Finding low-tension communities

Esther Galbrun Behzad Golshan
Aristides Gionis Evimaria Terzi

BOSTON UNIVERSITY
Aalto University School of Science
informatics mathematics

April 2017

Introduction

A community-search problem with opinion dynamics:

find a subgraph that connects the seed nodes
and has low social tension

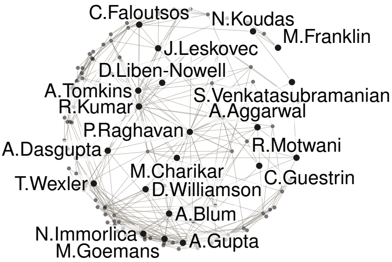
Potential applications areas:

online social media and collaboration networks

Network

We consider a social network $G = (V, E)$

- nodes in V represent individuals
- edges in E represent their interactions



Profiles

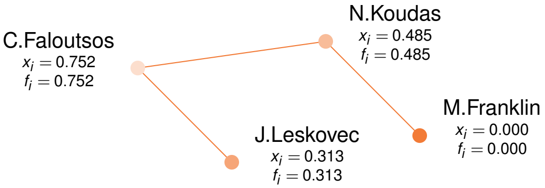
Each individual has his own preferences, habits, opinions…

Individuals may choose not to act in accordance with their true prefererences as they try to minimize peer pressure by conforming their preferences to those of their peers

Profiles

Each node i is associated to
- a **latent profile**, x_i: the individual's true preferences
- a **conformed profile**, f_i: his expressed preferences

Both take value in the interval $[0, 1]$

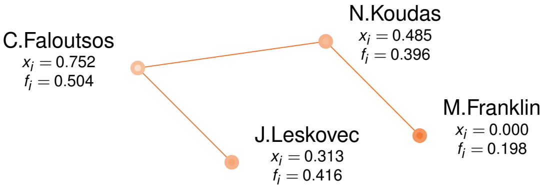


Profiles

Each node i is associated to

- a **latent profile**, x_i: the individual's true preferences
- a **conformed profile**, f_i: his expressed preferences

Both take value in the interval $[0, 1]$



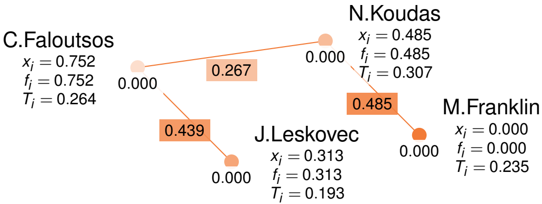
Tensions

Differences between profiles cause tensions

Each node i bears

- an **inner tension**: own latent and conformed profiles
- a **cross tension**: own and neighbors' conformed profile

$$T_i(G, \mathbf{x}, \mathbf{f}) = (x_i - f_i)^2 + \sum_{j \in N_G(i)} (f_i - f_j)^2$$



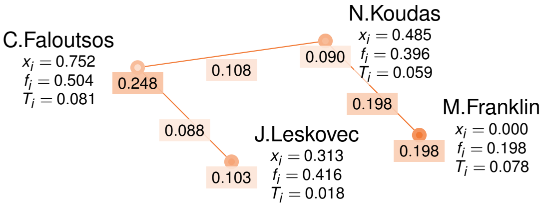
Tensions

Differences between profiles cause tensions

Each node i bears

- an **inner tension**: own latent and conformed profiles
- a **cross tension**: own and neighbors' conformed profile

$$T_i(G, \mathbf{x}, \mathbf{f}) = (x_i - f_i)^2 + \sum_{j \in N_G(i)} (f_i - f_j)^2$$



Tensions

Social tension of the network: sum of the individual tensions

$$\begin{aligned} T(G,\mathbf{x},\mathbf{f}) &= \sum_{i\in V} T_i(G,\mathbf{x},\mathbf{f}) \\ &= \sum_{i\in V} \left((x_i - f_i)^2 + \sum_{j\in N_G(i)} (f_i - f_j)^2\right) \\ &= \sum_{i\in V} (x_i - f_i)^2 + \sum_{(i,j)\in E} 2(f_i - f_j)^2 \end{aligned}$$

Conformation process

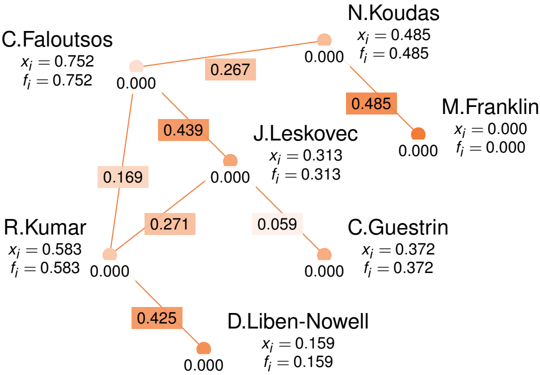
Consider a **repeated averaging process**
at each step each node adjusts its conformed profile
by setting it to the average of its latent profile
and the conformed profile of its neighbors

$$f_i(t+1) = \frac{x_i + \sum_{j \in N_G(i)} f_j(t)}{1 + |N_G(i)|}$$

[1] Bindel, Kleinberg and Oren (2011) *How Bad is Forming Your Own Opinion?* FOCS
[2] Gionis, Terzi, and Tsaparas (2013) *Opinion Maximization in Social Networks.* SDM

Conformation process

Apply the averaging process repeatedly

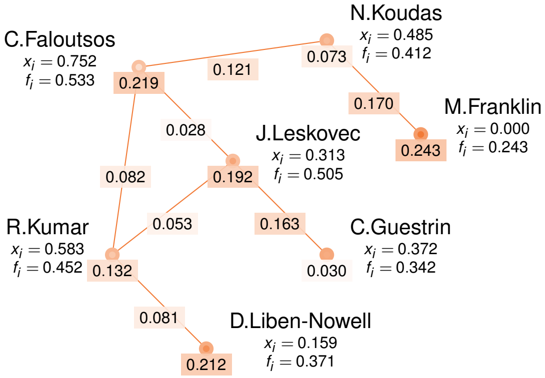


$$T(G') = 1.571$$
$$= 0.000 + 1.571$$

Conformation process

Apply the averaging process repeatedly

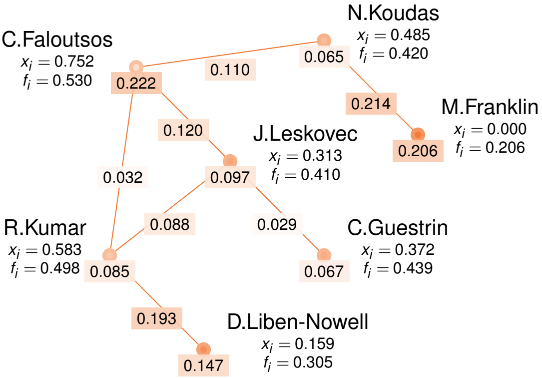
$$T(G') = 0.386$$
$$= 0.212 + 0.174$$



Conformation process

Apply the averaging process repeatedly

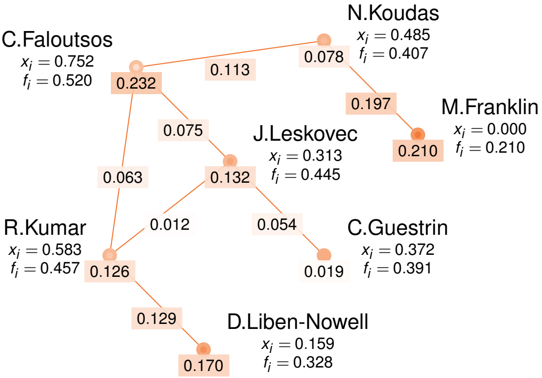
$$T(G') = 0.377$$
$$= 0.139 + 0.239$$



Conformation process

Apply the averaging process repeatedly

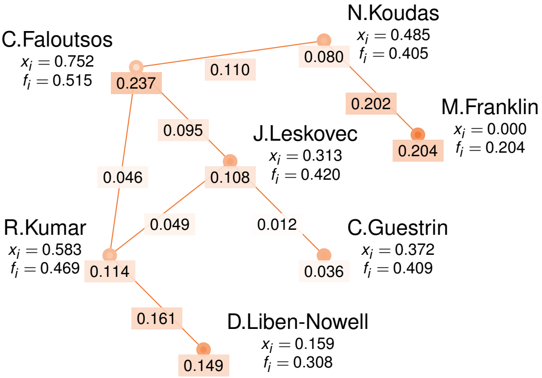
$$T(G') = 0.328$$
$$= 0.167 + 0.162$$



Conformation process

Apply the averaging process repeatedly

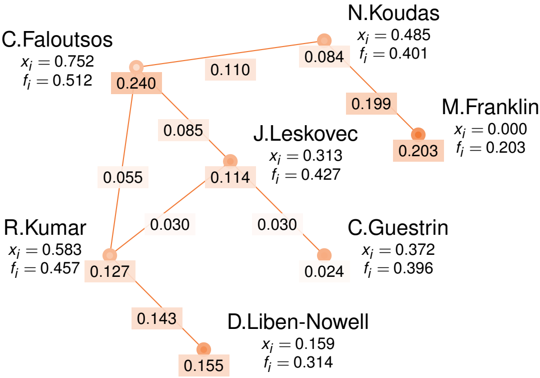
$$T(G') = 0.337 = 0.152 + 0.185$$



Conformation process

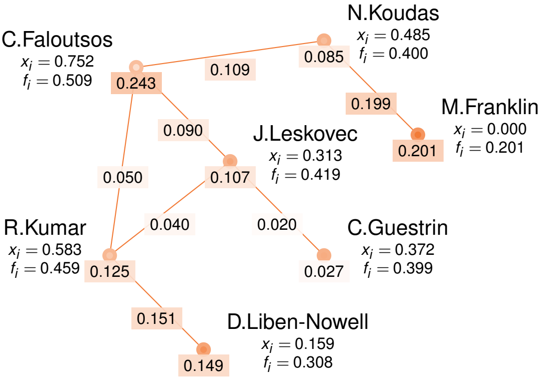
Apply the averaging process repeatedly

$$T(G') = 0.328$$
$$= 0.160 + 0.168$$



Conformation process

Apply the averaging process repeatedly

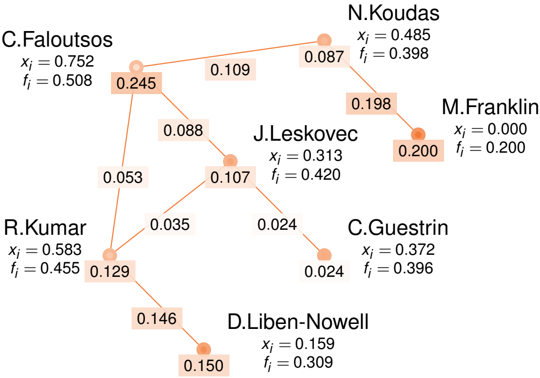


$$T(G') = 0.331$$
$$= 0.156 + 0.174$$

Conformation process

Apply the averaging process repeatedly

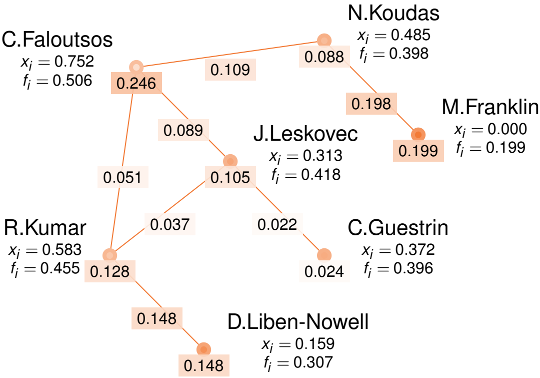
$$T(G') = 0.328$$
$$= 0.158 + 0.170$$



Conformation process

Apply the averaging process repeatedly

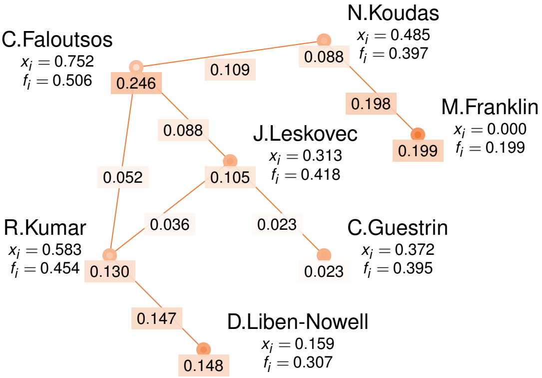
$$T(G') = 0.329$$
$$= 0.158 + 0.171$$



Conformation process

Apply the averaging process repeatedly

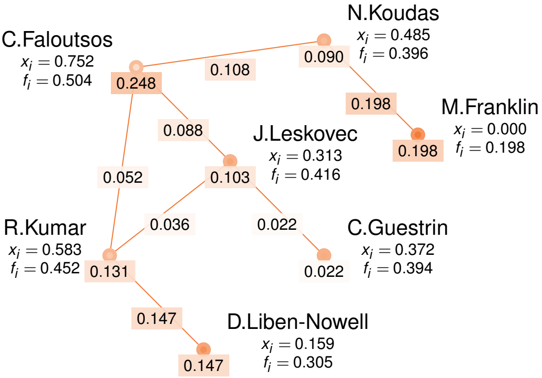
$$T(G') = 0.328$$
$$= 0.158 + 0.170$$



Conformation process

Apply the averaging process repeatedly until convergence

$$T(G') = 0.328$$
$$= 0.159 + 0.169$$



Conformation process

The repeated averaging model is equivalent to choosing f_i to minimize $T_i(G, \mathbf{x}, \mathbf{f})$

It yields a Nash equilibrium for the tension, not a social optimum

[1] Bindel, Kleinberg and Oren (2011) *How Bad is Forming Your Own Opinion?* FOCS
[2] Gionis, Terzi, and Tsaparas (2013) *Opinion Maximization in Social Networks.* SDM

Problem statement

Given a network $G = (V, E)$,

latent profiles $\mathbf{x}$ and a set of seed nodes $Q \subseteq V$,

find $V' \subseteq V$

such that

$Q \subseteq V'$,

the graph G' induced by V' on G is connected and

$T(G', \mathbf{x}, \mathbf{f})$ is minimized,

where $\mathbf{f}$ is computed by the repeated averaging model on G'.

Algorithms

Computing the conformed profiles is costly

Use a proxy for the contribution of pairs of neighboring nodes

Assign weight $w_{ij} = \left| x_i - x_j \right|$ to each edge $(i, j) \in E$

Two proposed algorithms to build a connecting subgraph: a **spanning-tree approach** and a **top-down approach**

Spanning-tree approach

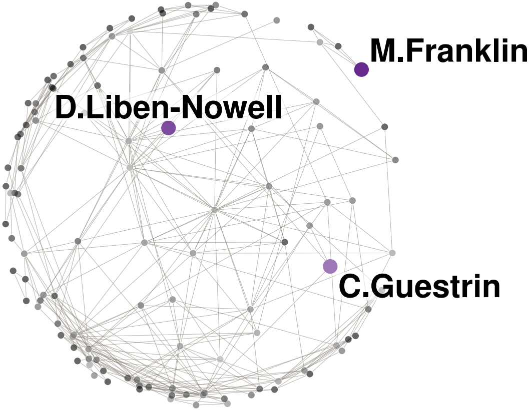
Build a spanning tree between the query nodes
using the 2-approximation to Steiner tree problem

Different ways to score the tree yield variants

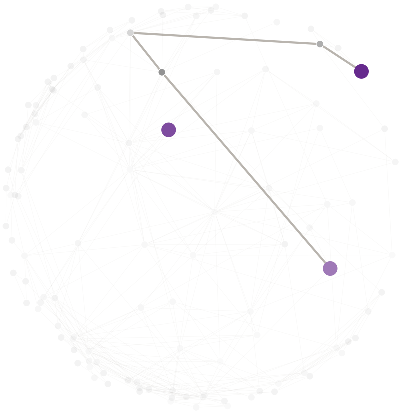
`CTree(e)` Number of edges involved

`CTree(s)` Sum of weights of the edges along the path

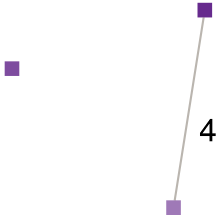
Spanning-tree approach



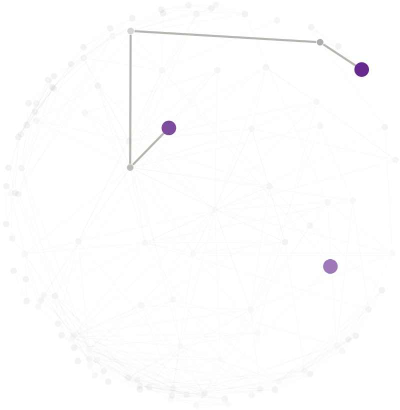
Spanning-tree approach



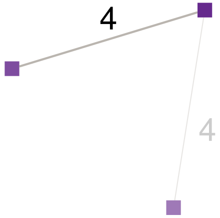
Find shortest paths between pairs of seed nodes



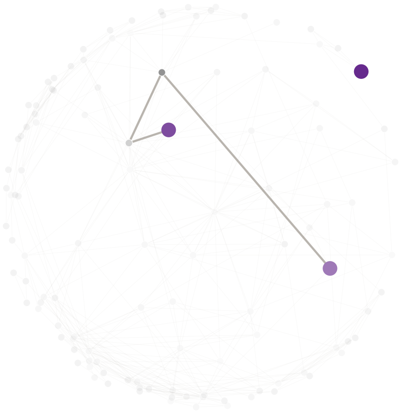
Spanning-tree approach



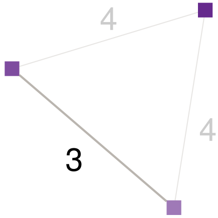
Find shortest paths between pairs of seed nodes



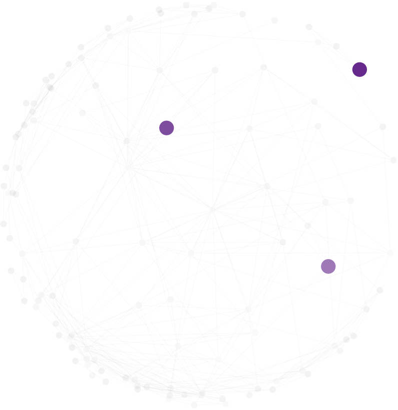
Spanning-tree approach



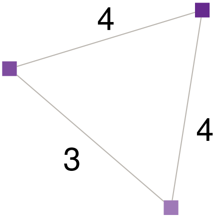
Find shortest paths between pairs of seed nodes



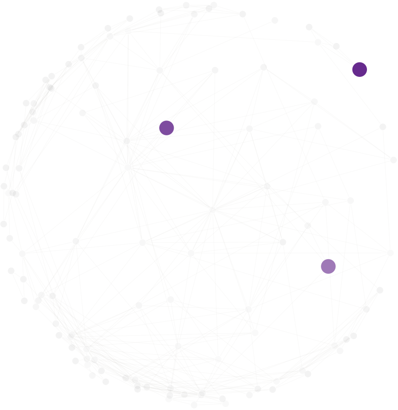
Spanning-tree approach



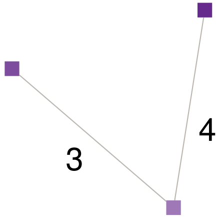
Considering each path as an edge find a minimum spanning tree



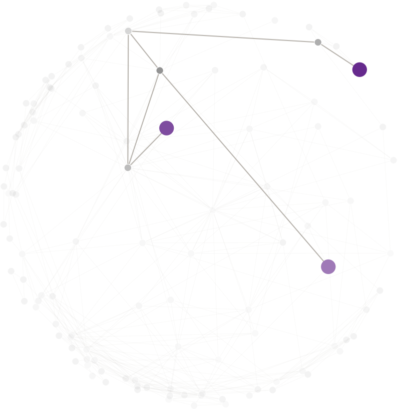
Spanning-tree approach



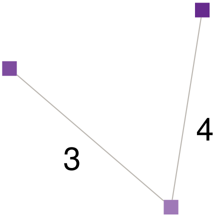
Considering each path as an edge find a minimum spanning tree



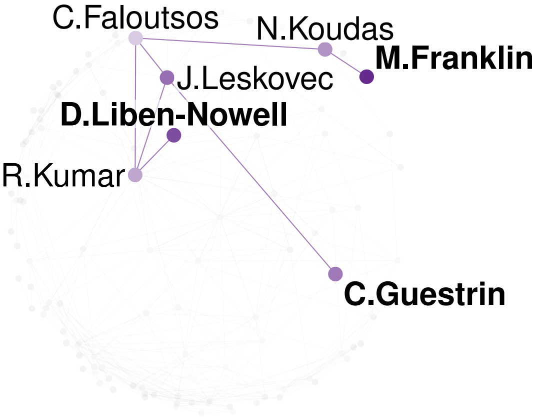
Spanning-tree approach



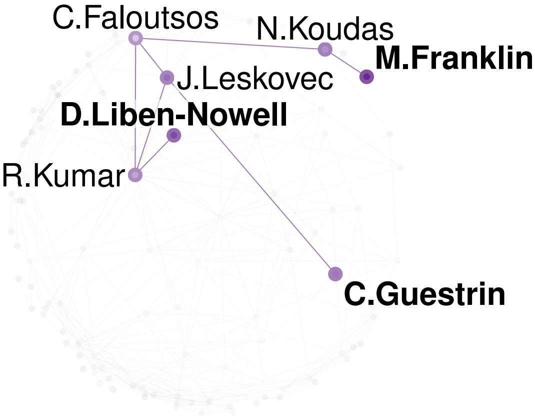
Finally expand back to paths



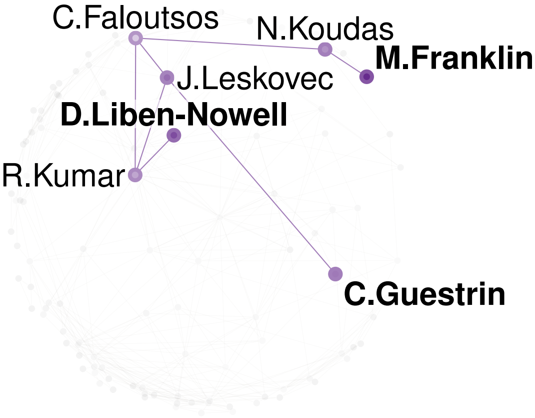
Spanning-tree approach



Spanning-tree approach



Spanning-tree approach



`CTree(e)`

$T(G') = 0.3279$

$|V'| = 7 \; |E'| = 7$

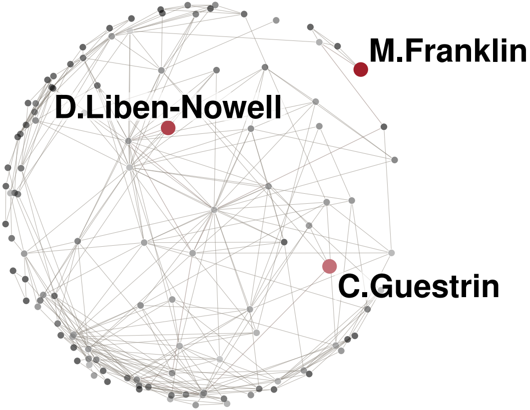
Top-down approach

Iteratively remove nodes until it is no longer possible without disconnecting the query nodes

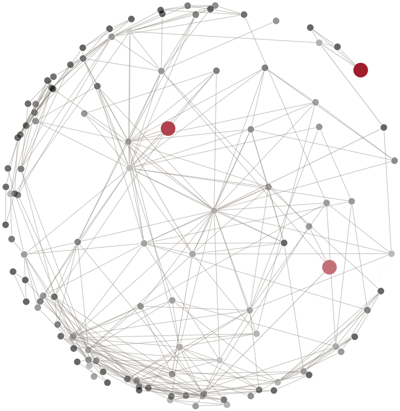
Different ways to pick next node yield variants

- `CPeel(r)` Random
- `CPeel(s)` Sum of adjacent edges weight
- `CPeel(m)` Max of adjacent edges weight

Top-down approach

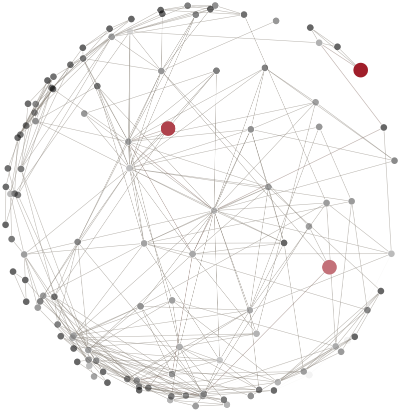


Top-down approach



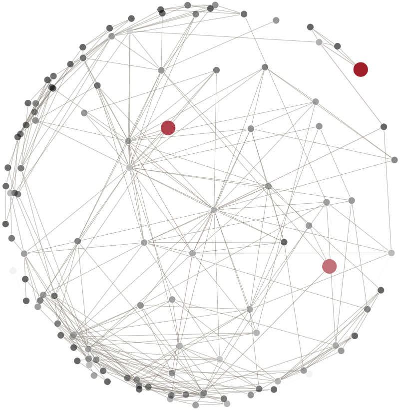
Remove nodes

Top-down approach



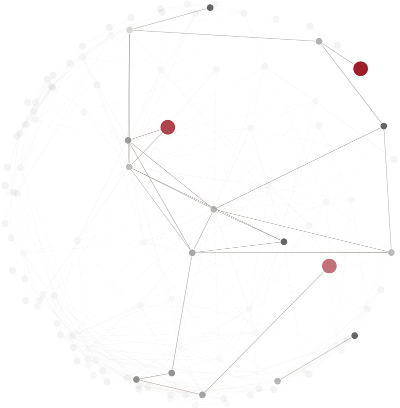
Remove nodes

Top-down approach



...

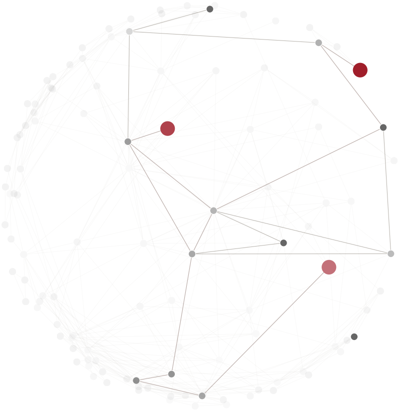
Top-down approach



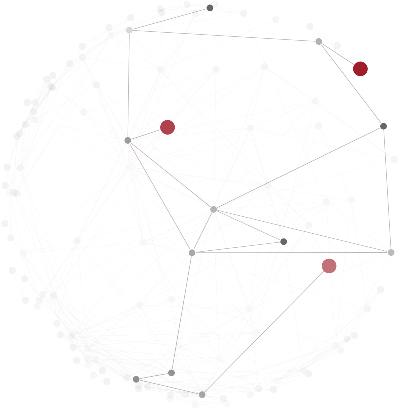
...

Top-down approach

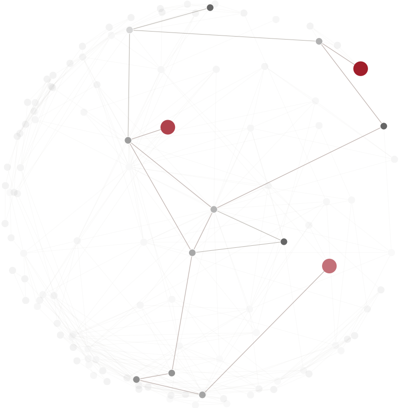
Top-down approach



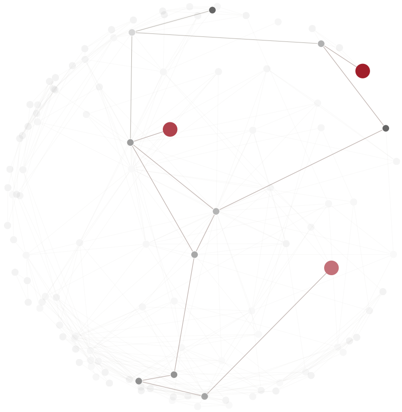
Top-down approach



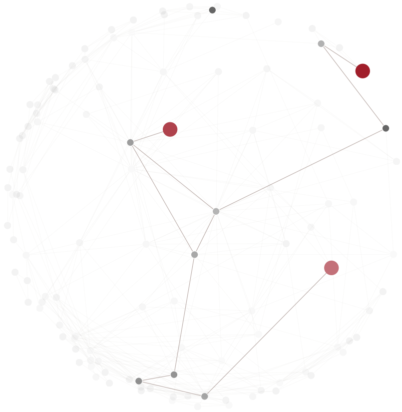
Top-down approach



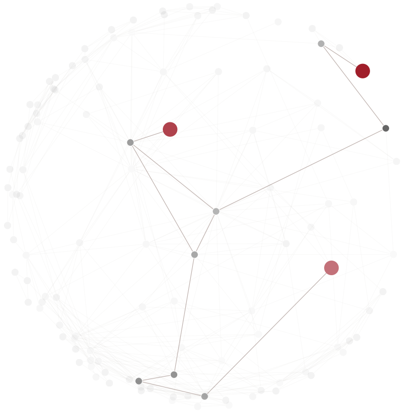
Top-down approach



Top-down approach

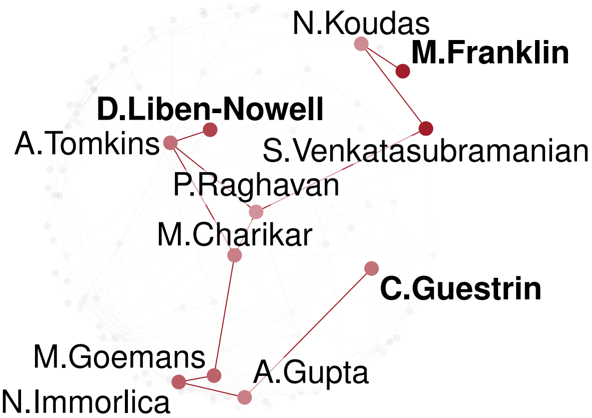


Top-down approach

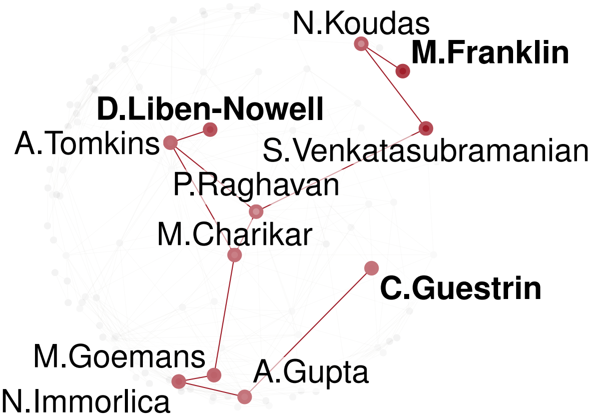


Until it is no longer possible without disconnecting the query nodes

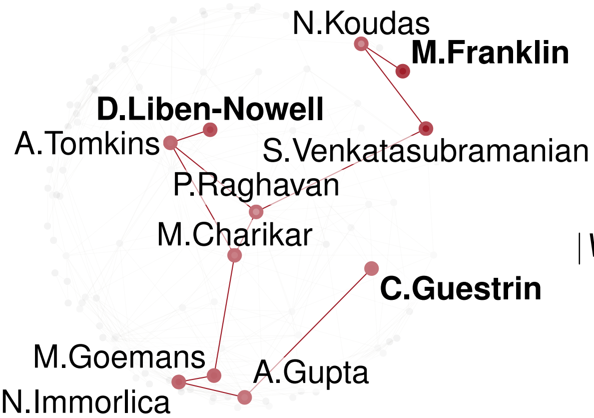
Top-down approach



Top-down approach



Top-down approach

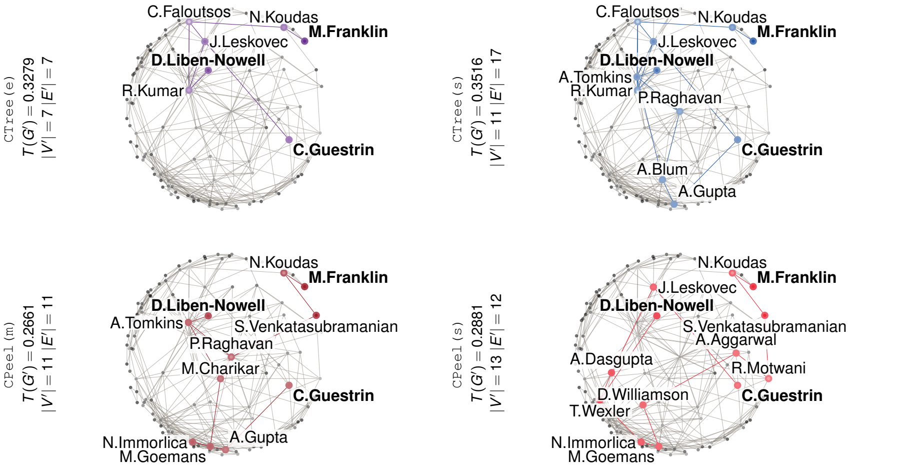


`CPeel(m)`

$T(G') = 0.2661$

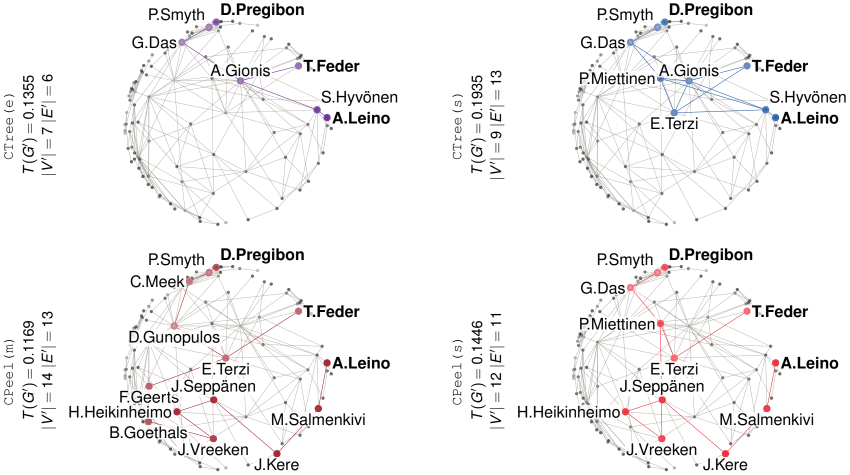
$|V'| = 11$ $|E'| = 11$

Example results



Example solutions for connecting three seed nodes in the 1-hop ego-network of J.M.Kleinberg with single-attribute latent profiles derived from keywords

Example results



Example solutions for connecting three seed nodes in the 1-hop ego-network of H.Mannila with single-attribute latent profiles derived from keywords

Quantitative experiments

Compared algorithms: proposed variants and `Cocktail` [3]

Evaluation measures:

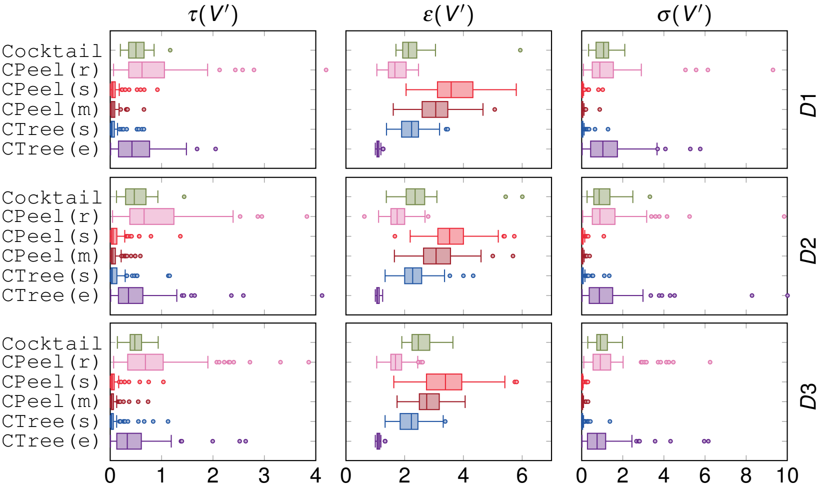
standardized social tension	$\tau(V') = T(V')/(2e_b \cdot \overline{w^2}(V))$
std. solution size (aux.)	$\varepsilon(V') = \lvert E(V') \rvert / e_b$
std. average edge weight (aux.)	$\sigma(V') = \overline{w^2}(V')/\overline{w^2}(V)$

e_b: size of the minimum spanning tree connecting the query nodes

$\overline{w^2}(V)$: average squared edge weight in V

[3] M. Sozio and A. Gionis (2010) *The community-search problem and how to plan a successful cocktail party.* KDD

Quantitative experiments



Results for the 2-hop ego-network of C.Papadimitriou
with single-attribute latent profiles derived from conferences

Running times

Network	$\lvert V \rvert$	CTree(e)	CTree(s)	CPeel(s)	CPeel(m)
IMDB WarnerBros 1970s	225	0.0 ($\pm$0.0)	0.1 ($\pm$0.0)	0.9 ($\pm$0.1)	0.7 ($\pm$0.1)
IMDB F.F.Coppola	678	0.0 ($\pm$0.0)	0.7 ($\pm$0.1)	8.8 ($\pm$1.6)	6.4 ($\pm$1.2)
DBLP E.Demaine	2234	0.1 ($\pm$0.0)	2.8 ($\pm$0.3)	75.7 ($\pm$12.3)	60.9 ($\pm$13.0)
DBLP C.Papadimitriou	2613	0.1 ($\pm$0.0)	3.2 ($\pm$0.3)	114.6 ($\pm$20.0)	91.6 ($\pm$22.0)
DBLP ICDM	2795	0.2 ($\pm$0.0)	3.3 ($\pm$0.3)	163.9 ($\pm$28.1)	133.9 ($\pm$29.7)
DBLP KDD	2737	0.2 ($\pm$0.0)	3.5 ($\pm$0.2)	166.8 ($\pm$27.8)	136.7 ($\pm$28.1)
DBLP P.Yu	4596	0.2 ($\pm$0.0)	4.4 ($\pm$0.3)	291.3 ($\pm$56.3)	242.3 ($\pm$43.1)
IMDB WarnerBros	2111	0.3 ($\pm$0.1)	3.0 ($\pm$0.2)	139.1 ($\pm$19.7)	57.2 ($\pm$13.0)
IMDB WB+Paramount+Fox	5758	1.4 ($\pm$0.2)	15.4 ($\pm$1.0)	2192.3 ($\pm$346.6)	670.5 ($\pm$168.1)

Average running times (in seconds) of the algorithms ($\pm$ std. dev.)

Conclusions

We defined a community-search problem with opinion dynamics:
find a subgraph that connects the seed nodes
and has low social tension

We proposed two algorithms, `CPeel` and `CTree`, with variants

Potential applications areas:
online social media and collaboration networks

Conclusions

We defined a community-search problem with opinion dynamics:
find a subgraph that connects the seed nodes
and has low social tension

We proposed two algorithms, `CPeel` and `CTree`, with variants

Potential applications areas:
online social media and collaboration networks

Extended version with team-formation problem variant:
`https://arxiv.org/abs/1701.05352`

Code and data:
`https://members.loria.fr/EGalbrun/resources/GGGT17_finding_code+data.zip`