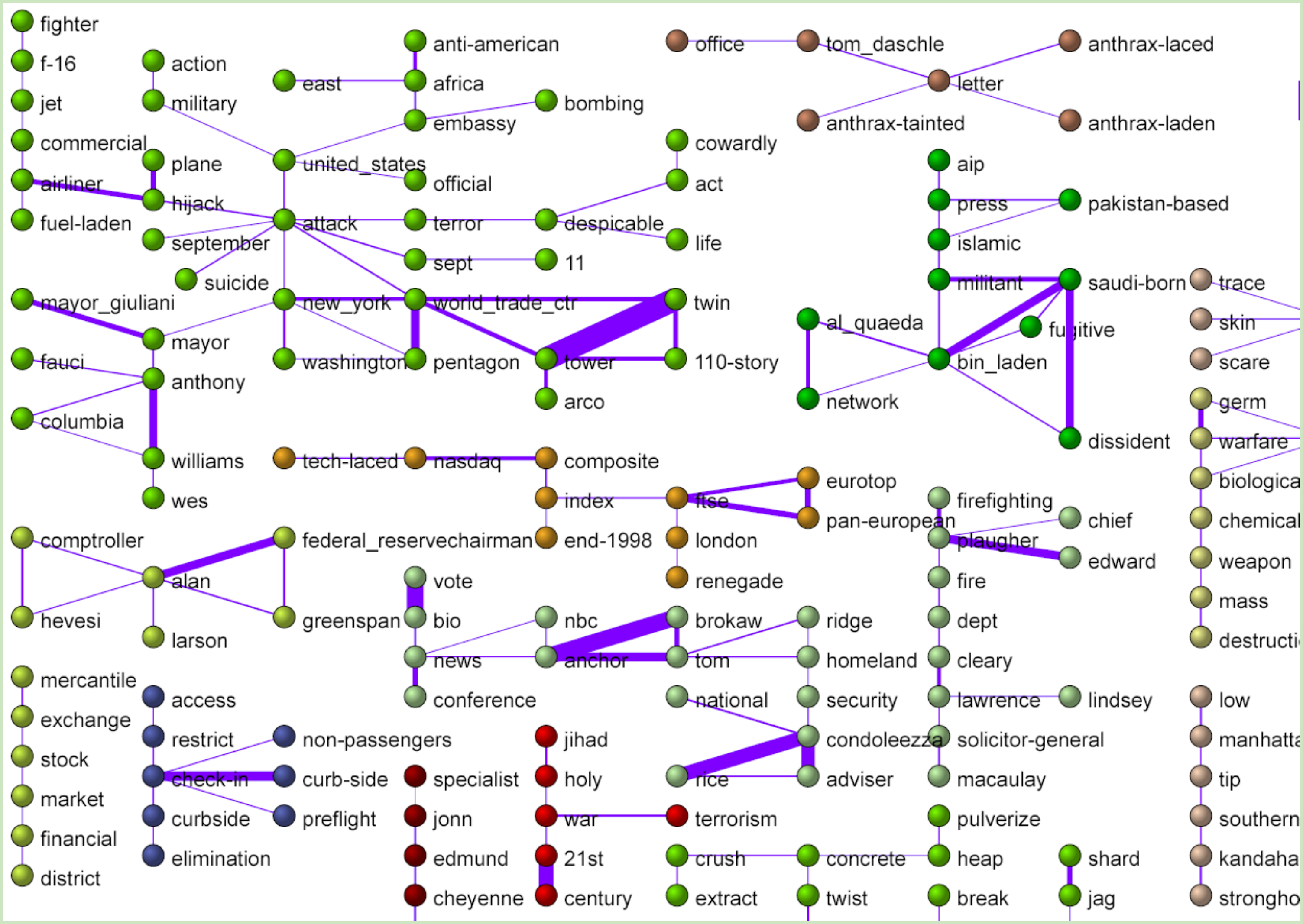
$$\mathrm{MinDir}_{uv} = \begin{cases} \frac{w_{uv}}{w_{uu}} & w_{uu} \leq w_{vv} \\ 0 & \text{otherwise} \end{cases} \qquad \mathrm{MaxDir}_{uv} = \begin{cases} \frac{w_{uv}}{w_{vv}} & w_{uu} \leq w_{vv} \\ 0 & \text{otherwise} \end{cases}$$

After a selected normalization the important parts of network are obtained by line-cuts or islands approaches.

Slovenian journals and magazins.

Reuters Terror News: GeoDeg, MaxDir, MinDir.

GeoDeg normalization of Reuters terror news network



Networks from data tables

RuthDELmain.csv

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
1	Ident	Num	File	ORGANISATION ORI	ORG	Org	Contact Name	Street	ZIP	Project	City	Country	coun	EU	Region
2	1	1480	613.html	3D PLUS SA	3D F	3D I	LIGNIER, Olivier	641 RU	78530	IST-2001-3440	Buc	FRANCE	20	2	ÎLE DE FF
3	2	1481	613.html	3D PLUS SA	3D PLUS		LIGNIER, Olivier	641 Ru	78530	IST-2001-3440	Buc	FRANCE	20	2	ÎLE DE FF
4	3	4001	924.html	3D VISION	3D V	3D '	MARIAT, Jacques	Savoie	73375	502909	Le Bc	FRANCE	20	2	CENTRE-I
5	4	1648	160.html	3D Web Technologies	3D WEB		DENNISON, Andrew		M31 4XL	BMH4989519	Carrir	UNITED KI	60	2	NORTH W
6	5	1406	442.html	3E	3E		PALMERS, Geerl	Eredier	1000	NNE5/51/1999	Bruxe	BELGIQUE	8	2	REG.BRU
7	6	1007	884.html	4M2C PATRIC SALON	4M2C PA		N/A	CRANA	12157	507255	Berlir	DEUTSCH	15	2	BERLIN E
8	7	7914	991.html	5T S.c.r.l.	5T S.C.R		N/A	C.so B	10126	Road2/506716	Torinc	ITALIA	26	2	NORD OV
9	8	6880	588.html	A & C 2000 S.R.L.	A &	SAN	CARLUCCI, Renz	VIALE	148	IST-2001-3454	Roma	ITALIA	26	2	LAZIO Rc
10	9	6881	588.html	A & C 2000 S.R.L.	A & C 20		CARLUCCI, Renz	Viale C	148	IST-2001-3454	Roma	ITALIA	26	2	LAZIO Rc
11	10	1647	176.html	A. BENETTI MACCHII	A. BENE		Federico BENETI	Via Prc	54033	BRST985466	Carra	ITALIA	26	2	CENTRO
12	11	6605	984.html	A. Mickiewicz Univers	A. M	Inst	PATKOWSKI, Ad	Ul. H. \	61-712	502235	Pozn	POLSKA	45	2	
13	12	6571	135.html	A.BRITO - INDUSTRIA	A.BRITO		VIEIRA DE BRITC	5109,E	4350-119	BRST985263	Porto	PORTUGA	46	2	CONTINEI
14	13	1813	409.html	A.L. DIGITAL LIMITED	A.L.	A.L.	LAURIE, Ben	VOYSE	W4 4GB	IST-2000-2633	Chisv	UNITED KI	60	2	SOUTH E.
15	14	1814	409.html	A.L. Digital Limited	A.L. DIG		LAURIE, Ben	Voysey	W4 4GB	IST-2000-2633	Chisv	UNITED KI	60	2	SOUTH E.
16	15	1885	960.html	A.P. MOLLER-MAER	A.P.	TEC	DRAGSTED, Jorr	Esplan	1098	506676	Kope	DANMARK	14	2	Københavı
17	16	6731	537.html	A.S.M. S.A.	A.S.M. S		MOYA GARCIA, .	Carrete	43206	IST-2000-3008	Reus	ESPAÑA	19	2	ESTE CAT
18	17	8150	232.html	AABO AKADEMI UNI	AAB	COI	NYBACKA-WILLI	14-18B	20500	ERK5-CT-1999	Turku	SUOMI/FII	53	2	MANNER-
19	18	8152	662.html	AABO AKADEMI UNI	AAB	DEF	BJORKSTRAND,	3,Tykis	20521	EVK1-CT-2002	Turku	SUOMI/FII	53	2	
20	19	8148	959.html	AABO AKADEMI UNI	AAB	Dep	HUPA, Mikko	Domky	20500	502679	Turku	SUOMI/FII	53	2	MANNER-
21	20	8151	233.html	AABO AKADEMI UNI	AAB	DEF	NYBACKA-WILLI	Lemmi	20500	ERK6-CT-1999	Turku	SUOMI/FII	53	2	MANNER-
22	21	125	116.html	AACHEN UNIVERSIT	AAC	GIE	E. NEUSSL	Intzest	52072	BRPR980663	Aach	DEUTSCH	15	2	NORDRHE
23	22	123	104.html	AACHEN UNIVERSIT	AAC	GIE	MEISER, Lukas	Intzest	52072	BRPR980695	Aach	DEUTSCH	15	2	NORDRHE
24	23	155	364.html	AACHEN UNIVERSIT	AAC	INS	RAUHUT, Burkha	18,Eilfs	52062	G1RD-CT-2000	Aach	DEUTSCH	15	2	NORDRHE

A *data table* $\mathcal{T}$ is a set of *records* $\mathcal{T} = \{T_k : k \in \mathcal{K}\}$, where $\mathcal{K}$ is the set of *keys*. A record has the form $T_k = (k, q_1(k), q_2(k), \ldots, q_r(k))$ where $q_i(k)$ is the value of the *property* (attribute) $\mathbf{q}_i$ for the key k.

…Networks from data tables

Suppose that the property $\mathbf{q}$ has the range $2^{\mathcal{Q}}$. For example: Authors[WasFau] = { S. Wasserman, K. Faust }, PubYear[WasFau] = { 1994 }, …If $\mathcal{Q}$ is finite (it can always be transformed in such set by partitioning the set $\mathcal{Q}$ and recoding the values) we can assign to the property $\mathbf{q}$ a two-mode network $\mathcal{K} \times \mathbf{q} = (\mathcal{K}, \mathcal{Q}, \mathcal{E}, w)$ where $(k, v) \in \mathcal{E}$ iff $v \in q(k)$, and $w(k, v) = 1$.

Also, for properties $\mathbf{q}_i$ and $\mathbf{q}_j$ we can define a two-mode network $\mathbf{q}_i \times \mathbf{q}_j = (\mathcal{Q}_i, \mathcal{Q}_j, \mathcal{E}, w)$ where $(u, v) \in \mathcal{E}$ iff $\exists k \in \mathcal{K} : (q_i(k) = u \wedge q_j(k) = v)$, and $w(u, v) = \mathrm{card}(\{k \in \mathcal{K} : (q_i(k) = u \wedge q_j(k) = v)\})$.

It holds $[\mathbf{q}_i \times \mathbf{q}_j]^T = \mathbf{q}_j \times \mathbf{q}_i$ and $\mathbf{q}_i \times \mathbf{q}_j = [\mathcal{K} \times \mathbf{q}_i]^T * [\mathcal{K} \times \mathbf{q}_j] = [\mathbf{q}_i \times \mathcal{K}] * [\mathcal{K} \times \mathbf{q}_j]$.

We can join a pair of properties $\mathbf{q}_i$ and $\mathbf{q}_j$ also with respect to the third property $\mathbf{q}_s$: we get a two-mode network $[\mathbf{q}_i \times \mathbf{q}_j]/\mathbf{q}_s = [\mathbf{q}_i \times \mathbf{q}_s] * [\mathbf{q}_s \times \mathbf{q}_j]$.

EU projects on simulation

For the meeting ***The Age of Simulation*** at Ars Electronica in Linz, January 2006 a dataset of EU projects on simulation was collected by FAS research, Vienna and stored in the form of Excel table (`SimPro.csv`).

The rows are the projects participants (idents) and colomns correspond to different their properties. Three two-mode networks were produced from this table using Jürgen Pfeffer's **`Text2Pajek`** program:

- `project.net` – $\mathbf{P} = [\text{idents} \times \text{projects}]$
- `country.net` – $\mathbf{C} = [\text{idents} \times \text{countries}]$
- `institution.net` – $\mathbf{U} = [\text{idents} \times \text{institutions}]$

$|\text{idents}| = 8869$, $|\text{projects}| = 933$, $|\text{institutions}| = 3438$, $|\text{countries}| = 60$.

EU projects – network multiplication

Since all three networks have the common set (idents) we can derive from them using *network multiplication* several interesting networks:

- `ProjInst.net` – $\mathbf{W} = [\text{projects} \times \text{institutions}] = \mathbf{P}^T * \mathbf{U}$
- `Countries.net` – $\mathbf{S} = [\text{countries} \times \text{countries}] = \mathbf{C}^T * \mathbf{C}$
- `Institutions.net` – $\mathbf{Q} = [\text{institutions} \times \text{institutions}] = \mathbf{W}^T * \mathbf{W}$
- ...

EU projects – deleted projects

Some projects (27) from original data set have to be deleted – in the final data set `SimPro` they were marked as deleted. When producing two-mode networks we could first physicaly delete them. Instead of this, we used another approach: we produced the cluster C_D of deleted idents and from it a two-mode matrix $\mathbf{D}$ (idents $\times$ idents). Matrix $\mathbf{D}$ is a 'diagonal' matrix with value 1 for idents not belonging to C_D and 0 otherwise. Using matrix $\mathbf{D}$ we can determine the network `ProjInst.net` by $\mathbf{W} = \mathbf{P}^T * \mathbf{D} * \mathbf{U}$ – the deleted idents don't contribute to the network.

Analysis of `ProjInst.net`

For identifying important parts of `ProjInst.net` we first computed the 4-rings weights and in the obtained network we determined the line islands

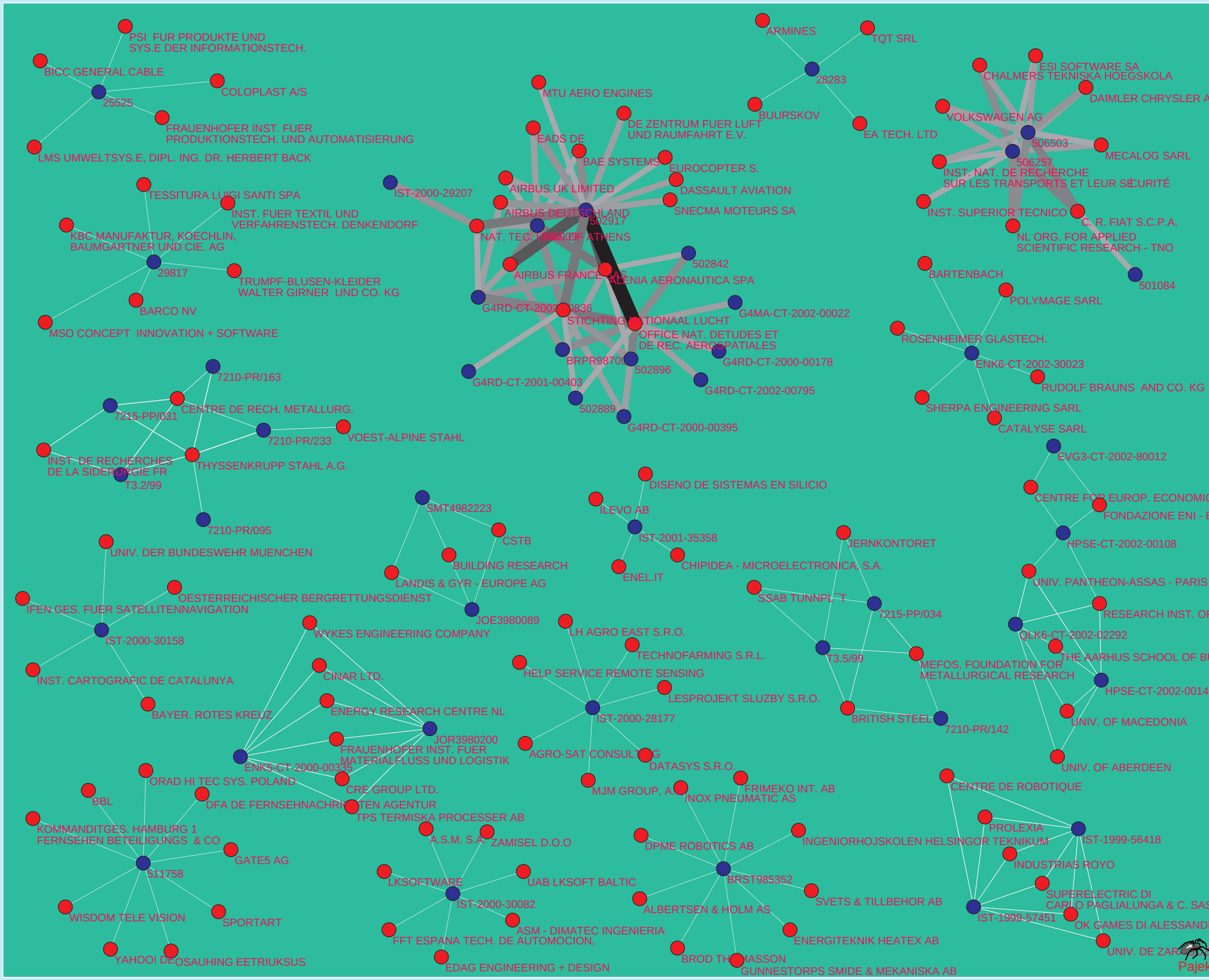
```
Net/Count/4-rings/Undirected
Net/Partitions/Islands/Line Weights[Simple [2,200]
```

We obtain 101 islands. We extracted 18 islands of the size at least 5. There are two most important islands: aviation companies and car companies.

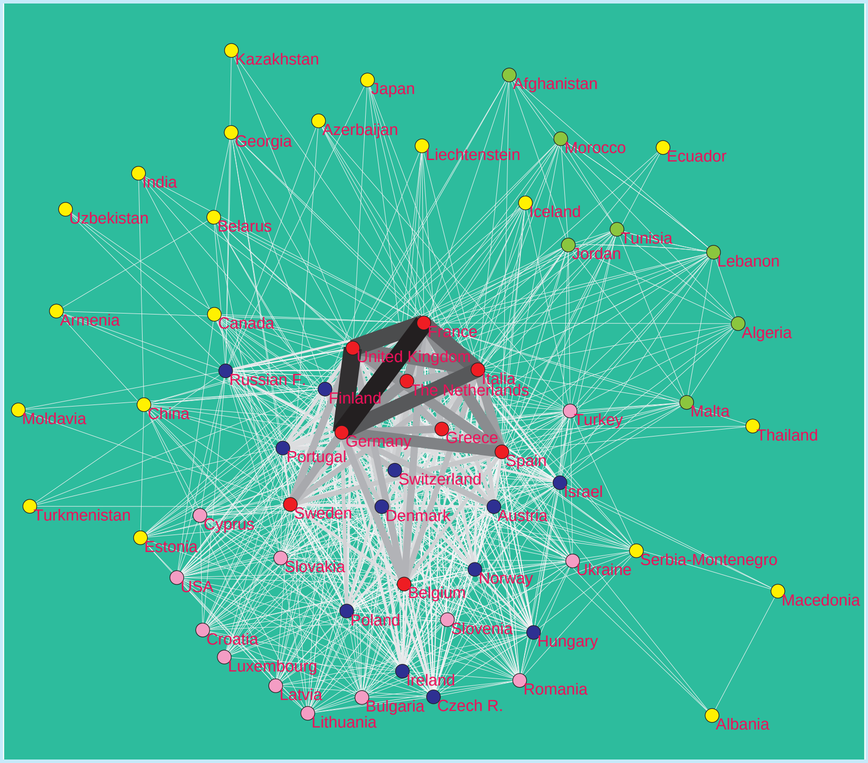
In labels we used a new option `\n`.

For analysis of two-mode networks we can use also $(p, q)-$cores.

Analysis of `ProjInst.net`



Analysis of `Countries.net`



To obtain picture in which the stronger lines cover weaker lines we have to sort them

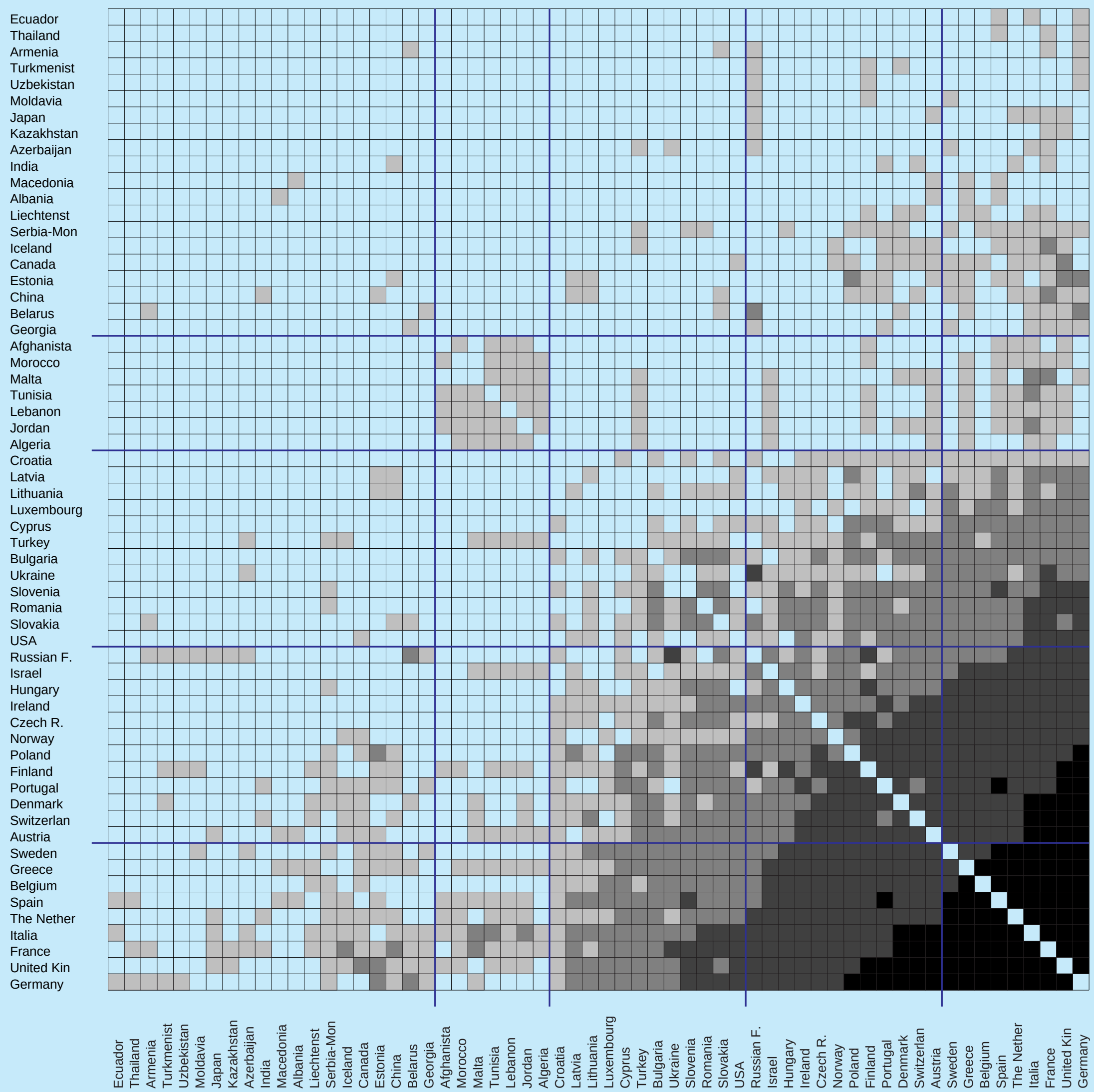
`Net/Transform/Sort lines/Line values/Ascending`

For dense (sub)networks we get better visualization by using matrix display. In this case we also recoded values (2,10,50). To determine clusters we used Ward's clustering procedure with dissimilarity measure d_5 (corrected Euclidean distance).

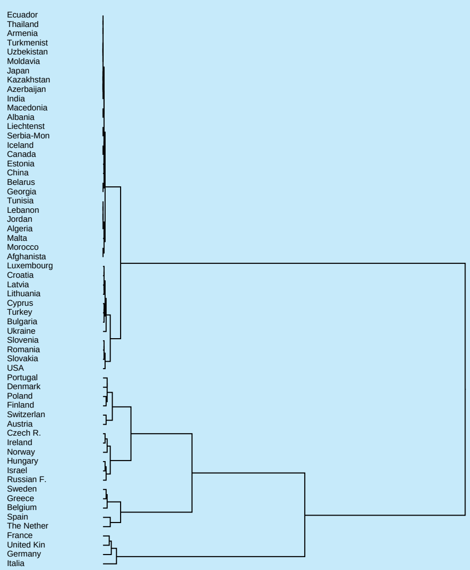
The permutation determined by hierarchy can often be improved by changing the positions of clusters. We get a typical center-periphery structure.

Analysis of Countries.net

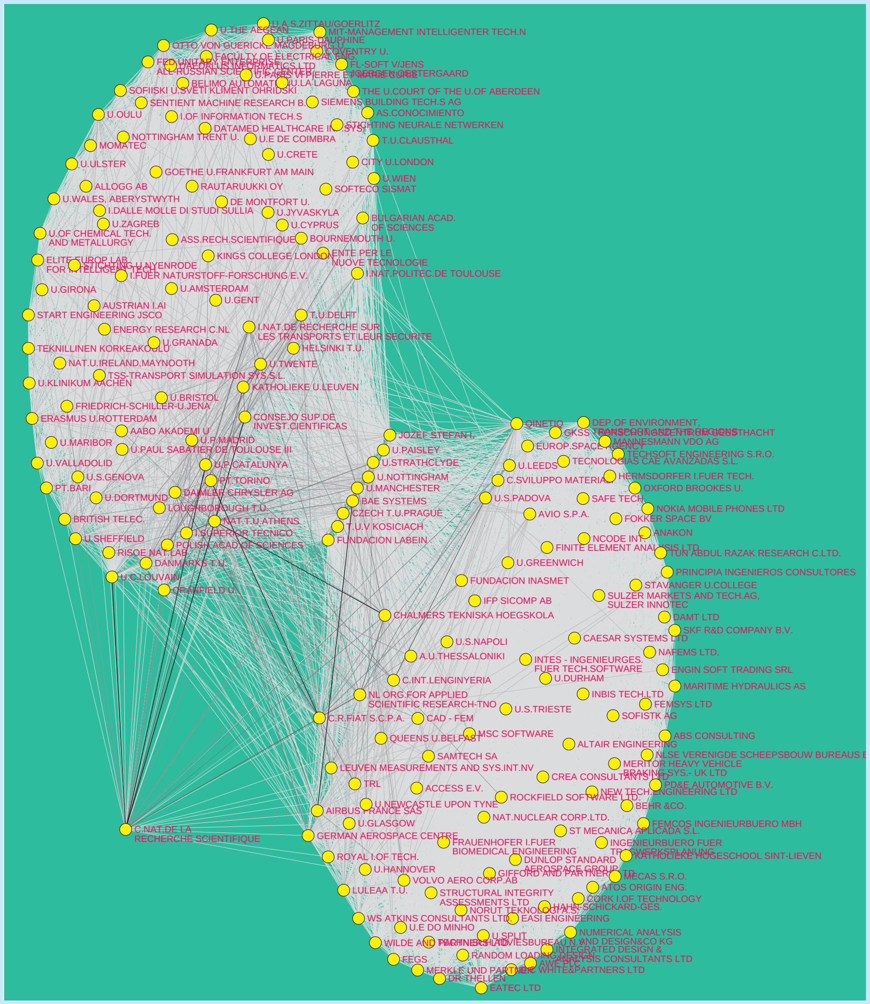
Pajek - shadow [0.00,4.00]



Pajek - Ward [0.00,4785.14]

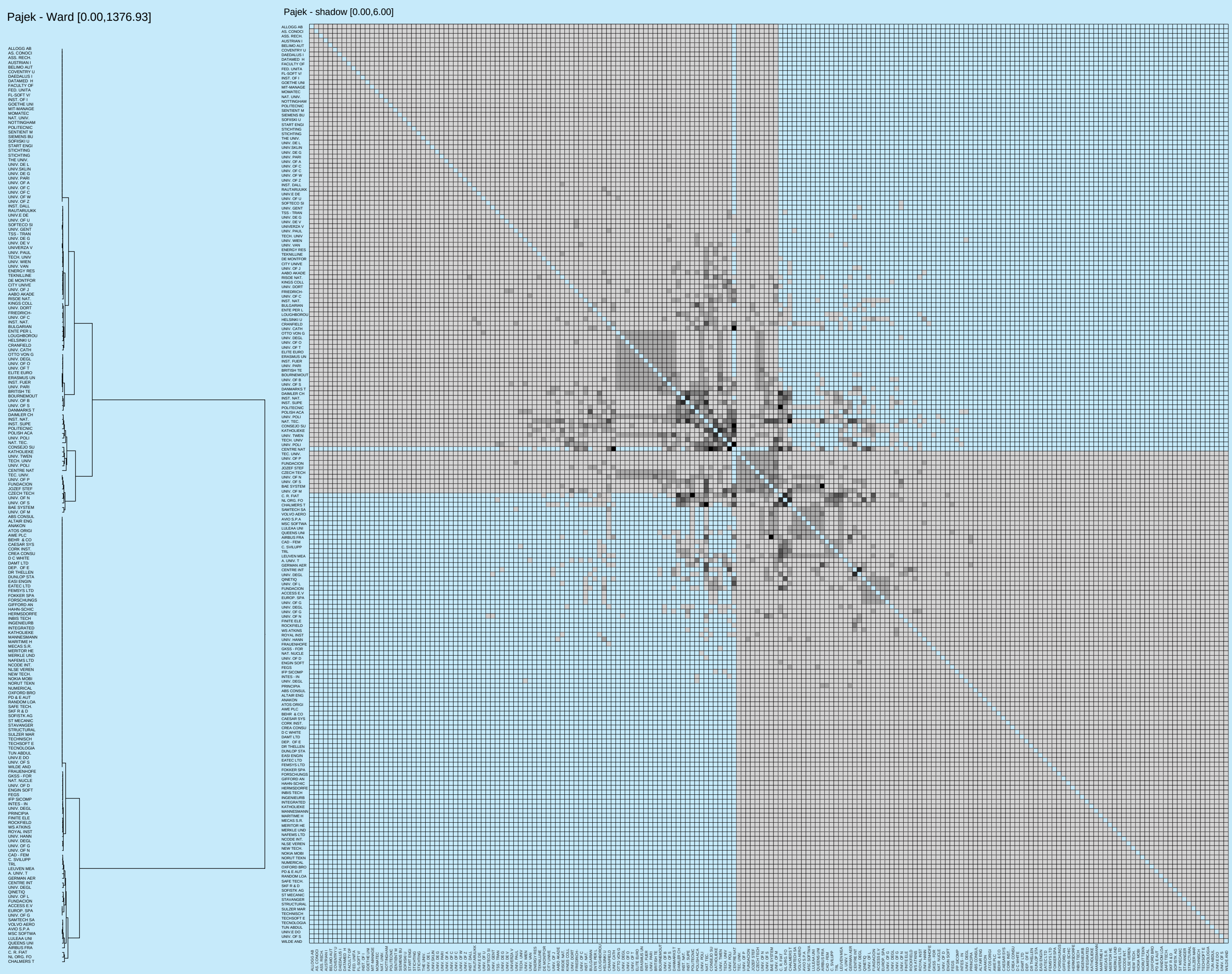


Analysis of `Institutions.net`



To identify the most important institutions we first computed p_S-cores vector and use it to determine the corresponding vertex islands. We got essentially one large island. Again the corresponding subnetwork is very dense. We prepared also a matrix display.

Analysis of `Institutions.net`



Searching on the Web of Science

The Web of Science – WoS (ISI/Thomson) allows us to save on a file the records corresponding to our queries.

For example, using **General search** with a query `"social network*"` we get 6936 hits (27. December 2007).

Trying to save them we are informed that we can save at once at most 500 records. We have to save the records by parts on separate files. At the end we concatenate all these files into a single file.

Program `WoS2Pajek`

For converting WoS file into networks in **`Pajek`**'s format a program **`WoS2Pajek`** was developed (in Python). It produces the following files:

- citation network: works $\times$ works;
- authorship (two-mode) network: works $\times$ authors, for works without complete description only the first author is known;
- keywords (two-mode) network: works $\times$ keywords, only for works with complete description;
- journals (two-mode) network: works $\times$ journals, field `J9`;
- partition of works by the publication year;
- partition of works – complete description (1) / ISI name only (0);
- vector number of pages, `PG` or `EP` $-$ `BP` $+1$.

Program `WoS2Pajek`

The keywords are obtained from the fields `TI` (title), `ID`, `DE` and `AB` (abstract). From the text the stopwords are removed and a list of words is produced. The words are lemmatized using MontyLingua package.

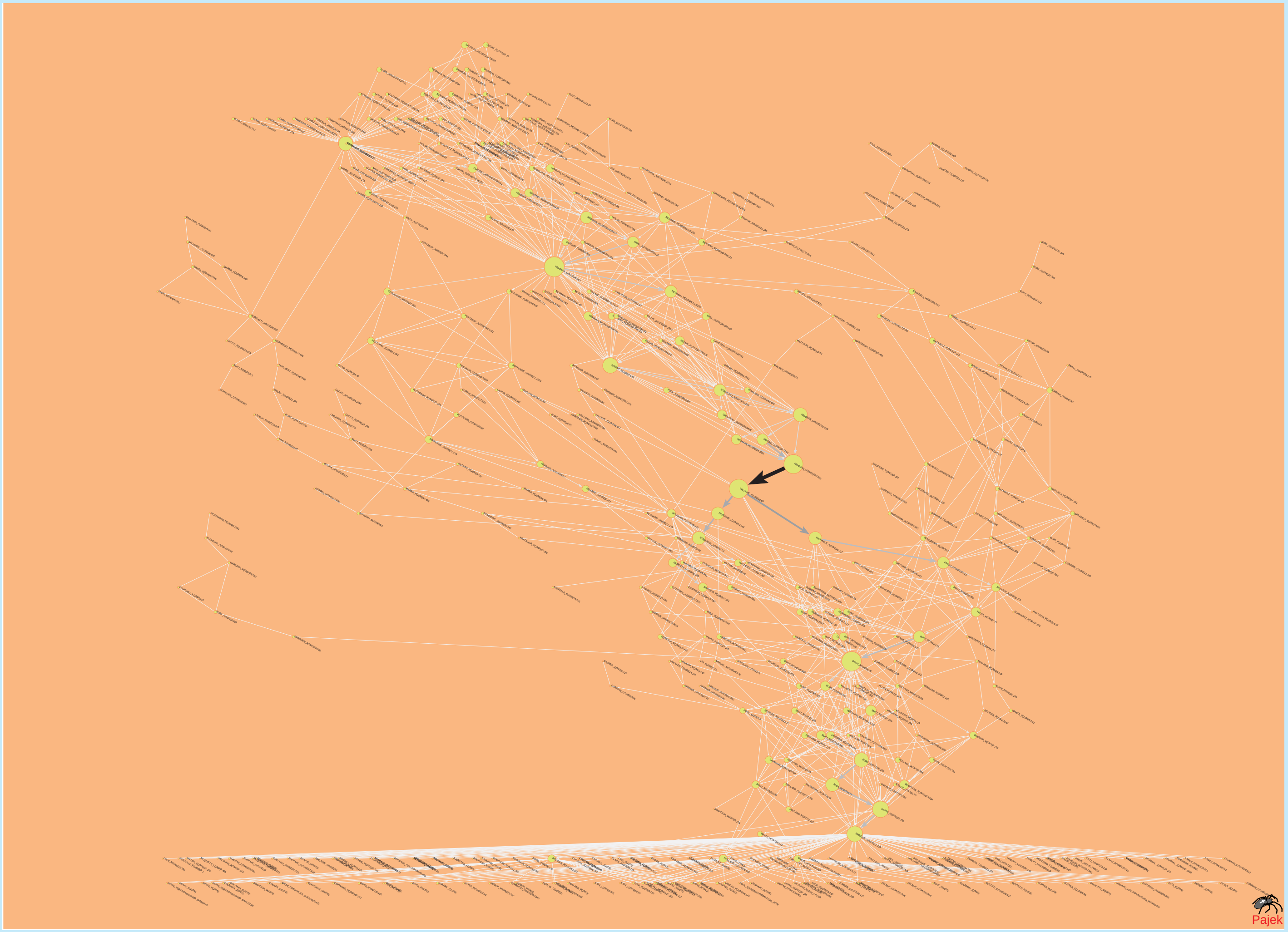
In future versions aditional networks can be derived: works $\times$ discipline, works $\times$ countries, ...

Network `SN5`

SN5 `SO=(SOCIAL NETWORKS)` + `SO=("social network*"` + most frequent references + around 100 social networkers

$|W| = 193376, |C| = 7950, |A| = 75930, |J| = 14651, |K| = 29267$

SN5: Main island in citation network for SPC



Analyses: Derived networks

Let us denote the citation network with $\mathbf{C}$, the (key)words network with $\mathbf{K}$ and the authorship network with $\mathbf{A}$.

Then $\mathbf{A}^T * \mathbf{K}$ is the a two-mode authors $\times$ keywords network – in how many works author a used the (key)word k.

Similary $\mathbf{A}^T * \mathbf{A}$ is the *collaboration network* and $\mathbf{A}^T * \mathbf{C} * \mathbf{A}$ is a network of citations between authors – how many times author a cited author b.

For analysis of the authors $\times$ keywords we used the bipartite (two-mode) cores $(p = 109, q = 113, n_1 = 556, n_2 = 396)$. We deleted words: social, network, study, result, use, 1, 2, 3.

For analysis of the network of citations between authors we applied the islands procedure on lines $[2, 90]$. We obtained 712 islands, the largest of size 51; and for $[10, 90]$ there are 16 islands.

SN5: Islands in authors' citations network