

Structured sampling and fast reconstruction of smooth graph signals

Gilles Puy, Patrick Pérez

▶ **To cite this version:**

Gilles Puy, Patrick Pérez. Structured sampling and fast reconstruction of smooth graph signals. 2017. hal-01485647

HAL Id: hal-01485647
https://hal.science/hal-01485647

Preprint submitted on 9 Mar 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structured sampling and fast reconstruction of smooth graph signals

Gilles Puy and Patrick Pérez

Abstract. This work concerns sampling of smooth signals on arbitrary graphs. We first study a structured sampling strategy for such smooth graph signals that consists of a random selection of few pre-defined groups of nodes. The number of groups to sample to stably embed the set of k-bandlimited signals is driven by a quantity called the *group* graph cumulative coherence. For some optimised sampling distributions, we show that sampling $O(k\log(k))$ groups is always sufficient to stably embed the set of k-bandlimited signals but that this number can be smaller – down to $O(\log(k))$ – depending on the structure of the groups of nodes. Fast methods to approximate these sampling distributions are detailed. Second, we consider k-bandlimited signals that are nearly piecewise constant over pre-defined groups of nodes. We show that it is possible to speed up the reconstruction of such signals by reducing drastically the dimension of the vectors to reconstruct. When combined with the proposed structured sampling procedure, we prove that the method provides stable and accurate reconstruction of the original signal. Finally, we present numerical experiments that illustrate our theoretical results and, as an example, show how to combine these methods for interactive object segmentation in an image using superpixels.

1. Introduction

This work was initially inspired by studying developments in edit propagation for interactive image or video manipulations, where the goal is to propagate operations made by a user in some parts of the image to the entire image, *e.g.*, propagate foreground/background scribbles for object segmentation [1–5] or propagate manual color/tone modifications [3–7]. First, a graph $\mathcal{G}$ modelling the similarities between the pixels of the image is built. Second, the user-edits specify values at some nodes (pixels) of $\mathcal{G}$. Finally, the complete signal is estimated by assuming that it is smooth on $\mathcal{G}$. The quality of the propagation depends on the structure of $\mathcal{G}$ and on the location of annotated nodes. If a part of the image is weakly connected to the rest, the user-edits do not propagate well to this region unless this region is edited directly. Therefore, highlighting beforehand which regions or groups of nodes are important to edit to ensure a good propagation would be a useful feature to facilitate user interactions. Furthermore, designing a fast reconstruction/propagation method is also important so that the user can visualise immediately the effect of his inputs. To this end, "superpixels" – small groups of connected pixels where the image varies only little – are sometimes used among other things to speed up computations, *e.g.*, in [3]. We address these problems from a graph signal processing point-of-view [8]. More precisely, we view them as a sampling problem where we should select a *structured* set of nodes (regions to edit), "measure" the signal on this set (user-edits), and reconstruct the signal on the entire graph. We believe that the sampling strategy and the fast reconstruction method proposed here provide useful tools to optimize user's inputs and accelerate computations.

As mentionned above, the user-edits are propagated to the entire image by assuming that the global signal is smooth on $\mathcal{G}$. In the context of graph signal processing, smooth or k-bandlimited signal is a widely used and studied model. Several sampling methods have been designed to sample such signals. Pesenson introduced the notion of uniqueness set (of nodes) for k-bandlimited graph signals in [9, 10]. Two different k-bandlimited signals are also necessarily different when restricted to a uniqueness set. Therefore, one can sample all k-bandlimited signals on a uniqueness set. Then, Anis *et al.*, [11, 12] and Chen *et al.*, [13, 14] proved that one can always find a uniquess set of k nodes to sample all k-bandlimited signals. Finding this set is however computationally expensive. In [15], graph spectral proxies are used to find such a set more efficiently. Yet the combinatorial problem that needs to be solved to find such a set makes the method still difficult to use for large graphs. Other authors used the idea of random sampling to be able to handle large graphs [14, 16, 17]. Recently, Puy *et al.* [17] proved that there always exists a random sampling strategy for which sampling $O(k\log(k))$ nodes

G. Puy and P. Pérez are with Technicolor, 975 Avenue des Champs Blancs, 35576 Cesson-Sévigné, France..

is sufficient to stably embed the set of bandlimited signals. They also designed a fast and scalable algorithm to estimate the optimal sampling distribution.

In this paper, we study first a random sampling strategy for k-bandlimited signals where we sample few groups of nodes instead of sampling individual nodes. We introduce the concept of *local group graph coherence* that quantifies the importance of sampling each group. Second, in order to build a fast reconstruction technique for k-bandlimited signals, we use the intuition that a smooth graph signal is a signal that varies slowly from one node to its connected nodes. If we group few connected nodes together, we usually expect a bandlimited signal to be essentially constant on this set of nodes; as long as we do not group together too many weakly connected nodes. We propose here to use this property to accelerate the reconstruction of such k-bandlimited signals, *i.e.*, k-bandlimited signals nearly piecewise constant over (pre-defined) groups of nodes. When combined with the proposed sampling technique, we prove that this fast method provides stable and accurate reconstructions of the signals of interest. Finally, we illustrate how to use these results for interactive object segmentation in an image, with required node groups being superpixels.

1.1. Contributions

The random sampling strategy that we propose generalises the method proposed in [17]. We use here a structured sampling strategy. Let us already acknowledge here that such strategies are also studied in the field of compressed sensing [18–21] and that some of our solutions are directly inspired by these works.

First, in this structured sampling setting, we show that the number of groups to sample is directly linked to a quantity called the *group* graph cumulative coherence. This quantity generalises the concept of graph cumulative coherence introduced in [17] and characterises how much the energy of k-bandlimited can stay concentrated in each group of nodes.

Second, we can then choose to sample the groups non-adaptively or to optimise the sampling distribution to minimise the number of groups to sample. With this optimised sampling distribution, our result shows that, in the worst case, sampling $O(k\log(k))$ groups of nodes is sufficient to ensure the reconstruction of all k-bandlimited signals. As each group can contain many nodes, we might have to sample a large number of nodes. This is the potential price to pay when sampling the nodes by groups. Fortunately, a smaller number of groups – down to $O(\log(k))$ – might already be sufficient if the groups are well designed.

Third, we describe a method to estimate the optimal sampling distribution without the need of computing the graph Fourier matrix and which is thus able to handle large graphs.

Fourth, estimating the optimal sampling distribution may still be too slow when a large number of groups are involved. We thus also present a sufficient recovery condition that involves a relaxed version of group graph cumulative coherence. The sampling distribution that minimises this relaxed coherence is fast to estimate for large graphs and large number of groups. With this sampling strategy, we prove that sampling $O(k\log(k))$ groups is always sufficient to ensure the reconstruction of all k-bandlimited signals. This strategy is mainly interesting at small k.

Finally, we propose a fast reconstruction method for k-bandlimited signals that are also nearly piecewise constant over pre-defined groups of nodes. We show that we can reduce drastically the dimension of the reconstruction problem for such signals. When the above sampling procedure is used to sample them, we prove that the proposed method provides accurate and stable recovery of this type of signals.

1.2. Applications

The proposed sampling methods can have interest in several applications. For example, if one needs to develop sensors in a large scale network, it might be easier to deploy and install these sensors at nodes around few spatial locations instead of scattering them all over the network. Finding the best regions to monitor is thus important. In a social network, one might be interested in monitoring a signal defined over communities and thus should find which groups of users are the most important

to sample. In semi-supervised learning, it may be easier to label jointly some similar nodes than individual nodes. Let us now detail such an example for semi-supervised classification.

The task we consider is interactive object segmentation in an image where, similarly to what is done in [22] for instance, we build a graph $\mathcal{G}$ that models similarities between the pixels of the image, ask the user to label some regions depending on whether they are part of the object or not, diffuse the result on the complete image by supposing that indicator vectors of the object is smooth on $\mathcal{G}$. To propose the regions to label, we view this segmentation problem as a sampling problem of a smooth signal on $\mathcal{G}$ where the regions to label are chosen to ensure the recovery of a k-bandlimited signal. To choose the regions to label, we start by partitioning the image into superpixels, *e.g.*, with SLIC technique [23]. We see in Fig. 1 that the superpixels divide the image into homogeneous regions. The superpixels follow the edges and most of them thus belong to either the tiger or the background but rarely both. One interest of dividing the image into superpixels is that it facilitates user interactions. It is easier to determine if a *superpixel* belongs to the object of interest than if a *pixel* belongs to the tiger, especially at the boundaries, as recently exploited for segmentation on touch-screens [24]. Using the sampling method that we developed we can propose to the user a small number of superpixels to label. Furthermore, this proposition is adapted to the structure of the graph. Another advantage of using superpixels is that the indicator function of the object, beyond being smooth on $\mathcal{G}$, is also approximately piecewise constant on the superpixels. We can thus use our reconstruction method to estimate rapidly the segmentation result from the labelled superpixels.

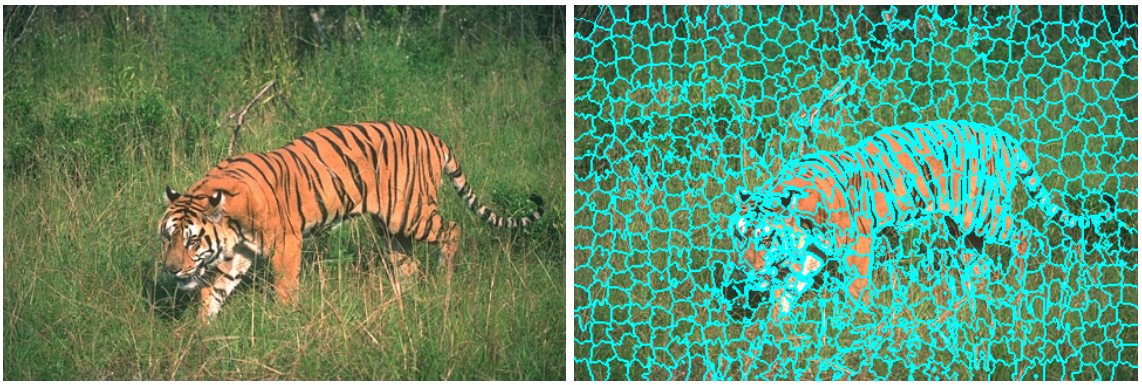


Figure 1. *One image (left) and its partition into superpixels with SLIC technique [23] (right).*

1.3. Notations and definitions

We consider that $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathsf{W}\}$ is an undirected weighted graph, where $\mathcal{V}$ is the set of n nodes, $\mathcal{E}$ is the set of edges, and $\mathsf{W} \in \mathbb{R}^{n \times n}$ is the weighted adjacency matrix with nonnegative entries. We denote the graph Laplacian by $\mathsf{L} \in \mathbb{R}^{n \times n}$. We assume that L is real, symmetric, and positive semi-definite, *e.g.*, the combinatorial graph Laplacian $\mathsf{L} := \mathsf{D} - \mathsf{W}$, or the normalised one $\mathsf{L} := \mathsf{I} - \mathsf{D}^{-1/2}\mathsf{W}\mathsf{D}^{-1/2}$. The matrix $\mathsf{D} \in \mathbb{R}^{n \times n}$ is the diagonal degree matrix and $\mathsf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix [25]. The diagonal degree matrix D has entries $D_{ii} := \sum_{i \neq j} \mathsf{W}_{ij}$.

We denote by $\mathsf{U} \in \mathbb{R}^{n \times n}$ the orthonormal eigenvectors of L and by $0 = \lambda_1 \leqslant \ldots \leqslant \lambda_n$ the ordered real eigenvalues of L. We have $\mathsf{L} = \mathsf{U}\mathsf{\Lambda}\mathsf{U}^\intercal$, where $\mathsf{\Lambda} := \operatorname{diag}(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n \times n}$. The matrix U is the graph Fourier basis [8]. For any signal $\boldsymbol{x} \in \mathbb{R}^n$ defined on the nodes of the graph $\mathcal{G}$, $\hat{\boldsymbol{x}} = \mathsf{U}^\intercal \boldsymbol{x}$ contains the Fourier coefficients of $\boldsymbol{x}$ ordered in increasing frequencies. This work deals with k-bandlimited signals $\boldsymbol{x} \in \mathbb{R}^n$ on $\mathcal{G}$, *i.e.*, signals whose Fourier coefficients $\hat{\boldsymbol{x}}_{k+1}, \ldots, \hat{\boldsymbol{x}}_n$ are null. Let U_k be the restriction of U to its first k vectors:

$$\mathsf{U}_k := (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k) \in \mathbb{R}^{n \times k}. \tag{1}$$

Definition 1.1 (k-bandlimited signal on $\mathcal{G}$)**.** *A signal $\boldsymbol{x} \in \mathbb{R}^n$ defined on the nodes of the graph $\mathcal{G}$ is k-bandlimited with $k \in \mathbb{N} \setminus \{0\}$ if $\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k)$, i.e., there exists $\boldsymbol{\eta} \in \mathbb{R}^k$ such that*

$$\boldsymbol{x} = \mathsf{U}_k \boldsymbol{\eta}. \tag{2}$$

This definition was also used in [11, 14, 17]. We assume that $\lambda_k \neq \lambda_{k+1}$ to avoid any ambiguity in the definition of k-bandlimited signals.

Finally, for any matrix $\mathsf{X} \in \mathbb{R}^{n_1 \times n_2}$, $\|\mathsf{X}\|_2$ denotes its spectral norm and $\|\mathsf{X}\|_F$ its Frobenius norm; when $n_1 = n_2$, $\lambda_{\max}(\mathsf{X})$ denotes its largest eigenvalue and $\lambda_{\min}(\mathsf{X})$ its smallest eigenvalue. For any vector $\boldsymbol{x} \in \mathbb{R}^{n_1}$, $\|\boldsymbol{x}\|_2$ denotes the Euclidean norm of $\boldsymbol{x}$. Depending on the context, $\boldsymbol{x}_j$ may represent the j^{th} entry of the vector $\boldsymbol{x}$ or the j^{th} column-vector of the matrix X. The entry on the i^{th} row and j^{th} column of X is denoted by X_{ij}. The identity matrix is denoted by I (its dimensions are determined by the context).

We present in Fig. 2 a representation of the important variables and processes involved in this paper in order to facilitate the understanding of the different results.

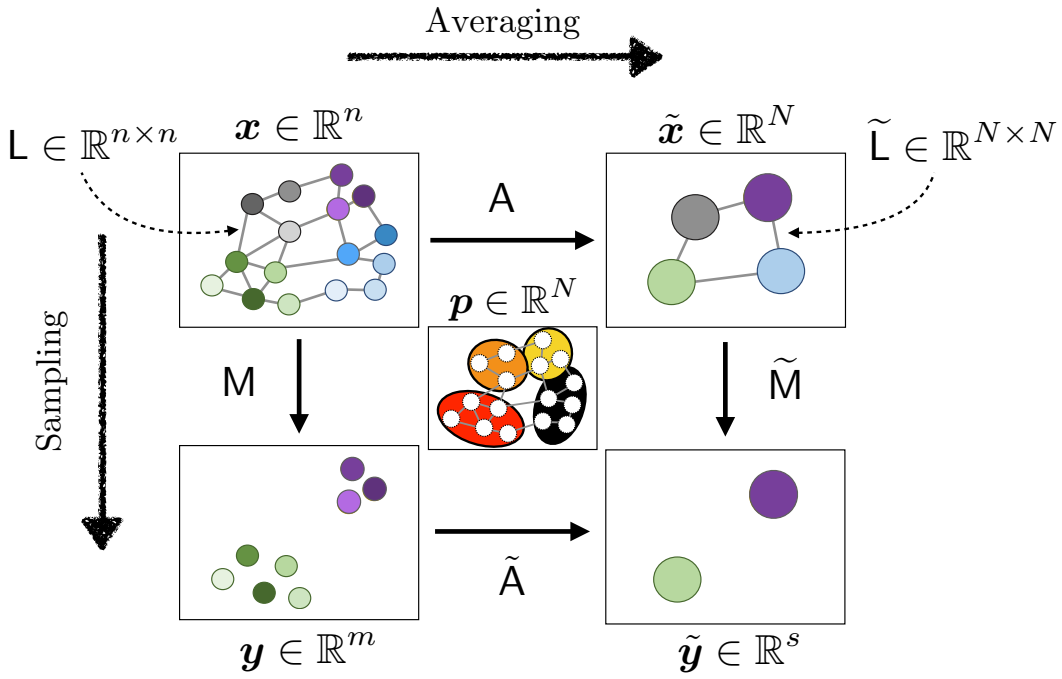


Figure 2. *Summary of the main variables and processes involved in the presented sampling strategies. All variables are defined when needed hereafter.*

2. Sampling using groups of nodes

In this section, we explain our sampling strategies, starting with the definition of the groups of nodes.

2.1. Grouping the nodes

We consider that the n nodes of $\mathcal{G}$ are divided into N different groups $\mathcal{N}_1, \ldots, \mathcal{N}_N \subseteq \{1, \ldots, n\}$. The size of the ℓ^{th} group is denoted $|\mathcal{N}_\ell|$. We suppose that these groups form a partition of $\{1, \ldots, n\}$, so that each node belongs to exactly one group. We have

$$\cup_{\ell=1}^{N} \mathcal{N}_\ell \;=\; \{1, \ldots, n\}, \text{ and } \mathcal{N}_\ell \cap \mathcal{N}_{\ell'} = \emptyset. \tag{3}$$

For the object segmentation application discussed in the introduction, these groups represent the superpixels. However, we do not impose the groups to be made of neighbouring nodes in the graph; they can be made of nodes "far" from each other.

For each group $\mathcal{N}_\ell = \{n_1^{(\ell)}, \dots, n_{|\mathcal{N}_\ell|}^{(\ell)}\}$, we associate a matrix $\mathsf{N}^{(\ell)} \in \mathbb{R}^{|\mathcal{N}_\ell| \times n}$ that restricts a graph signal to the nodes appearing in $\mathcal{N}_\ell$, *i.e.*,

$$\mathsf{N}_{ij}^{(\ell)} := \begin{cases} 1 & \text{for } j = n_i^{(\ell)}, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Note that

$$\sum_{\ell=1}^{N} \mathsf{N}^{(\ell)\mathsf{T}} \mathsf{N}^{(\ell)} = \mathsf{I}. \tag{5}$$

The case of overlapping groups can be handled by changing the definition of $\mathsf{N}^{(\ell)}$ to

$$\mathsf{N}_{ij}^{(\ell)} := \begin{cases} \beta_{n_i^{(\ell)}}^{-1/2} & \text{for } j = n_i^{(\ell)}, \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $1 \leqslant \beta_i \leqslant N$, $i = 1, \dots, n$, is the number of times node i appears in the different groups $\mathcal{N}_1, \dots, \mathcal{N}_N$. Equation (5) also holds in this case. All results presented in Section 2 are valid for overlapping groups with this definition of $\mathsf{N}^{(\ell)}$.

2.2. Sampling the groups

The sampling procedure consists in selecting s groups out of the N available ones. In the application of Section 1.2, it corresponds to the selection of the superpixels to label. We select these groups at random using a sampling distribution on $\{1, \dots, N\}$ represented by a vector $\boldsymbol{p} \in \mathbb{R}^N$. The probability of selecting the ℓ^{th} group is $\boldsymbol{p}_\ell$. We assume that $\boldsymbol{p}_\ell > 0$ for all $\ell = 1, \dots, N$, so that all groups may be selected with a non-zero probability. We obviously have $\sum_{\ell=1}^{N} \boldsymbol{p}_\ell = 1$.

The indices $\Omega := \{\omega_1, \dots, \omega_s\}$ of the selected groups are obtained by drawing independently – thus with replacements – s indices from the set $\{1, \dots, N\}$ according to $\boldsymbol{p}$, *i.e.*,

$$\mathbb{P}(\omega_j = \ell) = \boldsymbol{p}_\ell, \quad \forall j \in \{1, \dots, s\} \text{ and } \forall \ell \in \{1, \dots, N\}. \tag{7}$$

The selected groups are $\mathcal{N}_{\omega_1}, \dots, \mathcal{N}_{\omega_s}$ and the total number of selected nodes is

$$m := \sum_{j=1}^{s} \left| \mathcal{N}_{\omega_j} \right|. \tag{8}$$

Once the groups are selected, we build the sampling matrix $\mathsf{M} \in \mathbb{R}^{m \times n}$ that satisfies

$$\mathsf{M} := \begin{pmatrix} \mathsf{N}^{(\omega_1)} \\ \vdots \\ \mathsf{N}^{(\omega_s)} \end{pmatrix}, \tag{9}$$

and which restricts any signal to the nodes belonging to the selected groups. For a signal $\boldsymbol{x} \in \mathbb{R}^n$ defined on the nodes of $\mathcal{G}$, its sampled version $\boldsymbol{y} \in \mathbb{R}^m$ satisfies

$$\boldsymbol{y} := \mathsf{M}\boldsymbol{x}. \tag{10}$$

Our goal now is to determine what number s is enough to ensure that all k-bandlimited signals can be reconstructed from its sampled version obtained with M. To conduct this study, we need to define few more matrices. First, we associate the matrix

$$\mathsf{P}^{(\ell)} := \boldsymbol{p}_\ell^{-1/2}\, \mathsf{I} \in \mathbb{R}^{|\mathcal{N}_\ell| \times |\mathcal{N}_\ell|}, \tag{11}$$

to each group $\mathcal{N}_\ell$. Then, once the groups are drawn, we construct the block diagonal matrix $\mathsf{P} \in \mathbb{R}^{m \times m}$

$$\mathsf{P} := \operatorname{diag}\left(\mathsf{P}^{(\omega_1)}, \dots, \mathsf{P}^{(\omega_s)}\right). \tag{12}$$

This matrix takes into account the probability of sampling each group and will be used to rescale $\boldsymbol{y}$ for norm preservation. This matrix ensures that $s^{-1}\,\mathbb{E}_\Omega\,\|\mathsf{P}\boldsymbol{y}\|_2^2 = s^{-1}\,\mathbb{E}_\Omega\,\|\mathsf{PM}\boldsymbol{x}\|_2^2 = \|\boldsymbol{x}\|_2^2$.[1] Both matrices P and M depend on Ω and are random.

2.3. Group graph coherence

The sampling procedure in [17] is similar to the ones proposed here at the difference that the nodes are sampled individually and not by groups. It was proved there that the number of nodes to sample is driven by a quantity called the graph coherence. This quantity measures how the energy of k-bandlimited signals spreads over the *nodes*. Similarly, we prove here that the number of groups to sample is driven by a quantity that measures the energy of k-bandlimited signals spreads over the *groups*. We now introduce this quantity.

The matrix $\mathsf{N}^{(\ell)}\mathsf{U}_k$ is the matrix that restricts a k-bandlimited signal to the nodes belonging to $\mathcal{N}_\ell$. Therefore,

$$\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2 = \sup_{\boldsymbol{\eta}\in\mathbb{R}^k:\|\boldsymbol{\eta}\|_2=1} \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\boldsymbol{\eta}\right\|_2 \tag{13}$$

measures the energy on the nodes $\mathcal{N}_\ell$ of the normalised k-bandlimited signal that is most concentrated on $\mathcal{N}_\ell$. This energy varies between 0 and 1. When this energy is close to 1, there exists a k-bandlimited signal whose energy is essential concentrated on $\mathcal{N}_\ell$. This signal lives only on the nodes in $\mathcal{N}_\ell$ and does not spread elsewhere. On the contrary, when this energy is close to 0, there is no k-bandlimited signal living only on $\mathcal{N}_\ell$.

The sampling distribution $\boldsymbol{p}$ is adapted to the graph and the structure of the groups if: whenever $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ is high, $\boldsymbol{p}_\ell$ is high; whenever $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ is small, $\boldsymbol{p}_\ell$ is small. In other words, the ratio between $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ and $\boldsymbol{p}_\ell$ should be as constant as possible. This ensures that the groups where some k-bandlimited signals are concentrated are sampled with higher probability. Similarly to what was done in [17] with individual nodes, we define the group graph weighted coherence as the largest ratio between $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ and $\boldsymbol{p}_\ell^{-1/2}$.

Definition 2.1 (Group graph cumulative coherence)**.** *The group graph cumulative coherence of order k is*

$$\nu_{\boldsymbol{p}} := \max_{1\leqslant\ell\leqslant N} \left\{\boldsymbol{p}_\ell^{-1/2}\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2\right\}. \tag{14}$$

The quantity $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ is called the local group graph coherence.

In the extreme case where the groups $\mathcal{N}_1,\dots,\mathcal{N}_N$ reduce all to singletons, we recover the definition of the graph weighted coherence introduced in [17].

It is easy to prove that $\nu_{\boldsymbol{p}}$ is lower bounded by 1. Indeed, for any $\boldsymbol{\eta}\in\mathbb{R}^k$ with $\|\boldsymbol{\eta}\|_2 = 1$, we have

$$\begin{aligned} 1 = \|\mathsf{U}_k\boldsymbol{\eta}\|_2^2 &= \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\boldsymbol{\eta}\right\|_2^2 = \sum_{\ell=1}^N \boldsymbol{p}_\ell\cdot\frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\boldsymbol{\eta}\right\|_2^2}{\boldsymbol{p}_\ell} \leqslant \|\boldsymbol{p}\|_1\cdot\max_{1\leqslant\ell\leqslant N}\left\{\frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\boldsymbol{\eta}\right\|_2^2}{\boldsymbol{p}_\ell}\right\} \\ &= \max_{1\leqslant\ell\leqslant N}\left\{\frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\boldsymbol{\eta}\right\|_2^2}{\boldsymbol{p}_\ell}\right\} \leqslant \max_{1\leqslant\ell\leqslant N}\left\{\frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2}{\boldsymbol{p}_\ell}\right\} = \nu_{\boldsymbol{p}}^2. \end{aligned} \tag{15}$$

We have $\nu_{\boldsymbol{p}} = 1$ in, for example, the degenerated case where $\mathcal{N}_1 = \{1,\dots,n\}$.

2.4. Stable embedding

We now have all the tools to present our main theorem that shows that sampling $O(\nu_{\boldsymbol{p}}^2\log(k))$ groups is sufficient to stably embed the whole set of k-bandlimited signals. Hence, it is possible to reconstruct any $\boldsymbol{x}\in\operatorname{span}(\mathsf{U}_k)$ from its measurements $\boldsymbol{y} = \mathsf{M}\boldsymbol{x}$.

[1] This property is a consequence of (65), proved in Appendix A.

Theorem 2.2 (Restricted isometry property - RIP)**.** *Let* M *be a random subsampling matrix constructed as in* (9) *using the groups* $\mathcal{N}_1, \dots, \mathcal{N}_N$ *and the sampling distribution* $\boldsymbol{p}$*. For any* $\delta, \xi \in (0,1)$*, with probability at least* $1-\xi$*,*

$$(1-\delta)\,\|\boldsymbol{x}\|_2^2 \leqslant \frac{1}{s}\,\|\mathsf{PM}\,\boldsymbol{x}\|_2^2 \leqslant (1+\delta)\,\|\boldsymbol{x}\|_2^2 \tag{16}$$

for all $\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k)$ *provided that*

$$s \geqslant \frac{3}{\delta^2}\,\nu_{\boldsymbol{p}}^2\,\log\left(\frac{2k}{\xi}\right). \tag{17}$$

Proof. See Appendix A. □

In the above theorem, we recall that s is the number of selected groups, each of them containing several nodes. We thus control the number of groups to sample and not directly the number of nodes. As the lower bound on $\nu_{\boldsymbol{p}}$ is 1, sampling $O(\log(k))$ groups might already be sufficient if the groups and the sampling distribution are well-designed.

The number of groups to sample is driven by $\nu_{\boldsymbol{p}}$, which itself depends on the structure of the groups $\mathcal{N}_1, \dots, \mathcal{N}_N$ and on the sampling distribution $\boldsymbol{p}$. To reduce the number of samples to measure, we might optimise the structure of the groups and the sampling distribution. For example, if we were able to construct n/L groups ($L \geqslant 1$) such that $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2 \approx (L/n)^{1/2}$ – *i.e.*, no k-bandlimited signals have more than $100 \cdot (L/n)^{1/2}$ percent of its energy concentrated in each group – then setting $\boldsymbol{p}_\ell = L/n$, $\ell = 1, \dots, n/L$, would yield $\nu \approx 1$. In this case, sampling one group is enough to embed the set of k-bandlimited signals. However, it is not obvious how we can construct such groups in practice and we might not even have the flexibility to modify the structure of the groups. In such a case, the only possibility to reduce the number of measurements is to optimise the sampling distribution $\boldsymbol{p}$ to minimise $\nu_{\boldsymbol{p}}$.

The sampling distribution minimizing the coherence $\nu_{\boldsymbol{p}}$ is the distribution $\boldsymbol{p}^* \in \mathbb{R}^N$ that satisfies

$$\boldsymbol{p}_\ell^* := \frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2}{\sum_{\ell'=1}^N \left\|\mathsf{N}^{(\ell')}\mathsf{U}_k\right\|_2^2}, \text{ for all } \ell \in \{1, \dots, N\}, \tag{18}$$

and for which

$$\nu_{\boldsymbol{p}^*}^2 = \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2. \tag{19}$$

Indeed, let $\boldsymbol{p}' \neq \boldsymbol{p}^*$ be another sampling distribution. As the entries of $\boldsymbol{p}'$ and $\boldsymbol{p}^*$ are nonnegative and sum to 1, we necessarily have $\boldsymbol{p}'_{\ell'} < \boldsymbol{p}^*_{\ell'}$ for some $\ell' > 0$. Then,

$$\nu_{\boldsymbol{p}'}^2 \geqslant {\boldsymbol{p}'_{\ell'}}^{-1} \left\|\mathsf{N}^{(\ell')}\mathsf{U}_k\right\|_2^2 > {\boldsymbol{p}^*_{\ell'}}^{-1} \left\|\mathsf{N}^{(\ell')}\mathsf{U}_k\right\|_2^2 = \nu_{\boldsymbol{p}^*}^2, \tag{20}$$

where the last equality is obtained by replacing $\boldsymbol{p}^*_{\ell'}$ with its value. Therefore, $\nu_{\boldsymbol{p}'} > \nu_{\boldsymbol{p}^*}$ for any $\boldsymbol{p}' \neq \boldsymbol{p}^*$. As similar proof can be found in, *e.g.*, [18] where the authors derive the optimal sampling distribution for a compressive system.

We notice that

$$\nu_{\boldsymbol{p}^*}^2 = \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2 \leqslant \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2 = k. \tag{21}$$

Hence, by using this distribution, (17) shows that sampling $O(k\log(k))$ groups is always sufficient to sample all k-bandlimited signals. The exact number is proportional to $\nu_{\boldsymbol{p}^*}^2 \log(k)$. This is not in contradiction with the fact that at least k measurements are required in total as one group contains at least one node. We also have

$$\nu_{\boldsymbol{p}^*}^2 = \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2 \leqslant N, \tag{22}$$

as $\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2 \leqslant 1$. Therefore, in any case, the bound never suggests to sample much more than N groups, as one would expect it.

We recall that the results in [17] proves that it is always sufficient to sample $O(k\log(k))$ nodes to embed the set of k-bandlimited signals. When sampling the nodes by groups, it is the number of groups to sample that should be $O(k\log(k))$. This can be a large number of individual nodes but this is the potential price to pay when sampling the nodes by groups.

Variable density sampling [19, 26, 27] and structured sampling [18, 20, 21] are also important topics in compressed sensing. The method proposed here is closely inspired by these studies, especially by [20, 21]. Our results thus share several similarities with these works. We however benefit from a simpler signal model and take advantage of the graph structure to refine the results, propose simpler decoders to reconstruct the signal, and design efficient algorithms to estimate $\boldsymbol{p}^*$.

2.5. A more practical result at small k's

Optimising the sampling distribution reduces to estimating the spectral norm of the matrices $\mathsf{N}^{(\ell)}\mathsf{U}_k$. We will present in Section 4 a method avoiding the computation of U_k. However, the method might still be too slow when a large number of groups is involved as it requires an estimation of a spectral norm for each group separately. It is thus interesting to characterise the performance of the proposed method using other quantities easier to compute.

In this section, we present results involving the following quantity

$$\bar{\nu}_{\boldsymbol{p}} := \max_{1\leqslant\ell\leqslant N} \left\{ \boldsymbol{p}_\ell^{-1/2} \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F \right\}. \tag{23}$$

The only difference between $\nu_{\boldsymbol{p}}$ and $\bar{\nu}_{\boldsymbol{p}}$ is that we substituted the Frobenius norm for the spectral norm. As $\|\mathsf{X}\|_2 \leqslant \|\mathsf{X}\|_F$ for any matrix X, we have $\nu_{\boldsymbol{p}} \leqslant \bar{\nu}_{\boldsymbol{p}}$; hence the results involving $\bar{\nu}_{\boldsymbol{p}}$ will be more pessimistic than those involving $\nu_{\boldsymbol{p}}$. We have $\bar{\nu}_{\boldsymbol{p}} \geqslant k$. Indeed,

$$k = \|\mathsf{U}_k\|_F^2 = \sum_{\ell=1}^N \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2 = \sum_{\ell=1}^N \boldsymbol{p}_\ell \cdot \frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2}{\boldsymbol{p}_\ell} \leqslant \|\boldsymbol{p}\|_1 \cdot \max_{1\leqslant\ell\leqslant s} \left\{ \frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2}{\boldsymbol{p}_\ell} \right\} = \bar{\nu}_{\boldsymbol{p}}^2. \tag{24}$$

As with $\nu_{\boldsymbol{p}}$, the lower bound is also attained in, for example, the degenerated case where $\mathcal{N}_1 = \{1,\dots,n\}$. As $\nu_{\boldsymbol{p}} \leqslant \bar{\nu}_{\boldsymbol{p}}$, we have the following corollary to Theorem 2.2.

Corollary 2.3. *Let M be a random subsampling matrix constructed as in (9) using the groups $\mathcal{N}_1,\dots,\mathcal{N}_N$ and the sampling distribution $\boldsymbol{p}$. For any $\delta,\xi \in (0,1)$, with probability at least $1-\xi$, (16) holds for all $\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k)$ provided that*

$$s \geqslant \frac{3}{\delta^2}\, \bar{\nu}_{\boldsymbol{p}}^2\, \log\left(\frac{2k}{\xi}\right). \tag{25}$$

Proof. As $\bar{\nu}_{\boldsymbol{p}} \geqslant \nu_{\boldsymbol{p}}$, (25) implies (17). Theorem 2.2 then proves that (16) holds with probability at least $1-\xi$ for all $\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k)$. □

The sufficient condition (25) can be much more pessimistic than (17). Indeed, as we have $\bar{\nu}_{\boldsymbol{p}}^2 \geqslant k$, Condition (25) suggests to always sample more than $O(k\log(k))$ groups, while we know that sampling $O(\log(k))$ can be enough. The interest of this result is thus in the regime where k is small.

As we have done it with $\nu_{\boldsymbol{p}}$, we can also optimise $\boldsymbol{p}$ to minimise $\bar{\nu}_{\boldsymbol{p}}$. The sampling distribution that minimises $\bar{\nu}_{\boldsymbol{p}}$ is the distribution $\boldsymbol{q}^*$ satisfying

$$\boldsymbol{q}_\ell^* := \frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2}{k}, \text{ for all } l \in \{1,\dots,N\}, \tag{26}$$

and for which

$$\bar{\nu}_{\boldsymbol{q}^*}^2 = k. \tag{27}$$

With this distribution, (25) proves that sampling $O(k\log(k))$ groups is always enough to sample all k-bandlimited signals. This result is particularly interesting at small k's because estimating $\boldsymbol{q}^*$ is much easier and faster than estimating $\boldsymbol{p}^*$ (see Section 4).

In some particular cases, this result can be interesting also at large k's. Indeed, (25) is a pessimistic bound. In reality, we may have $\nu^2_{\boldsymbol{q}^*} \leqslant k$ so that, according to (17), fewer samples than $O(k\log(k))$ are actually sufficient when using $\boldsymbol{q}^*$. Furthermore, $\boldsymbol{q}^*$ might actually be close to $\boldsymbol{p}^*$, in which case we would reach almost optimal results with $\boldsymbol{q}^*$. The occurrence of these events is however quite difficult to predict and depends on the structure of the groups and of the graph.

Note that we need the knowledge of the truncated Fourier matrix U_k to compute the distributions $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$. Computing U_k can be intractable for large graphs. We will present a method to overcome this issue in Section 4. We continue first by explaining how to estimate $\boldsymbol{x}$ from its measurements.

3. Fast reconstruction

Once the signal has been sampled, we shall also be able to reconstruct it. In [17], the authors propose to estimate the original signal by solving

$$\min_{\boldsymbol{z}\in\mathbb{R}^n} \|\mathsf{P}(\mathsf{M}\boldsymbol{z}-\boldsymbol{y})\|_2^2 + \gamma\, \boldsymbol{z}^\intercal g(\mathsf{L})\boldsymbol{z}, \tag{28}$$

where $\gamma > 0$ and g is a nonnegative nondecreasing polynomial. We have

$$g(\mathsf{L}) := \sum_{i=0}^{d} \alpha_i\, \mathsf{L}^i = \sum_{i=0}^{d} \alpha_i\, \mathsf{U}\mathsf{\Lambda}^i\mathsf{U}^\intercal, \tag{29}$$

where $\alpha_0,\dots,\alpha_d \in \mathbb{R}$ are the coefficients of the polynomial and $d \in \mathbb{N}$ its degree. These polynomial's parameters can be tuned to improve the quality of the reconstruction. The role of g can be viewed as a filter on $\mathcal{G}$ and should ideally be a high-pass filter. The matrix P is introduced to account for the RIP. The advantage of this method is that it can be solved efficiently even for large graph for example by conjugate gradient. Indeed, each step of this algorithm can be implemented by using only matrix-vector multiplications with PM and L, which are both sparse matrices. The matrix $g(\mathsf{L})$ does not have to be computed explicitly.

For brevity, we do not recall the theorem proving that (28) provides accurate and stable recovery of all k-bandlimited signals. However, this theorem applies as soon as the restricted isometry property (16) holds, and thus applies here when (17) holds. The reconstruction quality is controlled by $g(\lambda_k)$ and the ratio $g(\lambda_k)/g(\lambda_{k+1})$. One should seek to design a filter g such that these quantities are as close as possible to zero to improve the reconstruction quality.

We propose now a method to obtain a faster estimation of the original signal when it is nearly piecewise constant.

3.1. Piecewise constant graph signals

Before continuing, we want to stress that we consider non-overlapping groups $\mathcal{N}_1,\dots,\mathcal{N}_N$ in this rest of Section 3.

If a graph signal is nearly piecewise constant over the groups $\mathcal{N}_1,\dots,\mathcal{N}_N$ then reconstructing the mean values of this signal for each group is enough to obtain a good approximation of the original signal. Instead of estimating n unknowns, we reduce the estimation to N unknowns. When $N \ll n$, this is a large reduction of dimension yielding a significant speed up and memory reduction.

The fact that a signal $\boldsymbol{x}$ is piecewise constant over the groups $\mathcal{N}_1,\dots,\mathcal{N}_N$ is characterized as follows. We construct the averaging row-vectors $\boldsymbol{a}^{(\ell)} \in \mathbb{R}^{1\times n}$ that satisfy

$$\boldsymbol{a}^{(\ell)} := \frac{\mathbf{1}^\intercal \mathsf{N}^{(\ell)}}{|\mathcal{N}_\ell|^{1/2}}, \tag{30}$$

and the matrix

$$\mathsf{A} := \begin{pmatrix} \boldsymbol{a}^{(1)} \\ \vdots \\ \boldsymbol{a}^{(N)} \end{pmatrix} \in \mathbb{R}^{N\times n}. \tag{31}$$

As the groups do not overlap, we have

$$\mathsf{A}\mathsf{A}^\intercal = \mathsf{I}, \tag{32}$$

hence $\|\mathsf{A}\|_2 = 1$. Applying A to $\boldsymbol{x}$ provides N values, each one of them corresponding to the sum of the values of $\boldsymbol{x}$ within the group $\mathcal{N}_\ell$, scaled by $|\mathcal{N}_\ell|^{-1/2}$. Then, applying $\mathsf{A}^\intercal$ to $\mathsf{A}\boldsymbol{x}$ gives an approximation of $\boldsymbol{x}$ where the values in the vector $\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}$ are constant within each group; this is a piecewise constant vector over the groups. The value of $\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}$ appearing within the group $\mathcal{N}_\ell$ is exactly the average of $\boldsymbol{x}$ within $\mathcal{N}_\ell$. Saying that $\boldsymbol{x}$ is nearly constant within each group corresponds to assume that

$$\|\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x} - \boldsymbol{x}\|_2 \leqslant \epsilon \|\boldsymbol{x}\|_2 , \tag{33}$$

where $\epsilon \geqslant 0$ is a small value. The signal model of interest in this section is thus

$$\mathcal{A}_\epsilon := \{\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k) \;\mid\; \|(\mathsf{A}^\intercal\mathsf{A} - \mathsf{I})\,\boldsymbol{x}\|_2 \leqslant \epsilon \|\boldsymbol{x}\|_2\} . \tag{34}$$

3.2. Reducing the dimension

To build a fast algorithm exploiting the above property, we use a reconstruction method similar to (28) but involving vectors of smaller dimension. We define the averaged vector $\tilde{\boldsymbol{x}} := \mathsf{A}\boldsymbol{x} \in \mathbb{R}^N$ of dimension N. As $\boldsymbol{x} \in \mathcal{A}_\epsilon$, estimating $\tilde{\boldsymbol{x}}$ is enough to get a good approximation of $\boldsymbol{x}$ – we just need to multiply it with $\mathsf{A}^\intercal$. Furthermore, as $\boldsymbol{x}$ is nearly piecewise constant over the groups $\mathcal{N}_1, \dots, \mathcal{N}_N$, by construction of the matrix M, the measurement vector $\boldsymbol{y} = \mathsf{M}\boldsymbol{x}$ is also almost piecewise constant over the sampled groups $\mathcal{N}_{\omega_1}, \dots, \mathcal{N}_{\omega_s}$. We thus average $\boldsymbol{y}$ over these groups by multiplying it with the matrix $\widetilde{\mathsf{A}} \in \mathbb{R}^{s\times m}$ that satisfies

$$\widetilde{\mathsf{A}}_{ji} := \begin{cases} |\mathcal{N}_{\omega_j}|^{-1/2} & \text{for } \sum_{j'=1}^{j-1} \left|\mathcal{N}_{\omega_{j'}}\right| \leqslant\ i\ \leqslant \sum_{j'=1}^{\ell} \left|\mathcal{N}_{\omega_{j'}}\right|, \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

We obtain

$$\tilde{\boldsymbol{y}} := \widetilde{\mathsf{A}}\boldsymbol{y} = \widetilde{\mathsf{A}}\mathsf{M}\boldsymbol{x} \in \mathbb{R}^s. \tag{36}$$

We now have to link $\tilde{\boldsymbol{y}}$ to $\tilde{\boldsymbol{x}}$. We create the matrix $\widetilde{\mathsf{M}} \in \mathbb{R}^{s\times N}$ that restricts the N mean value of $\tilde{\boldsymbol{x}}$ to the s mean value of the selected groups, *i.e.*,

$$\widetilde{\mathsf{M}}_{ji} := \begin{cases} 1 & \text{if } i = \omega_j, \\ 0 & \text{otherwise.} \end{cases} \tag{37}$$

We have therefore

$$\tilde{\boldsymbol{y}} = \widetilde{\mathsf{M}}\,\tilde{\boldsymbol{x}}. \tag{38}$$

The goal is now to estimate $\tilde{\boldsymbol{x}}$ from $\tilde{\boldsymbol{y}}$. To ensure that the reconstruction method is stable to measurement noise, we do not consider the perfect scenario above but instead the scenario where

$$\tilde{\boldsymbol{y}} = \widetilde{\mathsf{M}}\tilde{\boldsymbol{x}} + \tilde{\boldsymbol{n}}, \tag{39}$$

and $\tilde{\boldsymbol{n}} \in \mathbb{R}^s$ models noise. We now need a regularisation term to estimate $\tilde{\boldsymbol{x}}$. We obtain this term by reducing the dimension of the regularisation involving the Laplacian L in (28). We compute

$$\widetilde{\mathsf{L}} := \mathsf{A}\, g(\mathsf{L})\, \mathsf{A}^\intercal = (\mathsf{A}\mathsf{U})\, g(\Lambda)\, (\mathsf{A}\mathsf{U})^\intercal \in \mathbb{R}^{N\times N}. \tag{40}$$

Note that $\widetilde{\mathsf{L}}$ is a symmetric positive definite matrix. Like $g(\mathsf{L})$, it can thus be used as a regularisation. We thus propose to estimate $\tilde{\boldsymbol{x}}$ by solving

$$\min_{\tilde{\boldsymbol{z}}\in\mathbb{R}^N} \left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}} - \tilde{\boldsymbol{y}})\right\|_2^2 + \gamma\, \tilde{\boldsymbol{z}}^\intercal\, \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{z}}, \tag{41}$$

where $\gamma > 0$ and $\widetilde{\mathsf{P}} \in \mathbb{R}^{s\times s}$ is the diagonal matrix with entries satisfying

$$\widetilde{\mathsf{P}}_{jj} := \boldsymbol{p}_{\omega_j}^{-1/2}. \tag{42}$$

Let $\tilde{\boldsymbol{x}}^* \in \mathbb{R}^N$ be a solution of (41). We finally obtain an estimation of $\boldsymbol{x}$ by computing $\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^*$.

In the particular case where $g(\cdot)$ is the identity, one can notice that

$$\widetilde{\mathsf{L}}_{\ell\ell'} = \sum_{(i,j)\in\mathcal{N}_\ell\times\mathcal{N}_{\ell'}} \frac{\mathsf{L}_{ij}}{|\mathcal{N}_\ell|^{1/2}\,|\mathcal{N}_{\ell'}|^{1/2}} \tag{43}$$

is non-zero only if there is at least one edge in $\mathcal{E}$ joining the groups $\mathcal{N}_\ell$ and $\mathcal{N}_{\ell'}$. The Laplacian $\widetilde{\mathsf{L}}_{\ell\ell'}$ preserves the connections present in the original graph represented by L.

The dimension of the unknown vector in (41) is N, which can be much smaller than n. This leads to a large gain in memory and computation time when either $\widetilde{\mathsf{L}}$ or matrix-vector multiplications with $\widetilde{\mathsf{L}}$ can be computed rapidly. If g has a small degree, then $\widetilde{\mathsf{L}}$ can be computed explicitly in a short amount of time. In such a case, we can solve (41) faster than (28) as it involves matrices of (much) smaller dimensions. In general, it is however not always straightforward to find an a efficient implementation for matrix-vector multiplications with $\widetilde{\mathsf{L}}$ without temporary going back to the signal domain of dimension n, *i.e.*, multiplying the vector $\tilde{\boldsymbol{z}}$ with $\mathsf{A}^\intercal$, filtering the high dimensional signal $\mathsf{A}^\intercal\tilde{\boldsymbol{z}}$, and downsampling the result. Even though solving (41) might still be faster than solving (28) in this situation, we loose part of the efficiency by working temporarily in the high dimensional domain. We thus have less flexibility in the choice of g with this reconstruction technique.

Let us also mention that (41) can be used to initialise the algorithm used to solve (28) with a good approximate solution. As in multigrid approaches to solve linear systems of equations, see, *e.g.*, [28, 29].

The following theorem bounds the error between the signal recovered by (41) and the original vector.

Theorem 3.1. *Let $\Omega = \{\omega_1, \dots, \omega_s\}$ be a set of s indices selected independently from $\{1, \dots, N\}$ using a sampling distribution $\boldsymbol{p} \in \mathbb{R}^N$, $\mathsf{M}, \mathsf{P}, \widetilde{\mathsf{M}}, \widetilde{\mathsf{P}}$ be the associated matrices constructed respectively in* (9), (12), (37) *and* (42)*, and $M_{\max} > 0$ be a constant such that $\|\mathsf{PM}\|_2 \leqslant M_{\max}$. Let $\xi, \delta \in (0,1)$ and suppose that s satisfies* (17)*. With probability at least $1-\xi$, the following holds for all $\boldsymbol{x} \in \mathcal{A}_\epsilon$, all $\tilde{\boldsymbol{n}} \in \mathbb{R}^s$, all $\gamma > 0$, and all nonnegative nondecreasing polynomial functions g such that $g(\lambda_{k+1}) > 0$.*

Let $\tilde{\boldsymbol{x}}^*$ be the solution of (41) *with $\tilde{\boldsymbol{y}} = \widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{x} + \tilde{\boldsymbol{n}}$. Define $\boldsymbol{\alpha}^* := \mathsf{U}_k\mathsf{U}_k^\intercal\,\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^*$ and $\boldsymbol{\beta}^* := (\mathsf{I} - \mathsf{U}_k\mathsf{U}_k^\intercal)\,\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^*$. Then,*

$$\begin{aligned}\|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 \;\leqslant\; \frac{1}{\sqrt{s(1-\delta)}} \cdot \Bigg[&\left(2 + \frac{M_{\max}}{\sqrt{\gamma g(\lambda_{k+1})}}\right) \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2 + \left(M_{\max}\sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_k)}\right) \|\boldsymbol{x}\|_2 \\ &+ \epsilon\left(2M_{\max} + M_{\max}\sqrt{\frac{g(\lambda_n)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_n)}\right) \|\boldsymbol{x}\|_2 \Bigg],\end{aligned} \tag{44}$$

and

$$\|\boldsymbol{\beta}^*\|_2 \leqslant \frac{1}{\sqrt{\gamma g(\lambda_{k+1})}} \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2 + \sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}}\,\|\boldsymbol{x}\|_2 + \epsilon\sqrt{\frac{g(\lambda_n)}{g(\lambda_{k+1})}}\,\|\boldsymbol{x}\|_2 . \tag{45}$$

Proof. See Appendix B. □

The vector $\boldsymbol{\alpha}^*$ is the orthogonal projection of $\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^*$ onto $\mathrm{span}(\mathsf{U}_k)$. The vector $\boldsymbol{\beta}^*$ is the projection of $\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^*$ onto the orthogonal complement of $\mathrm{span}(\mathsf{U}_k)$. There are several remarks to make about the above theorem:

- Theorem 3.1 shows that the result obtained via (41) is similar to the one we would have obtained by solving (28) – see [17] for the error bounds – with additional errors controlled by ϵ. We recall that ϵ characterises how far $\boldsymbol{x}$ is from a piecewise constant signal. As expected, the smaller ϵ, the better the reconstruction.
- The reconstruction quality improves when $g(\lambda_k)$ and the ratio $g(\lambda_k)/g(\lambda_{k+1})$ go to 0, and when $g(\lambda_n)/g(\lambda_{k+1})$ tends to 1. We recall that we have $g(\lambda_n) \geqslant g(\lambda_{k+1}) > g(\lambda_k)$ by assumption.
- The effect of the noise $\tilde{\boldsymbol{n}}$ decreases when $g(\lambda_{k+1})$ increases, and, obviously, γ should be adapted to the signal-to-noise ratio.

Let us mention that the idea of "coarsening" a graph and the signals that live on it using a partition into different groups of nodes can also be found in [30], where a multiresolution analysis method of graph signals is proposed. The coarsening method is however different than the one used here.

4. Optimal sampling estimation

In this section, we come back to the sampling process of Section 2 and leave the reconstruction problem. We explain how to estimate the sampling distributions $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$ of Section 2 without computing the truncated Fourier matrix U_k, as this computation is intractable for large graphs. The methods below only involve matrix-vector multiplications with the sparse Laplacian matrix L and are thus computationally tractable even for large n.

4.1. Estimation of $\boldsymbol{p}^*$

The distribution $\boldsymbol{p}^*$, defined in (18), that minimises the coherence $\nu_{\boldsymbol{p}}$ is entirely defined by the values of $\|\mathsf{N}^{(\ell)}\mathsf{U}_k\|_2$ for $\ell = 1, \dots, N$, which are thus the quantities we need to evaluate.

We recall that, in graph signal processing, a filter is represented by a function $h : \mathbb{R} \to \mathbb{R}$, and that the signal $\boldsymbol{x}$ filtered by h is

$$\boldsymbol{x}_h := \mathsf{U}\,\mathrm{diag}(\hat{\boldsymbol{h}})\,\mathsf{U}^\intercal \boldsymbol{x} \in \mathbb{R}^n, \tag{46}$$

where $\hat{\boldsymbol{h}} = (h(\lambda_1), \dots, h(\lambda_n))^\intercal \in \mathbb{R}^n$. To filter the signal $\boldsymbol{x}$ without actually computing the graph Fourier transform of $\boldsymbol{x}$, we can approximate the function h by a polynomial

$$r(t) := \sum_{i=0}^{d} \alpha_i\, t^i \approx h(t) \tag{47}$$

of degree d, and compute $\boldsymbol{x}_r$ instead of $\boldsymbol{x}_h$. The filtered signal $\boldsymbol{x}_r$ is computed rapidly using the formula

$$\boldsymbol{x}_r = \sum_{i=0}^{d} \alpha_i\, \mathsf{U}\,\mathrm{diag}(\lambda_1^i, \dots, \lambda_n^i)\,\mathsf{U}^\intercal \boldsymbol{x} = \sum_{i=0}^{d} \alpha_i\, \mathsf{L}^i \boldsymbol{x}, \tag{48}$$

that involves only matrix-vector multiplications with the sparse Laplacian matrix L. We let the reader refer to [31] for more information on this fast filtering technique. For any polynomial function r of the form above and any matrix $\mathsf{A} \in \mathbb{R}^{n\times n}$, we define

$$r(\mathsf{A}) := \sum_{i=0}^{d} \alpha_i\, \mathsf{A}^i. \tag{49}$$

Note that $r(\mathsf{L}) = \mathsf{U}\, r(\Lambda)\, \mathsf{U}^\intercal$.

Let $i_{\lambda_k} : \mathbb{R} \to \mathbb{R}$ be the ideal low-pass filter at cutoff frequency λ_k, *i.e.*, the filter that satisfies

$$i_{\lambda_k}(t) = \begin{cases} 1 & \text{if } t \leqslant \lambda_k, \\ 0 & \text{otherwise.} \end{cases} \tag{50}$$

We have $\mathsf{U}_k\mathsf{U}_k^\intercal = i_{\lambda_k}(\mathsf{L})$. Then, we notice that

$$\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2 = \left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\mathsf{U}_k^\intercal\mathsf{N}^{(\ell)\intercal}\right\|_2 = \left\|\mathsf{N}^{(\ell)}\; i_{\lambda_k}(\mathsf{L})\; \mathsf{N}^{(\ell)\intercal}\right\|_2. \tag{51}$$

We recall that $\mathsf{N}^{(\ell)}$ is the matrix that restricts the signal to the nodes belonging to $\mathcal{N}_\ell$. The matrix appearing on the right hand side of the last equality corresponds to the linear operator that 1) extends a vector on the complete graph by inserting 0 in all groups $\ell' \neq \ell$, 2) low-pass filters the extended signal, 3) restricts the result to the group $\mathcal{N}_\ell$. This process can be approximated by replacing the ideal low-pass filter i_{λ_k} with a polynomial approximation $\tilde{i}_{\lambda_k}$ of i_{λ_k} and

$$\left\|\mathsf{N}^{(\ell)}\; i_{\lambda_k}(\mathsf{L})\; \mathsf{N}^{(\ell)\intercal}\right\|_2 \approx \left\|\mathsf{N}^{(\ell)}\; \tilde{i}_{\lambda_k}(\mathsf{L})\; \mathsf{N}^{(\ell)\intercal}\right\|_2. \tag{52}$$

To estimate $\boldsymbol{p}^*$, we estimate the spectral norm of the matrix appearing on the right hand side for which matrix-vector multiplication is fast. The quality of the approximation depends obviously on the choice of the polynomial $\tilde{i}_{\lambda_k}$.

Estimating $\|\mathsf{N}^{(\ell)}\ \tilde{i}_{\lambda_k}(\mathsf{L})\ \mathsf{N}^{(\ell)\mathsf{T}}\|_2$ amounts to computing the largest eigenvalue of $\mathsf{N}^{(\ell)}\ \tilde{i}_{\lambda_k}(\mathsf{L})\ \mathsf{N}^{(\ell)\mathsf{T}}$ which can be done, *e.g.*, by using the power method. This method requires matrix-vector multiplication only with $\mathsf{N}^{(\ell)}$ and $\tilde{i}_{\lambda_k}(\mathsf{L})$ and is thus fast. Finally, the approximation $\bar{\boldsymbol{p}} \in \mathbb{R}^N$ of $\boldsymbol{p}^*$ satisfies

$$\bar{\boldsymbol{p}}_\ell := \frac{\lambda_{\max}(\mathsf{N}^{(\ell)}\ \tilde{i}_{\lambda_k}(\mathsf{L})\ \mathsf{N}^{(\ell)\mathsf{T}})}{\sum_{\ell'=1}^{N} \lambda_{\max}(\mathsf{N}^{(\ell')}\ \tilde{i}_{\lambda_k}(\mathsf{L})\ \mathsf{N}^{(\ell')\mathsf{T}})}. \tag{53}$$

Note that an estimation of λ_k is required beforehand to define the filter $\tilde{i}_{\lambda_k}$. We estimate this value using the dichotomy method presented in [17].

4.2. Estimation of $\boldsymbol{q}^*$

Computing $\bar{\boldsymbol{p}}$ requires the estimation of N eigenvalues. Even though these estimations can be done in parallel, this process might still be too slow for certain applications. As explained before, when k is small, we can use the sampling distribution $\boldsymbol{q}^*$ in (26) that minimises $\bar{\nu}_{\boldsymbol{p}}$. This distribution is faster to compute than $\bar{\boldsymbol{p}}$.

We start by noticing that we have

$$\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_F^2 = \sum_{i\in\mathcal{N}_\ell} \|\mathsf{U}_k^\mathsf{T}\boldsymbol{\delta}_i\|_2^2 \tag{54}$$

for each group $\ell = 1, \dots, N$. The vector $\boldsymbol{\delta}_i \in \mathbb{R}^n$ is the unit vector that is null on all nodes except at node i. Hence, we only need an estimation of $\|\mathsf{U}_k^\mathsf{T}\boldsymbol{\delta}_i\|_2^2$, $i = 1, \dots, n$, to estimate $\boldsymbol{q}^*$. An algorithm was already proposed in [17] to estimate these values. We let the reader refer to Algorithm 1 in [17] for the details of the method. We just recall that this estimation is obtained by filtering $O(\log(n))$ random signals with a polynomial approximation of i_{λ_k}. Finally, our estimation $\bar{\boldsymbol{q}} \in \mathbb{R}^N$ of $\boldsymbol{q}^*$ has entries

$$\bar{\boldsymbol{q}}_\ell := \frac{\sum_{i\in\mathcal{N}_\ell} \|\mathsf{U}_k^\mathsf{T}\boldsymbol{\delta}_i\|_F^2}{\sum_{\ell=1}^{N}\sum_{i\in\mathcal{N}_\ell} \|\mathsf{U}_k^\mathsf{T}\boldsymbol{\delta}_i\|_F^2}, \tag{55}$$

where each $\|\mathsf{U}_k^\mathsf{T}\boldsymbol{\delta}_i\|_2^2$ are estimated by Algorithm 1 in [17].

This estimation is faster than for $\bar{\boldsymbol{p}}$ because the power method is an iterative method that involves one filtering at each iteration. Furthermore, the power method is run independently for each group $\mathcal{N}_\ell$. In total, (much) more than $\mathcal{N}_\ell$ filterings are thus required. On the contrary, for $\bar{\boldsymbol{q}}$, we just need to filter $O(\log(n))$ signals to obtain the estimation. In most situations, we already have $O(\log(n)) \leqslant \mathcal{N}_\ell$ and computing $\bar{\boldsymbol{q}}$ is thus faster than computing $\bar{\boldsymbol{p}}$.

5. Experiments

In this last section, we first test our sampling strategies on two different graphs to illustrate the effect of the different sampling distributions on the minimum number of samples required to ensure that the RIP holds. Then, we apply our sampling strategy for user-guided object segmentation. In this application, we also test the different recovery techniques proposed in Section 3.

5.1. Sampling distributions

We perform experiments on two different graphs: the Minnesota graph of size $n = 2642$ and the bunny graph of size $n = 2503$. Both graphs are presented in Fig. 3 and are available in the GSP toolbox [32]. For each graph, we group the nodes using the spatial coordinates associated to each node. For the Minnesota graph, we divide the space into 100 cells and group the nodes that fall in the same cell. After removing empty cells, we obtain the $N = 73$ groups represented in Fig. 3. For the bunny graph, we obtain $N = 213$ groups with a similar procedure (see Fig. 3).

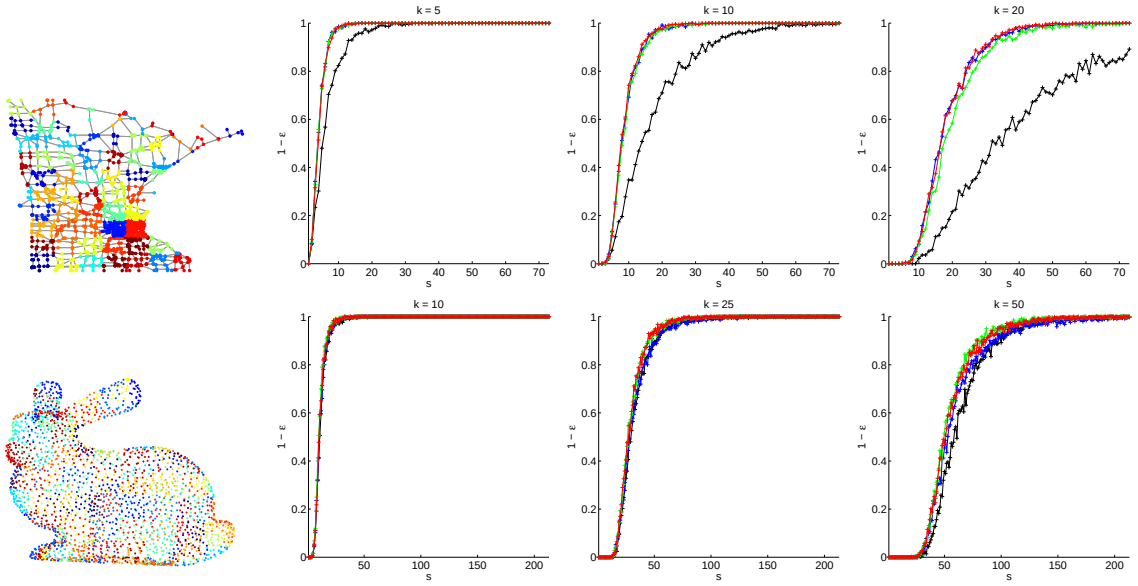


Figure 3. *First column from the left: Minnesota graph (top); bunny graph (bottom). The groups $\mathcal{N}_1, \dots, \mathcal{N}_N$ are indicated by different colors. Other columns: probability that $\underline{\delta}_k$ is less that* 0.995 *as a function of s. The curves in black, red, blue and green are obtained using the sampling distributions $\boldsymbol{u}$, $\boldsymbol{p}^*$, $\bar{\boldsymbol{q}}$ and $\bar{\boldsymbol{p}}$, respectively. The top row shows the result for the Minnesota graph. The bottom row shows the result for the bunny graph. The bandlimit k is indicated on top of each curve.*

For each graph, we compute the combinatorial Laplacian and U_k for different values of k. Then, we compute the lower RIP constant, *i.e.*, the constant $\underline{\delta}_k > 0$ that satisfies

$$\underline{\delta}_k = 1 - \frac{1}{s} \inf_{\substack{\boldsymbol{x} \in \mathrm{span}(\mathsf{U}_k) \\ \|x\|_2 = 1}} \|\mathsf{PM}\, \boldsymbol{x}\|_2^2 .$$

This constant is the smallest value that δ can take such that the left-hand side of the RIP (16) holds. Remark that

$$\underline{\delta}_k = 1 - \frac{1}{s} \lambda_{\min} \left(\mathsf{U}_k^\intercal \mathsf{M}^\intercal \mathsf{P}^2 \mathsf{M} \mathsf{U}_k \right) . \tag{56}$$

We estimate $\underline{\delta}_k$ for 500 independent draws of the set Ω, which defines the matrices PM, and different numbers of selected groups s. All samplings are done in the conditions of Theorem 2.2 using the sampling distributions $\boldsymbol{u}, \boldsymbol{p}^*, \bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$. The vector $\boldsymbol{u}$ denotes the uniform distribution over $\{1, \dots, N\}$. When conducting this experiment with the estimated distributions $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$, we re-estimate these distributions at each of the 500 trials. These distributions are estimated using Jackson-Chebychev polynomials of order 50 [33]. For the Minnesota graph, we consider the bandlimits $k = 5, 10, 20$. For the bunny graph, we consider the bandlimits $k = 10, 25, 50$.

We present the probability that $\underline{\delta}_k$ is less than 0.995, estimated over the 500 draws of Ω, as a function of s in Fig. 3. For the Minnesota graph, the performance is better when using the optimal distribution $\boldsymbol{p}^*$ than when using the uniform distribution $\boldsymbol{u}$ for all k, which is in line with the theory. The estimated $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$ yield performance equivalent to $\boldsymbol{p}^*$. This confirms that we can achieve similar sampling performance without having to compute the Fourier matrix U_k, which, we recall, is intractable for large graphs. This also shows that $\bar{\boldsymbol{q}}$ can lead to nearly optimal results. For the bunny graph, all sampling distributions yield essentially the same results at all bandlimits. We notice a slight improvement at $k = 50$ when using $\bar{\boldsymbol{p}}$, $\bar{\boldsymbol{q}}$ or $\boldsymbol{p}^*$ instead of $\boldsymbol{u}$.

For illustration, we present in Fig. 4 examples of computed sampling distributions $\boldsymbol{p}^*$, $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$. All sampling distributions exhibit similar structures, which explains why they all yield about the same performance in our experiments.

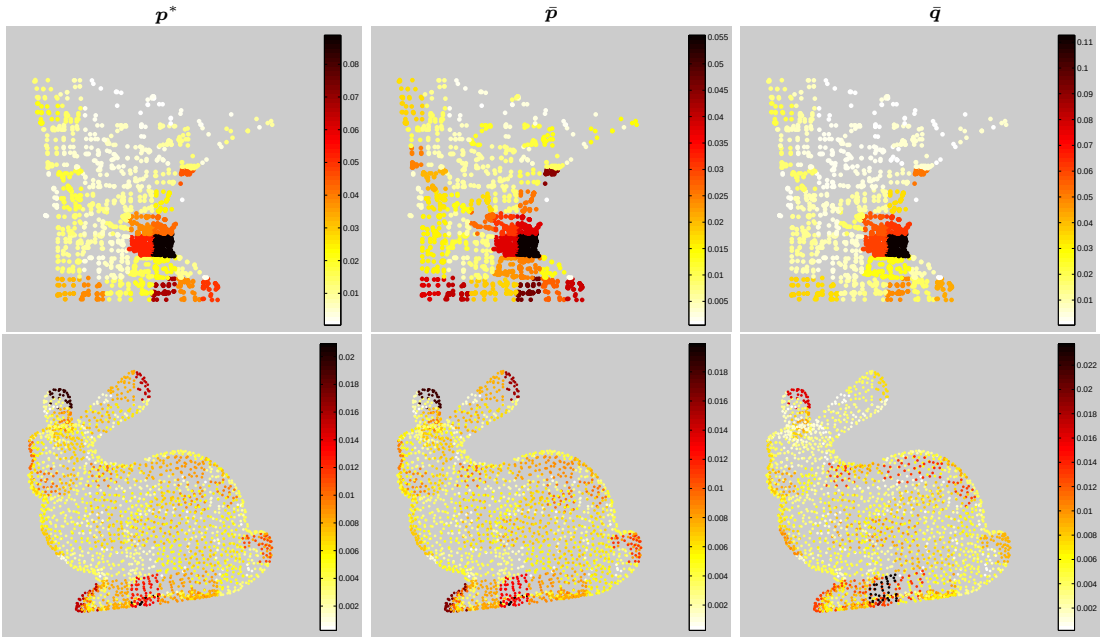


Figure 4. *Example of sampling distributions. Top panels:* $\boldsymbol{p}^*$ *(left),* $\bar{\boldsymbol{p}}$ *(middle), and* $\bar{\boldsymbol{q}}$ *(right) for the Minnesota graph at* $k = 10$*. Bottom panels:* $\boldsymbol{p}^*$ *(left),* $\bar{\boldsymbol{p}}$ *(middle), and* $\bar{\boldsymbol{q}}$ *(right) for the bunny graph at* $k = 25$*.*

5.2. Object segmentation

5.2.1. Protocol

We now test our method for interactive object segmentation. We consider the image of size $n = 321 \times 481$ presented in Fig. 1 for which our goal is to segment the tiger. The ground truth segmentation map $\boldsymbol{x} \in \{0,1\}^n$ is presented in Fig. 5. The value 1 (white) indicates the presence of the tiger and the value 0 (black) stands for the foreground. The original image and the ground truth image are part of the dataset available[2] in [34]. Our objective is to recover the original map $\boldsymbol{x}$ from few user-inputs. To facilitate the interactions with the user, we divide the original image into the $N = 600$ superpixels showed in Fig. 1 and computed with SLIC [23], choose a small number of superpixels at random and ask the user to label these superpixels: 1 if the superpixel belongs to the tiger; 0 otherwise.

The graph $\mathcal{G}$ used to propagate the user-labels to the complete image is constructed as follows. We build a feature vector for each pixel by extracting a color RGB patch of size 3×3 around the pixel, transform this patch in vector form, and augment this vector with the absolute 2D coordinates of the pixels in this extracted patch. This yields n feature vectors $\boldsymbol{g}_i \in \mathbb{R}^{45}$, $i = 1, \ldots, n$. We then connect each feature vector to its 9 nearest neighbours (in the Euclidean sense), which gives a set of $9n$ edges $\mathcal{E}$. The adjacency matrix $\mathsf{W} \in \mathbb{R}^{n \times n}$ satisfies $\mathsf{W}_{ij} := \exp[-\|\boldsymbol{g}_i - \boldsymbol{g}_j\|_2^2/\sigma^2]$, where $\sigma > 0$ is the 25^{th} percentile of the set $\{\|\boldsymbol{g}_i - \boldsymbol{g}_j\|_2 : (i,j) \in \mathcal{E}\}$. We finally symmetrise the matrix W and compute the combinatorial Laplacian $\mathsf{L} \in \mathbb{R}^{n \times n}$.

We study three strategies to choose the superpixels. The first strategy consists in choosing the superpixels uniformly at random, *i.e.*, using the sampling distribution $\boldsymbol{u}$. The second and third strategies consist in choosing the superpixels with respectively the optimised distributions $\bar{\boldsymbol{q}}$ and $\bar{\boldsymbol{p}}$, which we evaluate at $k_0 = 50$ using Jackson-Chebychev polynomials of order 75 [33]. For illustration, we present in Fig. 5 the estimated values $\|\mathsf{N}^{(\ell)}\mathsf{U}_{k_0}\|_F^2$ and $\|\mathsf{N}^{(\ell)}\mathsf{U}_{k_0}\|_2^2$, which define the optimised distributions $\bar{\boldsymbol{q}}$ and $\bar{\boldsymbol{p}}$, respectively. Both distributions indicate that one should label more superpixels around the tiger. The distribution $\bar{\boldsymbol{q}}$ "focuses" however more on specific regions, like the head of tiger. The distribution $\bar{\boldsymbol{p}}$ spreads the measurements over the entire tiger more uniformly.

[2] `http://www.ntu.edu.sg/home/asjfcai/Benchmark_Website/benchmark_index.html`.

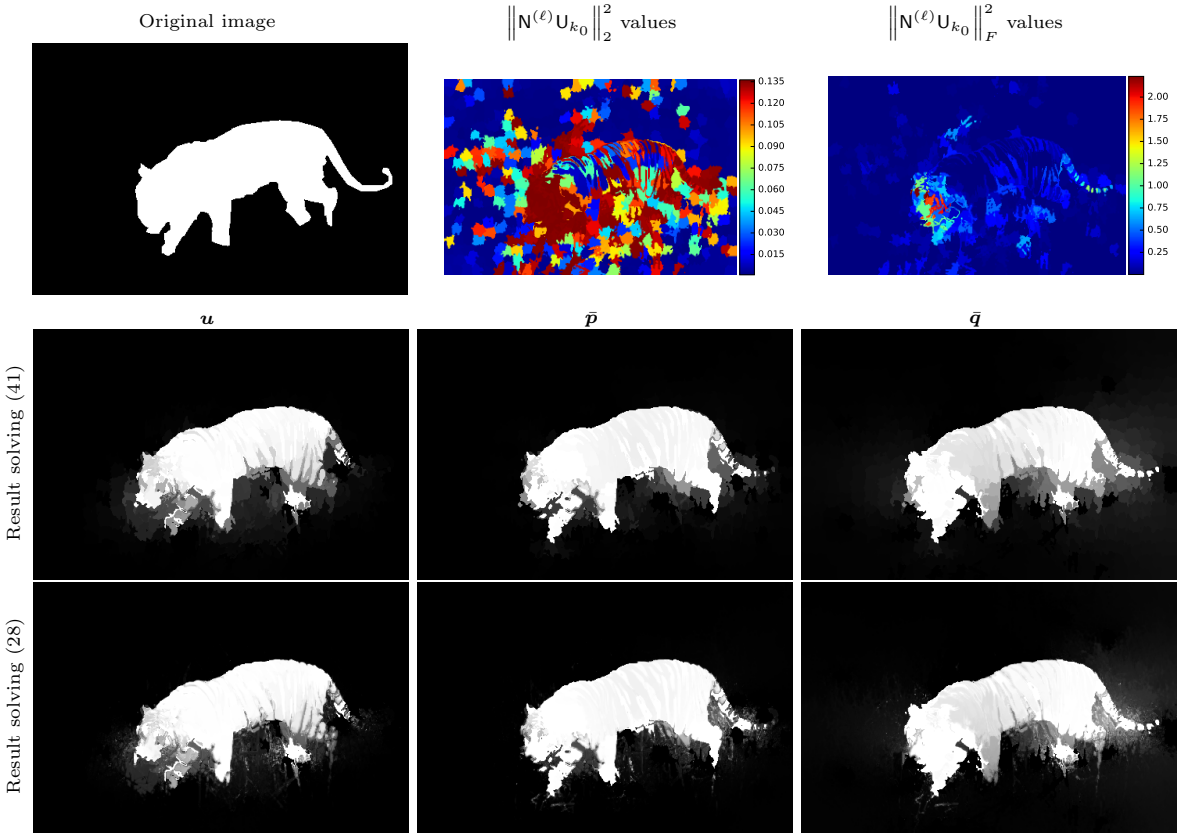


Figure 5. *Top row: ground truth segmentation map (left); local group coherence map* $\|\mathsf{N}^{(\ell)}\mathsf{U}_{k_0}\|_2^2$ *evaluated with the spectral norm (middle); local group coherence map* $\|\mathsf{N}^{(\ell)}\mathsf{U}_{k_0}\|_F^2$ *evaluated with the Frobenius norm (right). Middle row: segmentation map estimated from* $s = 50$ *sampled superpixels obtained by solving* (41) *and using a uniform sampling (left),* $\bar{\boldsymbol{p}}$ *(middle),* $\bar{\boldsymbol{q}}$ *(right). Bottom row: segmentation map estimated from* $s = 50$ *sampled superpixels obtained by solving* (28) *and using a uniform sampling (left),* $\bar{\boldsymbol{p}}$ *(middle),* $\bar{\boldsymbol{q}}$ *(right).*

We emulate user-interactions as follows. For each chosen superpixel, we compute the mean of the ground truth map $\boldsymbol{x}$ within this superpixel. If the mean value is larger than 0.5, we label the superpixel as part of the tiger. Otherwise, we label the superpixel as part of the background. This strategy obviously introduces noise if some superpixels cover part of the background and of the tiger. Once the labelling is done, we have access to the measurement vector $\tilde{\boldsymbol{y}} \in \mathbb{R}^m$ from which we want to reconstruct $\boldsymbol{x}$. We repeat this procedure for $s \in \{50, 70, \dots, 250\}$. For each s, we also repeat the experiments 50 times with independent draws of the superpixels. We draw the superpixels *with* replacements in all cases.

To reconstruct the original map $\boldsymbol{x}$, we first use the fast reconstruction method (41) and then refine the solution at the pixel level with (28), using the solution of the first minimisation problem as initialisation to solve the second minimisation problem. We choose $g(\mathsf{L}) = \mathsf{L}$ and solve both problems in the limit where $\gamma \to 0$. In this limit, the problems (41) and (28) become

$$\min_{\tilde{\boldsymbol{z}} \in \mathbb{R}^N} \tilde{\boldsymbol{z}}^\intercal \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{z}} \quad \text{subject to} \quad \widetilde{\mathsf{M}}\tilde{\boldsymbol{z}} = \tilde{\boldsymbol{y}} \tag{57}$$

and

$$\min_{\boldsymbol{z} \in \mathbb{R}^n} \boldsymbol{z}^\intercal g(\mathsf{L})\boldsymbol{z} \quad \text{subject to} \quad \mathsf{M}\boldsymbol{z} = \boldsymbol{y}, \tag{58}$$

respectively. Both problems are solved using FISTA [35]. The same stopping criteria are used for all experiments.

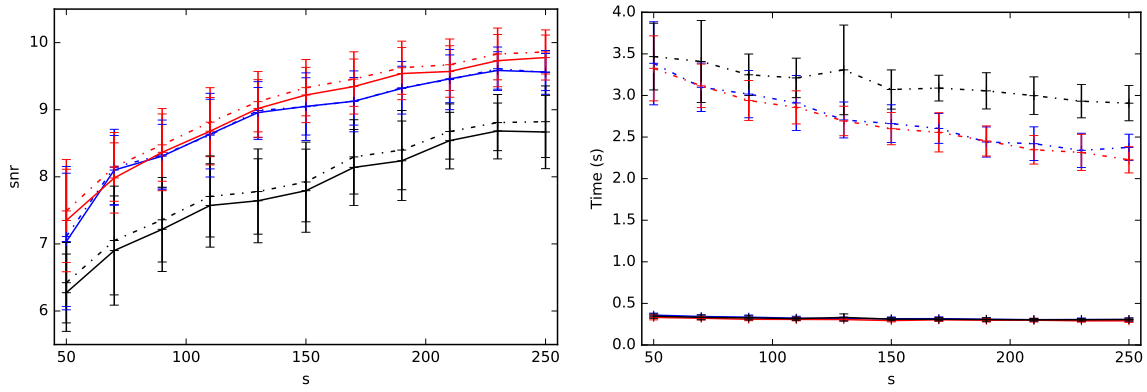


Figure 6. *Left: the curves represent the mean reconstruction snr as a function of the number of sampled superpixels s. The dash-dotted curves are obtained by solving* (28)*. The continuous curves are obtained by solving* (41)*. Right: mean computation time in seconds as a function of s. The dash-dotted curves are obtained by solving* (28)*. The continuous curves are obtained by solving* (41) *using the result of* (28) *as initial point. In both graphs, the errorbars are at one standard deviation. The black curves are obtained with the uniform sampling distribution $\boldsymbol{u}$. The blue curves are obtained with $\bar{\boldsymbol{q}}$. The red curves are obtained with $\bar{\boldsymbol{p}}$.*

5.2.2. Results

We present in the top panel of Fig. 6 the reconstruction snr obtained with the different methods. The reconstruction snr is defined as $-20\log_{10}(-\|\boldsymbol{x}-\boldsymbol{x}^*\|_2 / \|\boldsymbol{x}\|_2)$, where $\boldsymbol{x}^*$ is the reconstructed signal. We notice that the snr attained with the fast decoder (41) is very similar to the snr attained with (28). We also remark that the optimised distributions $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$ yield better reconstructions than the uniform distribution $\boldsymbol{u}$. The mean reconstruction snr is slightly better with $\bar{\boldsymbol{p}}$ than with $\bar{\boldsymbol{q}}$ at $s \geqslant 150$.

We present the computation time of each method in the bottom panel of Fig. 6. We notice that solving (41) is much faster than solving (28), while they yield almost the same quality. This highlight the interest of the fast reconstruction technique. It is also interesting to note that it is faster to solve (28) when the measurements are drawn with $\bar{\boldsymbol{p}}$ or with $\bar{\boldsymbol{q}}$ than with $\boldsymbol{u}$. The reason is probably a better initialisation of (28) or a better "quality" of the measurements with the optimised distributions than with the uniform distribution.

Finally, we present in Fig. 5 some examples of reconstructions from $s = 150$ sampled superpixels for each method. We notice that the optimised sampling distributions improve the reconstruction of $\boldsymbol{x}$ around the head and tail of the tiger, *i.e.*, where the optimised distributions have higher values. With a uniform distribution, the structure of the graph makes it difficult to reconstruct $\boldsymbol{x}$ around the head and tail from the values of other superpixels. The optimised sampling distribution compensate this issue by favouring this area when selecting the measurements.

6. Discussion and Conclusion

We presented a structured sampling strategy for k-bandlimited signals where the nodes are selected by groups. We proved that the local group graph cumulative coherence quantifies the importance of sampling each group to ensure a stable embedding of all k-bandlimited signals. Finally, we presented a fast reconstruction technique for k-bandlimited signals which are also nearly piecewise-constant over pre-defined groups of nodes.

Among the possible applications of these methods, we believe that they can also be useful to accelerate the compressive spectral clustering method proposed in [36]. After having computed some features vectors for each nodes, this compressive method works by downsampling the set of features of vectors, performing k-means on this reduced set to find k clusters, and interpolating the clustering results on all nodes by solving (28). To accelerate the method, one could 1) pre-group similar nodes to form N groups such that $k \leqslant N \ll n$, *e.g.*, by running few iterations of the k-means algorithm; 2)

subsample this set of N groups; 3) cluster this subset to find k clusters; and 4) solve (41) to cluster all nodes. If the overhead of computing the N groups is small, this method has the potential to be faster than the original compressive spectral clustering method.

Finally, we would like to discuss two limitations in the proposed methods. First, the optimal sampling distribution depends on the parameter k. In some applications, the final result may change a lot depending on the value of k which was chosen to compute this distribution. Finding a range of values of k which give acceptable and stable results is thus an important step in every application. Second, the estimation of the optimal sampling distribution depends on the quality of the polynomial approximation of the ideal low-pass filter λ_{λ_k}. It is sometimes necessary to use a polynomial of large degree to get a correct estimation, which limits the computational efficiency of the proposed methods. In such cases, it would be especially useful to find more efficient alternatives to estimate the distributions $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$.

Appendix A - Proof of the Theorem 2.2

As done in [17], the proof is obtained by applying the following lemma obtained by Tropp in [37].

Lemma A.1 (Theorem 1.1, [37]). *Consider a finite sequence $\{\mathsf{X}_j\}$ of independent, random, self-adjoint, positive semi-definite matrices of dimension $d \times d$. Assume that each random matrix satisfies $\lambda_{\max}(\mathsf{X}_j) \leqslant R$ almost surely. Define*

$$\mu_{\min} := \lambda_{\min}\left(\sum_j \mathbb{E}\,\mathsf{X}_j\right) \text{ and } \mu_{\max} := \lambda_{\max}\left(\sum_j \mathbb{E}\,\mathsf{X}_j\right). \tag{59}$$

Then

$$\mathbb{P}\left\{\lambda_{\min}\left(\sum_j \mathsf{X}_j\right) \leqslant (1-\delta)\mu_{\min}\right\} \leqslant d\,\left[\frac{\mathrm{e}^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\frac{\mu_{\min}}{R}} \text{ for } \delta \in [0,1], \tag{60}$$

$$\text{and } \mathbb{P}\left\{\lambda_{\max}\left(\sum_j \mathsf{X}_j\right) \geqslant (1+\delta)\mu_{\max}\right\} \leqslant d\,\left[\frac{\mathrm{e}^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\frac{\mu_{\max}}{R}} \text{ for } \delta \geqslant 0. \tag{61}$$

We will also use the facts that, for all $\delta \in [0,1]$,

$$\left[\frac{\mathrm{e}^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\mu_{\min}/R} \leqslant \exp\left(-\frac{\delta^2\mu_{\min}}{3\,R}\right) \text{ and } \left[\frac{\mathrm{e}^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu_{\max}/R} \leqslant \exp\left(-\frac{\delta^2\mu_{\max}}{3\,R}\right). \tag{62}$$

Proof of Theorem 2.2. We start by noticing that

$$\frac{1}{s}\,\mathsf{U}_k^\intercal \mathsf{M}^\intercal \mathsf{P}\mathsf{P}\mathsf{M}\mathsf{U}_k = \frac{1}{s}\sum_{j=1}^{s}\left(\mathsf{U}_k^\intercal \mathsf{N}^{(\omega_j)\intercal}\mathsf{P}^{(\omega_j)}\right)\left(\mathsf{P}^{(\omega_j)}\mathsf{N}^{(\omega_l)}\mathsf{U}_k\right). \tag{63}$$

We define

$$\mathsf{X}_j := \frac{1}{s}\left(\mathsf{U}_k^\intercal \mathsf{N}^{(\omega_j)\intercal}\mathsf{P}^{(\omega_j)}\right)\left(\mathsf{P}^{(\omega_j)}\mathsf{N}^{(\omega_j)}\mathsf{U}_k\right) \quad \text{and} \quad \mathsf{X} := \sum_{j=1}^{s}\mathsf{X}_j = \frac{1}{s}\,\mathsf{U}_k^\intercal \mathsf{M}^\intercal \mathsf{P}^2 \mathsf{M}\mathsf{U}_k. \tag{64}$$

The matrix X is thus a sum of s independent, random, self-adjoint, positive semi-definite matrices. We are in the setting of Lemma A.1. We continue by computing $\mathbb{E}\,\mathsf{X}_j$ and $\lambda_{\max}(\mathsf{X}_j)$.

The expected value of each X_j is

$$\mathbb{E}\,\mathsf{X}_j = \mathbb{E}\left[\frac{1}{s}\left(\mathsf{U}_k^\intercal \mathsf{N}^{(\omega_j)\intercal}\mathsf{P}^{(\omega_j)}\right)\left(\mathsf{P}^{(\omega_j)}\mathsf{N}^{(\omega_j)}\mathsf{U}_k\right)\right] = \frac{1}{s}\,\mathsf{U}_k^\intercal\left(\sum_{\ell=1}^{N}\boldsymbol{p}_\ell\left(\mathsf{N}^{(\ell)\intercal}\mathsf{P}^{(\ell)}\right)\left(\mathsf{P}^{(\ell)}\mathsf{N}^{(\ell)}\right)\right)\mathsf{U}_k$$

$$= \frac{1}{s}\,\mathsf{U}_k^\intercal\left(\sum_{\ell=1}^{N}\mathsf{N}^{(\ell)\intercal}\mathsf{N}^{(\ell)}\right)\mathsf{U}_k = \frac{1}{s}\,\mathsf{U}_k^\intercal\mathsf{U}_k = \frac{1}{s}\,\mathsf{I}. \tag{65}$$

Therefore, $\lambda_{\min}\left(\sum_j \mathbb{E}\,\mathsf{X}_j\right) = 1$ and $\lambda_{\max}\left(\sum_j \mathbb{E}\,\mathsf{X}_j\right) = 1$. Furthermore, for all $j = 1, \ldots, s$, we have

$$\lambda_{\max}(\mathsf{X}_j) = \|\mathsf{X}_j\|_2 \leqslant \max_{1\leqslant\ell\leqslant N} \left\| \frac{\mathsf{P}^{(\ell)}\mathsf{N}^{(\ell)}\mathsf{U}_k}{s} \right\|_2^2 = \frac{1}{s} \max_{1\leqslant\ell\leqslant N} \left\{ \frac{\left\|\mathsf{N}^{(\ell)}\mathsf{U}_k\right\|_2^2}{\boldsymbol{p}_\ell} \right\} = \frac{\nu_{\boldsymbol{p}}^2}{s}. \tag{66}$$

Lemma A.1 yields, for any $\delta \in (0,1)$,

$$\mathbb{P}\left\{\lambda_{\min}\left(\mathsf{X}\right) \leqslant (1-\delta)\right\} \;\leqslant\; k \cdot \left[\frac{\mathrm{e}^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{s/\nu_{\boldsymbol{p}}^2} \;\leqslant\; k\, \exp\left(-\frac{\delta^2 s}{3\,\nu_{\boldsymbol{p}}^2}\right) \tag{67}$$

$$\text{and}\;\; \mathbb{P}\left\{\lambda_{\max}\left(\mathsf{X}\right) \geqslant (1+\delta)\right\} \;\leqslant\; k \cdot \left[\frac{\mathrm{e}^{\delta}}{(1+\delta)^{1+\delta}}\right]^{s/\nu_{\boldsymbol{p}}^2} \;\leqslant\; k\, \exp\left(-\frac{\delta^2 s}{3\,\nu_{\boldsymbol{p}}^2}\right). \tag{68}$$

Therefore, for any $\delta \in (0,1)$, we have, with probability at least $1-\xi$,

$$1-\delta \leqslant \lambda_{\min}\left(\mathsf{X}\right) \quad \text{and} \quad \lambda_{\max}\left(\mathsf{X}\right) \leqslant 1+\delta \tag{69}$$

provided that (17) holds. Noticing that (69) implies that

$$(1-\delta)\,\|\boldsymbol{\alpha}\|_2^2 \leqslant \frac{1}{s}\,\|\mathsf{P}\mathsf{M}\mathsf{U}_k\boldsymbol{\alpha}\|_2^2 \leqslant (1+\delta)\,\|\boldsymbol{\alpha}\|_2^2\,, \tag{70}$$

for all $\boldsymbol{\alpha} \in \mathbb{R}^k$, which is equivalent to (16) for all $\boldsymbol{x} \in \mathrm{span}(\mathsf{U}_k)$, terminates the proof. □

Appendix B - Proof of Theorem 3.1

In order to prove Theorem 3.1, we need to establish few properties between the different matrices used in this work.

The first useful property is

$$\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A} = \widetilde{\mathsf{A}}\mathsf{P}\mathsf{M}. \tag{71}$$

Indeed, for any $\boldsymbol{z} \in \mathbb{R}^n$, the j^{th} entry of $\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{z}$ is

$$\left(\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{z}\right)_j = \frac{\mathbf{1}^\intercal \mathsf{N}^{(\omega_j)}\boldsymbol{z}}{\left(\boldsymbol{p}_{\omega_j}\;\left|\mathcal{N}_{\omega_j}\right|\right)^{1/2}}. \tag{72}$$

Then, the j^{th} entry of $\widetilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{z}$ is the scaled sum of the values in the j^{th} sampled group appearing in $\mathsf{P}\mathsf{M}\boldsymbol{z}$, which is $\boldsymbol{p}_{\omega_j}^{-1/2}\mathsf{N}^{(\omega_j)}\boldsymbol{z}$. From the definition of $\widetilde{\mathsf{A}}$, the sum is scaled by $\left|\mathcal{N}_{\omega_j}\right|^{-1/2}$. Therefore, $(\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{z})_j = (\widetilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{z})_j$ for all $j \in \{1, \ldots, s\}$, which terminates the proof.

The second property is

$$\widetilde{\mathsf{A}}\widetilde{\mathsf{A}}^\intercal = \mathsf{I}, \tag{73}$$

which implies $\|\widetilde{\mathsf{A}}\|_2 = 1$.

The third property is

$$\|\widetilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}\|_2 = \|\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}\|_2\,, \tag{74}$$

for all $\tilde{\boldsymbol{z}} \in \mathbb{R}^N$. To prove this property, we remark that $\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}} = \widetilde{\mathsf{A}}^\intercal\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}}$. Indeed, the entries in $\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}$ corresponding to the first selected group $\mathcal{N}_{\omega_1}$ are all equal to $(\boldsymbol{p}_{\omega_1}\,|\mathcal{N}_{\omega_1}|)^{-1/2}\tilde{\boldsymbol{z}}_{\omega_1}$. There are $|\mathcal{N}_{\omega_1}|$ such entries. One can also notice that the first $|\mathcal{N}_{\omega_1}|$ entries of $\widetilde{\mathsf{A}}^\intercal\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}}$ are also equal to $(\boldsymbol{p}_{\omega_j}\,|\mathcal{N}_{\omega_1}|)^{-1/2}\tilde{\boldsymbol{z}}_{\omega_1}$. Repeating this reasoning for all the sampled groups proves the equality. On the one side, we thus have $\|\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}\|_2 = \|\widetilde{\mathsf{A}}^\intercal\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}}\|_2 = \|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}}\|_2$, where we used (73). On the other side, we have $\|\widetilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}\|_2 = \|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\mathsf{A}^\intercal\tilde{\boldsymbol{z}}\|_2 = \|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{z}}\|_2$, where we used (71) and (32). This terminates the proof.

Proof of Theorem 3.1. As $\boldsymbol{x}^*$ is a minimiser of (41), we have

$$\left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \tilde{\boldsymbol{y}})\right\|_2^2 + \gamma\, (\tilde{\boldsymbol{x}}^*)^\intercal\, \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{x}}^* \;\leqslant\; \left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}})\right\|_2^2 + \gamma\, \tilde{\boldsymbol{x}}^\intercal\, \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{x}}. \tag{75}$$

To prove the theorem, we need to lower and upper bound the left and right hand sides of (75), respectively. We start with the bound involving $\widetilde{\mathsf{L}}$ and then with the ones involving $\widetilde{\mathsf{M}}$.

[**Bounding the terms in** (75) **involving** $\widetilde{\mathsf{L}}$]. We define the matrices

$$\bar{\mathsf{U}}_k := (\boldsymbol{u}_{k+1}, \ldots, \boldsymbol{u}_n) \in \mathbb{R}^{n\times(n-k)}, \tag{76}$$

$$\mathsf{G}_k := \operatorname{diag}\left(g(\lambda_1), \ldots, g(\lambda_k)\right) \in \mathbb{R}^{k\times k}, \tag{77}$$

$$\bar{\mathsf{G}}_k := \operatorname{diag}\left(g(\lambda_{k+1}), \ldots, g(\lambda_n)\right) \in \mathbb{R}^{(n-k)\times(n-k)}. \tag{78}$$

By definition of $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$, $\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* = \boldsymbol{\alpha}^* + \boldsymbol{\beta}^*$ with $\boldsymbol{\alpha}^* \in \operatorname{span}(\mathsf{U}_k)$ and $\boldsymbol{\beta}^* \in \operatorname{span}(\bar{\mathsf{U}}_k)$. We recall that $\widetilde{\mathsf{L}} = (\mathsf{AU})\, g(\Lambda)\, (\mathsf{AU})^\intercal$. Therefore, we obtain

$$(\tilde{\boldsymbol{x}}^*)^\intercal\, \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{x}}^* = (\mathsf{U}_k^\intercal\boldsymbol{\alpha}^*)^\intercal\, \mathsf{G}_k\, (\mathsf{U}_k^\intercal\boldsymbol{\alpha}^*) + (\bar{\mathsf{U}}_k^\intercal\boldsymbol{\beta}^*)^\intercal\, \bar{\mathsf{G}}_k\, (\bar{\mathsf{U}}_k^\intercal\boldsymbol{\beta}^*) \geqslant g(\lambda_{k+1})\, \|\boldsymbol{\beta}^*\|_2^2\,. \tag{79}$$

In the first step, we used the facts that $\bar{\mathsf{U}}_k^\intercal\boldsymbol{\alpha}^* = \mathbf{0}$ and $\mathsf{U}_k^\intercal\boldsymbol{\beta}^* = \mathbf{0}$. The second step follows form the fact that $\left\|\bar{\mathsf{U}}_k^\intercal\boldsymbol{\beta}^*\right\|_2 = \|\boldsymbol{\beta}^*\|_2$. We also have

$$\begin{aligned}
\tilde{\boldsymbol{x}}^\intercal\, \widetilde{\mathsf{L}}\, \tilde{\boldsymbol{x}} &= (\mathsf{U}^\intercal\mathsf{A}^\intercal\tilde{\boldsymbol{x}})^\intercal\, g(\Lambda)\, (\mathsf{U}^\intercal\mathsf{A}^\intercal\tilde{\boldsymbol{x}}) \;\leqslant\; g(\lambda_k)\, \|\mathsf{U}_k\mathsf{A}^\intercal\tilde{\boldsymbol{x}}\|_2^2 + g(\lambda_n)\, \left\|\bar{\mathsf{U}}_k^\intercal\mathsf{A}^\intercal\tilde{\boldsymbol{x}}\right\|_2^2 \\
&\leqslant g(\lambda_k)\, \|\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\|_2^2 + g(\lambda_n)\, \left\|\bar{\mathsf{U}}_k^\intercal\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\right\|_2^2 \;\leqslant\; g(\lambda_k)\, \|\boldsymbol{x}\|_2^2 + g(\lambda_n)\, \left[\left\|\bar{\mathsf{U}}_k^\intercal\boldsymbol{x}\right\|_2^2 + \left\|\bar{\mathsf{U}}_k^\intercal(\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x} - \boldsymbol{x})\right\|_2^2\right] \\
&\leqslant g(\lambda_k)\, \|\boldsymbol{x}\|_2^2 + \epsilon^2 g(\lambda_n)\, \|\boldsymbol{x}\|_2^2\,.
\end{aligned} \tag{80}$$

The second inequality follows from the facts that $\|\mathsf{U}_k\|_2 = 1$ and $\tilde{\boldsymbol{x}} = \mathsf{A}\boldsymbol{x}$. To obtain the third inequality, we used $\|\mathsf{A}\|_2 = 1$ and the triangle inequality. For the last step, notice that $\left\|\bar{\mathsf{U}}_k^\intercal\boldsymbol{x}\right\|_2 = 0$ (as $\boldsymbol{x} \in \operatorname{span}(\mathsf{U}_k)$), $\left\|\bar{\mathsf{U}}_k\right\|_2 = 1$ and use (33).

[**Bounding the terms in** (75) **involving** $\widetilde{\mathsf{M}}$]. By definition of $\tilde{\boldsymbol{y}}$, it is immediate that

$$\left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}})\right\|_2^2 = \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2^2. \tag{81}$$

For the other term involving $\widetilde{\mathsf{M}}$, the triangle inequality yields

$$\left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\tilde{\boldsymbol{y}}\right\|_2 \geqslant \left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{x}\right\|_2 - \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2.$$

Then, we have

$$\left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{x}\right\|_2 = \left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\mathsf{A}\boldsymbol{x}\right\|_2 = \left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{x}\right\|_2 \tag{82}$$

$$\geqslant \left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\right\|_2 - \left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x} - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{x}\right\|_2. \tag{83}$$

The first equality follows from $\mathsf{A}\mathsf{A}^\intercal = \mathsf{I}$, the second from (71), and the triangle inequality was used in the last step. To summarise, we are at

$$\left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\tilde{\boldsymbol{y}}\right\|_2 \geqslant \left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\right\|_2 - \left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x} - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{x}\right\|_2 - \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2. \tag{84}$$

We continue by lower bounding $\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\|_2$ and upper bounding $\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x} - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\boldsymbol{x}\|_2$ separately.

[Lower bound on $\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\|_2$*].* Equality (74) yields

$$\left\|\tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\right\|_2 = \left\|\mathsf{P}\mathsf{M}\mathsf{A}^\intercal\tilde{\boldsymbol{x}}^* - \mathsf{P}\mathsf{M}\mathsf{A}^\intercal\mathsf{A}\boldsymbol{x}\right\|_2. \tag{85}$$

Using the triangle inequality and the fact that $\boldsymbol{x} \in \mathcal{A}_\epsilon$, we obtain

$$\begin{aligned} \|\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \mathsf{PMA}^\intercal \mathsf{A}\boldsymbol{x}\|_2 &\geqslant \|\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \mathsf{PM}\boldsymbol{x}\|_2 - \|\mathsf{PM}\boldsymbol{x} - \mathsf{PMA}^\intercal \mathsf{A}\boldsymbol{x}\|_2 \\ &\geqslant \|\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \mathsf{PM}\boldsymbol{x}\|_2 - \|\mathsf{PM}\|_2 \, \|\mathsf{A}^\intercal \mathsf{A}\boldsymbol{x} - \boldsymbol{x}\|_2 \\ &\geqslant \|\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \mathsf{PM}\boldsymbol{x}\|_2 - \epsilon \, M_{\max} \, \|\boldsymbol{x}\|_2 \,. \end{aligned} \tag{86}$$

The restricted isometry property (16) then yields

$$\begin{aligned} \|\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \mathsf{PM}\boldsymbol{x}\|_2 = \|\mathsf{PM}\,(\boldsymbol{\alpha}^* - \boldsymbol{x}) + \mathsf{PM}\boldsymbol{\beta}^*\|_2 &\geqslant \|\mathsf{PM}(\boldsymbol{\alpha}^* - \boldsymbol{x})\|_2 - \|\mathsf{PM}\boldsymbol{\beta}^*\|_2 \\ &\geqslant \sqrt{s(1-\delta)} \;\|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 - M_{\max} \, \|\boldsymbol{\beta}^*\|_2 \,. \end{aligned} \tag{87}$$

We used the equality $\mathsf{A}^\intercal \tilde{\boldsymbol{x}}^* = \boldsymbol{\alpha}^* + \boldsymbol{\beta}^*$. We thus proved that

$$\|\tilde{\mathsf{A}}\mathsf{PMA}^\intercal \tilde{\boldsymbol{x}}^* - \tilde{\mathsf{A}}\mathsf{PMA}^\intercal \mathsf{A}\boldsymbol{x}\|_2 \geqslant \sqrt{s(1-\delta)} \;\|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 - M_{\max} \, \|\boldsymbol{\beta}^*\|_2 - \epsilon M_{\max} \, \|\boldsymbol{x}\|_2 \,. \tag{88}$$

[Upper bounding $\|\tilde{\mathsf{A}}\mathsf{PMA}^\intercal \mathsf{A}\boldsymbol{x} - \tilde{\mathsf{A}}\mathsf{PM}\boldsymbol{x}\|_2$*].* We have

$$\left\|\tilde{\mathsf{A}}\mathsf{PMA}^\intercal \mathsf{A}\boldsymbol{x} - \tilde{\mathsf{A}}\mathsf{PM}\boldsymbol{x}\right\|_2 \leqslant \|\tilde{\mathsf{A}}\|_2 \, \|\mathsf{PM}\|_2 \, \|\mathsf{A}^\intercal \mathsf{A}\boldsymbol{x} - \boldsymbol{x}\|_2 \leqslant \epsilon \, M_{\max} \, \|\boldsymbol{x}\|_2 \,. \tag{89}$$

We used the facts that $\|\tilde{\mathsf{A}}\|_2 = 1$ (see (73)) and that $\boldsymbol{x} \in \mathcal{A}_\epsilon$ in the last inequality.

Using the inequalities (88) and (89) in (84), we arrive at

$$\left\|\widetilde{\mathsf{P}}\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \widetilde{\mathsf{P}}\tilde{\boldsymbol{y}}\right\|_2 \geqslant \sqrt{s(1-\delta)} \;\|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 - M_{\max} \, \|\boldsymbol{\beta}^*\|_2 - 2\epsilon M_{\max} \, \|\boldsymbol{x}\|_2 - \left\|\widetilde{\mathsf{P}}\tilde{\boldsymbol{n}}\right\|_2 . \tag{90}$$

[Finishing the proof]. Remark that (75) implies

$$\left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}}^* - \tilde{\boldsymbol{y}})\right\|_2^2 \leqslant \left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}})\right\|_2^2 + \gamma \; \tilde{\boldsymbol{x}}^\intercal \, \widetilde{\mathsf{L}} \, \tilde{\boldsymbol{x}}, \tag{91}$$

$$\text{and} \quad \gamma \; (\tilde{\boldsymbol{x}}^*)^\intercal \, \widetilde{\mathsf{L}} \, \tilde{\boldsymbol{x}}^* \leqslant \left\|\widetilde{\mathsf{P}}(\widetilde{\mathsf{M}}\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}})\right\|_2^2 + \gamma \; \tilde{\boldsymbol{x}}^\intercal \, \widetilde{\mathsf{L}} \, \tilde{\boldsymbol{x}}. \tag{92}$$

Using (79), (80), (81) in (92), we obtain

$$\gamma \; g(\lambda_{k+1}) \, \|\boldsymbol{\beta}^*\|_2^2 \leqslant \; \left\|\widetilde{\mathsf{P}}\boldsymbol{n}\right\|_2^2 + \; \gamma \; g(\lambda_k) \, \|\boldsymbol{x}\|_2^2 + \epsilon^2 \; \gamma \; g(\lambda_n) \, \|\boldsymbol{x}\|_2^2 \,, \tag{93}$$

which implies (45) in Theorem 3.1. It remains to prove (44) to finish the proof.

Using (80), (81) and (90) in (91), we obtain

$$\sqrt{s(1-\delta)} \, \|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 \leqslant 2 \left\|\widetilde{\mathsf{P}}\boldsymbol{n}\right\|_2 + M_{\max} \, \|\boldsymbol{\beta}^*\|_2 + (\sqrt{\gamma g(\lambda_k)} + \epsilon\sqrt{\gamma g(\lambda_n)} + 2\epsilon M_{\max}) \, \|\boldsymbol{x}\|_2 \,. \tag{94}$$

Using (45) to bound $\|\boldsymbol{\beta}^*\|_2$ on the right hand side, we have

$$\begin{aligned} \sqrt{s(1-\delta)} \, \|\boldsymbol{\alpha}^* - \boldsymbol{x}\|_2 \;\leqslant\; & 2 \left\|\widetilde{\mathsf{P}}\boldsymbol{n}\right\|_2 + \frac{M_{\max}}{\sqrt{\gamma g(\lambda_{k+1})}} \left\|\widetilde{\mathsf{P}}\boldsymbol{n}\right\|_2 + M_{\max}\sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} \, \|\boldsymbol{x}\|_2 \\ & + \epsilon \, M_{\max}\sqrt{\frac{g(\lambda_n)}{g(\lambda_{k+1})}} \, \|\boldsymbol{x}\|_2 + \left(\sqrt{\gamma g(\lambda_k)} + \epsilon\sqrt{\gamma g(\lambda_n)} + 2\epsilon M_{\max}\right) \|\boldsymbol{x}\|_2 \qquad (95) \\ & = \left(2 + \frac{M_{\max}}{\sqrt{\gamma g(\lambda_{k+1})}}\right) \left\|\widetilde{\mathsf{P}}\boldsymbol{n}\right\|_2 + \left(M_{\max}\sqrt{\frac{g(\lambda_k)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_k)}\right) \|\boldsymbol{x}\|_2 \\ & + \epsilon \left(2M_{\max} + M_{\max}\sqrt{\frac{g(\lambda_n)}{g(\lambda_{k+1})}} + \sqrt{\gamma g(\lambda_n)}\right) \|\boldsymbol{x}\|_2 \,. \qquad (96) \end{aligned}$$

We only rearranged the term in the last step. This proves (44) and terminates the proof. □

References

[1] Y. Boykov and M. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," in *Proc. Int. Conf. Computer Vision*, 2001.

[2] L. Grady, "Random walks for image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1768–1783, 2006.

[3] X. Chen, D. Zou, Q. Zhao, and P. Tan, "Manifold preserving edit propagation," *ACM Transactions on Graphics*, vol. 31, no. 6, pp. 132–139, 2012.

[4] X. Chen, D. Zou, S. Zhou, Q. Zhao, and P. Tan, "Image matting with local and nonlocal smooth priors," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2013.

[5] L. Xu, Q. Yan, and J. Jia, "A sparse control model for image and video editing," *ACM Transactions on Graphics*, vol. 32, no. 6, p. 197, 2013.

[6] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 689–694, 2004.

[7] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski, "Interactive local adjustment of tonal values," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 646–653, 2006.

[8] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *Signal Processing Magazine, IEEE*, vol. 30, no. 3, pp. 83–98, May 2013.

[9] I. Pesenson, "Sampling in Paley-Wiener spaces on combinatorial graphs," *Transactions of the American Mathematical Society*, vol. 360, no. 10, pp. 5603–5627, 2008.

[10] I. Z. Pesenson and M. Z. Pesenson, "Sampling, Filtering and Sparse Approximations on Combinatorial Graphs," *J. Fourier Anal. Appl.*, vol. 16, no. 6, pp. 921–942, 2010.

[11] A. Anis, A. Gadde, and A. Ortega, "Towards a sampling theorem for signals on arbitrary graphs," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 3864–3868.

[12] A. Anis, A. E. Gamal, S. Avestimehr, and A. Ortega, "Asymptotic justification of bandlimited interpolation of graph signals for semi-supervised learning," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 5461–5465.

[13] S. Chen, A. Sandryhaila, and J. Kovacevic, "Sampling theory for graph signals," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, April 2015, pp. 3392–3396.

[14] S. Chen, R. Varma, A. Sandryhaila, and J. Kovacevic, "Discrete signal processing on graphs: Sampling theory," *Signal Processing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[15] A. Anis, A. Gadde, and A. Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. PP, no. 99, pp. 1–1, 2016.

[16] S. Chen, R. Varma, A. Singh, and J. Kovacevi, "Signal recovery on graphs: Random versus experimentally designed sampling," in *Sampling Theory and Applications (SampTA), 2015 International Conference on*, May 2015, pp. 337–341.

[17] G. Puy, N. Tremblay, R. Gribonval, and P. Vandergheynst, "Random sampling of bandlimited signals on graphs," *Appl. Comput. Harmon. Anal., in press*, 2016.

[18] N. Chauffert, P. Ciuciu, and P. Weiss, "Variable density compressed sensing in mri. theoretical vs heuristic sampling strategies," in *Proc. IEEE Int. Symp. on Biomedical Imaging*, 2013, pp. 298 – 301.

[19] B. Adcock, A. C. Hansen, C. Poon, and B. Roman, "Breaking the coherence barrier: a new theory for compressed sensing," *arXiv:1302.0561*, 2013.

[20] C. Boyer, J. Bigot, and P. Weiss, "Compressed sensing with structured sparsity and structured acquisition," *arXiv:1505.01619*, 2015.

[21] J. Bigot, C. Boyer, and P. Weiss, "An analysis of block sampling strategies in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 2125 – 2139, 2016.

[22] X. Chen, D. Zou, Q. Zhao, and P. Tan, "Manifold preserving edit propagation," in *ACM SIGGRAPH Asia*, vol. 31, no. 6, 2012.

[23] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, 2012.

[24] C. Korinke, T. C. Stratmann, T. Laue, and S. Boll, "Superselect: An interactive superpixel-based segmentation method for touch displays," in *Proc. ACM Int. Conf. Multimedia*, 2016.

[25] F. Chung, *Spectral graph theory*. Amer Mathematical Society, 1997, no. 92.

[26] G. Puy, P. Vandergheynst, and Y. Wiaux, "On variable density compressive sampling," *IEEE Signal Process. Lett.*, vol. 18, no. 10, pp. 595–598, 2011.

[27] F. Krahmer and R. Ward, "Stable and robust sampling strategies for compressive imaging," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 612–622, 2013.

[28] A. Borzi and V. Schulz, "Multigrid methods for pde optimization," *SIAM Rev.*, vol. 51, no. 2, pp. 361–395, 2009.

[29] T. Dreyer, B. Maar, and V. Schulz, "Multigrid optimization in applications," *J. Comput. Appl. Mathematics*, vol. 128, no. 1-2, pp. 67–84, 2000.

[30] N. Tremblay and P. Borgnat, "Subgraph-based filterbanks for graph signals," *IEEE Transactions on Signal Processing*, vol. PP, no. 99, pp. 1–1, 2016.

[31] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.
[32] N. Perraudin, J. Paratte, D. Shuman, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv:1408.5781*, 2014.
[33] E. D. Napoli, E. Polizzo, and Y. Saad, "Efficient estimation of eigenvalue counts in an interval," *arXiv:1308.4275*, 2013.
[34] H. Li, J. Cai, T. N. A. Nguyen, and J. Zheng, "A benchmark for semantic image segmentation," in *IEEE Int. Conf. on Multimedia and Expo*, 2013, pp. 1–6.
[35] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[36] N. Tremblay, G. Puy, R. Gribonval, and P. Vandergheynst, "Compressive spectral clustering," in *Machine Learning, Proceedings of the Thirty-third International Conference (ICML 2016), June 20-22, New York City, USA*, 2016.
[37] J. A. Tropp, "User-friendly tail bounds for sums of random matrices," *Found. Comput. Math.*, vol. 12, no. 4, pp. 389–434, 2012.