Scale-space module detection for irregular graphs

Bernard Chalmond, Xiaoyi Chen

Scale-space module detection for irregular graphs

Bernard Chalmond[†*] Xiaoyi Chen [†‡]

June 2013

Abstract

In the spirit of Lindeberg's approach for image analysis on regular lattice, we adapt the blob detection procedure to irregular graphs from a statistical point of view. We treat data observed on an undirected graph in the goal of detecting salient modules. This task consists in seeking subgraphs whose activity is strong or weak compared to those of their neighbors. This is performed by analyzing nodes activity at multi-scale levels. To do that, data are seen as the occurrence of a random field, for which we propose a multi-scale graphical modeling. In the framework of diffusion processes, the covariance matrix of the random field is decomposed into a sum of weighted graph Laplacians at different scales. Under the assumption of Gaussian law, the maximum likelihood estimation of the weights is performed that provides a set of relevant scales. As a result, we obtain a multi-scale decomposition of the random field on which the module detection is based. This method is experimentally analyzed on simulated data and biological networks.

Keywords. Blob Detection, Module Detection, Network Activity, Graphical Modeling, Scale-space Random Field, Graph Laplacian, Diffusion Kernel, Multi-scale Decomposition, Scale Selection

[*]CMLA, ENS-Cahan, 94235 Cachan Cedex, France
[†]Cergy-Pontoise University, France
[‡]Institut Pasteur, Systems Biology Lab, France

1 Introduction

This paper is related to the following general issue : given an undirected graph $G = (V, E)$ and an observation $\mathbf{x}$ of a random field $\mathbf{X}$ indexed by the nodes V of this graph, one searches for subgraphs $\{\mathcal{M}_k\}$ in V for which the respective observations $\{x_{\mathcal{M}_k}\}$ appear as salient profiles, in comparison to their surrounding in G. Such a subgraph with its profile $(\mathcal{M}_k, x_{\mathcal{M}_k})$ is called module. In the sequel, we use interchangeably the terms salient subgraph (or salient area) and salient profile for naming the modules. This concept depends on the context and what we seek. We focus in the following on a particular module called blob.

Blob detection on a lattice : a summary.

In the image analysis domain, when G is simply a regular lattice $\mathcal{L}$ inluded in $\mathbf{Z}^2$, the problem of salient area detection has received much attention and in particular for blob detection [14] as illustrated in Fig.1.

A blob can be likened to a spot and a simple model is given by the Gaussian profile [17]. In this case, we have the following result on which the scale-space blob detection is based. Consider an image $\mathbf{x} = \{x_s,\ s \in \mathcal{L}\}$ representing a Gaussian spot characterized by a width parameter λ_0 and centered at s_0 : $x_s \propto \exp -\|s - s_0\|^2/2\lambda_0$. [14] gives the following qualitative property that the spot center satisfies :

$$\frac{d}{d\lambda}\lambda[\Delta\widetilde{\mathbf{x}}^{(\lambda)}]_{s_0}\Big|_{\lambda=\lambda_0} = 0\ ,$$

where Δ denotes the discretized Laplacian operator on $\mathbf{Z}^2$ and $\widetilde{\mathbf{x}}^{(\lambda)}$ is the convolution of the image with Gaussian kernel $\mathcal{G}_\lambda$:

$$\begin{aligned} \widetilde{\mathbf{x}}^{(\lambda)}_s &= \sum_{s'} x_{s'}\ \mathcal{G}_\lambda(s, s')\ , \\ \mathcal{G}_\lambda(s, s') &\propto e^{-\|s-s'\|^2/2\lambda}\ . \end{aligned} \tag{1}$$

Hence, the derivative of the normalized Laplacian $\lambda\Delta$ is able to select the width λ_0 of the Gaussian spot. Essentially, $\lambda\Delta$ quantifies in some sense a curvature of the smoothed spot, and this curvature is optimal when $\lambda = \lambda_0$. This property is used to detect blobs in the image : the detected blobs are the local extrema of the discretized scale-space volume

$$\{\lambda[\Delta\widetilde{\mathbf{x}}^{(\lambda)}]_s,\ s \in \mathcal{L},\ \lambda \in \Lambda\}\ , \tag{2}$$

where Λ is a finite set of scales corresponding to an increasingly coarse sub-sampling of the regular grid.

From lattice to graph.

In the case of images, the salient areas are clearly detected by the human eye (Fig.1). In contrast, the salient subgraphs on a graph are hardly visible. In Fig.2 and Fig.3(a), without strong prior information, one cannot recognize specific modules. Furthermore,

our perception can be perturbed by the method used to display G. For instance, Fig.2 and Fig.3(a) arise from the same graph but using respectively the layouts "spring-embedded" and "force-directed" provided by the software Cytoscape.

In this paper, module refers to the extension of the blob concept to graphs. In a first step, this extension is straightforward since (1) is the solution of the heat equation on $\mathbf{Z}^2$ whose extension to irregular graphs is well known. However, extending the blob detection to undirected graphs requires some modifications with respect to scale and space. While for a regular lattice, it is natural to choose Λ from a sub-sampling of the grid, for any other graph this choice is much less trivial since the relevant scales are irregularly spread in $\mathbf{R}^+$, and the scale has no explicit dimension. In this goal, the multi-scale representation of the random field must be revisited in order to get a sparse representation $\{\mathbf{X}^{(\lambda)},\ \lambda \in \Lambda\}$ yielding a non-redundant decomposition of $\mathbf{X}$ in term of reconstruction : $\sum_\lambda \mathbf{X}^{(\lambda)} = \mathbf{X}$, a property that (1) does not satisfy. This property of non-redundancy is necessary for the identification of the right scales.

Related works : module and semantic module.

In image analysis, the module detection is used first for extracting areas of interest without using any strong prior information. These areas are then interpreted with greater precision or extended using high-level information in order to obtain semantic modules, as for object recognition [3].

This remark holds also in systemic biology where semantic modules correspond to biological modules (see [11, 4] among many others). The definition of biological modules does not rely solely on areas and profiles, but also uses complex biological knowledge. In several papers, the detection of biological modules operates in two stages: firstly, detection of module seeds, or more simply modules , and secondly refinement of the detected modules to finally obtain meaningful biological modules [21].

We comment some main approaches for module detection as introduced in the bioinformatic literature. Given a scoring function that allows to compute the importance of every subnetwork, finding the maximal-scoring connected subgraph is an NP-problem. In the seminal work [11], the main limitation is that node scores are treated independently since the subnetwork score is calculated as a sum of the node scores. To overcome this limitation, [4] proposes an inverse problem approach in which the node scores are modeled by a hidden Markov random field model under a constraint of regularity that is expressed by a bending energy. Two major well-known drawbacks are inherent to this approach [2] : the data-driven determination of the regularity scale (the trade-off parameter), and the energy minimization that requires stochastic optimization, a difficult computation task, already encountered in [11]. But conceptually, the main limitation of the Markovian model is that it is mono-scale, which is not suitable when the size of the modules is varying.

Instead of using the bending energy at a single scale, we propose to use it with a multi-scale formulation in order to adapt the scale to the module sizes. Technically, the advantage of this approach is twofold. First, the set of relevant scales can be estimated efficiently from the data. Second, we avoid the huge computation burden of the stochastic optimization. The computation is limited to scan a multi-scale representation of type (1)

by searching the differential local extrema as it is done for blob detection.

Method summary.

The first contribution of our paper is a multi-scale model and a data-driven method, which allow to estimate relevant scales from G and $\mathbf{x}$. With these relevant scales, the model leads to a decomposition of $\mathbf{x}$ from which a scale-space module detection is performed. There are three steps :

- Multi-scale modeling. Classicaly, the diffusion kernel $K = e^{-\lambda L}$ where L denotes the Laplacian of an undirected graph $G = (V, E)$, can be chosen to modeled the covariance structure of a random field $\mathbf{X}$ on G. However, instead of considering a single scale λ, we propose a multi-scale diffusion kernel $\bar{K} = \sigma_0^2 I_n + \sum_{j=1}^{r} \sigma_j^2 e^{-\lambda_j L}$. Each diffusion kernel $e^{-\lambda_j L}$ is weighted by a positive parameter σ_j^2 that is all the more great than the scale λ_j significantly contributes to the random field. Finally, we assume that the random field is governed by the Gaussian distribution $\mathbf{X} \sim \mathcal{N}(0, \bar{K})$.

- Multi-scale decomposition. The relevant scales $\Lambda = \{\lambda_j\}$ and their weights $\{\sigma_j^2\}$ are estimated using the maximum likelihood principle. Then, from the multi-scale model, we get the decomposition : $\mathbf{x} = \sum_j \mathbf{x}^{(j)}$, where $\mathbf{x}^{(j)}$ denotes the component at scale λ_j.

- Module detection. The detection is based on the scale-space representation $\{\lambda_j [L\mathbf{x}^{(j)}]_v \ , \ v \in V, \ \lambda_j \in \Lambda\}$ that quantifies the variations of the components in term of the graph Laplacian. The detection consists in searching local extrema of this scale-space function with respect to the spatial neighborhoods induced by G.

The model and the methodology are presented in Section 2. The multi-scale decomposition and the module detection are tested on simulated random fields and on real data in Section 3. We invite the reader to have a look on Fig. 6 that illustrates the method.

2 Models and Method

2.1 Random Field and Diffusion Process

This section summarizes a set of fundamental results on graph Laplacian and diffusion kernels. Consider a random field $\mathbf{X} = (X_1, ..., X_n)'$ observed on an undirected graph $G = (V, E)$. $V = \{v_1, ..., v_n\}$ denotes the node set and E the edges connecting them. The dependency structure between the random variables $\{X_i\}$ depends on the topological structure given by E. This dependency structure is here limited to a covariance structure modeled by a diffusion kernel [15], a choice explored in many domains and especially in pattern recognition, biological networks analysis and image processing [1, 19, 23].

For any observation $\mathbf{x}$ of $\mathbf{X}$, consider the energy of the local variations, called bending energy :

$$U(\mathbf{x}) = \sum_{i \sim j} (x_i - x_j)^2, \tag{3}$$

where $i \sim j$ denotes an edge, and the sum is taken over all edges $(i, j) \in E$. The energy U, which measures the smoothness of $\mathbf{x}$ on the graph, is widely used in several domains [22, 16]. It can be rewritten as

$$U(\mathbf{x}) = \mathbf{x}' L \mathbf{x}, \quad \text{with} \quad L_{i,j} = \begin{cases} -1 & \text{if} \quad i \sim j \\ d_i & \text{if} \quad i = j \\ 0 & \text{otherwise} , \end{cases} \tag{4}$$

where $d_i = \sum_j \mathbf{1}_{j \sim i}$ is the degree of node i, i.e. the number of edges connected to i. L is a symmetric positive semi-definite matrix, which can be written as $L = D - A$ where $D = \text{diag}[\{d_i\}]$ and A is the binary adjacency matrix (or connectivity matrix) defined by the edges. In (4) each edge $i \sim j$ carries a weight $a_{i,j} = 1$. This definition can be extended to the weighted case, where the weights are not necessarily all equal to 1. In both cases, we have $d_i = \sum_j a_{i,j}$.

We seek to represent $\mathbf{X}$ by a random field model on G, denoted $\mathbf{Y}(\lambda)$, whose covariance stucture depends on a scale parameter $\lambda > 0$. Essentially, this model is obtained by equalizing the variations due to a change of scale, with the spatial variations as follows :

$$Y_i(\lambda + d\lambda) - Y_i(\lambda) \doteq d\lambda \sum_{j \in V:\ j \sim i} (Y_j(\lambda) - Y_i(\lambda)) , \tag{5}$$

and in vector form :

$$\begin{aligned} \mathbf{Y}(\lambda + d\lambda) - \mathbf{Y}(\lambda) &= -d\lambda\, L\, \mathbf{Y}(\lambda) , \\ L &= D - A . \end{aligned} \tag{6}$$

(6) is the discretized version on G of the classical heat differential equation : [1]

$$\begin{cases} \frac{d}{d\lambda} \mathbf{Y}(\lambda) &= -L\, \mathbf{Y}(\lambda) , \\ \mathbf{Y}(0) &= \mathbf{X} . \end{cases} \tag{7}$$

whose solution is

$$\mathbf{Y}(\lambda) = K_\lambda\, \mathbf{X} . \tag{8}$$

$$K_\lambda = e^{-\lambda L} , \tag{9}$$

K_λ is a matrix exponential whose definition is $e^M = \sum_{i=0}^{\infty} \frac{M^i}{i!}$. For every node, one has :

$$Y_i(\lambda) = \sum_{j:\ j \sim i} K_\lambda(i, j) X_j = K_\lambda(i, .)\mathbf{X} , \tag{10}$$

[1] In the classic case of diffusion in $\mathbf{R}^2$, λ is a time parameter.

which is the generalization of (1). The exponential of a symmetric matrix providing a semi-definite positive matrix, the matrix K_λ, which is called diffusion kernel, can be used as a covariance matrix for modeling the covariance between the random variables $\{X_i\}$. The more λ is large, the more the off-diagonal effects in K_λ increase. λ is interpreted as a scale parameter and $Y_i(\lambda)$ as a scale-space random field on $V \times \mathbf{R}^+$.

For dimensionalty reduction applications, the multi-scale property of the diffusion kernel is well identified [13]. However, the choice of a single global parameter λ remains a difficulty [6]. For small λ, $K_\lambda(i,i)$ reflects local properties of G around the node v_i, while for large λ it captures some global structures. For instance, in the geometry processing field, the diagonal term $K_\lambda(i,i)$ has been used as a shape descriptor [18] by considering that for every λ, the local spatial extrema of this function provide a feature-based scale-space representation of shapes, usefull for shape matching.

2.2 Graphical Modeling

The outstanding issue at the end of the previous modeling step is the choice of λ. In other words what is the scale λ the most representative of the observed profile $\mathbf{x}$. In fact, several scales may explain this profile. Therefore, a natural approach consists of decomposing the random field $\mathbf{X}$ into r independant components according to a discrete set of relevant scales $\Lambda = \{\lambda_1, ..., \lambda_r\}$:

$$\mathbf{X} = \sum_{j=1}^{r} \mathbf{X}^{(j)} + \mathbf{X}^{(0)} \ , \tag{11}$$

where $\mathbf{X}^{(j)}$ denotes the component at scale λ_j and $\mathbf{X}^{(0)}$ a residual, [10, 20]. The decomposition (11) is related to the additive spline models introduced by Wahba [22] chap.10, and later reintroduced under the name of multiple kernel. Note that $\mathbf{X}^{(j)}$ does not match $\mathbf{Y}(\lambda_j)$ in (8), since the sum $\sum_j \mathbf{Y}(\lambda_j)$ over a given set of scales does not reconstruct $\mathbf{X}$.

In our approach, the covariance matrix $\mathrm{Cov}(\mathbf{X}^{(j)})$ of every component is modeled from the diffusion kernel (9). So we use r kernels $\{K_{\lambda_1}, ..., K_{\lambda_r}\}$ denoted $\{K_1, ..., K_r\}$, such that $\mathrm{Cov}(\mathbf{X}^{(j)}) = K_j = \sigma_j^2 \kappa_j$ where κ_j is given by (9) at scale λ_j. Due to the independance of the components, the covariance matrix $\mathrm{Cov}(\mathbf{X})$ is the following multi-scale diffusion kernel :

$$\begin{aligned} \bar{K}_{\sigma,\Lambda} = \sum_{j=0}^{r} K_j \ &= \sum_{j=0}^{r} \sigma_j^2 \kappa_j \\ &= \sum_{j=1}^{r} \sigma_j^2 e^{-\lambda_j L} + \sigma_0^2 I_n \ . \end{aligned} \tag{12}$$

Each kernel κ_j is weighted by a positive parameter σ_j^2 that is all the more great than the scale λ_j significantly contributes to the random field $\mathbf{X}$. The covariance matrix K_0 is that of a white noise. If we assume that the dependency structure of the random variables $\{X_i\}$

is uniquely described by its kernel, then it is legitimate to consider that $\mathbf{X}$ is distributed according to the Gaussian law

$$\mathbf{X} \sim \mathcal{N}(0, \bar{K}_{\sigma,\Lambda}) \, . \tag{13}$$

The scales $\{\lambda_j\}_{j=0}^r$ and their associated weights $\sigma \doteq \{\sigma_j^2\}_{j=0}^r$ are unknown parameters that are estimated using the maximum likelihood principle [2].

• *Weight estimation.* For a given Λ, let $\ell(\sigma|\Lambda) = \log(p_{\sigma,\Lambda}(\mathbf{x}))$ be the log-likelihood of σ, where $p_{\sigma,\Lambda}$ denotes the probability density of $\mathbf{x}$. Given an observation $\mathbf{x}$ and the Gaussian $\mathcal{N}(0, \bar{K}_{\sigma,\Lambda})$, the log-likelihood is

$$\ell(\sigma|\Lambda) = -\log|\bar{K}_{\sigma,\Lambda}| - \mathbf{x}' \bar{K}_{\sigma,\Lambda}^{-1} \, \mathbf{x} + Cte \, . \tag{14}$$

The maximum likelihood estimate is computed under the constraint of positivity of the parameters σ :

$$\hat{\sigma}(\lambda) = \underset{\sigma}{\operatorname{argmax}} \, \ell(\sigma|\Lambda) \;\; \text{under the constraint} \;\; \sigma > 0 \, . \tag{15}$$

For moderate sizes of n, the non-linear programming algorithms using gradient descent techniques are operational. For larger dimensions, the computation of the determinant $|\bar{K}_{\sigma,\Lambda}|$ and the inverse $\bar{K}_{\sigma,\Lambda}^{-1}$ becomes more difficult [12]. To reduce the amount of computation, one might also wonder whether it would be possible to remove the term $\log|\bar{K}_{\sigma,\Lambda}|$ in the likelihood, in order to work only with the generalized least-squares $\mathbf{x}' \bar{K}_{\sigma,\Lambda}^{-1} \, \mathbf{x}$. Theoretically, we know that this estimate is not statistically consistent [7]. Our experiments have confirmed this defect, by showing severe aberrations in the multi-scale decompositions (cf. Section 3.1).

• *Scale estimation.* A procedure for selecting the set Λ is now required. To do that, we choose a large number of scales $\Lambda^0 \doteq \{\lambda_j^0, \, j = 1, ..., r^0\}$ among which we seek to select $r < r^0$ relevant scales. Λ^0 can be seen as the scale domain. For this set we compute $\hat{\sigma}(\Lambda^0)$, that is the solution (15). To determine r, we perform a diagonalization of the covariance matrix

$$\widetilde{K} = \sum_{j=1}^{r^0} \hat{\sigma}_j^2(\Lambda^0) e^{-\lambda_j^0 L} \, , \tag{16}$$

of which we retain only the r largest eigenvalues $\nu_1 \geq ... \geq \nu_r$ according to the criteria

$$\frac{\sum_{i=1}^{r} \nu_i}{\sum_{i=1}^{r^0} \nu_i} = 1 - \epsilon \, , \tag{17}$$

where ϵ is a positive parameter chosen close to 0, typically $\epsilon = 0.01$ or 0.025. This criterion is related to that used in Principal Component Analysis [9]. It means the dispersion

[2] For notational convenience, we introduce $\lambda_0 = 0$ that is associated to K_0.

of $\mathbf{X}$ can be approximatively represented by r linearly independent components with an information loss determined by ϵ. Finally, from the estimated r, we can then achieve the selection of relevant scales Λ. These scales are associated with the r largest $\hat{\sigma}_j^2(\Lambda^0)$, i.e. the scales whose components are the most involved in the dispersion of $\mathbf{X}$. These scales are denoted $\{\lambda_{i_1}^0, ..., \lambda_{i_r}^0\}$ or more simply $\{\lambda_1, ..., \lambda_r\}$. As may be seen experimentally, the estimates $\hat{\sigma}_j^2$ associated to the scales in $\Lambda^0 \backslash \Lambda$ have values close to 0. This selection therefore allows a pruning of non-significant scales, what evokes the Ridge regression [9].

• *Statistical Multi-scale Decomposition.* The multi-scale model provides a multi-scale decomposition of $\mathbf{X}$ according to $r+1$ components denoted $\mathbf{X}^{(j)}$. Component estimation is closely linked to the scale selection problem.

Proposition 1 *Consider the model (11) based on the scale domain Λ^0 and assume this model is over-parameterized, i.e. the decomposition depends only on $r \leq r^0$ scales. Hence, given an observation* **x***, the Bayesian estimation provides the scale components :*

$$\hat{\mathbf{x}}^{/j} = K_j U D_\nu^{-1} \hat{\mathbf{b}} \ , \ \forall j = 1, ..., r \tag{18}$$

$$\hat{\mathbf{b}} = (\sigma_0^2 D_\nu^{-1} + I_r)^{-1} U' \mathbf{x} \ , \tag{19}$$

where D_ν is the diagonal matrix of the r largest eigenvalues $\nu_1 \geq ... \geq \nu_r$ of the covariance matrix $\widetilde{K} = \sum_{j=1}^{r^0} K_j$ and U is the matrix of the associated orthonormal eigenvectors : $\widetilde{K} U = U D_\nu$.

The proof is given in Appendix 5.1.

2.3 Module Detection

Given a graph G and an observation **x** of the random field, we first compute the estimated scales $\{\lambda_j, \ j = 1, ..., r\}$ and the associated decomposition (18) $\{\mathbf{x}^{(j)}, \ j = 1, ..., r\}$ as presented in the previous section. Rather than considering directly the components, we consider their spatial variations with respect to the graph Laplacian L :

$$\begin{aligned} (L\mathbf{x}^{(j)})_v &= \sum_{i:\ i \sim v} (x_v^{(j)} - x_i^{(j)}), \ \ \forall v \in V \\ &= d_v x_v^{(j)} - \sum_{i:\ i \sim v} x_i^{(j)} \ . \end{aligned} \tag{20}$$

This specifies the regularity of each component. $L\mathbf{x}^{(j)}$ is all the more great positively (resp. negatively) than the expression $x_v^{(j)}$ of the node v is strongly increasing (resp. decreasing) with respect to its neighbors. Therefore we look for nodes that are most differentially expressed with respect to L, and this by examining the expression of the components at different scales. Since the amplitude of variations of $L\widetilde{\mathbf{x}}^{(j)}$ decreases when the scale increases, a specific normalization is required. As in the case of blob detection (2) on a lattice $\mathcal{L}$, an efficient normalization is $\lambda_j L\widetilde{\mathbf{x}}^{(j)}$. A scale λ_j for which $\lambda_j (L\mathbf{x}^{(j)})_v$

is a local extremum with respect to scale and space, is seen as reflecting a module at position v and scale λ_j. This implies the following procedure.

For any node $v \in V$, we denote by $N_v^k \subset V$ the relative nodes of v of order k, ($k = 1, ..., \kappa$). $k = 1$ means the nearest neighbors (NN), $k = 2$ means the nearest neighbors of v to which their NN are added, etc... The module detection consists in searching local optima of the components with respect to the neighborhoods as follows[3] :

$$\begin{aligned} \forall\, v \in V \;:\; (j(v), k(v), v'(v)) = \underset{j,k,v' \in N_v^k}{\operatorname{argopt}} \lambda_j (L\mathbf{x}^{(j)})_{v'} \;; \\ \text{if } v'(v) = v, \text{ then } v \text{ is a module center at scale } \lambda_{j(v)} \;. \end{aligned} \tag{21}$$

When a module center is detected at v, its area is defined by the subgraph $\{v, N_v^{k(v)}\}$.

3 Experiments

Recall that a module is a couple $(\mathcal{M}, x_{\mathcal{M}})$ where $\mathcal{M}$ is a subgraph of G. The regular lattices $\mathcal{L}$ do not show particular structure like stars or clusters, unlike the case of irregular graphs G. Let us give an example of graph showing a particular structure. The graph structure is organized around known subgraphs $\{R_k = (V_k, E_k)\}$ called regulons (or hubs). A regulon is a set of nodes $V_k \subset V$ connected to one or several common nodes, called regulators. A regulon can be connected to several regulators and a regulator can be connected to several regulons. Fig.4 shows a graph with four regulons and four regulators.

In practice, such regulons can be used a posteriori for interpreting the detected modules or inferring semantic modules. Depending on the profile of $\mathbf{x}$, the area $\mathcal{M}$ of a module can be simply a regulon, a subregulon or the union of several regulons. Fig.5 shows a time-series $\{\mathbf{x}_1, ..., \mathbf{x}_{11}\}$ of a random field $\mathbf{X}$ observed on the graph of Fig.4. The colors depict the output of the scale-space module detection performed on every $\mathbf{x}_t$. We see the evolution of the modules. For instance at $t = 8$ there are three modules whereas at time $t = 9$ there are two modules. Using biological knowledge it would remain to interpret these modules as biological modules.

3.1 Evaluation on Simulated Data

3.1.1 Simulation Procedure

For phenomena of high complexity, simulated data are an important preliminary support for modeling when we do not have data with sufficient knowledge of the "ground truth". The simulation of the random field $\mathbf{X}$ requires to give the ground truth, consisting of a graph $G = (V, E)$ and the parameters (Λ, σ). In our procedure, G is organized in regulons : $G = \uplus_{k=1}^{m} R_k$. Here, we assume for simplicity that each regulon R_k is associated to only one regulator r_k. The symbol $+$ in $\uplus$ indicates that the regulons are mutually connected. This high-level of connection is equivalent to a graph G^B between the regulators : $G^B = (\{r_k\}, E_r)$.

[3] $v'(v)$ also depends on k, what is omitted to simplify the writing.

The simulation is done in two steps. First, the simulation of a graph G consisting of m regulons is done as described in Appendix 5.2. Second, a sample $\mathbf{x}$ of $\mathbf{X} \sim \mathcal{N}(0, \bar{K}_{\Lambda,\sigma})$ is drawn. To do that, we simply simulate $\boldsymbol{\alpha} \sim \mathcal{N}(0, I_n)$ since the diagonalization $\bar{K}_{\Lambda,\sigma} = VDV'$ implies $VD^{1/2}\boldsymbol{\alpha} \sim \mathcal{N}(0, \bar{K}_{\Lambda,\sigma})$.

3.1.2 Simulated Data

Fig.6(a) shows the inter-regulon graph G^B and Fig.6(b) the graph G. Each regulon has its own color. Fig.6(c) displays an observation $\mathbf{x}$ of the random field $\mathbf{X}$ on G, and Fig.7 shows its 1-D profile. In this experiment, each regulon is a potential module since the simulation procedure is based on a regulon structure. $\mathbf{x}$ was simulated using the multi-scale kernel (12) with 3 scales : $\lambda_1 = 8,\ \lambda_2 = 14,\ \lambda_3 = 24$ and $\sigma_1 = \sigma_2 = \sigma_3 = 1$.

3.1.3 Data Analysis

The estimation and detection tasks are illustrated in the figures 6, 7, 8 and 9. The maximum likelihood estimation (15) was performed using the scale domain $\Lambda^0 = \{2k, k = 0, 1, ..., 15\}$. Fig.8 displays the statistical multi-scale decomposition. The continuous black line connecting the data points is the sum $\sum_{k=1}^{15} \mathbf{x}^{(2k)}$ of all the components except the noise component $\mathbf{x}^{(0)}$. Since this line interpolates the data points, this means that the estimated noise component is very low. The selected Λ is computed from this decomposition using (17) with $\epsilon = 0.025$. The three main components associated to $\{\lambda_1 = 8, \lambda_2 = 10, \lambda_3 = 30\}$ are shown in Fig.7. It is interesting to compare this statistical decomposition with the ordinary scale-space decomposition (8) shown in Fig.9. The statistical decomposition has the ability to focus more clearly on the spectral content of $\mathbf{x}$. $\lambda_3 = 30$ reflects low frequencies whereas $\lambda_2 = 10$ contributes to high frequencies. But above all, it remedies the redundancy of ordinary scale-space representation, and therefore favors the identification of the right scales.

The detected modules shown in Fig.6(d) correspond to the rule (21). In Fig.7, the locations of the detected extrema are indicated by red circles. There is exactly one detected module per regulon. Given the knowledge of the regulons, we can therefore decide that the semantics modules are composed of the regulons.

Above, we have mentioned the prominent role of the term $\log|\bar{K}_{\sigma,\Lambda}|$ in the likelihood. This is confirmed experimentally. Without this term, the estimation-detection procedure was repeated on the same data as previously. The multi-scale decomposition is then quite inaccurate. All weights are very high and the scale components are close to zero[4].

3.2 Bacillus Subtilis Data

Fig.3 illustrates the multi-scale decomposition of a field $\mathbf{x}$ that represents gene expressions of Bacillus Subtilis. The underlying graph $G = (V, E)$ comes from the regulatory network of the bacterium. V denotes genes, E connections between genes and $\mathbf{x}$ gene

[4]For lack of space, the figure that illustrates this drawback is not given.

expressions on V (Fig.3-a), [5]. The entire graph contains 1607 genes, 2345 edges and 132 regulons. Fig.4 displays four connected regulons extracted from this network. In steady-state, gene expressions are assumed to be governed by the model (13). In Fig.Fig.3 we see the structuring effects of the method in terms of gene grouping as this had already been shown for other regulatory networks [6]. Fig.2 illustrates the module representation enlightening the regulon structure. It is still difficult for a non expert to visually analyze such a network and all the more than this display depends on the method type used to display it.

In order to illustrate the potential utility of our method, we present a case study. The expression of the regulons depends on the nutritional environment of the bacteria over time, some of them are over-expressed and other ones are under-expressed. With the module detection, we search to identify regions of the graph that are particularly expressed. Fig.5 illustrates this detection on a time series of the random field observed during $T = 11$ time points. The detection has been done at every time t, independently of the others. Every detected module is composed of one or several regulons. By analysing the evolution of the detected modules over time, we search to understand some aspects of the biological behavior of B.subtilis. For instance, we see that the yellow module, corresponding to nodes related to *comK*, becomes mainly activated from the time point $t = 7$. What could we say about it? Appendix 5.3 gives some elements to interpret this phenomenon.

4 Concluding remarks

This paper proposes and implements a multi-scale graphical modeling for random vectors observed on an undirected graph. The result is a multi-scale decomposition of the random field which provides an analysis tool to deal with specific treatments because it allows to select relevant scales. This tool is especially used for the detection of module. With hindsight, this detector seems relatively simple. However, emphasis has been put on a coherent modeling without tunable parameters and heuristics.

The experiments show that module detection puts into light the activated modules and therefore provides a mean to study dynamic random fields. However, module detection on time series has been performed without taking into account time dependence. At every time t, the observation of the random field has been treated independently of the others. In the B.subtilis example, we conjecture that the lack of dependence in the model is the reason why the detection at time $t = 8$ does not meet the assumption of expression stability from time $t = 7$. Nevertheless, it is well known that Markovian dependence can highly improve the sensitivity of the detection of isolated low signal. A next paper in Spatial Statistics Journal will present a Markovian spatio-temporal modeling that generalizes the present model and for which the detection at time $t = 8$ could satisfy the stability assumption.

[5] The biological network has been simplified by removing the protein level network and therefore in G the regulatory protein-encoding genes and their proteins are confused. Futhermore, the edge directions in E have been deleted. Consequently, we cannot speak strictly of regulation in the sense of regulatory networks. However, for the detection of modules of salient profiles, this simplification has no severe impact.

5 Appendix

5.1 Proof of Proposition 1

Let $\widetilde{\mathbf{X}} = \mathbf{X} - \mathbf{X}^{(0)} = \sum_{j=1}^{r^0} \mathbf{X}^{(j)}$, and recall the spectral equation $\widetilde{K}U = UD_\nu$ where $\widetilde{K} = \sum_{j=1}^{r^0} K_j$. First, since the columns of U are independent, we can write $\widetilde{\mathbf{X}} = U\mathbf{B}$ where $\mathbf{B}$ is a r-random vector. Then, the spectral equation allows to rewrite

$$\begin{aligned} \widetilde{\mathbf{X}} = U\mathbf{B} = \widetilde{K}UD_\nu^{-1}\mathbf{B} &= \sum_{j=1}^{r^0} K_j U D_\nu^{-1}\mathbf{B} = \sum_{j=1}^{r^0} \mathbf{X}^{/j}, \\ \text{where} \quad \mathbf{X}^{/j} &= K_j U D_\nu^{-1}\mathbf{B}\,, \end{aligned} \tag{22}$$

which provides the components (18). Second, $\widetilde{\mathbf{X}} = U\mathbf{B}$ implies the covariance matrix

$$\text{Cov}(\mathbf{B}) = U\widetilde{K}U' = D_\nu\,. \tag{23}$$

For a given observation $\mathbf{x}$, the Bayesian estimation of the occurrence of $\mathbf{B}$ consists in maximizing the log-likelihood $\log p(\mathbf{b} \mid \mathbf{x}) = \log p(\mathbf{x} \mid \mathbf{b}) + \log p(\mathbf{b}) + Cte$. Given the Gaussian laws $\mathbf{B} \sim \mathcal{N}(0, D_\nu)$ and $\mathbf{X}^{/0} \sim \mathcal{N}(0, \sigma_0^2 I_n)$, this amounts to compute

$$\hat{\mathbf{b}} = \underset{\mathbf{b}}{\text{argmax}} -\frac{1}{\sigma_0^2}\|\mathbf{x} - U\mathbf{b}\|^2 - \mathbf{b}' D_\nu^{-1}\mathbf{b}\,, \tag{24}$$

which provides the expression $\hat{\mathbf{b}}$ in (19).

5.2 Graph simulation

A simulation of a graph G consisting of m regulons : $G = \uplus_{k=1}^m R_k$, is done in three steps.

1. For each set of nodes V_k making up a regulon, a regulon graph $R_k = (V_k, E_k)$ is simulated.
2. At a larger scale, the m regulons are considered as m nodes of a graph, and thus an inter-regulon graph G^B is simulated.
3. The global graph G is obtained on the basis of these $m+1$ graphs, as follows. For each regulon R_k, a regulator r_k is drawn uniformly at random in this regulon. This regulator regulates the regulon(s) $R_{k'}$ such that $k \sim k'$ in G^B, [6] . The weight of the connections between r_k and nodes v in $R_{k'}$ are given by the probabilities of the Binomial law $\mathcal{B}(|R_{k'}|, p)$ where $0 < p < 1$. When a weight is below a threshold τ, for example 0.05, the weight is set to zero, then the probability distribution is renormalized. By ruling p and τ, one can modulate the number of edges between the regulator and the regulated unit. In this case, r_k regulates a subset of nodes in $R_{k'}$.

[6] A node $v \in V$ is regulated by another nodes v' if X_v is significantly correlated to $X_{v'}$.

5.3 A case study

Under stressful environments such as nutrient limitations, some cells enter competence, which is the ability of a cell to copy exogenous DNA [8]. Here we examine a GM nutrient-shift experiment : an experimental population of cells grew first in Glucose and then Malate was injected. Gene expression time series were collected during $T = 11$ time points, [www.basysbio.eu].

In fact, we know that the competence has two characteristics : first, a high level of *comK* expression, and second, a negative correlation between *comK* and *comS* (see Figure 4 in [5]). In Fig.10, which displays gene profiles of *comK* and *comS* for our GM data, we see that these two characteristics are verified in the case of our nutrient-shift experiment where negative correlation mainly appears from $t = 7$. Based on these characteristics, we are led to advance the hypothesis that in GM condition, bacteria cells enter competence at time $t = 7$. This is consistent with our module detection whose interpretation has yielded also this instant. These two identical results obtained by two different analyzes bring an element for reinforcing the validity of the hypothesis and that of the detector.

References

[1] Mikhail Belkin and Partha Niyogi. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56:209–239, 2004.

[2] Bernard Chalmond. *Modeling and Inverse Problems in Image Analysis*. Springer-Verlag, 2003.

[3] Bernard Chalmond, Benjamin Francesconi, and Stephane Herbin. Using hidden scale for salient object detection. *IEEE Trans. on Image Processing*, 15(9):2644–2656, 2006.

[4] Li Chen, Jianhua Xuan, Rebecca B. Riggins, Yue Wang, and Robert Clarke. Identifying protein interaction subnetworks by a bagging markov random field-based method. *Nucleic Acids Research*, 41(2), 2013.

[5] Sandra H. Dandach and Mustafa Khammash. Analysis of stochastic strategies in bacteria competence: a master equation approach. *PLoS Computational Biology*, 6(11):21027–21034, 2010.

[6] Guro Dorum, Lars Snipen, Margrete Solheim, and Solve Saebo. Smoothing gene expression data with network information improves consistency of regulated genes. *Statistical Applications in Genetics and Molecular Biology*, 10(1), 2011.

[7] Carlo Gaetan and Xavier Guyon. *Spatial Statistics and Modeling*. Springer-Verlag, 2009.

[8] Leendert W. Hamoen, Gerard Venema, and Oscar P. Kuipers. Controlling competence in bacillus subtilis: shared use of regulators. *Microbiology*, 1(49):9–17, 2003.

[9] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning*. Springer, 2009.

[10] Lasse Holmstrom, Leena Pasanen, Reinhard Furrer, and Stephan R. Sain. Scale space multiresolution analysis of random signals. *Computational Statistics and Data Analysis*, 55:2840–2855, 2011.

[11] Trey Ideker, Owen Ozier, Benno Schwikowski, and Andrew F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18(1):S233–S240, 2002.

[12] Harri Kiiveri and Frank de Hoog. Fitting very large sparse gaussian graphical models. *Computational Statistics and Data Analysis*, 56:2626–2636, 2012.

[13] Ann B. Lee and Larry Wasserman. Spectral connectivity analysis. *Journal of the American Statistical Association*, 105, 2010.

[14] Tony Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):77–116, 1998.

[15] Bojan Mohar. Some applications of laplace eigenvalues of graphs. In G. Hahn and G. Sabidussi, editors, *Graph Symmetry: Algebraic Methods and Applications,*, volume Ser. C 497, pages 225, 275. Kluwer, 1997.

[16] Jun Koo Park. Normal mode analysis and gaussian network model. Technical report, Iowa State University, 2008.

[17] Ihor Smal, Marco Loog, Wiro Niessen, and Erik Meijering. Quantitative comparison of spot detection methods in fluorescence microscopy. *IEEE Trans. on Medical Imaging*, 29(2):282–301, 2010.

[18] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Eurographics Symposium on Geometry Processing*, volume 28. Blackwell Publishing, 2009.

[19] Liang Sun, Shuiwang Ji, , and Jieping Ye. Adaptive diffusion kernel learning from biological networks for protein function prediction. *BMC Bioinformatics 9:162*, 9(162), 2008.

[20] Kevin Thon, Havard Rue, Stein Olav Skrovseth, and Fred Godtliebsen. Bayesian multiscale analysis of images modeled as gaussian markov random fields. *Computational Statistics and Data Analysis*, 56:49–61, 2012.

[21] Igor Ulitsky and Ron Shamir. Identification of functional modules using network topology and high-throughput data. *BMC Systems Biology*, 1(8), 2007.

[22] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[23] Fan Zhang and Edwin R. Hancock. Graph spectral image smoothing using the heat kernel. *Pattern Recognition*, 41:3328 – 3342, 2008.

[24] Jun Zhang, Partha Niyogi, and Mary Sara McPeek. Laplacian eigenfunctions learn population structure. *PloS One*, 4(12):1–6, 2009.

[25] Xiaojin Zhu, Jaz Kandola, John Lafferty, and Zoubin Ghahramani. Graph kernels by spectral transforms. In O. Chapelle, B. Scholkopf, and A. Zien, editors, *Semi-Supervised Learning*, 2006.

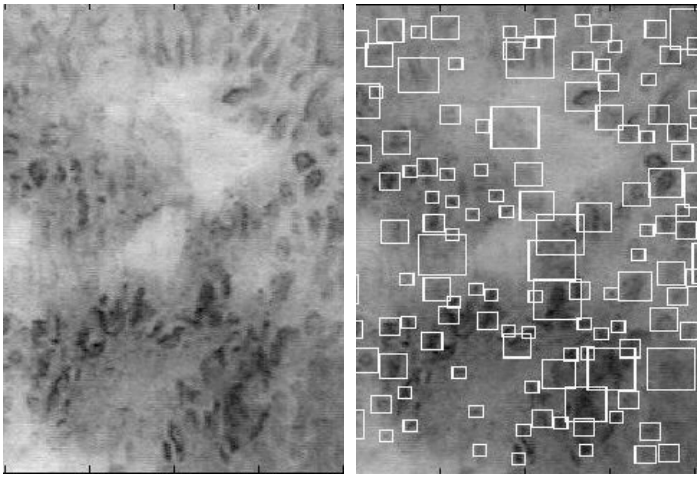


Figure 1: Scale-space blob detection on a dermoscopy image observed on a regular lattice (from a method inspired by [3]). For each detected area of interest, a thorough analysis can be later done by the expert in order to identify "semantic modules".

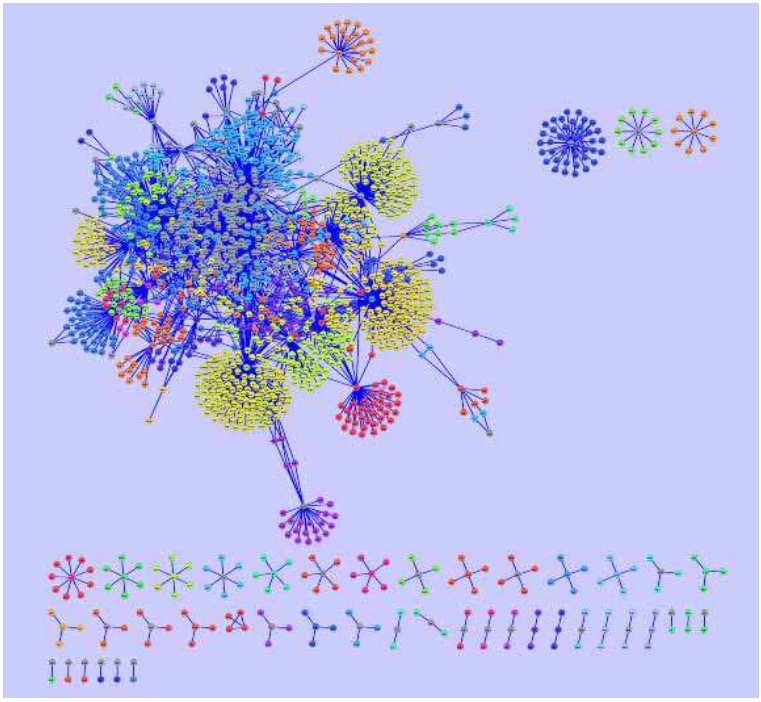


Figure 2: Modular network representation of the network of Fig.3. The pseudo color that highlights the regulons structure brings an help for the visual analysis of the network (see also Fig.6(c)).

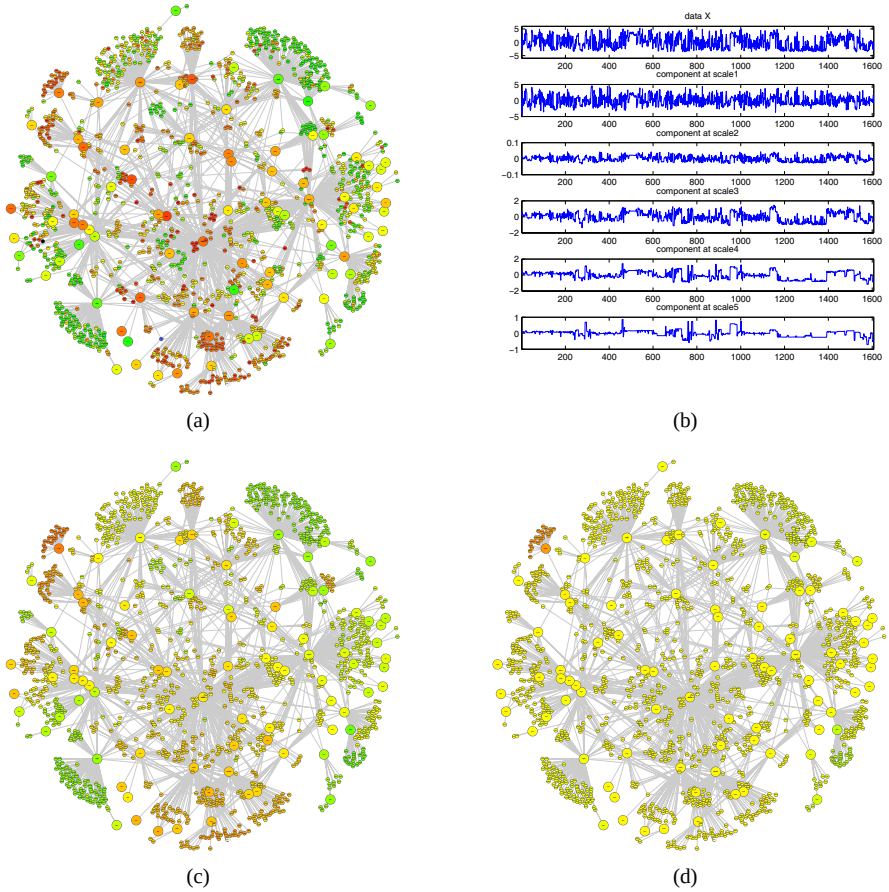


Figure 3: Multi-scale decomposition of gene expression network of Bacillus subtilis. (a) Original data. (b) Multi-scale decomposition profiles (1-D display). (c) Scale component for $\lambda = 2$. (d) Scale component for $\lambda = 16$. The decomposition has a structuring effect in terms of gene grouping.

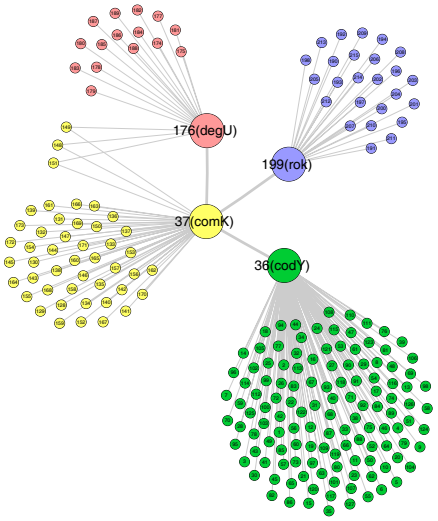
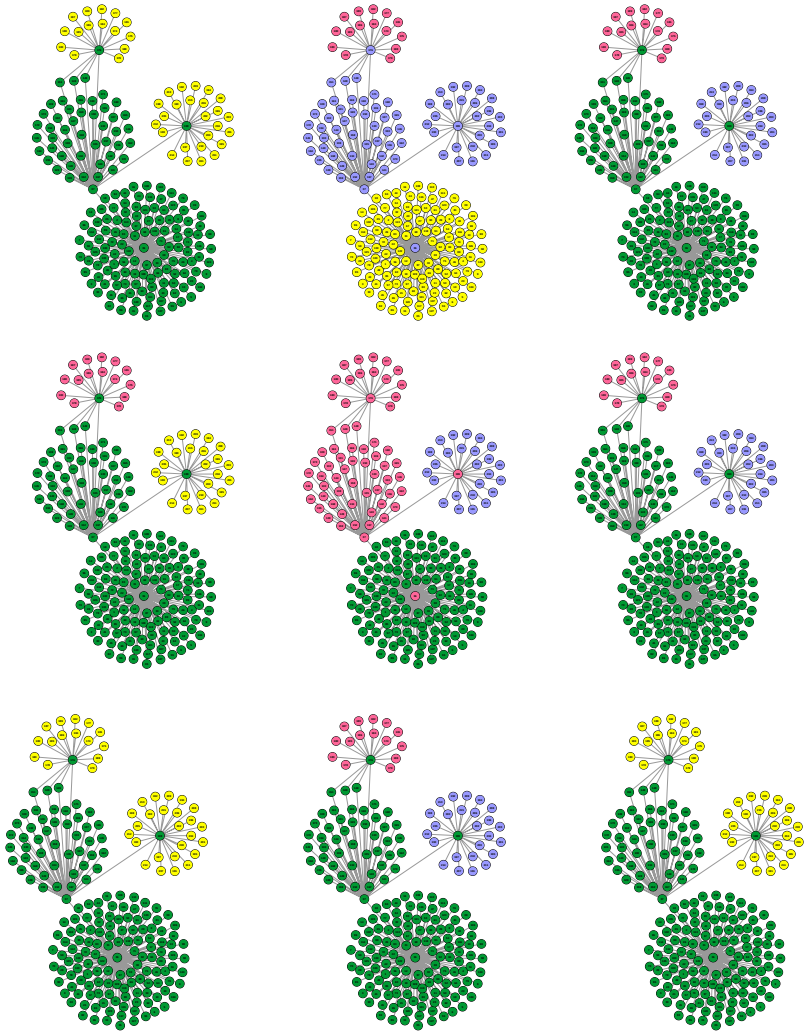


Figure 4: A graph consisting of four regulons and four regulators.



(a) t=7 (b) t=8 (c) t=9

(d) t=10 (e) t=11

Figure 5: Module detection for t=1,2,...,11. This treatment was done without taking into account time dependence. The detection at time $t = 8$ is different from that at times $t = 7, 9, 10, 11$. At these time points, two modules are detected : "green" and "yellow" modules.

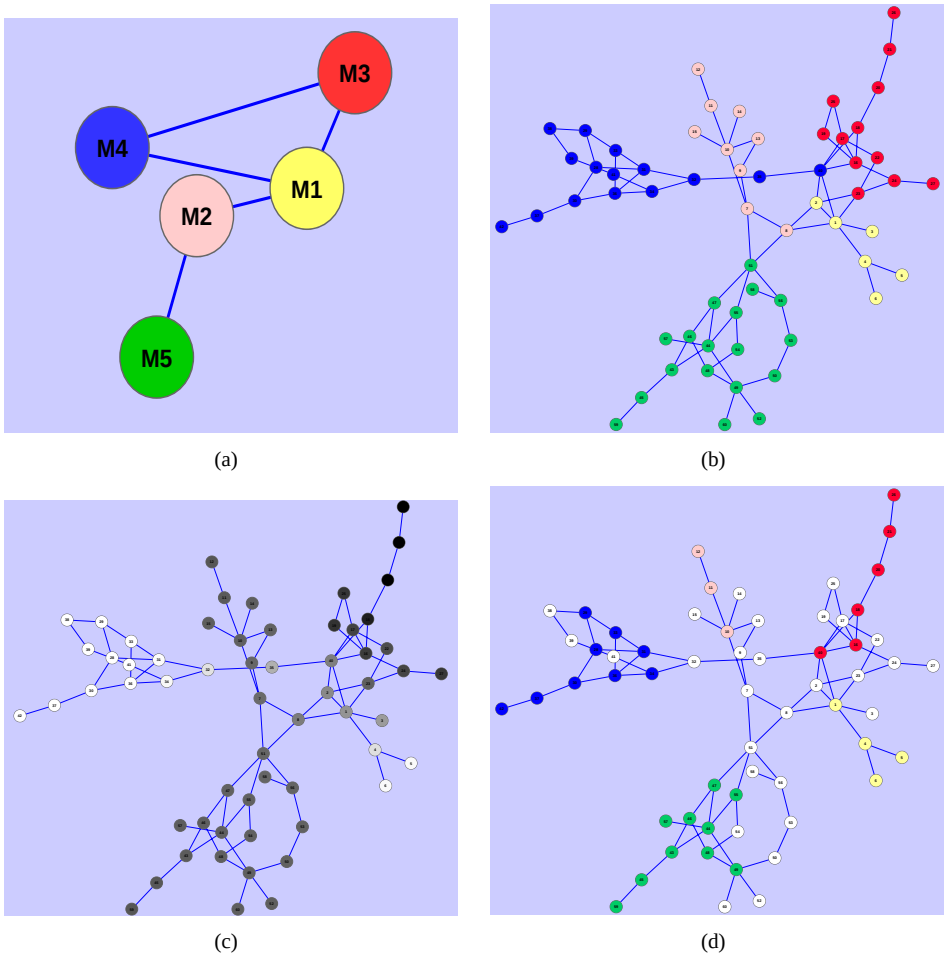


Figure 6: Experiment on simulated data. (a) The regulator graph G^B (between regulons). (b) The entire regulon graph G. (c) The observed random field $\mathbf{x}$. (d) Module detection outcome: there is one detected module per regulon. Given the knowledge of the regulons, we can therefore decide that each regulon is a semantic module.

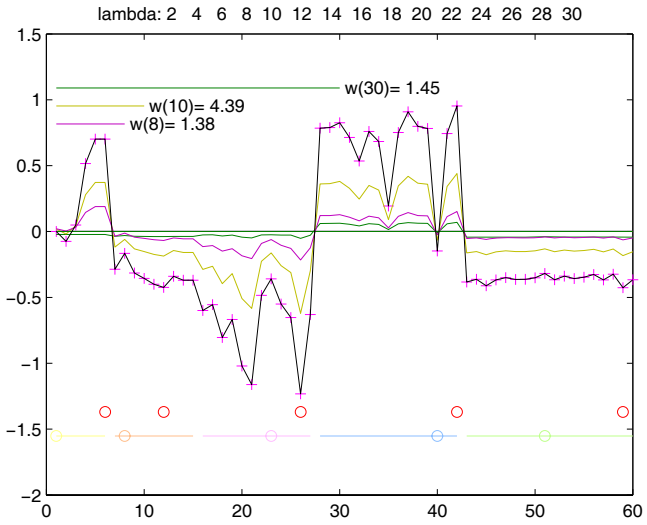


Figure 7: Module detection based on the multi-scale decomposition in Fig.8 restricted to Λ. The pink crosses show the 1-D profile of $\mathbf{x}$. The color curves display the three main components for scales $\{8, 10, 30\}$. The red circles are the locations of the detected module centers. The color segments at the bottom of the figure locate regulons, their colors are identical to those in Fig.6(a-b-d).

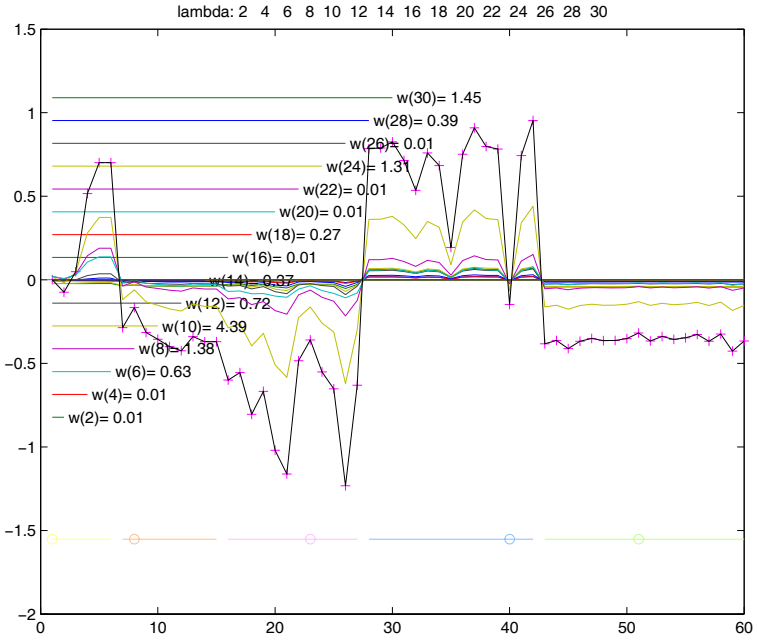


Figure 8: Statistical multi-scale decomposition for $\Lambda^0 = \{2, 4, ..., 28, 30\}$. Scale selection with $\epsilon = 0.025$ removes the scales $\{2, 4, 16, 20, 22, 26\}$ and keeps $\Lambda = \{6, 8, 10, 12, 14, 18, 24, 28, 30\}$. The information loss is negligeable since $\mathrm{w}(\lambda) = \hat{\sigma}^2_\lambda = 0.01$ for all $\lambda \in \Lambda^0 \backslash \Lambda$. The black curve is the sum of all Λ^0-components.

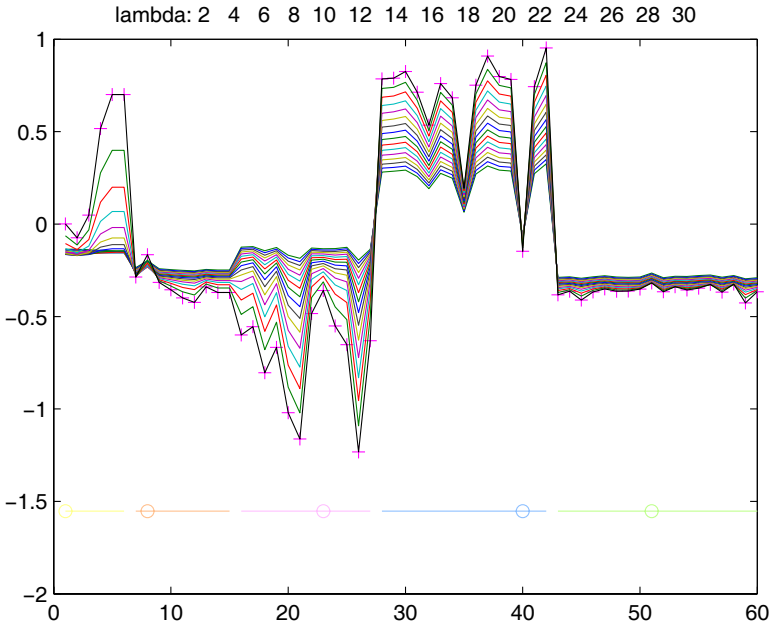


Figure 9: Ordinary scale-space decomposition.

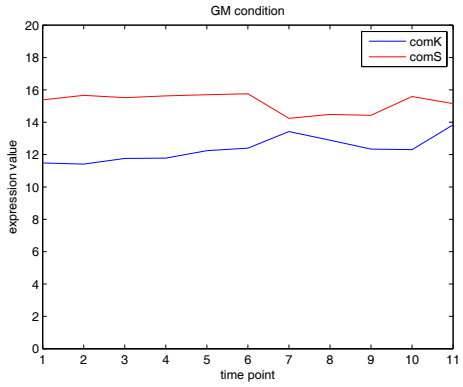


Figure 10: Negative correlation between *comK* and *comS* expressions in nutrient shift data GM.