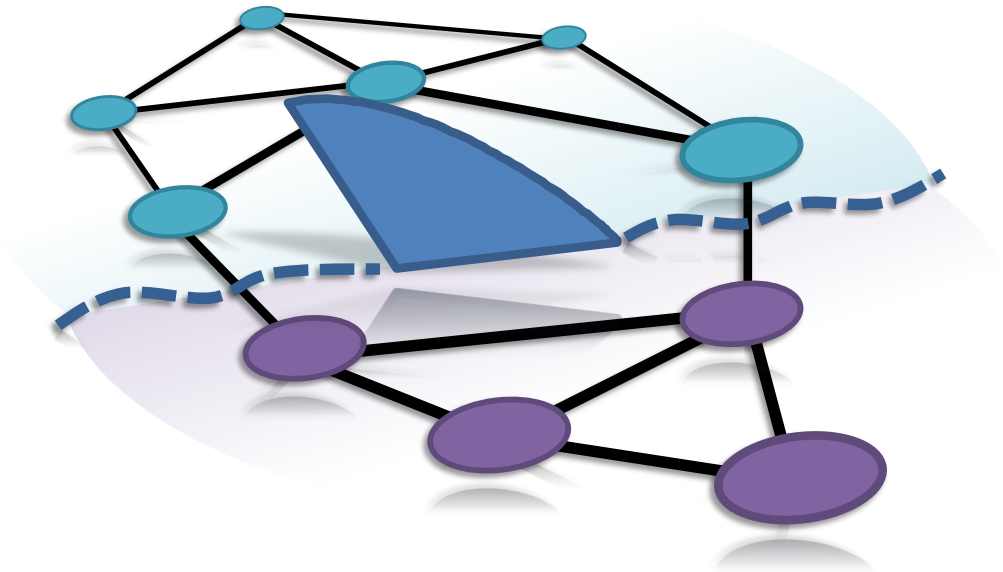
SHARC: Distributed Community Detection using a Neighborhood Similarity Measure



Guillaume-Jean Herbiet
Office E-006

UNIVERSITÉ DU LUXEMBOURG

TeamBouvry meeting
October 29th, 2009

Who am I?

Guillaume-Jean Herbiet

- Ph.D student here since 1 year + few months
- MSCS Georgia Institute of Technology (2007)
- Diplôme d'Ingénieur Supélec (2007)

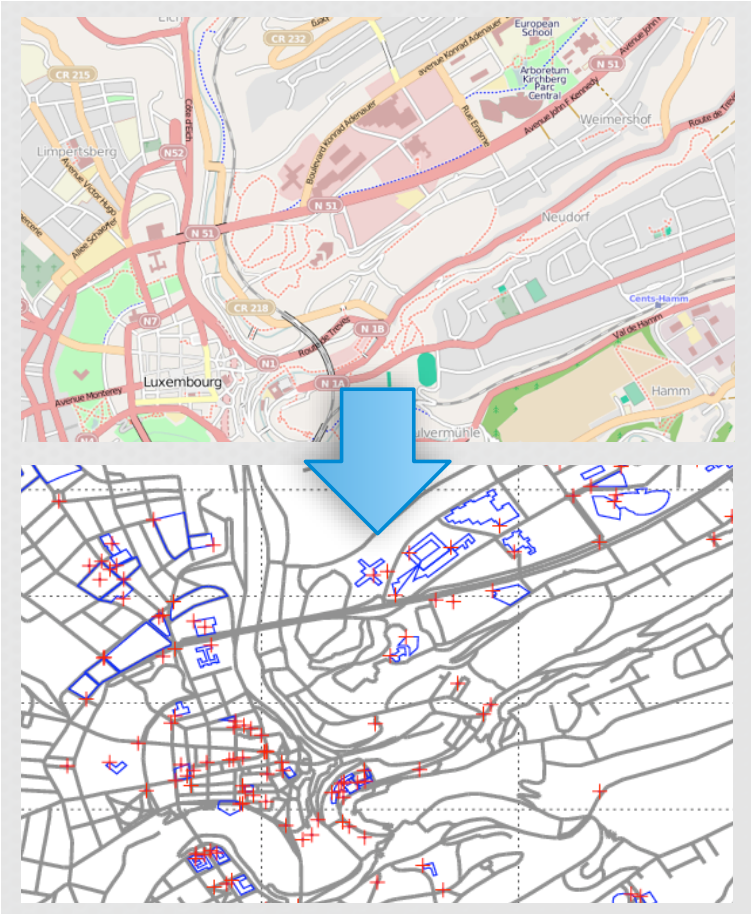
Research interests

- Ad hoc networks and user-oriented applications
- Realistic mobility and interaction between users (UrbiSim)
- Networking in general...

All this and more on `http://herbiet.gforge.uni.lu`

A word on UrbiSim

- A generic framework for ad hoc simulation in urban environments
  - Mobility of users between *spots* of the city section
  - Choice of destination based on social membership
- Based on OpenStreeMap:
  - Import streets with specificities (max speed, unidirectional, etc.)
  - Import *spots* from POI with type and shape
- Extensions: multimodal transportations, propagation model

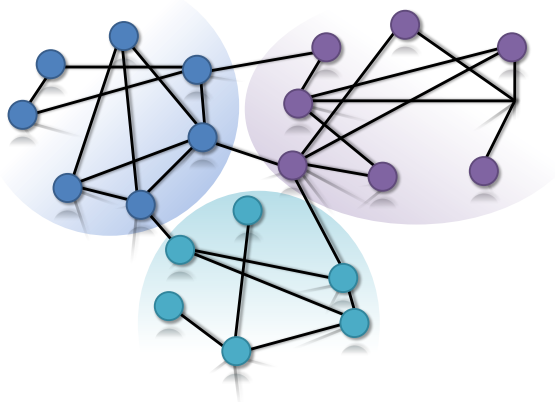


Presentation Outline

1. Communities and detection methods
2. Sharper detection with neighborhood similarity
3. Results
4. Conclusion and future work

Communities in networks

Groups of vertices with a high internal connectivity, and much sparser links to the outside of the group.[Girvan and Newman, 2002]



Encountered in many natural and social networks :

- Neural networks, species association
- Friendship, collaboration networks

Also present in wireless communication networks:

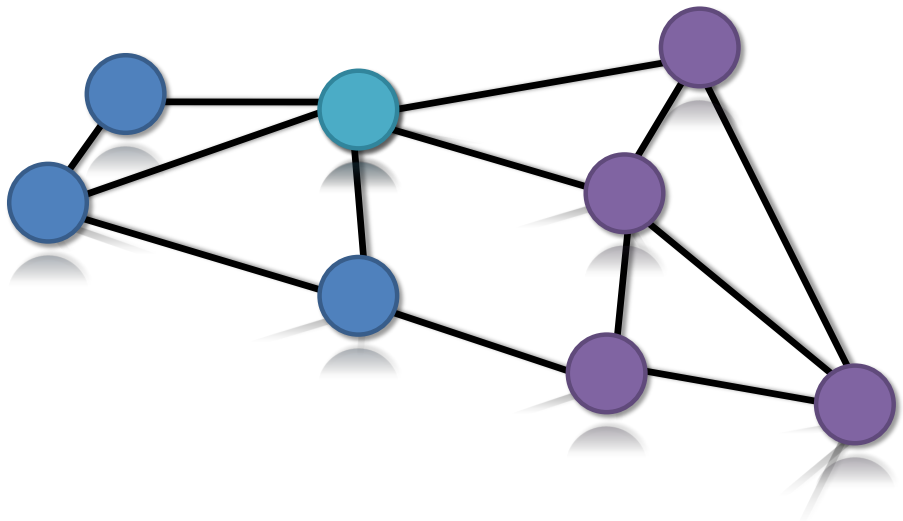
- Heterogenous mobility, social interactions of users
- Help scalability of other algorithms with 2-tier network
- Establish interactive/delay-tolerant communications

Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations

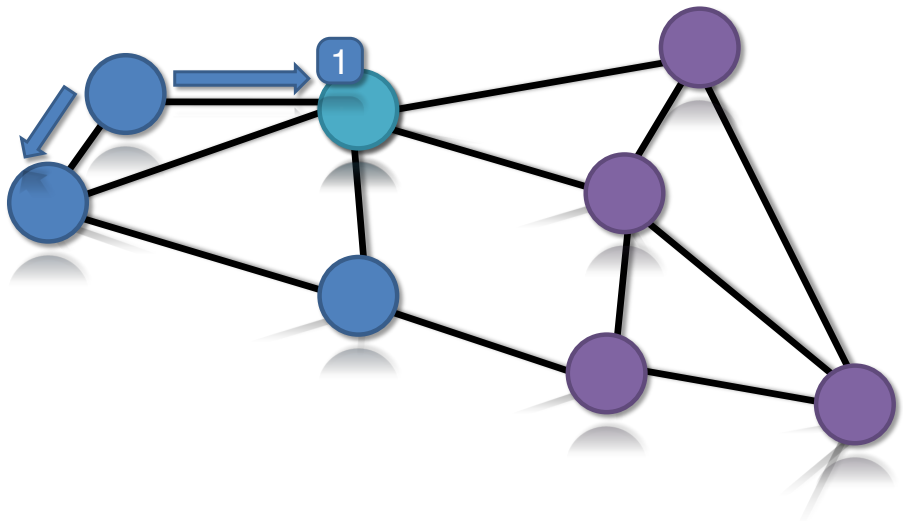
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



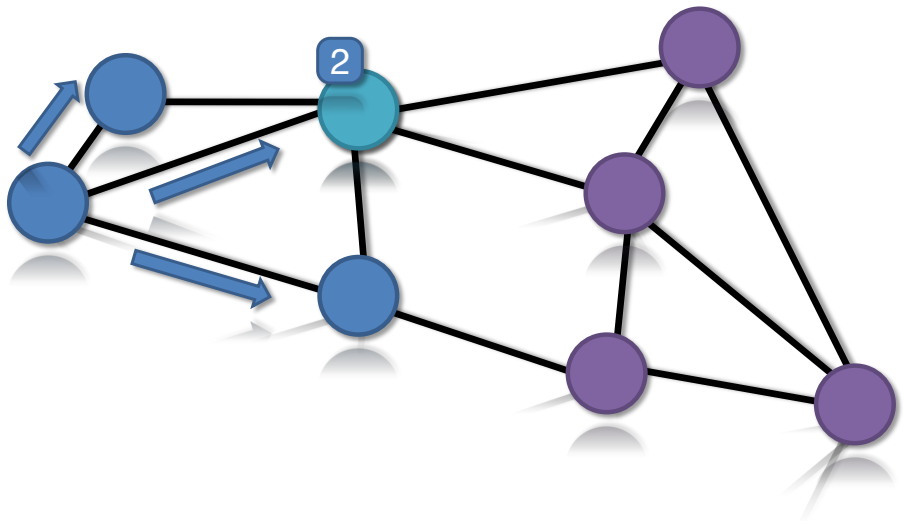
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



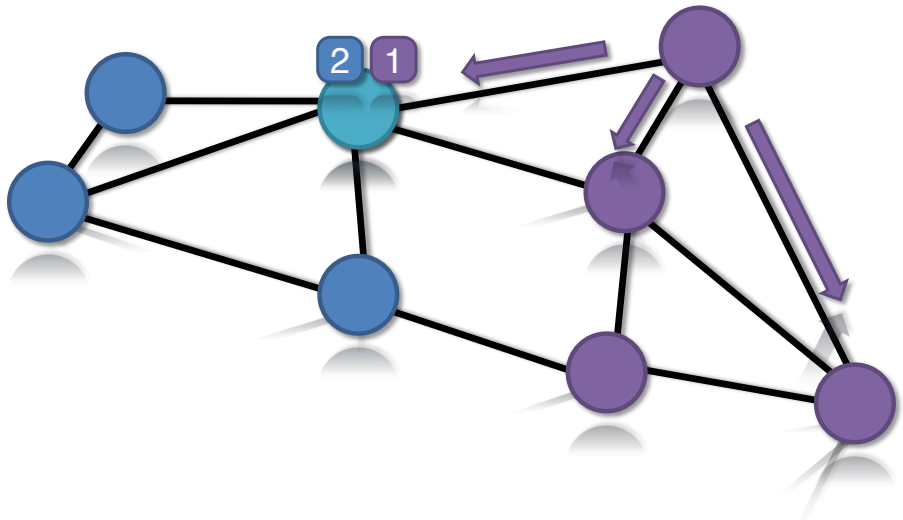
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



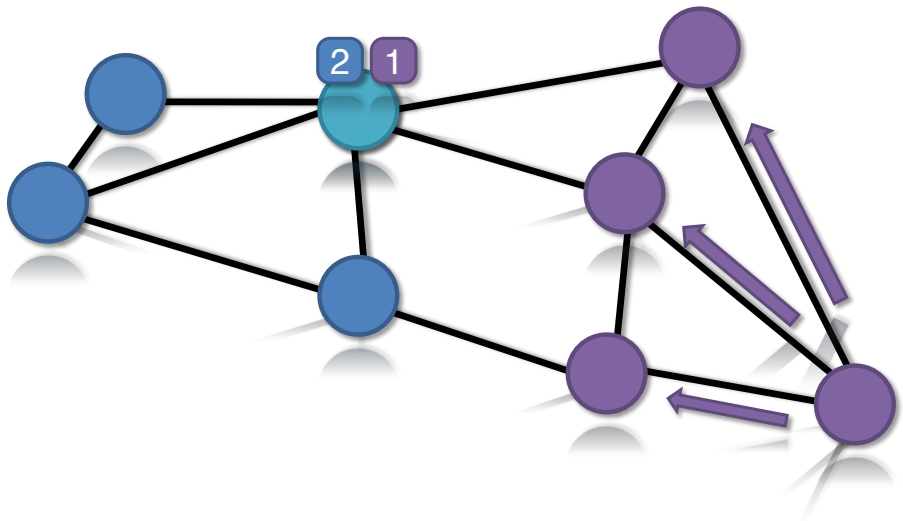
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



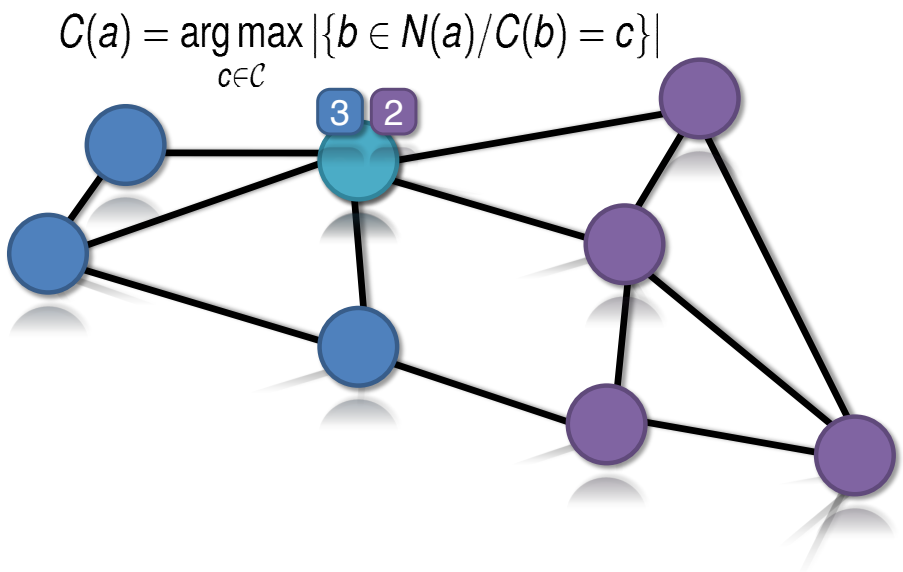
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



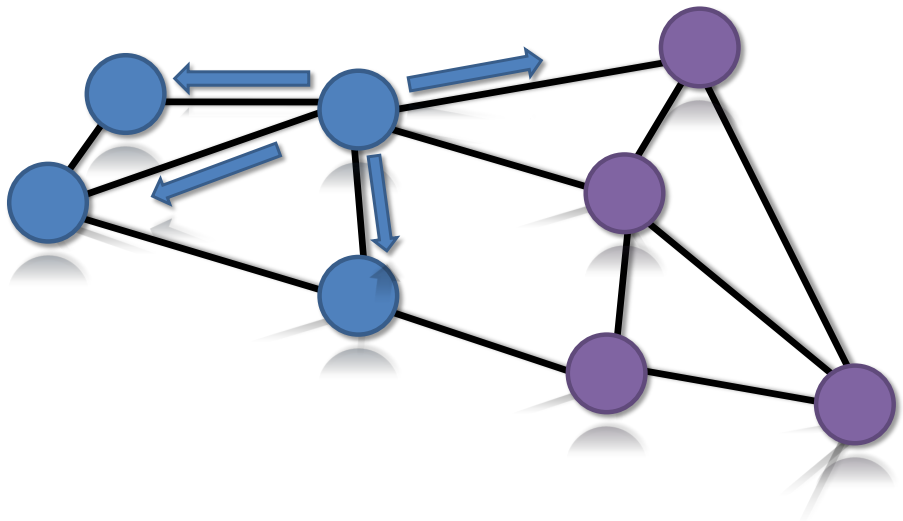
Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations



Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations
  - accuracy, convergence and dominant community problems



Community detection and assessment methods

- Centralized algorithms[Donetti and Muñoz, 2004][Newman, 2004]
- Greedy algorithms using local information
- Few decentralized algorithms[Raghavan *et al.*, 2007][Leung *et al.*, 2009]:
  - all based on epidemic label propagation with variations
  - accuracy, convergence and dominant community problems

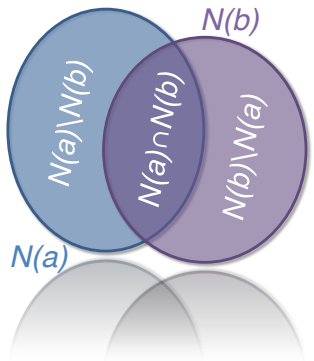
Assessment methods:

- Comparison with pre-existing *"natural"* assignments (Jaccard index, variation of information, misplaced nodes)
- Modularity Q: reference used for benchmarks[Newman and Girvan, 2004]
  - contained in $[-1; 1]$ with > 0 if better than random
  - absolute maximum depends on the network

Neighborhood similarity measure

For every vertices a of a network $\mathcal{N}$, and $b \in N(a)$ we define:

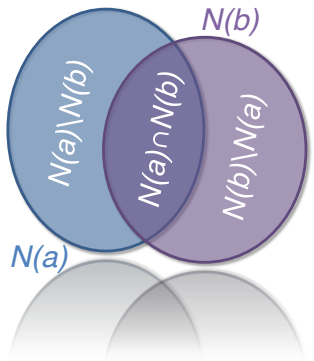
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$



Neighborhood similarity measure

For every vertices a of a network $\mathcal{N}$, and $b \in N(a)$ we define:

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$



- Directly implied by the definition of communities:
  - neighbor nodes of same community are likely to have many common neighbors due to high link density
- Help sharpen community detection:
  - Normalized measure
  - Does not favor high degree nodes
  - Can be interpreted as the *strength* of community membership

Using neighborhood similarity for detection

- Sharper Heuristic for Assignment of Robust Communities:

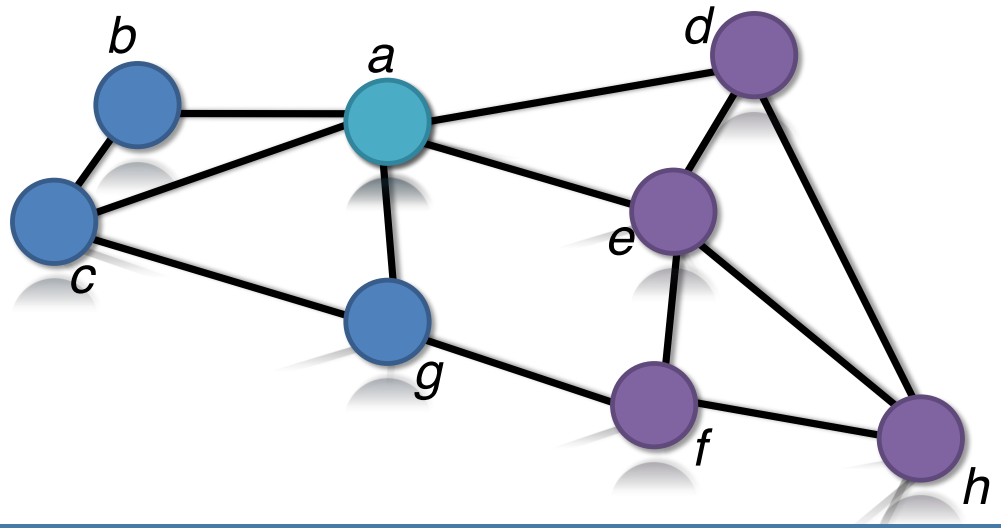
$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a, b)$$

Using neighborhood similarity for detection

- Sharper Heuristic for Assignment of Robust Communities:

$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a, b)$$

- Requires access to community id and neighborhood set

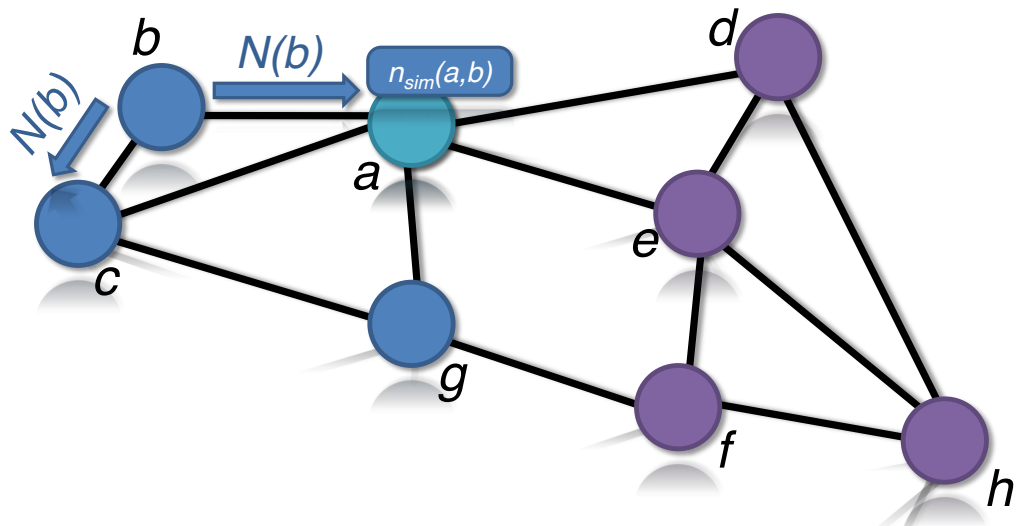


Using neighborhood similarity for detection

- Sharper Heuristic for Assignment of Robust Communities:

$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a,b)$$

- Requires access to community id and neighborhood set

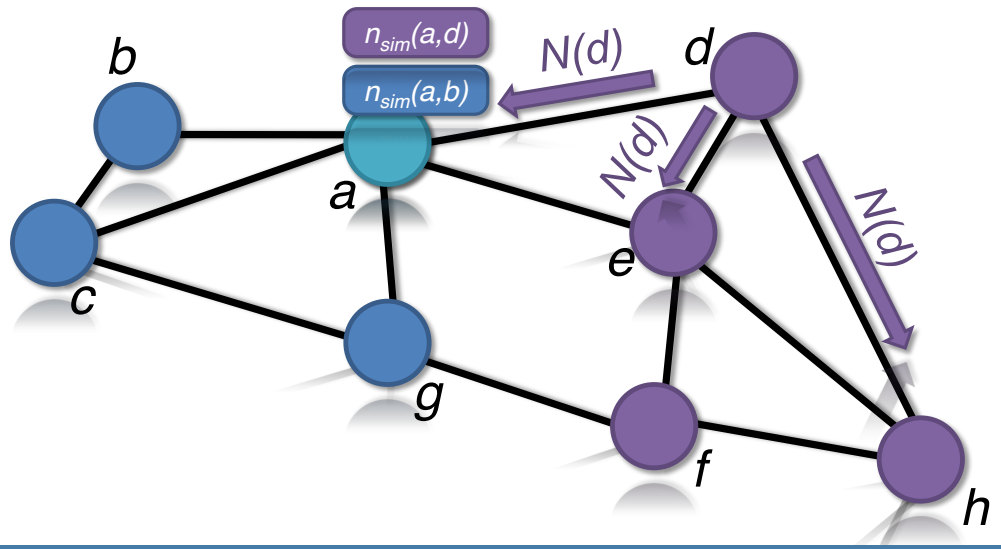


Using neighborhood similarity for detection

- Sharper Heuristic for Assignment of Robust Communities:

$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a,b)$$

- Requires access to community id and neighborhood set

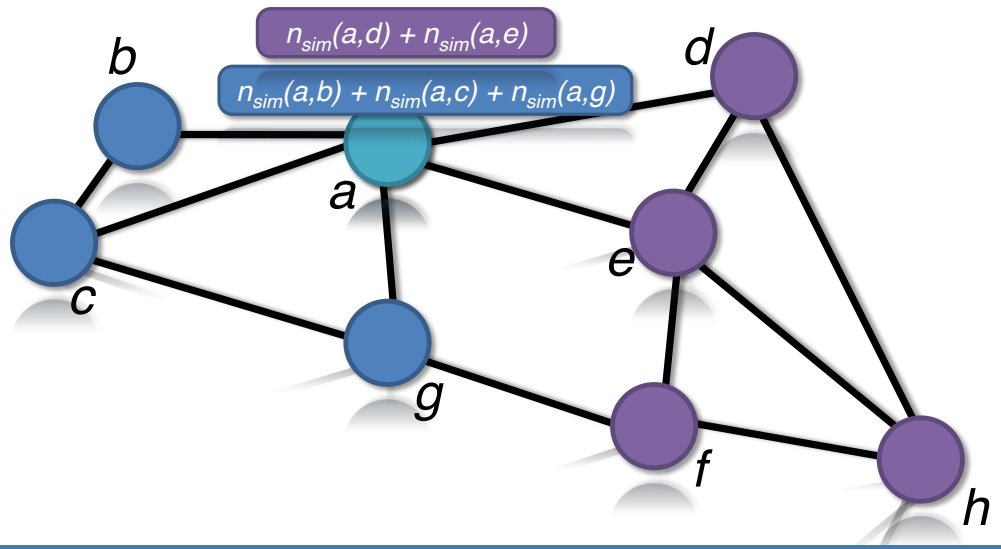


Using neighborhood similarity for detection

- Sharper Heuristic for Assignment of Robust Communities:

$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a,b)$$

- Requires access to community id and neighborhood set

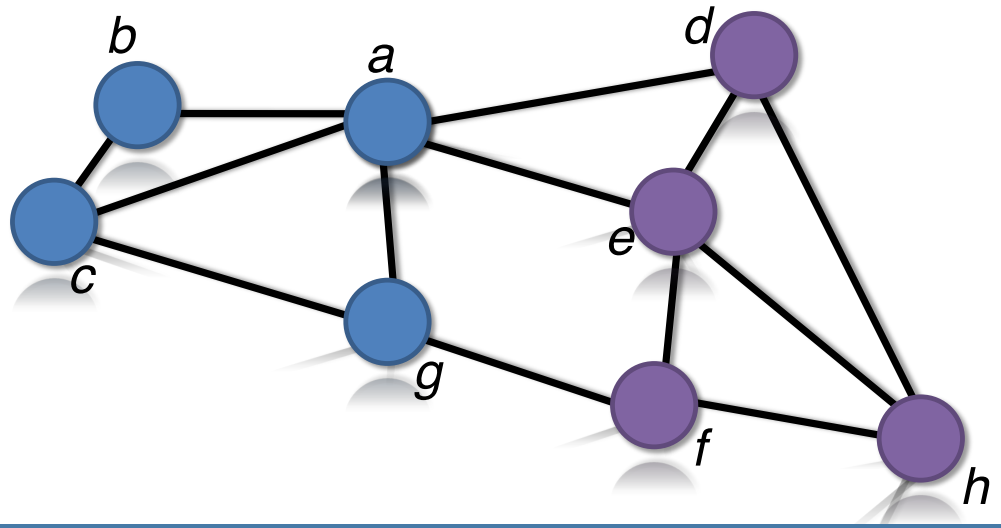


Using neighborhood similarity for detection

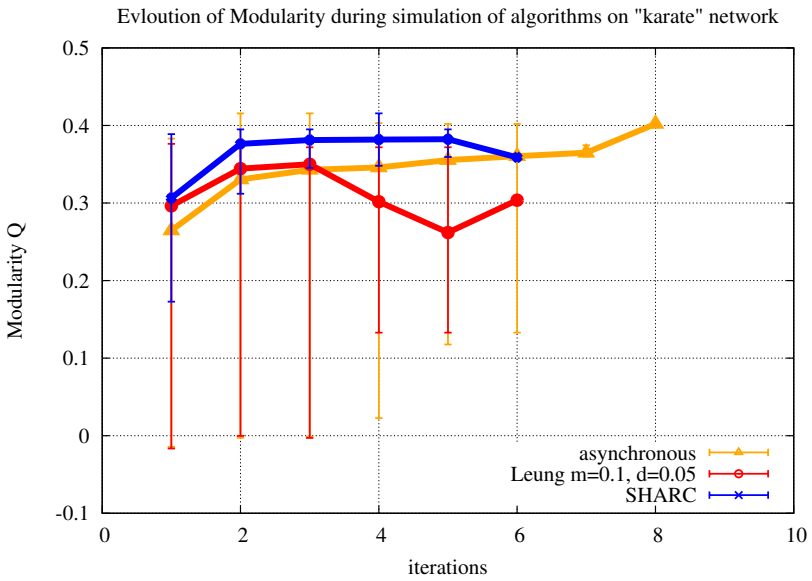
- Sharper Heuristic for Assignment of Robust Communities:

$$C(a) = \arg\max_{c \in \mathcal{C}} \sum_{b \in N(a)/C(b)=c} n_{sim}(a, b)$$

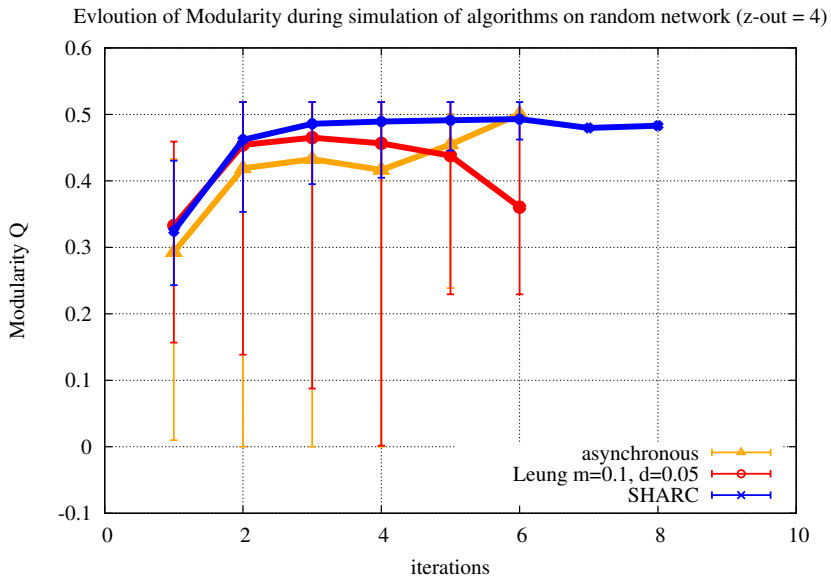
- Requires access to community id and neighborhood set



Performance assessment on static networks

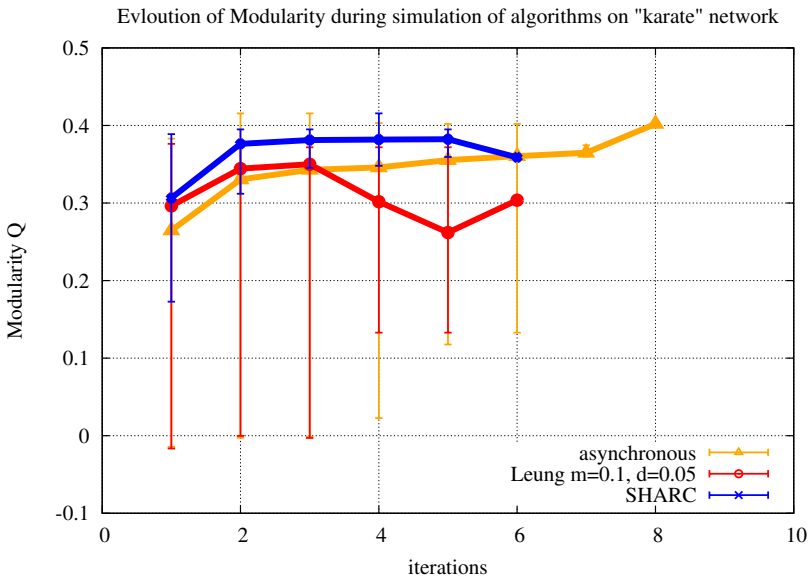


Results over 400 runs

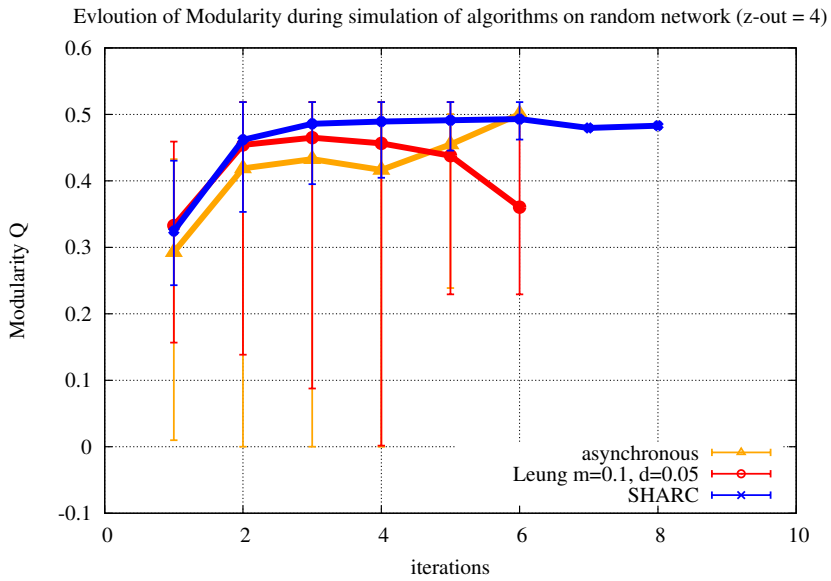


Results over 20 runs on 20 different networks

Performance assessment on static networks



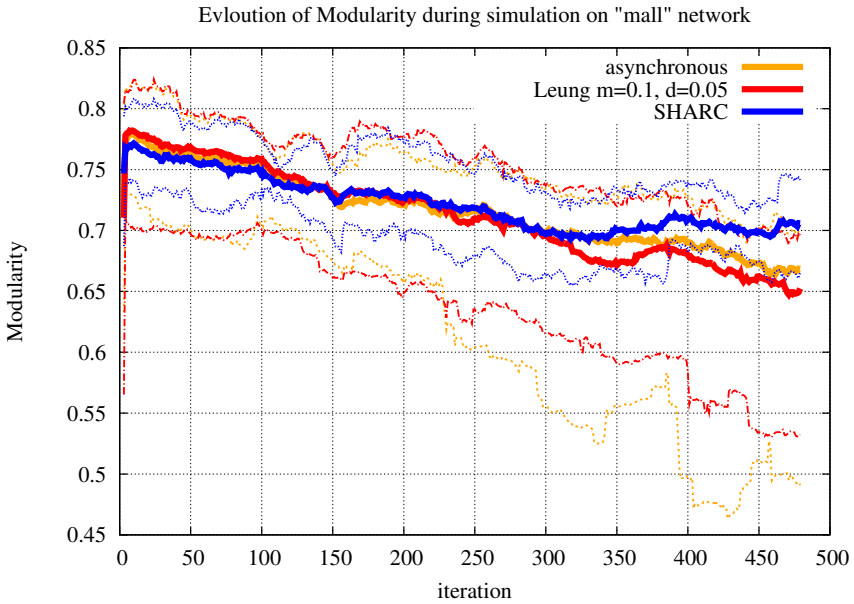
Results over 400 runs

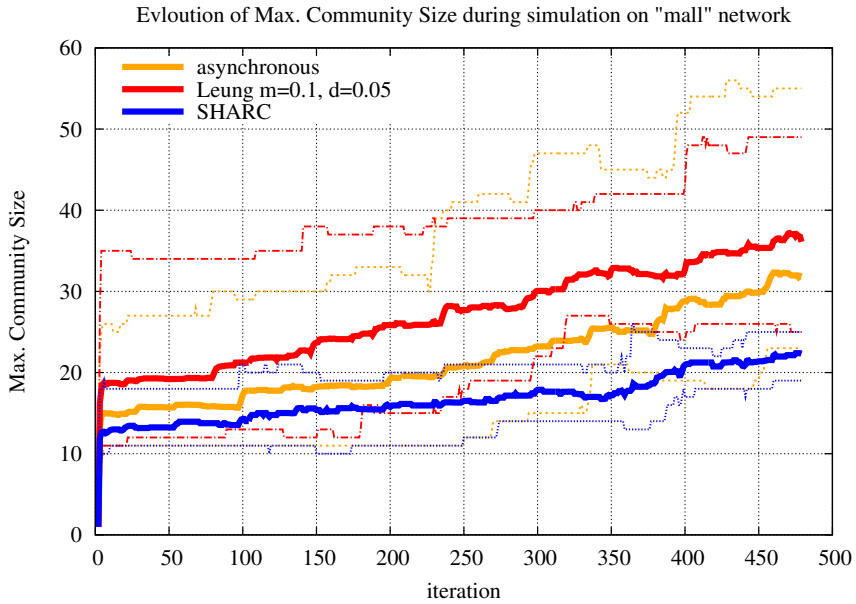


Results over 20 runs on 20 different networks

- SHARC reaches high values of modularity the fastest
- SHARC highest result is the best or close to best
- SHARC presents smaller std. dev.
  - resiliency to initialization parameters
  - resiliency to network topology changes

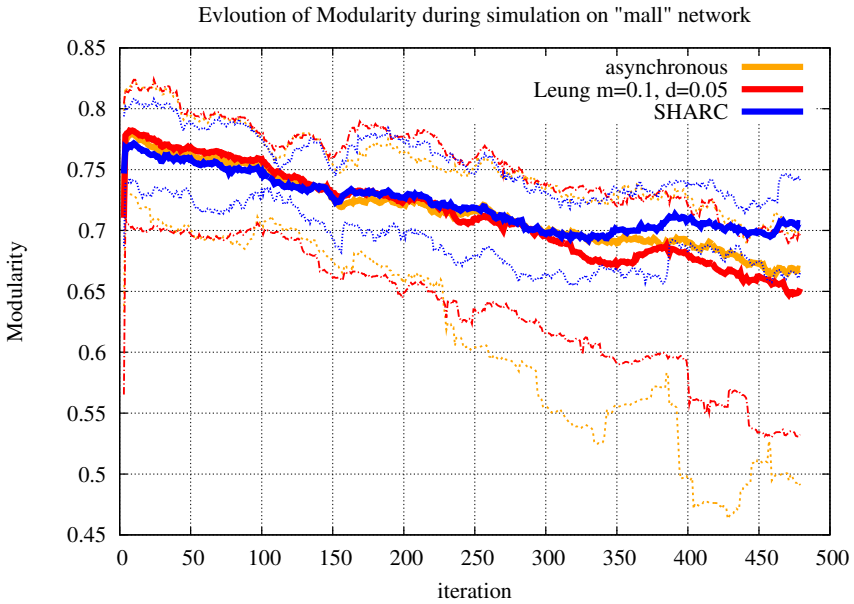
Performance assessment on dynamic networks

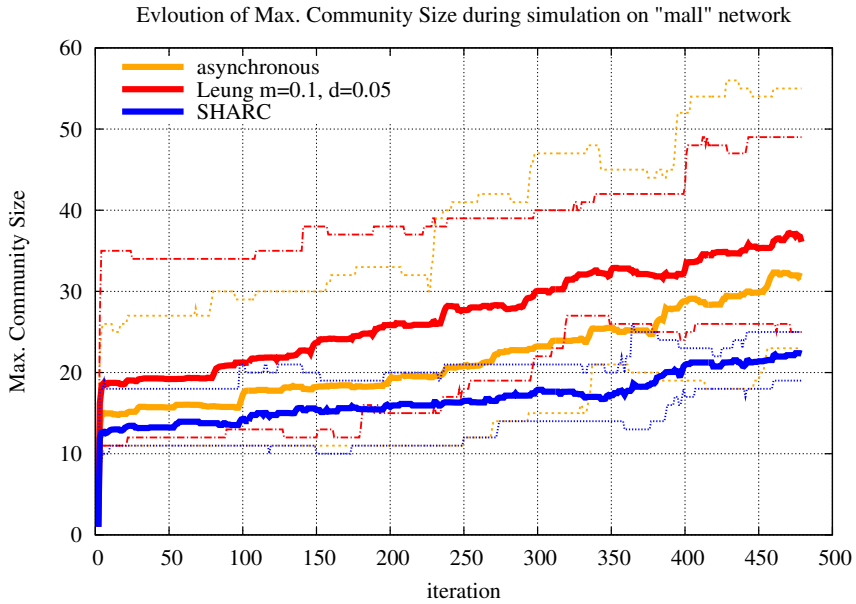




Results over 100 runs on 4 different networks

Performance assessment on dynamic networks





Results over 100 runs on 4 different networks

- SHARC achieves good avg. results, with smaller std. dev.
- SHARC limits the *jumbo* community effect with reasonable community size, even at end of simulation
- *Wandering* community effect still impairs results

Conclusion and future work

Outcome:

- Proven the validity of our approach (n_{sim}, SHARC)
- SHARC performs sharper community assignment, faster, and is more resilient to configuration changes

Conclusion and future work

Outcome:

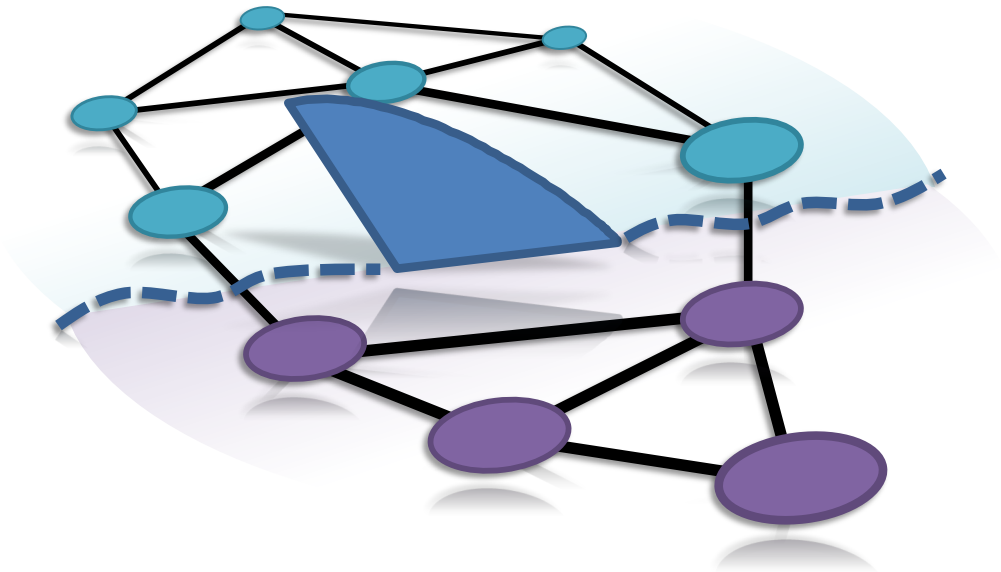
- Proven the validity of our approach (n_{sim}, SHARC)
- SHARC performs sharper community assignment, faster, and is more resilient to configuration changes

Extensions:

- Weighted networks to search for stable/reliable communities:
  - Use link quality as edge weights
- Study under more realistic simulation environments (UrbiSim)
- Use SHARC as building stone for other protocols
  - NPO problem, dual interactive/DTN communications

Thanks for your attention.

Any questions?



References (1)

Luca Donetti and Miguel A. Muñoz.

Detecting network communities: a new systematic and efficient algorithm.
Journal of Statistical Mechanics: Theory and Experiment, 2004(10), 2004.

M. Girvan and M. E. J. Newman.

Community structure in social and biological networks.
Proceedings of the National Academy of Sciences of the United States of America, 99(12):7821–7826, June 2002.

P Hui, A Chaintreau, J Scott, R Gass, J Crowcroft, and C Diot.

Pocket switched networks and human mobility in conference environments.
Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, pages 244–251, 2005.

Ian X. Y. Leung, Pan Hui, Pietro Liò, and Jon Crowcroft.

Towards real-time community detection in large networks.
Physical Review E (Statistical, Nonlinear, and Soft Matter Physics), 79(6):066107+, 2009.

D. Lusseau.

The emergent properties of a dolphin social network.
Proc. R. Soc. London B (suppl.), (270):S186–S188, 2003.

M. E. Newman and M. Girvan.

Finding and evaluating community structure in networks.
Physical Review E, 69(2):026113+, Feb 2004.

References (2)

M. E. J. Newman.

Fast algorithm for detecting community structure in networks.
Physical Review E, 69(6):066133+, Jun 2004.

M. E. J. Newman.

Finding community structure in networks using the eigenvectors of matrices.
physics.data-an, Jan 2006.

Usha N. Raghavan, Réka Albert, and Soundar Kumara.

Near linear time algorithm to detect community structures in large-scale networks.
Physical Review E (Statistical, Nonlinear, and Soft Matter Physics), 76(3), 2007.

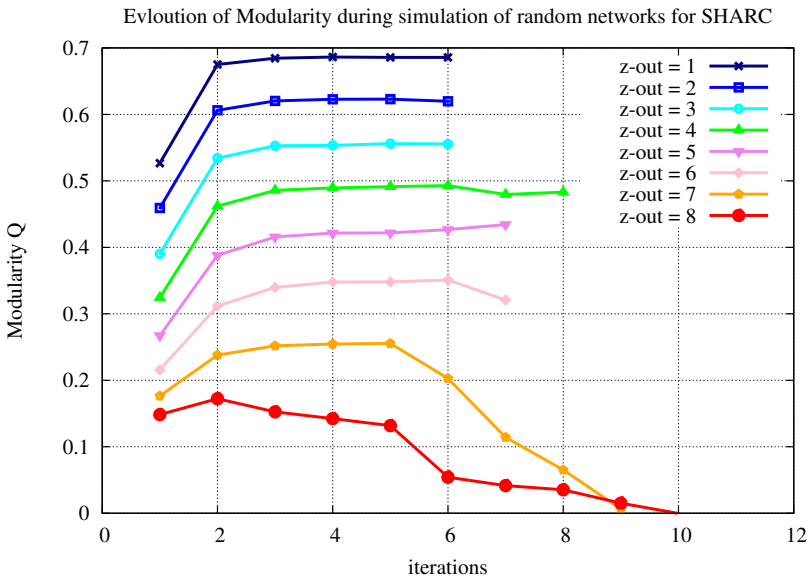
D. J. Watts and S. H. Strogatz.

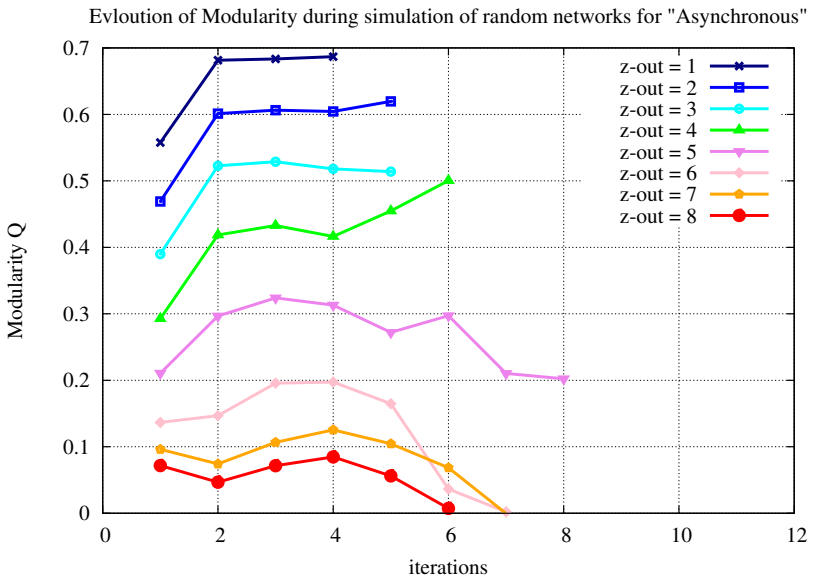
Collective dynamics of small-world networks.
Nature, (393):440–442, 1998.

W. W. Zachary.

An information flow model for conflict and fission in small groups.
Journal of Anthropological Research, (33):452–473, 1977.

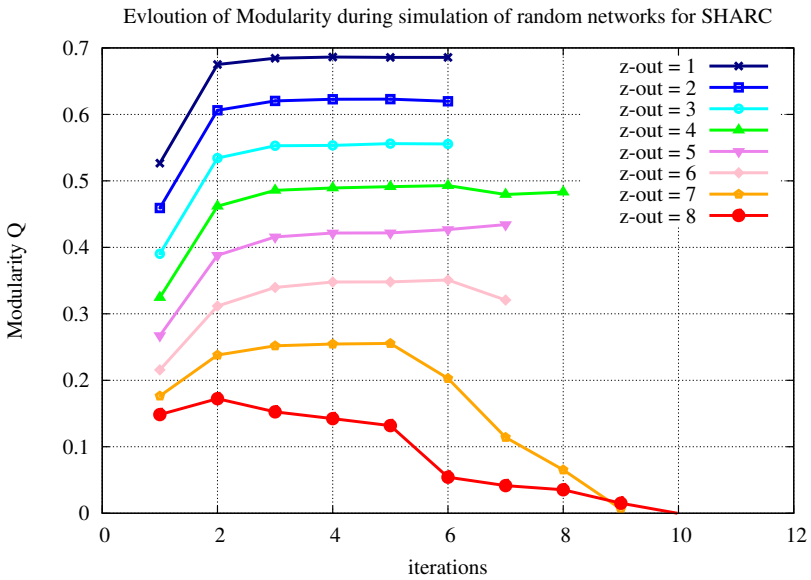
Impact of inter-community degree *z-out*

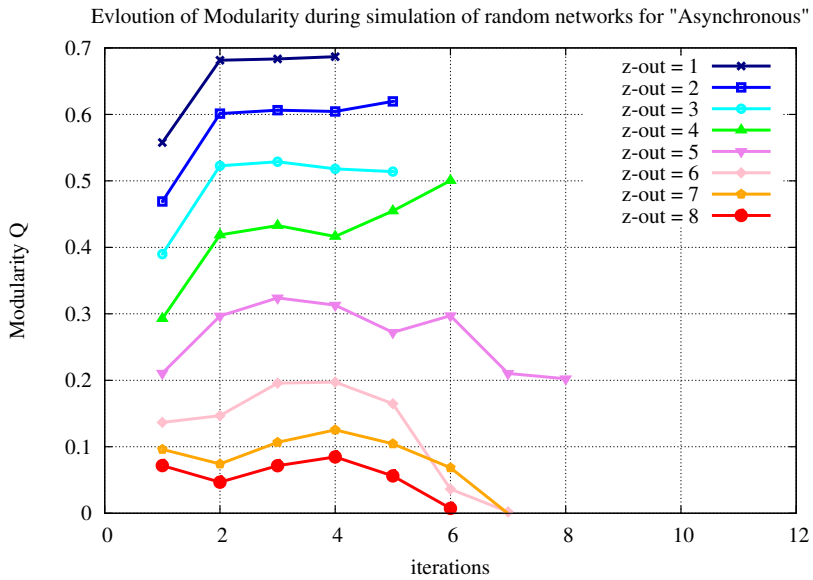




Results over 20 runs on 20 different networks per config. On 128-node network, 4 communities, avg. degree of 16

Impact of inter-community degree *z-out*





Results over 20 runs on 20 different networks per config. On 128-node network, 4 communities, avg. degree of 16

- Higher *z-out* means communities are less clear-cut
- SHARC provides equivalent avg. results for low *z-out*
- SHARC is more efficient when communities are not clear-cut