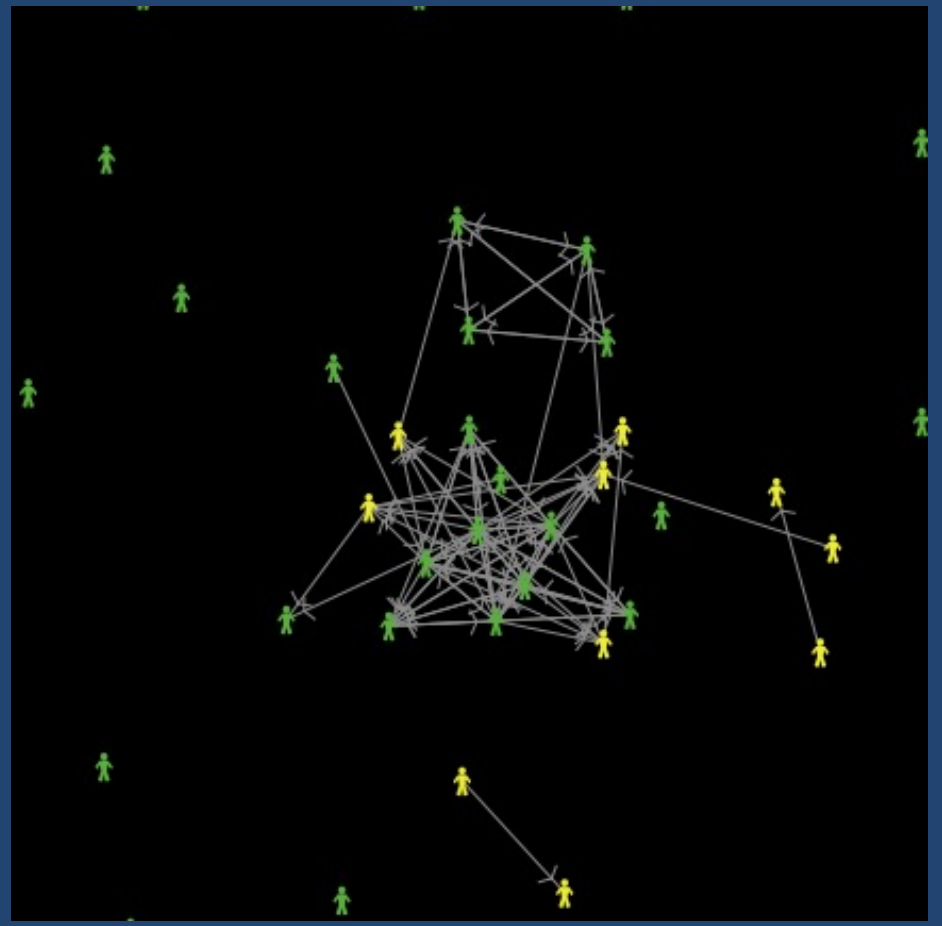
Investigating Network Formation Mechanisms with Agent-Based Models

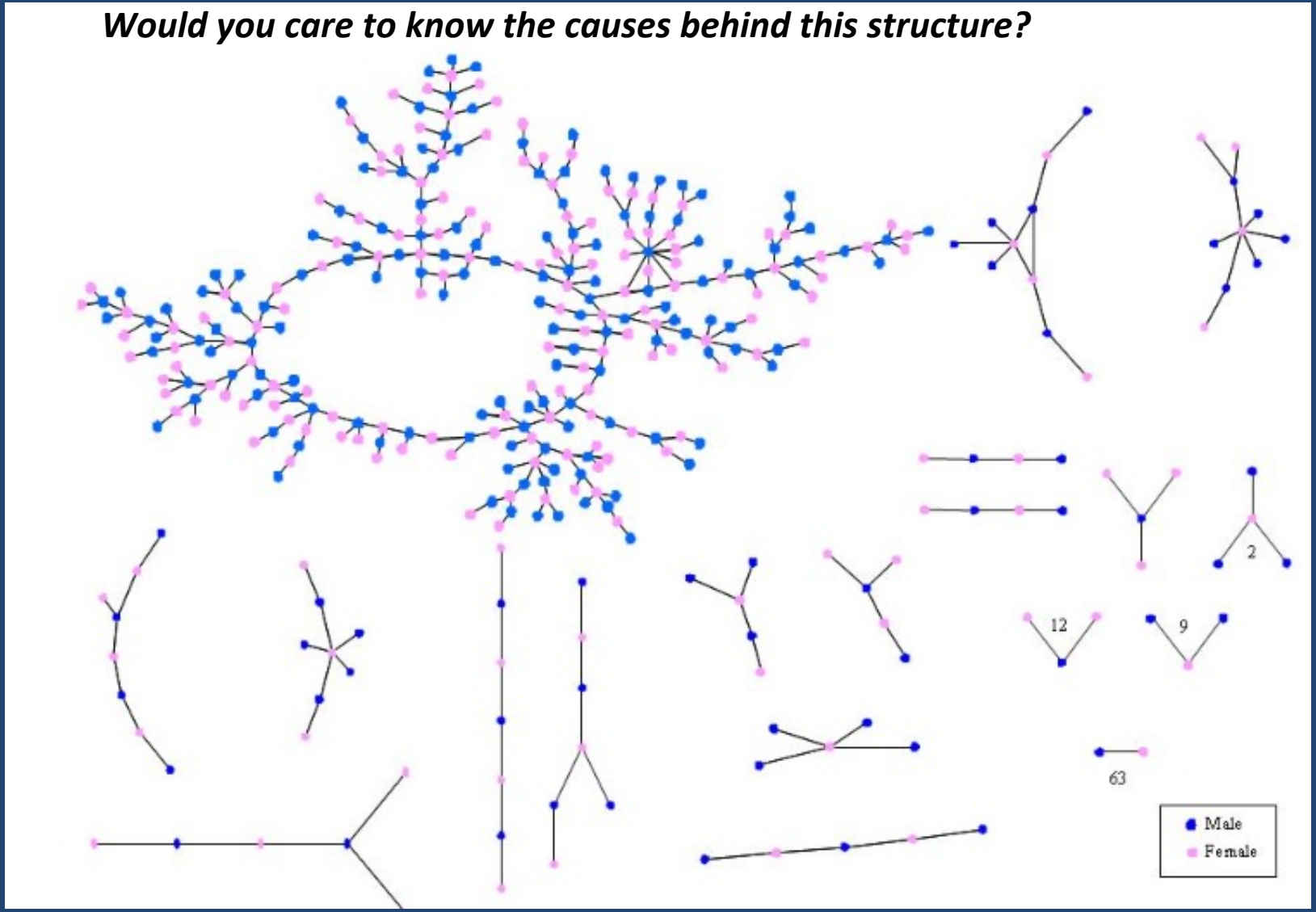


Francesco Renzini

francesco.renzini@unimi.it

Department of Social and Political Sciences, University of Milan
Behave Lab

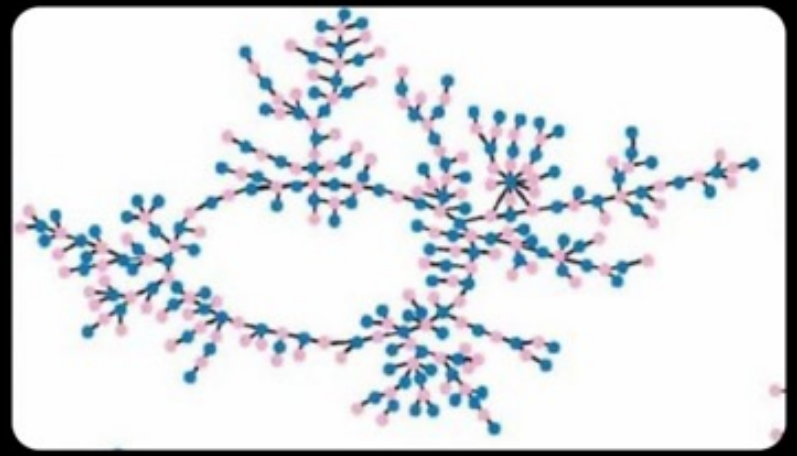
Bearman et al. (2004): Chains of affection



metakuna (13/100 bl... Segui
@metakuna

I've just realized how bizarre this is. At first you think "oh these are just the slutty people", but actually the max number of partners for ppl in the actual ring is only 4. They just somehow formed this massive structure with no cross connections???

Traduci post



15:23 · 16 gen 24 · **678K** Visualizzazioni

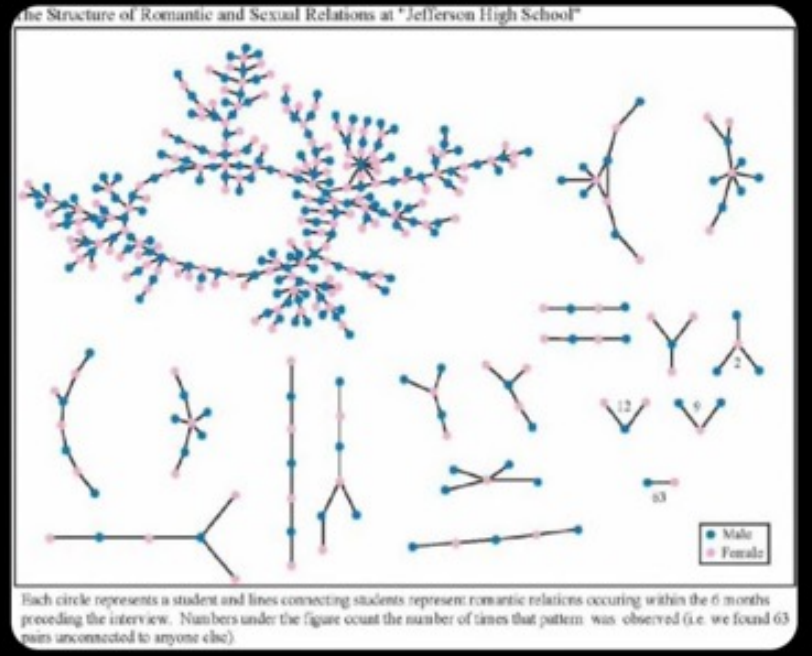
321 repost **102** citazioni **8.914** Mi piace

1.277 segnalibri

metakuna (13/100 bl... Segui
@metakuna

There are like 40 hops! And no smaller ring-like structures in the rest of the network! Why???

Traduci post



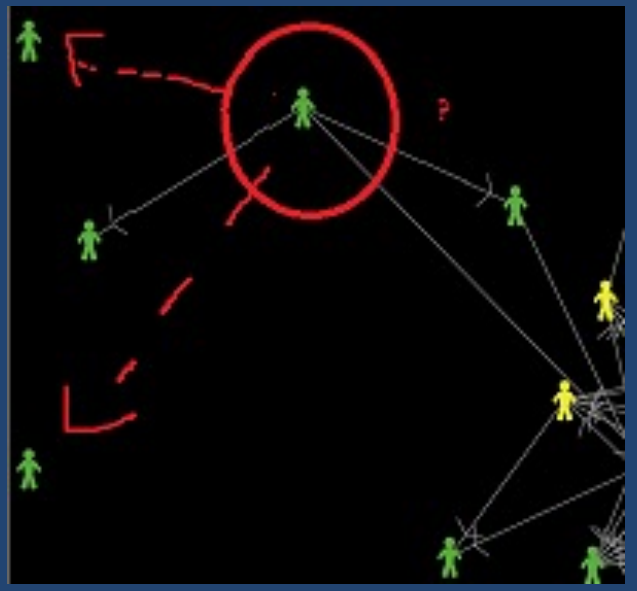
15:23 · 16 gen 24 · **76,6K** Visualizzazioni

22 repost **8** citazioni **1.670** Mi piace

71 segnalibri

Networks are complex and fascinating objects

- Outcome of behavior: *Network formation*



- Networks important determinants of many phenomena

We have a huge problem

- *Determinants behind the selection of network partners are rarely observed → Data scarcity on network formation behavior*
- Need to fill-in these gaps by proposing an explanation for the patterns you observe

- ABM: Help us to compare alternative explanations on common grounds and say something about the phenomenon of interest in case of data scarcity

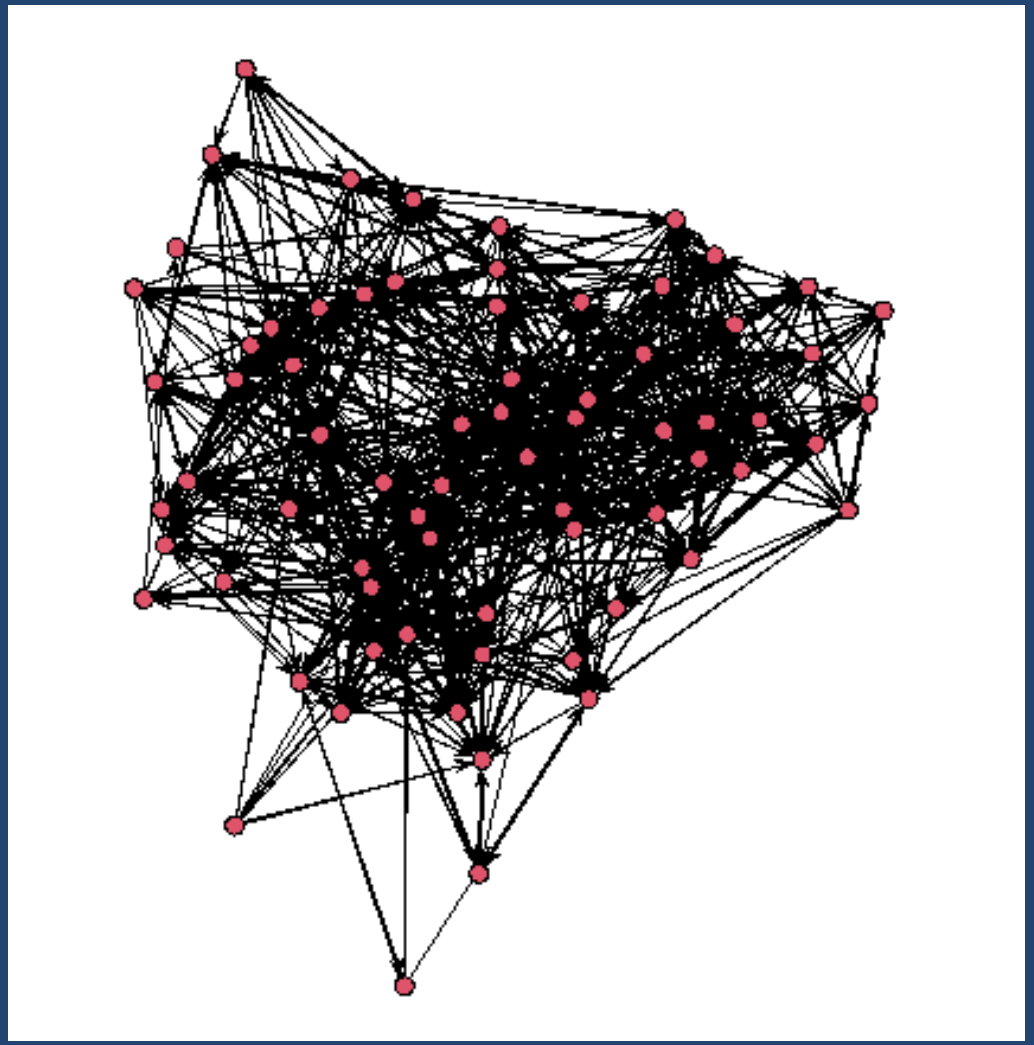
How to do so in practice?

Social Networks
Volume 76, January 2024, Pages 150-159

Status, cognitive overload, and incomplete information in advice-seeking networks: An agent-based model

Francesco Renzini, Federico Bianchi, Flaminio Squazzoni

Context: Lazega (2001)



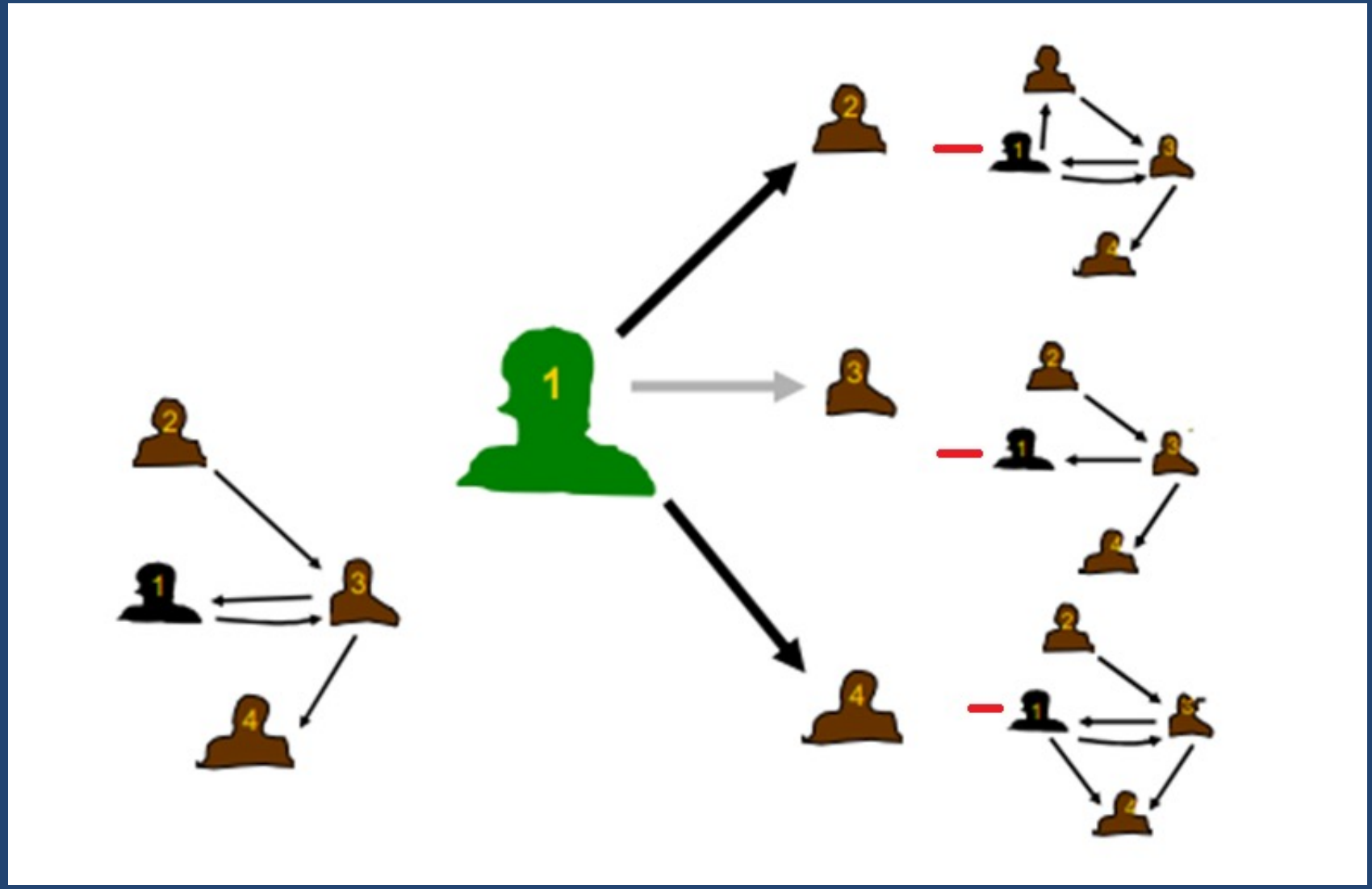
The puzzle

- Asking for advice is costly → Status loss
- If status is so important, why do people engage extensively in such a costly activity as advice?
- What network formation mechanism explains the emergence of such network patterns?
- Determinants of network formation were not measured; only the network and some characteristics of lawyers

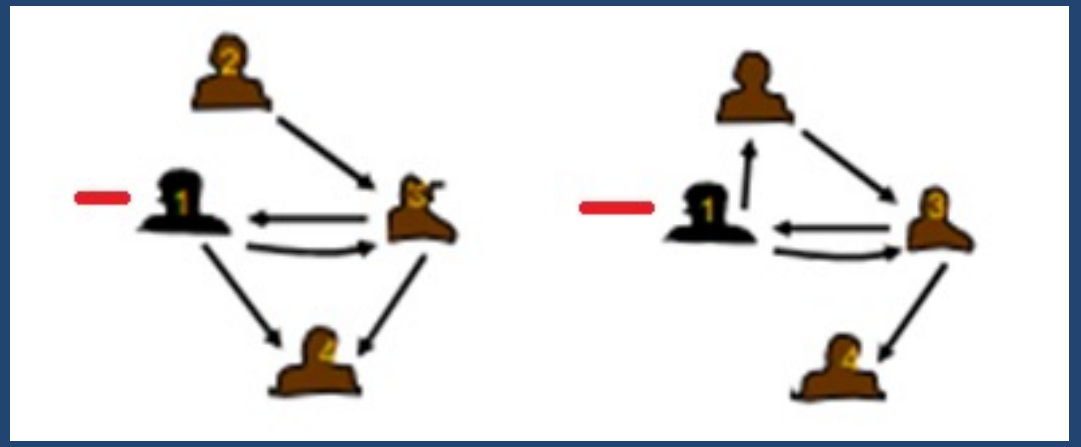
Fill in the gaps: We need a model

- Stochastic Actor Oriented Models (SAOM; Snijders, 2017; Snijders and Steglich, 2015)
- Simulation-based inference → infer/extract network formation behavior from network data (from the footprints)
- We were not satisfied by the conclusions generated with this approach
- Use ABM to build an alternative theory explaining Lazega's (2001) network
- *This comparison between SAOM and ABM on common grounds (Lazega's network) → Get important insights even if no network formation data*

SAOM (Snijders and Koskinen, 2010)



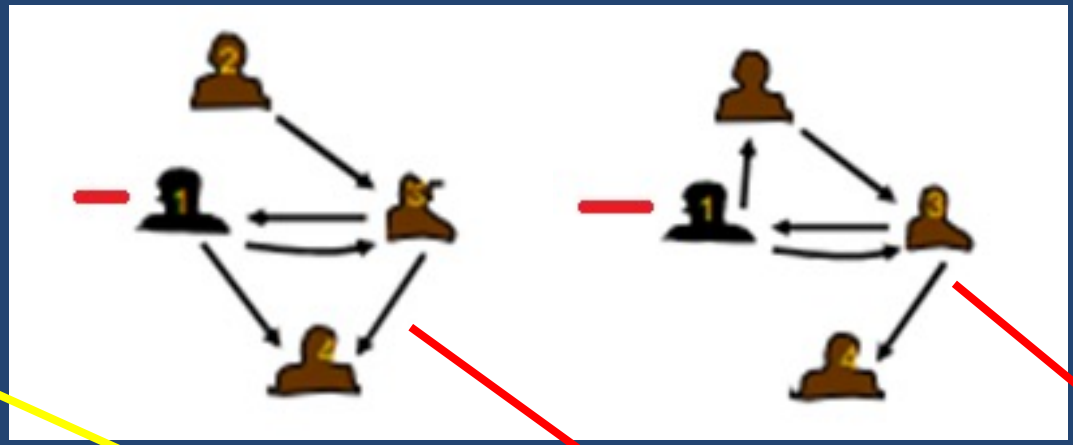
SAOM (Snijders and Koskinen, 2010)



- Local network configurations →Represent network formation behavior (e.g., local hierarchy Vs. generalized exchange)
- Alternatives → Traded off: via objective function = Satisfaction with structure of new state
- Resulting networks will be evaluated in terms of which local configurations arise → Choose the ones we like

SAOM (Snijders and Koskinen, 2010)

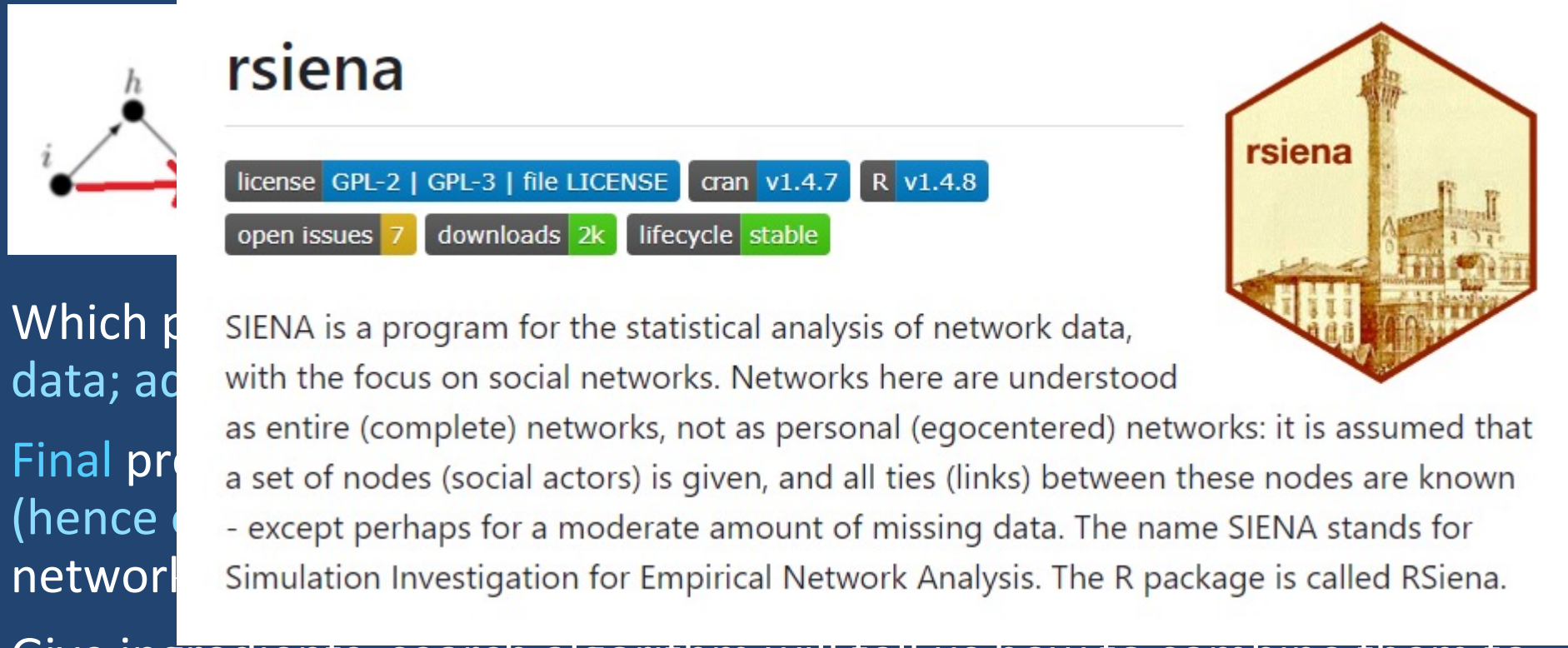
Preferences



$$f_i(x) = \beta_1 \sum_j x_{ij} + \beta_2 \sum_j x_{ij}x_{ji} + \beta_3 \sum_{j,h} x_{ih}x_{hj}x_{ij} + \beta_4 \sum_{j,h} x_{ih}x_{hj}x_{ji}.$$

SAOM Approach: Infer NFB from Data

- Specify many local configurations → reflecting many potential and concurrent network formation behaviors (or combinations of behaviors)



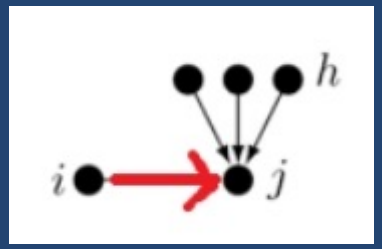
- Which p ith data; ac
- Final pr (hence network
- Give ingredients, search algorithm will tell us how to combine them to match data

Issues with the approach

- Agents are required to perform too demanding calculations, having complex combinations of network formation behaviors COMPLEX

$$f_i(x) = \beta_1 \sum_j x_{ij} + \beta_2 \sum_j x_{ij}x_{ji} + \beta_3 \sum_{j,h} x_{ih}x_{hj}x_{ij} + \beta_4 \sum_{j,h} x_{ih}x_{hj}x_{ji}$$
$$+ \beta_5 \sum_j x_{ij} \sum_h x_{hj} + \beta_6 \sum_j x_{ji} \sum_h x_{jh} + \beta_7 \sum_j x_{ij} \sum x_{ih}.$$

- Supposed to have complete network inf



- Not dealing with thresholds/tipping poi

Table 11. Model 5 for Advice Relations: Including also the GWESP Representation of Transitive Closure.

Effect	Parameters	(*SE*)
Out-degree	−1.745	(0.168)
Reciprocity	1.054	(0.128)
Transitive triplets	0.121	(0.016)
Three cycles	−0.055	(0.028)
In-degree—popularity	−0.011	(0.008)
Out-degree—popularity	−0.062	(0.013)
Out-degree—activity	−0.021	(0.005)
GWESP ($\alpha = .69$)	2.045	(0.272)
Seniority receiver	−0.002	(0.003)
Seniority sender	0.009	(0.003)
Seniority similarity	1.110	(0.197)
Seniority of indirect ties	−0.004	(0.002)

Note: GWESP = geometrically weighted edgewise shared partners.

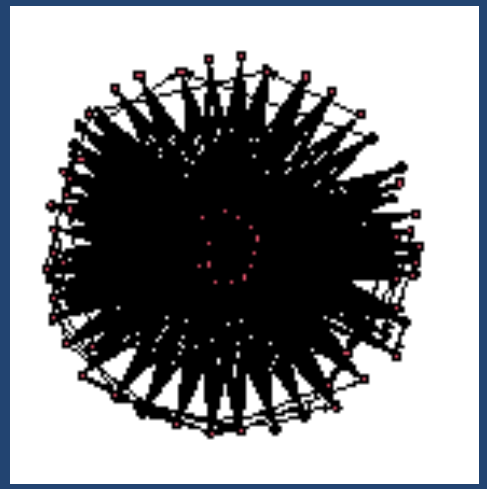
TIED TO DATA

Develop an alternative explanation

- More empirically plausible agent network formation behavior → delve in advice-networks literature and build from there
- Say something about advice formation by comparing the two approaches

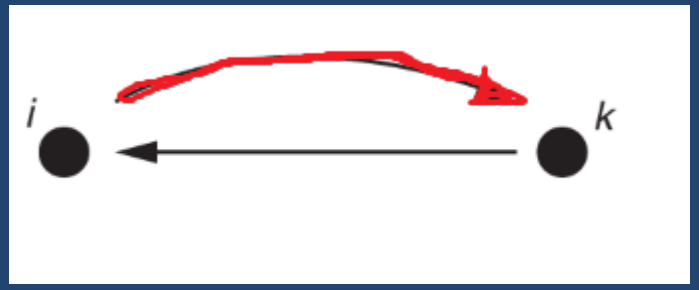
Our model

- Keep same SAOM structure: Agents max. objFun → change the local configurations
- Every agent has a baseline tendency to seek for advice
- Tendency differs depending on skills, which are unevenly distributed → Low skilled ask to high-skilled
- This might help us to explain centralization (few, attractive high skilled agents) and density (a lot of low skilled, very needy)
- Indeed:

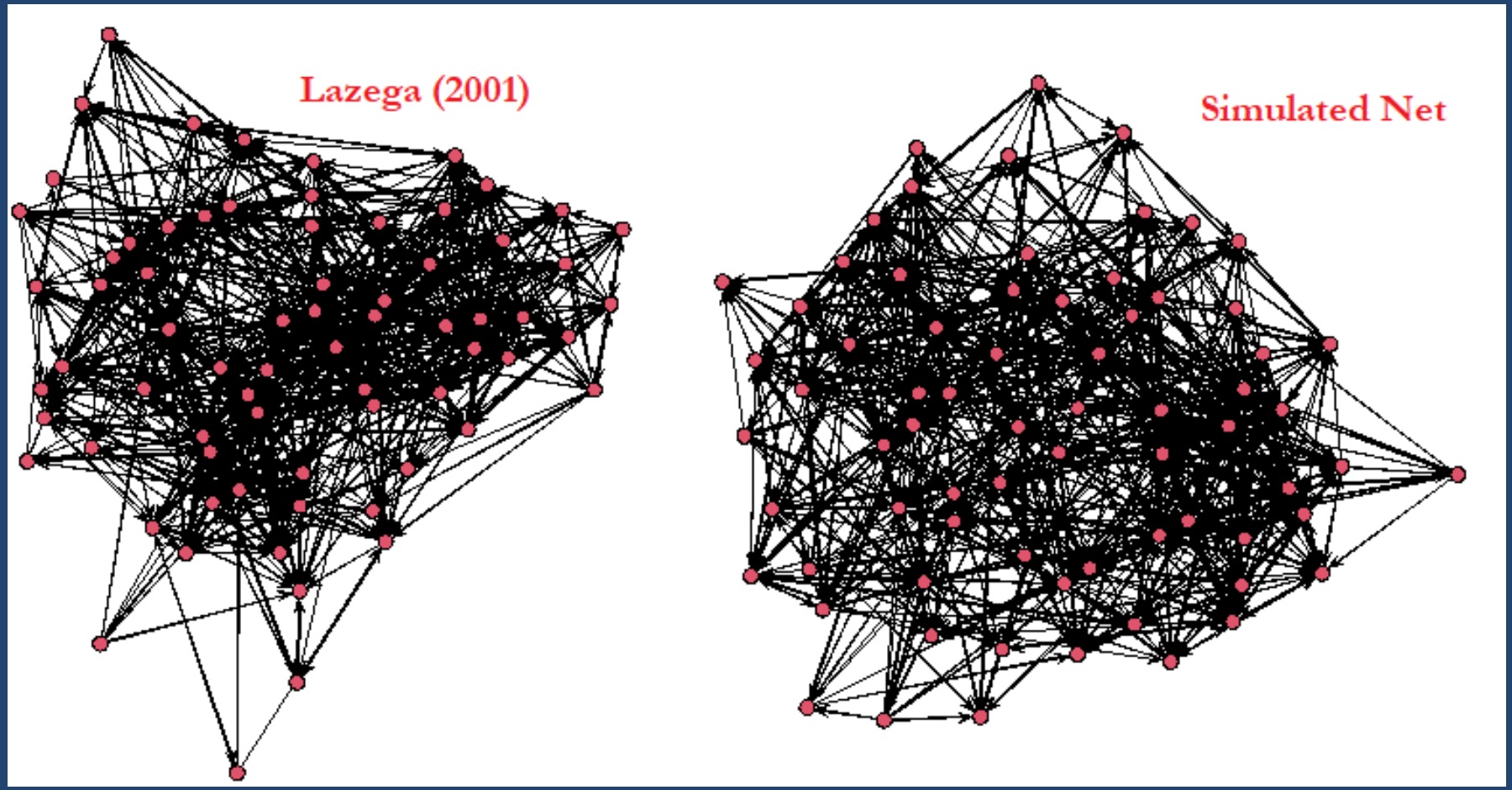


Our model

- High-skilled agents are easily overloaded; observed also in real world organizations (Lazega et al., 2006; Cross and Prusak, 2016)
- Introduce τ, limiting #of incoming requests for the High-skilled
- Low-skilled individuals must find alternative ways to compensate for this information loss → Exploitation (minimize status loss)



Our model



Parameters Search

- #of high-skilled, τ, and the β_s: attractiveness from low to high, and from high to high, baseline tendencies to seek for advice, exploitation (when redirecting)
- Define plausible intervals (e.g., τ should not be 1 or 40, should be an integer)
- Form a grid, list of unique combinations of plausible parameters
- Run each element of the grid 500 parallel times, compare distributions of simulated networks with Lazega (2001) under specific metrics

Our model – Empirical Validation

Plus 5 additional other metrics:

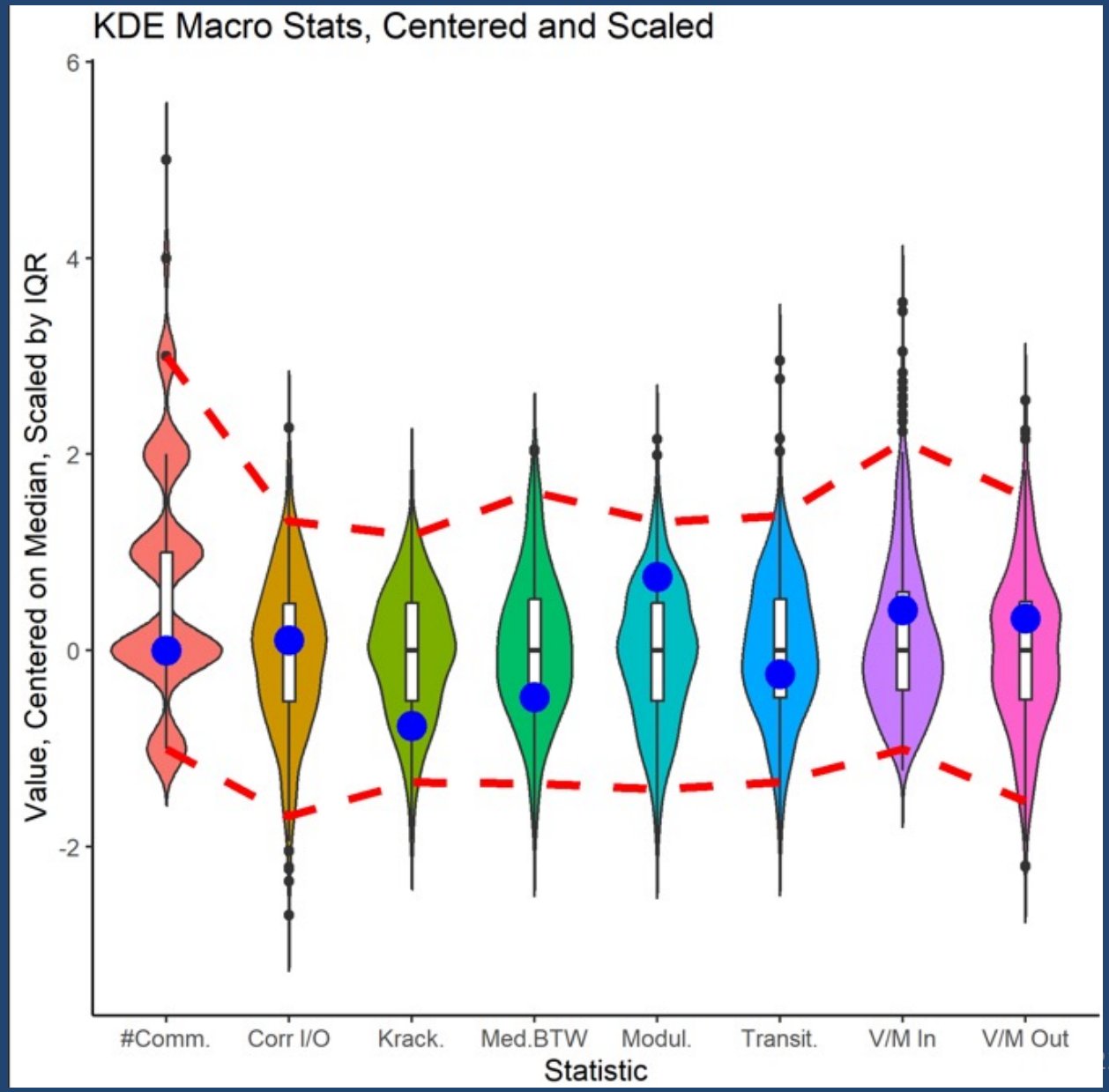
i) #components

ii) Comp. Size

iii) G50

iv) Diameter

v) #of agents per community



To sum up – Our model

- Advice networks are the spontaneously emerging “nervous systems” of organizations
- In competitive contexts, it is puzzling to observe dense and complex networks, since asking for advice is costly in terms of status
- Previous research: SAOM
- Our approach: Same contexts, same network, different assumptions, show *another empirically plausible advice formation model*
- *Guide future steps, to say something more about the phenomenon: E.g., search for thresholds (measure them)*
- *Exclude complex network formation behaviors? Say something, even when data is scarce or absent*

How to actually implement an ABM?

ABM: Building Blocks

- Agents (nodes of the network) → can have various attributes, either observed or not (e.g., in Lazega we observed seniority, but we augmented the dataset with skills and τ) → Correlation?
- Scheduling algorithm: decide who is acting and when (e.g., which agent can send or withdraw an outgoing link)
- Decision making algorithm: how to select advisor (in SAOM framework, objective function, linear combination of network configurations)
- Conditionals or switches: what happens in case a condition is satisfied (If any)(ex: triggered a threshold)
- Routine to estimate parameters and summary statistics for validation

Building blocks → Algorithms

Algorithm 2 Network formation from status preferences and cognitive overload

Require: $N > 0$ (number of agents); α (% of high-skilled agents); τ (cognitive overload threshold); $\beta_0^l, \beta_0^h, \beta_{attract}^l, \beta_{attract}^h, \beta_{EL}^l, \beta_{ER}^l, \epsilon$ (preferences and disturbance); T (number of iterations)

$t \leftarrow 0$

$G = (N, \emptyset)$ ▷ Initialize an empty network, with N nodes, agents

Determine who is high-skilled from data (if available) or randomly

Assign τ to high-skilled agents

while $t \leq T$ **do**

  $i \leftarrow Rand(1, N)$ ▷ Randomly select an agent

  if i is **low-skilled** (l) **then**

    Evaluate $f_i^l(\boldsymbol{\beta}, X)$ for each $j \neq i$ and for the do-nothing case

    Pick j that maximizes $f_i^l(\boldsymbol{\beta}, X)$, consider to do-nothing

    if j is high-skilled and In-Degree $(j) > \tau$ **then**

      Remove and redirect between 1 and τ l-agents asking to j

      for Every redirecting low-skilled l **do**

        Evaluate low-skilled agents via third term of Eq. (4)

        Pick j that maximizes $f_i^l(\boldsymbol{\beta}, X)$, consider to do-nothing

        Set x_{ij} to $x_{ij}^{\pm}$, if best option is to add or remove a link

      end for

    else

      Set x_{ij} to $x_{ij}^{\pm}$, if best option is to add or remove a link

    end if

  else if i is **high-skilled** (h) **then**

    Evaluate $f_i^h(\boldsymbol{\beta}, X)$ for each $j \neq i$ and for the do-nothing case

    Pick j that maximizes $f_i^h(\boldsymbol{\beta}, X)$, consider to do-nothing

    Set x_{ij} to $x_{ij}^{\pm}$, if best option is to add or remove a link

  end if

  $t \leftarrow t + 1$

end while

Algorithms → Computer code

Use languages Optimized for OOP (Python, C++, Java, Scala) or Agent-based languages (NetLogo, Repast) → Easier and cleaner to embed in further code for analysis, parameters exploration/estimation

$G = (N, \emptyset)$ ▷ Initialize an empty network, with N nodes, agents
Determine who is high-skilled from data (if available) or randomly

Social Networks

Volume 17, Issue 1, January 1995, Pages 1-26

Actors and networks as objects ☆

Norman P. Hummon, Thomas J. Fararo

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/0378-8733(94)00245-6 Get rights and content

```
if type_
    attr
    attr
    max_
    if v
```

```
ated
g("low"), "Hold"
shold"][max_index]:
```

Computer code → Run experiments

```
import gridSearch as gr
import os

os.chdir('') #place here YOUR directory, where you will see the edgelists

#Define combinations of parameters to investigate as a dictionary, where key:param_name, value:list of values to explore

exp = gr.ABMSweeps(**{'N': [71], 'alpha': [0.30, 0.70], 'tau':[15], 'outdeg_l':[-1], 'outdeg_h':[-3],
                      'beta_attr_high':[2.5], 'beta_exploit_low':[1.0], 'beta_explor_low':[0.3, 0.8],
                      'epsilon_location':[0], 'epsilon_scale':[0.3], 'nu_location':[0], 'nu_scale':[0.3],
                      'change_factor_attract':[1]})

#Set up grid of experiment; grid = Cartesian product of parameters
grid_exp = exp.set_up_grid()

nets_to_generate = 100 #for each parameters' combination we generate 100 networks
t = 0
T = 200
path_where_to_save_ts = '' #place here YOUR directory path, in which you want to write time series in .csv
edges_to_stop = 20

results_exp = exp.grid_search(nets_to_generate, t, T, stopping_condition = edges_to_stop, compute_all = True, path_ts =
path_where_to_save_ts)
```

Run experiments → Estimate parameters

```
with pm.Model() as my_model:
    exploit = pm.Normal("beta_exploit", mu=0, sigma=3)
    tau = pm.HalfNormal("tau", sigma=1)
    s = pm.Simulator("s", NetEvolution, params=(exploit, tau), \
                     sum_stat="lazega_metrics", epsilon=2, observed=lazega_net)
    idata = pm.sample_smc()
    idata.extend(pm.sample_posterior_predictive(idata))
```

NetLogo – Pedagogical tool

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add Input

normal speed

ticks: 3297

view updates

continuous

Settings...

setup step iterate

max-links 300

possible-number-links 1560

N 40

number-networkers 30

#satisficers (breed1)

beta_outdeg_sat -1.1

beta_hom1_sat 0.4

beta_attract_seniority 0.008

attr gender

#networkers(breed2)

beta_outdeg_net -1

beta_rec_net 0.9

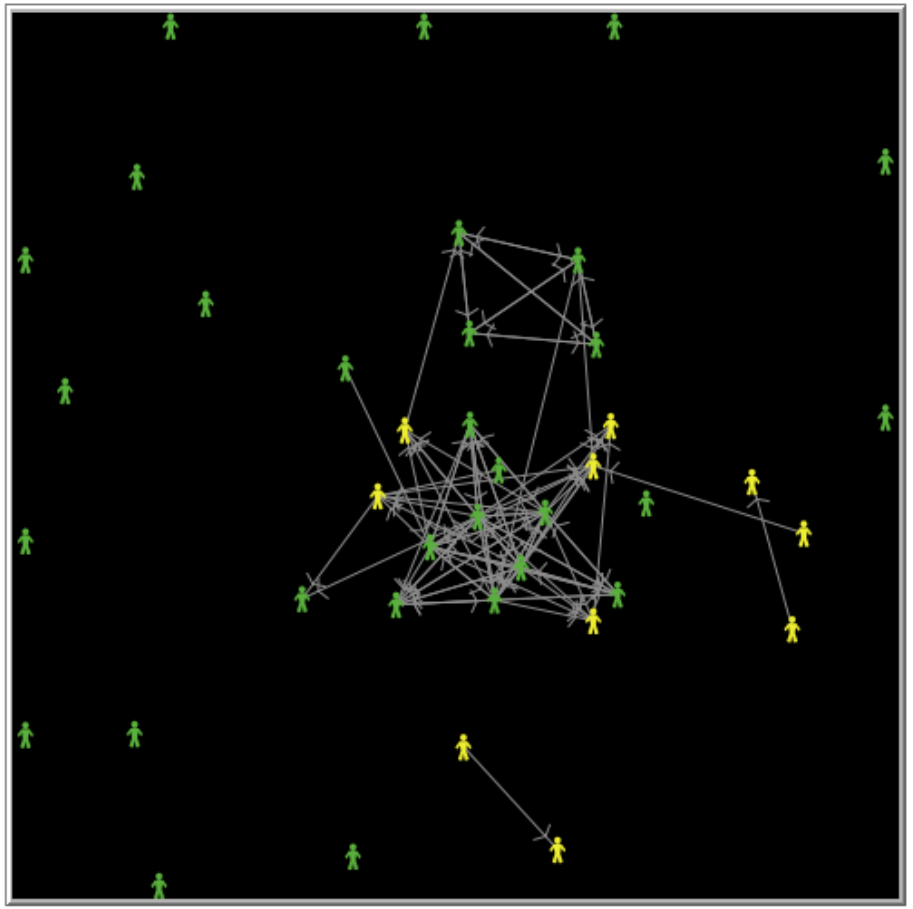
beta_trans_net 0.7

#parameters for Gamma distr. of shocks

alpha 0.2

zeta 2

[GitHub Link]



CoMSES NETWORK

About Model Library Education Events Jobs Resources Forums Sign In

A growing collection of resources for computational model-based science.

AN INTERNATIONAL COMMUNITY AND CYBERINFRASTRUCTURE TO SUPPORT TRANSPARENCY AND REPRODUCIBILITY FOR COMPUTATIONAL MODELS & THEIR DIGITAL CONTEXT + EDUCATIONAL RESOURCES AND FAQ'S FOR AGENT BASED MODELING

Browse the Model Library

Dashboard

Type / to search

Top Repositories

New

Find a repository...

- ceco51/Status-hierarchy-cognitive-overload-and-incomplete-network-information-in-advice-seeking-networks-
- ceco51/Evolutionary-network-formation
- ceco51/SIENA-like-model-BEHAVE-SummerSchool
- ceco51/Brokerage
- ceco51/ABM-Peer-Review-NK-
- ceco51/Trial-Repo-to-Submit-SMR-

For you Beta | Following | Send feedback | Filter

pyscript has a new discussion in Announcements · 3 weeks ago

pyscript / pyscript

PyScript NEXT (new core) Tech-Preview Release!!

Announcement: PyScript NEXT (new core) Tech-Preview Release!!

After invaluable community feedback, many experiments in code, and careful cooperation with valued collaborators, we are thrilled to announce the tech preview of the new version of PyScript. This work marks a significant milestone in our journey towards PyScript's vision and what our commun...

Read more

Latest changes

Yesterday
Migrate Bitbucket Server and Data Center repos to GitHub.com (public beta)

Yesterday
GitHub Actions: Removal of Node12 from the Actions runner

4 days ago
GitHub Copilot July 14th Update

4 days ago
Codespaces Port Forwarding Domain Name Updates

View changelog →

Thanks! Any questions?

Contact me: francesco.renzini@unimi.it

Twitter: @FrancescoRenz27