CR16: Signal Processing and Networks
Data analysis and processing for networks

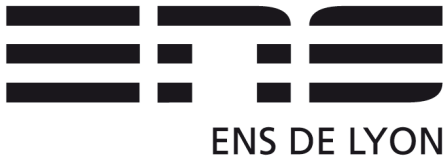
Part 2 - Some applications of spectral graph wavelets

Pierre Borgnat

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

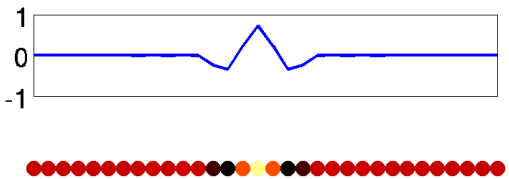
Équipe SISYPHE : Signaux, Systèmes et Physique

09/2014

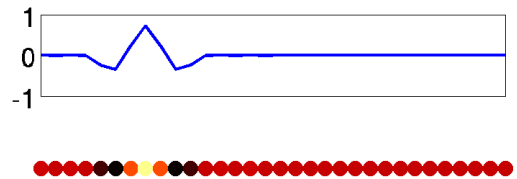


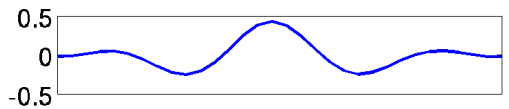
Graph Wavelets

- 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

The classical wavelets

Each wavelet $\psi_{s,a}$ is derived by translating and scaling a mother wavelet ψ:

$$\psi_{s,a}(x) = \frac{1}{s}\psi\left(\frac{x-a}{s}\right)$$

Equivalently, in the Fourier domain:

$$\begin{aligned}
\hat{\psi}_{s,a}(\omega) &= \int_{-\infty}^{\infty} \frac{1}{s}\psi\left(\frac{x-a}{s}\right) \exp^{-i\omega x}\, dx \\
&= \exp^{-i\omega a} \int_{-\infty}^{\infty} \frac{1}{s}\psi\left(\frac{X}{s}\right) \exp^{-i\omega X}\, dX \\
&= \exp^{-i\omega a} \int_{-\infty}^{\infty} \psi\left(X'\right) \exp^{-i\omega X'}\, dX' \\
&= \hat{\delta}_a(\omega)\, \hat{\psi}(s\omega) \qquad \text{where} \quad \delta_a = \delta(x-a)
\end{aligned}$$

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

The classical wavelets

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

- 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]

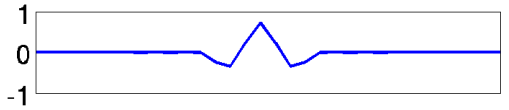
	Classical (continuous) world	Graph world
Real domain	x	node a
Fourier domain	ω	eigenvalues λ_i
Filter kernel	$\hat{\psi}(\omega)$	$g(\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}$
Filter bank	$\hat{\psi}(s\omega)$	$g(s\lambda_i) \Leftrightarrow \hat{\boldsymbol{G}}_{\boldsymbol{s}}$
Fourier modes	$\exp^{-i\omega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

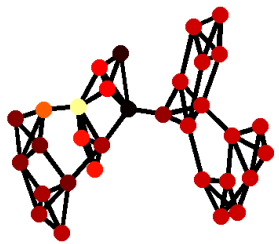
The wavelet at scale s centered around node a is given by:

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

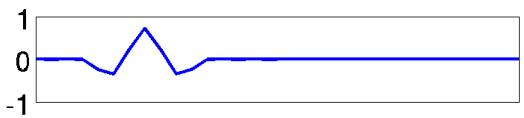
Examples of graph wavelets

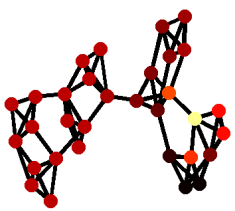
A WAVELET:



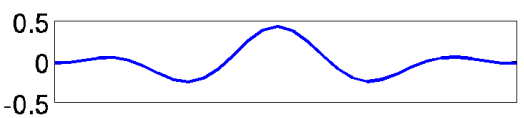


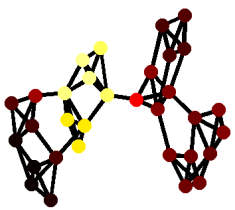
TRANSLATING:





SCALING:





Examples of wavelets: they encode the local topology

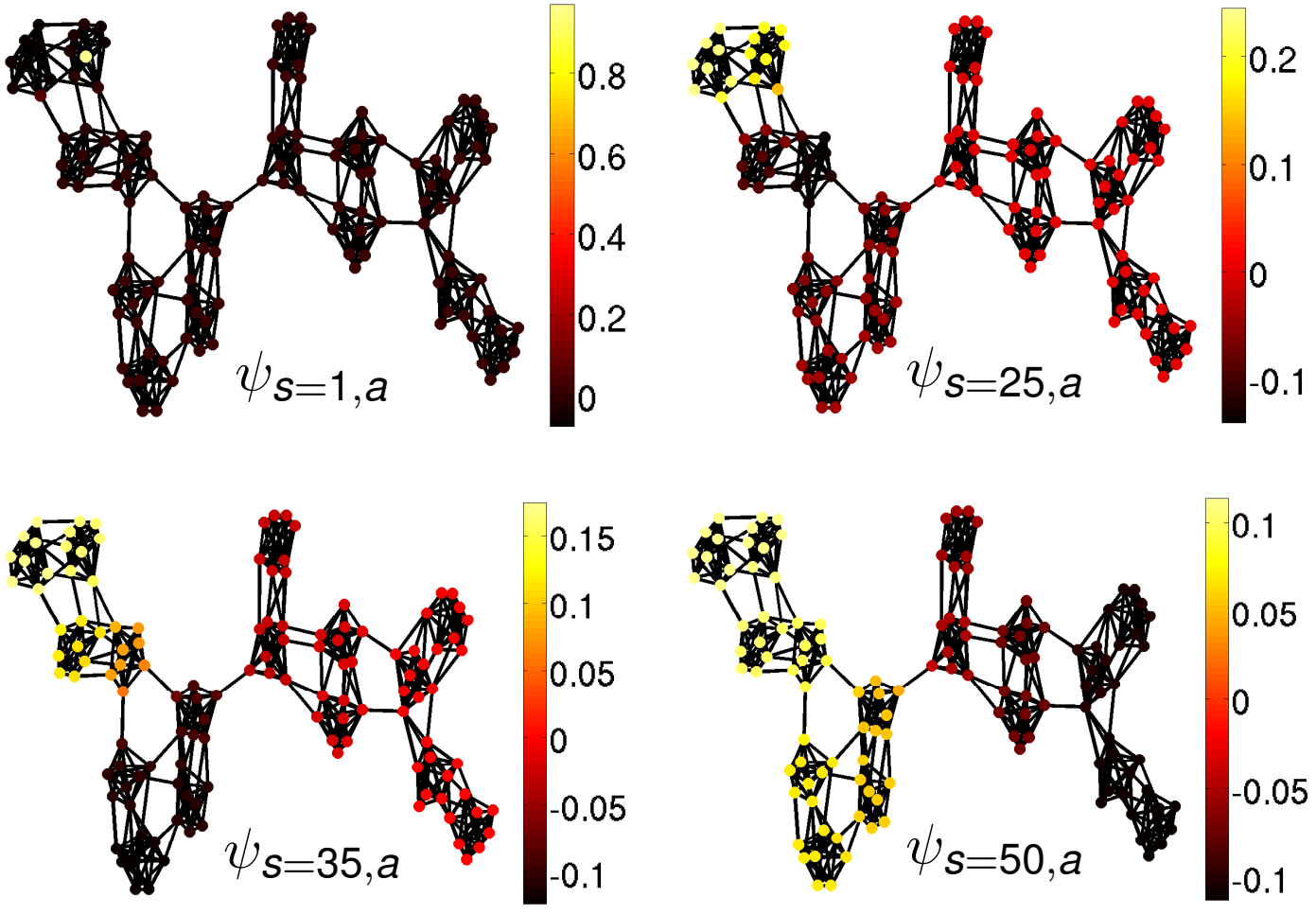
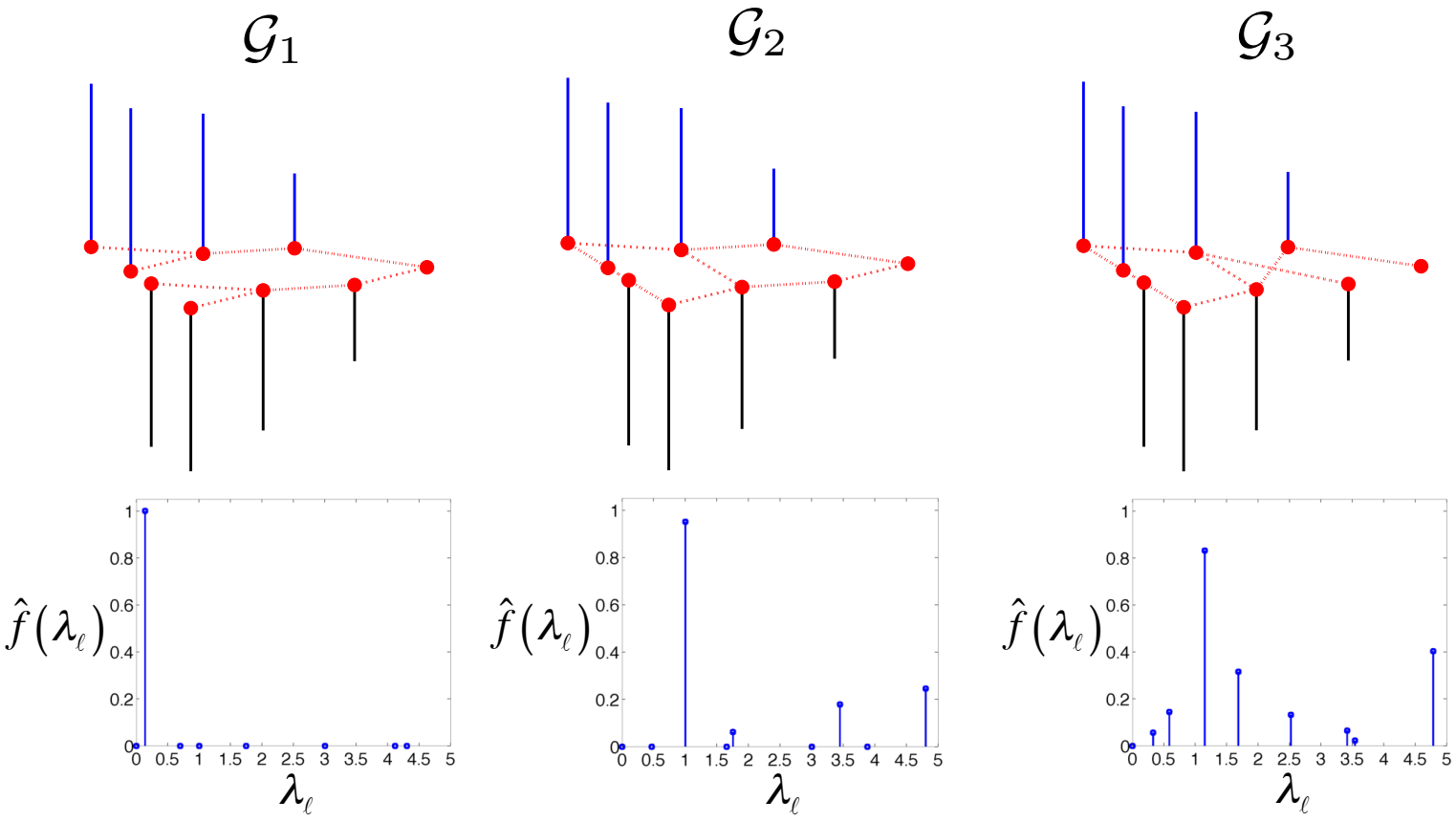
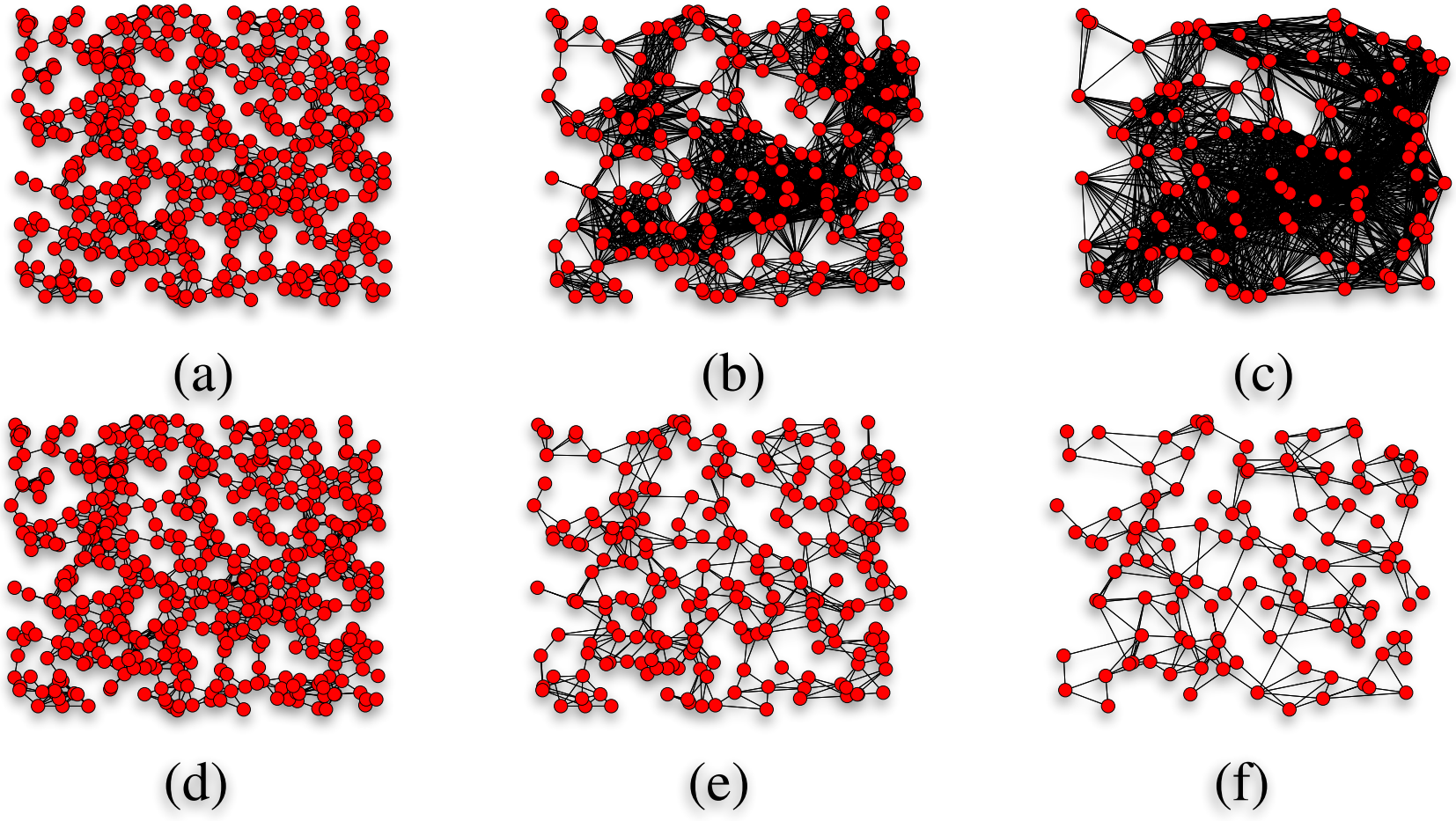


Illustration on the smoothness of graph signals

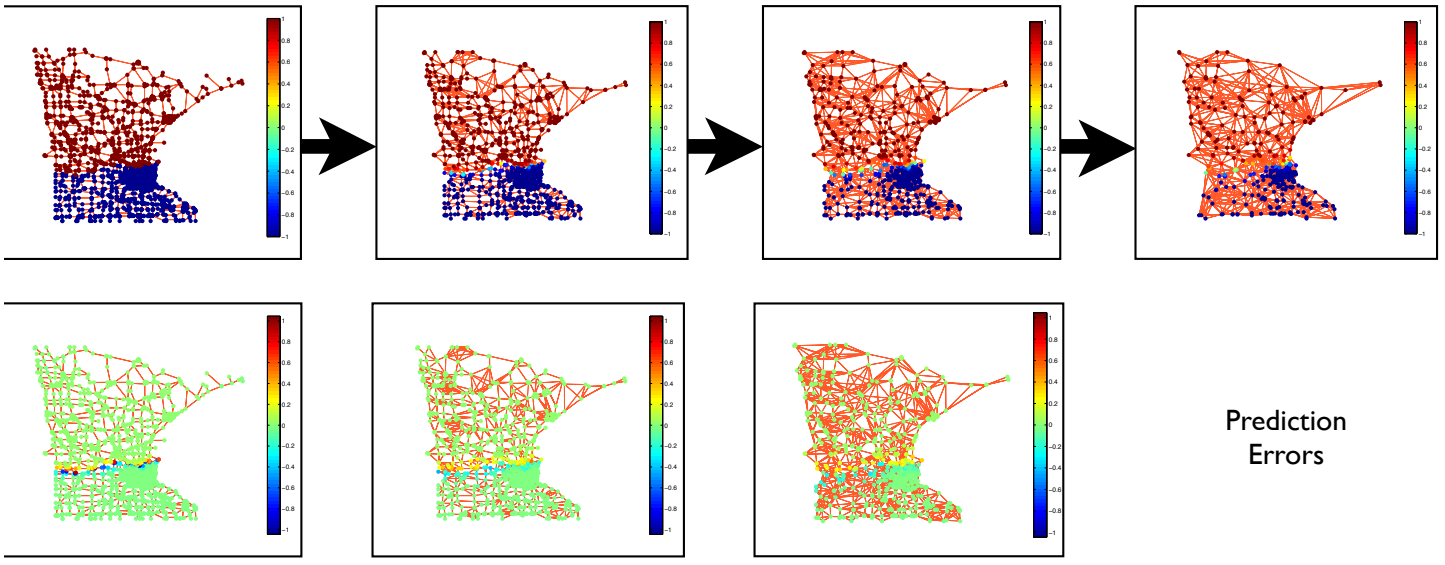


$\mathbf{f}^{\mathrm{T}}\mathcal{L}_1\mathbf{f} = 0.14$ $\mathbf{f}^{\mathrm{T}}\mathcal{L}_2\mathbf{f} = 1.31$ $\mathbf{f}^{\mathrm{T}}\mathcal{L}_3\mathbf{f} = 1.81$

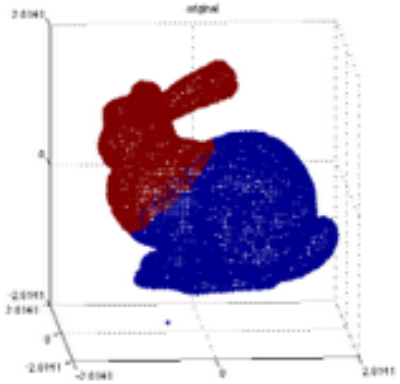
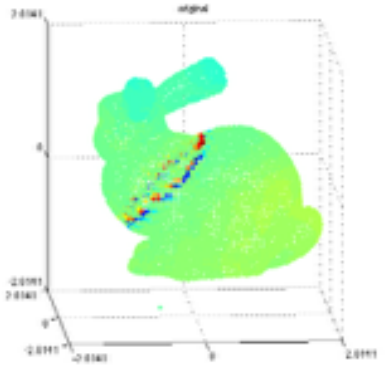
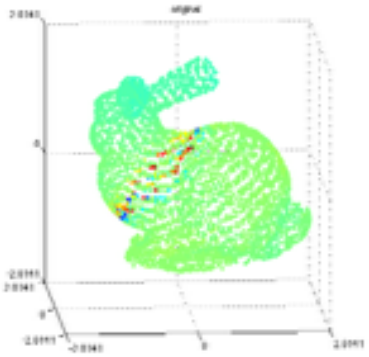
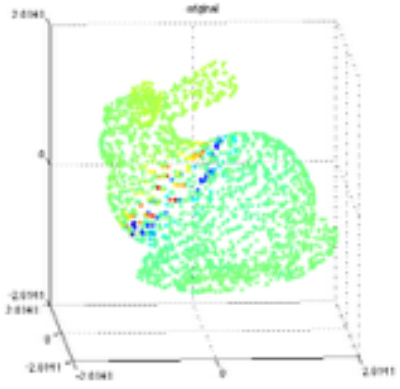
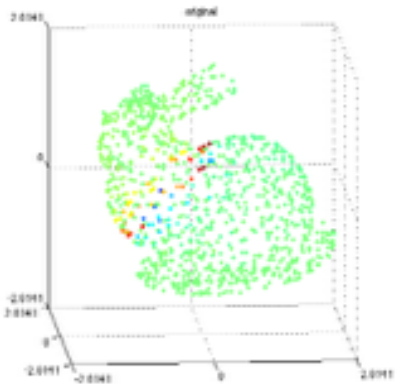
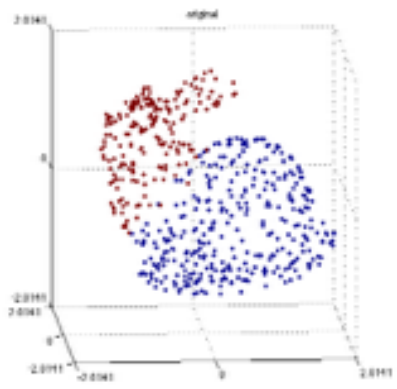
Kron reduction and spasification



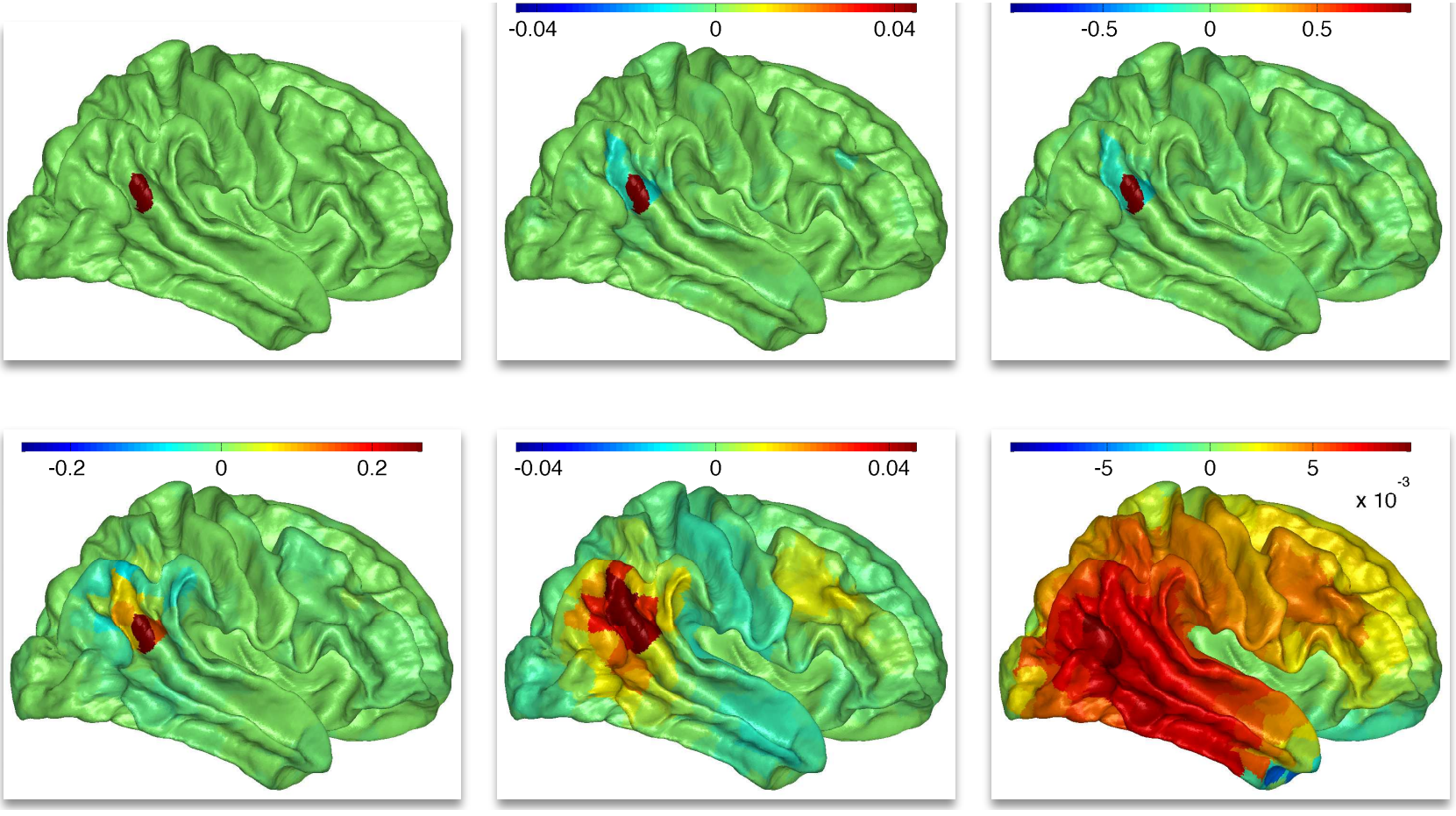
Graph and Signal graph coarsening



Graph and Signal graph coarsening

Graph wavelets for brain fMRI data



Purpose of the last part of the lecture

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

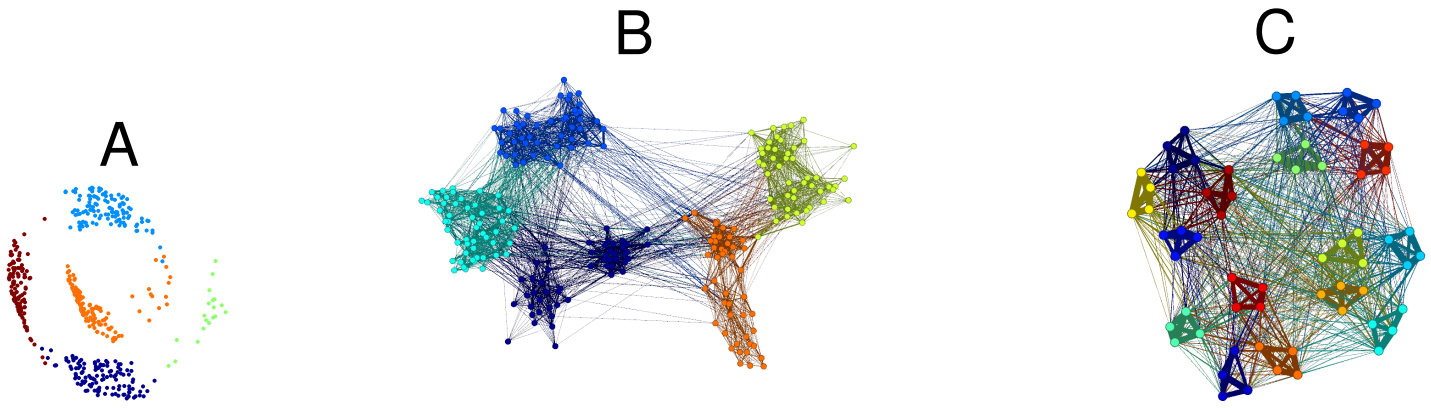
General Ideas

- 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

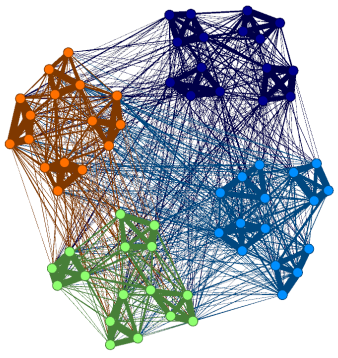
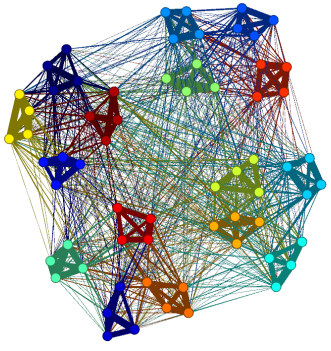
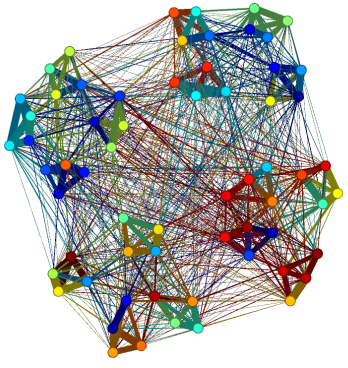
Let us recall: objective of community detection

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]



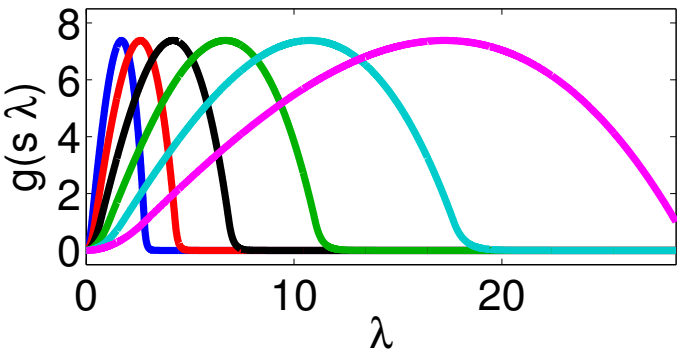
or *multiscale* community detection ?

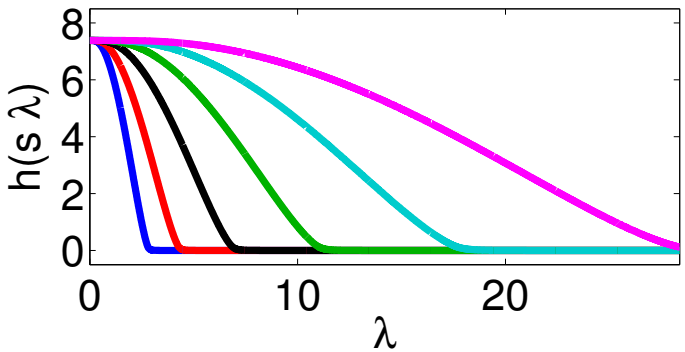




Example of filters for community detection

For each graph under study, we automatically find the good filter parameters for g by imposing:

- The coarsest scale needs to be focused on the eigenvector χ_1 (Fiedler vector).
- All scales need at least to keep some information from χ_1.
- The finest scale needs to use the information from all eigenvectors (i.e., Fourier modes).





Example of wavelet filters

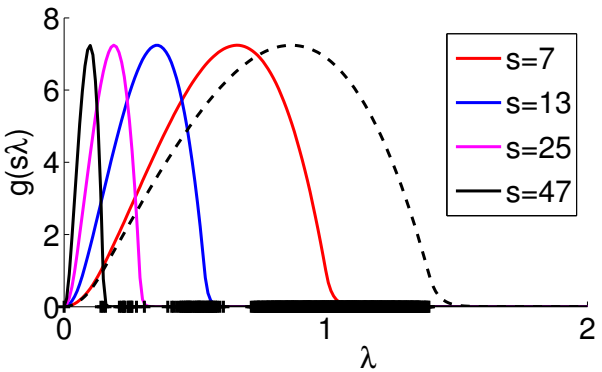
- More precisely, we will use the following kernel:

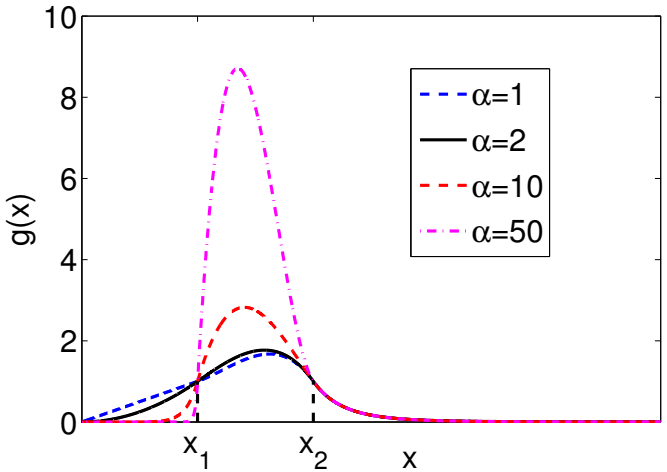
$$g(x;\alpha,\beta,x_1,x_2) = \begin{cases} x_1^{-\alpha}\, x^{\alpha} & \text{for} \quad x < x_1 \\ p(x) & \text{for} \quad x_1 \le x \le x_2 \\ x_2^{\beta}\, x^{-\beta} & \text{for} \quad x > x_2. \end{cases}$$

- The parameters will be:

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

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

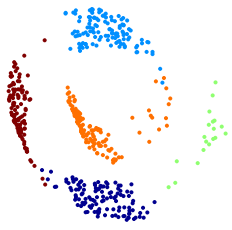




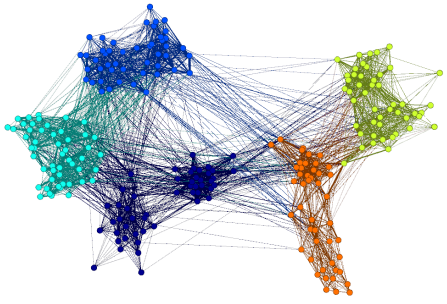
Multiscale community structure in a graph

Classical community detection algorithm based (for instance) on modularity optimisation only finds one solution:

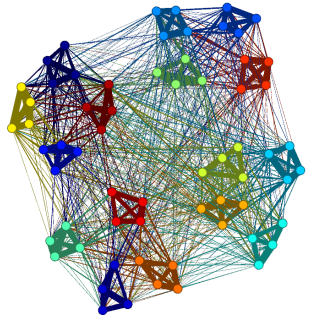
A



B



C



Where the modularity function reads:

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

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:

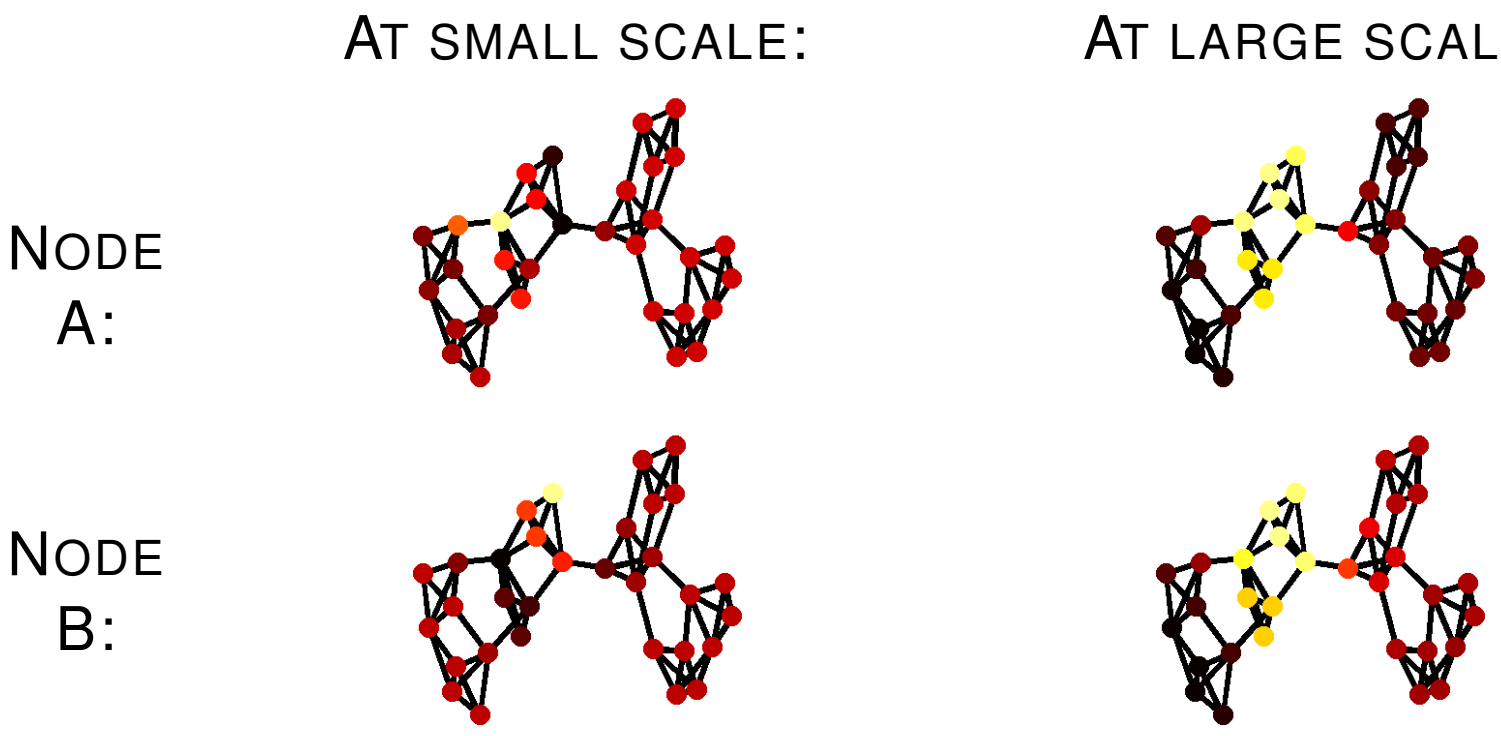
1. wavelets as feature vectors
2. the correlation distance
3. the complete linkage clustering algorithm

1) Wavelets as features

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

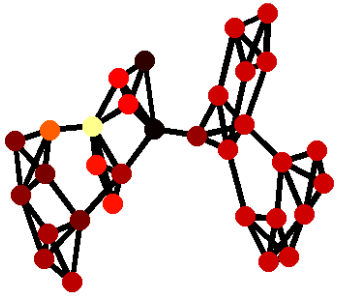
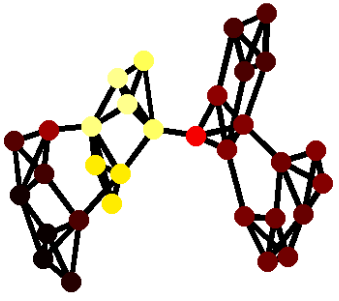
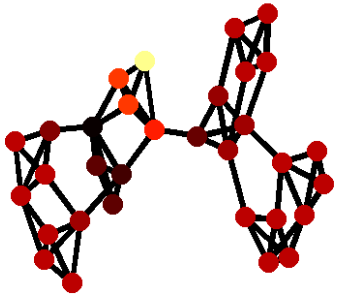
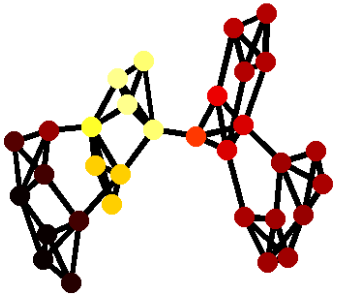
Globally, one will need Ψ_s, all wavelets at a given scale s, i.e.

$$\Psi_s = (\psi_{s,1}|\psi_{s,2}|\dots|\psi_{s,N}) = \chi G_s \chi^\top.$$



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
RESULT:	Far appart in the dendrogram	Close to each other in the dendrogram

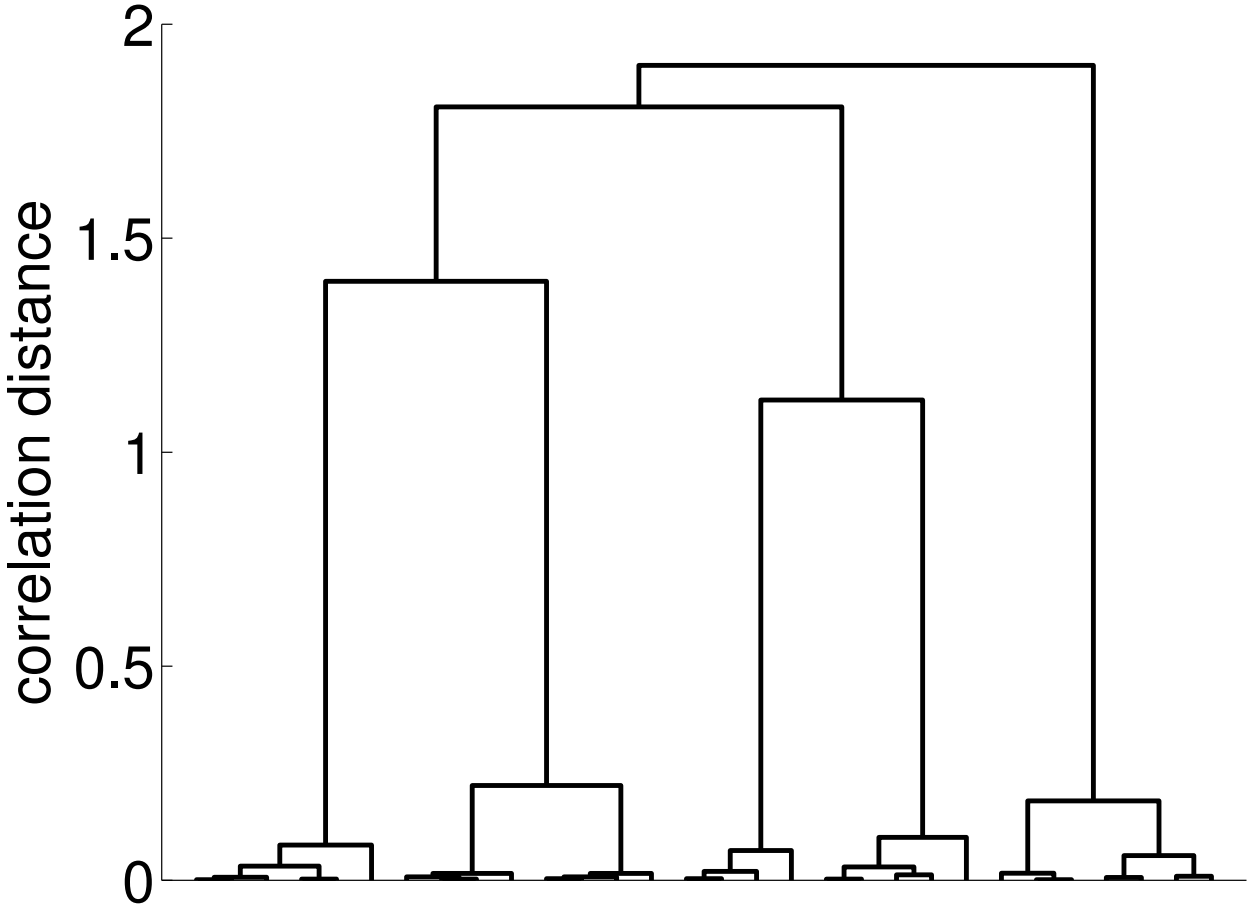
3) Complete linkage clustering and dendrogram

- It is a bottom to top hierarchical algorithm: it starts with as many clusters as nodes and works its way up to fewer clusters (by linking subclusters together) until it reaches one global cluster.

- To compute the distance between two subclusters under examination : all possible pairs of nodes, taking one from each cluster, are considered. The *maximum* possible node-to-node distance is declared to be the cluster-to-cluster closeness.

- Outputs a dendrogram (from Greek dendron "tree" and gramma "drawing").

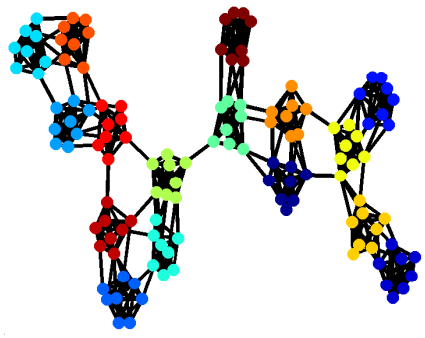
Example of a dendrogram **at a given scale s**



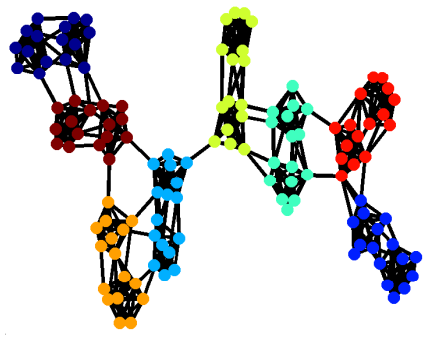
The big question: where should we cut the dendrogram?

A toy graph for introducing the method

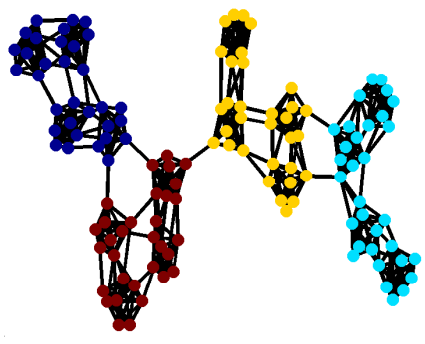
smallest scale (16 com.):



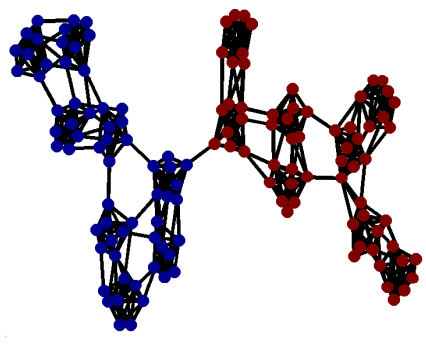
small scale (8 com.):



medium scale (4 com.):



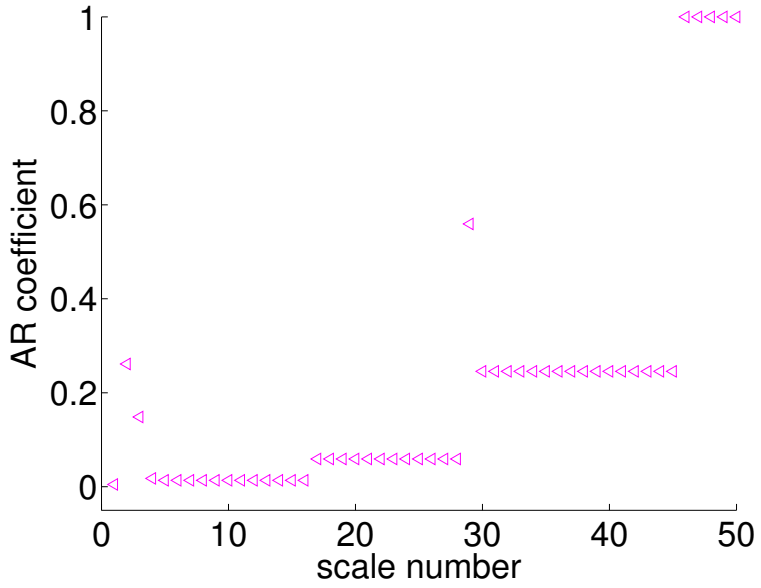
large scale (2 com.):



Dendrogram cut with prior knowledge

Let us cheat by using prior knowledge on the number of communities we are looking for.

If we cut each dendrogram in two clusters



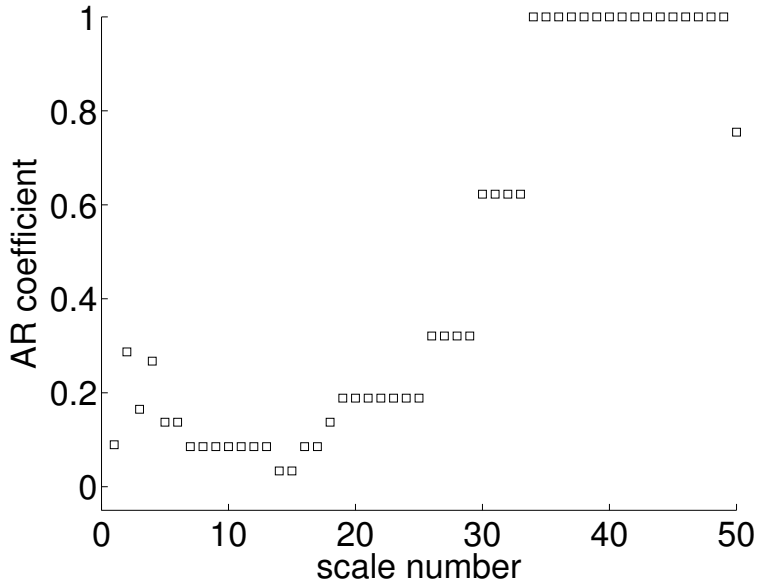
Using wavelets as features

Conclusion: the dendrograms at different scales contain the community structure at various scales.

Dendrogram cut with prior knowledge

Let us cheat by using prior knowledge on the number of communities we are looking for.

If we cut each dendrogram in four clusters



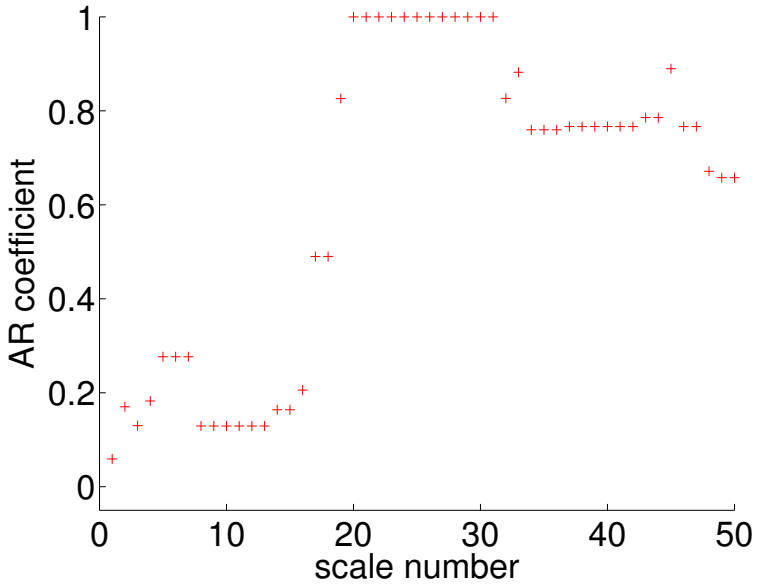
Using wavelets as features

Conclusion: the dendrograms at different scales contain the community structure at various scales.

Dendrogram cut with prior knowledge

Let us cheat by using prior knowledge on the number of communities we are looking for.

If we cut each dendrogram in eight clusters



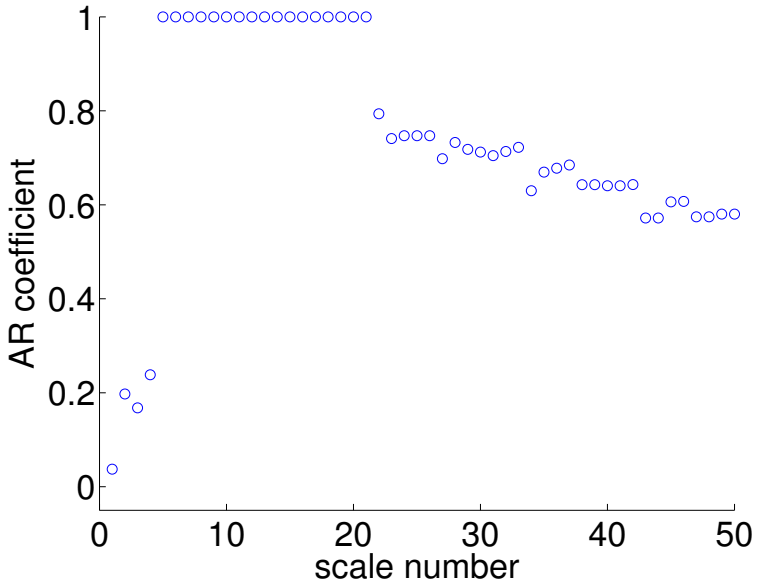
Using wavelets as features

Conclusion: the dendrograms at different scales contain the community structure at various scales.

Dendrogram cut with prior knowledge

Let us cheat by using prior knowledge on the number of communities we are looking for.

If we cut each dendrogram in sixteen clusters

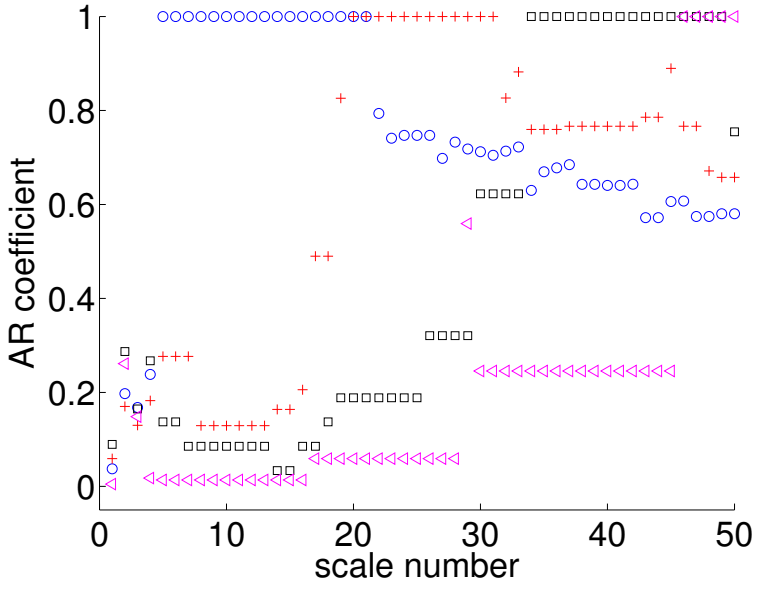


Using wavelets as features

Conclusion: the dendrograms at different scales contain the community structure at various scales.

Dendrogram cut with prior knowledge

Let us cheat by using prior knowledge on the number of communities we are looking for.
The four levels of communities.



Using wavelets as features

Conclusion: the dendrograms at different scales contain the community structure at various scales.

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:

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

Dendrogram cut with classical modularity

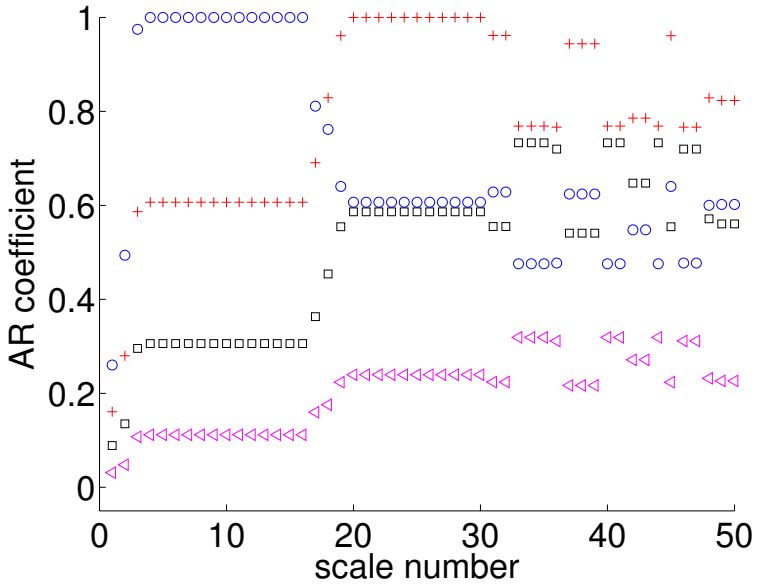
Recall that the classical modularity matrix reads:

$B(A) = \frac{1}{2m}(A + \frac{dd^\top}{2m})$

where d is the strength vector and $2m = \sum d(i)$

Classical modularity is

$$Q = Tr(S^\top BS)$$



Solution not good at large scale.

Dendrogram cut with filtered modularity

We define the filtered adjacency matrices at scale s:

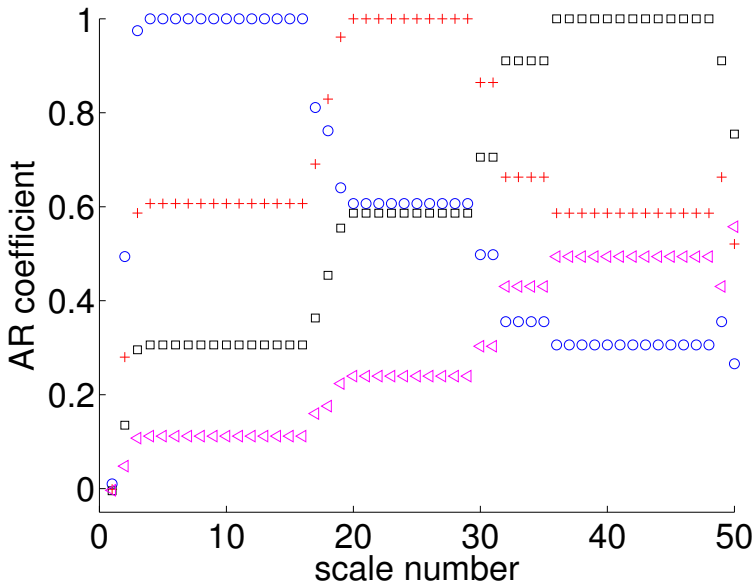
- recall that $A = D^{\frac{1}{2}} \chi (I - \Lambda) \chi^\top D^{\frac{1}{2}}$
- $A_s^g = A + D^{\frac{1}{2}} \chi \hat{G}_s \chi^\top D^{-\frac{1}{2}} A$

We define the filtered modularity matrices at scale s:

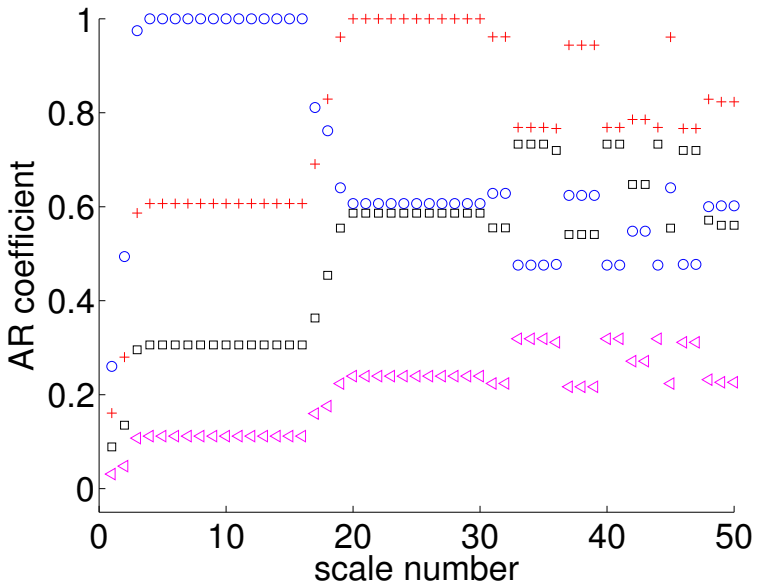
$$B_s^g = B(A_s^g)$$

Maximize filtered modularity

Filtered Modu Opt.



Classical Modu Opt.



Notes about the filtered modularity

$$A_s^g = A + D^{\frac{1}{2}} \chi \hat{G}_s \chi^\top D^{-\frac{1}{2}} A$$

Consider d the vector of strengths of A and $2m$ the sum of the strengths. The classical modularity reads:

$$B = \frac{A}{2m} - \frac{dd^\top}{(2m)^2}$$

Consider d' the vector of strengths of A_s^g and $2m'$ the sum of the strengths. We can show that:

$$\frac{dd^\top}{(2m)^2} = \frac{d'd'^\top}{(2m')^2}$$

Moreover, if $g_s(1) = 0$ (which is the case), the filtered modularity reads:

$$B_s^g = \frac{A + D^{\frac{1}{2}} \chi \hat{G}_s \chi^\top D^{-\frac{1}{2}} A}{2m} - \frac{dd^\top}{(2m)^2}$$

Notes about the filtered modularity

$$B_s^g = \frac{A + D^{\frac{1}{2}} \chi \hat{G}_s \chi^\top D^{-\frac{1}{2}} A}{2m} - \frac{dd^\top}{(2m)^2}$$

Recall that modularity compares the actual normalised weight $\frac{A_{ij}}{2m}$ to the expected weight if the graph was a random Chung-Lu graph: $\frac{d_i d_j}{(2m)^2}$.

The filtered modularity does not change the expected weight but rather changes the actual normalised weigth: it adds or retrieve value to $\frac{A_{ij}}{2m}$. At small scale, it will increase the weights important for small scale structures and decrease the weights important for superstructures.

Notes about the filtered modularity

It can be written:

$$B_s^g = \frac{1}{2m}\sum_{i=2}^{N}(1+g_s(i))(1-\lambda_i)D^{\frac{1}{2}}\chi_i\chi_i^\top D^{\frac{1}{2}}$$

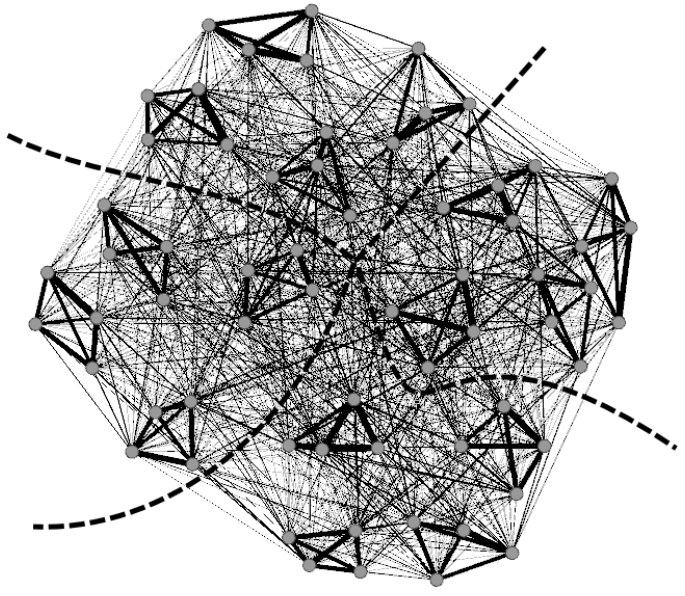
To compare to Schaub-Delvenne's filtered modularity:

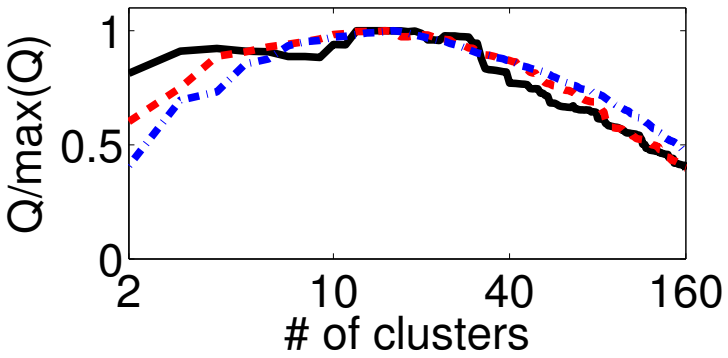
$$B_t = \frac{1}{2m}\sum_{i=2}^{N}(1-\lambda_i)^t D^{\frac{1}{2}}\chi_i\chi_i^\top D^{\frac{1}{2}}$$

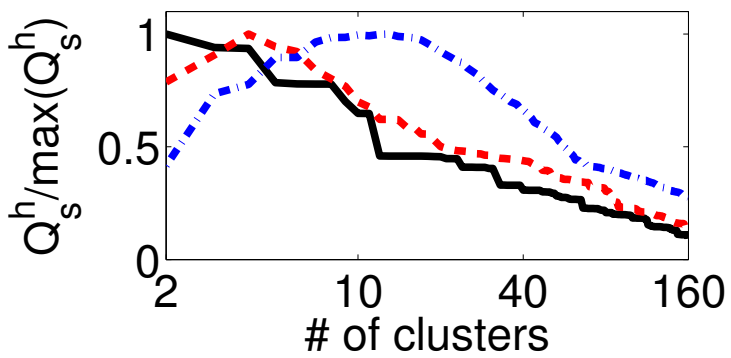
And Arenas' version: (here for regular networks)

$$B_\alpha = \frac{1}{2m}\sum_{i=2}^{N}(1-\frac{\lambda_i}{\alpha})D^{\frac{1}{2}}\chi_i\chi_i^\top D^{\frac{1}{2}}$$

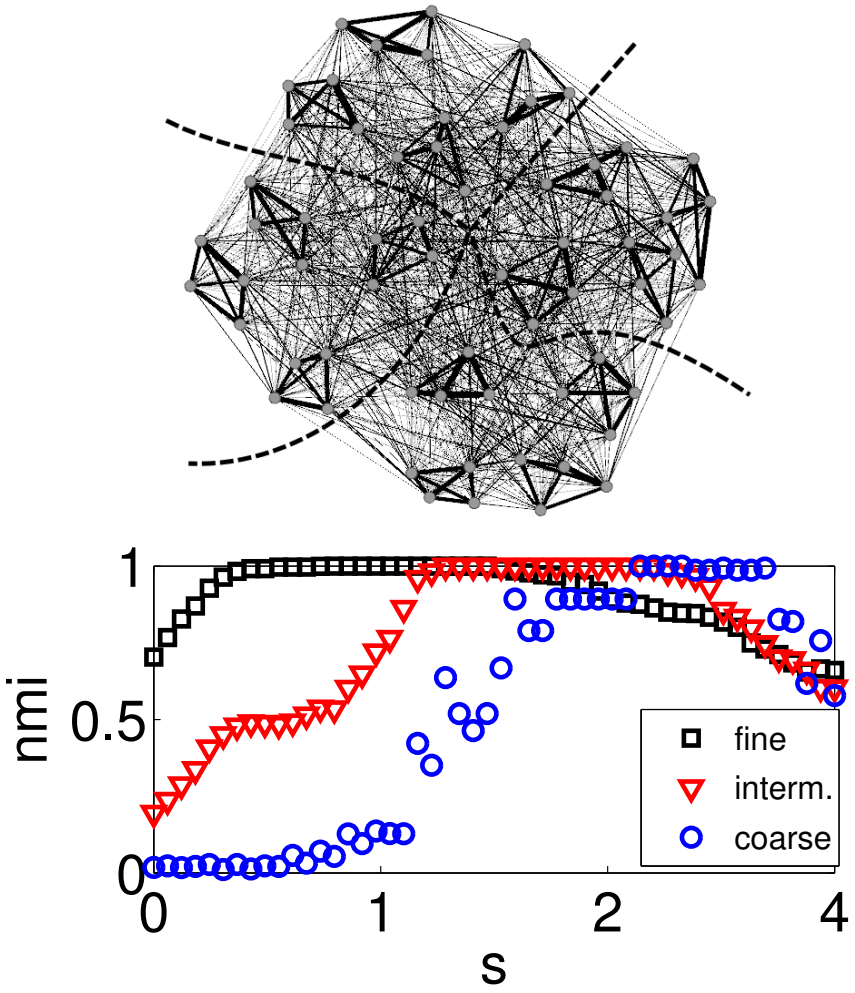
Maximize filtered modularity on Sales-Pardo network



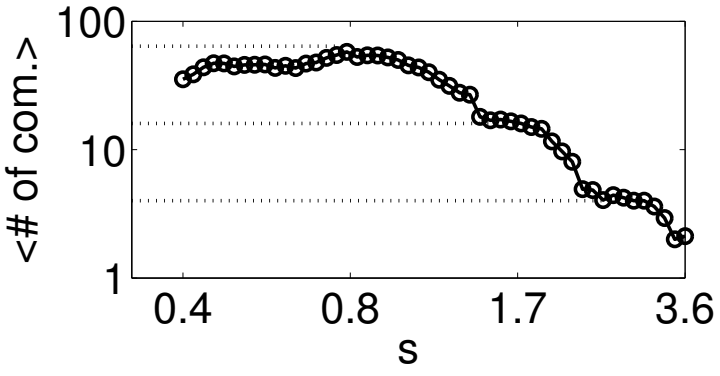


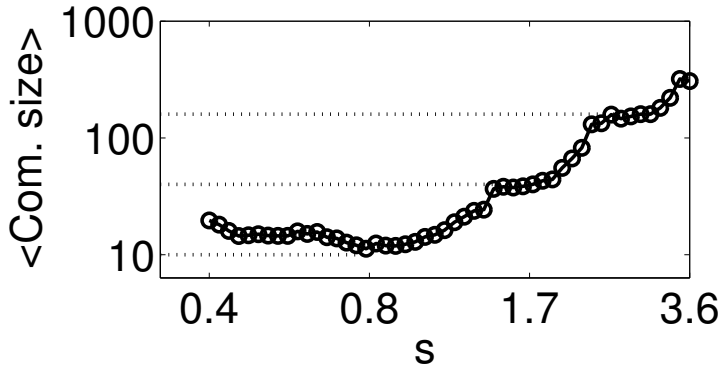


Maximize filtered modularity on Sales-Pardo network



Maximize filtered modularity on Sales-Pardo network





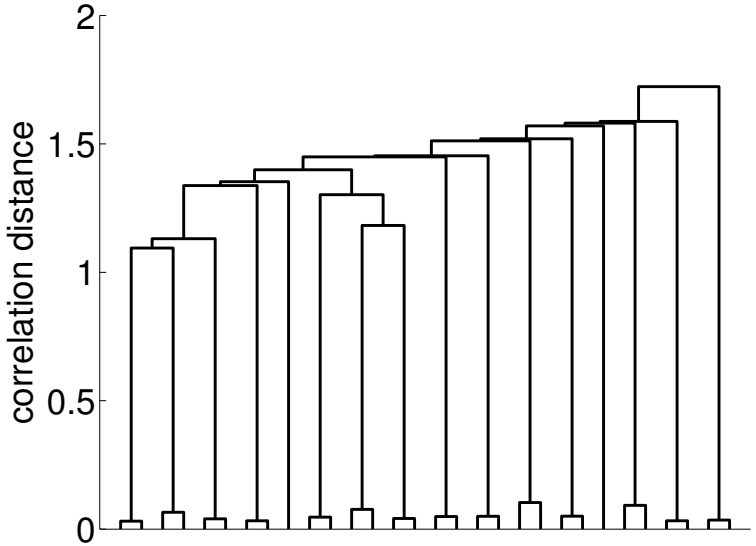
Intermediate bilan

- As expected, the method works week with filtered modularity
- Fundamental reason: it is related to Arenas or Schaub-Delvenne modified modularity to take into account a scale
- However: the dendrogram has already in itself the good solutions, with no need of the step of (filtered) modularity optimization.
- For that: look at the gaps !

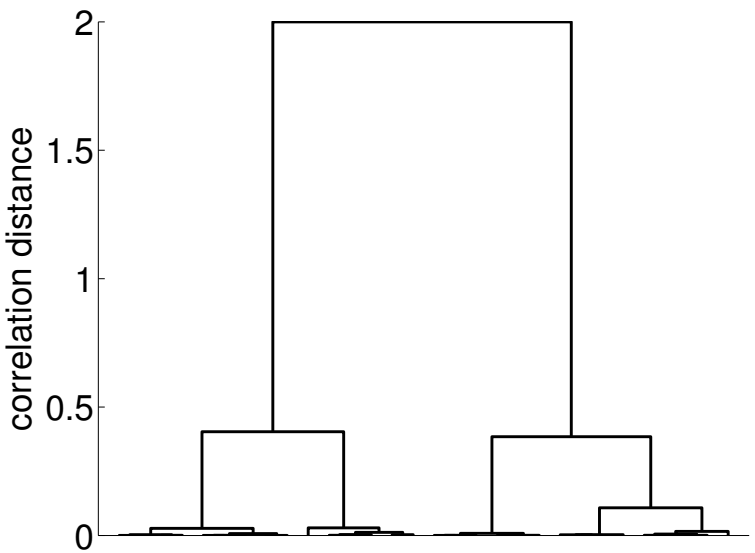
Dendrogram cut at maximal gap

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

At small scale:



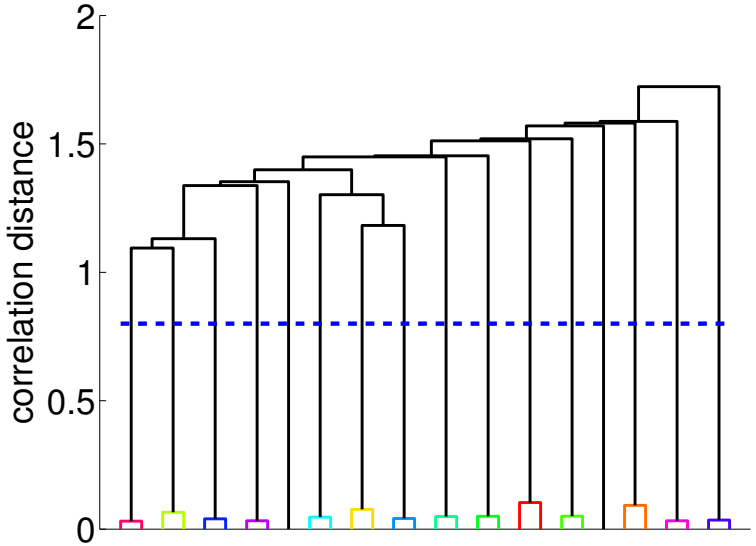
At large scale:



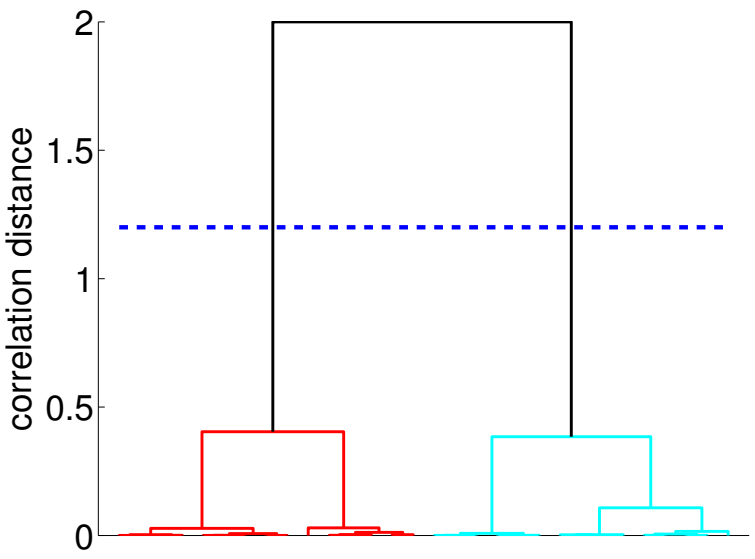
Dendrogram cut at maximal gap

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

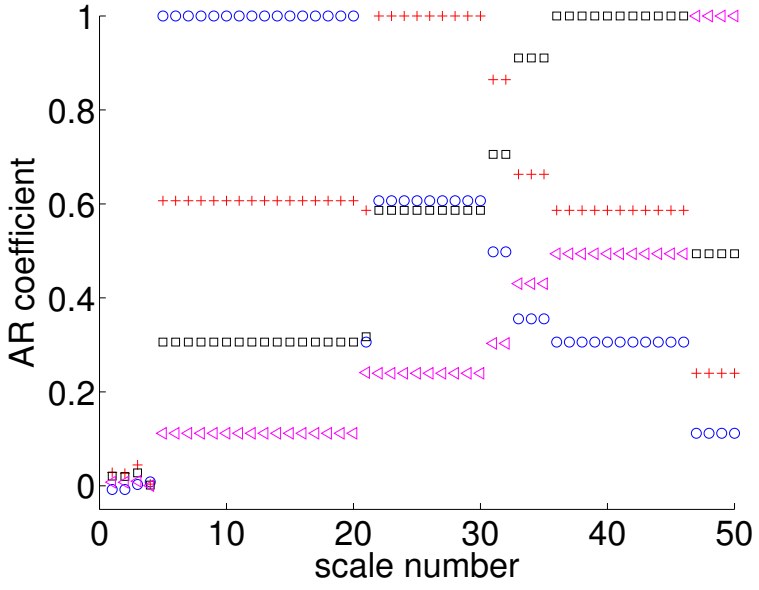
At small scale:



At large scale:

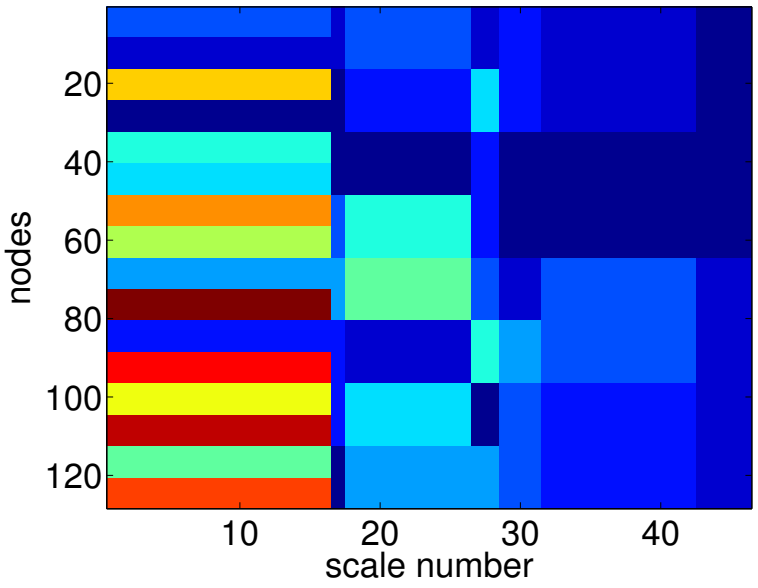


Dendrogram cut at maximal gap



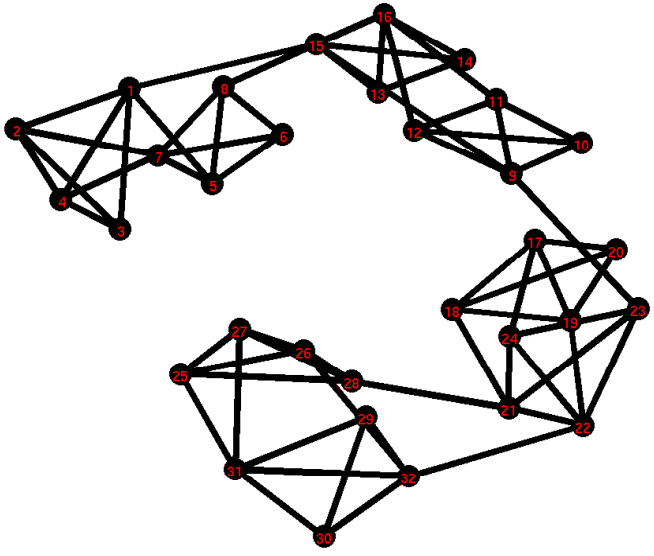
Using wavelets

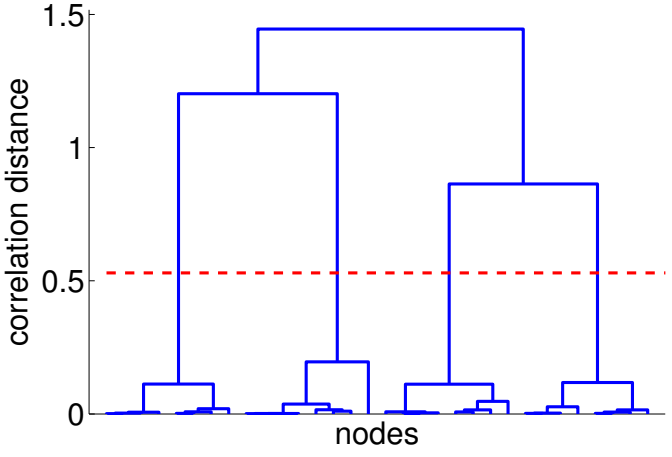
Dendrogram cut at maximal gap

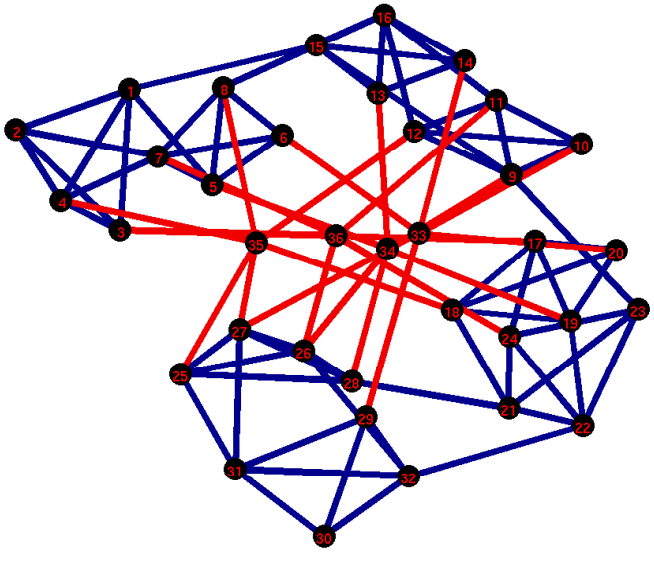


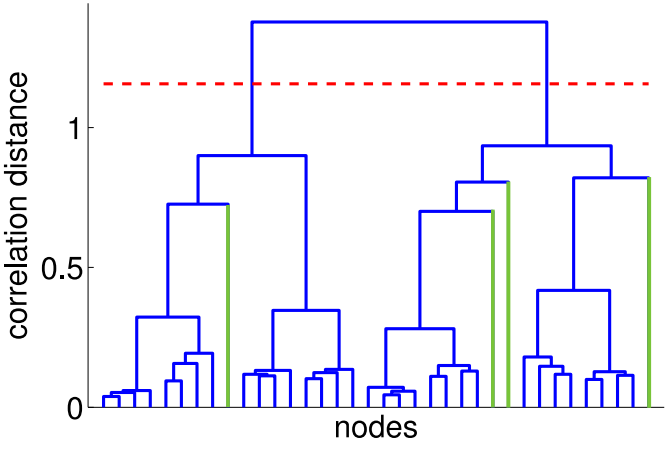
Using wavelets

Dendrogram cut at maximal gap: non robust to outliers

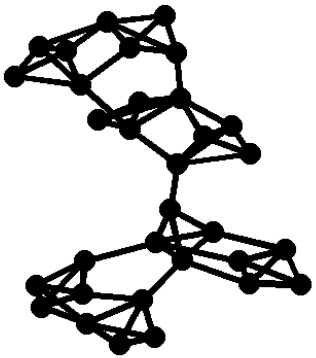


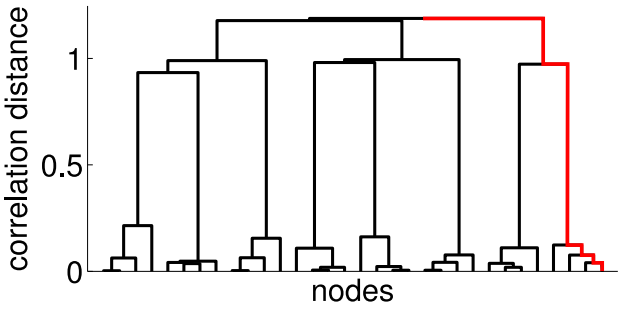


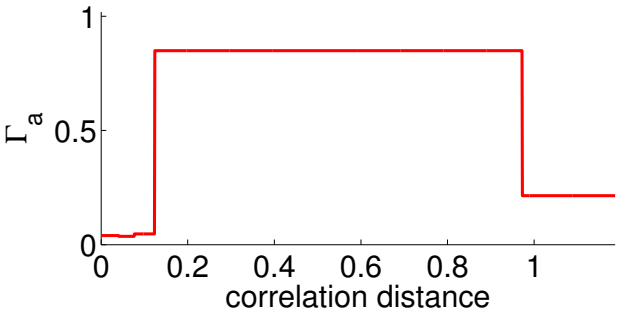




Dendrogram cut at maximal average gap

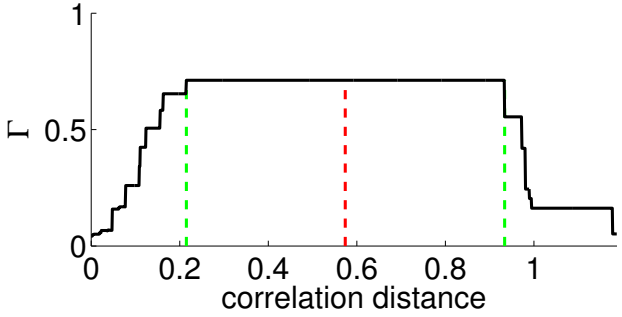


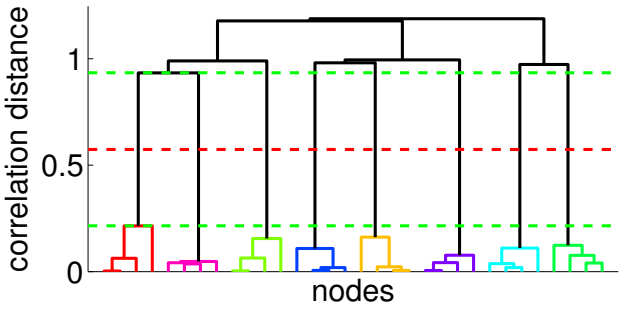




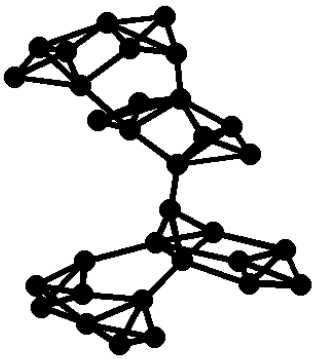
$$\Gamma = \frac{1}{N \max(\text{corr. dist.})} \sum_{a \in \mathcal{V}} \Gamma_a$$

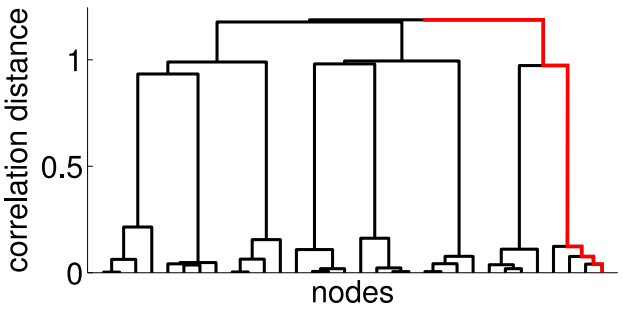
At small scale

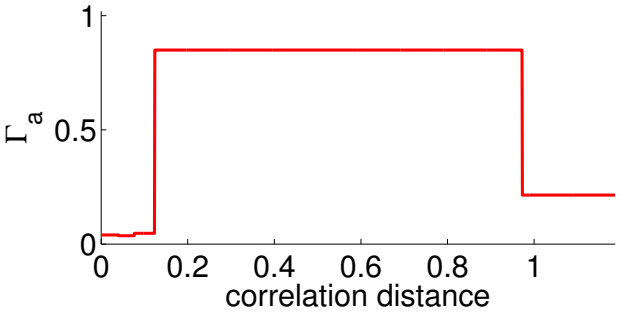




Dendrogram cut at maximal average gap

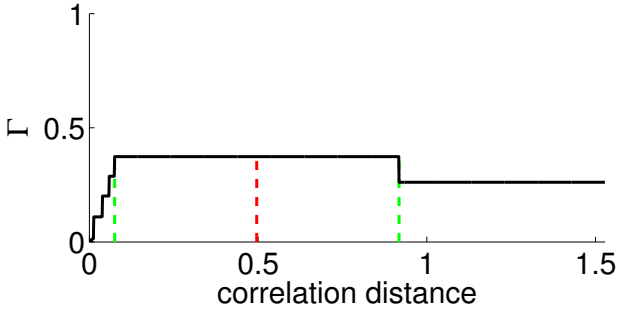


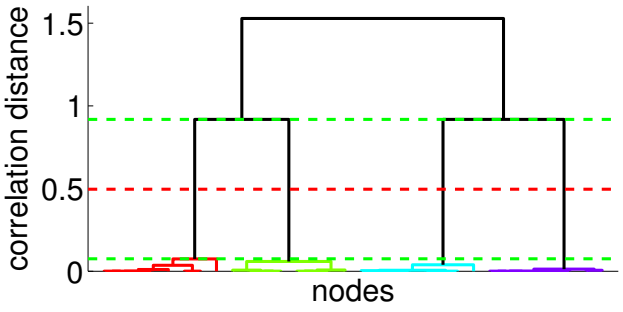




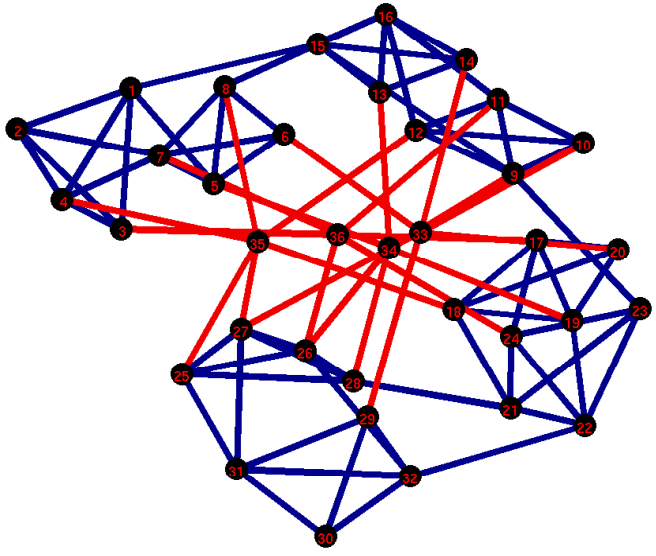
$$\Gamma = \frac{1}{N \max(\text{corr. dist.})} \sum_{a \in \mathcal{V}} \Gamma_a$$

At larger scale

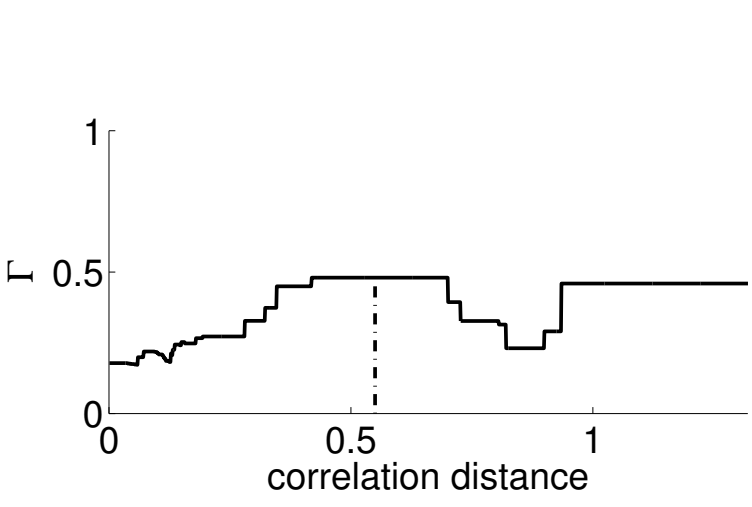


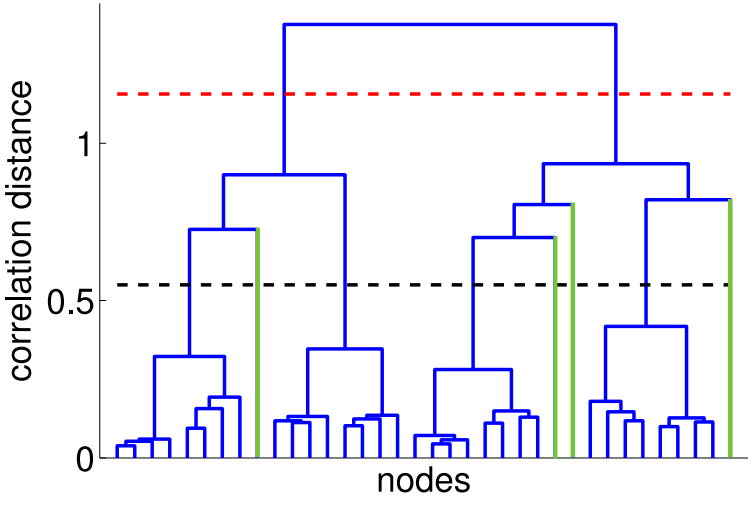


Dendrogram cut at maximal average gap



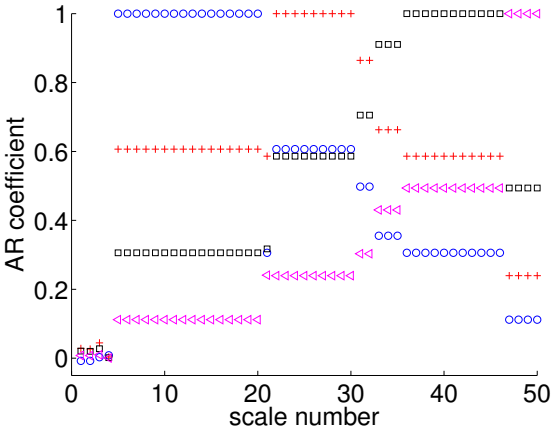
For the previous graph:



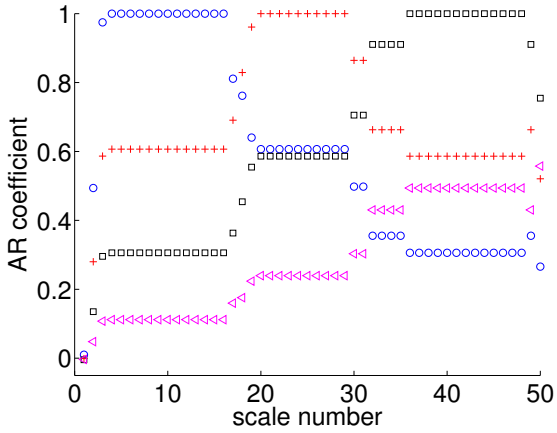


Comparison maximal gap vs. filtered modularity

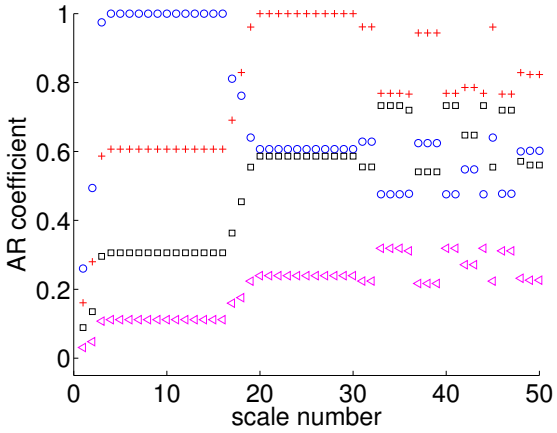
Maximal Gap



Filtered Modu Opt.

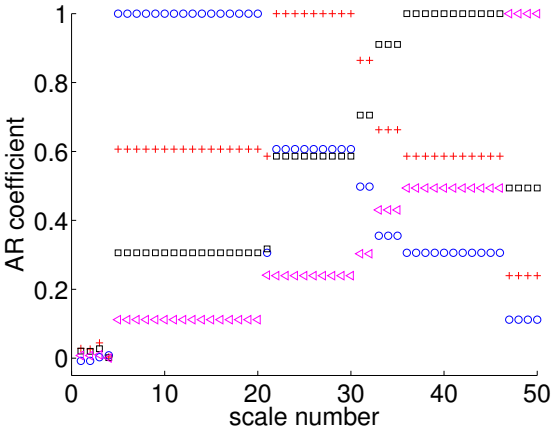


Classical Modu Opt.

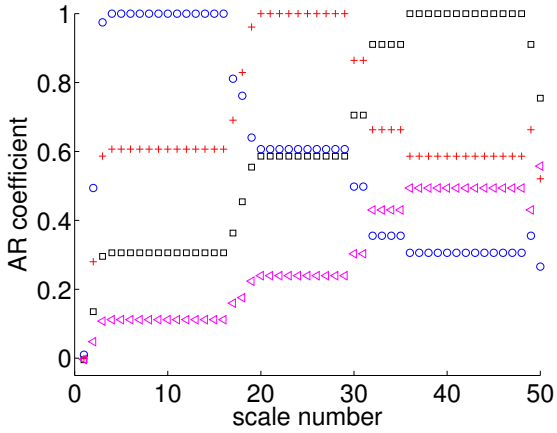


Comparison maximal gap vs. filtered modularity

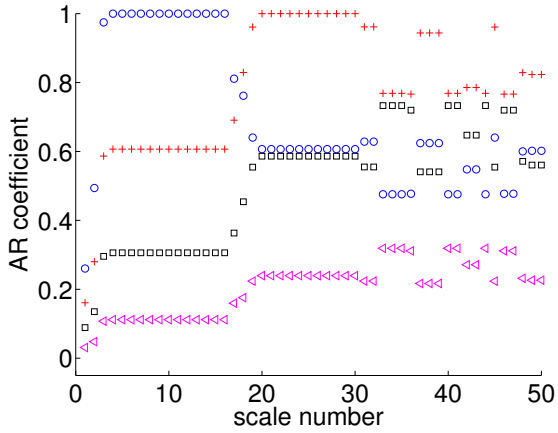
Maximal Gap



Filtered Modu Opt.

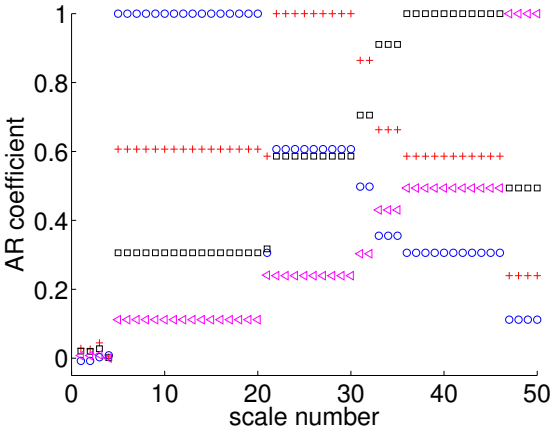


Classical Modu Opt.

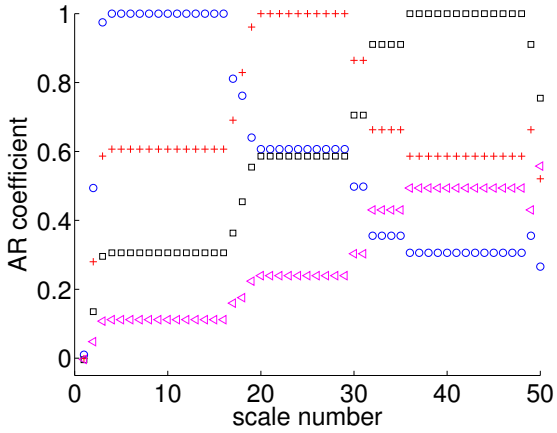


Comparison maximal gap vs. filtered modularity

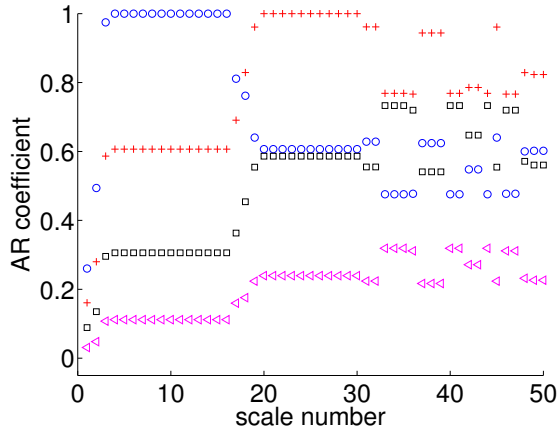
Maximal Gap



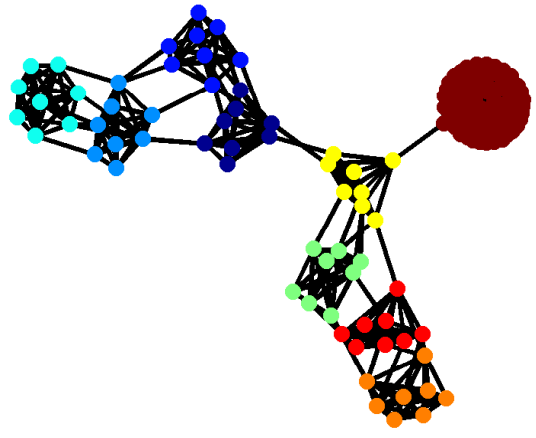
Filtered Modu Opt.



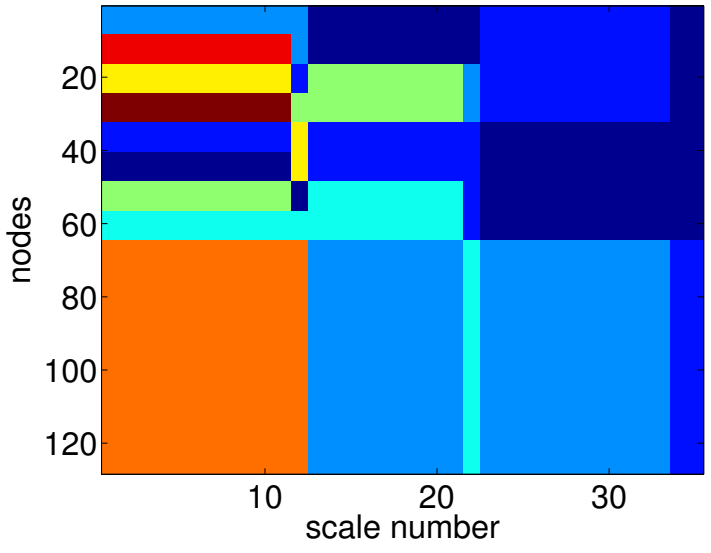
Classical Modu Opt.



Multiscale community detection on a simple network

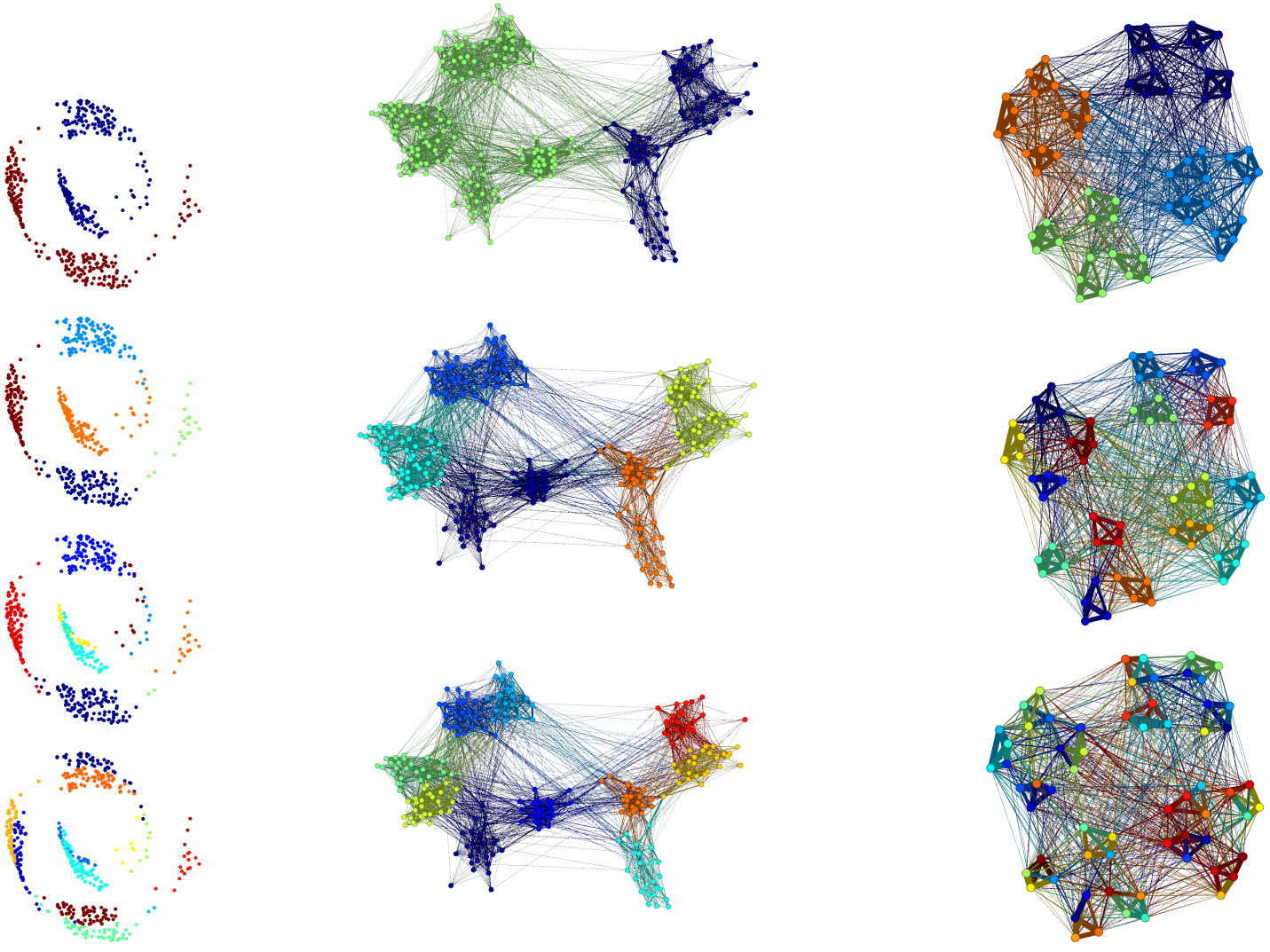


Another toy graph



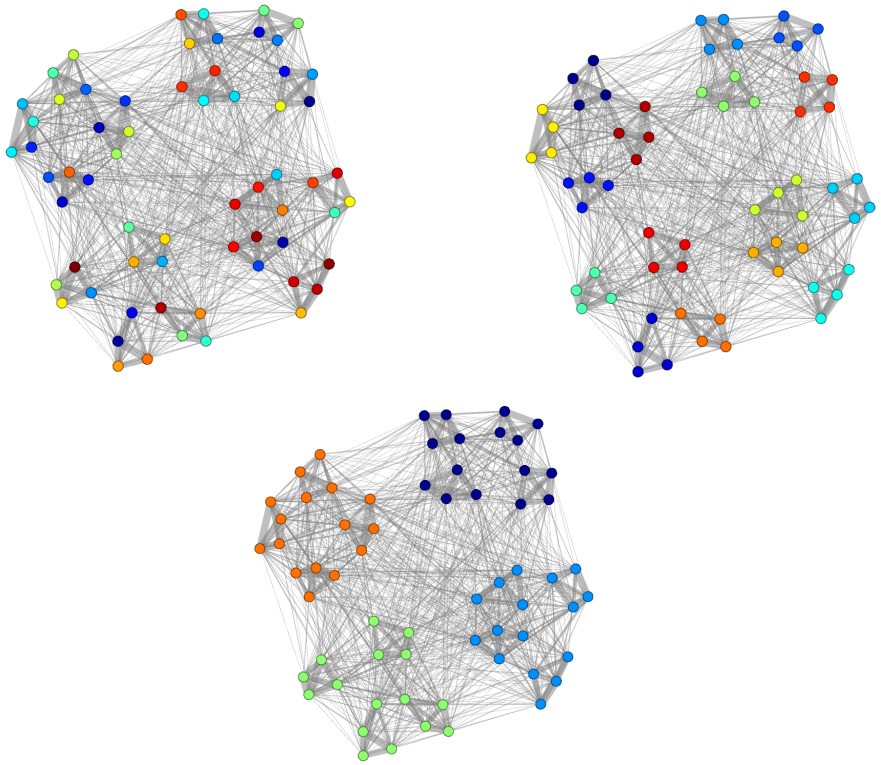
Using wavelets

Multiscale community detection on more elaborate networks

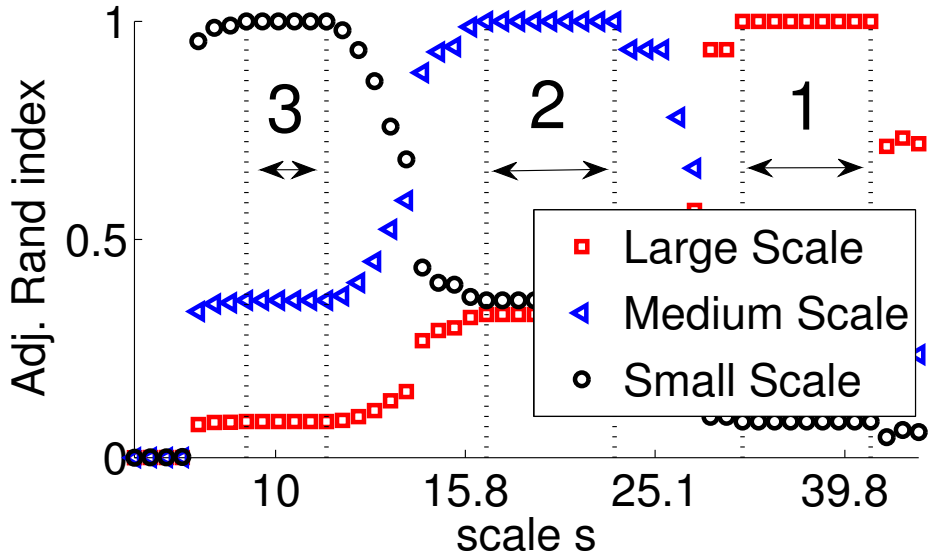


The Sales-Pardo benchmark

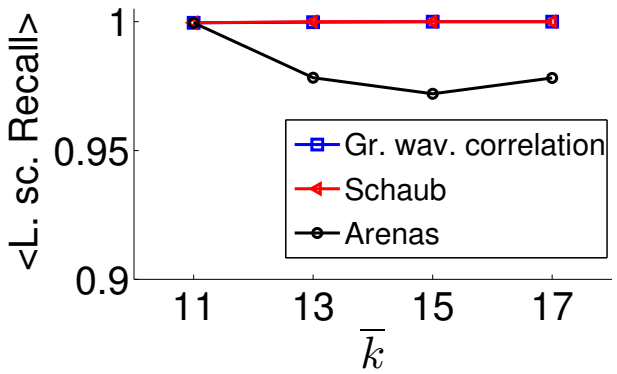
- Three community structures nested in one another
- Parameters:
  - sizes of the communities ($N = 640$)
  - ρ tunes how well separated the different scales are
  - $\bar{k}$ is the average degree; the sparser is the graph, the harder it is to recover the communities.

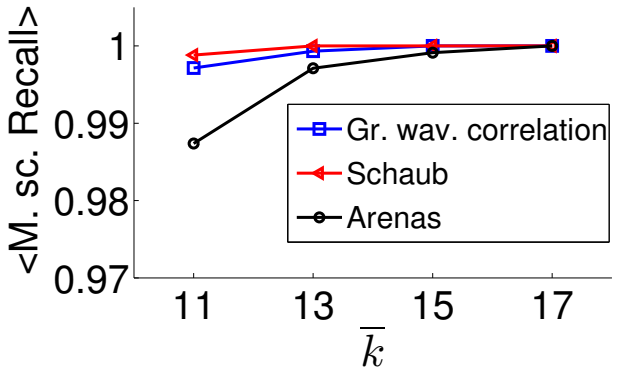


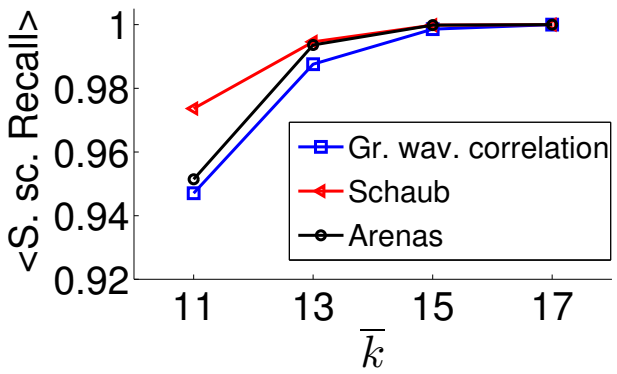
Results on the Sales-Pardo benchmark



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:

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

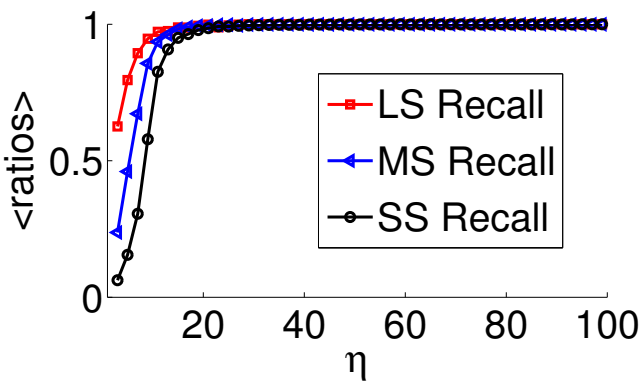
- Therefore, the sample correlation estimator $\hat{C}_{ab,\eta}$ satisfies:

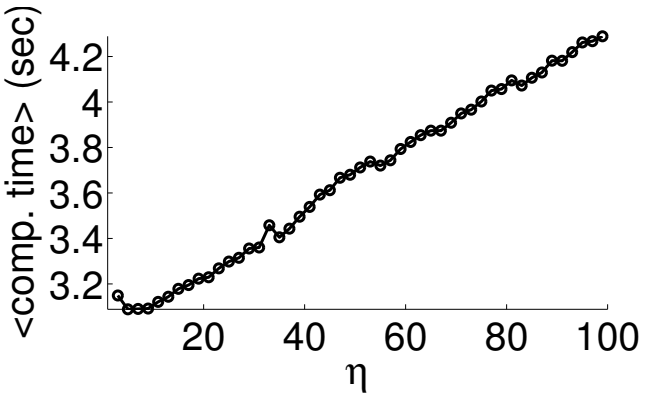
$$\lim_{\eta \to +\infty} \hat{C}_{ab,\eta} = \frac{\boldsymbol{\psi}_{s,a}^{\top}\boldsymbol{\psi}_{s,b}}{||\boldsymbol{\psi}_{s,a}||_2 ||\boldsymbol{\psi}_{s,b}||_2} = 1 - \boldsymbol{D}_s(a,b).$$

- 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

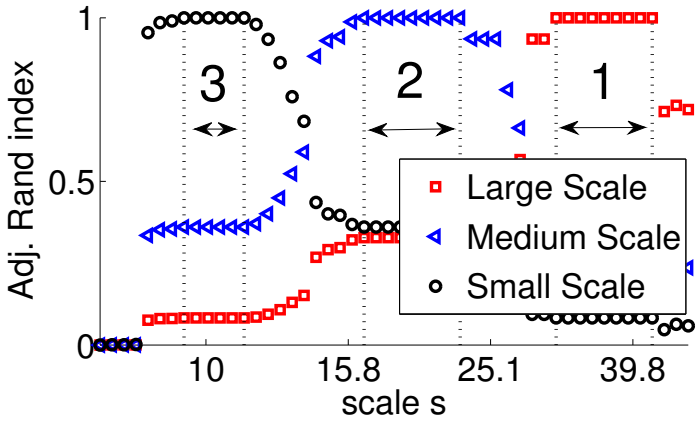
- Not all partitions are relevant: only those stable enough convey information about the network
- Lambiotte's approach to stability:
  Create B resampled graphs by randomly adding $\pm p\%$ (typically $p = 10$) to the weight of each link and computing the corresponding B sets of partitions $\{P_s^b\}_{b\in[1,B],s\in\mathcal{S}}$.
  Then, stability:

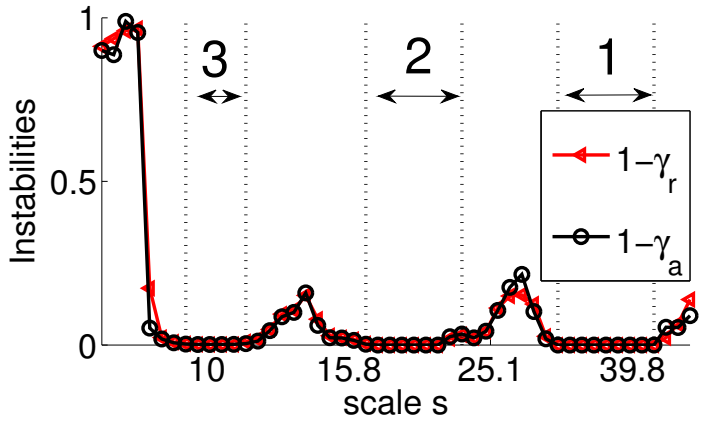
$$\gamma_r(s) = \frac{2}{B(B-1)} \sum_{(b,c)\in[1,B]^2, b\neq c} \texttt{ari}(P_s^b, P_s^c), \qquad (1)$$

- New approach: we have a stochastic algorithm.
  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

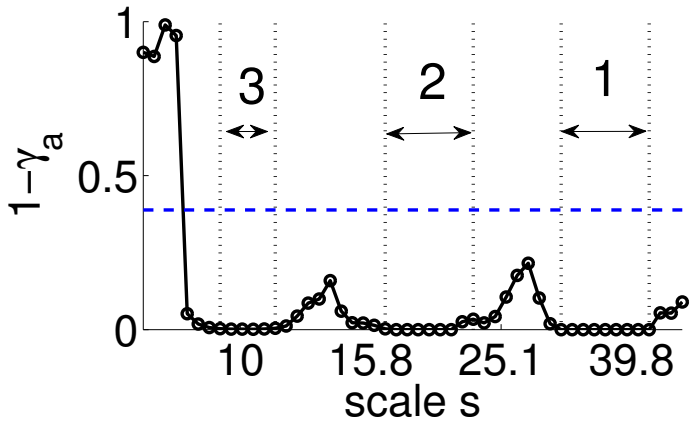




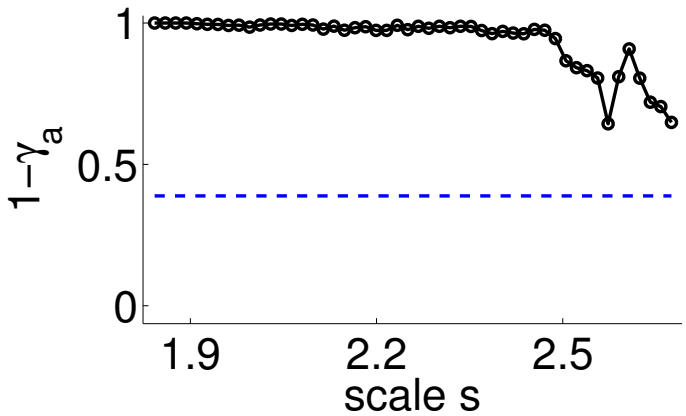
In addition: statistical test of relevance of the communities

- 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



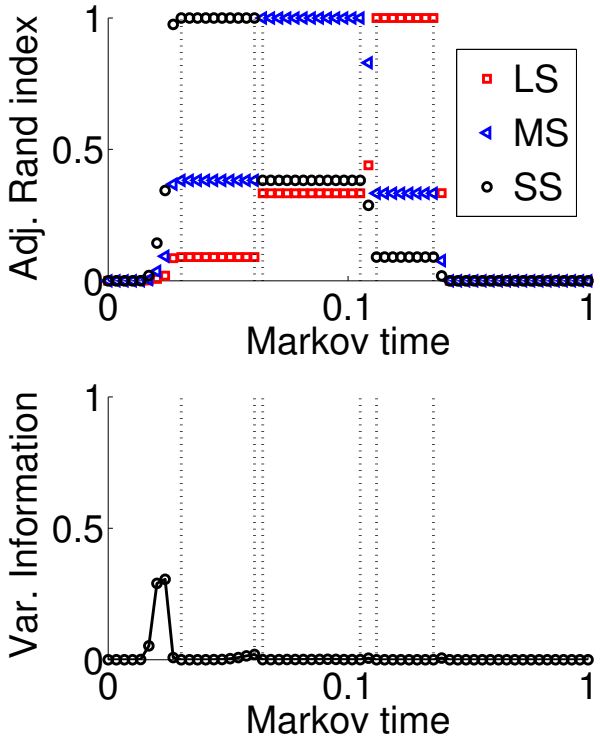
Chung-Lu graph



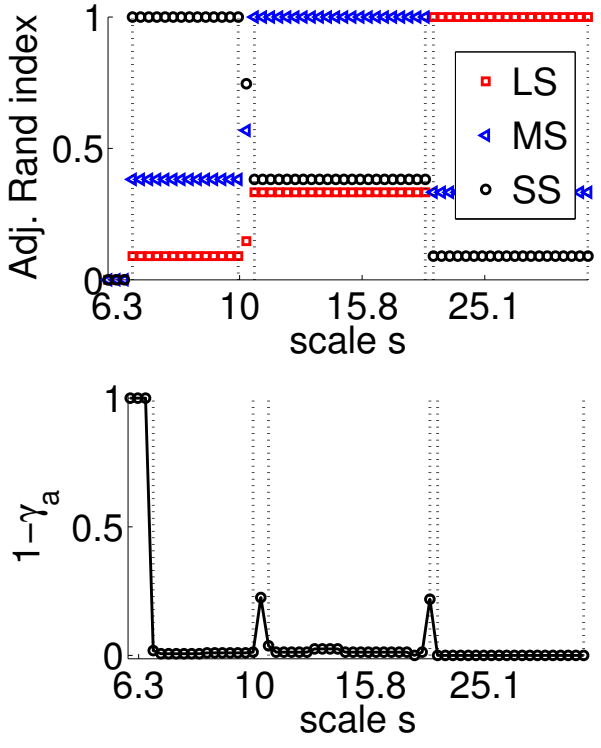
Comparison on larger Sales-Pardo graphs

$N = 6400$ nodes

Schaub-Delvenne



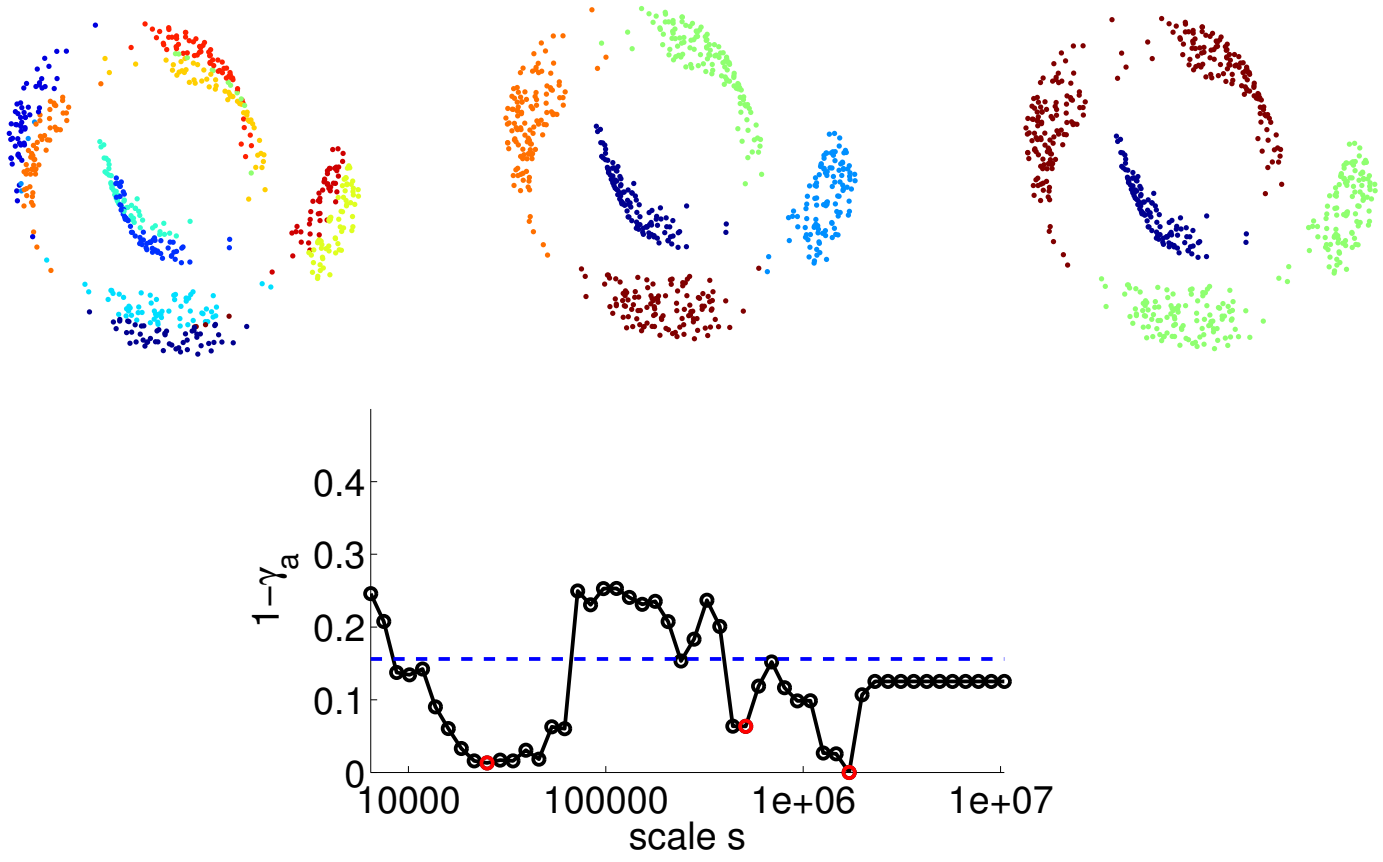
Wavelets



More elaborate graphs

Sensor network on the swiss roll manifold

- Three scale ranges of relevant community structure

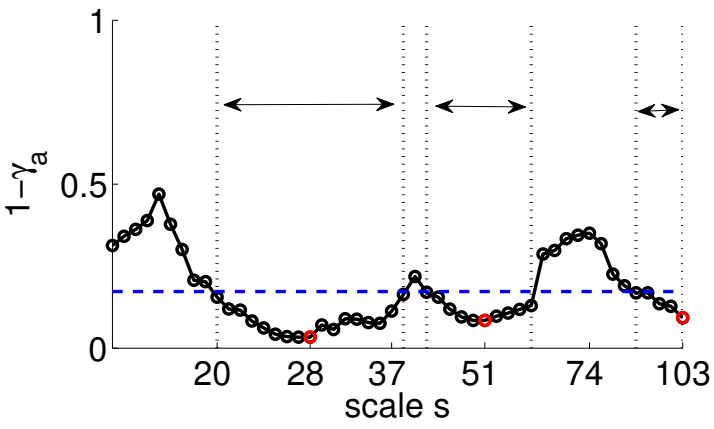


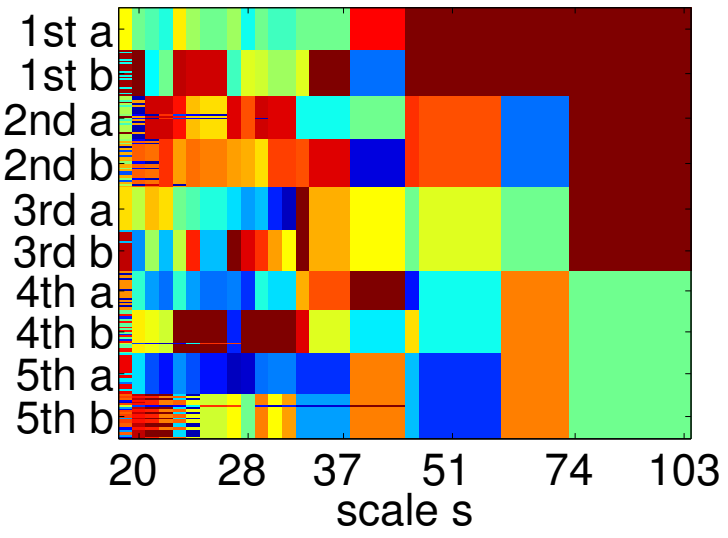
The dynamic social network of a primary school

Collaboration with A. Barrat (CPT Marseille)

Multi-scale Communities in Primary School

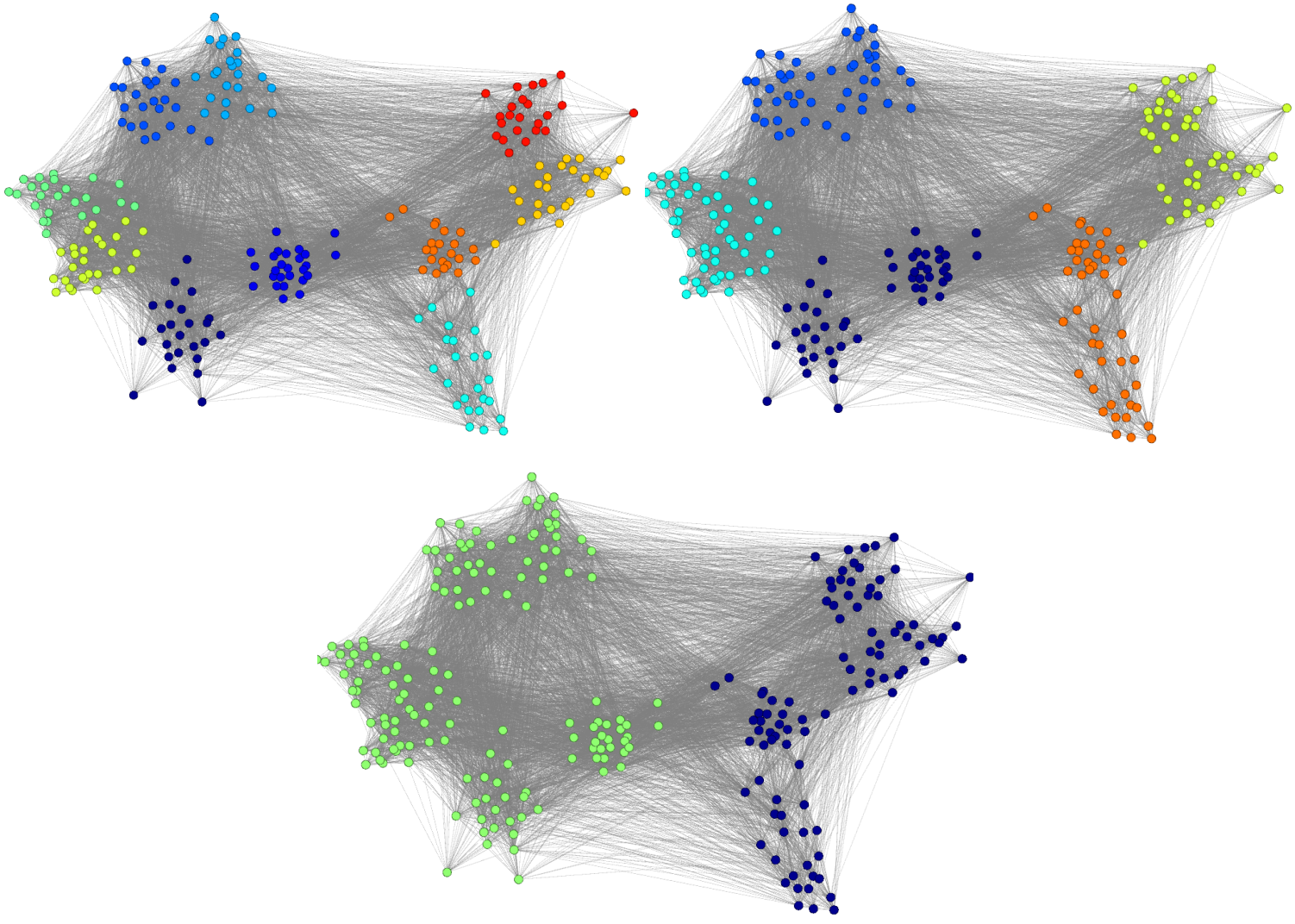
Collaboration with A. Barrat (CPT Marseille)





Multi-scale Communities in Primary School

Collaboration with A. Barrat (CPT Marseille)



Conclusion

- 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

- I hope that you were attracted to the emerging field of graph signal processing for networks.

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

Acknowledgements: thanks to Nicolas Tremblay for borrowing many of his figures or slides.