Structural-world conjecture: On functional modules and communities in real-world networks

Lovro Šubelj

University of Ljubljana
Faculty of Computer and Information Science
Slovenia

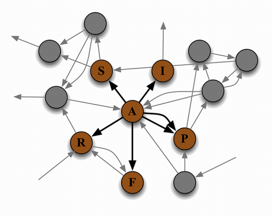
April 30, 2012

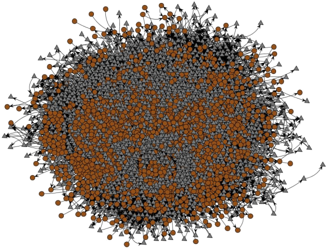
Outline

Motivation

Can network modules explain the structure of software networks?

```
class A extends S implements I {

  F field;

  public A (P parameter) {
    ...
  }

  public R function(P parameter) {
    ...
    return R;
  }

}
```





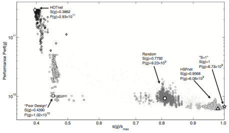
Outline

Degree mixing

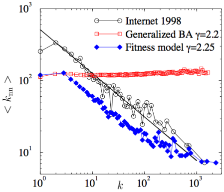
- Degree mixing coefficient $r \in [-1, 1]$. (Newman [30])

$$r = \frac{1}{2m\sigma_k} \sum_{ij} (k_i - k)(k_j - k),$$

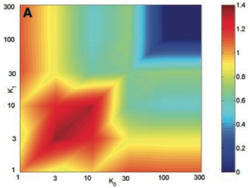
  where σ_k is the standard deviation and k_i degree of node i.
- Assortative mixing refers to $r > 0$, and disassortative to $r < 0$.
- r is simply a Pearson correlation coefficient of k_i at links' ends.



1) s-metric [23]



2) Γ connectivity [38]



3) Correlation profiles [27]

Degree mixing (II)

- Social networks are assortative, while most other are disassortative!

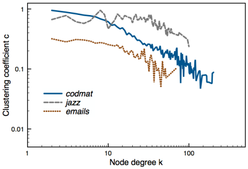
Type	Network	n	m	k	C	D	r
Collaboration	*netsci* [33]	1589	2742	3.5	0.638	0.690	0.462
	condmat [29]	27519	116181	8.4	0.655	0.722	0.166
	comsci [3]	239	568	4.8	0.479	0.561	−0.044
Online social	*pgp* [5]	10680	24316	4.6	0.266	0.317	0.238
Social	*football* [11]	115	613	10.7	0.403	0.419	0.162
	jazz [12]	198	2742	27.7	0.617	0.703	0.020
	dolphins [25]	62	159	5.1	0.259	0.319	−0.044
	karate [58]	34	78	4.6	0.571	0.666	−0.476
Communication	*emails* [14]	1133	5451	9.6	0.220	0.253	0.078
	enron [20]	36692	183831	10.0	0.497	0.530	−0.111
Road network	*euro* [50]	1039	1305	2.5	0.019	0.025	0.090
Power grid	*power* [56]	4941	6594	2.7	0.080	0.100	0.003
Citation	*hepart* [1]	27770	352285	25.4	0.312	0.353	−0.030
Documentation	*javadoc* [49]	2089	7934	7.6	0.373	0.433	−0.070
Protein	*yeast1* [37]	2445	6265	5.1	0.215	0.250	−0.101
	yeast2 [15]	2114	2203	2.1	0.059	0.072	−0.162
Software	*javax* [53]	1595	5287	6.6	0.381	0.440	−0.120
	jung [53]	317	719	4.5	0.366	0.423	−0.190
	guava [54]	174	355	4.1	0.320	0.375	−0.218
	java [53]	1516	10049	13.3	0.685	0.731	−0.283
Web graph	*blogs* [2]	1490	16715	22.4	0.263	0.293	−0.221
Metabolic	*elegans* [16]	453	2025	8.9	0.646	0.710	−0.226
Internet	*oregon* [20]	767	1734	4.5	0.293	0.317	−0.299
Bipartite	*women* [8]	32	89	5.6	0.000	0.000	−0.337

Network clustering

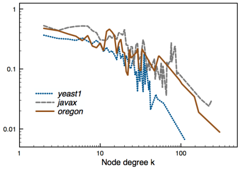
- Network clustering coefficient $C = \frac{1}{n}\sum_i c_i$. (Watts and Strogatz [56])

$$c_i = \frac{t_i}{\binom{k_i}{2}},$$

  where t_i is number of links among Γ_i, $c_i \in [0,1]$.
- For many real-world networks $c_i \sim 1/k_i$. [41, 42, 48]



4) Degree assortative



5) Degree disassortative

- Hub nodes never have high c_i! (Opposite implies a large clique.)

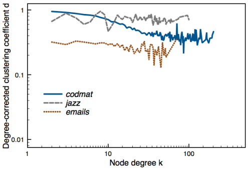
Degree-corrected clustering

- Network degree-corrected clustering co. $D = \frac{1}{n}\sum_i d_i$. (Soffer and Vázquez [46])

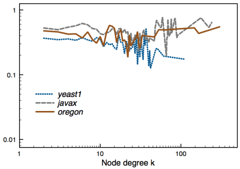
$$d_i = \frac{t_i}{\omega_i},$$

where ω_i is the max. number of links with respect to $\{k_i\}$, $d_i \in [0,1]$.

- Since $\omega_i \leq \binom{k_i}{2}$, $d_i \geq c_i$ and $D \geq C$ by definition.



6) Degree assortative

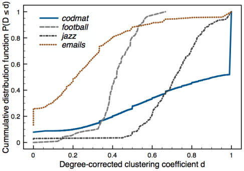


7) Degree disassortative

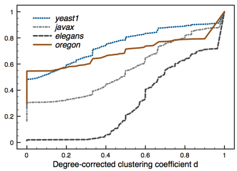
- For pseudo-fractal model $c_i \sim 1/k_i$ implies $c_i \sim 1/\log k_i$. [46]

Degree-corrected clustering (II)

- Most nodes in assortative networks share similar $d_i \gg 0$, whereas 30-55% of nodes in disassortative networks have $d_i \approx 0$!

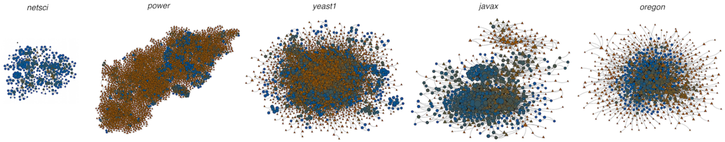


8) Degree assortative



9) Degree disassortative

- d_i appear to capture certain characteristics of the underlying domain.



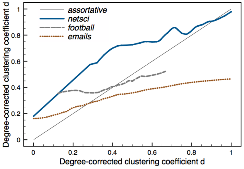
Clustering mixing

- Define clustering mixing coefficients $r_c, r_d \in [-1, 1]$. (Šubelj and Bajec [54])

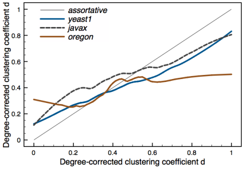
$$r_d = \frac{1}{2m\sigma_d} \sum_{ij} (d_i - D)(d_j - D),$$

where σ_d is the standard deviation. (Similarly for r_c.)

- Contrary to r_c, $r_d \gg 0$ in real-world networks!



10) Degree assortative



11) Degree disassortative

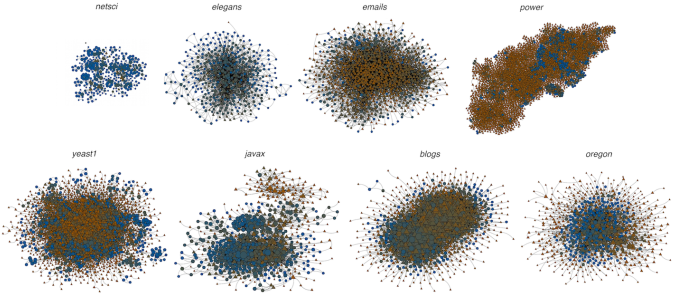
Clustering mixing (II)

Type	Network	n	m	k	C	D	r	r_c	r_d	$d_i < p_r$	$d_i < p_c$
Collaboration	*netsci* [33]	1589	2742	3.5	0.638	0.690	0.462	0.442	0.679	1%	1%
	condmat [29]	27519	116181	8.4	0.655	0.722	0.166	0.116	0.291	1%	1%
	comsci [3]	239	568	4.8	0.479	0.561	−0.044	0.123	0.355	6%	6%
Online social	*pgp* [5]	10680	24316	4.6	0.266	0.317	0.238	0.497	0.632	27%	27%
Social	*football* [11]	115	613	10.7	0.403	0.419	0.162	0.369	0.385	0%	0%
	jazz [12]	198	2742	27.7	0.617	0.703	0.020	0.008	0.198	1%	1%
	dolphins [25]	62	159	5.1	0.259	0.319	−0.044	0.192	0.234	15%	15%
	karate [58]	34	78	4.6	0.571	0.666	−0.476	−0.229	0.277	3%	6%
Communication	*emails* [14]	1133	5451	9.6	0.220	0.253	0.078	0.214	0.317	14%	15%
	enron [20]	36692	183831	10.0	0.497	0.530	−0.111	0.185	0.379	4%	4%
Road network	*euro* [50]	1039	1305	2.5	0.019	0.025	0.090	0.395	0.499	91%	91%
Power grid	*power* [56]	4941	6594	2.7	0.080	0.100	0.003	0.469	0.653	74%	74%
Citation	*hepart* [1]	27770	352285	25.4	0.312	0.353	−0.030	0.132	0.370	6%	6%
Documentation	*javadoc* [49]	2089	7934	7.6	0.373	0.433	−0.070	0.090	0.440	9%	9%
Protein	*yeast1* [37]	2445	6265	5.1	0.215	0.250	−0.101	0.372	0.534	29%	29%
	yeast2 [15]	2114	2203	2.1	0.059	0.072	−0.162	0.576	0.675	68%	68%
Software	*javax* [53]	1595	5287	6.6	0.381	0.440	−0.120	−0.041	0.545	17%	17%
	jung [53]	317	719	4.5	0.366	0.423	−0.190	0.092	0.443	21%	21%
	guava [54]	174	355	4.1	0.320	0.375	−0.218	0.075	0.734	34%	34%
	java [53]	1516	10049	13.3	0.685	0.731	−0.283	−0.574	0.536	1%	100%
Web graph	*blogs* [2]	1490	16715	22.4	0.263	0.293	−0.221	−0.057	0.308	8%	13%
Metabolic	*elegans* [16]	453	2025	8.9	0.646	0.710	−0.226	−0.240	0.183	1%	3%
Internet	*oregon* [20]	767	1734	4.5	0.293	0.317	−0.299	−0.231	0.262	35%	70%
Bipartite	*women* [8]	32	89	5.6	0.000	0.000	−0.337			100%	100%

$p_r = \frac{k}{n-1}$ and $p_c = \frac{(\sum_i k_i^2 - nk)^2}{n^3 k^3}$, while percentages ignore nodes with $k_i \leq 1$.

Clustering assortativity

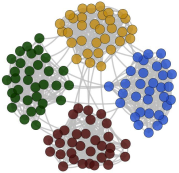
- $r_d \gg 0$ in real-world networks! ($r_c < 0$ in disassortative networks.)
- $d_i \approx 0$ and $r_d \gg 0$ imply connected regions with no clustering.
- r_d captures how well separated are different network modules.



- $r_d \nrightarrow 0$ when $n \to \infty$ in a random graph, however, $D \approx 0$.

Network modules

- Let community be a densely linked group of nodes that are sparsely linked with the rest of the network.
  - An artifact of homophily [28, 34] or triadic closure [13] in social networks.
  - Result in degree assortativity, as long as their sizes differ. [36]
- Recent work shows that communities are an artifact of clustering. [10]
- There is substantial evidence that communities appear concurrently with high clustering and assortative mixing by degree. [31, 21, 57]



- Non-social real-world networks deviate from this picture!

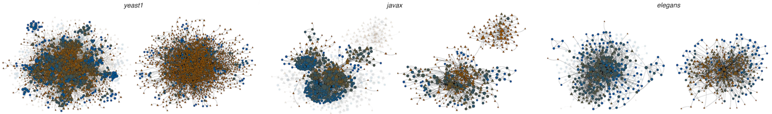
Network modules (II)

- Most real-world networks contain at least some communities.
- Community extraction: (Zhao et al. [59])
  1. generate a pool of candidate communities,
  2. extract community S with the highest value of W,

$$W = s(n-s)\left(\frac{\sum_{i \in S} k_i^S}{s^2} - \frac{\sum_{i \in S} k_i - k_i^S}{s(n-s)}\right),$$

  where k_i^S and $k_i - k_i^S$ are internal and external degree of node i.

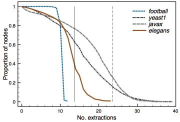
  3. repeat step 1. until W drops below the value expected at random.
- Extract only the links within S, but not those between S and S^C!



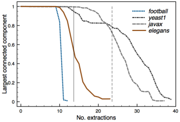
Communities overlaid over original networks and networks after extraction, respectively.

Network modules (III)

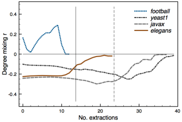
- After extraction of communities ≈ 80% nodes remain!
- Network structure beyond communities is characterized by:
  - disassortative mixing by degree,
  - lower (degree-corrected) clustering,
  - short distances between the nodes.



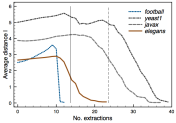
12) # nodes n



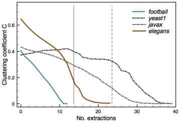
13) LCC



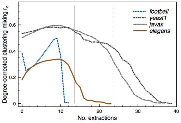
14) Mixing r



15) Distances l



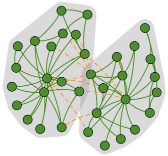
16) Clustering C



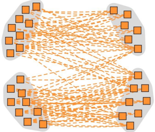
17) Mixing r_d

Network modules (IV)

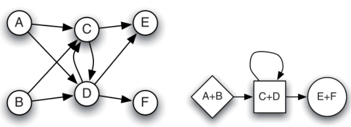
- Are there mesoscopic structures that could explain these properties?
- Let functional module be a group of nodes with common neighbors.



18) Communities



19) Func. modules

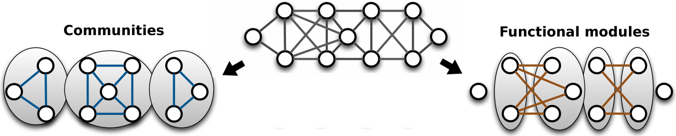


20) Role models [43]

- Functional modules coincide with groups of regularly equivalent nodes.
- Functional modules should result in:
  - disassortative mixing by degree, as long as their sizes differ,
  - low (degree-corrected) clustering (absence of triangles),
  - short distances between the nodes (efficient global navigation).

Structural-world conjecture

- Structural-world conjecture:
  Real-world networks are composed of functional modules characterizing different functions within the system, and overlaid by communities based on some assortative property (and noise).



- Functional modules explain degree disassortativity and efficient long-range navigation, whereas communities increase overall clustering and degree assortativity and explain efficient short-range navigation.
- Structural-world networks must necessarily be heterogeneous!

Note that degree disassortativity and low clustering are already expected properties of scale-free networks.

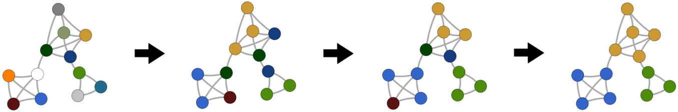
Outline

Label propagation

- Let g_i be unknown node (module) labels.
- Label propagation algorithm (LPA): (Raghavan et al. [40])
  1. initialize nodes with unique labels, $g_i = i$,
  2. node i adopts the label shared by most in Γ_i,

$$g_i = \operatorname*{argmax}_g \sum_{j \in \Gamma_i} \delta(g_j, g),$$

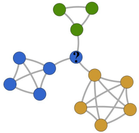
  3. repeat step 2. until convergence.



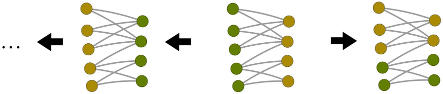
- Algorithm has near linear time complexity $\mathcal{O}(m^{1.2})$. [51]

Label propagation (II)

- Convergence issues for, e.g., overlapping communities.
  $\hookrightarrow$ g_i is retained, when among most frequent in Γ_i.



- Oscillation of labels in, e.g., bipartite networks.
  $\hookrightarrow$ g_i are updated in a random order (sequentially).



- Results can be improved by applying node preferences f_i. [22]

$$g_i = \underset{g}{\operatorname{argmax}} \sum_{j \in \Gamma_i} f_j \cdot \delta(g_j, g)$$

BALANCED PROPAGATION

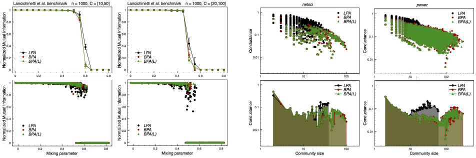
- Balanced propagation algorithm (BPA): (Šubelj and Bajec [50])

$$g_i = \operatorname*{argmax}_{g} \sum_{j \in \Gamma_i} b_j \cdot \delta(g_j, g),$$

where $b_i = \frac{1}{1+e^{-\eta(i_i-\lambda)}}$ (or $b_i = i_i$) and i_i is index of i, $i_i \in (0,1]$.

- Algorithm retains scalability and improves stability and performance.

Algorithm	# distinct in 1000 partitions					
	karate	*dolphins*	*books*	*football*	*jazz*	*elegans*
LPA	184	525	269	414	63	707
BPA	19	36	29	154	20	75

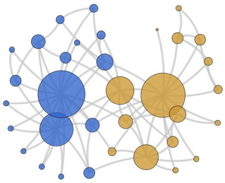


Defensive propagation

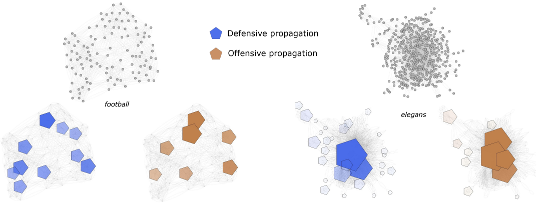
- Defensive propagation algorithm (DPA): (Šubelj and Bajec [51])

$$g_i = \operatorname*{argmax}_g \sum_{j \in \Gamma_i} p_j \cdot \delta(g_j, g),$$

where p_i is the probability of a random walker utilized on g_i.



23) Community cores



24) Defensive and offensive propagation

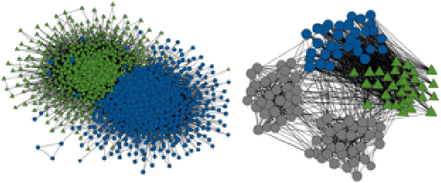
- Defensive and offensive prop. obtain high "recall" and "precision".

General propagation

- Label propagation can detect only connected (cohesive) modules.
- For functional modules, labels must be propagated through neighbors!
- General propagation algorithm (GPA): (Šubelj and Bajec [55])

$$g_i = \operatorname*{argmax}_g \left(\nu_g \sum_{j \in \Gamma_i} f_j \cdot \delta(g_j, g) + (1 - \nu_g) \sum_{j \in \Gamma_i} \sum_{l \in \Gamma_j \setminus \Gamma_i} \tilde{f}_l / k_j \cdot \delta(g_l, g) \right),$$

where $\nu_g \in [0, 1]$ are parameters and $f_i = b_i p_i$ (similarly for $\tilde{f}_i$).

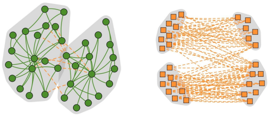


- ν_g are ≈ 1 and ≈ 0 for communities and functional modules, resp.

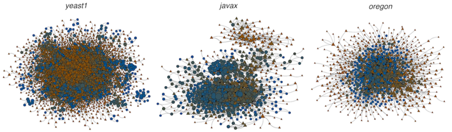
General propagation (II)

- Modeling of ν_g is of vital importance (guides the algorithm).
  - Dynamic based on conductance Φ. (Šubelj and Bajec [55])
  - Dynamic based on clustering C. (Šubelj and Bajec [52])
- Simple model based on clustering D (and mixing r_d): (Šubelj and Bajec [54])

$$\nu_{g_i} = \begin{cases} 1 & \text{for } d_i \geq p_c \ (D \geq p_c), \\ 0 & \text{for } d_i < p_c \ (D < p_c), \\ 0.5 & \text{otherwise.} \end{cases}$$



25) $d_i \geq p_c$ or $d_i < p_c$.

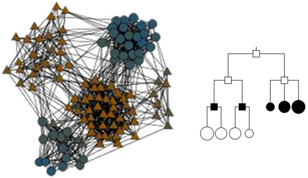


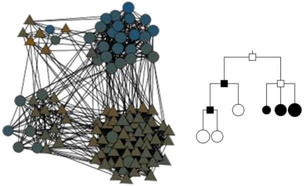
26) $d_i \geq p_c$ and $d_i < p_c$!

- ν_g seem to ignore most functional modules (structural-worlds)!

Hierarchical propagation

- k-partite network on n nodes becomes a clique when $k \to n$ or $n \to k$.
- Functional modules can be obscure in the presence of communities!
- How community detection algorithms identify functional modules?



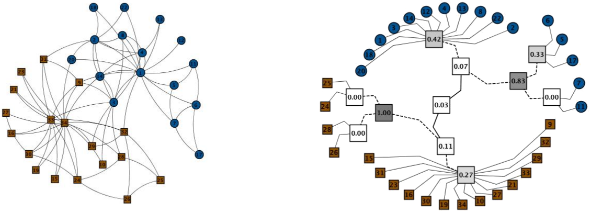


$\hookrightarrow$ Functional modules can be identified as a community and refined.

- Note that functional modules must be detected "twice".

Hierarchical propagation (II)

- Hierarchical propagation algorithm (HPA): (Šubelj and Bajec [54])
  1. partition the network into modules using GPA,
  2. refine each module (step 1.) and accept refinements that increase $\mathcal{L}$,
  3. repeat step 1. on a super-network induced by initial modules.
- Algorithm reveals entire hierarchy $\mathcal{H}$, where $\mathcal{L}$ is the likelihood of $\mathcal{H}$.



Bottom-most level of $\mathcal{H}$ is reported for module detection.

- Time complexity for each level of $\mathcal{H}$ can be estimated to $\mathcal{O}((km)^{1.2})$.

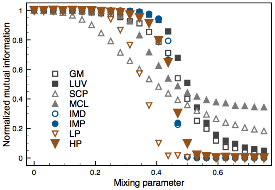
Hierarchical propagation (III)

- Single algorithm for communities and functional modules.
- No prior knowledge is required (e.g., number of modules)!
- Algorithm uses only local information (parallelization).
- Relatively simple to extend (e.g., prior knowledge).
- Time complexity is near ideal $\mathcal{O}(km)$!
- Relatively simple to implement.

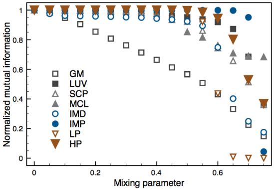
Outline

Community detection

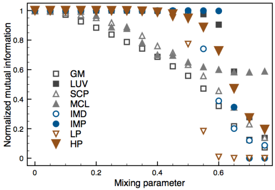
Community detection algorithms: greedy modularity [32, 6] (GM), multi-stage modularity [4] (LUV), sequential clique percolation [18] (SCP), Markov clustering [47] (MCL), Infomod [45] (IMD), Infomap [44] (IMP), label propagation [40] (LP) and hierarchical propagation [54] (HP).



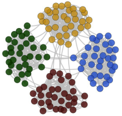
27) (Girvan and Newman [11])



28) (Lancichinetti et al. [19]) (small)

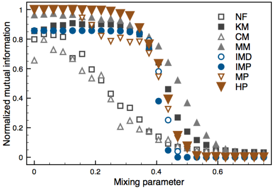


29) (Lancichinetti et al. [19]) (big)

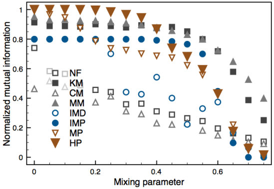


Module detection

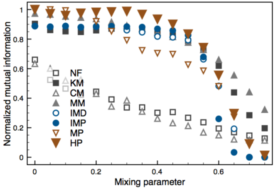
Module detection algorithms: matrix factorization [9] (NF), *k*-means [26] based on [24] (KM), mixture model [35] (MM), degree-corrected mixture model [17] (CM), Infomod [45] (IMD), Infomap [44] (IMP), model propagation [52] (MP) and hierarchical propagation [54] (HP).



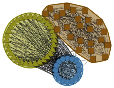
30) (Pinkert et al. [39])



31) (Šubelj and Bajec [54]) (HN6)



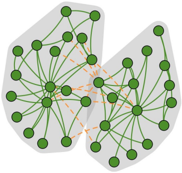
32) (Šubelj and Bajec [54]) (HN7)



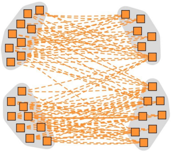
Real-world networks

Algorithms: multi-stage modularity [4] (LUV), mixture model [35] (MM), classical propagation [54] (CP) and hierarchical propagation [54] (HP).

Network	NMI				ARI			
	LUV	MM	CP	HP	LUV	MM	CP	HP
football	0.876	0.823	0.905	**0.909**	0.771	0.683	0.841	**0.850**
karate	0.629	**0.912**	0.834	0.866	0.510	**0.912**	0.823	0.861
jung	0.605	0.662	0.650	**0.684**	0.269	0.276	0.218	**0.280**
women	0.309	0.825	0.217	**0.932**	0.174	0.716	0.119	**0.936**

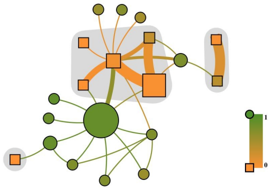


33) Zachary karate net.

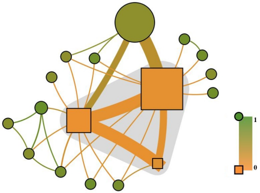


34) Davis women net.

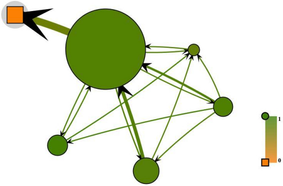
Real-world networks (II)



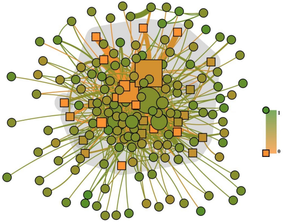
35) jung software network



36) javax software network



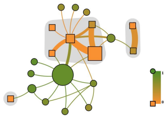
37) Amazon web graph



38) Protein interactions

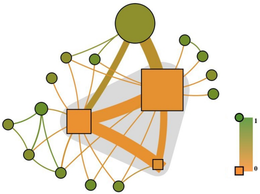
Real-world networks (III)

Network	Module	n	$1-\Phi$	Description
jung	Core community	65	0.86	[jung.visualization.] *(Server|Viewer|Pane|Model|Context) (9); control.* (4) control.*Control (5); layout.* (7); picking.*State (3); picking.*Support (6); renderers.*Renderer (13); renderers.*Support (3); etc.
	5-conf. (upper left)	3	0.00	[jung.algorithms.filters.] *Filter (3).
	5-conf. (upper right)	21	0.33	[jung.graph.] *(Graph|Multigraph|Tree) (18); etc.
	5-conf. (central)	28	0.07	[jung.] algorithms.generators.*Generator (2); algorithms.importance.* (4) algorithms.layout.*Layout* (3); algorithms.scoring.*Scorer (2); algorithms.shortestpath.* (2); graph.*(Graph|Tree|Forest) (4); etc. (interfaces)
	5-conf. (lower left)	13	0.00	[jung.algorithms.] layout.*Layout* (7); layout3d.*Layout (3); etc.
	5-conf. (lower right)	44	0.03	[jung.] algorithms.cluster.*Clusterer* (4); algorithms.generators.random.*Generator (5); algorithms.importance.*Betweenness* (3); algorithms.metrics.* (3); algorithms.scoring.** (5); algorithms.shortestpath.* (5); graph.util.* (7); etc. (implementations)
	2-conf. (upper)	13	0.03	[jung.io.graphml.] parser.*Parser (10); etc.
	2-conf. (lower)	13	0.38	[jung.io.graphml.] *Metadata (8); etc.
	1-conf. (central)	2	0.00	[jung.visualization.control.] *Plugin (2).



Real-world networks (IV)

Network	Module	n	$1-\Phi$	Description
javax	Core community	179	0.64	[javax.swing.] plaf.*UI (24); plaf.basic.Basic*UI (42); plaf.metal.Metal*UI (22); plaf.multi.Multi*UI (30); plaf.synth.Synth*UI (40); etc.
	3-conf. (upper)	193	0.15	[javax.] accessibility.Accessible* (10); swing.J* (41); swing.**(Border|Borders|Box|Button|Dialog|Divider|Editor|Factory|Filter|Icon|Kit|LookAndFeel|Listener|Model|Pane|Panel|Popup|Renderer|UIResource|View) (92); etc.
	3-conf. (left)	113	0.11	[javax.] accessibility.Accessible* (6); swing.* (34); swing.event.*Event (8); swing.event.*Listener (13); swing.plaf.*UI (6); etc.
	3-conf. (lower)	44	0.19	[javax.swing.] text.*View (15); text.html.*View (16); etc.



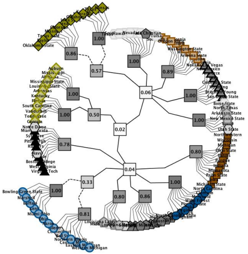
Structure prediction

- How well the model fits the network observed?
- Not link prediction (functional modules sensitive to link removal)!

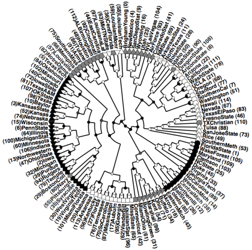
Network			$-\log\mathcal{L}$			
	Groups		LUV	MM	CP	HP
netsci			2622.3		**2024.7**	2152.0
comsci			2399.7		**2011.0**	3326.5
football	12	1184.3	1138.5	1351.7	1096.0	**1095.2**
karate	2	196.3	*178.0*	195.8	188.2	**186.9**
euro			5954.1		4202.1	**4072.3**
power			33874.6		21132.9	**20678.8**
javadoc	22	23011.5	19943.4	25361.7	19405.7	**19346.0**
yeast1			32621.3		**28280.9**	28689.4
yeast2			15766.8		12826.9	**12181.3**
javax	21	16140.7	13728.9	17299.2	13497.5	**13063.5**
jung	38	3081.6	2583.8	2997.5	2497.4	**2489.6**
elegans			*8673.5*		8955.7	**8856.9**
oregon			*9351.5*		**9865.6**	10456.6
women	3	193.3	204.2	*163.6*	207.4	**183.8**

Structure prediction (II)

Network	Runs	$-\log\mathcal{L}$ and # levels						(Clauset et al. [7])	
		CP		HP—p_r and p_c					
football	10^4	1010.9	3	**954.8**	**5**	1004.1	3	*884.2*	*11*
karate	10^5	174.1	3	**172.3**	**3**	173.9	2	*73.3*	*10*
euro	10^3	4108.9	6	**3883.2**	**8**	3924.4	5		
yeast2	10^2	12495.0	6	11611.2	7	**11596.4**	**4**		
javax	10^2	13020.7	4	12894.1	4	**11512.2**	**3**		
jung	10^3	2354.5	5	2312.5	4	**2272.9**	**4**		
elegans	10^2	8734.1	5	8640.9	6	**8243.3**	**5**		
women	10^4	193.9	2	**163.6**	**1**	**163.6**	**1**		

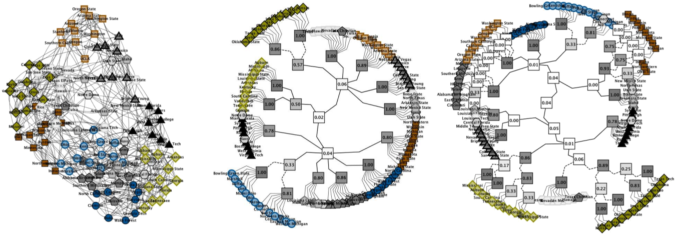


39) Hierarchy of modules

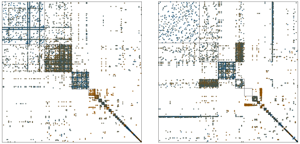


40) Binary hierarchy

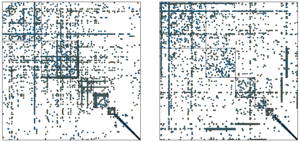
Structure prediction (III)



Hierarchies revealed with CP and HP algorithms, respectively.



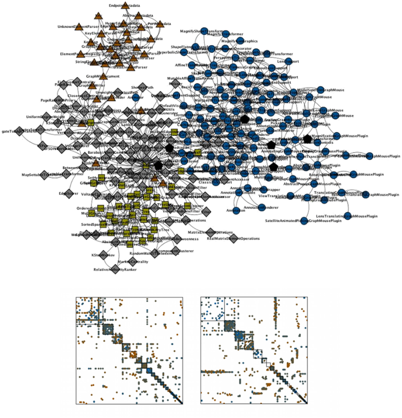
41) javax software network

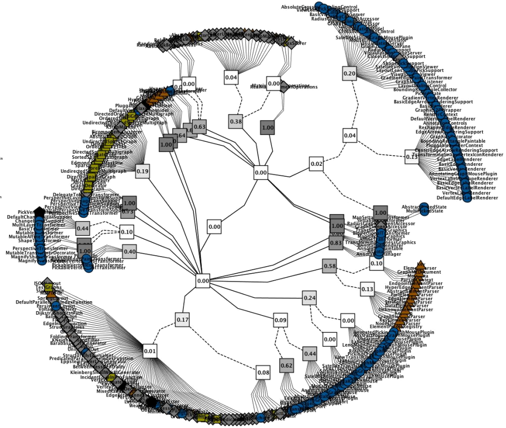


42) elegans metabolic network

Hierarchies and blockmodels revealed with CP and HP algorithms, respectively.

Software networks





Blockmodels revealed with CP and HP algorithms, respectively.

Software networks (II)

- Software network modules coincide with software packages.
- Communities and functional modules more accurately predict packages than communities alone!

NMI									
Network		LUV		CP		MM		HP*	
flamingo	16	0.580	14	**0.609**	27	0.521	16	**0.610**	26
colt	19	0.519	10	0.473	20	**0.533**	19	**0.530**	26
jung	39	0.614	13	0.650	30	0.661	39	**0.680**	41
org	47	0.503	11	**0.537**	30	0.378	39	**0.536**	33
weka	81	**0.558**	26	0.410	49	0.430	63	0.314	28
javax	107	0.704	59	**0.761**	155	0.392	89	0.747	192

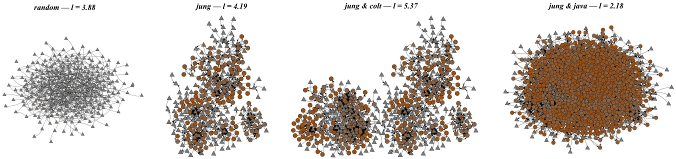
		#			Σ_J			Σ_S	
Network	Γ^1	Γ^2	HP*	Γ^1	Γ^2	HP*	Γ^1	Γ^2	HP*
flamingo	0.531	0.472	0.496	0.501	0.570	**0.585**	0.488	0.504	0.552
colt	0.309	0.431	0.427	0.436	0.498	**0.535**	0.402	0.494	0.527
jung	0.574	0.505	0.545	0.594	0.564	**0.606**	0.583	0.548	0.580
org	0.424	0.466	0.481	0.503	0.542	**0.561**	0.533	0.544	0.550
weka	0.428	0.341	0.249	**0.607**	0.418	0.439	0.599	0.400	0.425
javax	0.590	0.549	0.572	0.581	**0.638**	**0.636**	0.576	0.630	0.623

Software networks (III)

- Software packages can be predicted with $\approx 80\%$ accuracy, whereas complete hierarchy can be precisely identified for over 60% of classes!

			CA				
Network	l	l_∞	P	P_4	P_3	P_2	P_1
flamingo	2.65	4	**0.566**	$\leftarrow$	0.572	*0.793*	1.000
colt	3.35	4	**0.654**	$\leftarrow$	*0.756*	0.942	1.000
jung	2.97	4	**0.617**	$\leftarrow$	0.663	*0.857*	1.000
org	3.50	7	**0.616**	0.616	*0.714*	0.989	1.000
weka	3.02	6	**0.684**	0.692	*0.736*	0.871	1.000
javax	3.11	5	**0.626**	0.631	*0.816*	0.982	1.000

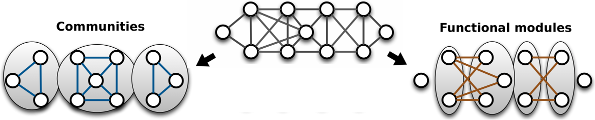
- Networks should not be combined with the core of the language.



Outline

Conclusions

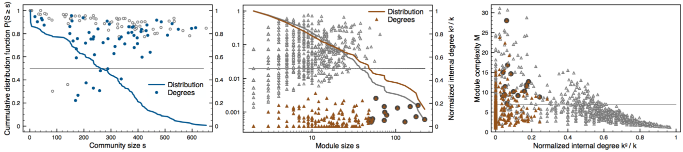
- Structural-world conjecture provides a mesoscopic view on the structure of real-world networks!
  - ↪ Different network modules imply different macroscopic properties.
  - ↪ Clustering assortativity captures how different modules are merged.
  - ↪ Conjecture combines scale-free and small-world phenomena.



- Parameter-free algorithm for detection of general network modules.
  - ↪ Algorithm is (at least) comparable to current state-of-the-art.
  - ↪ Network properties could be further utilized within the algorithm!

Future work

- *How do network modules link between each other?*
  ↪ Necessary to develop a measure of module quality.
- Results suggest that module complexity is much larger than expected!



- *How to utilize degree mixing within the algorithm?*
  ↪ Necessary to analyze network with millions (billions) of nodes.

Thank you.

✉ lovro.subelj@fri.uni-lj.si
WWW http://lovro.lpt.fri.uni-lj.si/

[1] KDD-Cup. http://www.sigkdd.org/kddcup/, 2003.

[2] Lada A. Adamic and Natalie Glance. The political blogosphere and the 2004 U.S. election. In *Proceedings of the KDD Workshop on Link Discovery*, pages 36–43, Chicago, IL, USA, 2005.

[3] Neli Blagus, Lovro Šubelj, and Marko Bajec. Self-similar scaling of density in complex real-world networks. *Physica A: Statistical Mechanics and its Applications*, 391(8): 2794–2802, 2012. doi: 10.1016/j.physa.2011.12.055.

[4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, P10008, 2008.

[5] Marian Boguná, Romualdo Pastor-Satorras, Albert Díaz-Guilera, and Alex Arenas. Models of social networks based on social distance attachment. *Physical Review E*, 70(5):056122, 2004.

[6] Aaron Clauset, M. E. J Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):066111, 2004.

[7] Aaron Clauset, Cristopher Moore, and Mark E. J. Newman. Structural inference of hierarchies in networks. In *Proceedings of the ICML Workshop on Statistical Network Analysis*, pages 1–13, Pittsburgh, PA, USA, 2006.

[8] A. Davis, B. B. Gardner, and M. R. Gardner. *Deep South*. Chicago University Press, Chicago, IL, 1941.

[9] Chris Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 126–135, Philadelphia, PA, USA, 2006.

[10] David V. Foster, Jacob G. Foster, Peter Grassberger, and Maya Paczuski. Clustering drives assortativity and community structure in ensembles of networks. *Physical Review E*, 84(6): 066117, 2011.

[11] M. Girvan and M. E. J Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of United States of America*, 99(12): 7821–7826, 2002.

[12] P. Gleiser and L. Danon. Community structure in jazz. *Advances in Complex Systems*, 6 (4):565, 2003.

[13] Mark S. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6): 1360–1380, 1973.

[14] R. Guimerà, L. Danon, A. Díaz-Guilera, F. Giralt, and A. Arenas. Self-similar community structure in a network of human interactions. *Physical Review E*, 68(6):065103, 2003.

[15] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai. Lethality and centrality of protein networks. *Nature*, 411:41–42, 2001.

[16] Hawoong Jeong, B. Tombor, Reka Albert, Zoltán N. Oltvai, and Albert-László Barabási. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000.

[17] Brian Karrer and M. E. J Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1):016107, 2011.

[18] Jussi M. Kumpula, Mikko Kivelä, Kimmo Kaski, and Jari Saramäki. Sequential algorithm for fast clique percolation. *Physical Review E*, 78(2):026109, 2008. doi: 10.1103/PhysRevE.78.026109.

[19] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4):046110, 2008.

[20] Jure Leskovec, Jon Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 177–187, Chicago, IL, USA, 2005.

[21] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1):1–41, 2007.

[22] Ian X. Y. Leung, Pan Hui, Pietro Liò, and Jon Crowcroft. Towards real-time community detection in large networks. *Physical Review E*, 79(6):066107, 2009.

[23] Lun Li, David Alderson, John C. Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4): 431–523, 2005.

[24] Chen-Yi Lin, Jia-Ling Koh, and Arbee L. P. Chen. A better strategy of discovering link-pattern based communities by classical clustering methods. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 56–67, Hyderabad, India, 2010.

[25] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. can geographic isolation explain this unique trait? *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

[26] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, Berkeley, CA, USA, 1967.

[27] Sergei Maslov and Kim Sneppen. Specificity and stability in topology of protein networks. *Science*, 296(5569):910–913, 2002. doi: 10.1126/science.1065103.

[28] Miller McPherson, Lynn Smith-Lovin, and James M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001. doi: 10.1146/annurev.soc.27.1.415.

[29] M. E. J. Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of United States of America*, 98(2):404–409, 2001.

[30] M. E. J. Newman. Assortative mixing in networks. *Physical Review Letters*, 89(20): 208701, 2002. doi: 10.1103/PhysRevLett.89.208701.

[31] M. E. J. Newman. Mixing patterns in networks. *Physical Review E*, 67(2):026126, 2003. doi: 10.1103/PhysRevE.67.026126.

[32] M. E. J Newman. Detecting community structure in networks. *European Physical Journal B*, 38(2):321–330, 2004.

[33] M. E. J Newman. Symmetrized snapshot of the structure of the internet at the level of autonomous systems. http://www-personal.umich.edu/˜mejn/netdata/, 2006.

[34] M. E. J. Newman and M. Girvan. Mixing patterns and community structure in networks. *Physical Review E*, 67(2):026126, 2003.

[35] M. E. J Newman and E. A Leicht. Mixture models and exploratory analysis in networks. *Proceedings of the National Academy of Sciences of United States of America*, 104(23): 9564, 2007.

[36] M. E. J. Newman and Juyong Park. Why social networks are different from other types of networks. *Physical Review E*, 68(3):036122, 2003. doi: 10.1103/PhysRevE.68.036122.

[37] Gergely Palla, Imre Derényi, Illes Farkas, and Tamas Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043): 814–818, 2005.

[38] Romualdo Pastor-Satorras, Alexei Vázquez, and Alessandro Vespignani. Dynamical and correlation properties of the internet. *Physical Review Letters*, 87(25):258701, 2001. doi: 10.1103/PhysRevLett.87.258701.

[39] Stefan Pinkert, Jörg Schultz, and Jörg Reichardt. Protein interaction networks: More than mere modules. *PLoS Computional Biology*, 6(1):e1000659, 2010.

[40] Usha Nandini Raghavan, Reka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3):036106, 2007.

[41] E. Ravasz and Albert László Barabási. Hierarchical organization in complex networks. *Physical Review E*, 67(2):026112, 2003.

[42] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and Albert László Barabási. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586): 1551–1555, 2002.

[43] J. Reichardt and D. R. White. Role models for complex networks. *European Physical Journal B*, 60(2):217–224, 2007. doi: 10.1140/epjb/e2007-00340-y.

[44] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences of United States of America*, 105:1118–1123, 2008.

[45] Martin Rosvall and Carl T. Bergstrom. An information-theoretic framework for resolving community structure in complex networks. *Proceedings of the National Academy of Sciences of United States of America*, 104(18):7327–7331, 2007.

[46] Sara Nadiv Soffer and Alexei Vázquez. Network clustering coefficient without degree-correlation biases. *Physical Review E*, 71(5):057101, 2005. doi: 10.1103/PhysRevE.71.057101.

[47] Stijn van Dongen. *Graph clustering by flow simulation*. PhD thesis, University of Utrecht, 2000.

[48] Alexei Vázquez, Romualdo Pastor-Satorras, and Alessandro Vespignani. Large-scale topological and dynamical properties of the internet. *Physical Review E*, 65(6):066130, 2002. doi: 10.1103/PhysRevE.65.066130.

[49] Lovro Šubelj and Marko Bajec. Unfolding network communities by combining defensive and offensive label propagation. In *Proceedings of the ECML PKDD Workshop on the Analysis of Complex Networks*, pages 87–104, Barcelona, Spain, 2010.

[50] Lovro Šubelj and Marko Bajec. Robust network community detection using balanced propagation. *European Physical Journal B*, 81(3):353–362, 2011. doi: 10.1140/epjb/e2011-10979-2.

[51] Lovro Šubelj and Marko Bajec. Unfolding communities in large complex networks: Combining defensive and offensive label propagation for core extraction. *Physical Review E*, 83(3):036103, 2011. doi: 10.1103/PhysRevE.83.036103.

[52] Lovro Šubelj and Marko Bajec. Generalized network community detection. In *Proceedings of the ECML PKDD Workshop on Finding Patterns of Human Behaviors in Network and Mobility Data*, pages 66–84, Athens, Greece, 2011.

[53] Lovro Šubelj and Marko Bajec. Community structure of complex software systems: Analysis and applications. *Physica A: Statistical Mechanics and its Applications*, 390(16): 2968–2975, 2011. doi: 10.1016/j.physa.2011.03.036.

[54] Lovro Šubelj and Marko Bajec. Clustering assortativity, communities and functional modules in real-world networks. page 21, 2012.

[55] Lovro Šubelj and Marko Bajec. Ubiquitousness of link-density and link-pattern communities in real-world networks. *European Physical Journal B*, 85(1):32, 2012. doi: 10.1140/epjb/e2011-20448-7.

[56] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[57] Zhi-Xi Wu and Petter Holme. Modeling scientific-citation patterns and other triangle-rich acyclic networks. *Physical Review E*, 80(3):037101, 2009. doi: 10.1103/PhysRevE.80.037101.

[58] Wayne W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.

[59] Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. Community extraction for social networks. *Proceedings of the National Academy of Sciences of United States of America*, 108(18): 7321–7326, 2011. doi: 10.1073/pnas.1006642108.