Long range community detection

(ongoing work)

Thomas Aynaud and Jean-Loup Guillaume

jean-loup.guillaume@lip6.fr

www.complexnetworks.fr

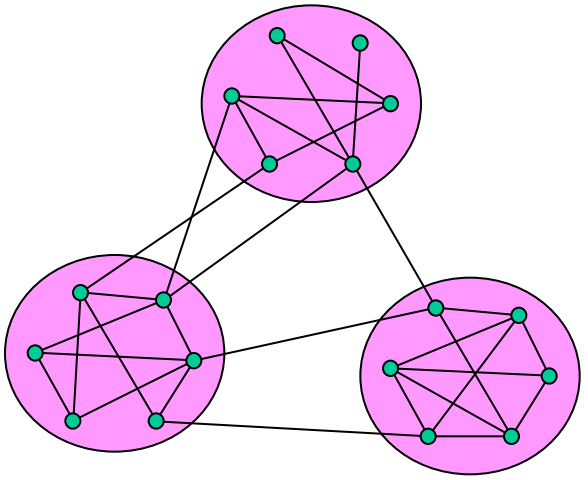
What is a community?

Set of nodes which share something:

- Persons with a similar interest (family members, friends)
- Web pages with a similar content
- Proteins with a similar function
- Blogs on a same topic, etc.

Relation with the structure of the network?

- Dense connections within communities
- Sparse connections between them
- -> Modularity



Dynamic networks

Most networks are evolving:

- New pages/sites appear on the web
- People begin new relationships
- Posts are created on blogs, etc.

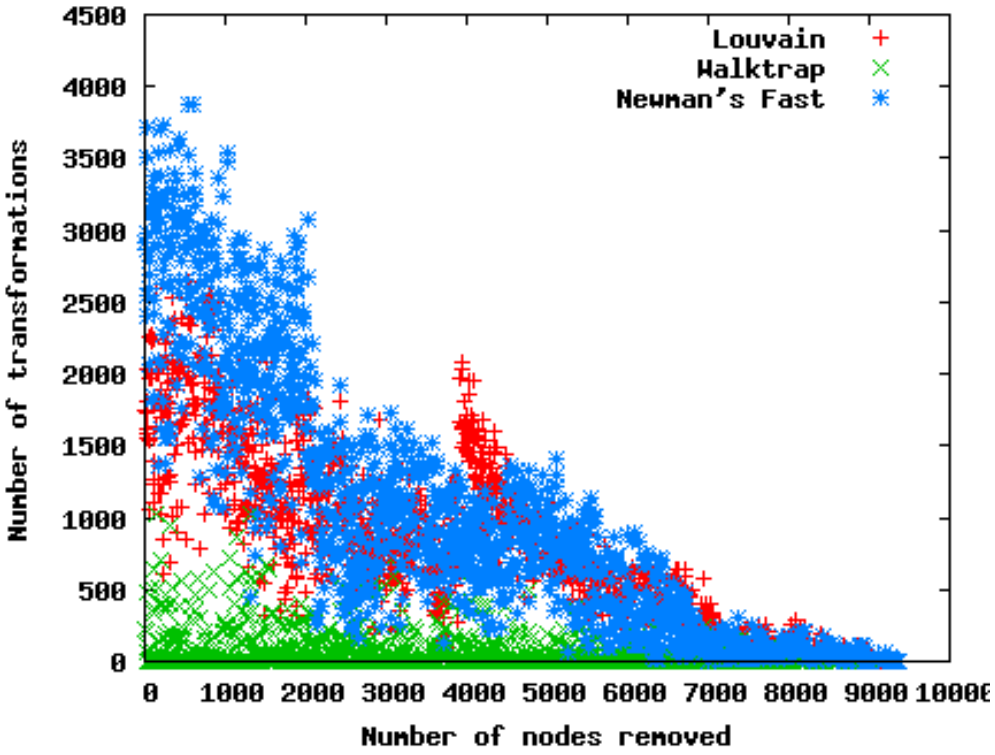
Classical approaches for communities dynamic networks:

- Forget the evolution
- Build a temporal network
- Compute communities (independently) at each time step

Evolving communities

Problems with computing communities at each time step:

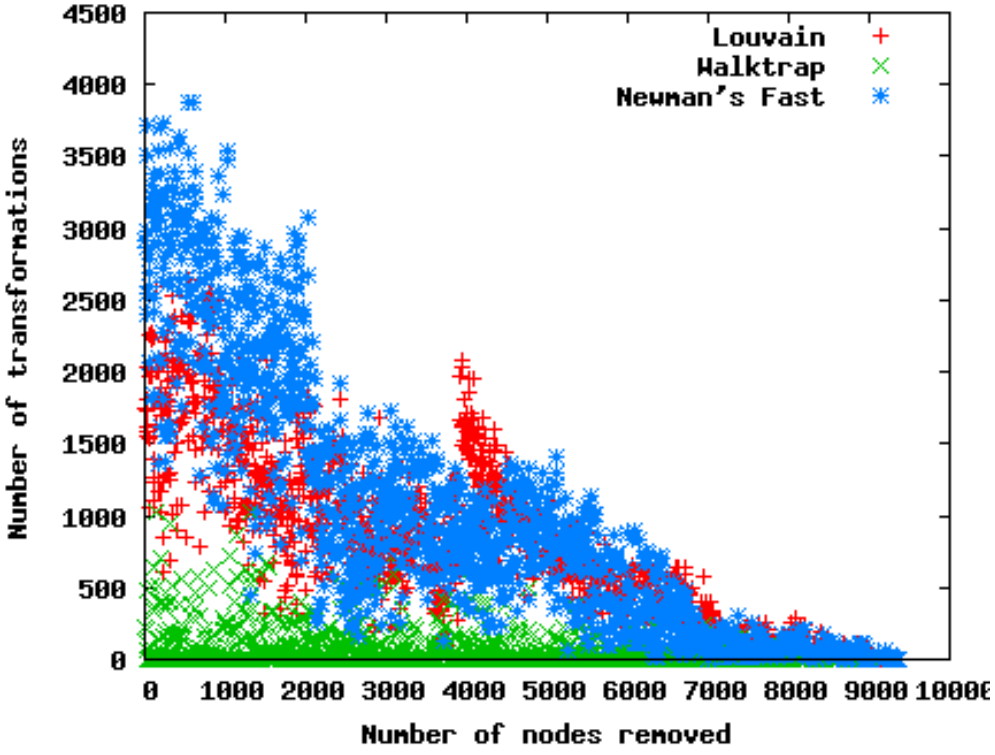
- Stability

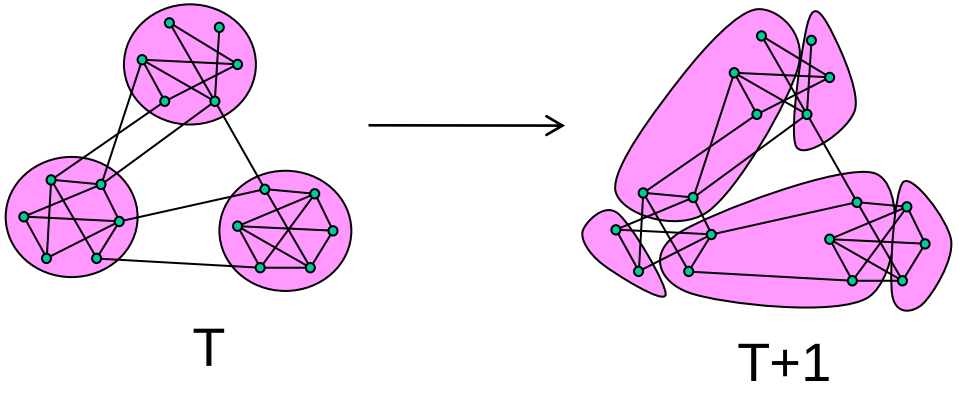


Evolving communities

Problems with computing communities at each time step:

- Stability
- Tracking communities from t to t+1





Dealing with stability

Basic idea [Song et al. SIGKDD 2007]:

- Do not compute communities independently
- Quality(t+1) = f(static quality(t+1), evolution quality(t->t+1))
- Communities at time t+1 are constrained by communities at time t

Extending this idea consists in removing all instabilities:

- We search ONE partition good in average

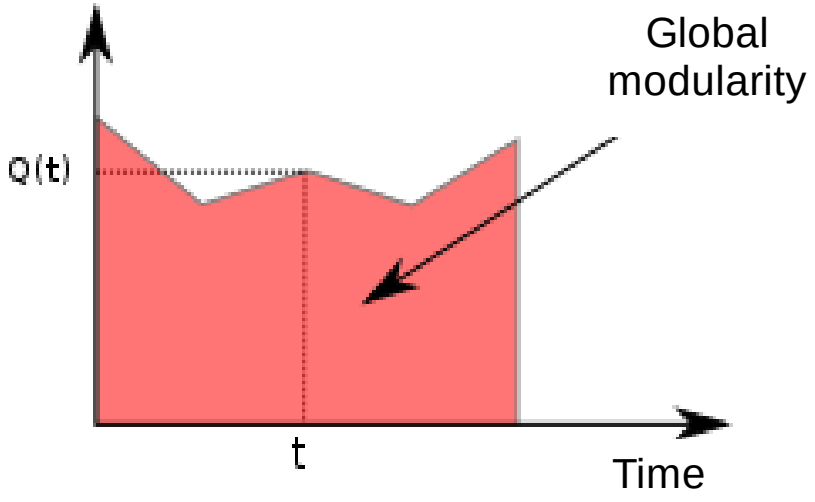
Global modularity - definition

Classical modularity: quality of a partition

Global modularity: sum of instantaneous/classical modularities

$G_i = (V_i, E_i)$ network at time i

$$Q_{global}\ (\pi, G = (G_1, G_2, ..., G_T)) = \sum_{t=1}^{t=T} Q(G_t, \pi_{|V_i})$$



Global modularity - computation

Louvain algorithm alternates two phases:

- Nodes are moved one by one using the modularity gain maximization
- Communities are grouped into a graph between communities

Gain maximization is easy to compute => fast algorithm

The same scheme can be applied to global modularity:

- Gain: sum of gain of modularity at each time step
- Grouping: each static graph is grouped independently
- Can be done more efficiently than the naive parallel implementation

Global modularity - validation

We use a simple dataset and show that:

- The partition is good (high modularity) at each time step
- The partition obtained on the dataset is plausible

Dataset used in this talk:

- Multicast routers neighbordhood topology (dataset from Pansiot et al.)
- Everyday, the connected component of a given router is acquired
- Measured on a daily basis during several years (low evolution)

Other datasets have been used:

- blogs, other Internet topologies, etc.

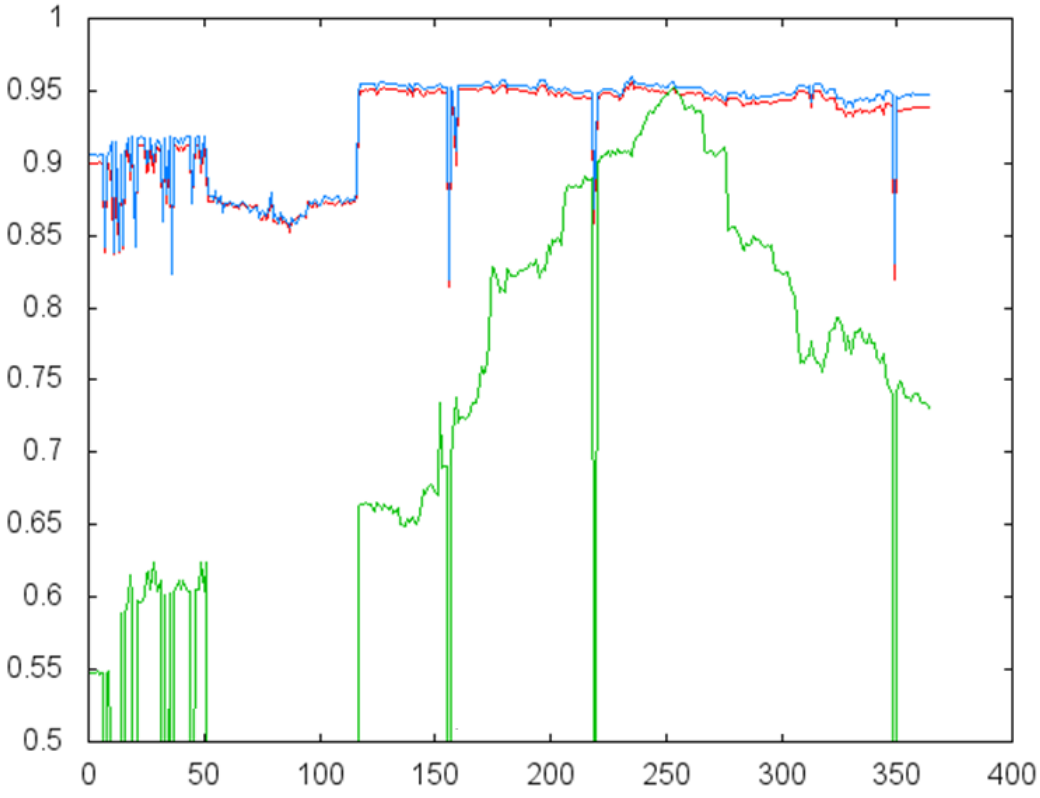
Comparing partitions

We compare:

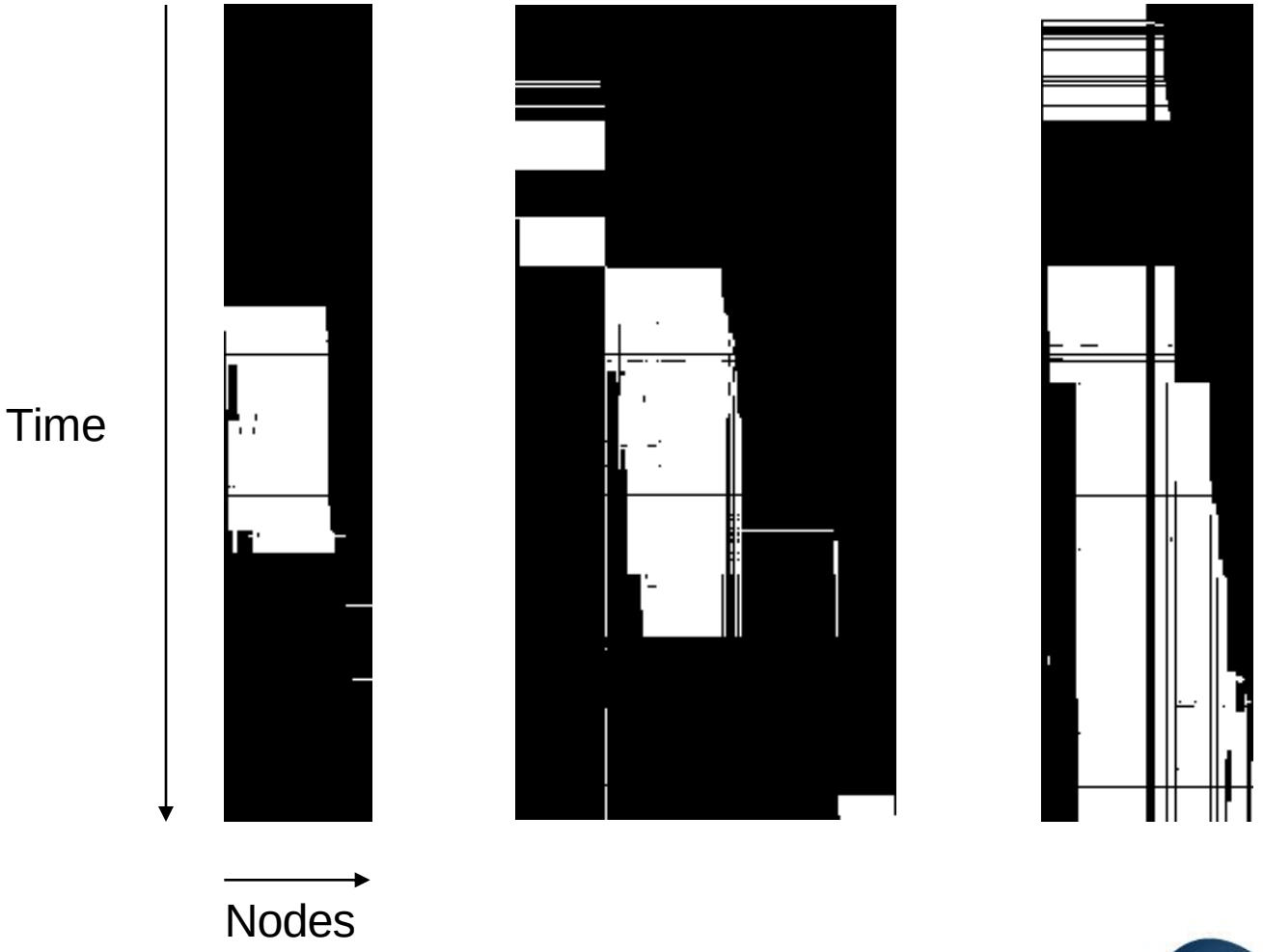
- The optimal at each time step
- The global modularity
- The best static partition (used as a global one)

The global partition:

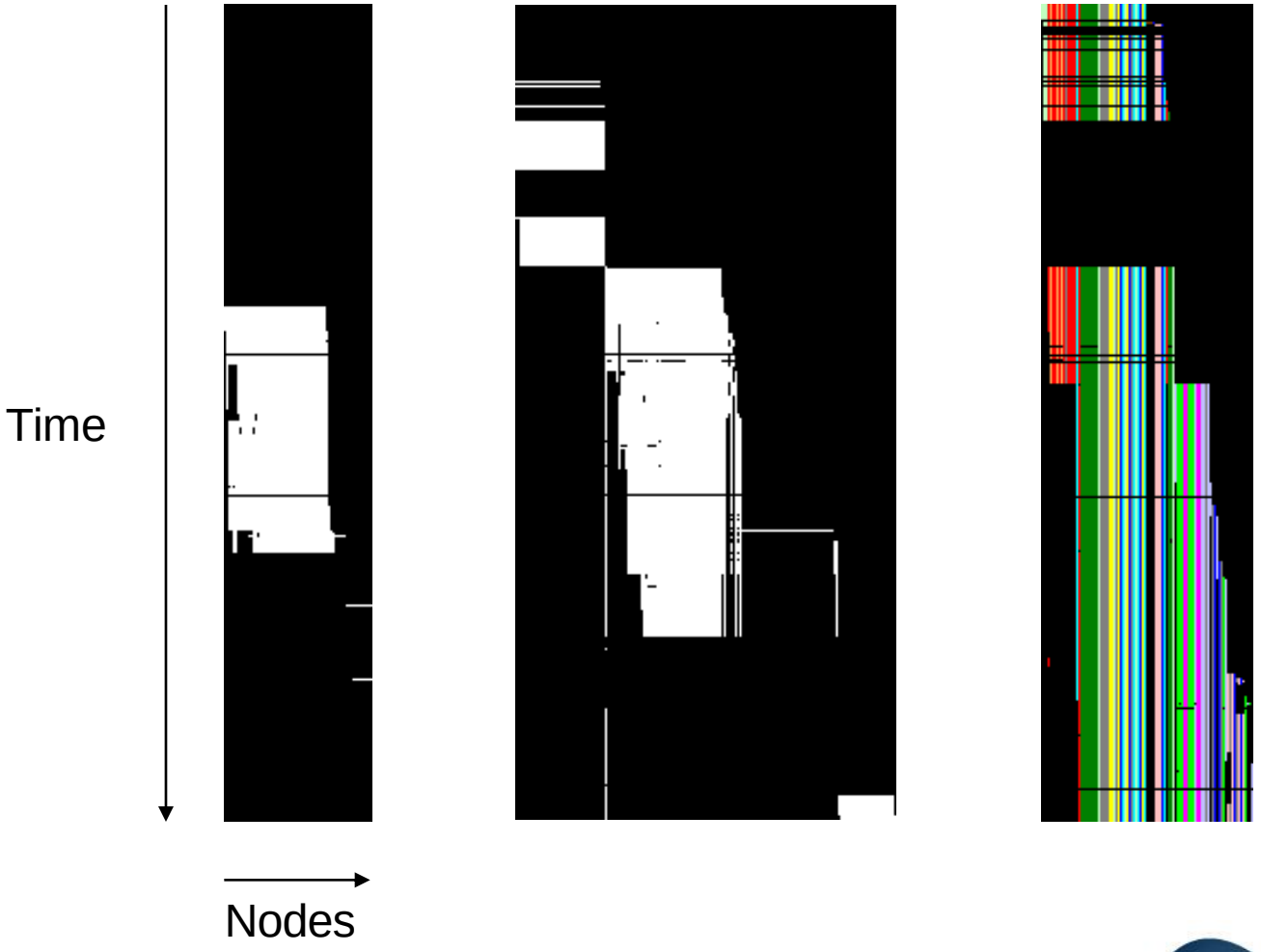
- Is always nearly optimal
- Cannot be found using classical Louvain
- Allows to find different phases



Nodes life inside communities

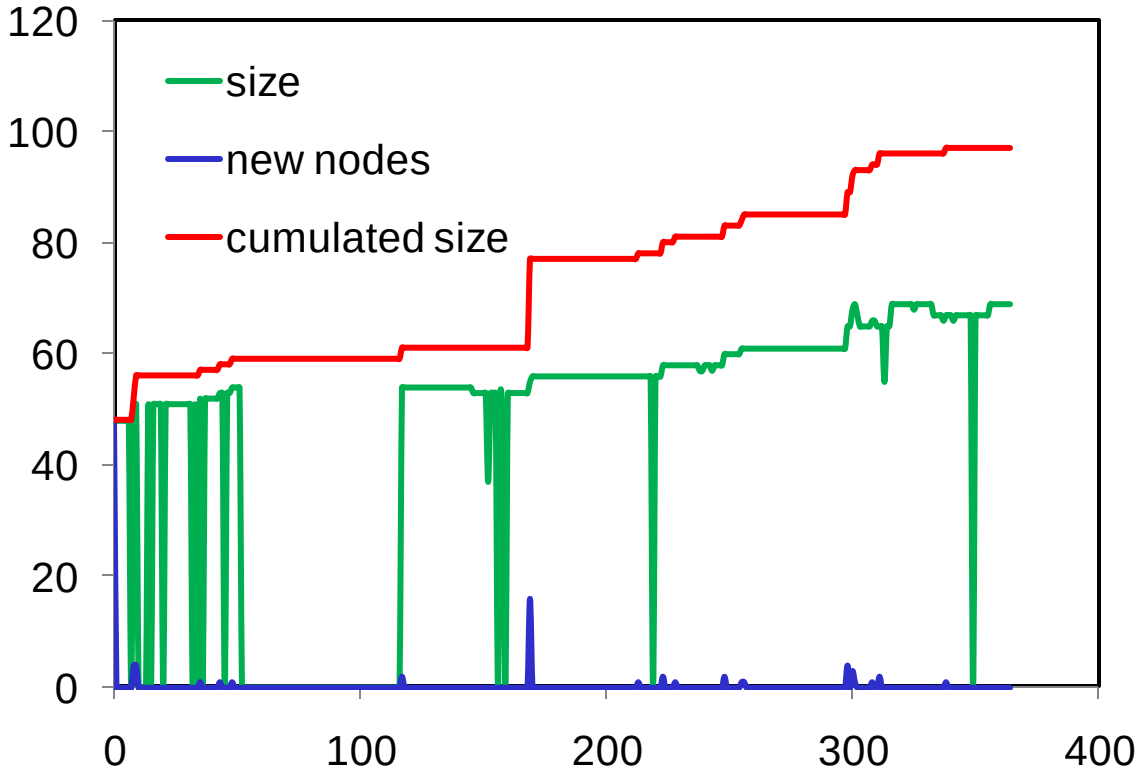


Nodes life inside communities



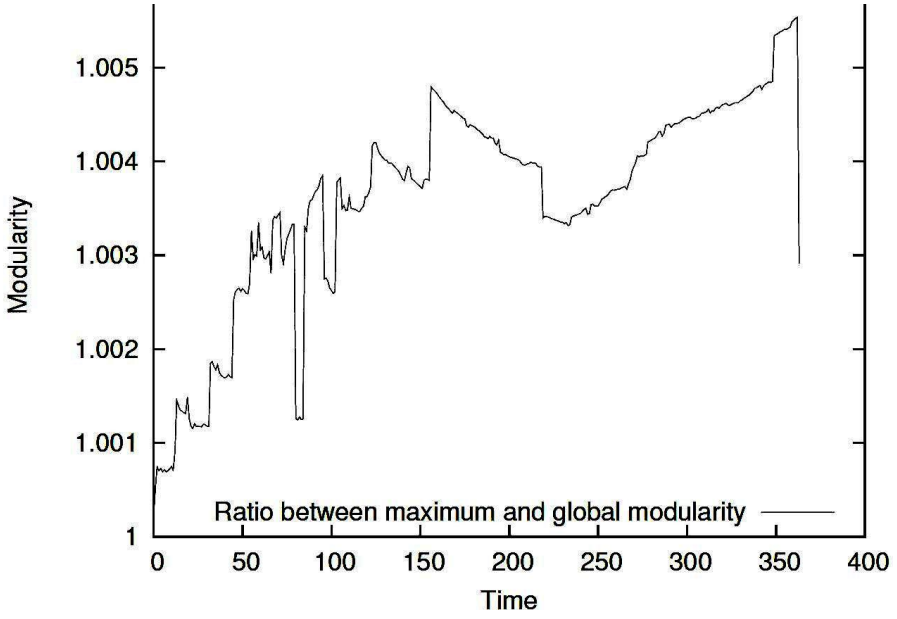
Community evolution

The partition does not evolve but the communities do.



Window size

Do we really want to optimize over the whole period?



Ratio between optimal and global modularity from 0 to time t

- Decreasing value: the structure is averaging
- Increasing value: the structure is changing

Conclusions and future work

Conclusion:

- A simple approach for community detection
- Allows to study the dynamic structure of a network
- Can (must) be optimized on smaller time windows

Future work:

- Analyze other datasets (with more dynamics)
- Causality

$$Q_{global}\ (\pi, G = (G_1, G_2, ..., G_T)) = \sum_{t=1}^{t=T} Q(G_t, \pi_{|V_i})$$

Thanks for your attention

www.complexnetworks.fr