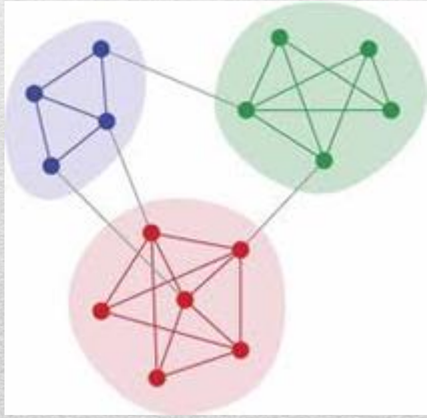
Movement of Vertices Across Communities in Dynamic Networks

Sriram Srinivasan
In Collaboration With
Sanjukta Bhowmick
University of Nebraska at Omaha
SIAM CSE2017
March 02 2017

Community Detection

- Communities are groups of vertices that are more tightly connected to each other than other vertices in the network

- Numerous methods/metrics exist
  - Modularity (more connections than random)
  - Conductance (less connections between in a communities)
  - Random Walk (vertices most often visited are in a community)

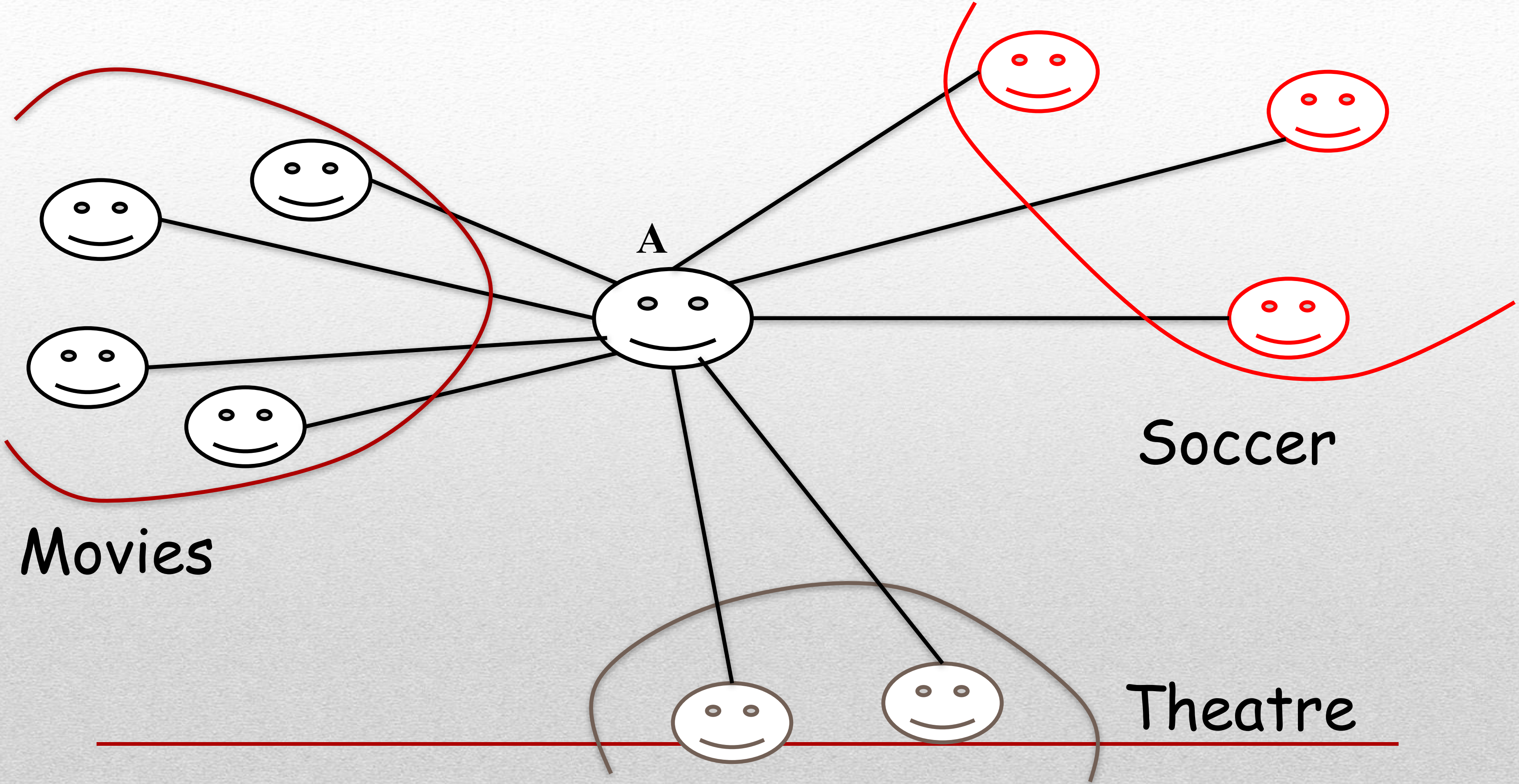


Community Detection in Dynamic Networks

- As the structure of the network changes, the communities can also alter
- Work has been done on updating communities without recomputing
  - Fast community detection for dynamic complex networks Bansal, Bhowmick, Paymal 2010.
  - Tracking local communities in streaming graphs with a dynamic algorithm Zakrzewska, Bader 2016
- **Our focus is on identifying a-priori the "fickle" vertices that are more likely to leave the community**

Challenges

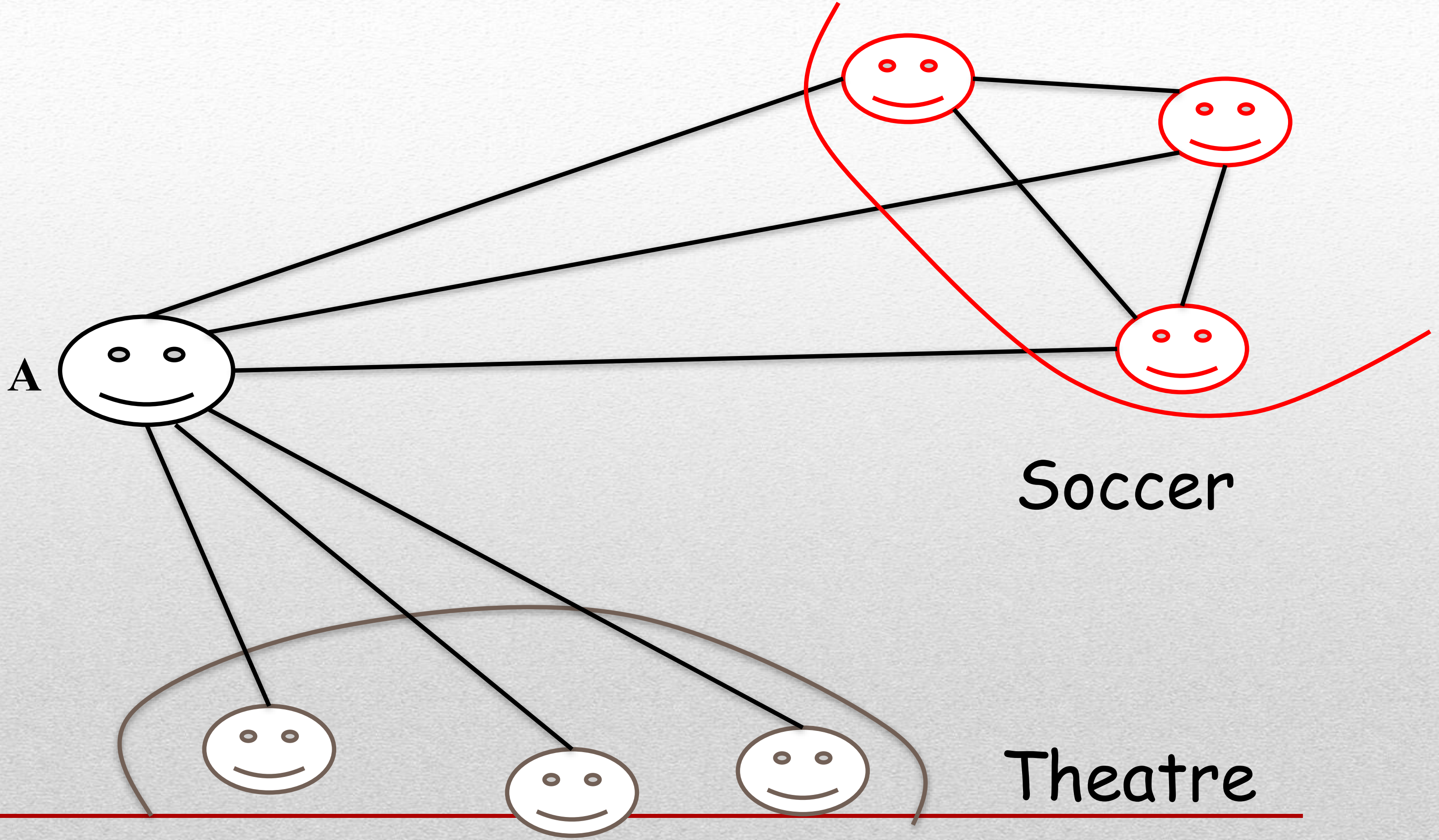
- Most community detection metrics are based on the entire network—not per vertex
- Due to resolution limit smaller communities are absorbed into larger ones
- The optimum value of metric depends on the network size, not quality of community
- **We require a metric that is vertex-based and sensitive to the changes in network structure**


A
Soccer
Movies
Theatre

Total Internal connections > maximum external connections to **any one** of the external communities

Modularity, Conductance

consider **total external connections**


A
Soccer
Theatre

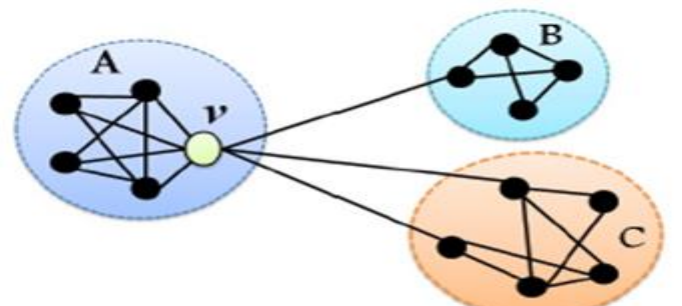
Internal neighbors should be **highly connected** =>
high clustering coefficient among **internal neighbors**

Modularity, Conductance do not consider **clustering coefficient**

Permanence

$$Perm(v) = \left[\frac{I(v)}{E_{max}(v)} \times \frac{1}{D(v)}\right] - (1 - C_{in}(v))$$

$I(v)$=internal deg of v
$D(v)$=degree of v
$E_{max}(v)$=Max connection to an external community
$C_{in}(v)$=Clustering coefficient of internal neighbors



$Perm(v)=0.12$

$I(v)=4$, $D(v)=7$,
$E_{max}(v)=2$, $C_{in}(v)=5/6$

Permanence of entire network= $\frac{1}{N} \sum_{v=1:N} Perm(v)$

Find community by maximizing permanence of the network

T. Chakraborty, S. Srinivasan, N. Ganguly, A. Mukherjee, S. Bhowmick, *On the permanence of vertices in network communities*. KDD 2014

Properties of Permanence

- Vertex-based.
  - Computes the “belongingness” of a vertex in a community
- Uniform Scale.
  - Ranges from -1 (vertex placed in completely wrong community) to 1 (vertex in a clique).
- Relatively independent of network size
- Handles Resolution Limit
- Is Sensitive to Changes

Test Suites with Ground Truth

- Real World Networks
  - Network of Inter-college Football
  - Network of Indian Railways
  - Network of Co-authorship in Technical Articles

- Synthetic Networks
  - LFR networks using different values of mixing parameter (μ)
  - Lower value of μ indicates tighter community structure

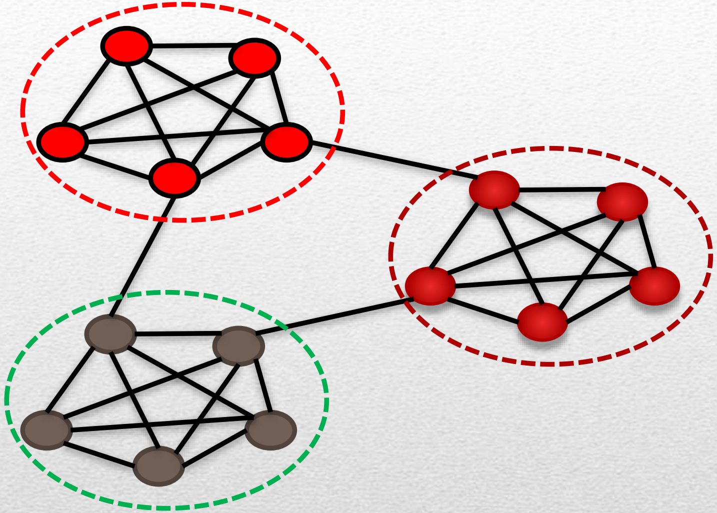
Experimental Results

Method	LFR (μ=0.1)	LFR (μ=0.3)	LFR (μ=0.6)	Football	Railway	Coauthors
Louvain	0.02	0.00	-0.75	0.02	0.14	0.00
FastGrdy	0.00	0.87	0.02	0.01	0.37	0.14
CNM	0.14	0.40	-0.13	0.30	0.20	0.05
WalkTrap	0.00	0.00	-0.50	0.02	0.03	0.03
Infomod	0.06	0.08	-0.20	0.01	0.19	-0.04
Infomap	0.00	0.00	-0.72	0.00	0.02	-0.02

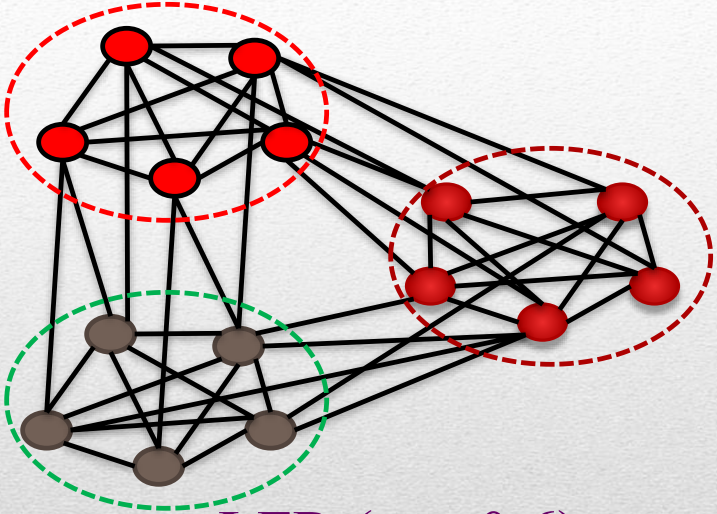
Differences of our algorithm with the other algorithms averaged over all 6 different validation measures

(High Value means Max Perm was more accurate)

LFR (μ = 0.1) vs. LFR (μ = 0.6)



LFR (μ = 0.1)



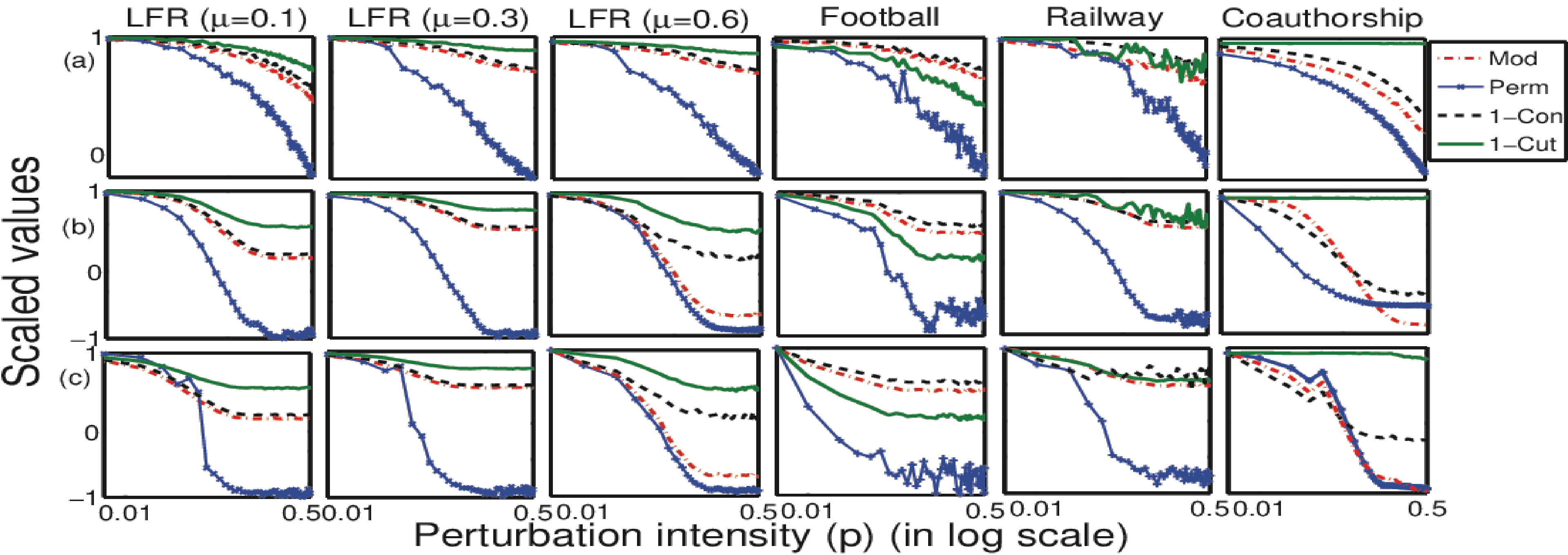
LFR (μ = 0.6)

For networks with bad community structure ground truth may be biased. Permanence can capture this

Co-Authorship Results

- By inspecting the meta-data [keywords; subgroups] we find that permanence detects the sub-communities
- Main Communities as per Ground Truth
  - Algorithms and Theory;
  - Databases
- Communities obtained by maximizing permanence have these groups
  - Theory of Computation, Formal Methods, Information and Coding Theory, Computational Geometry, Data Structure
  - Models, Query Optimization, Database Languages, Storage,
- Permanence can detect smaller community, overcoming resolution limit.

Sensitivity Under Perturbation



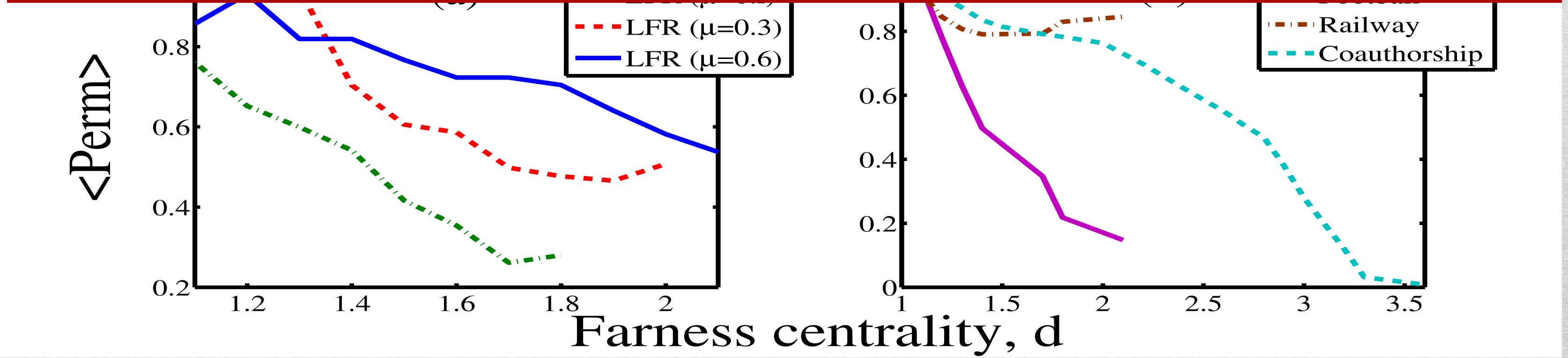
Each row represents a different methods of swapping vertices across two communities

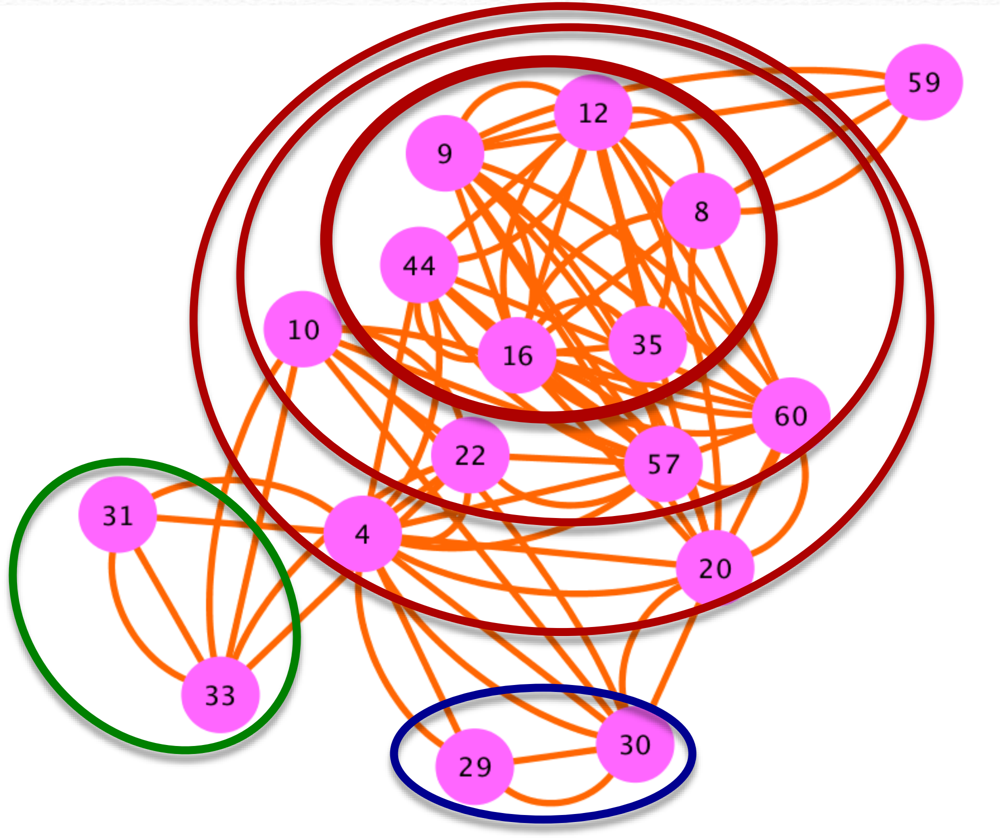
Permanence is robust to small changes and sensitive to large changes

Core-Periphery and Permanence

- Permanence can be used to identify whether the vertex is at the core or the periphery of its community
- Farness Centrality: Mean shortest path of a vertex to all other vertices in its community
- **Higher permanence=> Low Farness Centrality=> Closer to the core**

Vertices with lower permanence are easier to remove from communities





Can move outer vertices to other communities with just one edge addition
Adding edge 4-28 moves it to blue community

Can move external vertices to communities with just one edge addition
Adding edge 59-44 moves it to red community

Random insertion of edges in the network does not change membership of red, blue, green communities

Conclusions

- We introduce permanence—a metric that measures by how much a vertex belongs to a community
- Permanence is sensitive to changes in the community structure
- Permanence maps to the core-periphery structure of the communities
  - higher permanence vertices are at the core, lower permanence at the periphery
- Due to this phenomena, it should be easier to move lower permanence (periphery) vertices to other communities
- We have seen in our initial experiments, that this hypothesis is true for small networks with well defined communities'—dolphin, karate
  - Targeted change leads to correct movement of vertices to desired communities
  - Random changes do not change the communities significantly

- Next steps: to test on larger networks, and more fuzzy communities.