# Simulating Games on Networks with **R**

## Application to Coordination in Dynamic Social Network Under Heterogeneity

Michał Bojanowski

ICS / Department of Sociology
Utrecht University

UseR!2008, Dortmund

# Introduction: Networks in social sciences

- **"No man is an island!"**
- Outcomes of social and economic processes are determined not only by actors' attributes but also by the structure of relations between them (Granovetter, 1985).
- Actors (nodes/vertexes): individuals, organizations, states...
- Relations (links/edges): cooperation, friendship, communication, joint activities...
- Node attributes: gender, race, age, music tastes...
- Dyadic attributes: geographical distance, taste similarity...

# In this talk

- Using **R** to aid theory development
- (No data, . . . at least from the real world)
- Framework for simulating certain kinds of models of social network dynamics
- Example of specific theoretical model

# Outline

# Network games

- Game-theoretical models of network and behavior dynamics
- Actors' action spaces include both relational and behavioral alternatives
- Network utility function of actor $i$

$$U_i : \mathcal{G}, \mathcal{X} \mapsto \Re$$

$$U_i(g, \mathbf{X})$$

where $g \in \mathcal{G}$ is the graph, and $\mathbf{X} \in \mathcal{X}$ is the matrix of individual (node-level) attributes

- Existing models: Connections, Co-author (Jackson & Wolinski 1996), coordination (Goyal & Vega-Redondo 2005, Jackson & Watts 2002), R&D collaboration (Goyal, 2007) and more
- "Solving" by looking for various forms of equilibria/stability

# Coordination in dynamic social network

- Fixed population of $n$ actors composed of two groups (**types**) $A$ and $B$
- Every actor chooses one of the two behavioral options $x$ or $y$ (**behavior**)
- Actors form an undirected network $g = [g_{ij}]_{n \times n}$

Utility of actor $i$:

$+w$ for every relation with actor behaving the same as $i$

$+b$ if $i$ is of type $A$ choosing $x$ or of $B$ choosing $y$

$-\alpha\mu_i - \beta\mu_i^2$ Cost of maintaining ties where $\mu$ is the total number of relations of $i$

# Network benefits

| | $A$ | |
|---|---|---|
| $A$ | $x$ | $y$ |
| $x$ | $b+w,\ b+w$ | $b,\ 0$ |
| $y$ | $0,\ b$ | $w,\ w$ |

Within type $A$

$$0 < b \leq w$$

| | $B$ | |
|---|---|---|
| $B$ | $x$ | $y$ |
| $x$ | $w,\ w$ | $0,\ b$ |
| $y$ | $b,\ 0$ | $b+w,\ b+w$ |

Within type $B$

| | $B$ | |
|---|---|---|
| $A$ | $x$ | $y$ |
| $x$ | $b+w,\ w$ | $b,\ b$ |
| $y$ | $0,\ 0$ | $w,\ b+w$ |

Between types

# Solution concept

**Pairwise stability** (Jackson & Wolinsky, 1996)

## Definition (Pairwise stability)

The network $g$ is **stable** if and only if the following three conditions are jointly satisfied:

1. There is no pair of actors in $g$ who would benefit from creating a tie.
2. There is no actor in $g$ who is interested in deleting a tie
3. No actor would benefit from changing his behavior

# Simulation setup

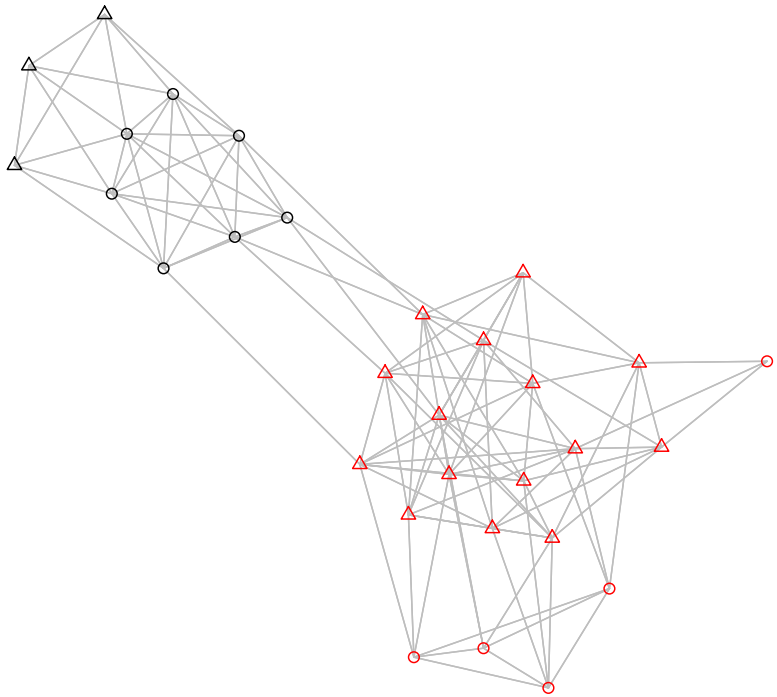
- Generate a set of initial conditions: model parameters, initial network, type and behavior distribution
- Actors update their network or behavior in a random order
- Tie formation requires consent from the other player, deletion not (bilateral formation, unilateral deletion)
- The process is run until no change is possible. The final state is pairwise-stable

# Results

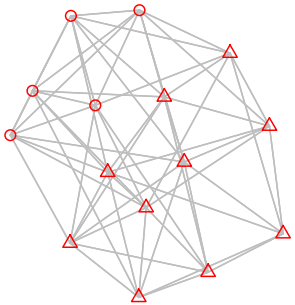
- Small scale: qualitative analysis (visualizations with pictures and movies)
- Large scale: statistical analysis of generated data

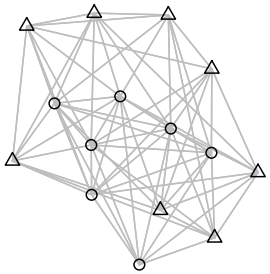
# Types of stable networks

## Connected center-periphery structures

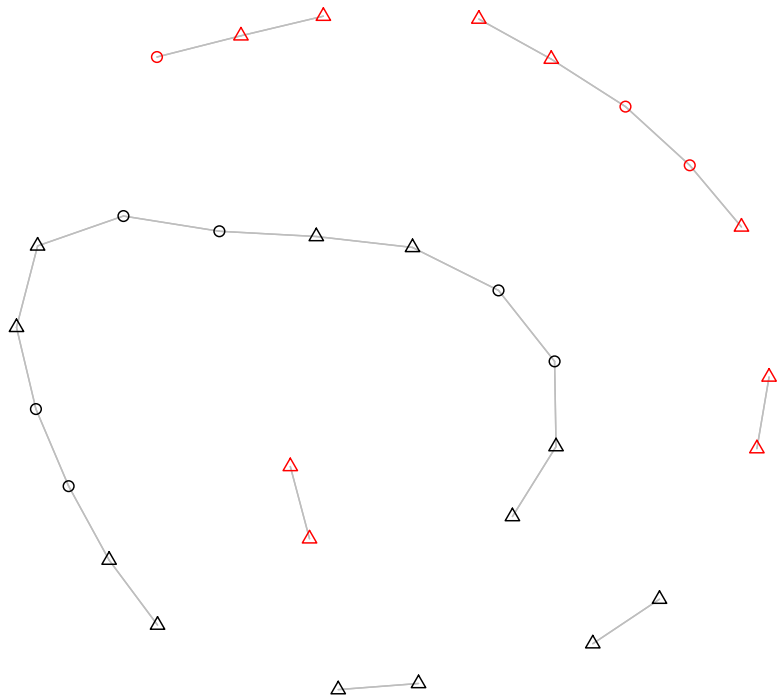


# Behavior-segregated components

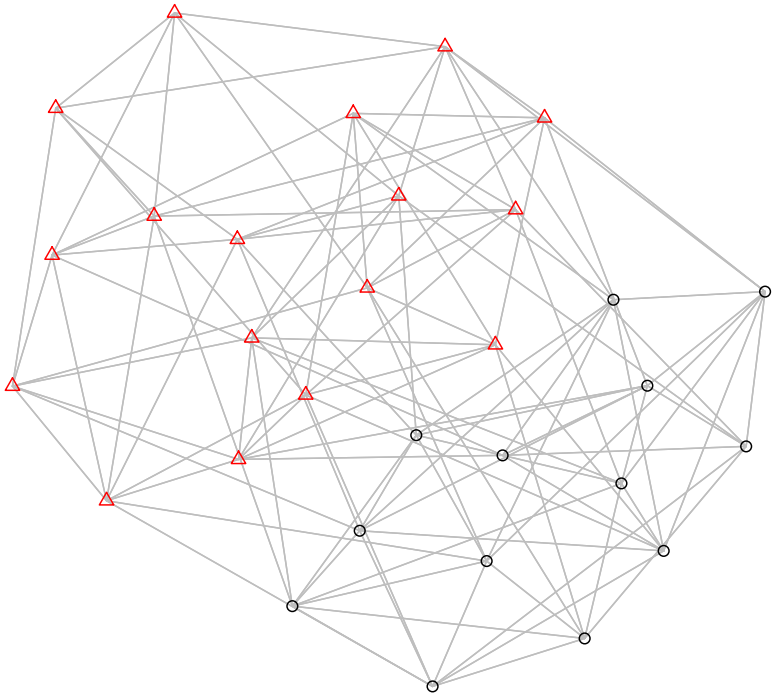




# Sparse "lines"



# Fairly integrated populations of "native" players



▸ show movie

# Used packages

- `network` for storing networks with vertex attributes
- `simecol` as the simulation workhorse
- `sna` and `rSoNIA` for network analysis and visualization
- plus a lot of tweaking (saving results, reading condition data)

# Summary and TODO

Conclusions

- **R** is a convenient simulation platform, although perhaps not the most efficient
- Already some social network analysis functionality (`network`, `dynamicnetwork`, `igraph`, `sna`, `ergm` and more)

On the agenda:

- Finishing development of a package for simulating any network utility function
  - Modular architecture: type of dynamics, modeling dyadic interactions, tie cost functions, reputation, beliefs
  - Flexible result saving
  - Visualization