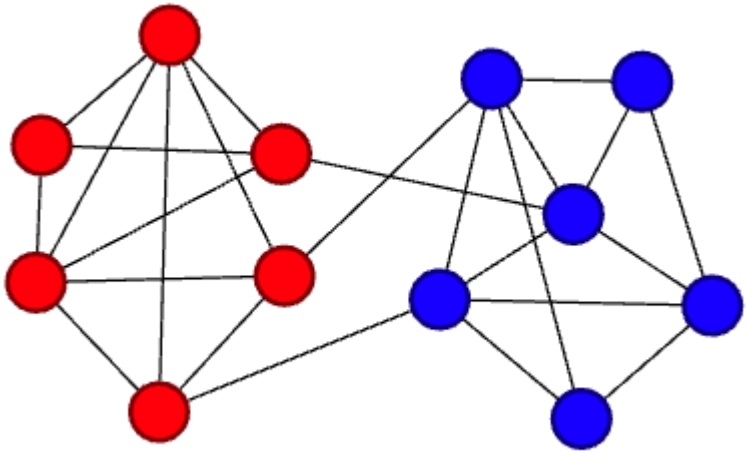
Edge Weight Method for Community Detection in Scale-Free Networks

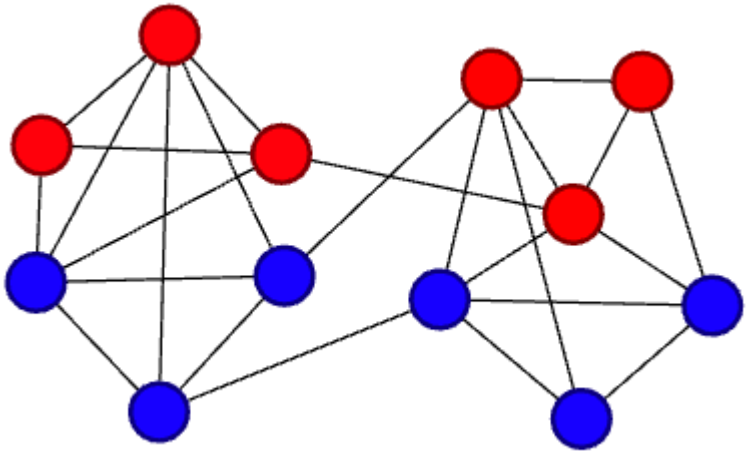
Sorn Jarukasemratana
Tsuyoshi Murata

Tokyo Institute of Technology

WIMS'14
June 2-4, 2014 - Thessaloniki, Greece

Modularity





High modularity score

- edges inside communities = 22
- edges between communities = 3

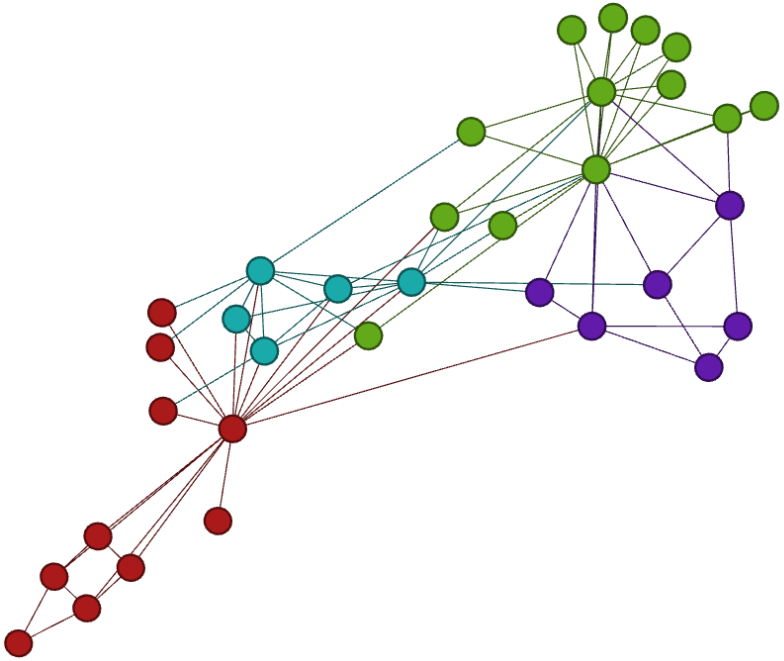
Low modularity score

- edges inside communities = 14
- edges between communities = 11

The higher the modularity the better

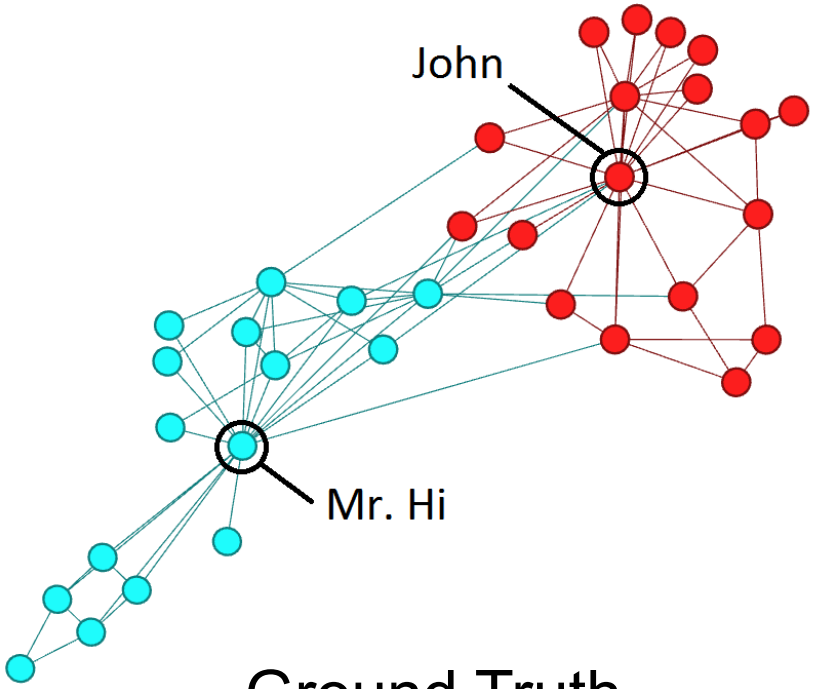
M. Girvan and M. E. J. Newman. Community structure in social and biological networks. 2002

Modularity on network with scale-free property



Louvain Algorithm

Modularity = 0.417



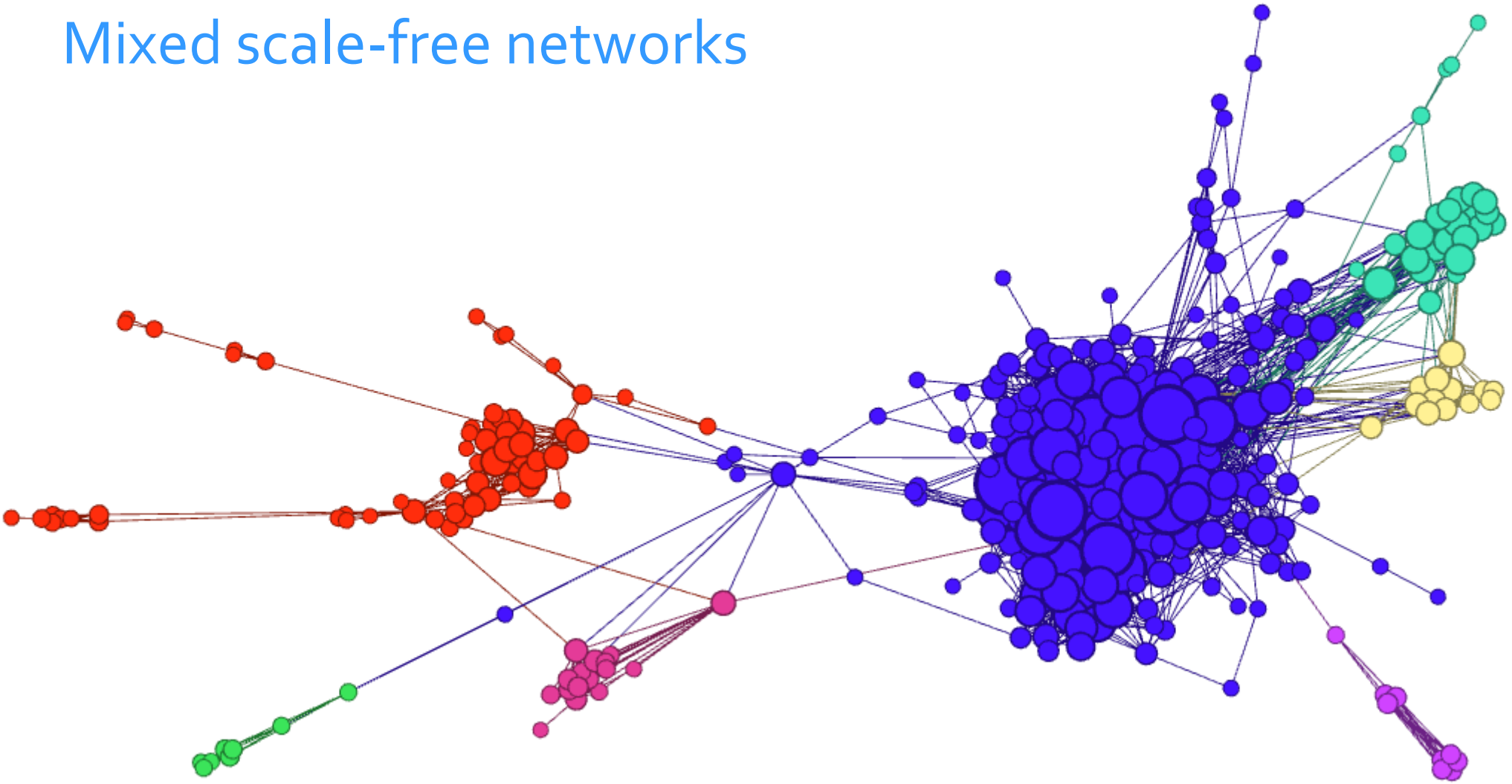
Ground Truth

Modularity = 0.370

Modularity considered all nodes and edges as equal.

W. W. Zachary. An Information Flow Model for Conict and Fission in Small Groups. 1997

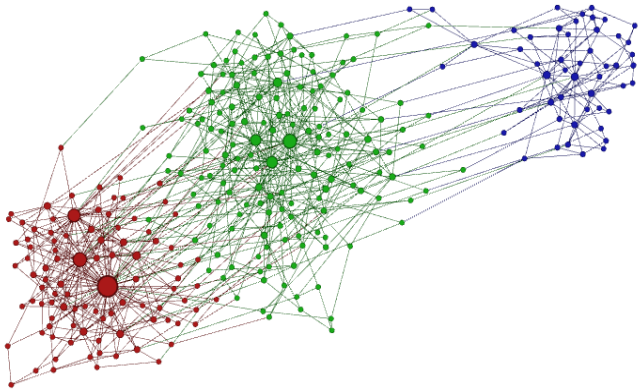
Mixed scale-free networks



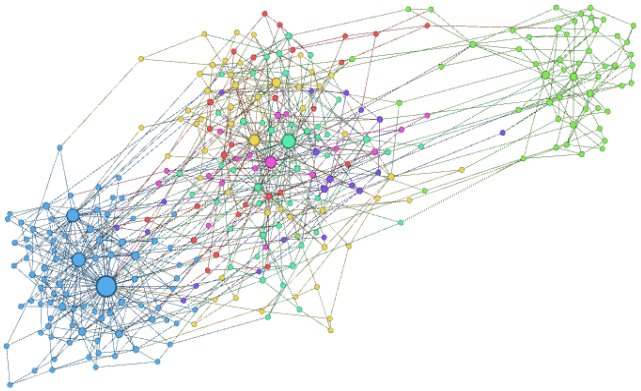
Facebook0 network from Ego Facebook Datasets

Previous limitation

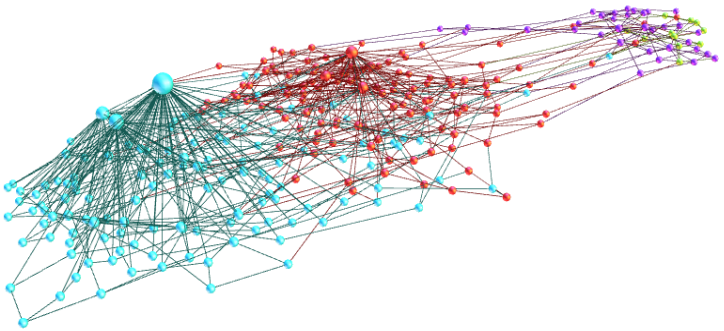
- Graphs that contain both communities with scale-free attribute and non scale-free attribute.



Ground Truth



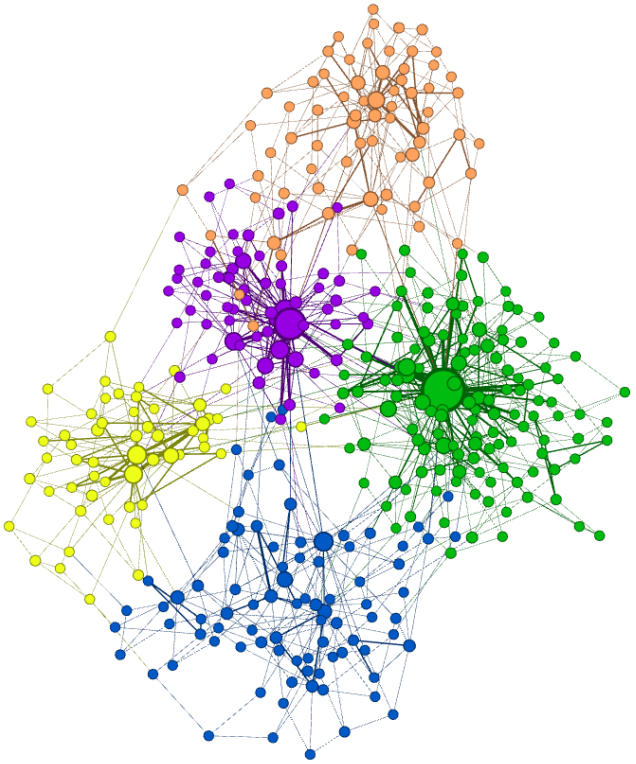
Louvain Algorithm
NMI = 0.480



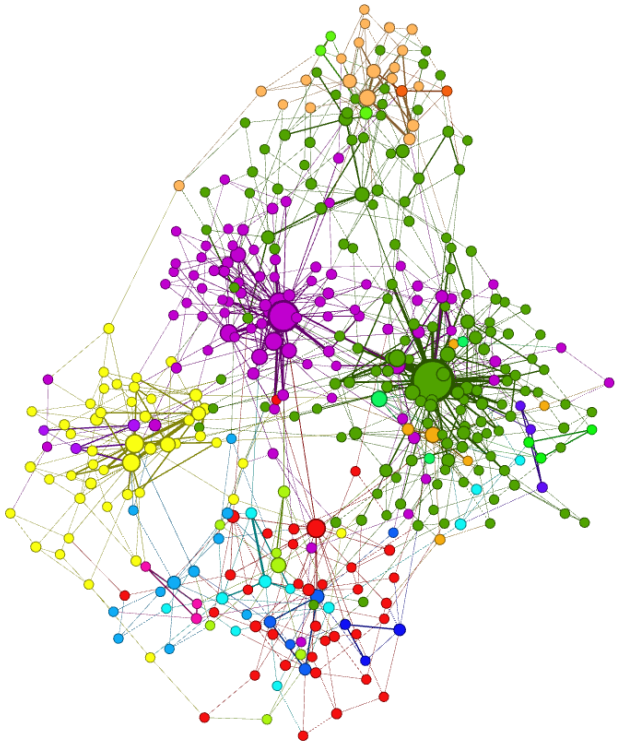
Node Centrality Algorithm
NMI = 0.289

The goal of this research

- Community detection in scale-free networks that some communities' node degree follows power law (scale-free) while some communities' node degree follows normal distribution.



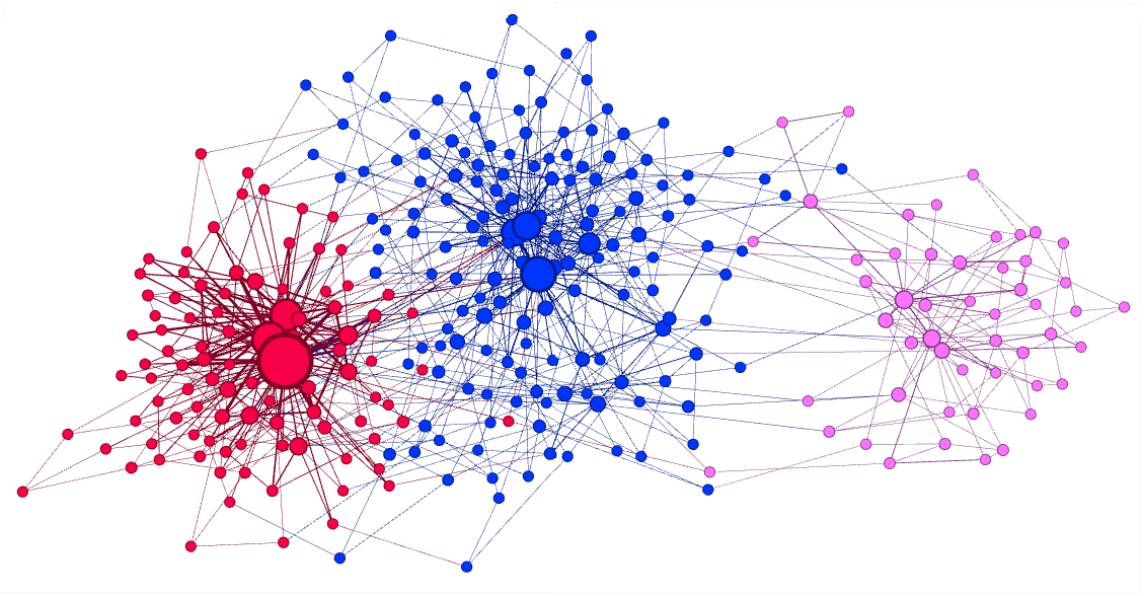
Ground Truth



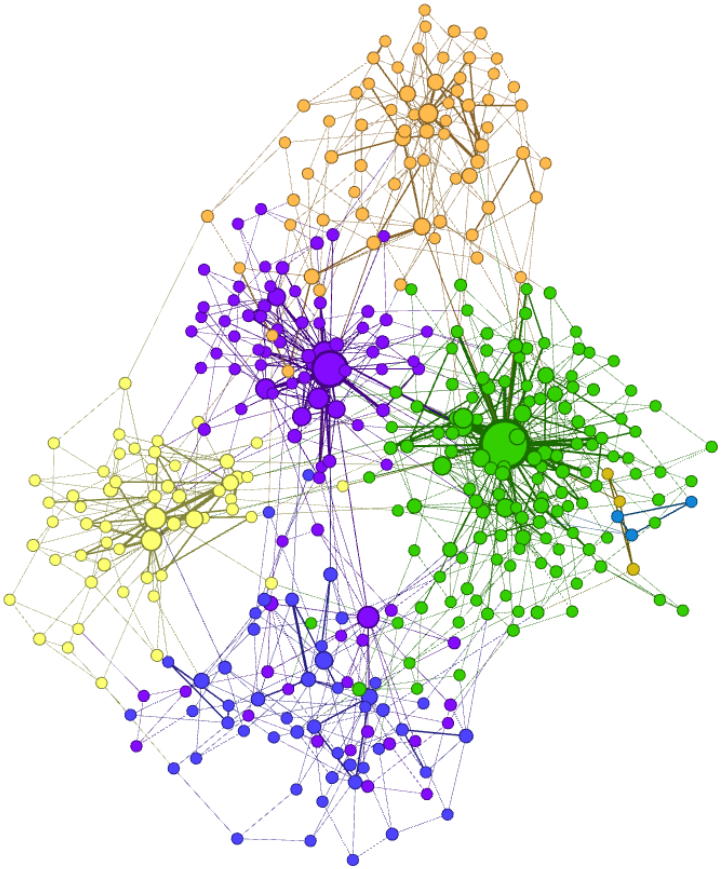
Scale-free Approach
NMI – 0.246

New approach

- Combine previous approach with modularity optimization method
  - Perform scale-free approach then followed by non scale-free approach.



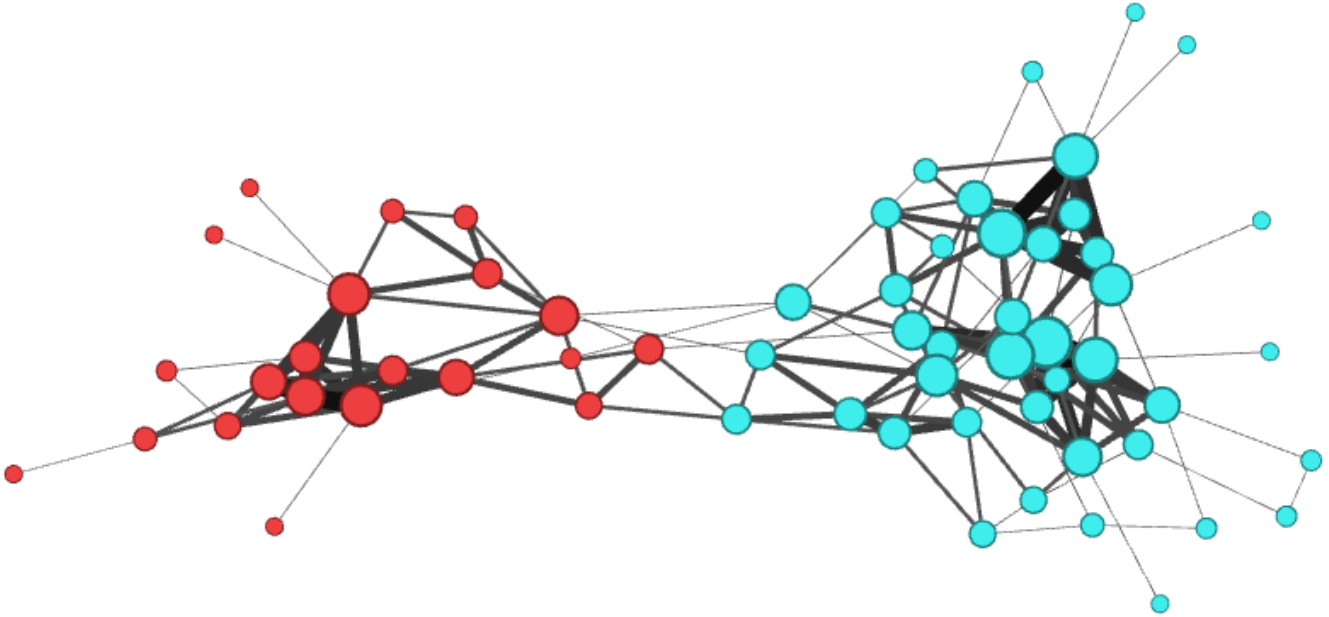
Proposed Method
NMI = 0.951



Proposed Method
NMI = 0.656

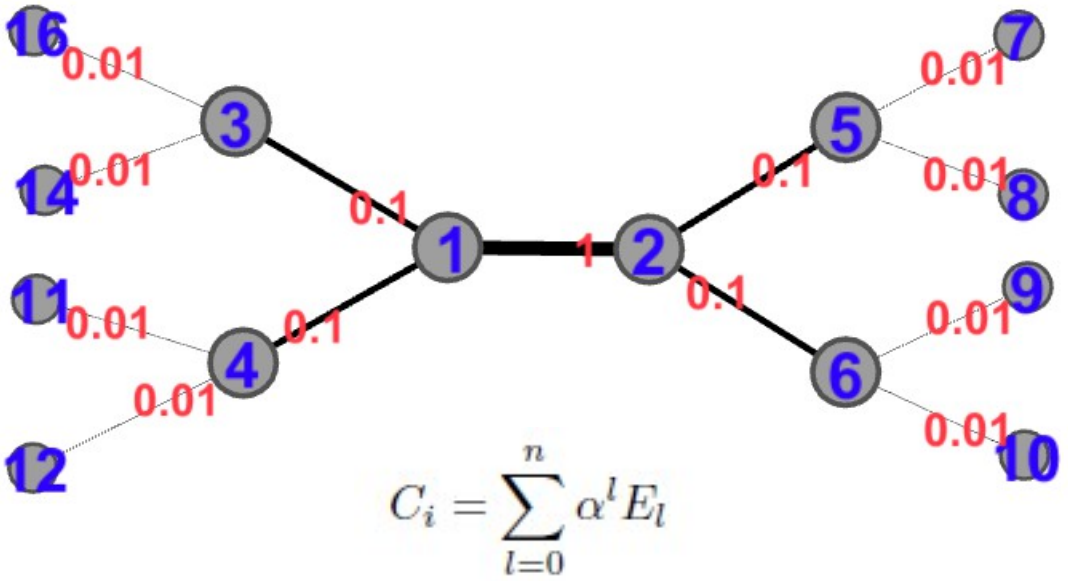
Edge weight

- Similar to rubber band
- Weight is similar to pulling power of the rubber band.
- Edge with high pulling power, should be in the same Community.



Edge weight calculation - 1

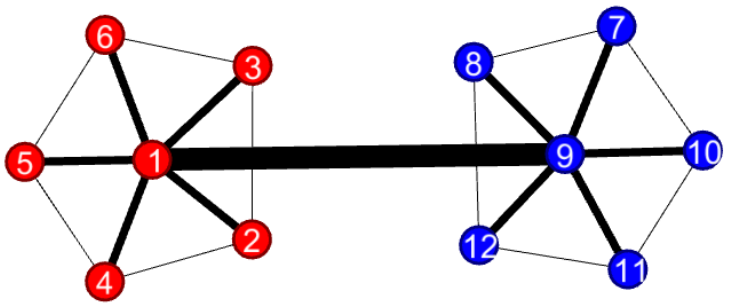
- Centrality – the more central, the more important.
  - Using attenuation factor, similar to Katz centrality. The farther away, less effect it has.



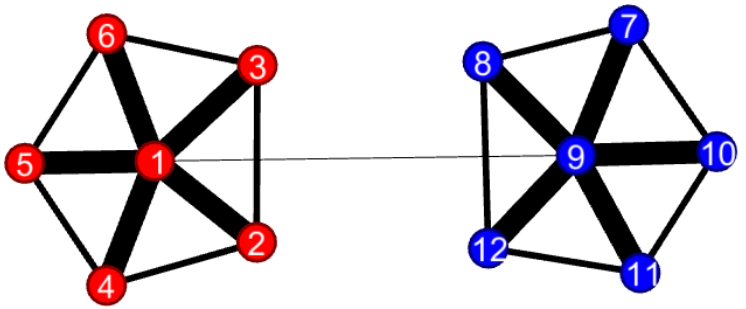
Weight of edge 1-2 = 1 + 0.1*4 + 0.01*8 = 1.48

Edge weight calculation - 2

- Clustering coefficient – used for differentiate edge that connects two communities
  - Can be counted by using triplets (3 nodes, 3 edges forming triangle).



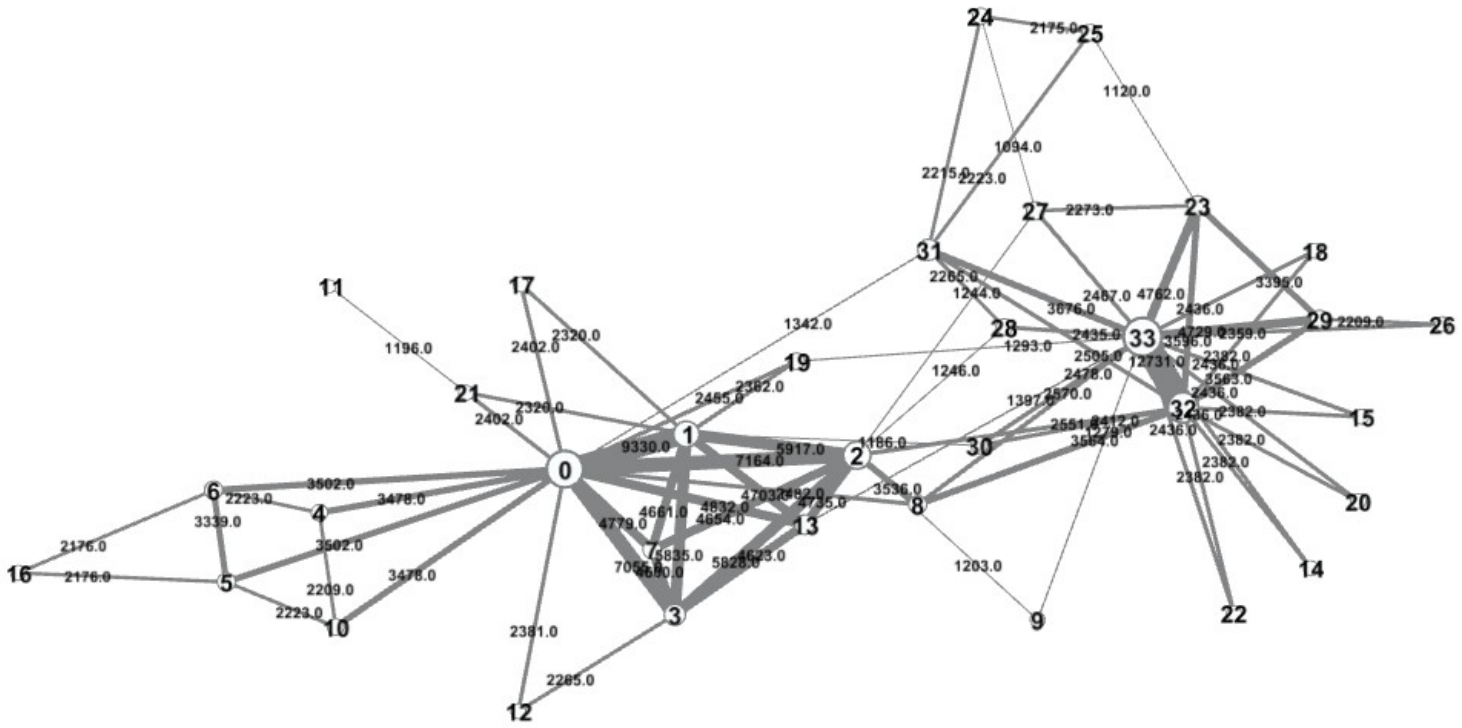
without clustering coefficient



with clustering coefficient

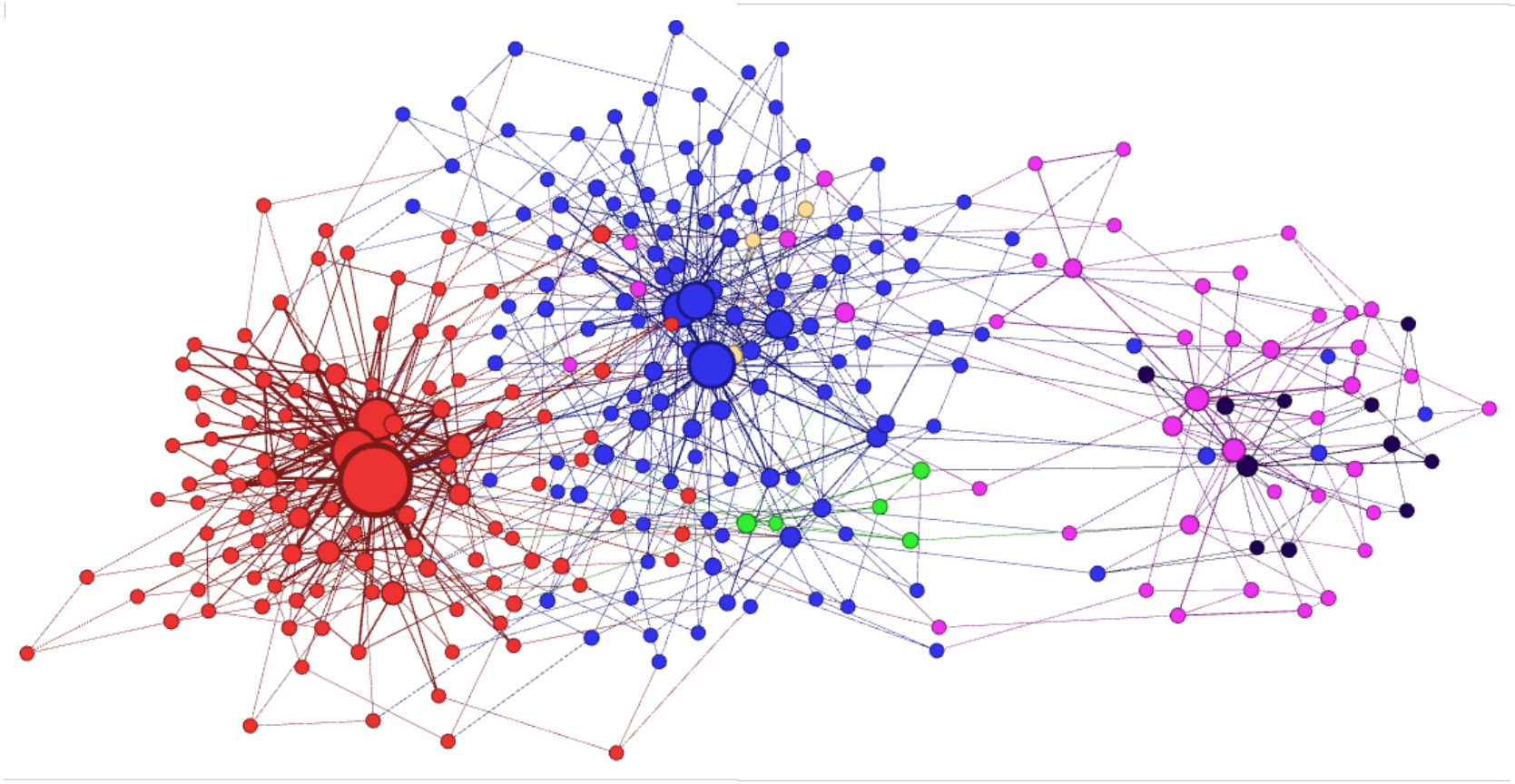
Community detection phase - step 1 initial

- From edge with the highest weight to the lowest weight,
  - If both nodes that connected by this edge has no community, assign new community.
  - If one node has community, assign this community to other node.
  - If both have communities, do nothing.
- Scale-free community is detected

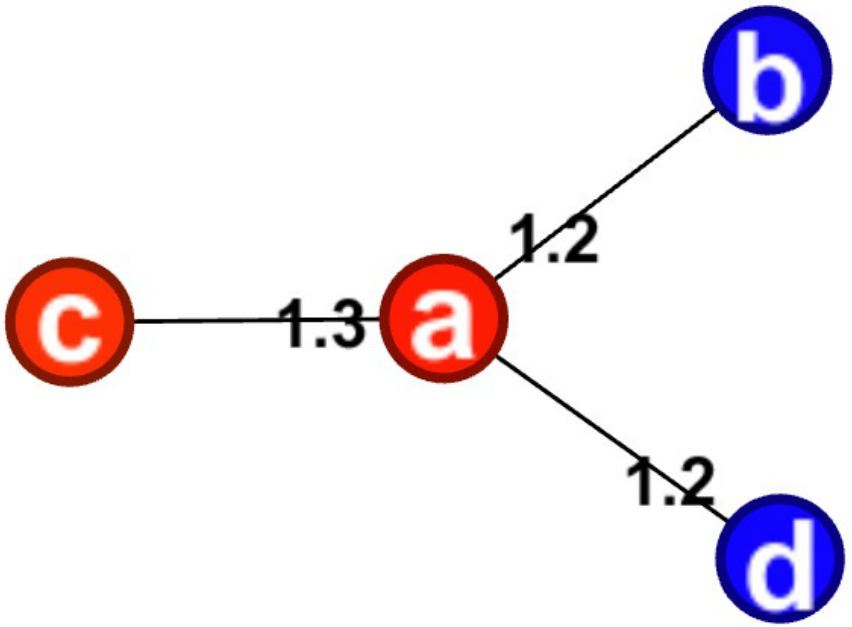


Source node	Target node	Weight
32	33	12731.0
0	1	9330.0
0	2	7164.0
0	3	7055.0
1	2	5917.0

Result- step 1 initial



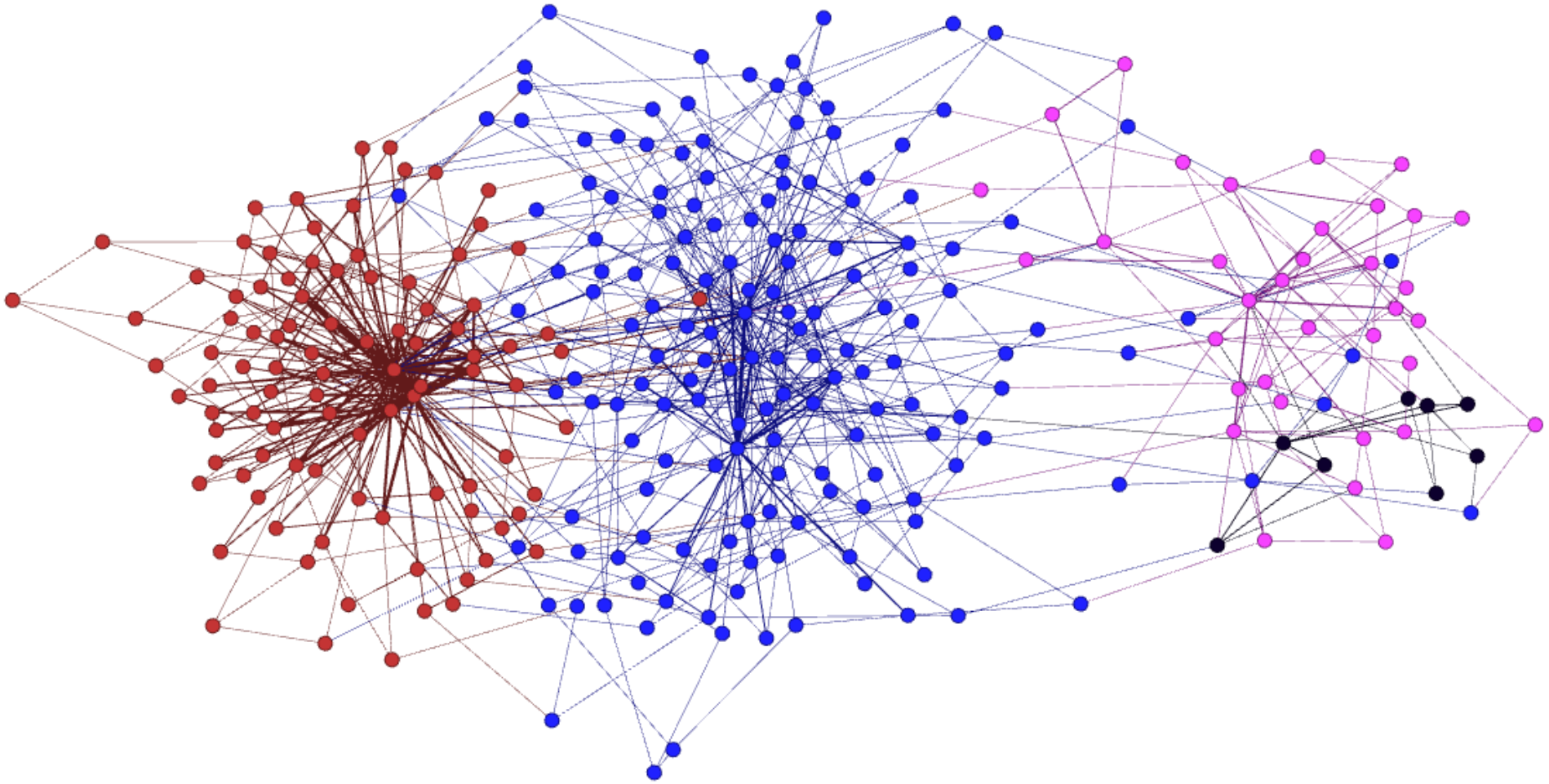
Community detection phase - step 1 con.



- This case can happen and this is wrong !
  - node "a" should be in blue community

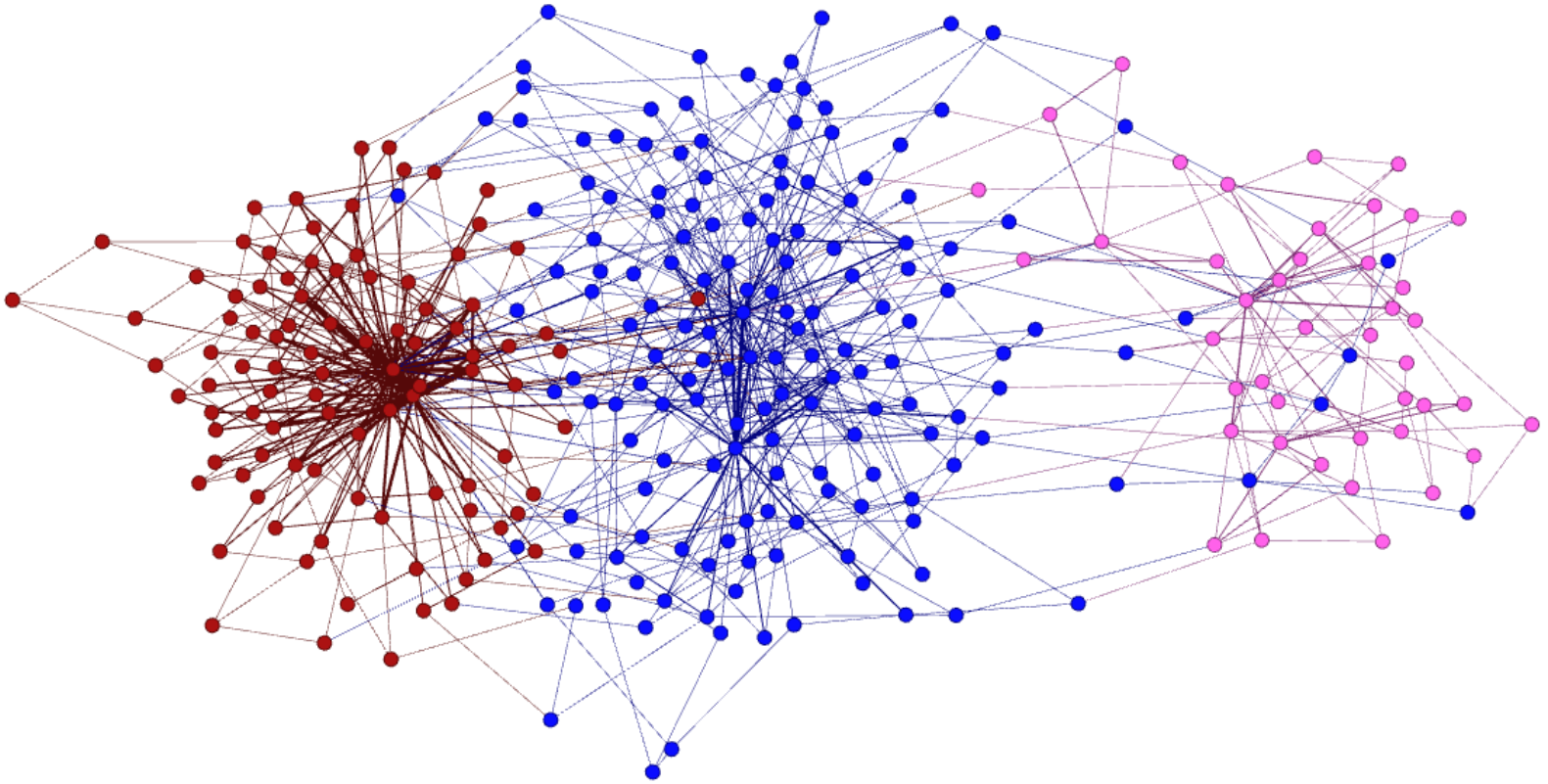
- Check all nodes at the edges of each community.

Result- step 1 converged



Community detection phase - step 2

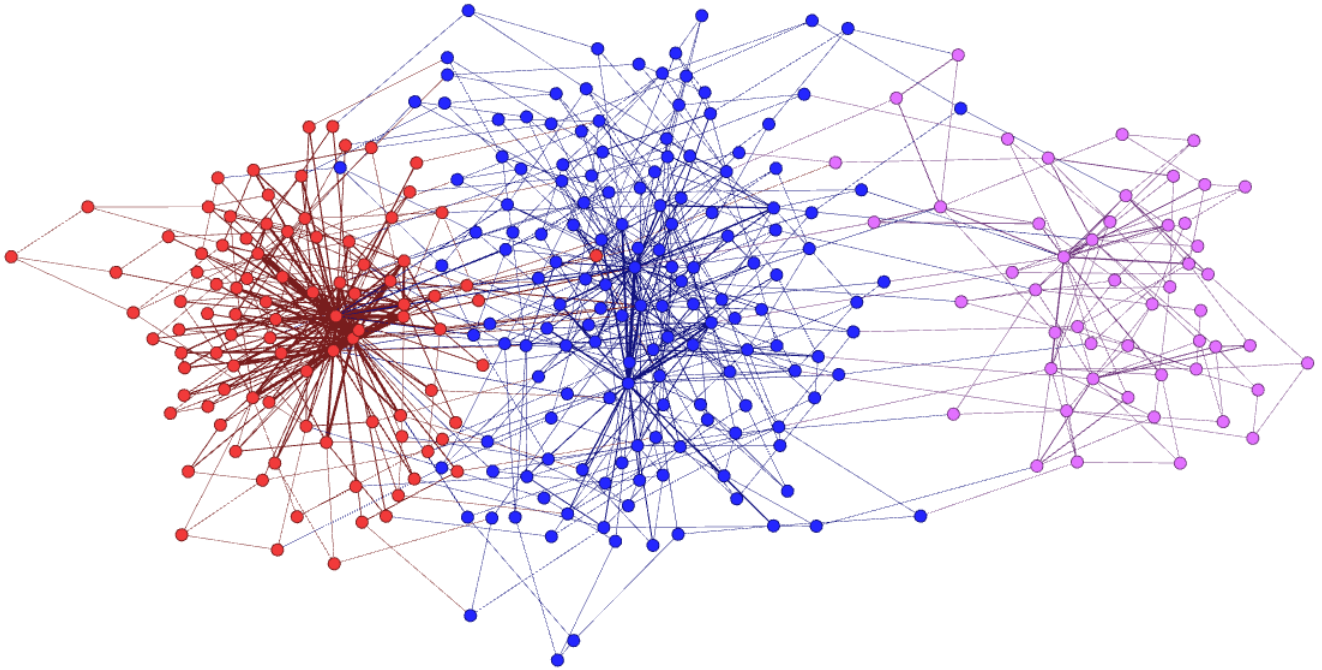
- Apply Lovain algorithm on our result, by using our weight.
- Communities that node degree follows normal distribution is detected.



- Black and pink community merge into one community due to modularity gain

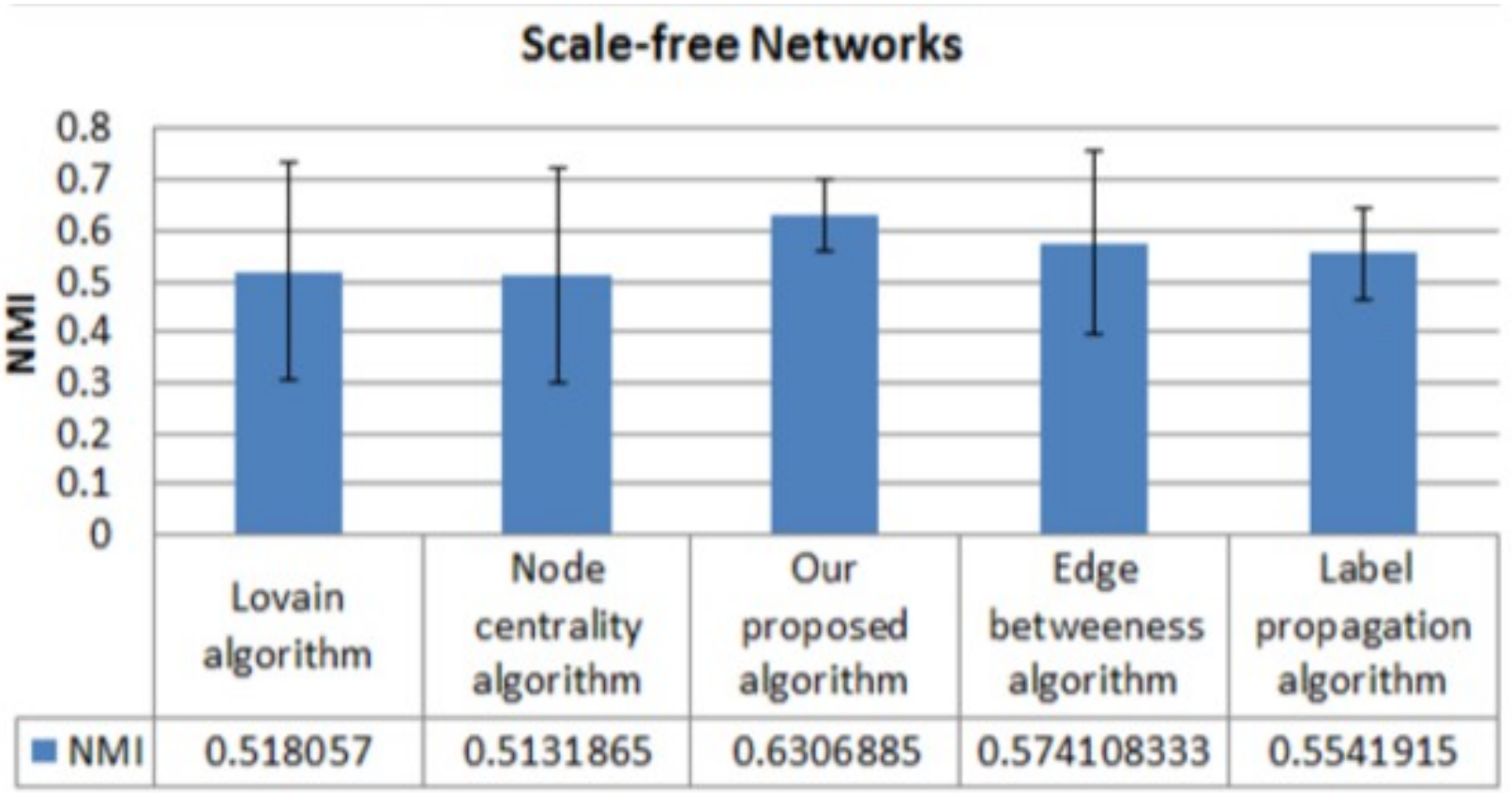
Step 1 again

- If there are community changes in step 2, it is possible that the result from step 2 can be changed to.



- Blue nodes that stay inside right most community are now reassigned to pink.
- With the same idea, if there are changes in step 1, step 2 need to be performed again too. The loop between step 1 and 2 is going until all nodes are balanced.

Result using NMI – synthetic networks



	Lovain algorithm	Node centrality algorithm	Our proposed algorithm	Edge betweeness algorithm	Label propagation algorithm
NMI	0.518057	0.5131865	0.6306885	0.574108333	0.5541915

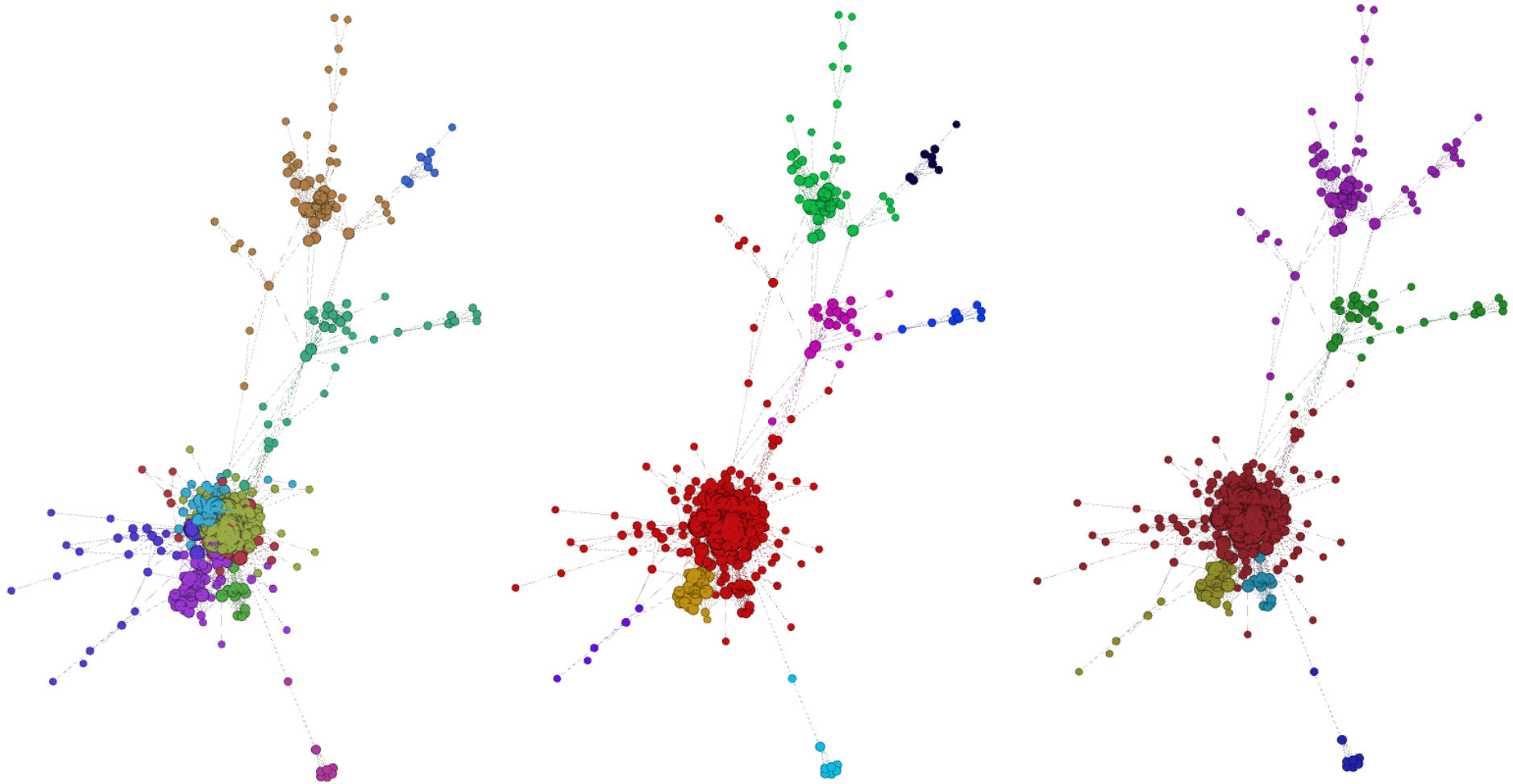
Result using NMI – datasets with ground truth

NetworkName	*Louvains*	*NodeCentrality*	*ProposedAlgorithm*	*EdgeBeweeness*	*LabelPropagation*
Zachary karate club	0.359	1.0	1.0	0.370	0.678
American College Football	0.765	0.765	0.765	0.723	0.725
Doubtfulsound dolphins	0.328	0.609	0.580	0.486	0.332
Political books	0.349	0.308	0.449	0.318	0.238

*Does not show significant improvement over previous approaches.

Example result on real-world dataset

- Facebook0 network from Ego Facebook Datasets



Louvains Algorithm | Node Centrality | Proposed Methods

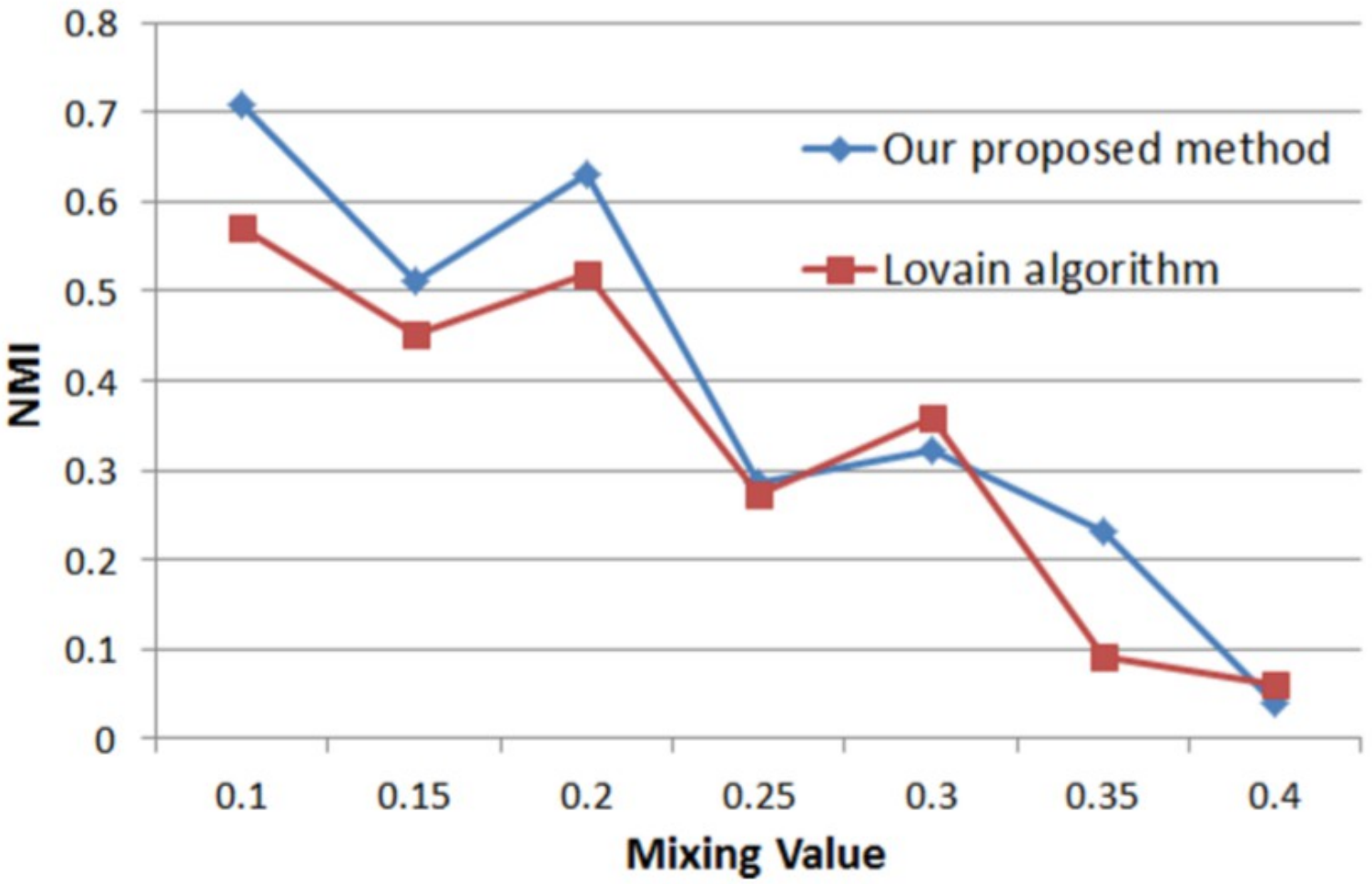
Computation time

Networks	*Nodes*	*Edges*	*Weighting*	*CommunityDetection*	*Time(seconds)*
GRQC	4158	13422	24	3	27
HepTh	6301	20777	26	1	27
Gnutella8	8638	24806	19	12	31
PGP	10,680	24,316	43	12	55
DBLP	317,080	1,049,866	3029	3161	6190

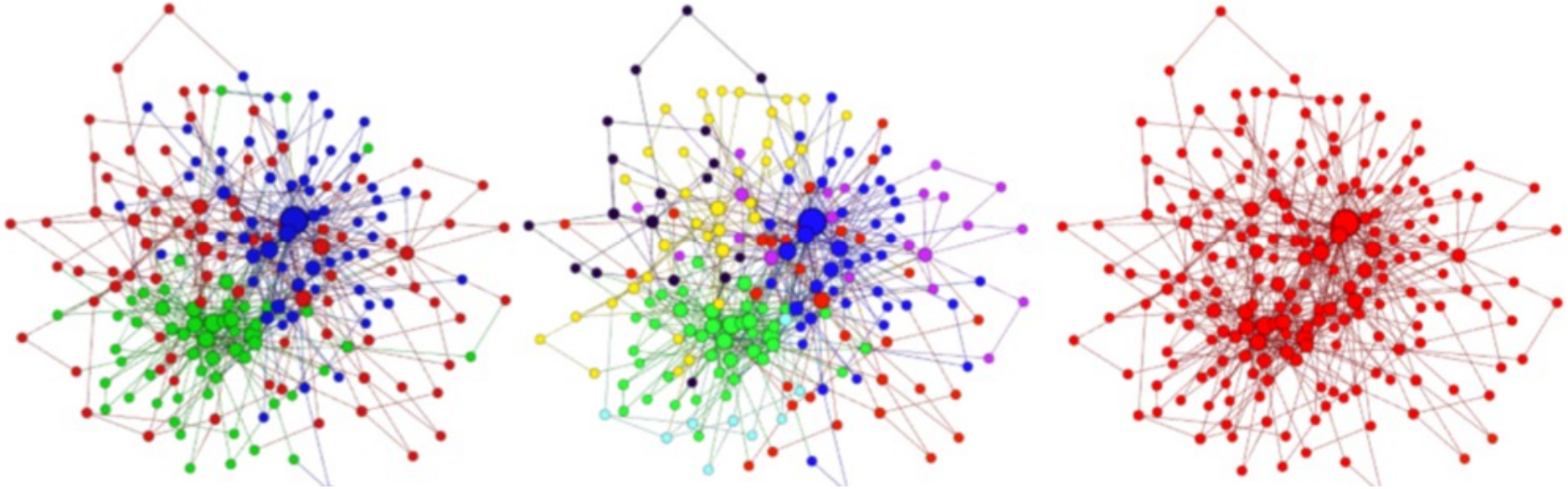
$$O(mk^2 + n \log n)$$

m is number of edges, *k* is average degree, *n* is number of nodes

Current limitation - noises



Noises – the worst case scenario



Ground Truth Louvain Algorithm Proposed Method

On rare instances, when edges that link 2 hub nodes receive high centrality score, whole graph can be merged into one community.

Q & A