Wrocław University of Technology

MuNeG - The Framework for Multilayer Network Generator

Adrian Popiel, Tomasz Kajdanowicz, Przemysław Kazienko

Wrocław University of Technology, Poland

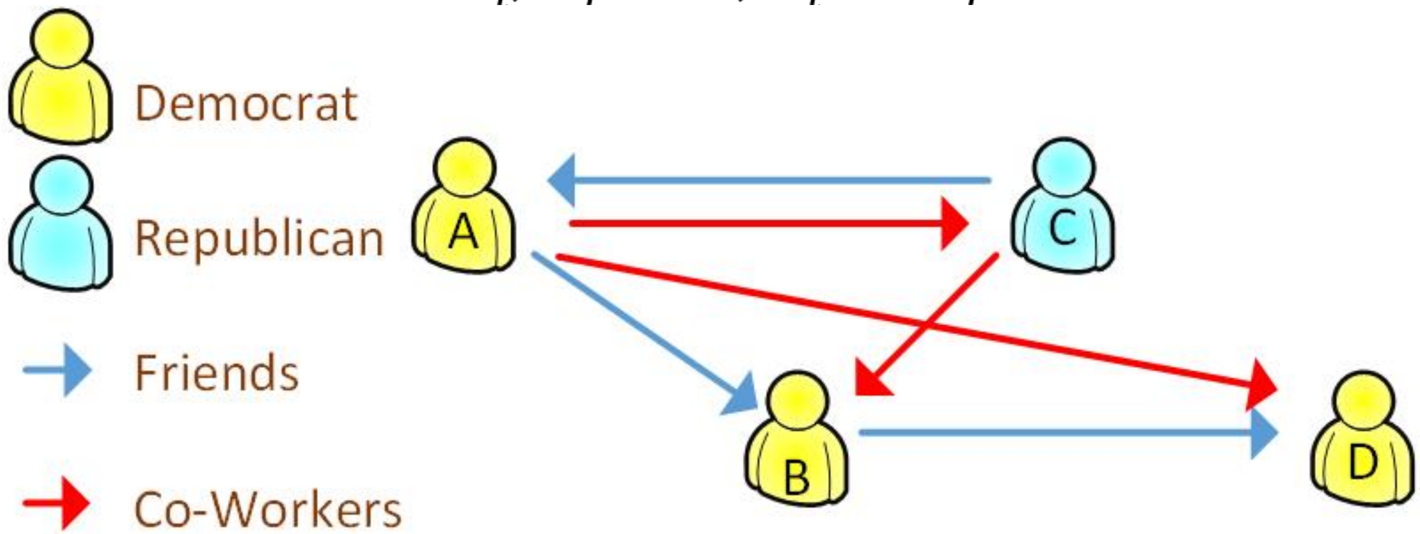
MANEM 2015, August 25, 2015

Outline

- Uniplex and multiplex networks
- Network models
- Network properties
- Multiplex network generation
- Behind MuNeG
- Experimental setup
- Results
- Conclusions

Uniplex and multiplex networks

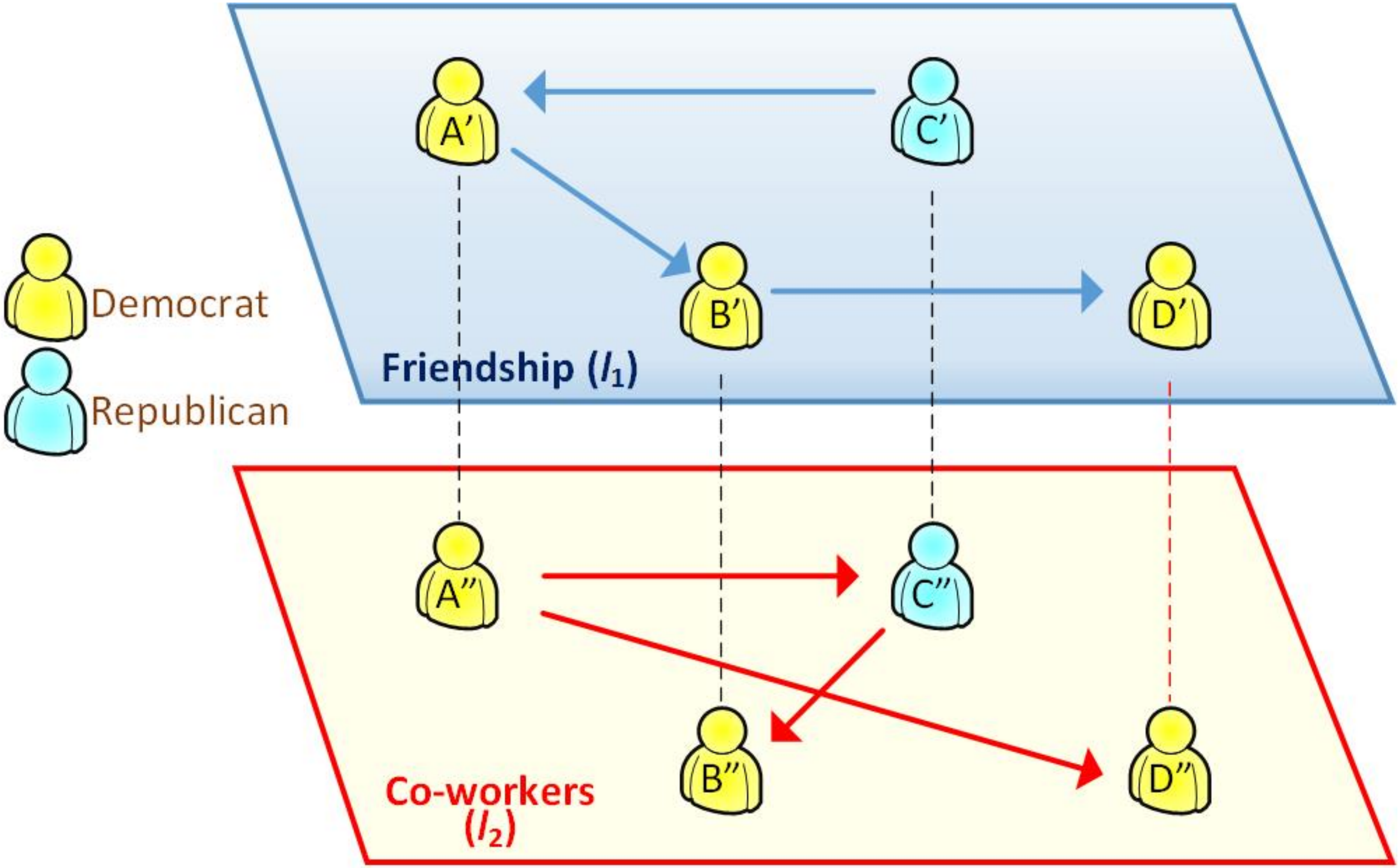
- A uniplex network $G = (V, E)$
  - Nodes: $V = \{v_1, \ldots, v_n\}$
  - Directed edges $\forall e \in E$, $e = (v_i, v_j)$, $v_i, v_j \in V, v_i \neq v_j$



Uniplex and multiplex networks(2)

- **Multiplex** is a **multi-layer** network (multigraph) $\mathrm{MG} = (V, V^L, E, EL, L)$
  - V - set of nodes
  - L - set of uniplexes / layers
  - each node v_i has own representation $v_{il} \in VL \in V^L$ in each layer $l \in L$
  - edges on layers: $e = (v_{il}, v_{jl}, l, w_{ijl}), l \in L$
- Nodes representations are connected via special links between their representations from l to k: $(v_{il}, v_{ik}) \in EL$

Uniplex and multiplex networks(3)



Network models

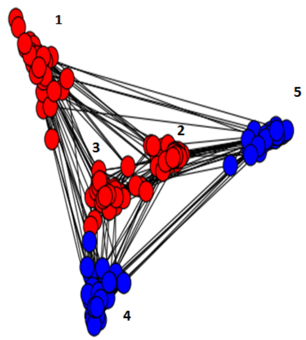
- Random graphs(Erdős-Rényi model)
  - Edge existence comes from binomial distribution
- Configurational model
  - Takes degree distribution as an input
- Small worlds(Watts and Strogatz)
  - Uses node degree as input
- Scale-free networks(Barabási-Albert)
  - Networks with power-law distribution

Properties of a network

- Node degree
- Maximum node degree
- Number of edges
- Clustering coefficient
- Number of triangles
- Average shortest path
- Diameter

MuNeG

- Version 1.0 is released
  - https://github.com/Adek89/MuNeG/releases/tag/1.0
- Generator is still in development
  - Version 1.1-SNAPSHOT: https://github.com/Adek89/multiplex/tree/master/MuNeG



Main goal of MuNeG

- Generate multiplex networks for a collective classification
- Enhance network models to domain of multiplex networks
- Generate networks with expected properties
- Generate networks similar to real data
- Check if generated networks are similar to existing network models

Real data properties

(Newman, Mark EJ. "The structure and function of complex networks." *SIAM review* 45.2 (2003): 167-256.)

Network domain and name	Number of nodes	Mean node degree	Clustering coefficient	Mean distance between nodes
Social – student relationships	573	1.66	0.005	16.01
Information – Roget's Thesaurus	1022	4.99	0.13	4.87
Technological - Internet	10697	5.98	0.035	3.31
Biological – protein interactions	2115	2.12	0.072	6.80

MuNeG algorithm – input parameters

- MuNeG takes as an input 6 parameters:
  - Number of nodes - N^V
  - Number of groups - N^{Gr}
  - Group homophilly - p_{Gr}
  - Probability that two nodes from same group are connected - p_{in}
  - Probability that two nodes from different groups are connected - p_{out}
  - Number of layers - L

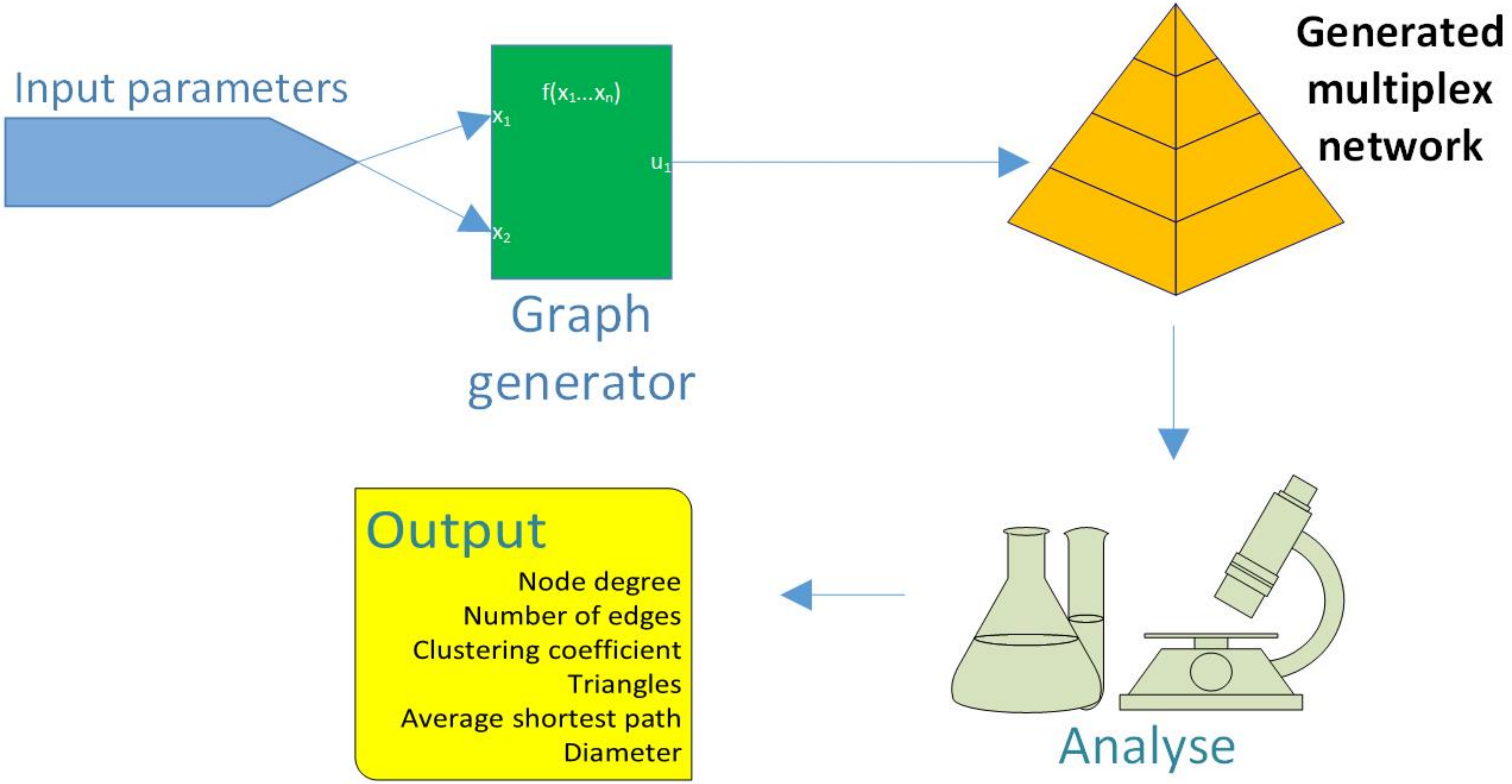
MuNeG algorithm - groups

- MuNeG generates networks with binary labelings
  - $C = \{0, 1\}$
- Groups represent communities
- Group can be red or blue with same probability:
  - $P(Gr = red) = P(Gr = blue) = 0.5$

MuNeG algorithm - homophilly and edge existence

- Group homophilly answers for labels in groups
- Group color influences on labeling:
  - $p_{Gr} = P(C = 0|Gr = red) = P(C = 1|Gr = blue)$
- Edges inside and outside of groups are dependent on input parameters:
  - $p_{in} = P(E = 1|Gr_i = Gr_j)$
  - $p_{out} = P(E = 1|Gr_i \neq Gr_j)$

Experiments



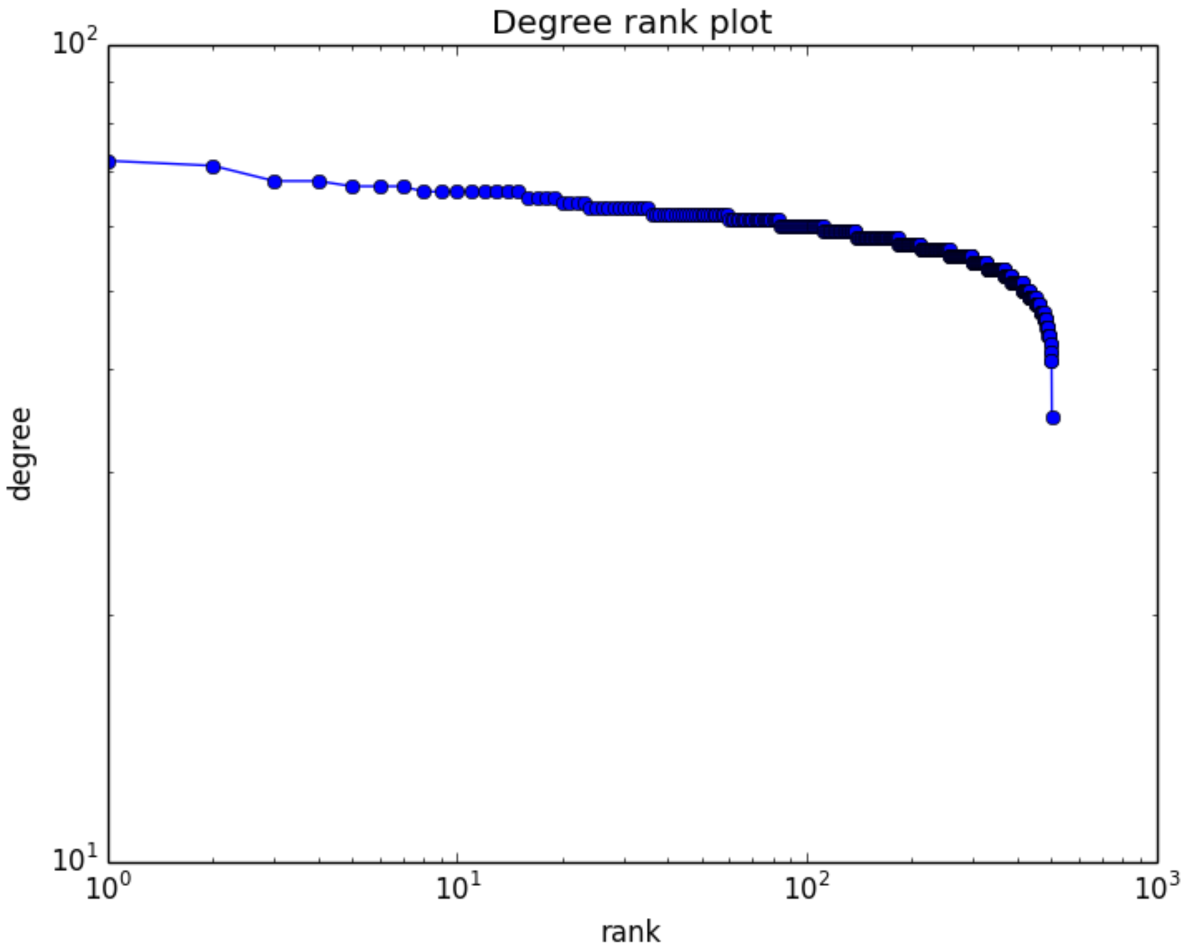
Experiments - parameters

- $N^V = \{100, 500, 1000\}$
- $N^{Gr} = \{2, 3, 4, 5, 6, 7, 8, 9\}$
- $p_{Gr} - < 0.5; 1 >$, with step 0.1
- $p_{in} - < 0.5; 0.9 >$, with step 0.1
- $p_{out} = \{0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$
- $L = \{2, 3, 5, 6, 8, 10, 13, 21\}$
- All combinations of parameters give about 300000 analysed networks

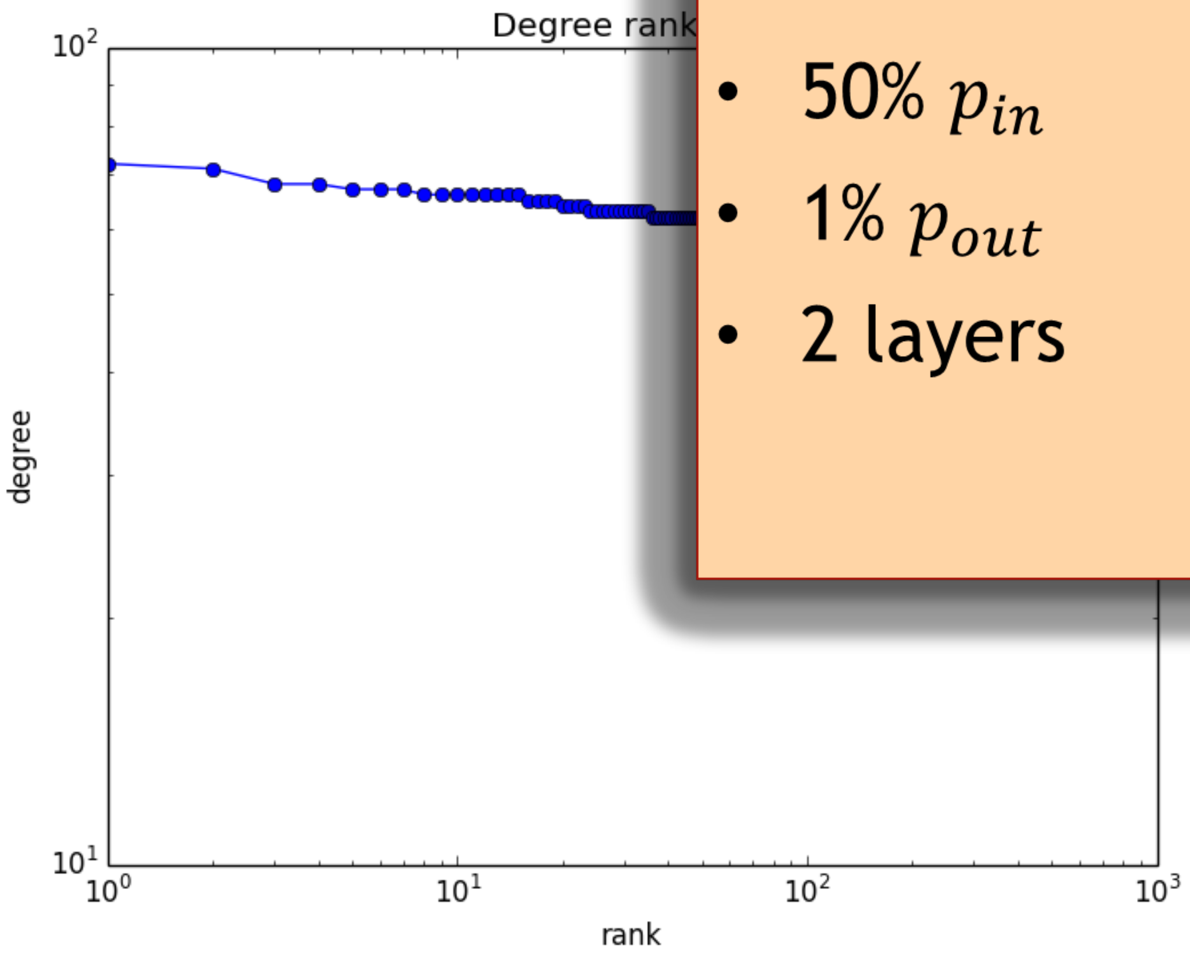
Experiments - calculations

- $$\frac{\sum \frac{\sum value}{number\ of\ nodes}}{number\ of\ experiments}$$
- Where value is:
  - Clustering coefficient
  - Node degree
- Other parameters are averaged over all experiments

Results - node degree

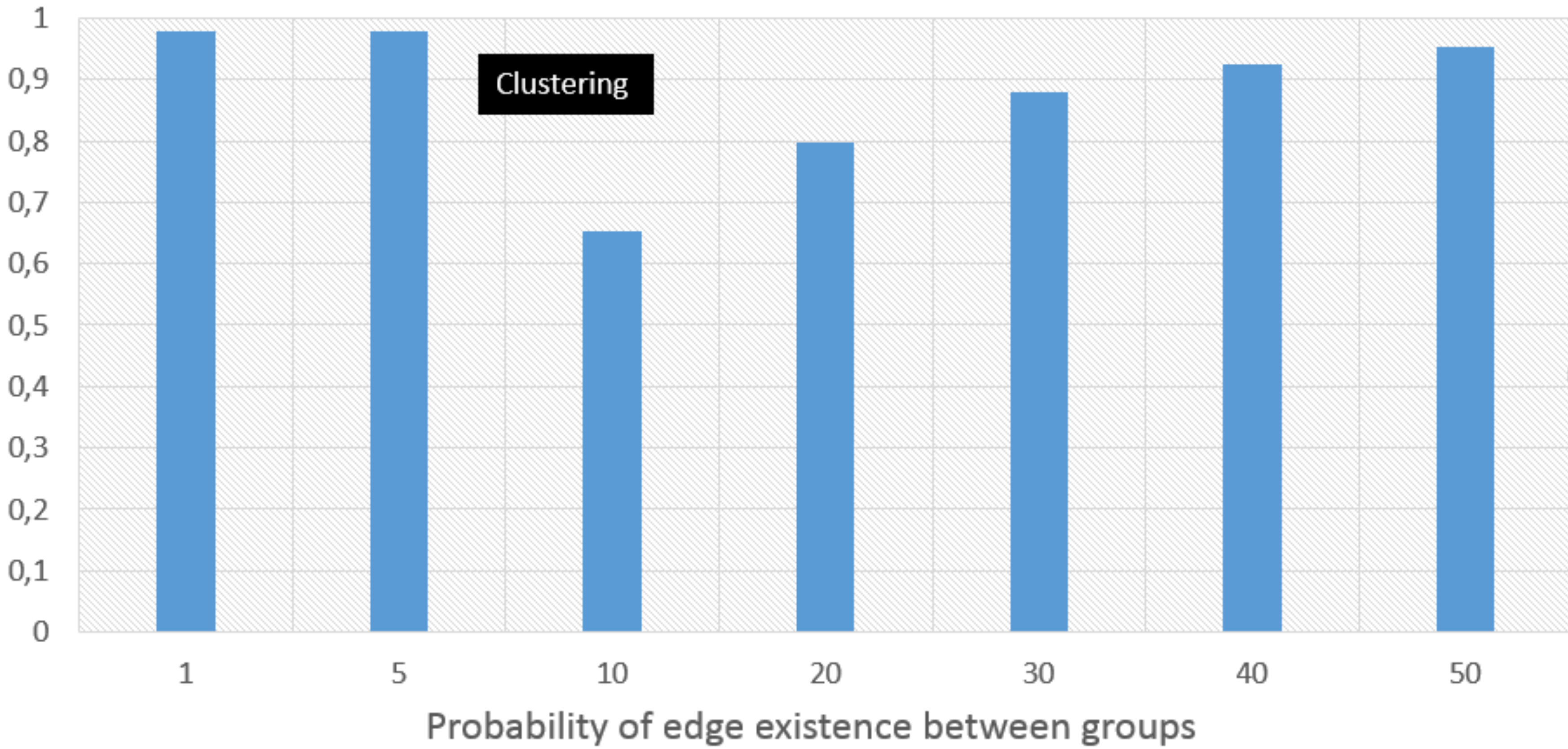


Results – node degree

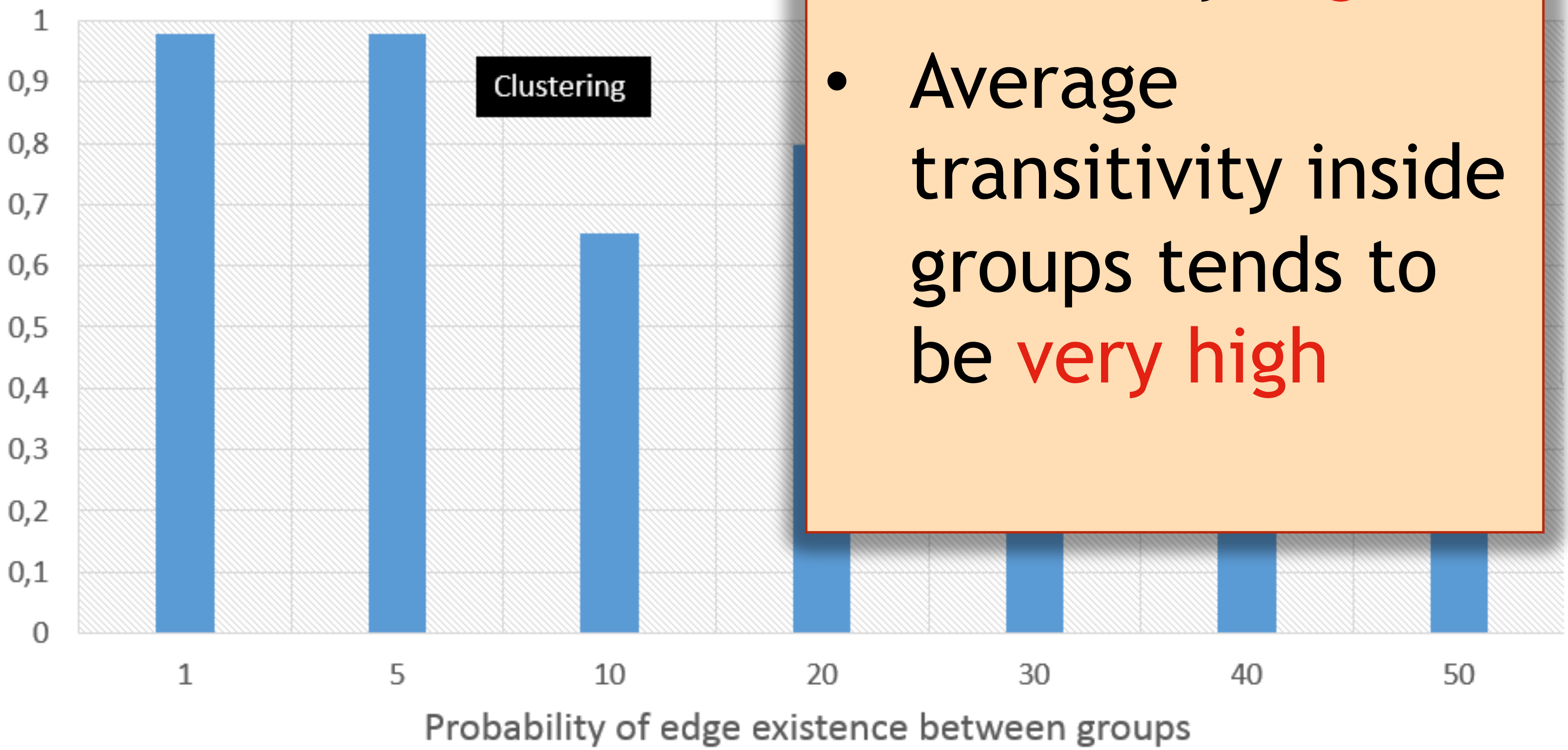


- 500 nodes
- 50% p_{in}
- 1% p_{out}
- 2 layers

Results - clustering

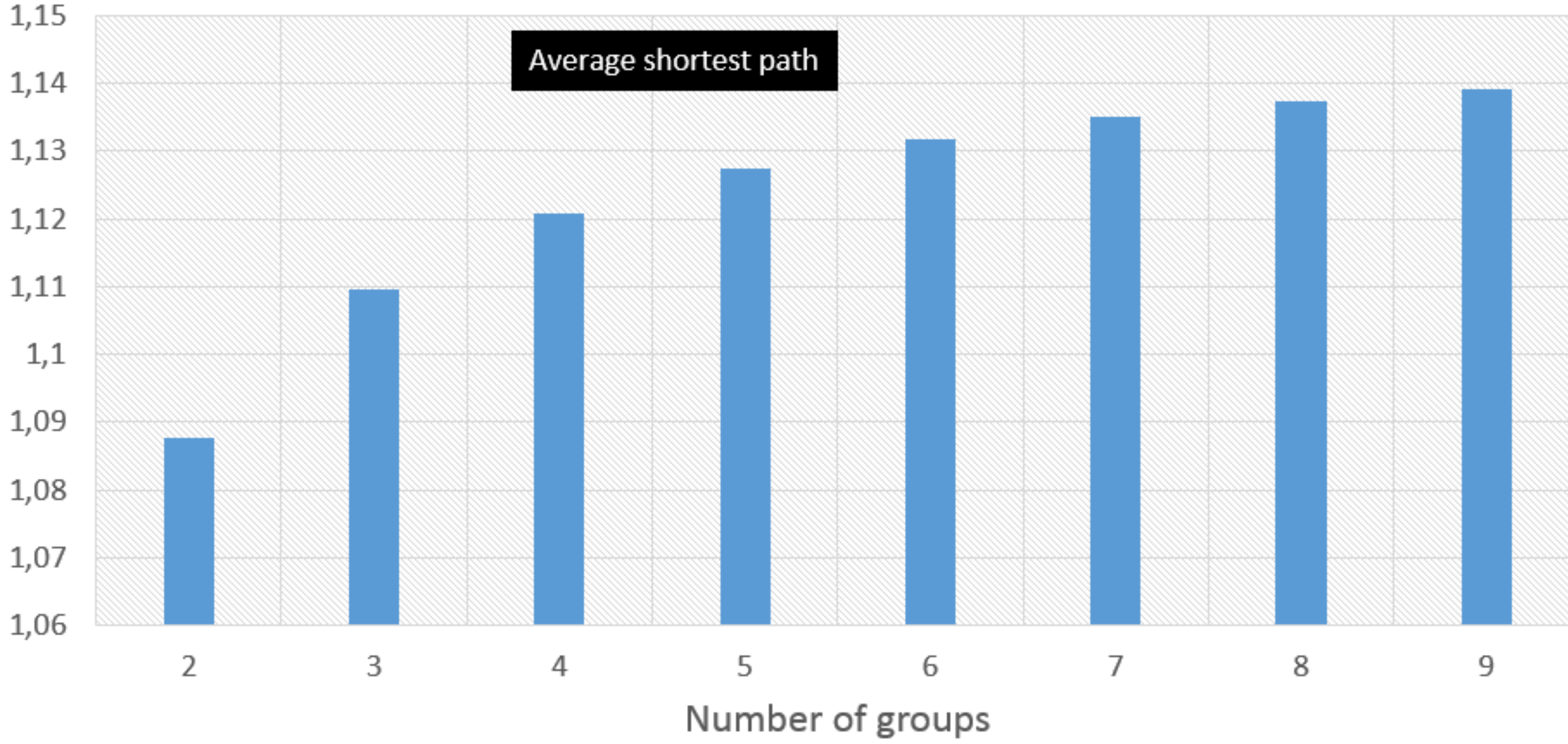


Results - clustering

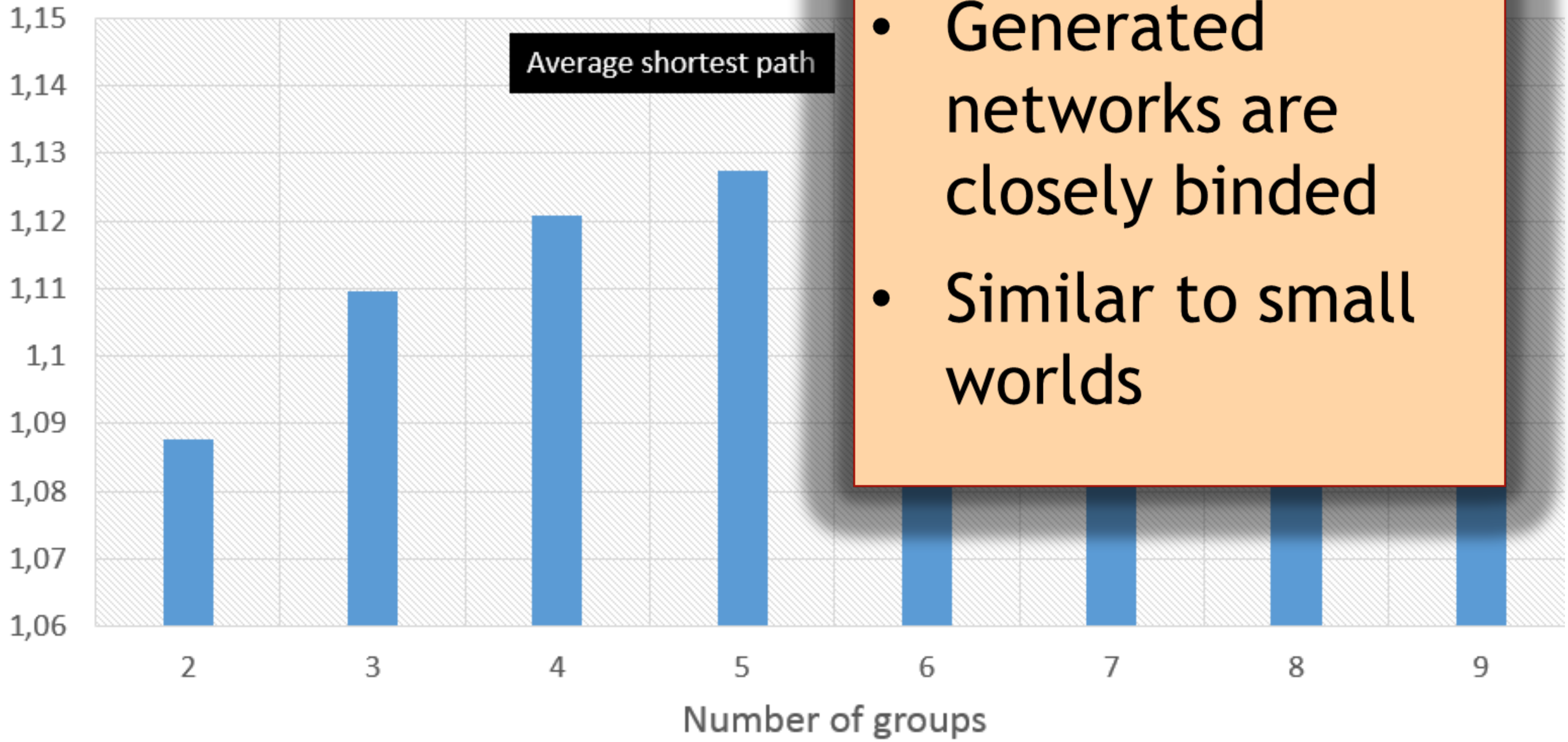


- Clustering is relatively big
- Average transitivity inside groups tends to be very high

Results - average shortest path



Results - average sho



- Paths are short $< 1.0; 1.5 >$
- Generated networks are closely binded
- Similar to small worlds

Conlusions

- MuNeG generated networks are good to simulate close connected complex networks or small worlds - ☺
- Distributions of parameters should be compared with real data distributions
- p_{in} and p_{out} parameters have the most significant influence on generated networks

Future work

- Algorithm improvements:
  - API to generate networks similar to real
  - API to generate mulitplex networks similar to known network models
- Each layer should represent uniplex network similar to known models or real data
- https://github.com/Adek89/multiplex/tree/master/MuNeG

Thank you!


THANK YOU FOR YOUR ATTENTION!
memy.pl
PLEASE CLAP AND DON'T
ASK TOUGHT QUESTIONS
memy.pl