Signaux-graphes pour l'étude d'interactions sociales

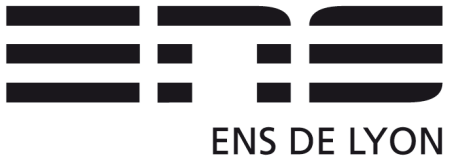
Parties 1 et 2 - Notions de groupes dans les interactions sociales

Pierre Borgnat

CR1 CNRS – Laboratoire de Physique, ENS de Lyon, Université de Lyon

Équipe SISYPHE : Signaux, Systèmes et Physique

06/2014



Qui je suis

- Membre de l'équipe SISYPHE du laboratoire de Physique de l'ENS de Lyon : **traitement statistique du signal**, en particulier pour les approches **multi-échelles** ou **non stationnaires**
- Traitement du signal sur les graphes
  / Traitement de graphes comme signaux
- Exemples de sujets d'étude et de données :

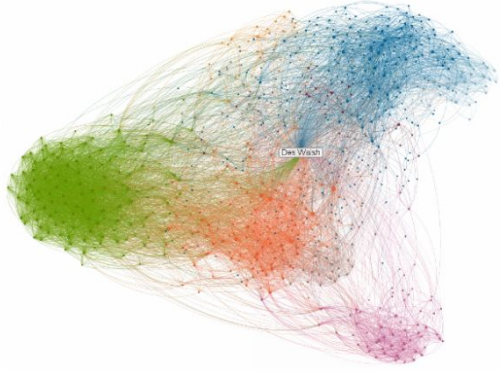
  métrologie des réseaux technologiques (internet, téléphones portables,...), réseaux de capteurs, réseaux sociaux, réseaux de mobilité (Vélo'v), réseaux neurones (par fMRI), réseaux génomiques, ...

1. Introduction : des signaux et des graphes

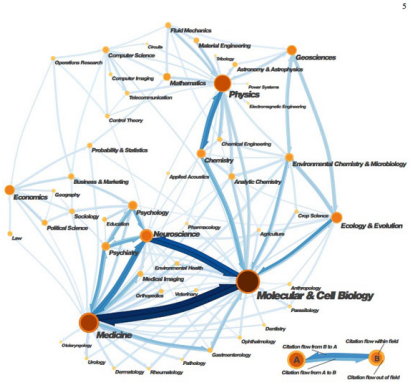
Enjeux scientifiques :

- Problèmes d'estimation non triviaux (ex.: mesures sans répétition et non stationnaires)
  $\rightarrow$ **méthodes statistiques avancées**
- grands graphes
  $\rightarrow$ **méthodes multi-échelles**
- graphes dynamiques
  $\rightarrow$ **méthodes non stationnaires**

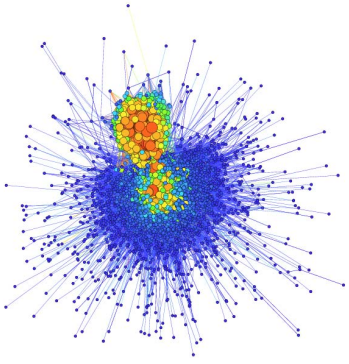
Quelques réseaux de notre monde numérique



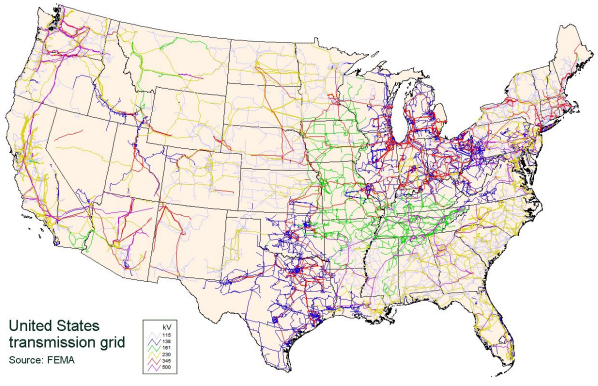
LinkedIn



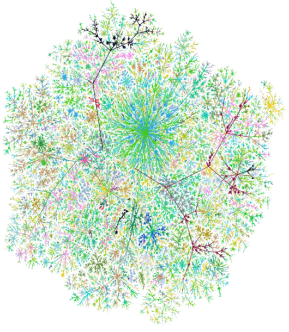
Citations scientifiques



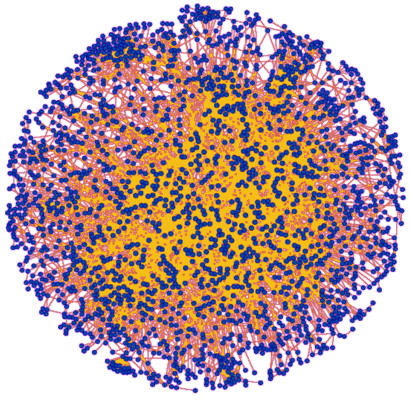
Véhicle



USA Power grid

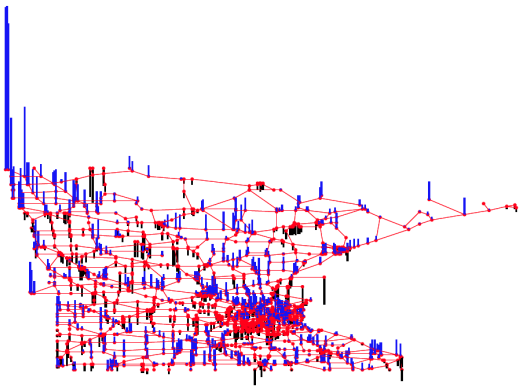


Web

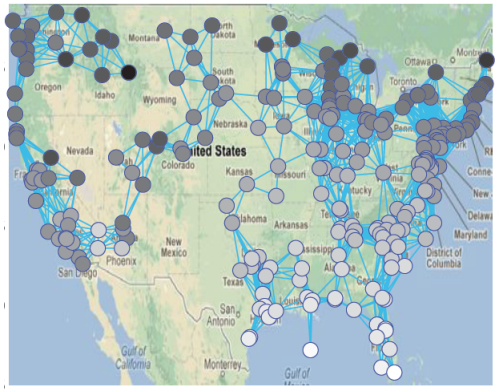


Protéines

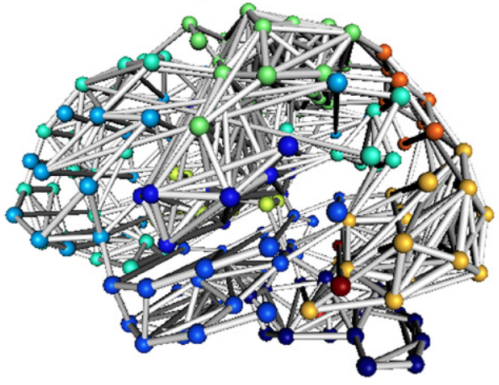
Exemples de signaux sur graphes



Minnesota Roads



USA Temperature



fcMRI Brain Network

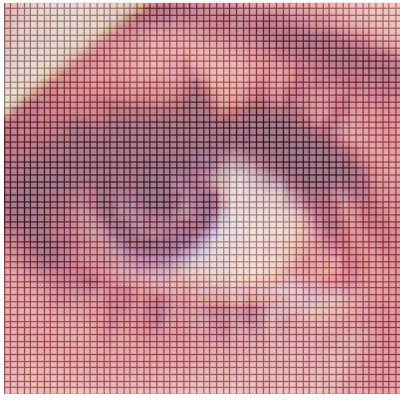


Image Grid

Color Point Cloud

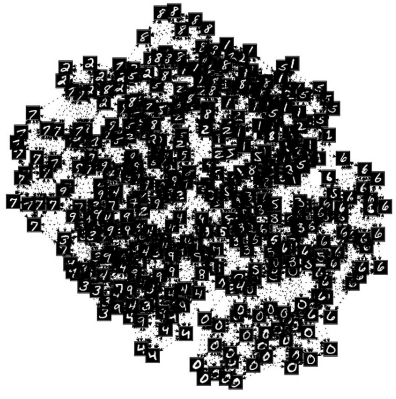
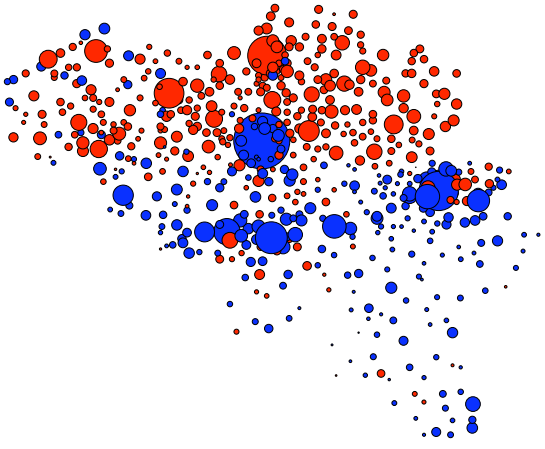


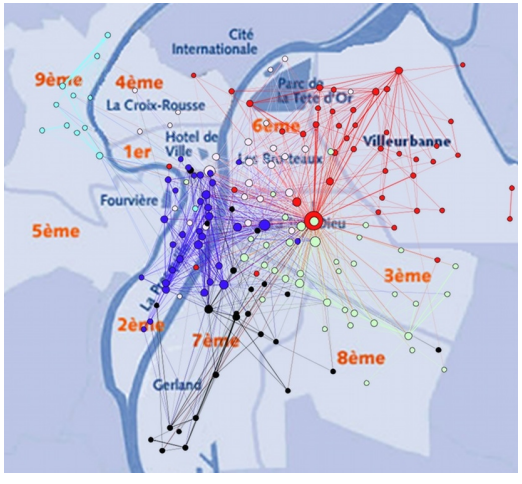
Image Database

Exemples de (quelques) données sociales

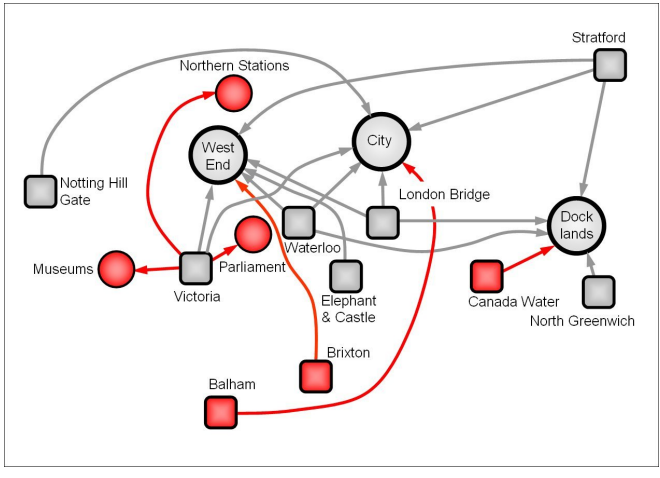
- Les outils numériques permettent de reprendre des problèmes d'analyse de données sociales ou liées aux activités humaines avec des données riches (grandes !)
- Les activités laissent des traces numériques : téléphonie, traces de déplacements (GPS, cartes métro ou bus, Vélo'v/Vélib',...)



Téléphones portables
[Blondel et al., 2008]



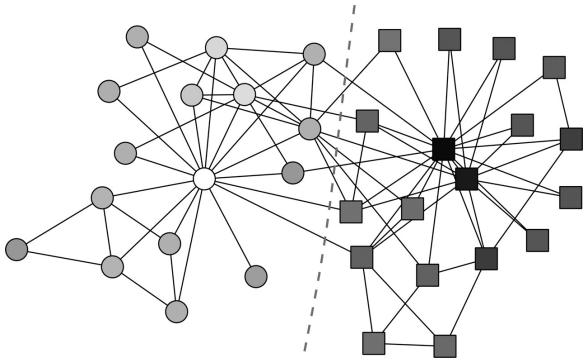
Vélo'v à Lyon
[Borgnat et al., 2013]



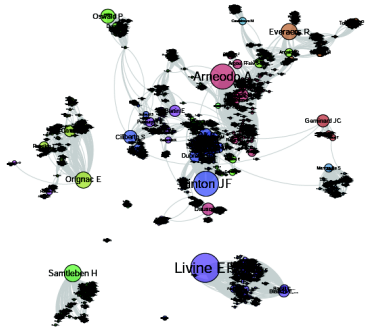
Métro Londres
[Roth et al., 2011]

Exemples de (quelques) données sociales

- Réseaux sociaux : de la sociologie classique...

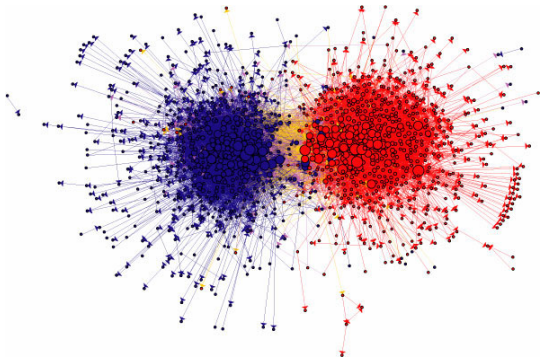


Zachary Karatee Club (1977) [Newman, 2006]

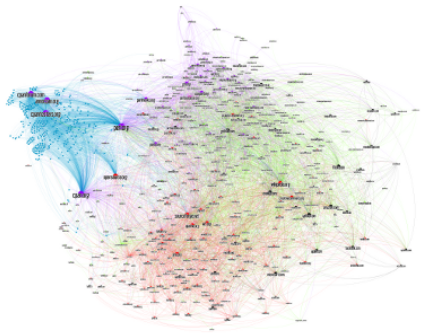


Co-citations pour scientometrie [Jensen et al., 2011]

- ... aux réseaux sociaux en ligne : Facebook et autres



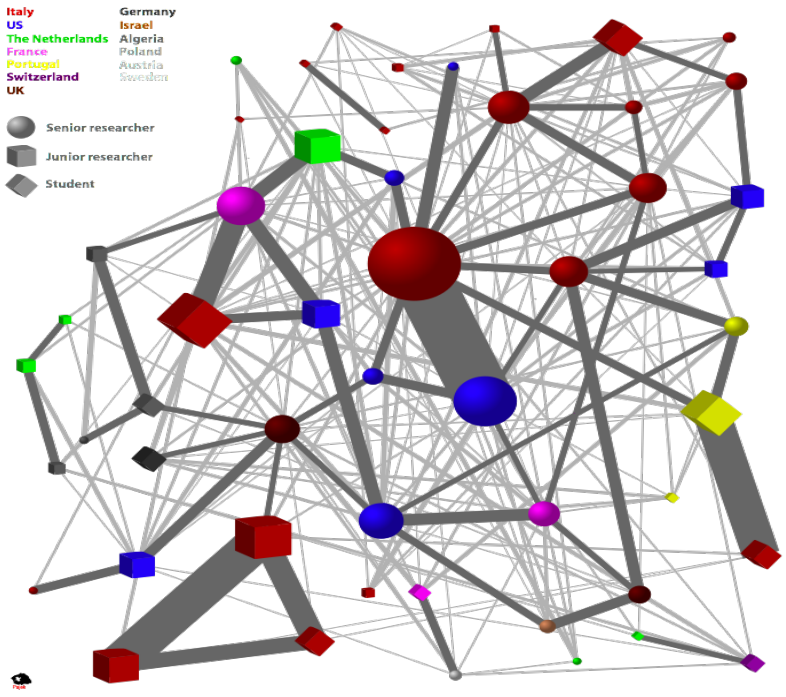
Blogosphère des élections présidentielles US de 2004



Communauté Perl (CPANTS data, vizualization Gephi)

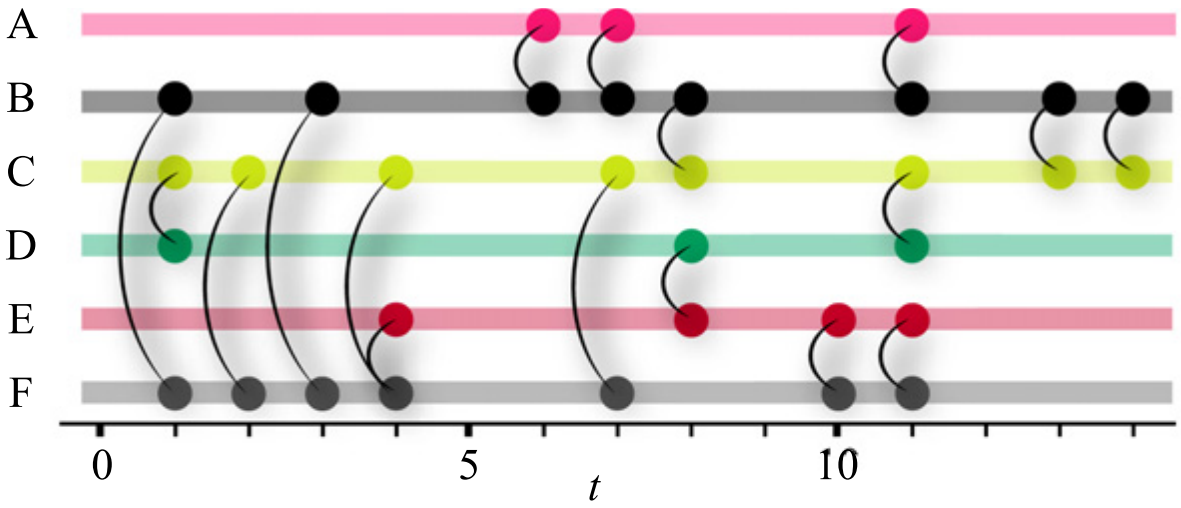
Exemples de (quelques) données sociales

- Des expériences conçues pour suivre les activités et les contacts entre personnes
- IMOTE : 41 nœuds, 3 jours [Chaintreau et al. 2006]
- MIT Reality Mining : 100 nœuds, 9 mois [Eagle et al. 2007]
- Projet MOSAR : 200 nœuds, qq mois[Fleury et al., 2010]
- Sociopatterns : qq 100 nœuds, qq jours [ISI, depuis 2009]



Aspects temporels

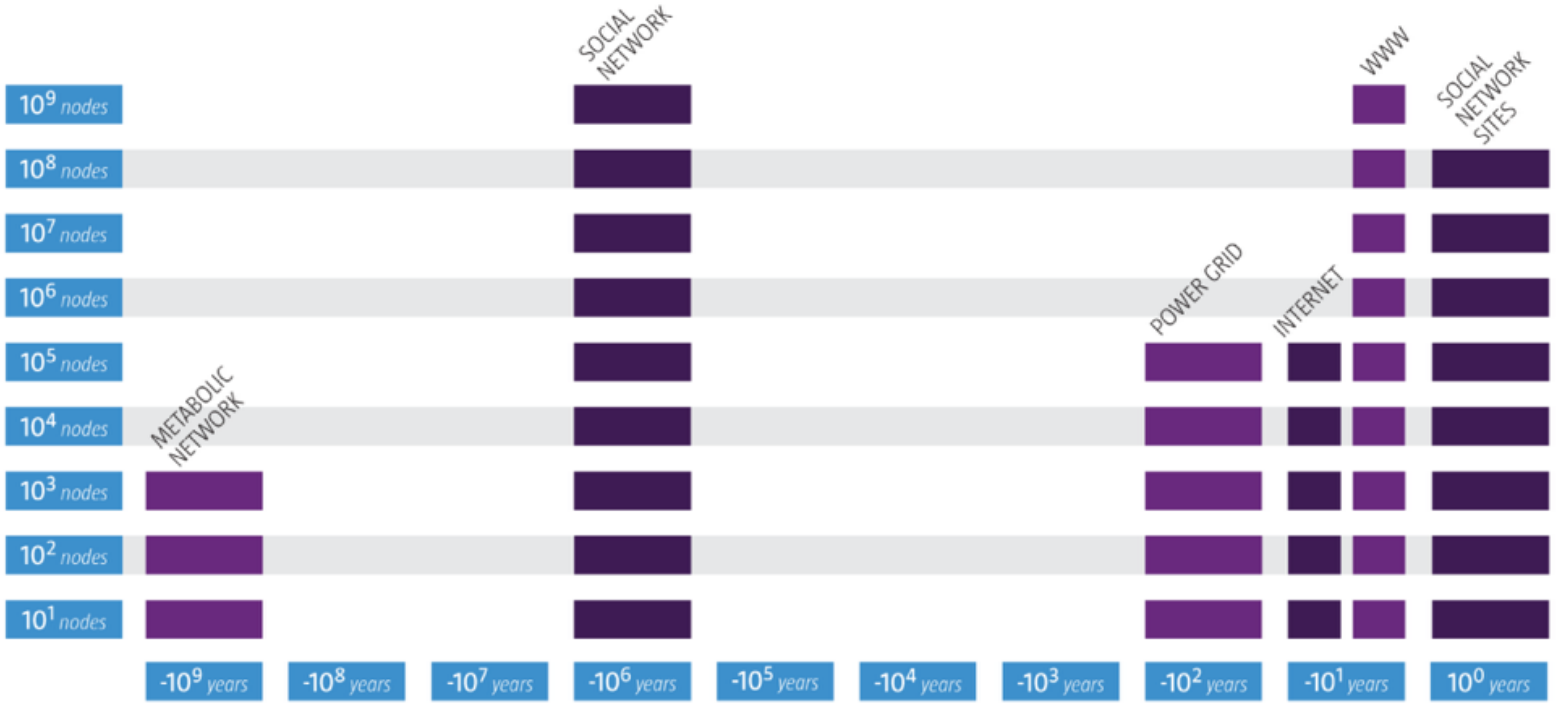
- Dynamiques temporelles non triviales
- Souvent, on a plutôt des flots temporels de relations que des réseaux statiques



(On reprendra cela en fin de cours)

Quelques propriétés de ces réseaux complexes

- Ces données sont-elles massives ?



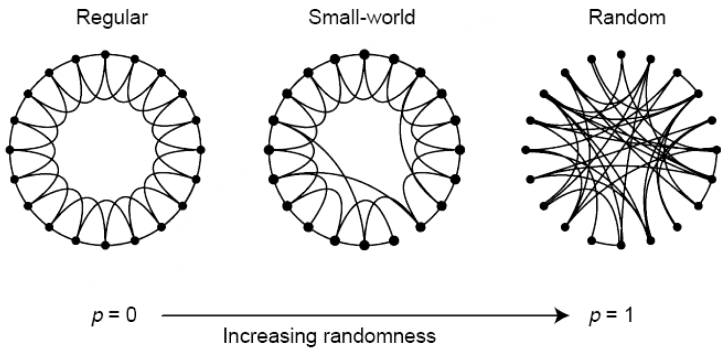
[Barabasi, 2012]

- Oui, mais pas vraiment dans cet exposé...

Quelques propriétés de ces réseaux complexes

- Propriété de **petit monde** : chemins courts

Exp. Milgram (1967)

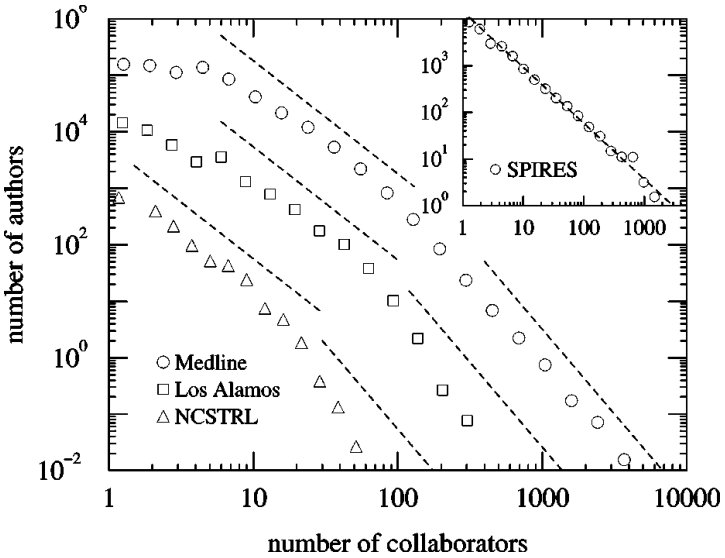


D.J. Watts and S. Strogatz, Nature (1998)

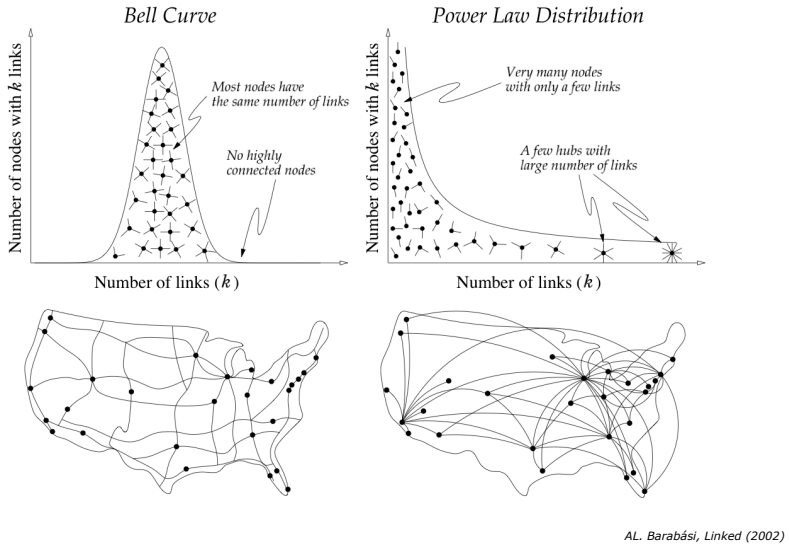
Modèles de Watts et Strogatz (1998)

Quelques propriétés de ces réseaux complexes

- Propriété “**scale free**” : pas d’échelles (ou lois d’échelles) [Barabasi, 1999]



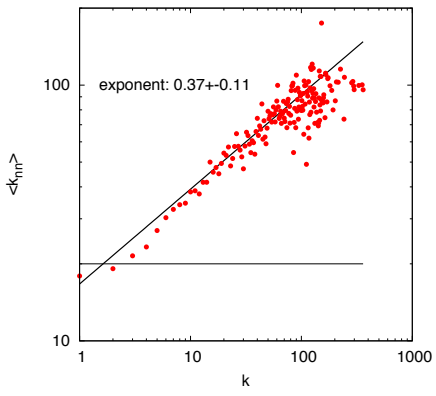
Réseaux de citations [Newman 2001]



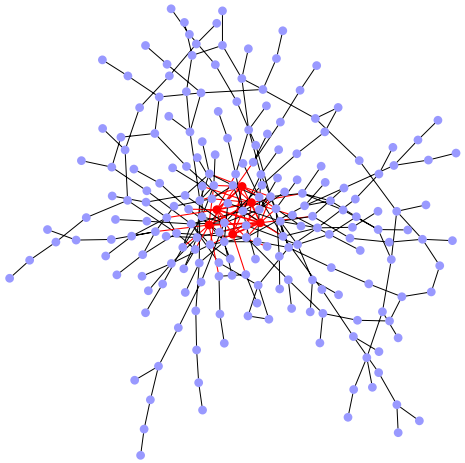
Allure des réseaux

Quelques propriétés de ces réseaux complexes

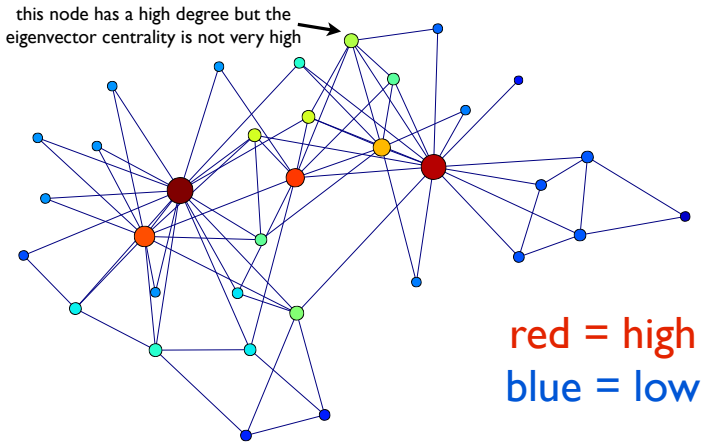
Assortativité (corrélation degré, degré des NN)



Astrophysics co-authorship network



Centralité (certains nœuds sont “au centre")

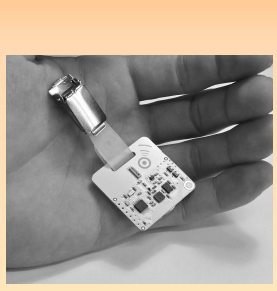


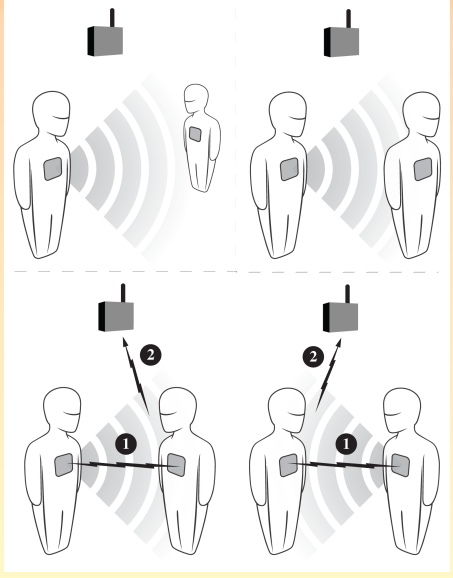
Zachary’s Karate Club network (ex. 2.1, 2.2),
colour = eigenvector centrality

Exemples : contacts face-à-face entre personnes

[N. Tremblay, A. Barrat, et al., PRE, 2013]

- Mesures à l'aide de la plate-forme `sociopatterns.org`
- Collecte de contacts entre personnes, résolus en temps à des conférences, des musées, des écoles, des hôpitaux,...
- Contacts face-à-face résolus dans le temps

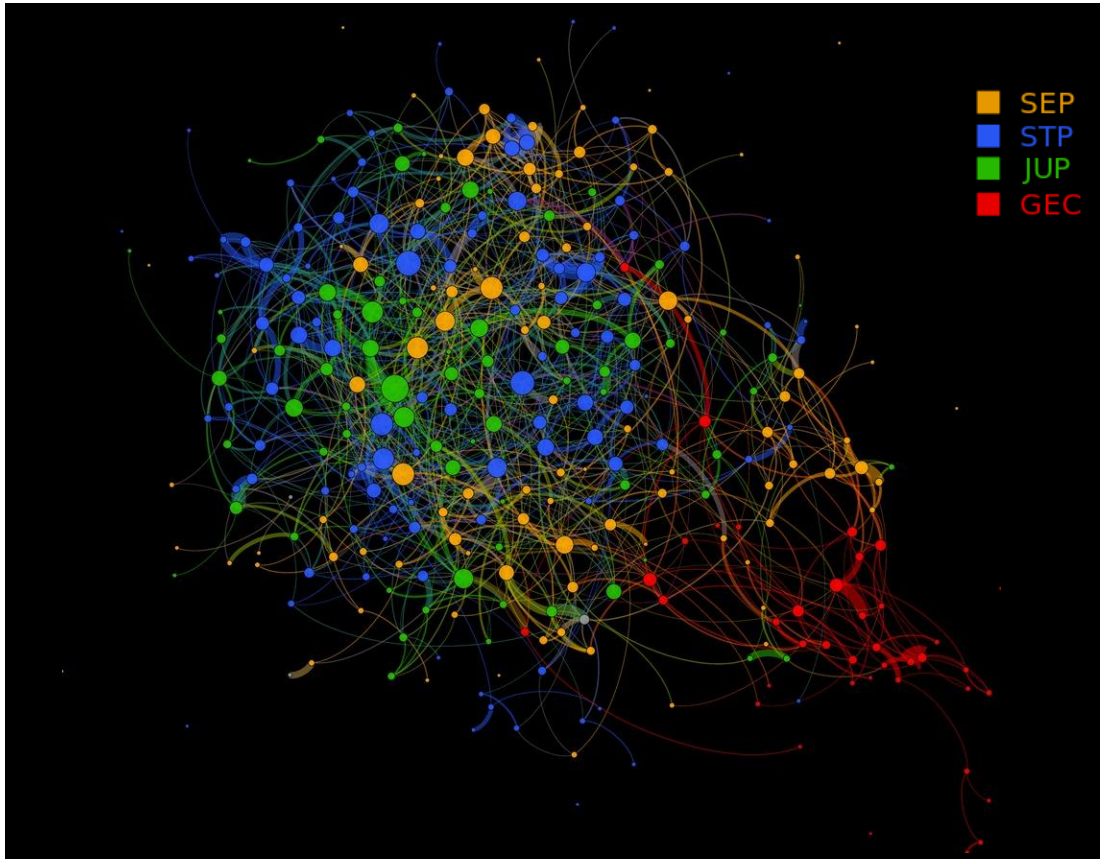




Exemples : contacts face-à-face entre personnes

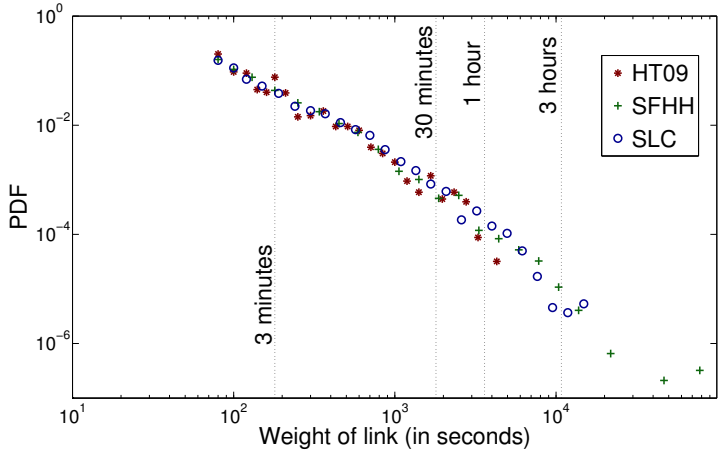
[N. Tremblay, A. Barrat, et al., PRE, 2013]

- Collecte 11/2011 à SLC, 2 conférences colocalisées : DPP et GEC
- 320 participants, 5 journées, 25 000 contacts mesurés

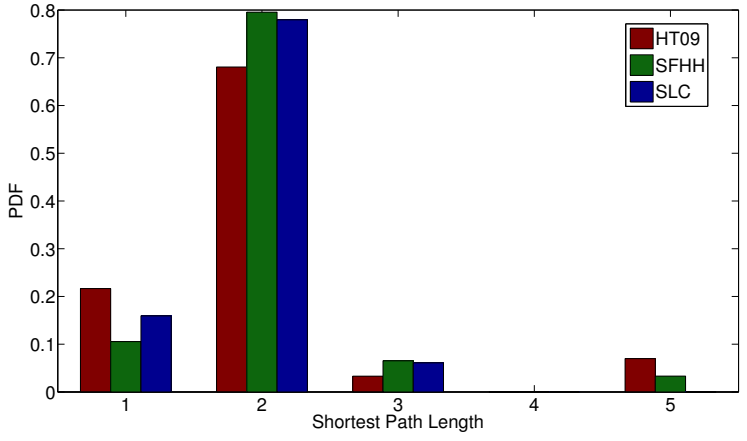


Des propriétés classiques de réseaux complexes

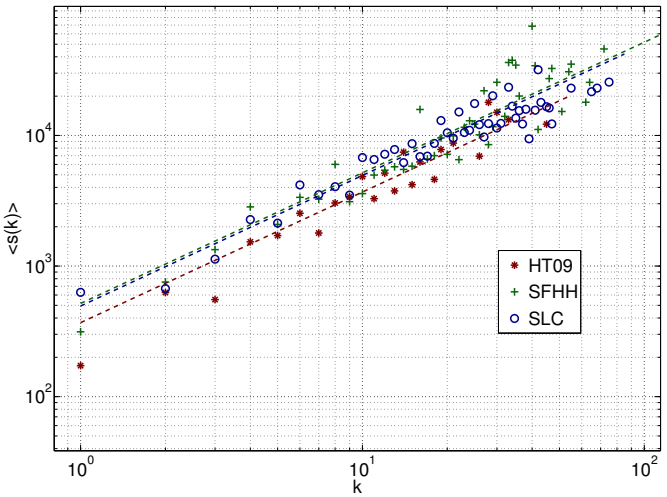
Poids sur les liens



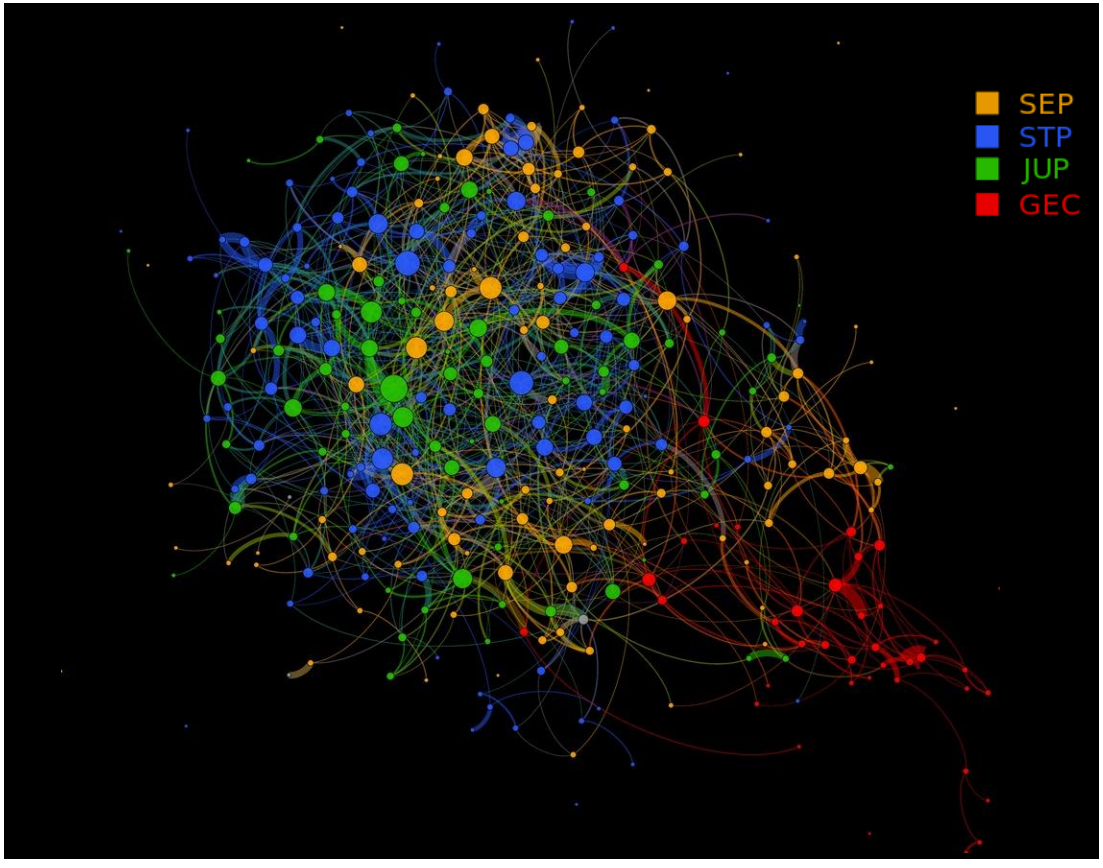
Petit-monde : plus courts chemins



Corrélation degré - force



Réseaux complexes : échelles et dynamique



[FILM]

- À quelle échelle décrire les groupes (communautés) ?
- Comment extraire des informations de la dynamique ?

2. Communities in complex networks

- Networks are often inhomogeneous in their contacts and made of **communities (or modules)**:
  groups of nodes having a larger proportion of links inside the group than with the outside
- This is observed in various types of networks: social, technological, biological,...
- There exist several extensive surveys:

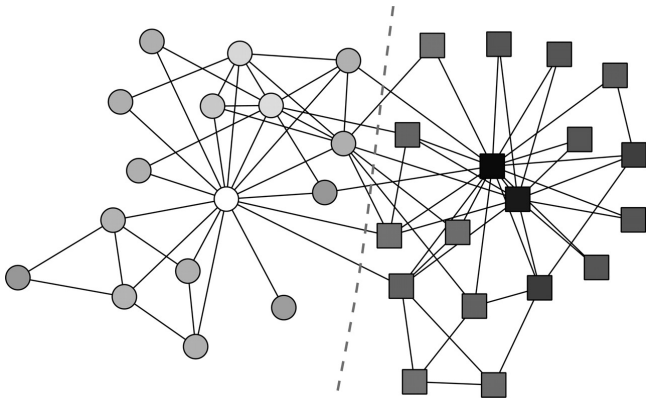
  [S. Fortunato, *Physic Reports*, 2010]

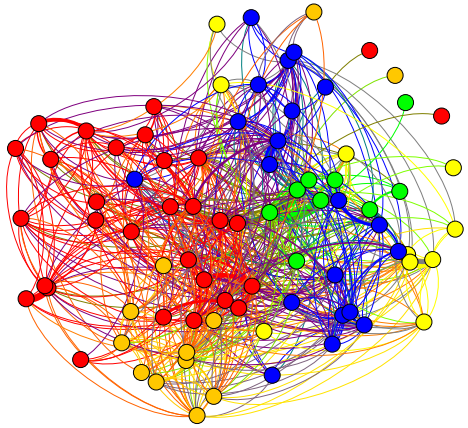
  [von Luxburg, *Statistics and Computating*, 2007]

  ...

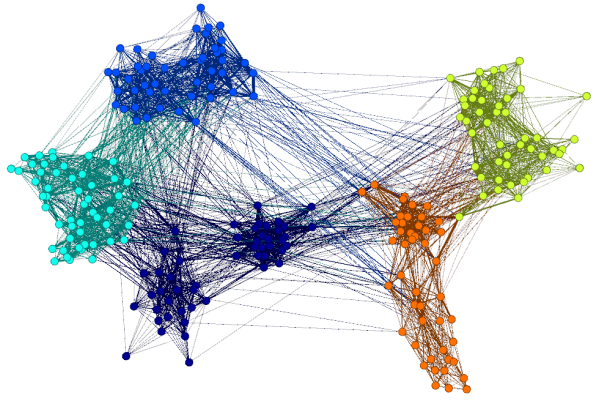
Communities in social or human-related networks

- Zachary Karatee Club;



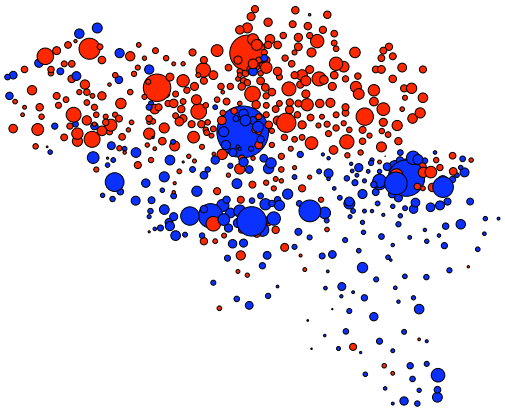


Sociopatterns data

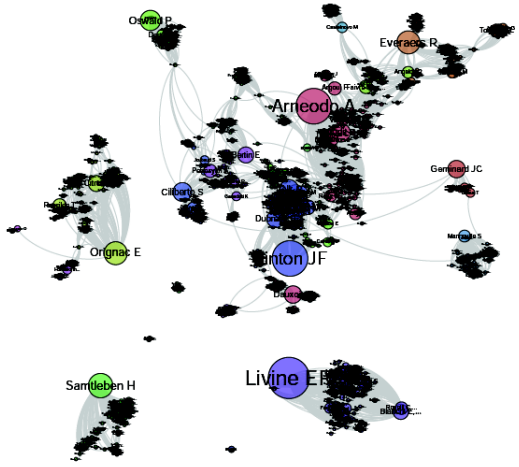


(Lab. physique, ENSL, 2013) (école primaire, Sociopatterns)

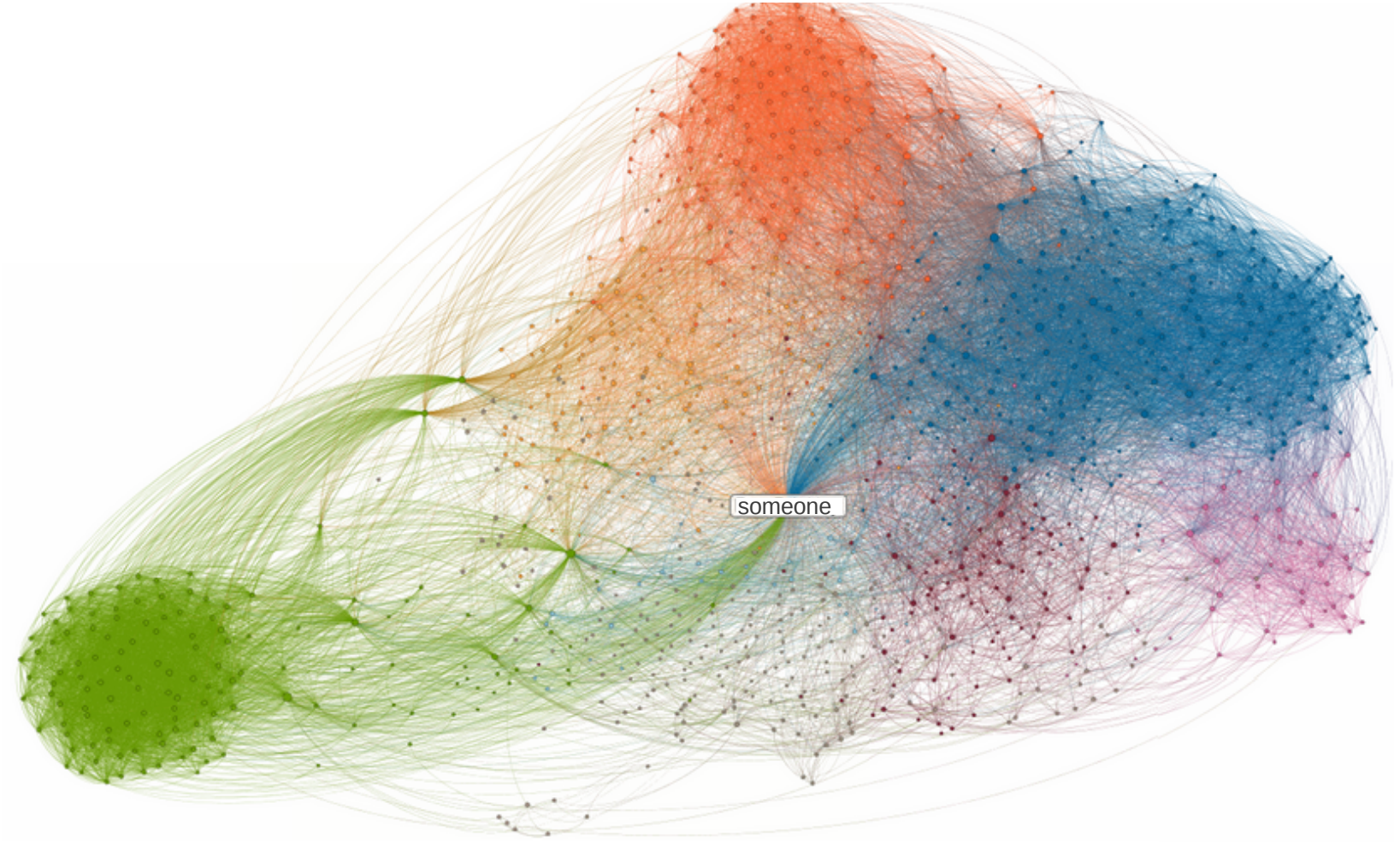
- Mobile phones;



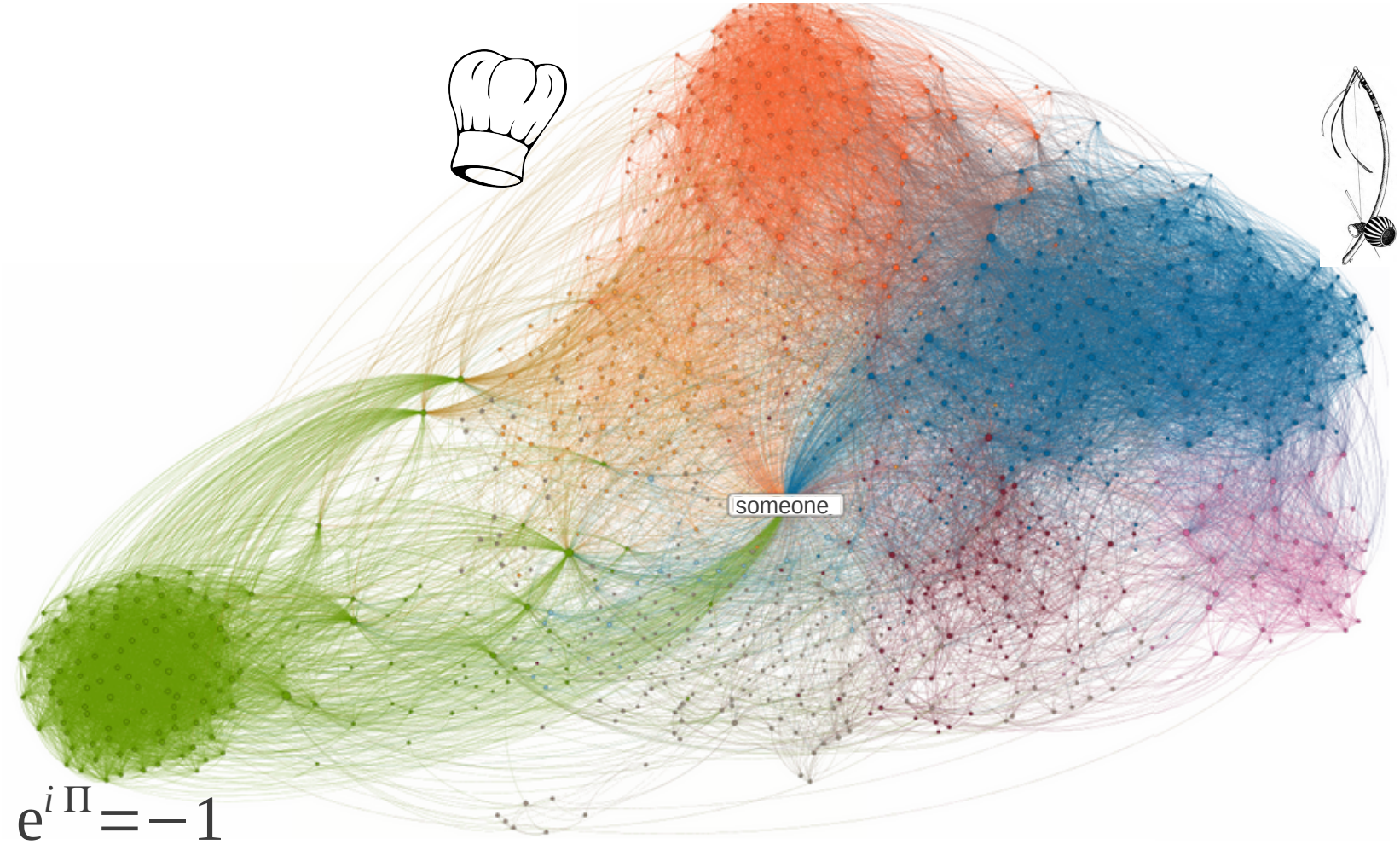
Scientometric networks



Purpose of community detection?

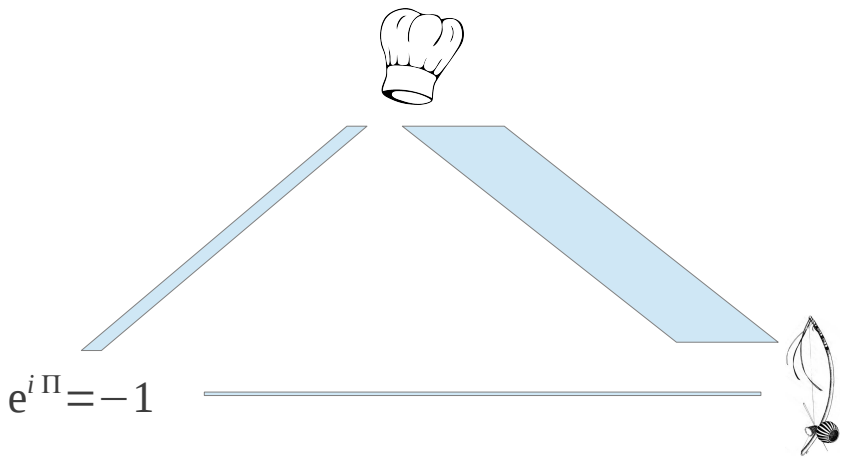


Purpose of community detection?



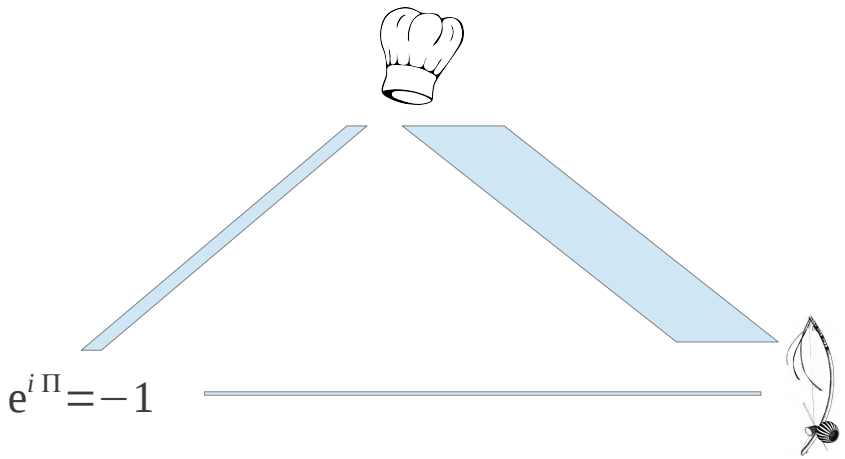
Purpose of community detection?

1) It gives us a sketch of the network:



Purpose of community detection?

1) It gives us a sketch of the network:



2) It gives us intuition about its components:

$e^{i\pi}=-1$   ?

Methods to find communities

- No pretention of a full survey... Some important steps are:
- Cut algorithms (legacy from computer science)
- Spectral clustering (relaxed cut problem)
- Modularity optimization (there arrive the physicists) [Newman, Girvan , 2004]
- Greedy modularity optimization a la Louvain (computer science strikes back) [Blondel et al., 2008]
- Ideas from information compression [Rosvall, Bergstrom, 2008]

From graph bisection to spectral clustering

- Graph bisection (or cuts): find the partition in two (or more) groups of nodes that minimize the cut size (i.e., the number of links cut)
- Exhaustive search can be computationally challenging
- Also, the cut is not normalized correctly to find groups of relevant sizes
- Spectral interpretation: compute the cut as function of the adjacency matrix A

Wait... What means **spectral** for networks ?

Spectral analysis of networks

Spectral theory for network

This is the study of graphs through the **spectral analysis** (eigenvalues, eigenvectors) of matrices **related to the graph**: the adjacency matrix, the Laplacian matrices,....

Notations

$\mathcal{G} = (V, E, w)$	a weighted graph	
$N = \lvert V \rvert$	number of nodes	
A	adjacency matrix	$A_{ij} = w_{ij}$
d	vector of strengths	$d_i = \sum_{j \in V} w_{ij}$
D	matrix of strengths	$D = diag(d)$
f	signal (vector) defined on V	

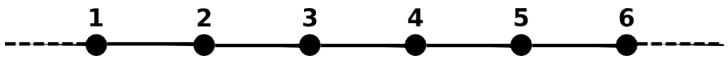
Definition of the Laplacian matrix of graphs

Laplacian matrix

L	laplacian matrix	$L = D - A$
(λ_i)	L's eigenvalues	$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_{N-1}$
(χ_i)	L's eigenvectors	$L\,\chi_i = \lambda_i\,\chi_i$

Note: $\chi_0 = \mathbf{1}$.

A simple example: the straight line



$$\longleftrightarrow \quad L = \begin{pmatrix} \ddots & & & & & \\ \dots & -1 & 0 & 0 & 0 & 0 \\ \dots & 2 & -1 & 0 & 0 & 0 \\ & -1 & 2 & -1 & 0 & 0 \\ & 0 & -1 & 2 & -1 & 0 \\ & 0 & 0 & -1 & 2 & -1 \\ & 0 & 0 & 0 & -1 & 2 \\ & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \begin{matrix} \\ \\ \\ \\ \\ \dots \\ \dots \\ \ddots \end{matrix}$$

For this regular line graph, L is the 1-D classical laplacian operator (i.e. double derivative operator).

Going back to spectral clustering

- Let $R = \frac{1}{2} \sum_{i,j \text{ in} \neq \text{groups}} A_{ij}$.
  This is equal to the cut size between the two groups
- Let us note $s_i = \pm 1$ the assignment of node i to group labelled $+1$ or -1
- $R = \frac{1}{2} \sum_{i,j} A_{ij}(1 - s_i s_j) = \frac{1}{4} \sum_{i,j} L_{ij} s_i s_j = \frac{1}{4} \mathbf{s}^\top L \mathbf{s}$
- Spectral decomposition of the Laplacian:

$$L_{ij} = \sum_{k=1}^{N-1} \lambda_k (\chi_k)_i (\chi_k)_j$$

- The optimal assignment vector (that minimizes R) would be $s_i = (\chi_1)_i \ldots$ if there were no constraints on the s_i's...
- However, $s_i = +1$ or -1.

Spectral clustering

- Problem with relaxed constraints:

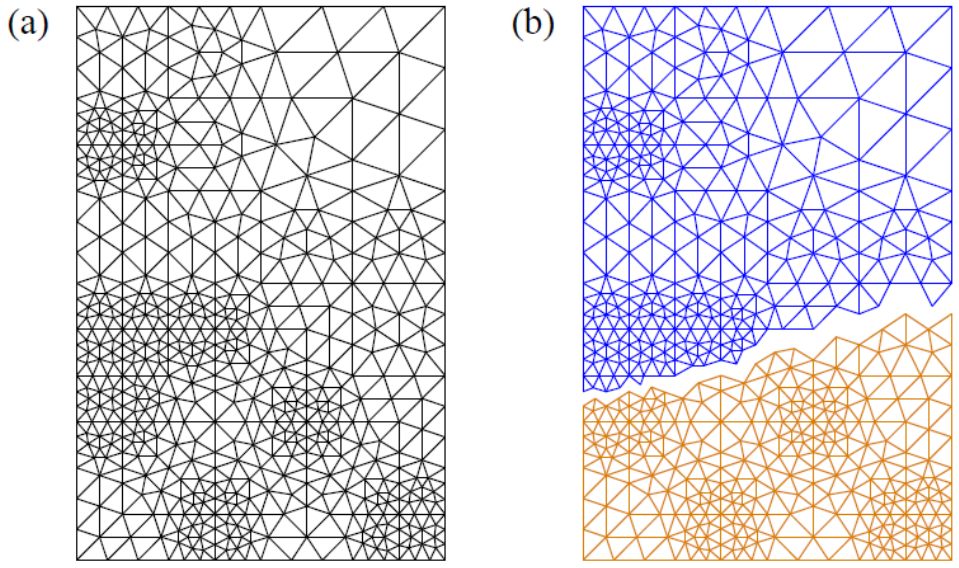
$$\min_{\mathbf{s}} \quad \mathbf{s}^\top L\mathbf{s}$$
$$\text{such that } \mathbf{s}^\top \mathbf{1} = 0, \ ||\mathbf{s}||_2 = \sqrt{\mathrm{N}}$$

- Simplest solution of this spectral bisection: $s_i = \mathrm{sign}((\chi_1)_\mathrm{i})$
- This estimates communities that are close to χ_1 (known as the the Fiedler vector)
- This allows also for *Spectral clustering of data* represented by networks

cf. [von Luxburg, *Statistics and Computating*, 2007]

Spectral clustering

- Example of spectral bisection on an irregular mesh



- Not really good for natural modules / communities

Spectral clustering

- More general spectral clustering: Use all (or some) of the eigenvectors χ_i of L (embedding !)
- Then, use a classical K-means on these first K non-null eigenvectors of L
  (each node a having the $(\chi_k)_a$ as feature)
- If large heterogeneity of degrees: the normalized Laplacian gives better results

Normalized Laplacian matrix

$\mathscr{L}$	Laplacian matrix	$\mathscr{L} = I - D^{-1/2} A D^{-1/2}$
(λ_i)	$\mathscr{L}$'s eigenvalues	$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{N-1}$
(χ_i)	$\mathscr{L}$'s eigenvectors	$\mathscr{L}\,\chi_i = \lambda_i\,\chi_i$

Interpretation as random walks (part 1)

- A random walk on a graph can be described by means on the adjacency operator. In particular, the occupancy probability $\mathbf{p}(t)$ at time t evolves like:

$$\mathbf{p}(t) = AD^{-1}\mathbf{p}(t-1) = W\mathbf{p}(t-1)$$

- Transition matrix W has a symmetrized version

$$S = D^{-1/2}AD^{1/2}$$

  which has same eigenvalues
- Many properties of random walks derives from the normalized Laplacian (symmetric or not)

Interpretation as random walks (part 1)

- Example 1: lazy random walk (which stays in place with prob. 1/2) converges to equilibrium π in

$$||\mathbf{p}_a(t) - \pi(a)||_2 \leq \sqrt{\frac{d(a)}{\min_u d(u)}}\,(1 - \lambda_{N-1}(W))^t$$

  and $1 - \lambda_{N-1}(W) = \lambda_1(\mathscr{L})$.
- Example 2: relation to normalized cuts

$$\lambda_1(\mathscr{L}) = \min_{\mathbf{s},\, d^\top \mathbf{s}=0} \frac{\mathbf{s}^\top L \mathbf{s}}{\mathbf{s}^\top D \mathbf{s}}$$

Quality of a partition: the Modularity

- Problems with spectral clustering:
  1) No assessment of the quality of the partitions
  2) No reference to comparison to some null hypothesis (or "mean field") situation
- Improvement: the modularity [Newman, 2003]

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{d_i d_j}{2m} \right] \delta(c_i, c_j)$$

  where $2m = \sum_i d_i$.
- Q is between -1 and $+1$
  (in fact, lower than $1 - 1/n_c$ if n_c groups)
- Algebraic form: modularity matrix $B = \frac{A}{2m} - \frac{dd^\top}{(2m)^2}$ and $Q = Tr(c^\top B c)$ for a partition vector c of the nodes.

Quality of a partition: the Modularity

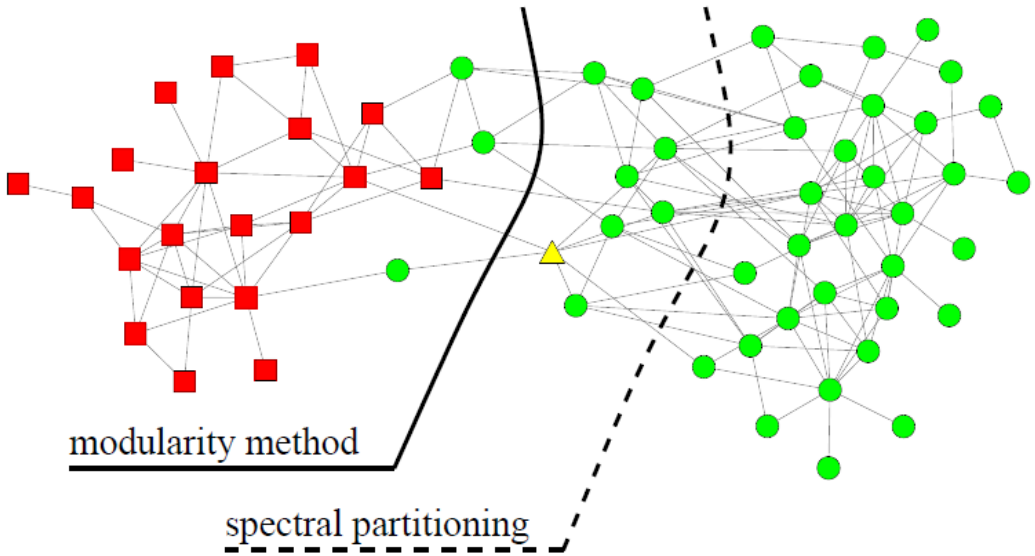
- Interpretation: $\frac{d_i d_j}{2m}$ is, for a null model as a Bernoulli random graph (with prob. $2m/N/(N-1)$ of existence of each edge), the fraction of edges expected between nodes i and j.
  (Note: in fact, it's best seen as a Chung-Lu model (2002))
- Re-written in term of groups, it gives

$$Q = \sum_{c=1}^{n_c} \left[\frac{l_c}{m} - \left(\frac{d_c}{2m} \right)^2 \right]$$

  where l_c is the number of edges in group c and d_c is the sum of degrees of nodes in group c.
- Consequence: the larger Q is, the most pronounced the communities are

Community detection with modularity

- By optimization of Q
- For instance: by simulated annealing, by spectral approaches,...
- Works well, better than spectral clustering.



Communities from modularity for big data network

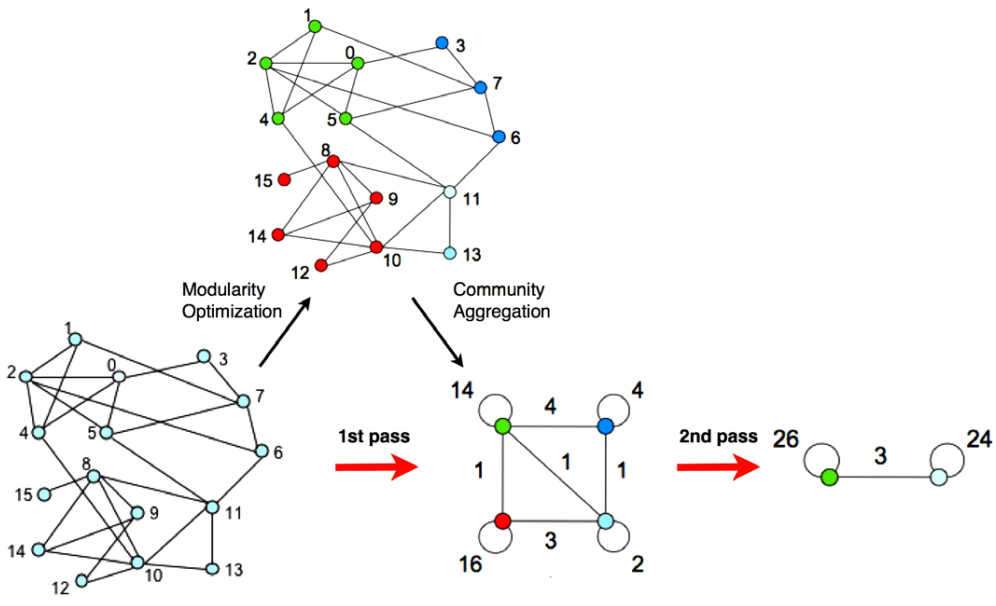
- An algorithm suited for this task: the greedy Louvain approach (ok for millions of nodes !) [Blondel et al., 2008]
- Principle: consider each node i in turn and put it in the community C of it with its neighbor j that maximally increases the modularity:

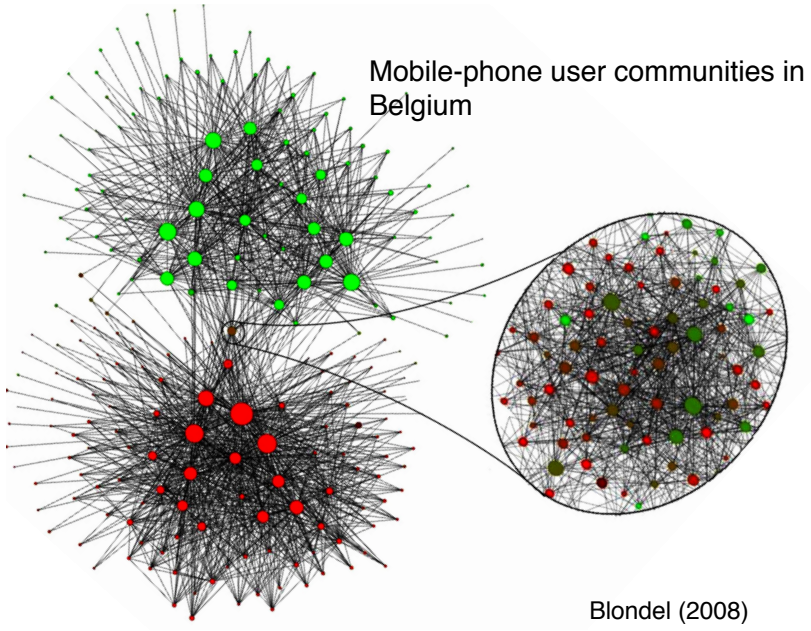
$$\Delta Q = \left[\frac{\sum_{in} + 2k_{i,\text{in}}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 \left(\frac{k_i}{2m}\right)^2\right]$$

- Each community is now a super-node
- Calculate the links and weights between super nodes as the sum of link weights connecting the merged nodes in the two super nodes
- Iterate (hence, it depends of the order on the nodes)

Communities from modularity for big data network

- Illustrating the algorithm

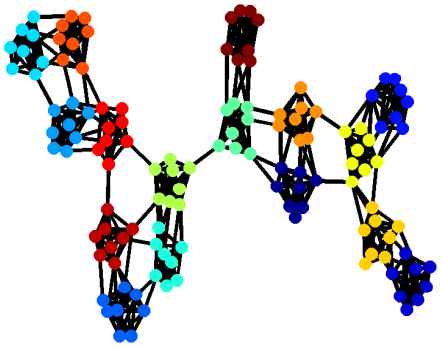




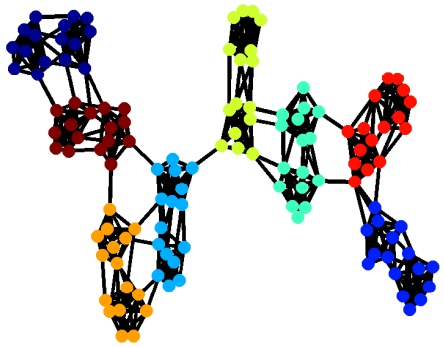
Blondel (2008)

Existence of multiscale community structure in a graph

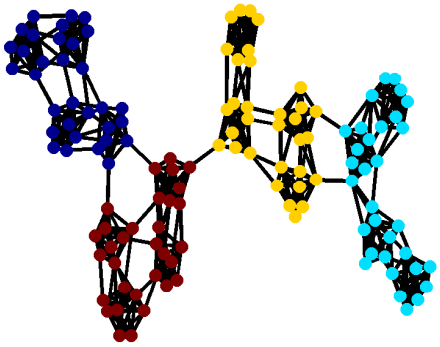
finest scale (16 com.):



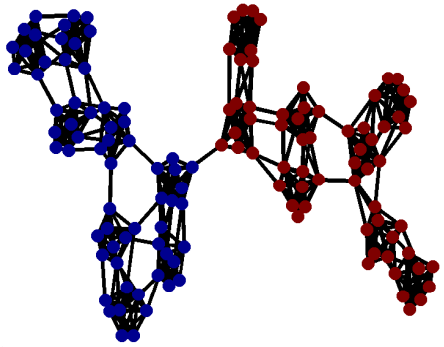
fine scale (8 com.):



coarser scale (4 com.):

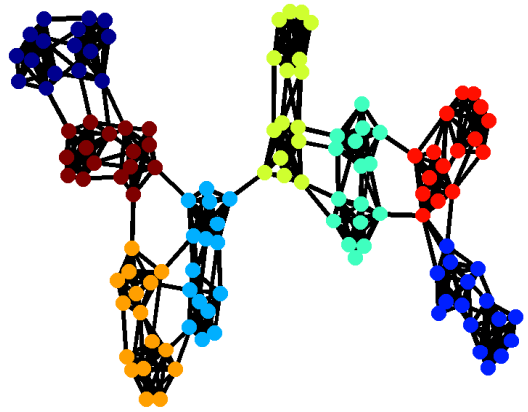


coarsest scale (2 com.):



Multiscale community structure in a graph

Classical community detection algorithms do not have this "scale-vision" of a graph. Modularity optimisation finds:

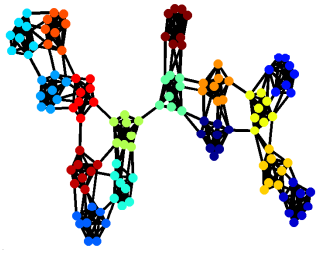


Where the modularity function reads:

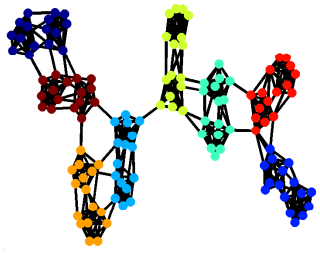
$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{d_i d_j}{2m} \right] \delta(c_i, c_j)$$

Multiscale community structure in a graph

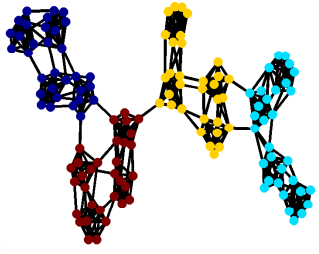
Q=0.80 :



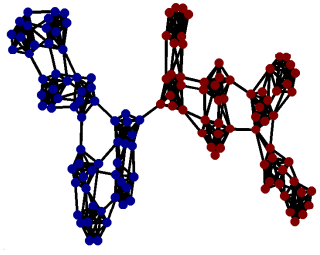
Q=0.83 :



Q=0.74 :



Q=0.50 :



All representations are correct but modularity optimisation favours one.

- Added to that: limit in resolution for modularity [Fortunato, Barthelemy, 2007]

Integrate a scale into modularity

- [Arenas et al., 2008] Remplace A by $A + rI$ in Q. Self-loops.
- [Reichardt and Bornholdt, 2006]

$$Q_\gamma = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \gamma \frac{d_i d_j}{2m} \right] \delta(c_i, c_j)$$

- Equivalent for regular graph if $\gamma = 1 + \frac{r}{\bar{d}}$.
- "Corrected Arenas modularity": use $A_{ij} + r \frac{d_i}{\bar{d}} \delta_{ij}$; equivalent to Reichardt and Bornholdt [Lambiotte, 2010]

Interpretation as random walks (part 2)

- Let us recall that $\mathbf{p}(t) = AD^{-1}\mathbf{p}(t-1) = W\mathbf{p}(t-1)$
- Equilibrium distribution: $\pi_i = \dfrac{d_i}{2m}$
- One step of random walk; the probability of staying in the same community is

$$R(1) = \sum_{ij} \left[\frac{A_{ij}}{d_j} \frac{d_j}{2m} - \frac{d_i d_j}{(2m)^2} \right] \delta(c_i, c_j) = Q$$

- Random walk after t steps (even if t continuous)

$$R(t) = \sum_{ij} \left[\left(e^{t(D^{-1}A - I)} \right)_{ij} \frac{d_j}{2m} - \frac{d_i d_j}{(2m)^2} \right] \frac{d_i d_j}{(2m)^2}$$

This is called **stability**.

Interpretation as random walks (part 2)

- If $t = 0$, $R(0) = 1 - \sum_{ij} \frac{d_i d_j}{(2m)^2} \frac{d_i d_j}{(2m)^2}$;
  best partition = single nodes
- If t small, $R(t) \simeq (1 - t)R(0) + tQ_c$;
  trade-off between single nodes and modularity; falls down in the Reichardt and Bornholdt formulation
- If $t = 1$, classical modularity
- If t large, the optimum partition is in 2 groups, as given by spectral clustering (Fiedler vector)

- In practice, optimization with same methods as for modularity
- It works well

Referenced works on multiscale communities

- Lambiotte, ”Multiscale modularity in complex networks“ [*WiOpt*, 2010]
- Schaub, Delvenne et al., ”Markov dynamics as a zooming lens for multiscale community detection: non clique-like communities and the field-of-view limit” [*PloS One*, 2012]
- Arenas et al., ”Analysis of the structure of complex networks at different resolution levels” [*New Journal of Physics*, 2008]
- Reichardt and Bornholdt, ”Statistical Mechanics of Community Detection” [*Physical Review E*, 2006]
- Mucha et al., ”Community Structure in Time-Dependent, Multiscale, and Multiplex Networks” [*Science*, 2010]

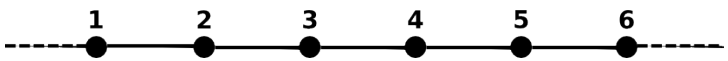
In the following: use wavelets on graphs to define a proper scale.

Relating the Laplacian of graphs to Signal Processing

Laplacian matrix

L or $\mathscr{L}$	laplacian matrix	$L = D - A$ or $\mathscr{L} = I - D^{-1/2} A D^{-1/2}$
(λ_i)	L's eigenvalues	$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{N-1}$
(χ_i)	L's eigenvectors	$L\chi_i = \lambda_i \chi_i$

A simple example: the straight line



$$\longleftrightarrow \quad L = \begin{pmatrix} \ddots & & & & & & \\ \ldots & -1 & 0 & 0 & 0 & 0 & \\ \ldots & 2 & -1 & 0 & 0 & 0 & \\ & -1 & 2 & -1 & 0 & 0 & \\ & 0 & -1 & 2 & -1 & 0 & \\ & 0 & 0 & -1 & 2 & -1 & \\ & 0 & 0 & 0 & -1 & 2 & \ldots \\ & 0 & 0 & 0 & 0 & -1 & \ldots \\ & & & & & & \ddots \end{pmatrix}$$

For this regular line graph, L is the 1-D classical laplacian operator (i.e. double derivative operator):
its eigenvectors are the Fourier vectors, and its eigenvalues the associated (squared) frequencies

Objective and Fundamental analogy

[Shuman, Vandergheynst et al., *IEEE SP Mag*, 2013]

Objective: Definition of a Fourier Transform adapted to graph signals

f : signal defined on V $\longleftrightarrow$ $\hat{f}$: Fourier transform of f

Fundamental analogy

On *any* graph, the eigenvectors χ_i of the Laplacian matrix L or $\mathscr{L}$ will be considered as the Fourier vectors, and its eigenvalues λ_i the associated (squared) frequencies.

- Works exactly for all regular graphs (+ Beltrami-Laplace)
- Conduct to natural generalizations of signal processing

The graph Fourier transform

- $\hat{f}$ is obtained from f's decomposition on the eigenvectors χ_i :

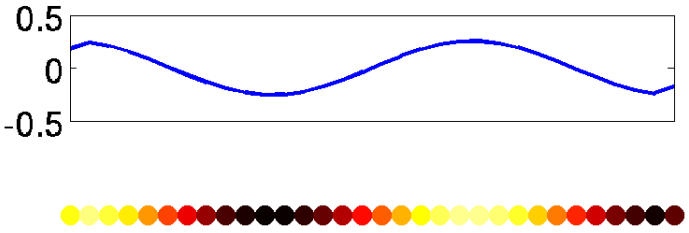
$$\hat{f} = \begin{pmatrix} < \chi_0, f > \\ < \chi_1, f > \\ < \chi_2, f > \\ \dots \\ < \chi_{N-1}, f > \end{pmatrix}$$

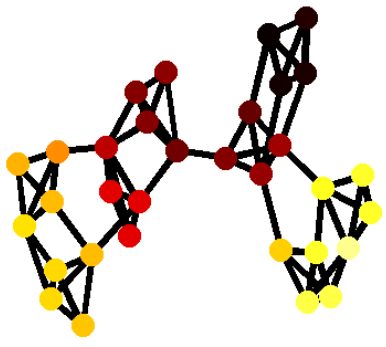
Define $\chi = (\chi_0 | \chi_1 | \dots | \chi_{N-1})$: $\boxed{\hat{f} = \chi^\top f}$

- Reciprocally, the inverse Fourier transform reads: $\boxed{f = \chi \hat{f}}$
- Parseval theorem: $\forall (g, h) \quad < g, h > = < \hat{g}, \hat{h} >$
- Filtering: apply $g(\lambda_i)$ in the Fourier domain on the $\hat{f}(i)$.

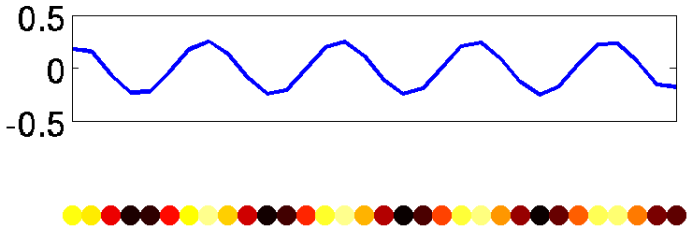
Fourier modes: examples in 1D and in graphs

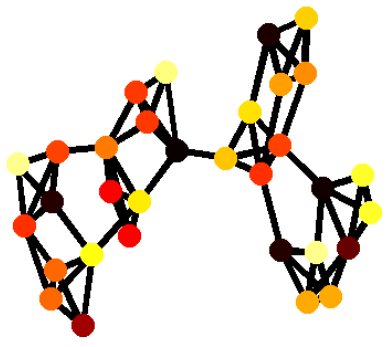
LOW FREQUENCY:





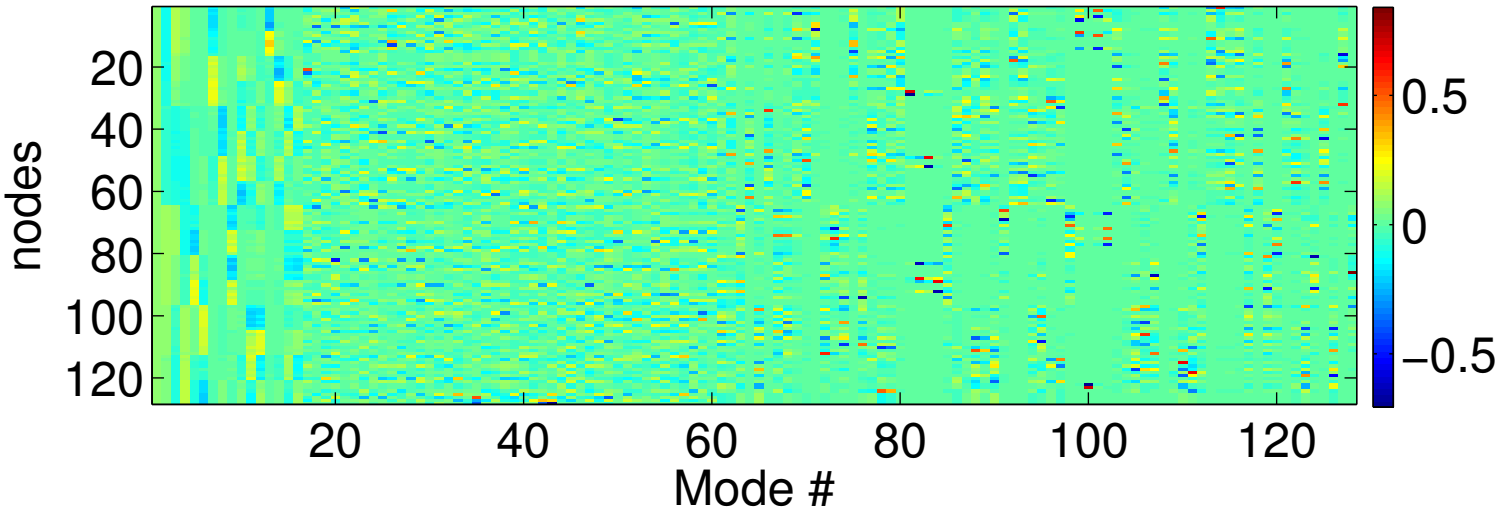
HIGH FREQUENCY:

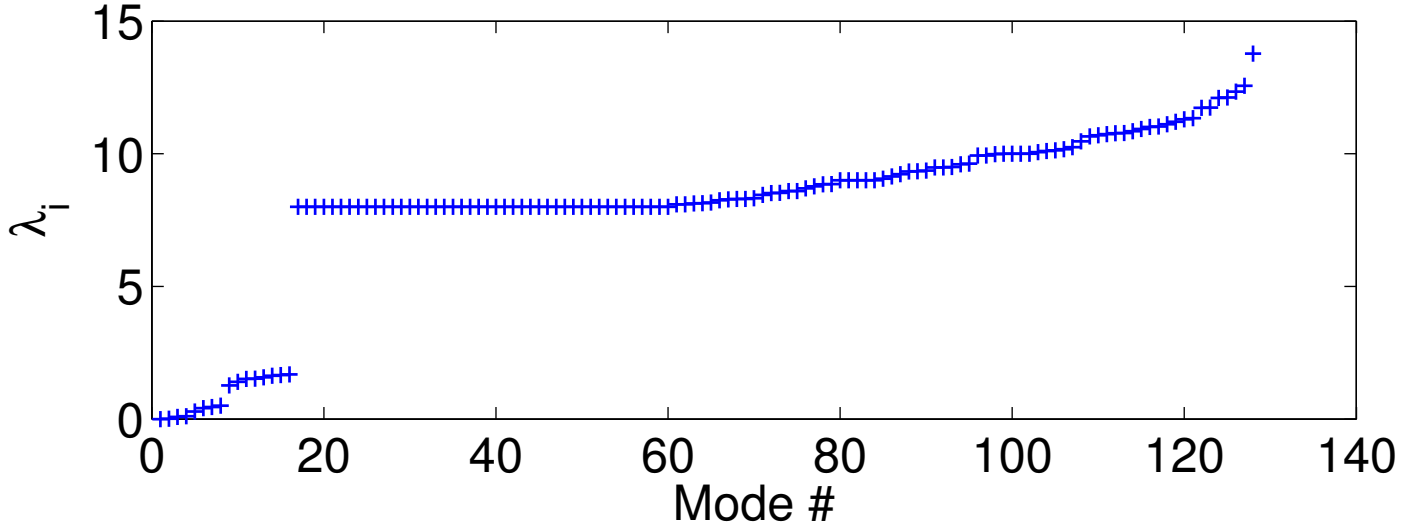




- Alternative Fourier transform: use the adjacency matrix A [Sandryhaila, Moura, *IEEE TSP*, 2013]

Spectral analysis: the χ_i and λ_i of a multiscale toy graph

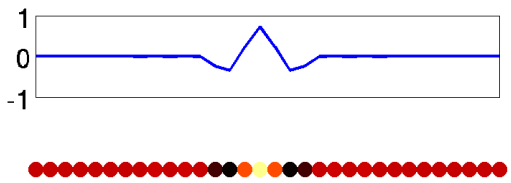




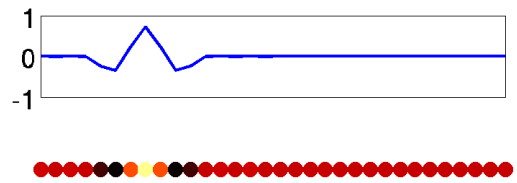
Spectral Graph Wavelets

[Hammond et al., ACHA 2011]

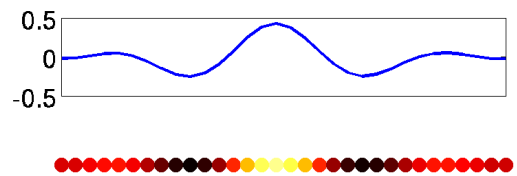
- Fourier is a global analysis. Fourier modes (eigenvectors of the laplacian) are used in classical spectral clustering, but do not enable a jointly local and scale dependent analysis.
- For that classical signal processing (or harmonic analysis) teach us that we need **wavelets**.
- Wavelets : local functions that act as well as a filter around a chosen scale.
  A wavelet:



– Translated:



– Scaled



- Classical wavelets $\xrightarrow{\textit{by analogy}}$ Graph wavelets

Recall the classical wavelets

- Usual definition: $\psi_{s,a}(x) = \frac{1}{s}\psi\left(\frac{x-a}{s}\right)$
- Equivalently, in the Fourier domain:

$$\psi_{s,a}(x) = \int_{-\infty}^{\infty} \hat{\delta}_a(\omega)\hat{\psi}(s\omega)\exp^{i\omega x} d\omega$$

  - In this definition, $\hat{\psi}(s\omega)$ acts as a filter bank defined by scaling by a factor s a *filter kernel function* defined in the Fourier space: $\hat{\psi}(\omega)$
  - The filter kernel function $\hat{\psi}(\omega)$ is necessarily a *bandpass filter* with:
    - $\hat{\psi}(0) = 0$: the mean of ψ is by definition null
    - $\lim_{\omega \to +\infty} \hat{\psi}(\omega) = 0$: the norm of ψ is by definition finite

  (Note: the actual condition is the admissibility property)

Classical wavelets $\xrightarrow{\textit{by analogy}}$ Graph wavelets

[Hammond et al. ACHA '11]

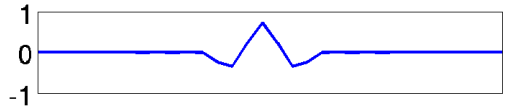
	Classical (continuous) world	Graph world
Real domain	x	node a
Fourier domain	ω	eigenvalues λ_i
Filter kernel	$\hat{\psi}(\omega)$	$g(\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}$
Filter bank	$\hat{\psi}(s\omega)$	$g(s\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}_{\boldsymbol{s}}$
Fourier modes	$\exp^{-i\omega x}$	eigenvectors χ_i
Fourier transf. of f	$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x) \exp^{-i\omega x} dx$	$\hat{f} = \boldsymbol{\chi}^\top f$

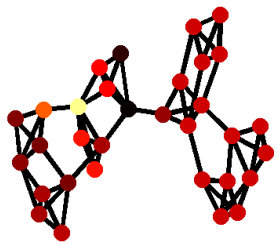
The wavelet at scale s centered around node a is given by:

$$\psi_{s,a}(x) = \int_{-\infty}^{\infty} \hat{\delta}_a(\omega)\hat{\psi}(s\omega) \exp^{i\omega x} d\omega \longrightarrow \boxed{\psi_{s,a} = \boldsymbol{\chi}\, \hat{\boldsymbol{G}}_{\boldsymbol{s}} \hat{\delta}_a = \boldsymbol{\chi}\, \hat{\boldsymbol{G}}_{\boldsymbol{s}}\, \boldsymbol{\chi}^\top \delta_a}$$

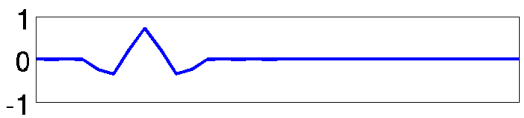
Examples of graph wavelets

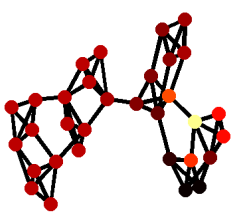
A WAVELET:



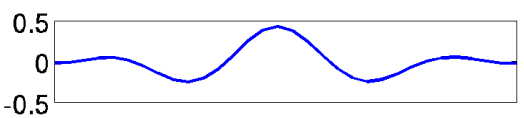


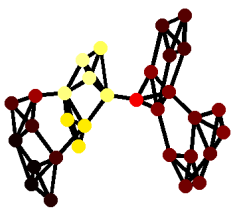
TRANSLATING:



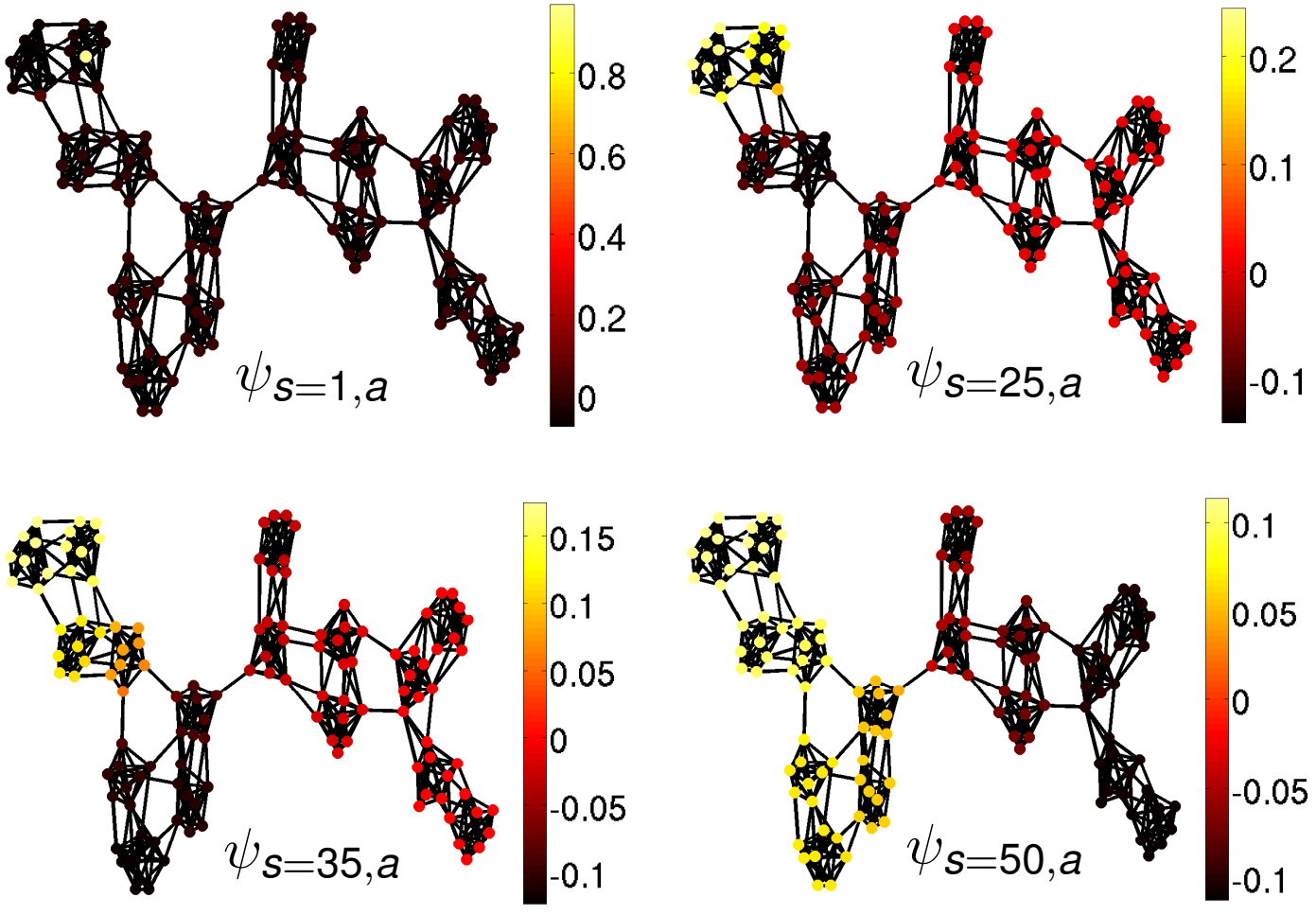


SCALING:





Examples of wavelets: they encode the local topology



Example of wavelet filters

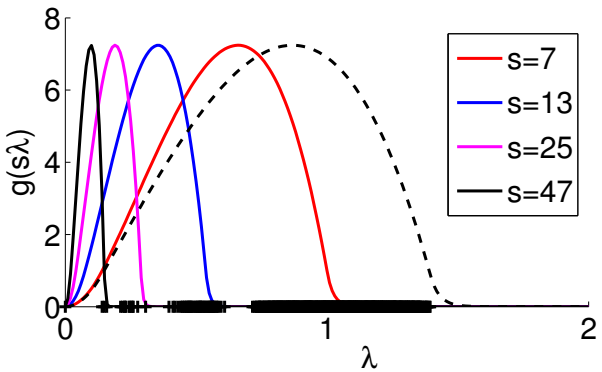
- More precisely, we will use the following kernel:

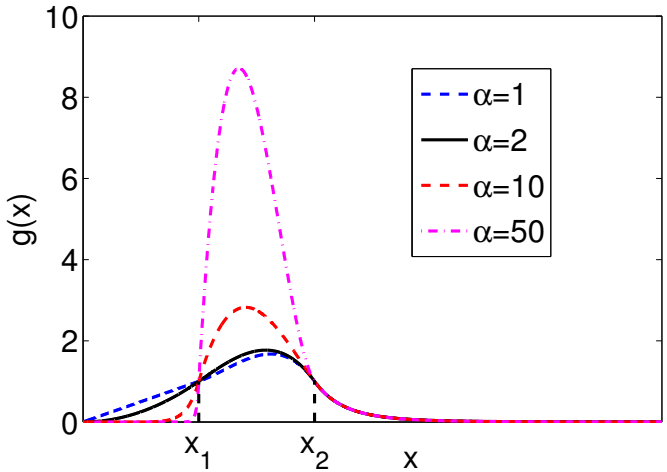
$$g(x;\alpha,\beta,x_1,x_2) = \begin{cases} x_1^{-\alpha}\, x^{\alpha} & \text{for} \quad x < x_1 \\ p(x) & \text{for} \quad x_1 \le x \le x_2 \\ x_2^{\beta}\, x^{-\beta} & \text{for} \quad x > x_2. \end{cases}$$

- To emphasize χ_1, the parameters are:

$$s_{min} = \frac{1}{\lambda_2}, \quad x_2 = \frac{1}{\lambda_2}, \quad s_{max} = \frac{1}{\lambda_2^2}, \quad x_1 = 1, \quad \beta = 1/\log_{10}\left(\frac{\lambda_3}{\lambda_2}\right)$$

- This leads to: (choice $\alpha = 2$)





Purpose of the next slides

Develop a scale dependent community mining tool using concepts from graph signal processing.
Why ? For joint processing of graph signals and networks.

General Ideas

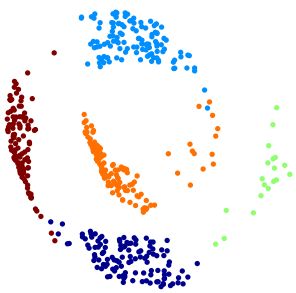
- Take advantage of local topological information encoded in Graph Wavelets.
  Wavelet = ego-centered vision from a node
- Group together nodes whose local environments are similar at the description scale
- This will naturally offer a multiscale vision of communities

Examples of networks with communities at different scales

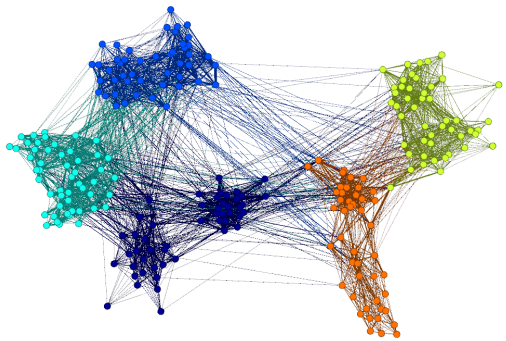
Three examples of community detections:

- (A) A complex sensor network (non-uniform swiss roll topology)
- (B) A contact network in a primary school [Stehle '11]
- (C) A hierarchical graph benchmark [Sales-Pardo '07]

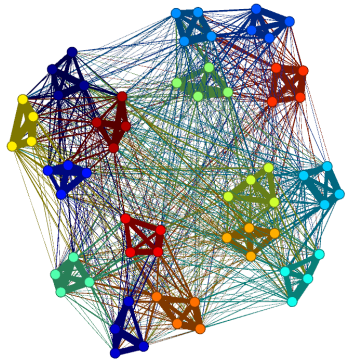
A



B



C



A new method for multiscale community detection

[N. Tremblay, P. Borgnat, 2013]

The problem of community mining is considered as a problem of clustering. We then need to decide upon:

1. feature vectors for each node
2. a distance to measure two given vectors' closeness
3. a clustering algorithm to separate nodes in clusters

The method uses:

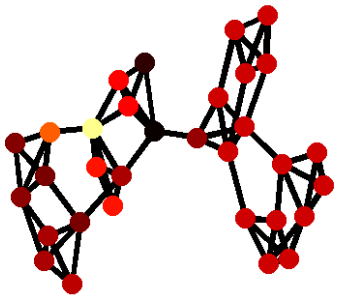
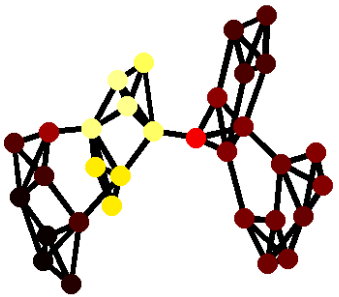
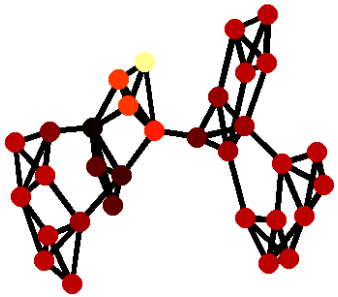
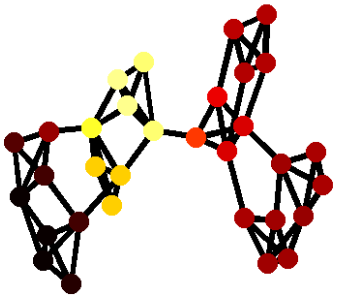
1. wavelets (resp. scaling functions) as feature vectors
2. the correlation distance
3. the complete linkage clustering algorithm

1) Wavelets as features

Each node a has feature vector $\psi_{s,a}$.

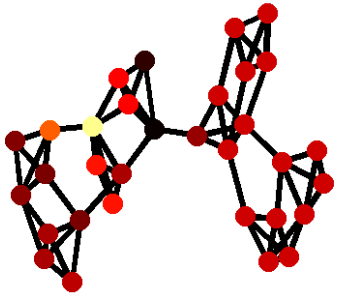
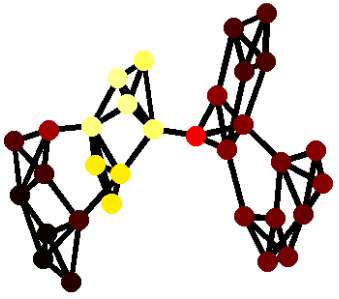
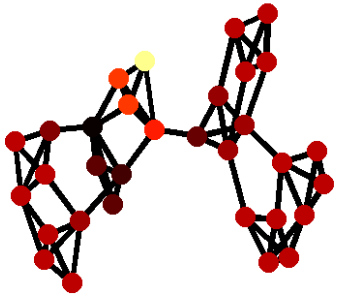
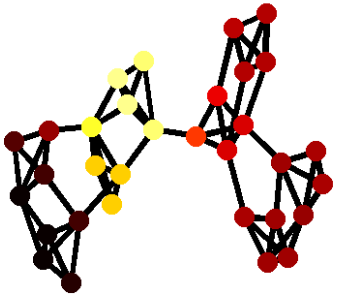
Globally, one will need Ψ_s, all wavelets at a given scale s, i.e.

$$\Psi_s = \left(\psi_{s,1}|\psi_{s,2}|\dots|\psi_{s,N}\right) = \chi \boldsymbol{G}_s \chi^\top.$$

	AT SMALL SCALE:	AT LARGE SCALE:
NODE A:		
NODE B:		

2) Correlation distances

$$\boldsymbol{D}_s(a,b) = 1 - \frac{\boldsymbol{\psi}_{s,a}^{\top}\boldsymbol{\psi}_{s,b}}{||\boldsymbol{\psi}_{s,a}||_2 ||\boldsymbol{\psi}_{s,b}||_2}.$$

NODE A:		
NODE B:		
CORR. COEF.:	-0.50	0.97

3) Complete linkage clustering and dendrogram

- Bottom to top hierarchical algorithm:
  start with as many clusters as nodes and work the way up to fewer clusters (by linking subclusters together) until reaching one global cluster.

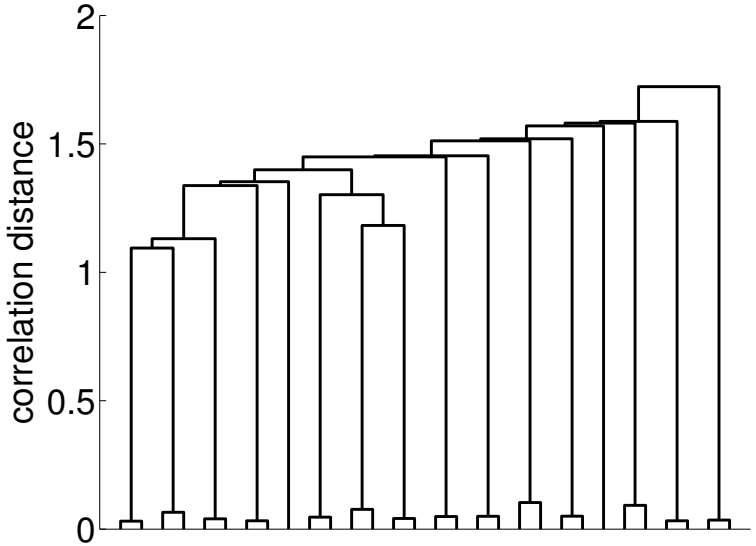
- Computation of the distance between two subclusters:
  the average distance between all pairs of nodes, taking one from each cluster

- Output: a dendrogram

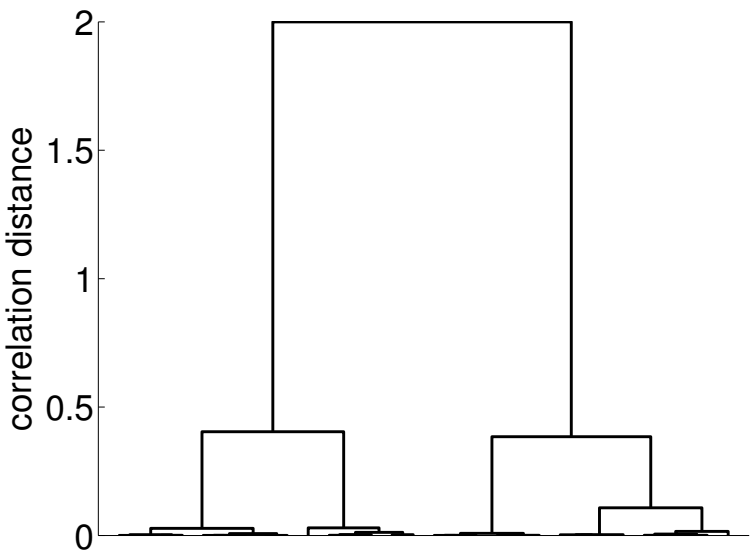
Dendrogram cut at maximal gap

To avoid the cumbersome multiscale modularity optimization, we can simply cut the dendrogram at its maximal gap.

At small scale:



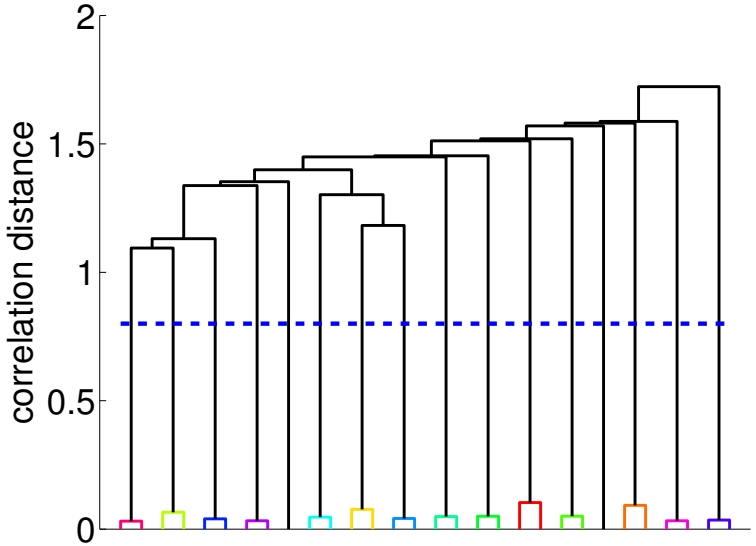
At large scale:



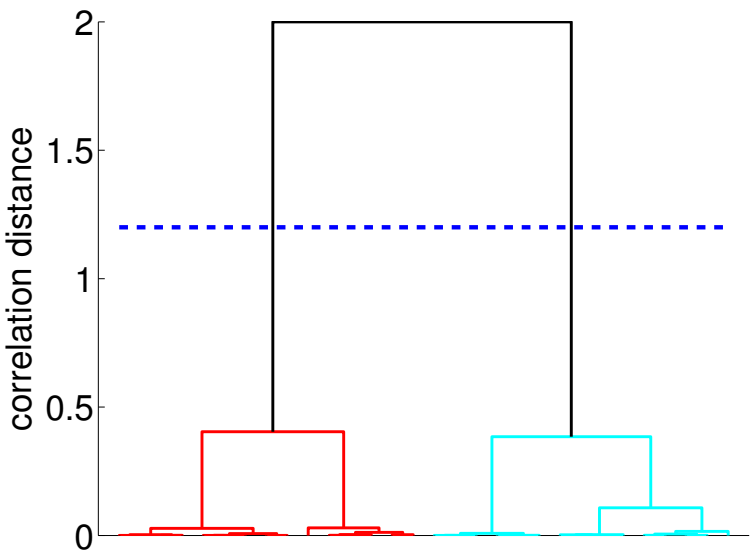
Dendrogram cut at maximal gap

To avoid the cumbersome multiscale modularity optimization, we can simply cut the dendrogram at its maximal gap.

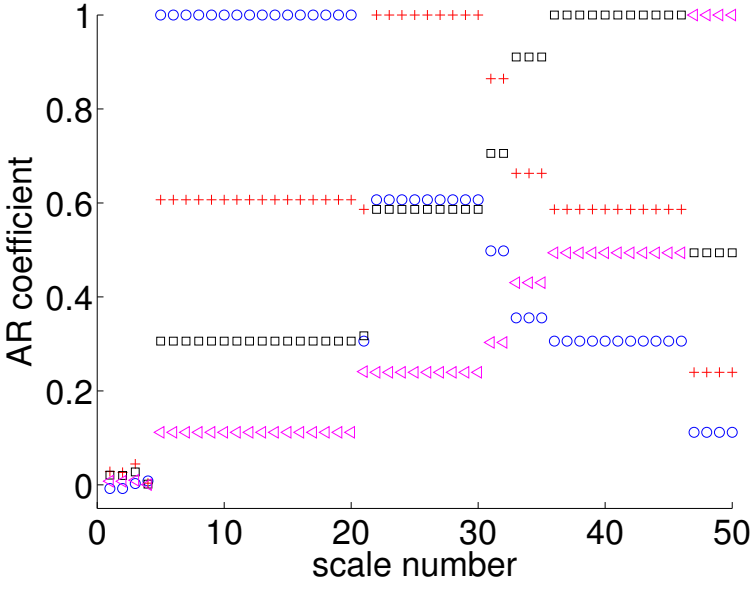
At small scale:

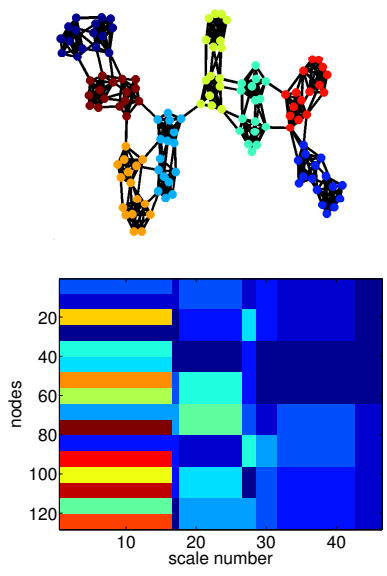


At large scale:



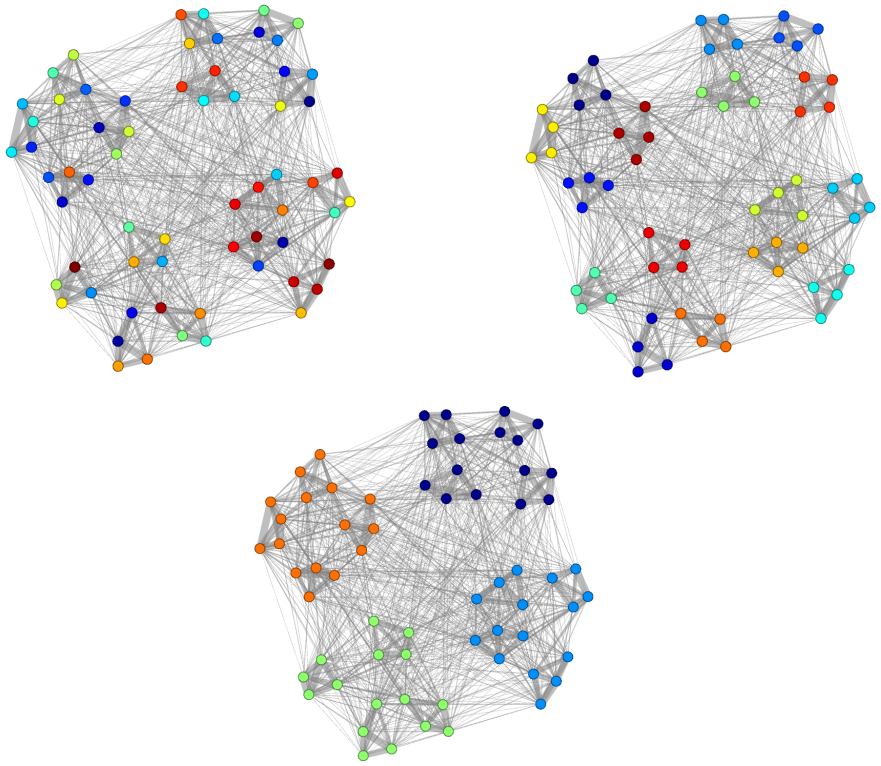
Dendrogram cut at maximal gap



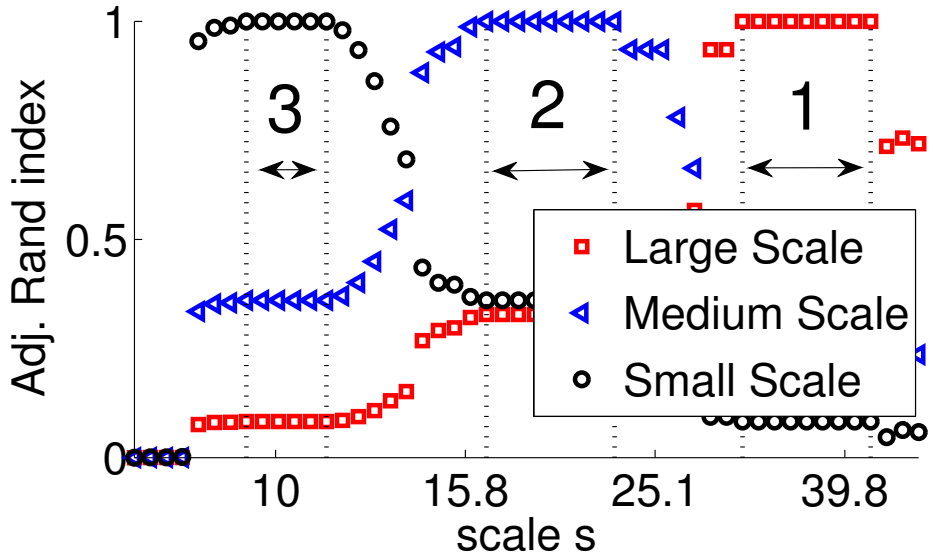


The Sales-Pardo benchmark

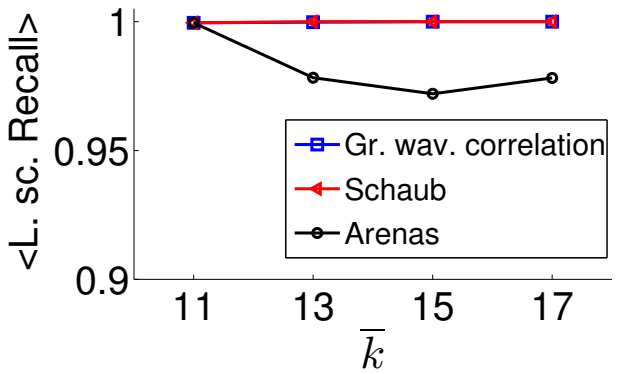
- Three community structures nested in one another
- Parameters:
  - sizes of the communities ($N = 640$)
  - ρ tunes how well separated the different scales are
  - $\bar{k}$ is the average degree; the sparser is the graph, the harder it is to recover the communities.

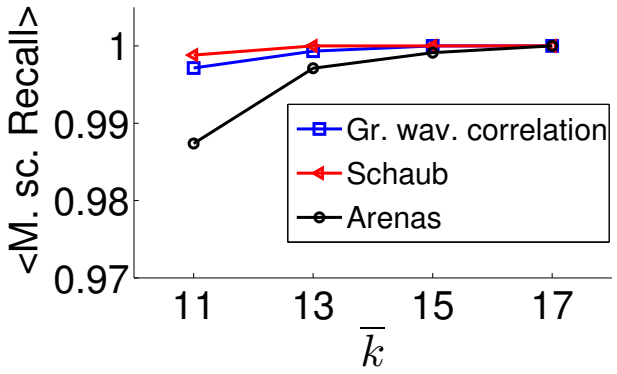


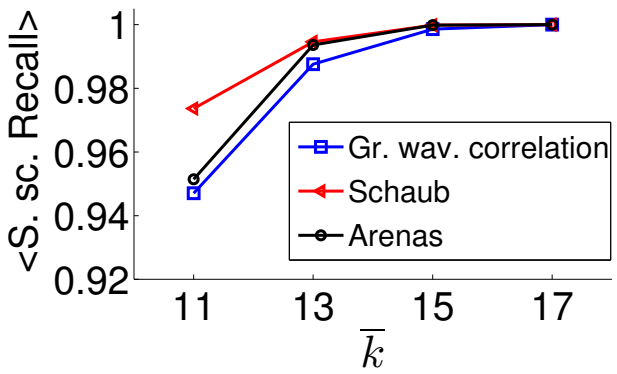
Results on the Sales-Pardo benchmark



Results on the Sales-Pardo benchmark







The case of larger networks

- Limit of the method: computation of the $N \times N$ matrix of the wavelets Ψ_s.
- Improvement: use of random features.
- Let $\boldsymbol{r} \in \mathbb{R}^N$ be a random vector on the nodes of the graph, composed of N independent normal random variables of zero mean and finite variance σ^2.
- Define the feature $f_{s,a} \in \mathbb{R}$ at scale s associated to node a as

$$f_{s,a} = \psi_{s,a}^\top \boldsymbol{r} = \sum_{k=1}^{N} \psi_{s,a}(k) r(k).$$

The case of larger networks

- Let us define the correlation between features

$$\mathrm{Cor}(f_{s,a}, f_{s,b}) = \frac{\mathbb{E}((f_{s,a} - \mathbb{E}(f_{s,a}))(f_{s,b} - \mathbb{E}(f_{s,b})))}{\sqrt{\mathrm{Var}(f_{s,a})\mathrm{Var}(f_{s,b})}}.$$

- It is easy to show that:

$$\mathrm{Cor}(f_{s,a}, f_{s,b}) = \frac{\boldsymbol{\psi}_{s,a}^{\top}\boldsymbol{\psi}_{s,b}}{||\boldsymbol{\psi}_{s,a}||_2 ||\boldsymbol{\psi}_{s,b}||_2}.$$

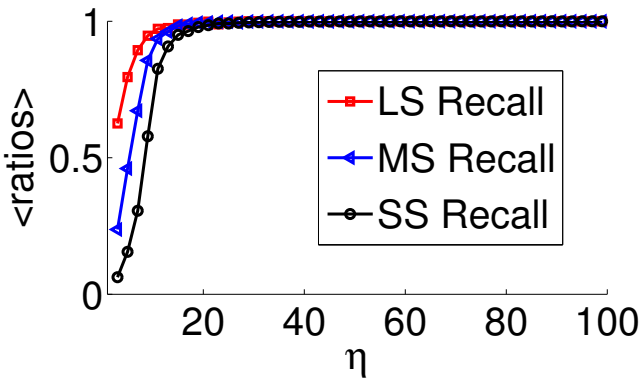
- Therefore, the sample correlation estimator $\hat{C}_{ab,\eta}$ satisfies:

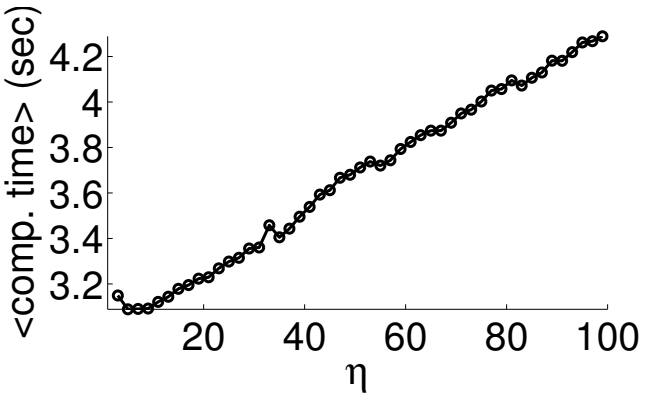
$$\lim_{\eta \to +\infty} \hat{C}_{ab,\eta} = \frac{\boldsymbol{\psi}_{s,a}^{\top}\boldsymbol{\psi}_{s,b}}{||\boldsymbol{\psi}_{s,a}||_2 ||\boldsymbol{\psi}_{s,b}||_2} = 1 - \boldsymbol{D}_s(a,b).$$

- This leads to a faster algorithm.

Results on the Sales-Pardo benchmark

- As a function of η, the number of random vectors used





Stability of the communities

- Not all partitions are relevant: only those stable enough convey information about the network
- Modularity optimization is based on an objective function: the higher the modularity is, the better the partition is
  $\rightarrow$ discard partitions with low modularity (threshold ?)
- For multiscale: Lambiotte's approach to stability:
  Create B resampled graphs by randomly adding $\pm p\%$ (typically $p = 10$) to the weight of each link and computing the corresponding B sets of partitions $\{P_s^b\}_{b\in[1,B],s\in\mathcal{S}}$.
  Then, stability:

$$\gamma_r(s) = \frac{2}{B(B-1)} \sum_{(b,c)\in[1,B]^2, b\neq c} \texttt{ari}(P_s^b, P_s^c), \qquad (1)$$

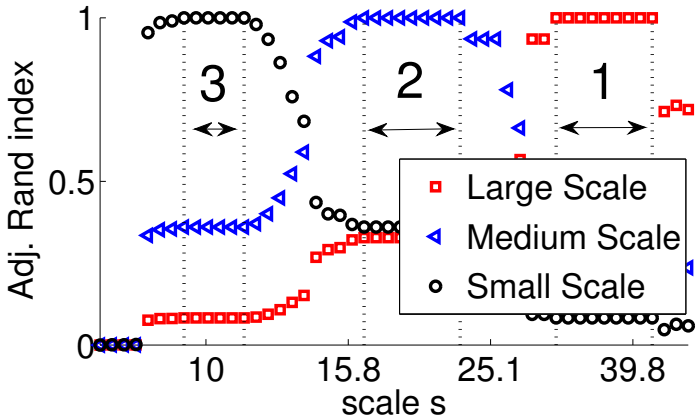
  $\rightarrow$ keep local maximum of stability

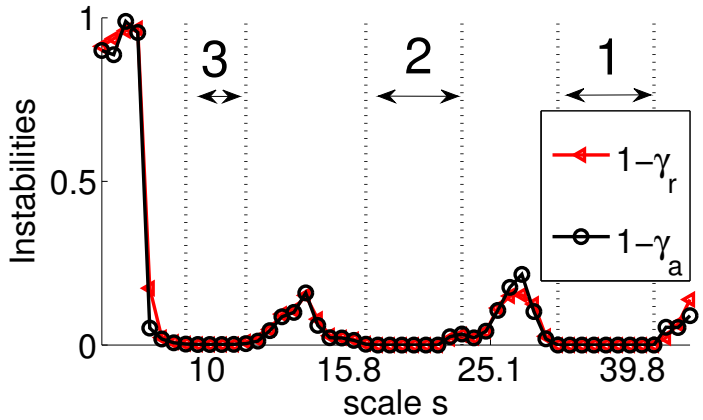
Stochastic Stability of the communities

- A new approach for the stochastic algorithm with wavelets: Consider J sets of η random signals and compute the associated sets of partitions $\{P_s^j\}_{j\in[1,J],s\in\mathcal{S}}$. Let stability be:

$$\gamma_a(s) = \frac{2}{J(J-1)} \sum_{(i,j)\in[1,J]^2, i\neq j} \texttt{ari}(P_s^i, P_s^j). \qquad (2)$$

Results with stabilities on the Sales-Pardo benchmark

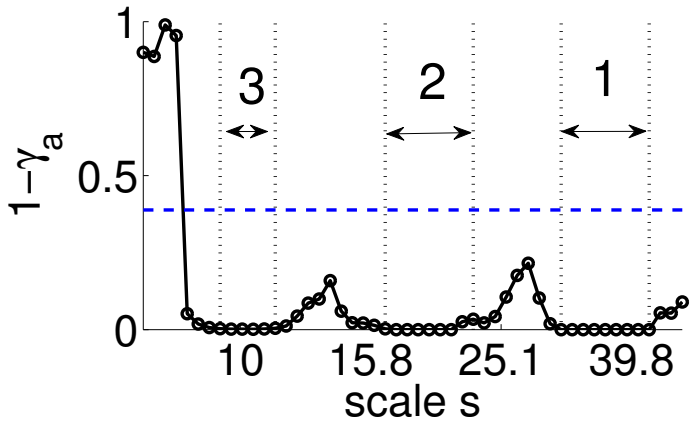




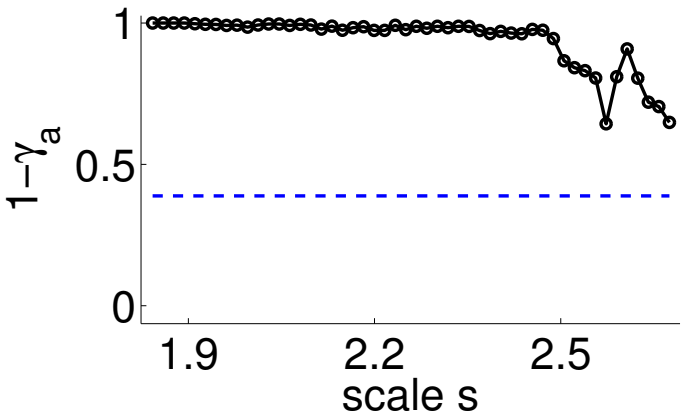
In addition: statistical test of relevance of the communities

- It is possible to design a data-driven test on γ_a (not explained here).
- Result: threshold for $1 - \gamma_a$ above which the partition in communities is irrelevant.

Sales-Pardo graph



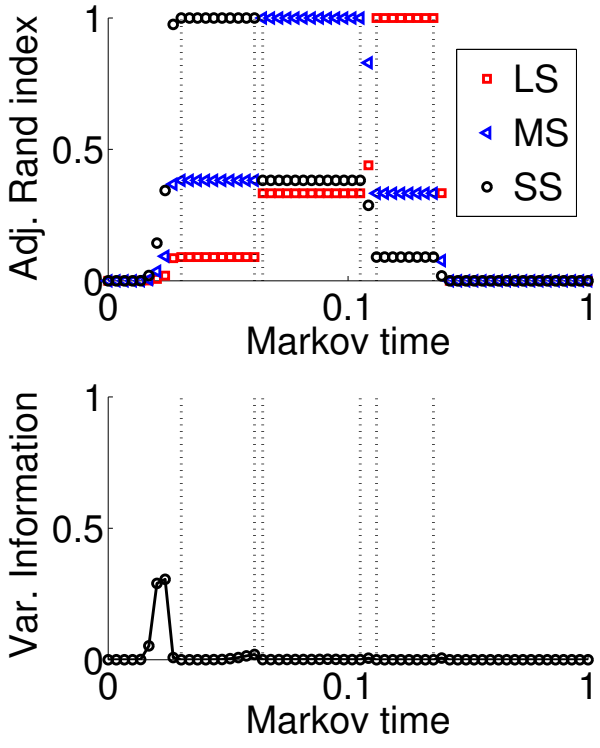
Chung-Lu graph



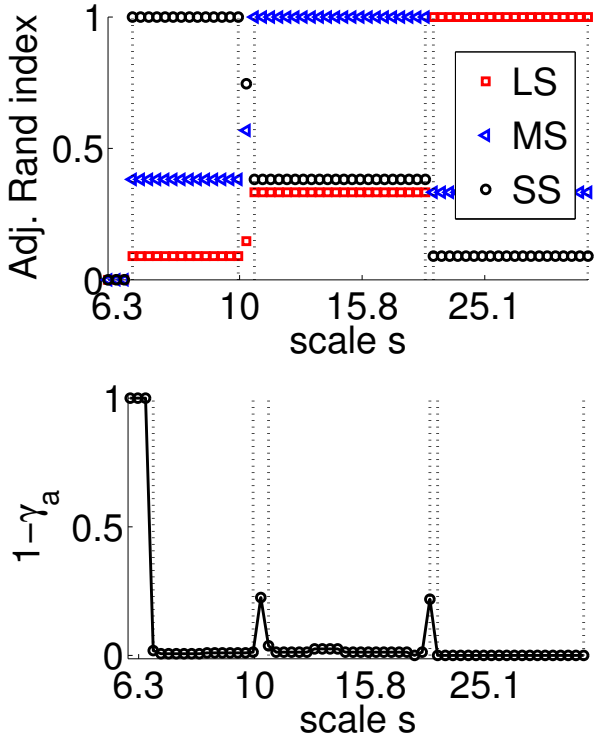
Comparison on larger Sales-Pardo graphs

$N = 6400$ nodes

Schaub-Delvenne

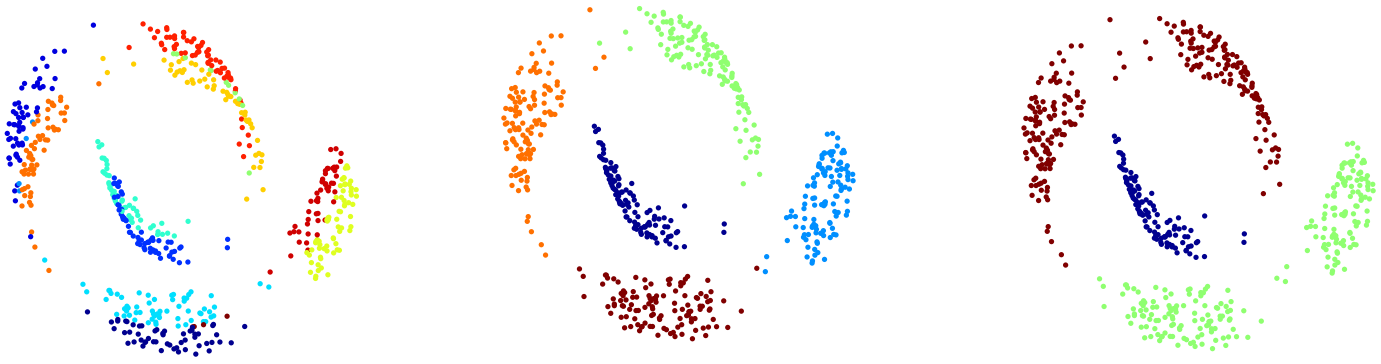


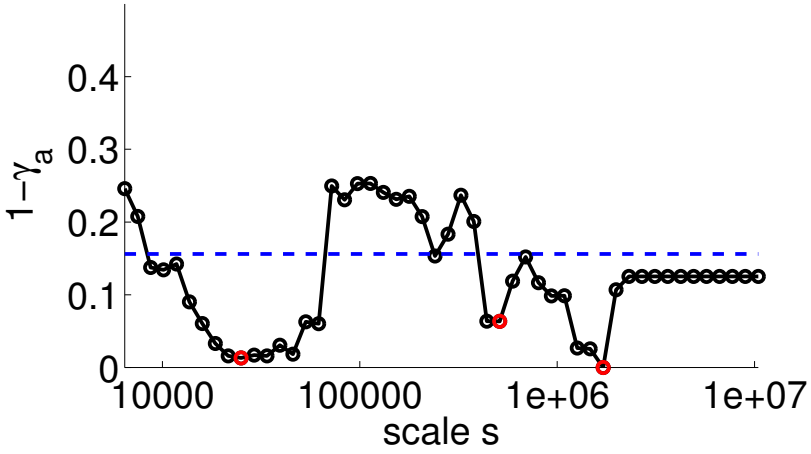
Wavelets



Sensor network on the swiss roll manifold

- Three scale ranges of relevant community structure



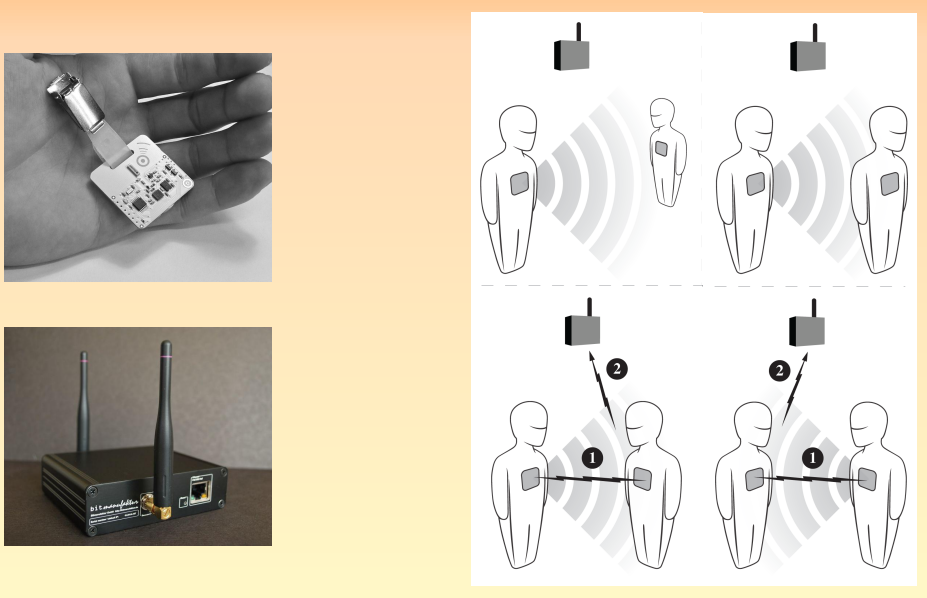


The dynamic social network of a primary school

Collaboration with A. Barrat (CPT Marseille), C. Cattuto (ISI, Turin)

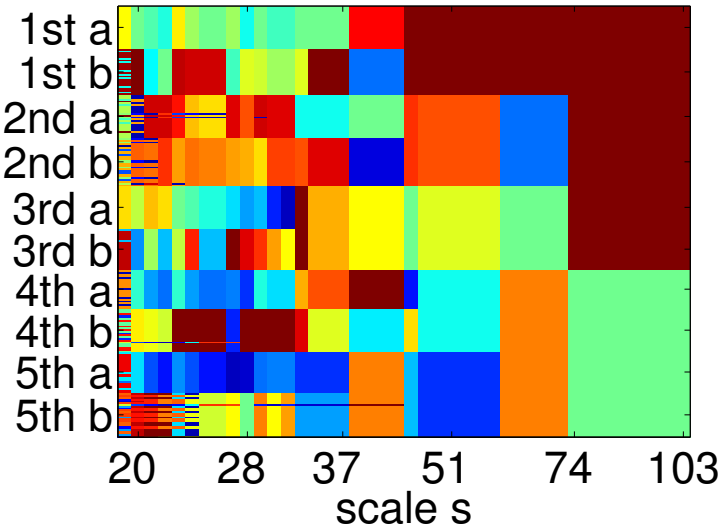
Sociopatterns project

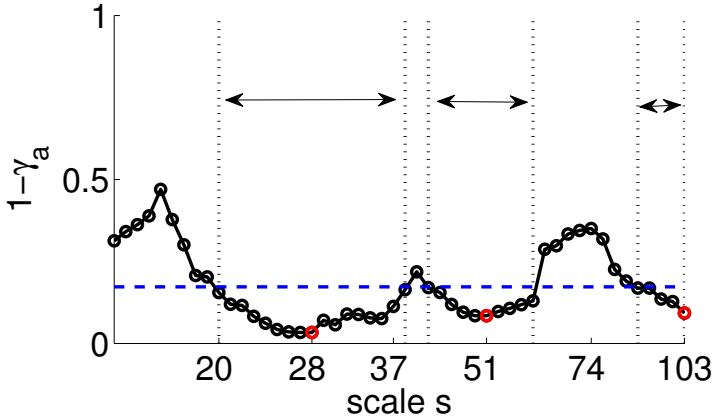
- Acquisition of face-to-face human contacts (resolved in time) using active RFID tags and + fixed antenna



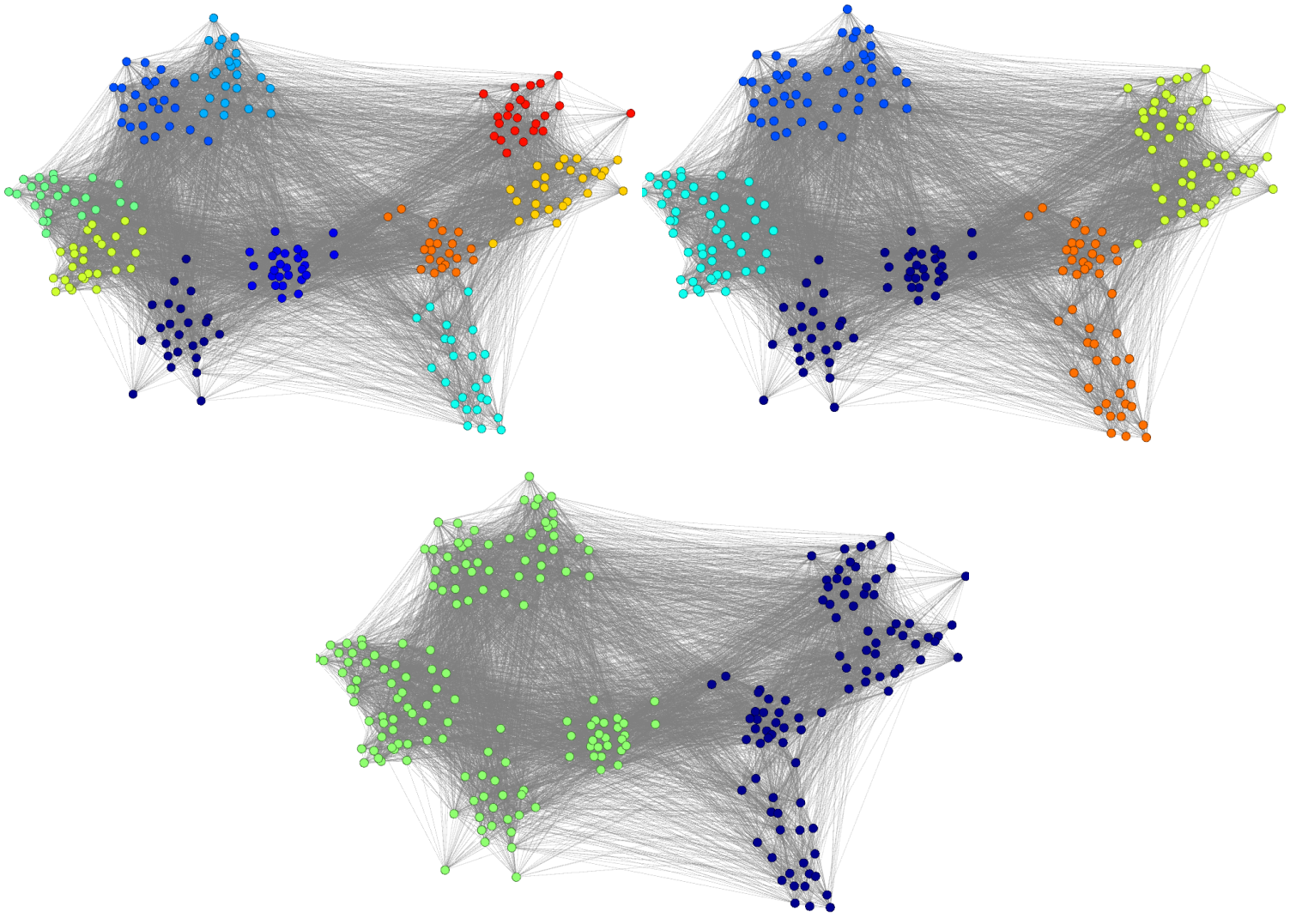
- Interest: social studies, spreading processes (of information, of epidemic,...), contact dynamics,...
- Time for a movie!

Multi-scale Communities in Primary School





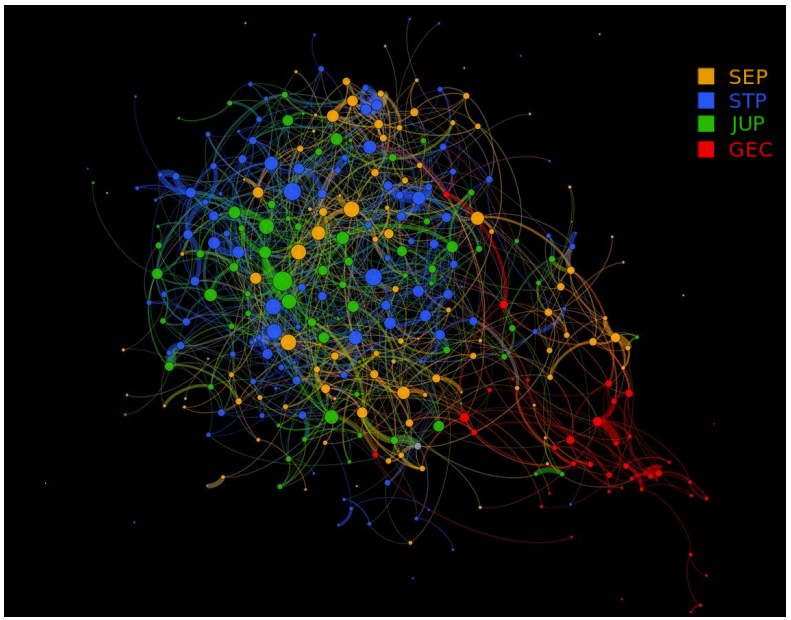
Multi-scale Communities in Primary School



Tests statistiques sur des propriétés de groupe

[N. Tremblay, A. Barrat, et al., soumis, 2012]

- Question sur les données SLC : existe-t-il 2 groupes différents (communautés) (GEC / DPP) ? Se mélangent-ils bien ?
- Difficulté : une seule réalisation.
  Comment caractériser le comportement normal ?
- Méthode proposée : **bootstrap** pour test statistique des propriétés de groupes dans un graphe



Tests statistiques sur des propriétés de groupe

Bootstrap par tirage de graphes contraints

Objectif: test statistique sur les propriétés d'un groupe X^0 qui est un sous-graphes du graphe de contact, confronté à une hypothèse nulle H_0 : "comportement normal"

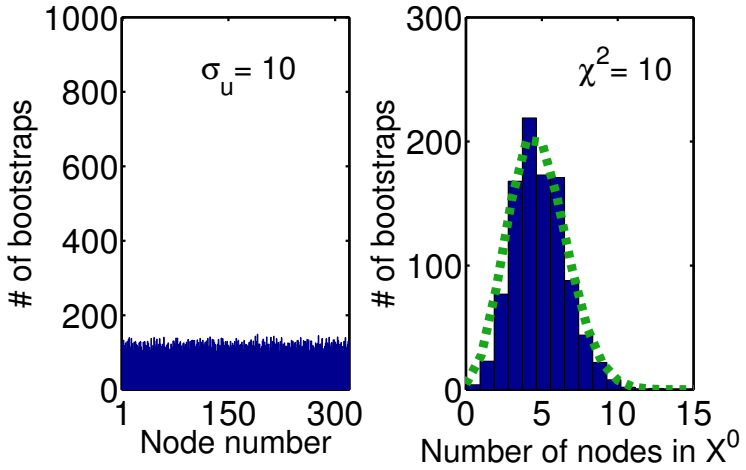
- Décider d'un ensemble d'observables $\mathcal{O}$ qui peuvent dire en quoi un groupe apparaît comme "normal"
- Traduire H_0 comme étant des contraintes sur les sous-graphes "normaux"
  Exemples : même cardinalité que X^0 ; mêmes interactions dans le groupe ; etc.
- Construire un ensemble de bootstrap de sous-graphes contraints par H_0, tirage par recuit simulé avec remise
- Etablir les statistiques attendues sous H_0 grâce à l'ensemble de bootstrap
- Décider si H_0 peut être rejeté ou non pour le groupe d'intérêt X^0

Tests statistiques sur des propriétés de groupe

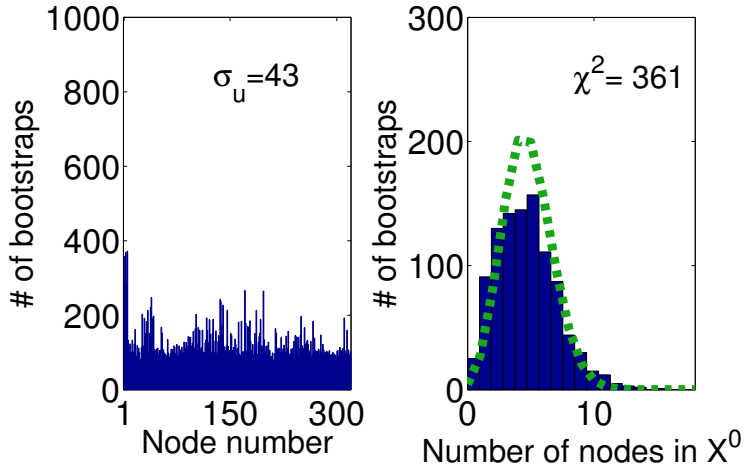
Bootstrap par tirage de graphes contraints

- Variable de contrôle de la taille de l'ensemble de bootstrap
  - σ_u est l'écart type du nombre fois qu'un nœud donné est pris par le bootstrap
  - χ^2 est la distance entre les distributions attendues et empiriques du nombre de nœud pris dans X^0 dans chaque échantillon bootstrap.

H_0 même cardinal



H_0 même modularité

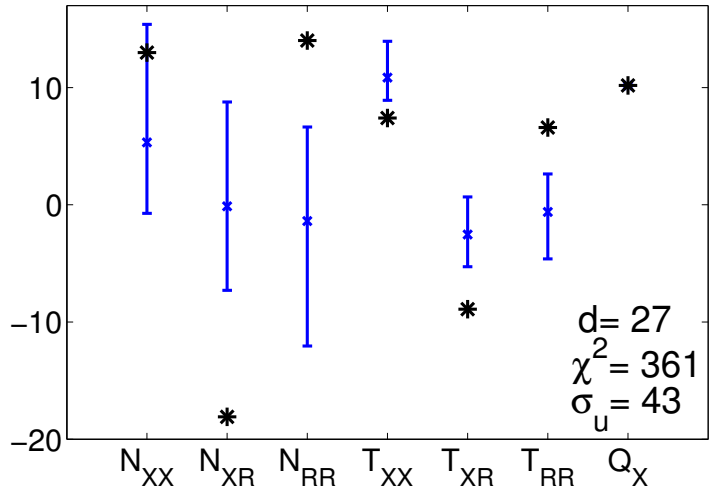


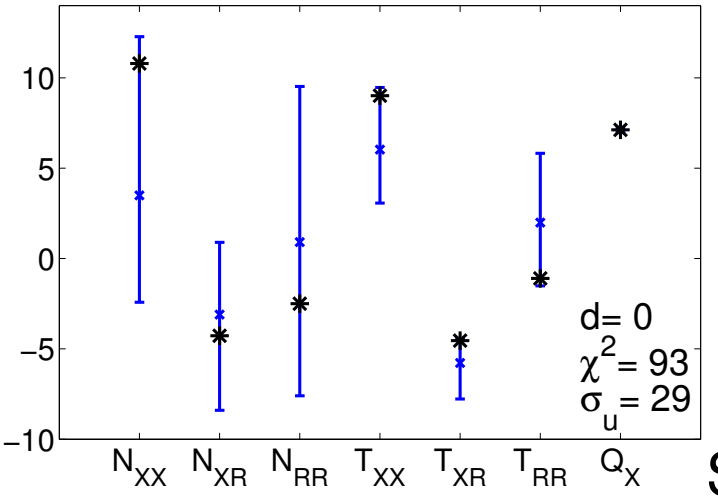
Tests statistiques sur des propriétés de groupe

Bootstrap par tirage de graphes contraints

- Test appliqué sur les groupes GEC et STP (étudiants de DPP) : *d* est la distance à l'intervalle de confiance à 5%

GEC





STP

- Conclusion: GEC ne se comporte pas comme les groupes "normaux", contrairement aux autres groupes identifiables $\rightarrow$ mélange faible entre les conférences
- Conclusion plus affinée : données filtrées par lieux. GEC et DPP se mélangent surtout dans les espaces communs !

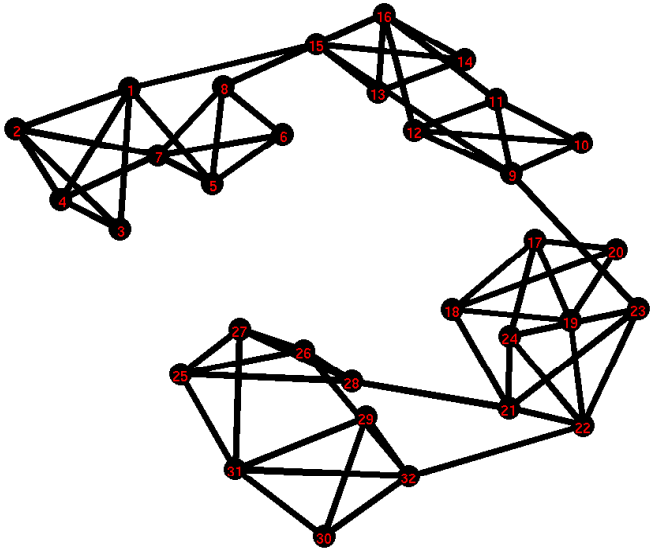
Conclusion (partie 2)

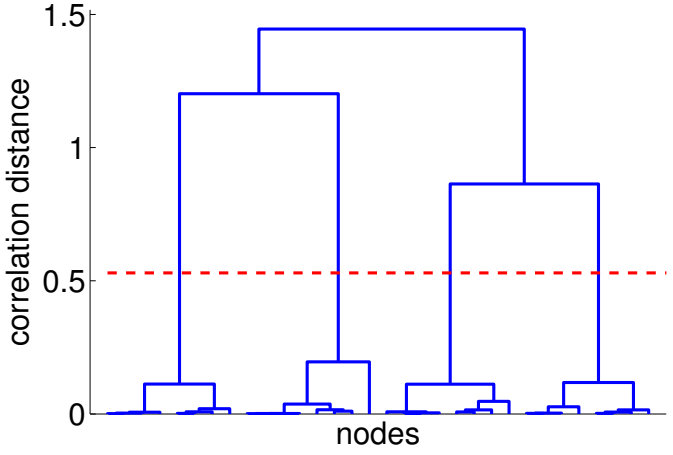
- Wavelet $\psi_{s,a}$ gives an "egocentered view" of the network seen from node *a* at scale *s*
- Correlation between these different views gives us a distance between nodes at scale *s*
- This enables multi-scale clustering of nodes in communities
- Notion of stability and of statistical detection of relevance of groups

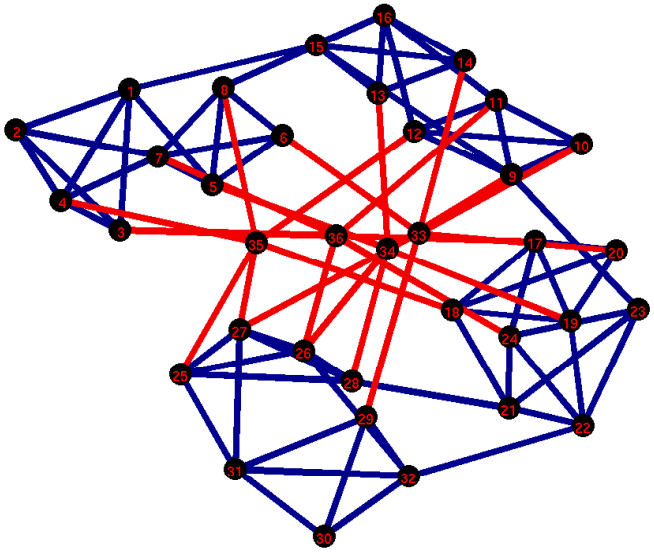
http://perso.ens-lyon.fr/pierre.borgnat

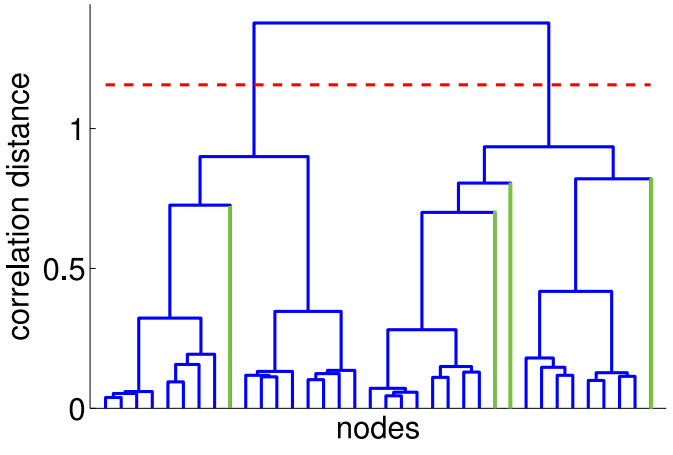
Acknowledgements: thanks to Nicolas Tremblay and Marton Karsai for the borrowing many of their figures or slides.

Dendrogram cut at maximal gap: non robust to outliers

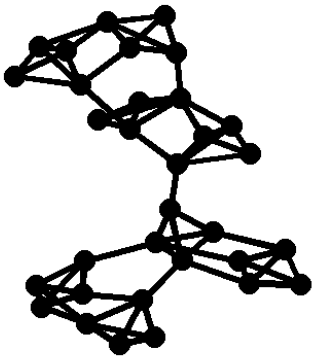


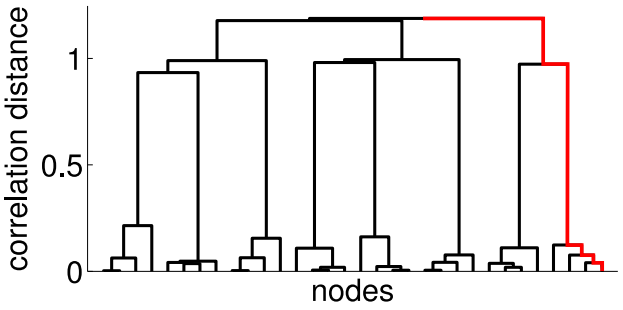


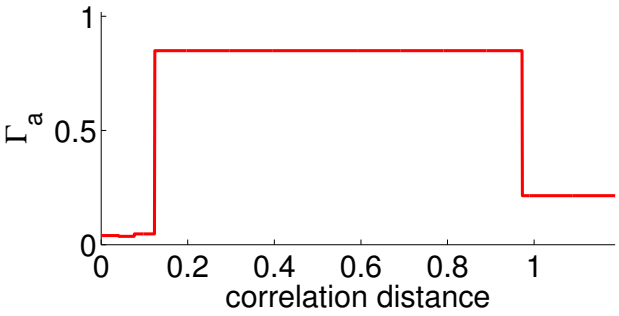




Dendrogram cut at maximal average gap

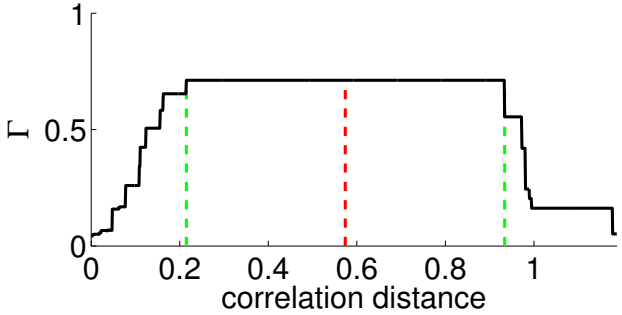


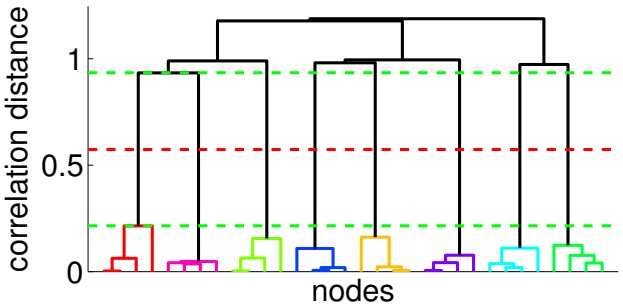




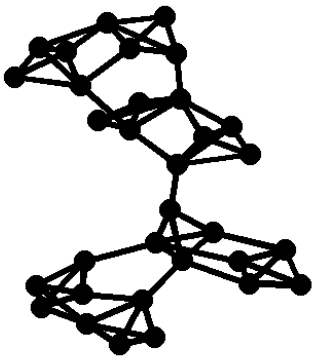
$$\Gamma = \frac{1}{N \max(\text{corr. dist.})} \sum_{a \in \mathcal{V}} \Gamma_a$$

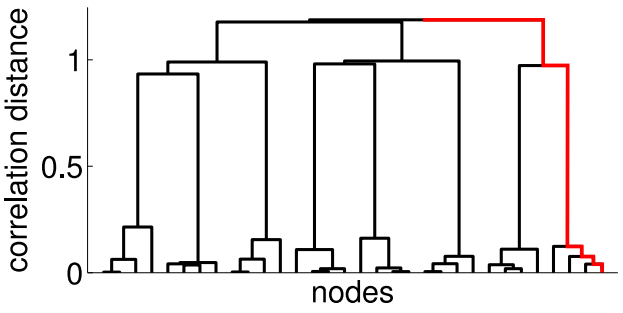
At small scale

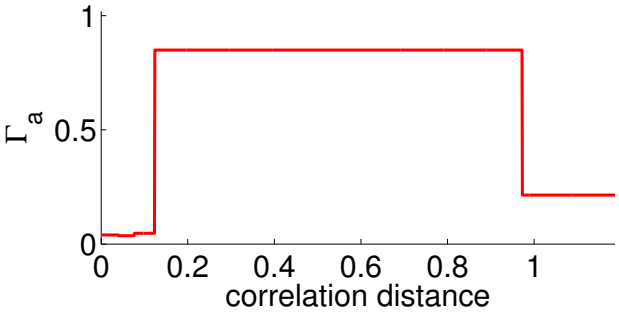




Dendrogram cut at maximal average gap

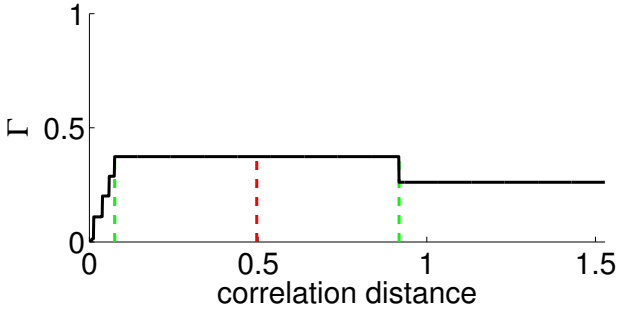


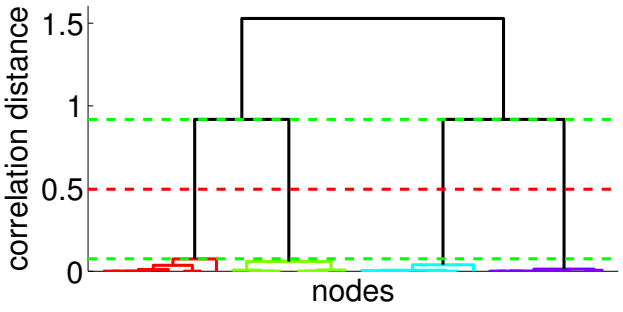




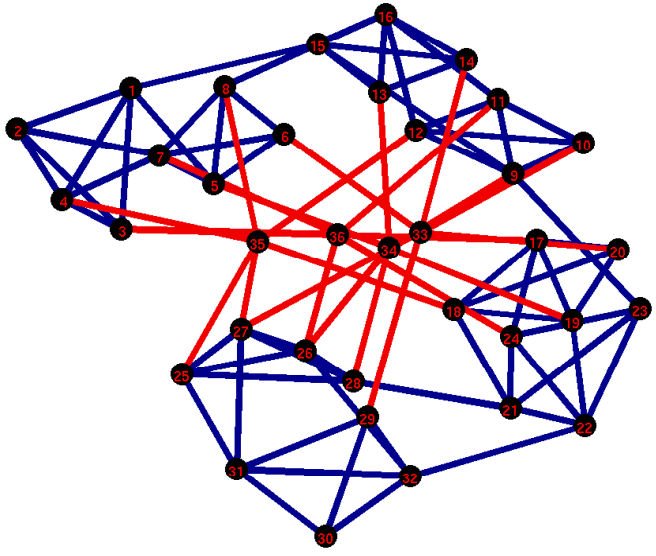
$$\Gamma = \frac{1}{N \max(\text{corr. dist.})} \sum_{a \in \mathcal{V}} \Gamma_a$$

At larger scale

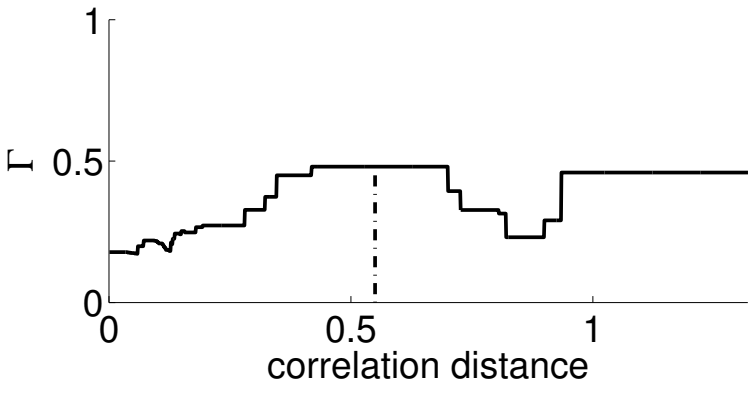


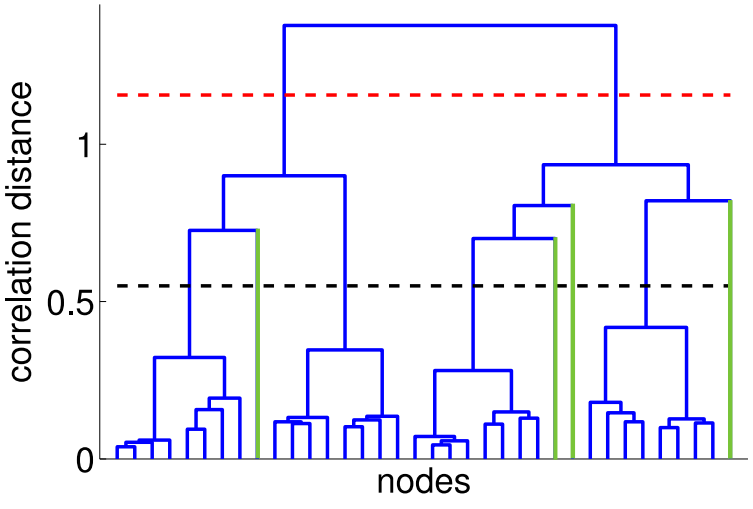


Dendrogram cut at maximal average gap



For the previous graph:





Recall: The Adjusted Rand Index

Let:

- $\mathcal{C}$ and $\mathcal{C}'$ be two partitions we want to compare.
- a be the $\#$ of pairs of nodes that are in the same community in $\mathcal{C}$ *and* in the same community in $\mathcal{C}'$
- b be the $\#$ of pairs of nodes that are in different communities in $\mathcal{C}$ *and* in different communities in $\mathcal{C}'$
- c be the $\#$ of pairs of nodes that are in the same community in $\mathcal{C}$ *and* in different communities in $\mathcal{C}'$
- d be the $\#$ of pairs of nodes that are in different communities in $\mathcal{C}$ *and* in the same community in $\mathcal{C}'$

$a + b$ is the number of "agreements" between $\mathcal{C}$ and $\mathcal{C}'$.
$c + d$ is the number of "disagreements" between $\mathcal{C}$ and $\mathcal{C}'$.

The Adjusted Rand Index

The Rand index, R, is:

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}}$$

The Adjusted Rand index AR is the corrected-for-chance version of the Rand index:

$$AR = \frac{R - ExpectedIndex}{MaxIndex - ExpectedIndex}$$