A random walk on image patches

Kye M. Taylor † and François G. Meyer ‡

Abstract. In this paper we address the problem of understanding the success of algorithms that organize patches according to graph-based metrics. Algorithms that analyze patches extracted from images or time series have led to state-of-the-art techniques for classification, denoising, and the study of nonlinear dynamics. The main contribution of this work is to provide a theoretical explanation for the above experimental observations. Our approach relies on a detailed analysis of the commute time metric on prototypical graph models that epitomize the geometry observed in general patch-graphs. We prove that a parametrization of the graph based on commute times shrinks the mutual distances between patches that correspond to rapid local changes in the signal, while the distances between patches that correspond to slow local changes expand. In effect, our results explain why the parametrization of the set of patches based on the eigenfunctions of the Laplacian can concentrate patches that correspond to rapid local changes, which would otherwise be scattered in the space of patches. While our results are based on a large sample analysis, numerical experimentations on synthetic and real data indicate that the results hold for datasets that are very small in practice.

Key words. image patches, diffusion maps, Laplacian eigenmaps, graph Laplacian, commute time

AMS subject classifications. 62H35, 05C12, 05C81, 05C90

1. Introduction.

Problem statement and motivation. In this paper we address the problem of understanding the success of algorithms that organize patches according to graph-based metrics. Patches are local portions, or snippets, of a signal or an image. The set of patches can be organized by constructing a graph that connects patches that are similar. Indeed, it is reasonably straightforward to measure the similarity between patches that are alike. The graph can then be used to extend the notion of similarity to patches that are very different. For instance, one can measure the distance between two visually different patches by computing the number of edges of the shortest path (geodesic) connecting them. In this work we explore a distance defined by the commute time associated with a random walk defined on the graph.

Algorithms that analyze patch data using graph-based metrics have led to state-of-the-art techniques for classification [42, 48], denoising [5, 7, 22, 25, 35, 44, 45], and studying dynamics [4, 26, 51]. The graph provides a new perspective from which to analyze the similarities between patches, and consequently, the local signal or image content they contain. For example, in [4], properties of the graph’s geometry, such as the distribution of clustering-coefficients and the average geodesic distance between two vertices, are used to separate chaos and noise, or different types of chaos. In [42, 44, 45, 48], the geometry of the graph is analyzed by studying a random walk on it. Specifically, the diffusion distance [15] (or spectral distance [34]),

\*This work was partially supported by National Science Foundation Grants DMS 0941476, ECS 0501578, and DOE award DE-SC0004096.

†Department of Applied Mathematics, 526 UCB University of Colorado, Boulder, CO 80309-0526 (taylorkm@colorado.edu).

‡Department of Electrical Engineering, 425 UCB, University of Colorado, Boulder, CO 80309-0425 (fmeyer@colorado.edu).
2 K. M. TAYLOR AND F. G. MEYER

and the commute time distance \[6, 40, 20, 38\] (which is equivalent to the resistance distance \[21\]) are two related graph metrics that are derived from the random walk, and that can be used to parametrize the graph’s geometry. These metrics can be used to efficiently organize patches in a manner that reveals the local behavior of the associated signal or image. In our previous work \[42, 48\], we have noticed that metrics based on a diffusion, or a random walk concentrate patches that contain rapid changes in the signal or image data. These patches contain changes associated with singularities (edges), rapid changes in frequency (textures, oscillations), or energetic transients contained in the underlying function. Furthermore, patches that contain only the smooth parts of the image are more spread out according to such graph metrics.

Outline of our approach and results. The main contribution of this work is to provide a theoretical explanation for the above experimental observations. Our approach relies on a detailed analysis of the commute time metric on prototypical graph models that epitomize the geometry observed in general patch-graphs. We assume that the set of patches is composed of two broad classes: patches within which the function varies smoothly and slowly, and patches where the function exhibits anomalies: singularities, very rapid change in local frequency, etc. We prove that a parametrization of the graph based on commute times shrinks the mutual distances between vertices that correspond to rapid local changes relative to the distances between vertices that correspond to slow local changes. In effect, our results explain why the parametrization of the set of patches based on the eigenfunctions of the Laplacian \[44, 45\] can concentrate anomalous patches, which would otherwise be scattered in the space of patches. This concentration phenomenon can then be exploited for further processing of the patches (e.g. denoising, classification, etc). While our results are based on a large sample analysis, numerical experimentations on synthetic and real data indicate that the results hold for datasets that are very small in practice.

Organization. This paper is organized as follows. In the next section, we describe the patch-based representation of a signal, and the associated patch-graph. We develop some intuition about the graph of patches by studying several examples in section 3. In section 4, we describe the embedding of the graph of patches based on commute time. The prototypical graph models that allow us to study the parametrization are defined in section 5. The main theoretical result about the embedding of the graph models are presented in section 5.3. Numerical experiments confirming our theoretical analysis are presented in section 6. We finish with a discussion in section 7.

2. Preliminaries and Notation. For simplicity and without loss of generality, we assume that the signal of interest is formed by a sequence of samples, \(x_n\) \(n=1\). Because we want to extract \(N = N' - (d - 1)\) patches from this sequence, we need \(d - 1\) extra samples at the end (hence the \(N'\) samples). We first define the notion of a patch.

Definition 2.1. We define a patch as a vector in \(\mathbb{R}^d\) formed by a subsequence of \(d\) contiguous samples extracted from the sequence \(\{x_n\}_{n=1}^{N'}\).

\[
x_n = \begin{bmatrix} x_n & x_{n+1} & \cdots & x_{n+(d-1)} \end{bmatrix}^T, \quad \text{for } n = 1, 2, \ldots, N.
\] (2.1)
As we collect all the patches, we form the patch-set in $\mathbb{R}^d$.

**Definition 2.2.** The patch-set is defined as the set of patches extracted from the sequence $\{x_n\}_{n=1}^{N'}$,

$$\text{patch-set} = \{x_n, n = 1, 2, \ldots, N\}.$$  \hspace{1cm} (2.2)

A main objective of this paper is to understand the organization of the patch-set and relate this organization to the presence of local changes in the signal or the image. We note that the concept of patch is related to the concept of time-delay embedding. Specifically, if the sequence comprises measurements of a dynamical system, then Taken’s embedding theorem \cite{41, 46} allows us to replace the unknown phase space of the dynamical system with a topologically equivalent phase space formed by the patch-set (2.2). While in this work we do not assume that the sequence $\{x_n\}$ is an observable of a dynamical system, we are nevertheless interested in a similar goal: the organization of patches in $\mathbb{R}^d$.

Throughout this paper, we think about a patch, $x_n$, in several different ways. Originally, $x_n$ is simply a snippet of the time series. Then, we think about $x_n$ as a point in $\mathbb{R}^d$. Later, we also regard $x_n$ as a vertex of a graph. Keeping these three perspectives in mind is critical to our approach and understanding.

In order to study the discrete structure formed by the patch-set (2.2), we connect patches together (using their nearest neighbors) and define a graph (or network) that we call the patch-graph.

**Definition 2.3.** The patch-graph, $\Gamma$, is a weighted graph defined as follows.

1. The vertices of $\Gamma$ are the patches $\{x_n, n = 1, \ldots, N\}$.
2. Each vertex $x_n$ is connected to its $\nu$ nearest neighbors using the metric

$$\rho(x_n, x_m) = \left\| \frac{x_n}{\|x_n\|} - \frac{x_m}{\|x_m\|} \right\|.$$ \hspace{1cm} (2.3)

3. The weight $w_{n,m}$ along the edge $\{x_n, x_m\}$ is given by

$$w_{n,m} = \begin{cases} e^{-\rho^2(x_n, x_m)/\sigma^2} & \text{if } x_n \text{ is connected to } x_m, \\ 0 & \text{otherwise.} \end{cases} \hspace{1cm} (2.4)$$

The edges of the patch-graph encode the similarities between its $N$ vertices. We work with the metric $\rho$ (defined in (2.3)) because it is not sensitive to changes in the local energy of the signal (measured by $\|x_n\|$). Indeed, the average local intensity in images taken with standard cameras may vary greatly from one region of the image to another. In contrast, the human visual system performs a renormalization akin to the one performed by $\rho$ \cite{39}. The metric $\rho$ allows us to detect changes in the signal’s local frequency content, or local smoothness. The parameter $\sigma$ controls the scaling of the similarity $\rho(x_n, x_m)$ between $x_n$ and $x_m$ when defining the edge weight $w_{n,m}$. In particular, $w_{n,m}$ will drop rapidly to zero as $\rho(x_n, x_m)$ becomes larger than $\sigma$. In some of the experiments in section 6, we choose $\sigma$ to be very large. This choice of $\sigma$ emphasizes the topology of the graph and promotes a very fast diffusion of the random walk through the patch-set. The other alternative: $\sigma \approx 0$ accentuates the difference between the patches, but is very sensitive to noise. An important remark about the way we
measure distances on the graph is in order here. We use $\rho$ to define the graph topology defined by the edges: which patch is connected to which patch. This is appropriate since we can compare patches that are similar using $\rho$ (e.g. two patches containing the same edge, but at different locations). On the other hand, as explained in section 4.2, we use the commute time to analyze the global geometry of the patch-graph. Indeed, the distance defined by $\rho$ becomes useless when we need to compare very different patches (e.g. a patch of a uniform region vs a patch that contains an edge). As explained in section 4.2, the global organization of the patches can be discovered by studying the speed at which a random walk propagates along the graph (via hitting times). Finally, we note that the weighted graph is fully characterized by its *weight matrix*.

**Definition 2.4.** The weight matrix $W$ is the $N \times N$ matrix with entries $W_{n,m} = w_{n,m}$. The degree matrix is the $N \times N$ diagonal matrix $D$ with entries $D_{n,n} = \sum_{l=1}^{N} w_{n,l}$.

**3. Warm up: A first look at the patch-set.** The goal of this section is to provide the reader with some intuition about the geometry of the patch-set and the associated patch-graph. This will help us motivate our graph models and the analysis of their geometry. At the end of the section, we provide a sketch of our plan of attack.

**3.1. Examples of signals and images.** We construct the patch-set associated with some examples of signals and images. Because it is not practical to visualize the patch-set in $\mathbb{R}^d$ when $d = 25$, we display the projection of the patch-set onto the three-dimensional space that captures the largest variance in the patch-set (computed using principal component analysis). Figure 1 displays three signals $\{x_n\}, n = 1, \ldots, N'$, with $N' = 2072$. Patches of size $d = 25$ samples are extracted around each time sample, which results in the maximum overlap between patches. Signal A is a chirp, signal B is a row of a high resolution version of the image Lenna (shown in Fig. 3-D), and signal C is a column of a larger image from which image I in figure 3 was extracted. Figure 2 displays six images: the first two images are standard test images, and the next four are images of outdoor scenes, and therefore qualify as natural images. We
Figure 2. D, E, F, G, H, and I: image of size 128 × 128, 128 × 128, 240 × 240, 256 × 256, 400 × 400, and 400 × 400 pixels, respectively. The blue square in figure I represents the sub-image extracted for analysis in Figure 3.

Figure 3. The color of the pixel at the center of each patch encodes the local variance of the image intensity.
extract patches of size $5 \times 5$ from a grid that is coarser than the initial image sampling grid. The patch centers are located on a grid of resolution $3 \times 3$ pixels, for images D and E, $5 \times 5$ pixels, for images F and G, and $8 \times 8$ pixels for images H and I. This results in patch-sets of sizes $42 \times 42$ for images D and E, $48 \times 48$ for image F, $50 \times 50$ for image G, and $50 \times 50$ for images H and I. In figure 3, we quantify the local regularity of the signals and the images by computing the variance over each patch. We then color the signals and images according to the magnitude of the local variance: orange when the variance is large, and blue when the variance is small. Throughout these experiments, the local variance is deemed large if it is greater than the median of the distribution of the variances over all patches associated with a particular signal. It is small, otherwise. We notice that high variance patches correspond to regions that contain texture, or very abrupt edges in the images D-I in figure 3.

3.2. Projections of the patch-sets. Figure 4 shows the projections of each of the nine patch-sets onto the three-dimensional space that captures the largest variance in the patch-set. Distances in Figure 4 correspond to the normalized distance $\rho$. We observe that patches with high variance (orange) appear to be scattered all over $\mathbb{R}^d$. These patches correspond to regions where the image intensity varies rapidly (e.g. texture in the grass or the trees in image I, see figure 3). Patches with low variance (blue), which correspond to regions where the image intensity is smooth and varies very slowly, tend to be concentrated along one-dimensional curves (for time series) and two-dimensional surfaces (for images). These visual observations can be confirmed when computing the actual mutual distances between patches (data not shown).

The organization of the patches in the patch-set can be explained using simple arguments. We first observe that if a patch is in a region of the image where the intensity $x(n, m)$ at pixel $(n, m)$ is smooth, then the distance (in patch space) to another patch at location $(n', m')$ is of the order

$$\rho(x_{n,m}, x_{n',m'}) \approx d \sqrt{(n-n')^2 + (m-m')^2} \| \nabla x(n, m) \| / \| x(n, m) \|. \quad (3.1)$$

If the gradient $\nabla x(n, m)$ is sufficiently small, then all the nearest neighbors of the patch $x_{n,m}$ will be in the close spatial neighborhood of the pixel $(n, m)$. This observation implies that patches in smooth regions of the image are organized along two-dimensional surfaces in $\mathbb{R}^d$.

In order to understand the distribution of the high variance (orange) patches, we can expand each texture patch $x_{n,m}$ into a local Fourier basis (while not all texture is periodic, a local Fourier basis provides an efficient representation for periodic texture; see [36] for a similar model of periodic texture). Consider two patches $x_{n,m}$ and $x_{n',m'}$ extracted from a fast oscillatory texture, such as the grass in images F and I in figure 2. For the sake of the argument, let us consider the effect of the dominant frequency of the texture.

Now, assume that the two pixels $(n, m)$ and $(n', m')$ are close in the image, then as soon as the distance $\sqrt{(n-n')^2 + (m-m')^2}$ becomes of the order of one-fourth of the wavelength of the periodic texture, the patches are misaligned, and the norm of their difference $\| x_{n,m} - x_{n',m'} \|$ will be large. As a result, the nearest neighbor $x_{n',m'}$ (in patch space) of a high frequency texture patch $x_{n,m}$ is more likely to be faraway (see figure 4: images F to I), and $\sqrt{(n-n')^2 + (m-m')^2}$ will be large, i.e. the neighbor $x_{n',m'}$ will not be found in the spatial (in the image domain) proximity of $x_{n,m}$. We note that similar experimental observations
have been reported by other researchers working on the statistics of image patches. Indeed, as demonstrated by several authors (e.g. [10, 23, 33, 43, 53]), patches in images can be classified into two classes: patches extracted from smooth regions, and patches extracted from regions that contain texture, fast oscillation, large gradient, etc. These authors also observe that patches selected from regions where the intensity is slowly varying are at a small distance (in patch space) of one another, and are also at a close spatial distance (in pixels). Neighbors of patches that contain texture or a large gradient are not found in the proximate neighbors (in pixel space), and are at a large distance in patch space (the authors in [53] estimate that the distance to the nearest neighbor of a patch grows exponentially with the gradient within that patch). As a result, proximity between texture patches is not dictated by spatial proximity, and large variance patches appear in effect to be scattered randomly across patch space.

3.3. From the patch-set to the patch-graph: the weight matrix $W$. Having gained some understanding about the organization of the patch-set, we now move to the structure of the patch-graph and its weight matrix $W$. Figure 5 displays the weight matrices built from the patch-sets that correspond to the time series A-C (first row), the images D-F (second row), and the images G-I (third row). Each entry in the matrix is color-coded as follows: $w_{n,m}$ is orange if $x_n$ and $x_m$ are two patches with high variance, $w_{n,m}$ is blue if $x_n$ and $x_m$ are two patches with small variance, and $w_{n,m}$ is green if $x_n$ is a patch with small variance and $x_m$
Figure 5. The weight matrices $W$ associated with the time series A-C and the images D-I displayed as images. $w_{n,m}$ is orange if $x_n$ and $x_m$ are two patches with high variance, $w_{n,m}$ is blue if $x_n$ and $x_m$ are two patches with small variance, and $w_{n,m}$ is green if $x_n$ is a patch with small variance and $x_m$ is a patch with high variance (or vice versa). The indices $n, m$ grow linearly with time for the signals A-C. For the images, we scan each image column by column, from left to right.

is a patch with high variance (or vice versa). We recall that during these experiments, the local variance is considered to be large if it is greater than the median of the distribution of the variances over all patches extracted from a given signal or image; it is small otherwise.

Note that when processing time series A-C, the columns and the rows of $W$ can be identified with temporally-ordered time-samples. Therefore, a cluster of large entries along the diagonal in the weight matrix corresponds to patches that are close in time and also close in $\mathbb{R}^d$. For instance, consider the time series C (extracted from a column of a larger version of image I, see figure 1) and its associated weight matrix (see figure 5). The top-left part of the matrix is composed of blue patches extracted from the sky, where the intensity varies smoothly. These large entries in the weight matrix, which correspond to neighbors in the patch-set, are in close spatial proximity: very close to the diagonal. On the other hand, the bottom right corner of the matrix contains patches extracted from the grass, where the
The weight matrices of the images E-I are harder to decipher. This is due to the raster-scan ordering of the patches that destroys the two-dimensional spatial correlation. Coherent blocks of blue entries along the diagonal are indicative of smooth regions of the image (e.g. the mirror’s smooth contour in image E created a block which spans roughly one fifth of the number of columns in figure 5-E). To facilitate the interpretation of the weight matrices, we applied to each matrix a (different) permutation $\tau$ of the patch indices in order to reorder patches according to their variance: $\text{Var}(x_{\tau(1)}) \leq \text{Var}(x_{\tau(2)}) \leq \cdots \leq \text{Var}(x_{\tau(N)})$. We display in figure 6 the weight matrices after permuting the rows and the columns using $\tau$. A clear pattern emerges: the low variance patches (blue entries) have neighbors that are low variance patches. Few low variance patches have neighbors among the high variance patches (few

Figure 6. Weight matrices $W$ shown in figure 5 after a permutation $\tau$ of the rows and the columns that guarantees that the variance of each patch $x_{\tau(n)}$ increases with $n$. The color coding scheme remains the same as in figure 5.
green entries in $W$). Furthermore, blue entries are clustered along the diagonal (e.g. images C, H and I), whereas orange entries are scattered in the bottom right quadrant. We also note that neighbors of high variance patches are high variance patches and are randomly distributed throughout all the high variance patches. The pattern that we observe in the weight matrices of the patch-set of these natural images can be explained by the analysis of the mutual distances that we conducted in the previous paragraph.

We complete this section with a quantitative analysis of the distribution of the edges in the patch-graph. Figure 7 displays the proportion of edges within and between the subsets of low and high variance patches. We note that very few edges exist between the low variance and high variance patches (and vice versa). This observation suggest that the patch-graph is composed of two subgraphs that are connected with a small number of edges.

3.4. Summary of the experiments and our plan of attack. The experiments in sections 3.1, 3.2, and 3.3 highlight the fact that patches with large variance, which are extracted from texture and other regions with large gradient, are scattered all over the patch-set, making their detection and identification extremely difficult (see Figures 4-C and 4-F-I). In contrast, patches with low variance, which come from regions of the image where the intensity varies smoothly, appear to cluster along low dimensional surfaces. Because the high variance patches are usually the most interesting ones, we need to find a new parametrization of the patch-set that concentrates these patches and separates them from the smooth baseline part of the image. The structure of $W$ in these “rough regions” suggests that neighbors of high variance patches are “randomly scattered” across the patch-set, giving rise to weight matrices with entries that are uniformly spread among the high variance patches (see figure 6-C, H and I). In order to study the structure of such matrices, we will quantify the speed at which a random walk on the patch-set would reach all patches in these rough regions of $W$, and suggest that we should consider studying the hitting times associated with a random walk on the patch-graph. In the next section we formalize this concept and propose a parametrization of the patch-set in terms of commute time. A theoretical analysis of this approach is provided in section 5.
4. Parametrizing the patch-graph.

4.1. The fast and slow patches. We first introduce the concept of fast and slow patches. We have noticed that patches that contain anomalies (discontinuities, edges, texture, etc.) in the original signal lead to regions of the matrix $W$ where the nonzero entries are scattered all around. We call such patches fast patches because, as we will see in the following, a random walk will diffuse extremely fast in such regions of the patch-graph. Conversely, smooth regions of the signal lead to slow patches that are associated with entries in $W$ that are concentrated around the diagonal. A random walk on the patch-graph that is initialized at a slow patch will diffuse very slowly.

4.2. A better metric on the graph: the commute time. As explained previously, we propose to replace the Euclidean distance, which leads to the scattering of the fast patches seen in Figure 4 by a notion of distance that quantifies the speed at which a random walk diffuses on the patch-graph. We propose to use the commute time. Parametrizing the graph using its commute time distance is closely related to parametrizing the graph using its diffusion distance [15, 20, 24, 40, 38] (see Section 4.2.2). Although the works [7, 44, 45] do not explicitly embed vertices of the patch-graph based on the diffusion distance, they also study a random walk on the patch-graph, and define the diffusion distance in terms of this walk. In these studies, noise is removed by evolving the diffusion process for a small time. A detailed comparison of our approach with the seminal work of [44] is provided in section 7.3. We note that the notion of first-passage time associated with a diffusion (which is equivalent to the hitting time associated with a random walk) has been used extensively to characterize the geometry of complex networks, and random media (e.g. [3] and references therein). It is therefore natural to analyze the patch-set with this distance.

4.2.1. A random walk on the patch-graph. In order to define the commute time between two vertices, we first need to define a random walk on the graph. In our problem, the random walk does not correspond to a physical process, but will lead to a notion of global proximity between patches. We consider a first-order homogeneous Markov process, $Z_k$, defined on the vertices of the patch-graph, $\Gamma$, that evolves with the transition probability matrix $P$ given by

$$P_{n,m} = \text{Prob}(Z_{k+1} = x_m | Z_k = x_n) \triangleq \frac{w_{n,m}}{\sum_l w_{n,l}} = \frac{W_{n,m}}{D_{n,n}}.$$ (4.1)

Consider a slow patch $x_n$ extracted from a smooth part of the image. If the random walk starts at $x_n$, then it can only travel along the low-dimensional structure that corresponds to the patches extracted from the neighboring pixels. The existence of this narrow bottleneck is visible in the $W$ matrix. For instance, in figure 5-C: a random walk initialized within the fat diagonal of the bottom half of the upper left quadrant of $W$ is trapped in this region of the matrix, and can only travel along this fat diagonal. As a result, it will take many steps for the random walk to reach another slow patch $x_m$ if $|n - m|$ is large. This notion can be quantified by computing the average hitting-time, $h(x_n, x_m)$, which measures the expected minimum number of steps that it takes for the random walk, started at vertex $x_n$, to reach the vertex $x_m$ [6]

$$h(x_n, x_m) = \mathbb{E}_n \min\{j \geq 0 : Z_j = x_m\},$$
where the expectation $E$ is computed when the random walk is initialized at vertex $x_n$, i.e. when $Z_0 = x_n$. The commute time [6]: provides a symmetric version of $h$, and is defined by
\[ \kappa(x_n, x_m) = h(x_n, x_m) + h(x_m, x_n). \] (4.2)

### 4.2.2. Spectral representation of the commute time.

When the random walk is reversible and the graph is fully connected, the commute time can be expressed using the eigenvectors $\phi_1, \ldots, \phi_N$ of the symmetric matrix
\[ D^{-1/2}WD^{-1/2} = D^{1/2}PD^{-1/2}. \]
The corresponding eigenvalues can be labeled such that $-1 < \lambda_N \leq \ldots \leq \lambda_2 < \lambda_1 = 1$. Each eigenvector $\phi_k$ is a vector with $N$ components, one for each vertex of the graph. Hence, we write
\[ \phi_k = [\phi_k(x_1) \ \phi_k(x_2) \ \ldots \ \phi_k(x_N)]^T, \]
to emphasize the fact that we consider $\phi_k$ to be a function sampled on the vertices of $\Gamma$. The commute time can be expressed as
\[ \kappa(x_n, x_m) = \sum_{k=2}^{N} \frac{1}{1 - \lambda_k} \left( \frac{\phi_k(x_n)}{\sqrt{\pi_n}} - \frac{\phi_k(x_m)}{\sqrt{\pi_m}} \right)^2, \] (4.3)
where $\pi_n = \sum_{m=1}^{N} w_{n,m} / \sum_{j,l=1}^{N} w_{j,l}$ is the stationary distribution associated with the transition probability matrix $P$ [28, 42].

### 4.2.3. The relationship to diffusion maps.

The diffusion distance [15] between vertices $x_m$ and $x_n$, $D_t(x_m, x_n)$, measures the distance between the transition probability distributions -- computed at time $t$ -- of two random walks initialized at $x_n$ and $x_m$, $\sum_{l=1}^{N} |[P_t]_{n,l} - [P_t]_{m,l}|^2$. The diffusion distance can also be decomposed in terms of the eigenvectors $\phi_k$ [15],
\[ D_t^2(x_m, x_n) = \frac{1}{V} \sum_{k=2}^{N} \lambda_k^{2t} \left( \frac{\phi_k(x_m)}{\sqrt{\pi_m}} - \frac{\phi_k(x_n)}{\sqrt{\pi_n}} \right)^2, \] (4.4)
where $V = \sum_{m',n'} w_{m',n'}$ is the volume of the graph. It follows that the commute time is a scaled sum of the squares of diffusion distances computed at all times,
\[ \kappa(x_m, x_n) = V \sum_{t=0}^{\infty} D_t^2(x_m, x_n). \] (4.5)
The significance of this equation is that the commute time includes the short term evolution ($t \approx 0$) as well as the asymptotic regime ($t \to \infty$) of the random walk. We will come back to this analysis in section 5.4.
4.3. Parametrizing the patch-graph. Equation (4.3) suggests the following embedding \( \Psi \) of the patch-graph \( \Gamma \) into \( \mathbb{R}^{N-1} \),

\[
\Psi: x_n \rightarrow \frac{1}{\sqrt{\pi n}} \begin{bmatrix} \phi_2(x_n) / \sqrt{1-\lambda_2} & \phi_3(x_n) / \sqrt{1-\lambda_3} & \ldots & \phi_N(x_n) / \sqrt{1-\lambda_N} \end{bmatrix}^T, \quad n = 1, 2, \ldots, N. \tag{4.6}
\]

If we agree to measure the distance on the graph \( \Gamma \) using the square root of the commute time, then the mutual Euclidean distance after embedding is equal to the original distance on the graph,

\[
\|\Psi(x_n) - \Psi(x_m)\| = \sqrt{\kappa(x_n, x_m)}. \tag{4.7}
\]

The result is a direct consequence of (4.6) and (4.3). Similar ideas were first proposed in [34] to embed manifolds and are the foundation of the parametrizations given in [2, 15]. Saerens and co-author [40] provide a similar derivation based on the pseudo-inverse of the Laplacian (see also [20, 38]). In practice, we need not use all the \( N - 1 \) coordinates in the embedding defined by (4.6). Indeed, since \( \lambda_N \leq \cdots \leq \lambda_2 < \lambda_1 \), we have that \( \frac{1}{\sqrt{1-\lambda_N}} \leq \cdots \leq \frac{1}{\sqrt{1-\lambda_3}} \leq \frac{1}{\sqrt{1-\lambda_2}} \), and therefore, if we can accept some approximation error, we can use only the first \( d' \) coordinates of \( \Psi \). As we will see in section 5.4, this dimension reduction further improves the separation between slow patches and fast patches. In the remainder of the paper we will work with the embedding of \( \Gamma \) into \( \mathbb{R}^{d'} \) defined by

\[
\Phi: x_n \rightarrow \frac{1}{\sqrt{\pi n}} \begin{bmatrix} \phi_2(x_n) / \sqrt{1-\lambda_2} & \phi_3(x_n) / \sqrt{1-\lambda_3} & \ldots & \phi_{d'+1}(x_n) / \sqrt{1-\lambda_{d'+1}} \end{bmatrix}^T. \tag{4.8}
\]

We note that we can always choose \( d' \) such that the embedding \( \Phi \) almost preserves the commute time,

\[
\|\Phi(x_n) - \Phi(x_m)\|^2 \approx \kappa(x_n, x_m). \tag{4.9}
\]

In fact, our experiments indicate that this approximation holds for small values of \( d' \).

4.4. Examples (revisited). Figure 8 displays the embedding of the patch-sets associated with the signals A-C and the images D-I using the map \( \Phi \) (4.8), where \( d' = 3 \). After parametrization, the slow patches are aligned along smooth low-dimensional manifolds (curve for signals and surfaces for images). The fast (high variance) patches have been concentrated and appear to form more coherent structures (see figure 8). In the next section, we will provide some theoretical explanation for this phenomenon.

5. A model for the patch-graph and the analysis of its embedding.

5.1. Our approach. The embedding of the patch-graph \( \Gamma \) defined by \( \Phi \), in (4.8), should lead to a representation of the patch-set in \( \mathbb{R}^{d'} \) where distances correspond to commute times measured along the graph before embedding. Our goal is to explain the concentration of the fast patches created by the embedding \( \Phi \) (see e.g. Figure 8). Our approach is based on a theoretical analysis of a graph model that epitomizes the characteristic features observed in patch-graphs composed of a mixture of fast and slow patches. This model is composed of two subgraphs: a subgraph of slow patches, which are extracted from the smooth regions of the signal, and a subgraph of fast patches, which are extracted from the regions of the signal that contain singularities, changes in frequency, or energetic transients. We confirm our
Figure 8. Scatterplot of the patch-set shown in Figure 4 after parametrizing using $\Phi$ in (4.8), with $d' = 3$. The fast patches (orange) are now concentrated and have been lumped together. The slow patches (blue) remain aligned along curves (for time-series) and surfaces (for images).

Theoretical analysis with numerical experimentations using synthetic signals in section 6, and we demonstrate that our conclusions are in fact applicable to a larger class of patch-graphs. The graph models are introduced in section 5.2. Our theoretical analysis of the embedding of the graph models is given in section 5.3. We evaluate the performance of the embedding $\Phi$ when $d'$ is small in section 5.4.

5.2. The prototypical graph models. We define the graph models in terms of the nonzero entries in the associated weight matrix $W$. Without loss of generality, we assume that the number of vertices $N$ is even.

The slow graph model. The large entries in a weight matrix $W$ of a patch-graph composed only of slow patches will have large entries when $|n - m|$ is small\(^1\): temporal/spatial proximity

---

\(^1\)We assume that the rows/columns of $W$ are ordered according to increasing index $n$ of the sequence $\{x_n\}$. This assumption does not affect the graph's parametrization nor our theoretical conclusions, but allows us to
implies proximity in patch-space (see e.g. Figure 5-A, top corner). We therefore define the slow graph model as follows.

**Definition 5.1.** The slow graph $S(N, L)$ is a weighted graph composed of $N$ vertices, $x_1, \ldots, x_N$. The weight on the edge $\{x_n, x_m\}$ is defined by

$$w_{n,m} = \begin{cases} w_S & \text{if } |n - m| \leq L, \\ 0 & \text{otherwise}, \end{cases} \quad \text{for } 1 \leq n, m \leq N \text{ and } 2L + 1 \leq N. \quad (5.1)$$

The weight $w_S$ is a positive real number that models the distance between two temporally adjacent patches. The parameter $L$ characterizes the thickness of the diagonal in $W$. The slow graph is fully connected and each vertex has at most $2L$ neighbors, not including self-connections (see Figure 9). Hence, we require that $2L + 1 \leq N$. Finally, note that the slow graph is distinct from a regular ring, since the first and last vertices are not connected. We do not consider a regular ring since it would imply that the underlying signal is periodic.

**The fast graph model.** We now consider the model for a patch-graph built from a patch-set comprising only fast patches. As demonstrated in section 3, most of the entries in $W$ appear to be scattered throughout the matrix: temporal/spatial proximity does not correlate with proximity in patch space. In fact, fast patches are all far away from one another. We therefore define the fast graph model as follows.

**Definition 5.2.** The fast graph $F(N, p)$ is a random weighted graph composed of $N$ vertices, $x_1, \ldots, x_N$. The weight on the edge $\{x_n, x_m\}$ is defined by

$$w_{n,m} = w_{m,n} = \begin{cases} w_F & \text{with probability } p, \\ 0 & \text{with probability } 1 - p \end{cases} \quad \text{if } 1 \leq n < m \leq N,$$

and

$$w_{n,m} = 1 \quad \text{if } n = m.$$

The weight $w_F$ is a positive real number that models the distance between two fast patches. The fast graph model is equivalent to a weighted version of the Erdős-Rényi graph model [19], except that $F(N, p)$ contains self-connections and has edge weights possibly less than one. The parameter $p$ controls the density of the edges; $p = 1$ corresponds to a fully connected graph (clique).

**The fused graph model.** The fused graph model exemplifies the patch-set associated with a signal, or an image, which exhibits regions of fast and slow changes. The fused graph combines a slow and a fast subgraph of equal size (see Figure 9).

**Definition 5.3.** The fused graph $\Gamma^*(N)$ is a weighted graph composed of a slow subgraph $S(N/2, L)$ and a fast subgraph $F(N/2, p)$. In addition, edges between $S(N/2, L)$ and $F(N/2, p)$ are created randomly and independently with probability $q$ and assigned the edge weight $w_c > 0$.

Edges between $S(N/2, L)$ and $F(N/2, p)$ simulate those edges that bridge fast and slow patches. These edges allow us to model patches that are extracted from regions of the image that combine edges/transients and smooth intensity. If $q$ is so small that no edges are created between the two subgraphs, then an edge is placed at random between the two subgraphs.

interpret the structure in $W$. 
Figure 9. The fused graph model $\Gamma^*(N)$ is composed of a slow graph $S(N/2, L)$ (blue) and a fast graph $F(N/2, p)$ (orange), connected by random edges (green).

Figure 10. The weight matrix $W$ of the fused graph model $\Gamma^*(256)$ is displayed as an image: $w_{n,m}$ is encoded as a grayscale value: from white ($w_{n,m} = 0$) to black ($w_{n,m} = 1$). The entries of $W$ associated with the slow graph appear in the upper-left quadrant of $W$. Entries associated with the fast graph appear in the lower right quadrant. Random edges between the fast graph and slow graph appear in the upper right and lower left quadrants.

The true patch-graph is always constructed using a $\nu$ nearest neighbor rule (see section 2): each patch is connected to $\nu$ other patches. In order to mimic a true patch-graph, we adjust the thickness $L$ of the slow subgraph to the density of the edge connection, $p$, in the fast subgraph, so that on average, each vertex in the fused graph is connected to approximately $2L$ vertices\(^2\). We know that the number of edges between distinct vertices in $F(N, p)$ is a binomial random variable with expectation $\frac{N(N-1)}{2}p$. Since the total number of edges between distinct vertices of $S(N, L)$ is equal to\(^3\)

$$\sum_{j=1}^{L} (N - j) = NL - \frac{L(L+1)}{2},$$

we choose

$$p = \frac{2L}{N-1} - \frac{L(L+1)}{N(N-1)}. \quad (5.3)$$

This choice of $p$ guarantees that the expected number of edges in $F(N, p)$ is equal to the number of edges in $S(N, L)$. Furthermore, provided that $L = o(\sqrt{N})$ (we will in fact choose

\(^2\)The first few and last vertices of the slow sub-graph have fewer than $2L$ connections (see 5.1).

\(^3\)This is equivalent to the number of entries along the first $L$ upper diagonals of the matrix $W$. 

later \( L \sim \log(N) \), a short computation shows that, for large values of \( N \), this choice of \( p \) also ensures that the expected degree of a vertex in \( F(N, p) \) is equal to the average degree of a vertex in \( S(N, L) \). Figure 10 shows the nonzero entries in the weight matrix associated with one realization of the fused graph model using parameters \( N = 256, L = \lceil 2 \log N \rceil = 12 \) and \( q = \frac{\log(N)}{N} \). Vertices \( x_n \) with \( n \leq 128 \) are only connected to other vertices \( x_m \) if \( |n - m| \leq L \). This connectivity mimics the spatial (temporal) connectivity present in the smooth parts of an image (signal).

### 5.3. The main result

Our goal is to understand the effect of the embedding \( \Phi \) defined by (4.8) on the fused graph. It turns out that studying the embedding of each individual subgraph (slow and fast) separately is much more tractable than considering the entire fused graph. To complement our theoretical study of the fast and the slow subgraphs, we provide numerical evidence in section 5.4 that indicates that our understanding of the embedding of the subgraphs can be used to analyze the embedding of the fused graph. In section 6, we confirm that our theoretical analysis can be applied to true patch-graphs. Instead of studying \( \Phi \) directly, we take advantage of the fact that the embedding \( \Phi \) almost preserves the commute time (see (4.9)). We can therefore understand the effect of the embedding on the distribution of mutual distances \( \| \Phi(x_n) - \Phi(x_m) \| \) within a subgraph by studying the distribution of the commute times \( \kappa(x_n, x_m) \) on that subgraph. While it would appear that it is a straightforward affair to compute the commute time on the slow graph, the computation becomes rapidly intractable. For this reason we provide lower and upper bounds for the average commute time on the slow and fast subgraphs, respectively. This is sufficient for our needs, since the two bounds rapidly separate, even for low values of \( N \).

Before stating the main Lemma, let us take a moment to compute some rough estimates of the commute times on the slow and fast graphs. To get some quick answers, we consider the simplest versions of the two graph models. When \( L = 1 \), the slow graph \( S(N, 1) \) is a path with self-connections. On a path of \( N \) vertices without self-connections, the commute time between vertex \( x_n \) and \( x_m \) is equal to \( 2(N - 1)|m - n| \) \([28]\). Therefore, the average commute time (computed over all pairs of vertices) on a path of length \( N \) is \( \mathcal{O}(N^2) \). While it would make sense that adding edges to a path should decrease the commute time, this is generally not true \([28]\). Nevertheless, the presence of edges that allow the random walk to move forward by a distance \( L \) at each time step lead us to conjecture that the average commute time on \( S(N, L) \) should be \( \mathcal{O}(N^2/L) \). In fact, as we will see in Lemma 5.6, the average commute time of the slow graph is \( \mathcal{O}(N^2/L^2) \). With regard to the fast graph, we can analyze the case where the density of edges \( p = 1 \). In this case, the fast graph \( F(N, 1) \) is a complete graph, or clique, and every vertex is connected to every other vertex. In a complete graph, the average commute time is \( \mathcal{O}(N) \). Since the fast graph can be regarded as a complete graph whose edges have been removed with probability \( 1 - p \), we expect the commute time to be slightly larger than \( \mathcal{O}(N) \), since removing edges restricts the random walker’s options to get from one vertex to another. Again, in agreement with our intuition, Lemma 5.6 asserts that in the fast graph, the commute time is \( \mathcal{O}(NL\log N/\log L) \).

We are now ready to state the main lemma. Our results will be stated in terms of the “average behavior” of the commute time on each graph, a concept that we need to define properly. In the case of the slow graph, which is deterministic, we consider the average com-
mute time computed over all pairs of vertices.

Definition 5.4. Let \( \kappa_S \) be the average commute time between vertices in the slow graph \( S(N, L) \)

\[
\kappa_S \triangleq \frac{2}{N(N-1)} \sum_{1 \leq m < n \leq N} \kappa(x_n, x_m). \tag{5.4}
\]

In the case of the fast graph, the “average behavior” of the commute time needs to be defined more carefully. Indeed, each fast graph is a realization of a stochastic process, and therefore we consider how the commute time behaves as \( N \) gets large.

Definition 5.5. Let \( \kappa_F \) be defined as the average commute time between vertices in the giant component of a realization of the fast graph \( F \) generated according to (5.2)

\[
\kappa_F \triangleq \frac{1}{N_G} \sum \kappa(x_n, x_m), \tag{5.5}
\]

where \( N_G \) is the number of distinct pairs of vertices in the giant component.

With probability approaching one as \( N \) gets large, there is only one giant component of the fast graph to consider [17]. Furthermore, as shown in appendix A, we choose \( L \) so that the fast graph is connected with probability approaching one as \( N \) gets large. Therefore, with increasingly high probability, our results apply to any two randomly chosen vertices in the fast graph realization itself.

Lemma 5.6. We have

\[
(N(2L + 1) - L(L + 1)) \frac{2(N + 1)}{3L^2(L + 1)} \leq \kappa_S. \tag{5.6}
\]

We also have

\[
\kappa_F \leq (N(2L + 1) - L(L + 1)) \left( \frac{\log N}{\log \left( 2LN^{-1} - \frac{L+1}{N-1} \right)} \right), \tag{5.7}
\]

with probability approaching 1.

Proof. The proofs are given in appendix B. \( \blacksquare \)

As described in appendix A, choosing \( L = c \log N \) for some constant \( c > 1 \) suffices to ensure that \( F(N, p) \) is connected with high probability. Consequently, the upper bound on \( \kappa_F \) is negligible relative to the lower bound on \( \kappa_S \), when \( N \) is large.

Corollary 5.7. Assume that \( L = c \log N \) for some constant \( c > 1 \). It follows that, as \( N \to \infty \), the lower bound on \( \kappa_S \) grows like \( \left( \frac{N}{\log N} \right)^2 \), and the upper bound on \( \kappa_F \) grows like \( \frac{N(\log N)^2}{\log \log N} \).

\(^4\)The giant component is a set of connected vertices whose size is a positive fraction of \( N \).
Furthermore, the lower bound on $\kappa_S$ grows faster than the upper bound on $\kappa_F$, and so with a probability that approaches one as $N$ gets large,

$$\frac{\kappa_F}{\kappa_S} \to 0.$$ 

Proof. Notice that $\kappa_S$ is bounded away from zero. Therefore the ratio $\kappa_F/\kappa_S$ is bounded below by zero and from above by a ratio of the bounds from Lemma 5.6. The ratio of bounds goes to zero, which follows from a simple, but lengthy, limit calculation. $lacksquare$

We can translate the corollary in terms of the mutual distance between vertices of the subgraphs after the embedding $\Phi$: $\Phi(\mathcal{F}(N, p))$ will be more concentrated than $\Phi(\mathcal{S}(N, L))$.

5.4. Spectral decomposition of commute times on the graph models. The results of section 5.3 apply to the exact commute times on the graph models. However, as mentioned in section 4.3, it is more practical to use a truncated version of the spectral expansion of the commute time, defined by Equation (4.3). We also noticed that the commute time encompasses the short term evolution ($t \approx 0$) as well as the asymptotic regime ($t \to \infty$) of the behavior of the random walk. Neglecting eigenvectors $\phi_k$ for large $k$ emphasizes the long term behavior of the random walk, and we expect that it should further increase the difference between the slow and fast graphs. In this section, we confirm experimentally that approximating the commute times by truncating the expansion (4.3) actually emphasizes the separation between the fast subgraph and the slow subgraph in the fused graph model. In all the numerical experiments in this section, unless otherwise stated, we fix $N = 1024$, $L = \lceil 2 \log(N) \rceil$, $p$ is chosen according to (5.3), $q = \log(N)/N$, and $w_S = w_F = w_c = 1$. In all experiments, we compute the eigenvalues $\lambda_k$ of the matrix $D^{-1/2}WD^{-1/2}$ associated with the fast graph, the slow graph, and the fused graph.

Slow and fast subgraphs: two different dynamics revealed by the spectral decomposition. We first provide a back-of-the-envelope computation of the spectrum of the slow and fast graphs. As we have noticed before, the slow graph model is a “fat” path. We know that the spectrum of a path without self-connections [12] is given by

$$\cos \left[ \pi (k-1)/(N-1) \right], \quad k = 1, 2, \ldots, N.$$ 

We expect therefore that the eigenvalues associated with the slow graph will decay slowly away from one for small $k$. Figure 11 (inset) displays the eigenvalues associated with the slow graph model. As expected, the spectrum is flat around $k = 0$ and exhibits the slowest decay of all the graph models. We use the similarity between the fast graph model and the Erdős-Renyi graph to predict the spectrum of the fast graph. Except for $\lambda_1 = 1$, all the other eigenvalues of an Erdős-Renyi graph asymptotically follow the Wigner semicircle distribution [13]. Our numerical experiments confirm this prediction: as shown in Figure 11-right, the eigenvalues of the fast graph appear to be distributed along a semicircle.

The decay of the spectrum has a direct influence on the dynamics of the random walk. Specifically, the spectral gap controls the mixing rate, which measures the expected number of time-steps that are necessary to reduce the distance between the probability distribution
after $t$ steps $P_{n,m}^{(t)}$ and the stationary distribution $\pi_m$ by a certain factor [49]. This concept is justified by the fact that the convergence of $P_{n,m}^{(t)}$ is exponential [17], and is given by

$$\max_{n,m} \left| \frac{P_{n,m}^{(t)}}{\pi_m} - 1 \right| \leq \frac{\lambda_{\text{max}}^t}{\pi_{\text{min}}} \quad t = 1, \ldots$$

(5.8)

where $\lambda_{\text{max}} = \max\{\lambda_2, |\lambda_N|\}$ (which is related to the spectral gap), and $\pi_{\text{min}}$ is the smallest entry of the stationary distribution. Since $\lambda_2$ is much larger in the slow graph than in the fast graph, we expect that convergence to the associated stationary distribution will take longer on the slow graph than on the fast graph.

The dynamics of the fused graph is enslaved by the slow subgraph. We now consider a random walk on the fused graph. If this random walk begins at $x_n$ in the fast subgraph of the fused graph, then after a small number of steps, $t_0$, the probability of finding the random walker at any other vertex $x_m$ in the fast subgraph is close to the stationary distribution, $P_{n,m}^{(t_0)} \approx \pi_m$. On the other hand, during the same amount of steps, a random walk initialized in the slow subgraph will only explore a small section of the slow subgraph, and consequently, the transition probabilities will still be similar to its initial values $P_{n,m}^{(t_0)} \approx P_{n,m}$. As a result, the restriction imposed by the geometry of the slow subgraph is expected to decrease the convergence rate of the transition probabilities on the fused graph. We confirm this analysis with experimental results. Figure 11 (inset) shows that for $k < 23$ the eigenvalues associated with the fused graph and the eigenvalues associated with the slow graph exhibit slow decay away from one, thereby decreasing the convergence rate given in (5.8). For $25 \leq k \leq 400$, the eigenvalues of the fused graph decay at a rate similar to that of the fast graph. Finally, for $k \geq 400$ the eigenvalues of the fused graph join those of the slow graph (see also the histogram in Figure 11-right). We have observed experimentally that these transitions in the behavior of the spectrum of the fused graph are not affected by varying the parameters $N$, $L$, and $q$. We conclude that the slow subgraph has the largest influence on the first few (small $k$) eigenvalues $\lambda_k$ of the fused graph.

The eigenvectors of the fused graph and their impact on the commute time. The transition exhibited in the spectrum of the fused graph can also be detected in the corresponding eigenvectors $\phi_k$. Figure 12 shows the eigenvectors $\{\phi_1, \phi_2, \phi_8, \phi_{16}, \phi_{32}\}$ corresponding to the three graph models. The first eigenvector $\phi_1$ has entries equal to the square root of the stationary distribution, $\phi_1(x_n) = \sqrt{\pi_n}$, and is not used in the expansion of the commute time (4.3). As expected, the random walk spends most of its time inside the slow subgraph of the fused graph, as indicated by the larger values of $\phi_1$ for the first (blue) $N/2$ vertices (see Figure 12-right). The eigenvectors $\{\phi_2, \phi_8, \phi_{16}\}$ of the fused graph exhibit large amplitude oscillations over the vertices belonging to the slow subgraph (first half – shown in blue – of the plots in Figure 12-right), which resemble those found in the eigenvectors associated with the slow graph (Figure 12-left). As $k$ increases, the eigenvectors $\phi_k$ of the fused graph become more and more similar to the eigenvectors of the fast graph.

The impact of the eigenvectors $\phi_k$ on the commute time on the fused graph can be analyzed by estimating the size of the terms

$$\frac{1}{1 - \lambda_k} \left( \frac{\phi_k(x_n)}{\sqrt{\pi_n}} - \frac{\phi_k(x_m)}{\sqrt{\pi_m}} \right)^2$$

(5.9)
Figure 11. The eigenvalues $\lambda_k$ of the matrix $D^{-1/2}W D^{-1/2}$ associated with the fused (green), slow (blue), and fast (orange) graphs. Left: $\lambda_k$ as a function of $k$; right: histogram of the $\lambda_k$.

Figure 12. The eigenvectors $\{\phi_1, \phi_2, \phi_8, \phi_{16}, \phi_{32}\}$ associated with the slow (left), fast (center), and fused (right) graphs. Right: the large amplitude of the eigenvectors $\phi_k$ on the first half of vertices (blue) belonging to the slow subgraph leads to a larger separation between the fast and slow subgraphs when truncating the commute time expansion.

in the spectral expansion (4.3) of the commute time $\kappa$. We claim that $\kappa(x_n, x_m)$ will be small if both vertices $x_n$ and $x_m$ are in the fast subgraph, and that $\kappa$ will be large if either vertex is in the slow subgraph.

We can first estimate the size of $\phi_k(x_n)/\sqrt{\pi_n} - \phi_k(x_m)/\sqrt{\pi_m}$. We observe that the eigenvectors $\phi_k$ for small values of $k$ have large amplitude oscillations on vertices belonging to the slow subgraph, but are relatively constant on the fast subgraph (see Figure 12-right). Therefore, for small values of $k$, each term (5.9) will be small when $x_n$ and $x_m$ both belong to the fast subgraph (we also have $\pi_n \approx \pi_m$ when two vertices belong to the same subgraph). Conversely, these terms will be large when either $x_n$ or $x_m$ belongs to the slow subgraph. While this analysis of the size of the terms (5.9) only holds for small values of $k$, it turns out that these are the terms that have the largest influence in the expansion of the commute time (4.3). Indeed, the spectrum of the fused graph decays slowly, and therefore the first few coef-
ficients \((1 - \lambda_k)^{-1}\) in the commute time expansion (4.3) are much larger than the remainders, and therefore the terms (5.9) for small values of \(k\) will provide the largest contribution in the expansion of the commute time.

We conclude that \(\kappa(x_n, x_m)\) is small when \(x_n\) and \(x_m\) belong to the fast subgraph, and \(\kappa(x_n, x_m)\) is large when either vertex is in the slow subgraph. Furthermore, we expect that this difference will be further magnified if we replace the exact expansion of \(\kappa\) in (4.3) by an approximation that only includes the first few values of \(k\).

The truncated spectral expansion of the commute time increases the contrast between the slow and fast subgraphs. We finally come to the heart of the section: the numerical computation of the average approximate commute time defined by

\[
\kappa' = \frac{2}{N(N-1)} \sum_{n<m} \|\Phi(x_n) - \Phi(x_m)\|^2.
\]  

Because of (4.7), we expect that \(\kappa'\) will be close to the true commute time \(\kappa\). We compute \(\kappa'\) for the three graphs: slow, fast and fused. We generated 25 realizations of the fast and fused graphs, and we estimated the expected commute time with the sample mean, given by \(\kappa'\) in (5.10).

Figure 13-A displays \(\kappa'_{\text{F}}/\kappa'_{\text{S}}\) as a function of the number of terms \(d'\) used in the embedding (4.8), for several values of the number of vertices \(N\), for the slow and fast graphs. Our theoretical analysis of \(\kappa'_{\text{F}}/\kappa'_{\text{S}}\), performed in Corollary 5.7, is only valid for large values of \(N\). Nevertheless, our numerical simulations indicate that for very low values of \(N\), \(\kappa'_{\text{F}}\) is already smaller than \(\kappa'_{\text{S}}\), since all ratios are below one (see Figure 13-A). Furthermore, we see that this ratio is even smaller for smaller values of \(d'\). We observe similar results when the commute times \(\kappa'_{\text{F}}\) and \(\kappa'_{\text{S}}\) are computed within the slow the fast subgraphs of the fused graph (see Figure 13-B). These results confirm that the embedding \(\Phi\) will further concentrate the vertices of the fast graph if \(d'\) is chosen to be much smaller than \(N\). We have observed experimentally that choosing \(d' \approx \log(N)\) leads to the smallest ratio of averages, not only on the graph models, but also on the general patch-graphs studied in section 6.

Because of the stochastic nature of the fast graph, we show in Fig. 14-A the histogram of the distribution of \(\kappa'\) for the fast graph (orange) and slow graph (blue). Note that only the histogram for the fast graph has error bars (the slow graph being deterministic). We note that the distribution of \(\kappa'\) for the fast graph is very concentrated and well separated from the distribution of \(\kappa'\) for the slow graph. We confirm visually (what was shown quantitatively in Fig. 13) that the average \(\kappa'\) for the fast graph is about two orders of magnitude smaller than the average \(\kappa'\) for the slow graph.

The enslaving of the fused graph by the slow graph is clearly shown in Figure 14-B, where the normalized histogram of \(\kappa'\) is shown for the three types of transition between the subgraphs of the fused graph \(\Gamma^*\): slow \(\rightarrow\) slow, fast \(\rightarrow\) fast, and slow \(\rightarrow\) fast. The histogram of the slow \(\rightarrow\) fast transition is very similar to the histogram of the slow \(\rightarrow\) slow transition, clearly indicating that once the random walk is trapped in the slow subgraph, the presence of the fast subgraph does not help the random walk escape from the slow graph. We also notice that the average of \(\kappa'\) for the fast \(\rightarrow\) fast transition is roughly two orders of magnitude smaller than the average of \(\kappa'\) for the slow \(\rightarrow\) slow, or slow \(\rightarrow\) fast transitions. In addition, the variance of each distribution is small enough to limit the overlap between the distributions.
Figure 13. $\kappa'_F/\kappa'_S$ as a function of the dimension $d'$ of the embedding $\Phi$, for several values of the number of vertices $N$. Left: slow $S$ and fast $F$ graphs separately; right: slow and fast subgraphs in the fused graph $\Gamma^*$.

Figure 14. Histogram of $\kappa'$. Left: slow graph $S$ and fast graph $F$. Right: fused graph $\Gamma^*$, $\kappa'$ for the three types of transition between the two subgraphs. Note the logarithmic scale on the horizontal axes.

Figure 15. $\kappa'$ as function of $N$. Left: slow graph $S$ and fast graph $F$. Right: fused graph $\Gamma^*$, $\kappa'$ for the three types of transition between the two subgraphs.

Figure 15 displays $\kappa'$ as a function of the number of vertices $N$, where $d' = \log N$. Again, this result confirms that the asymptotic analysis of the ratio $\kappa_F/\kappa_S$, performed in Corollary 5.7, actually holds for very small values of $N$. Indeed, whether the slow and fast graphs are considered separately (Figure 15-A), or are the components of the fused graph (Figure 15-B), the ratio $\kappa_F/\kappa_S \to 0$ (note the logarithmic scale). It is important to bear in mind that when
analyzing images, \( N \) is typically of the order of \( 10^6 \) and therefore our theoretical analysis will hold without any difficulty. Lastly, we again note in Figure 15-B that the transitions slow → fast in the fused graph have the same dynamics as the transition slow → slow.

5.5. Summary of the experiments. We have confirmed experimentally that embedding the fused graph using \( \Phi \) shrinks the mutual distance between vertices of the fast subgraph, effectively concentrating these vertices closer to one another. As a result, the embedding helps divide the fused graph into the slow and the fast subgraphs by concentrating the vertices of the fast subgraph away from the vertices of the slow subgraph. Our analysis of the embedding is based on the fact that \( \Phi \) approximately preserves the commute time measured on the fused graph. Furthermore, we have demonstrated that a truncated version of the commute time, \( \kappa' \), is even more conducive to identifying vertices of the fast subgraph of the fused graph.

The implication of these results is that the embedding of the true patch-graph \( \Gamma \) using \( \Phi \) will concentrate the high variance patches, which contain rapid changes in the signal, away from the smooth patches. This concentration of the fast patches happens for values of the embedding dimension \( d' \) that are of the order of \( \log(N) \): this choice of \( d' \) results in a low-dimensional embedding of the patch-graph. Because the fast patches are more clustered after embedding, their detection – for the purpose of detection of anomalies, classification, or segmentation – will become much easier. Finally, we note that our theoretical analysis can be extended to a more general context where patches are replaced by a vector of local features extracted from elements of a large dataset. The only requirement is that the graph of features exhibits a geometry similar to the fused graph \( \Gamma^* \).

6. Numerical experiments with synthetic signals. In this section we validate our theoretical results using synthetic signals. Each signal is the realization of a stochastic process with a prescribed autocorrelation function. We study two types of stochastic processes: one that generates signals that transition from low to high local frequency, and a second one that yields signals with varying local smoothness. We argue that these signals embody the types of local changes that are of fundamental importance in many areas of image processing. For both classes of signals, we embed the patch-sets using \( \Phi \) in (4.8). We study the property of the embedding by quantifying the average commute time \( \kappa' \), defined in (5.10) between fast and slow patches, and we compare the numerical results with the theoretical predictions given in section 5.

6.1. The signals. We consider two types of models: a time-frequency signal model and a local regularity signal model. Each model is characterized by an autocorrelation function. The autocorrelation function can be modified using a parameter that controls the local frequency, or the local regularity of the signal. We partition the interval [0,1] into subintervals over which the autocorrelation parameter is kept constant. The parameter alternates between two different values creating subintervals of alternating local frequency, or alternating local regularity. The number of alternations is chosen randomly according to a homogeneous Poisson process with intensity \( \mu \): there are on the average \( \mu + 1 \) subintervals. A simpler version of this model has been used in [14] to mimic the presence of edges in images. Unlike the model used in [14], we adjust the signal defined on each subinterval so that the result is continuous on [0,1]. In all experiments that we report here we use \( \mu = 3 \). The autocorrelation function
Figure 16. A realization of the time-frequency model. The low frequency portion ($\beta_S = 8$) is shown in blue; the high frequency portion ($\beta_F = 4096$) is shown in orange. There are four subintervals ($\mu = 3$).

Figure 17. A realization of the local regularity model. The smooth portion ($H_S = 0.9$) is shown in blue; the irregular portion ($H_F = 0.3$) is shown in orange. There are four subintervals ($\mu = 3$).

associated with the time-frequency signal model is given by

$$E(x(t)x(t + \tau)) = 2 \left(1 + \cos \left(\frac{2\pi \tau}{2}\right)\right)^\beta - 1,$$

(6.1)

where $\tau \in [0, 1)$, $\beta \geq 0$. As the autocorrelation parameter $\beta$ increases, the range of frequencies present in the signal also increases. Figure 16 displays a realization of this model where the signal’s covariance parameter alternates four times between $\beta_S = 8$, and $\beta_F = 4096$ (details about the generation of the signals can be found in Appendix C). The autocorrelation function associated with the local regularity signal model is equal to that of fractional Brownian motion, given by

$$E(x(\tau_1)x(\tau_2)) = \frac{1}{2}\left(|\tau_1|^{2H} + |\tau_2|^{2H} - |\tau_2 - \tau_1|^{2H}\right),$$

(6.2)

where $H$ is the Hurst parameter. As $H$ decreases, the local regularity decreases. A realization of this model is shown in Figure 17 where the signal’s covariance parameter alternates four times between $H_S = 0.9$ and $H_F = 0.3$. We use the method described in [1] to generate the fractional Brownian motion.

6.2. Embedding the patch-graph. For each realization of a specific signal model, we construct a patch-set of $N = 1024$ maximally overlapping patches. The patch size is given by $d = 32$ for the time-frequency model, and $d = 16$ for the local regularity model. We compute the embedding $\Phi$ (4.8) and keep $d'$ eigenvectors $\phi_k$. The dimension $d'$ of the embedding used to compute $\kappa'$ was chosen so that

$$(1 - \lambda_k)^{-1} < 0.1(1 - \lambda_2)^{-1}, \quad \text{for all } k > d' + 1.$$  

(6.3)

Figure 18 shows the patch-set associated with the realization of the time-frequency signal displayed in Figure 16 before (left) and after (right) embedding. The scatterplot before
embedding is computed using the first three principal components. Figure 19 shows the patch-set associated with the realization of the local regularity signal displayed in Figure 17 before (left) and after (right) embedding. The fast patches of the time-frequency signal are the orange patches extracted from the high frequency segments. The slow patches are the blue patches extracted from the low frequency sections. Similarly, the fast patches of the local regularity signal are the orange patches extracted from the irregular segments, and the slow patches are the blue patches extracted from the smooth sections. For both signals, the fast patches are scattered across the space before embedding. After embedding, the fast patches are aligned along smooth structures (curves and surfaces). This visual impression is confirmed by computing the mutual distance between patches after embedding, \( \| \Phi(x_n) - \Phi(x_m) \| \). In principle, we should report the value of the Lipschitz ratio \( \| \Phi(x_n) - \Phi(x_m) \| / \| x_n - x_m \| \) to quantify the contraction experienced through the mapping \( \Phi \). However, we have noticed that because the mutual distances \( \| x_n - x_m \| \) between fast patches is always large (as explained in section 3), the Lipschitz ratio ends up being always small for fast patches. Therefore studying the size of the Lipschitz ratio associated with \( \Phi \) does not reveal whether the map concentrates the fast patches or not, but only indicates that the sampling of the fast patches (in the patch-set) is coarse. For this reason we prefer to study how \( \| \Phi(x_n) - \Phi(x_m) \| \) varies for pairs of slow and fast patches. Based on our theoretical analysis, we expect that after the embedding the mutual distance between fast patches will become much shorter than the mutual distance between slow patches.

We point out that the eigenvectors \( \phi_k \) used in the embedding \( \Phi \) (4.8) are designed to have,
on average, small gradients (as measured along edges of the graph). Indeed, these eigenvectors are also the eigenvectors of the graph Laplacian \([12]\), and therefore minimize a Rayleigh ratio that quantifies the average norm of the gradient of \(\phi_k\). Thus, if we further restricted our computation of the commute times inside each subset of fast and slow patches to only those pair of neighboring patches that are connected by an edge in the graph, we would expect to see smaller values and little dependence on whether or not the patch was fast or slow. However, since our theoretical analysis of section 5 is based on the average commute time between all vertices belonging to the fast or slow graph models, we choose to compute the commute times between all patches, not just between neighboring patches that are connected with an edge.

For each signal model, we compute the square root of the average approximate commute time

\[
\sqrt{\kappa'} = \sqrt{\frac{2}{N(N-1)} \sum_{n<m} \|\Phi(x_n) - \Phi(x_m)\|^2},
\]

for pairs of patches \(x_n, x_m\) that are either both fast, or both slow patches. We study how \(\kappa'\) varies as a function of the autocorrelation parameter that controls how irregular the fast patches are. \(\kappa'\) was computed using ten realizations of each signal model. The slow patches were generated using \(\beta_S = 8\) and \(H_S = 0.9\). As before, we used \(N = 1024\) and \(d = 32\) for the time-frequency model and \(d = 16\) for the local regularity model. We observed that the overall shapes of the curves in Figure 20 is invariant under variation of the parameters (as long as the ratio of the patch length to the average subinterval length remains less than 10%).

Figure 20 shows \(\sqrt{\kappa'}\) as a function of the frequency parameter (left), and smoothness parameter (right). We note that as the signal exhibits more rapid, local changes (increasing \(\beta_F\), or decreasing \(H_F\)), the associated fast patches are increasingly concentrated (smaller \(\|\Phi(x_n) - \Phi(x_m)\|\)) through the parametrization. These experiments confirm that the theoretical analysis can be applied to the true patch-set constructed from realistic signals.

In order to further validate our model, we compute in figure 21 the proportion of edges within and between the slow and fast patches. We note that very few edges exist between the slow and fast patches (green curve). This separation between the two subgraphs is even more exacerbated as the fast patches of the local regularity model become rougher and rougher.

7. Discussion. Using realistic graph models, probabilistic arguments, and the connection between the commute time of random walks on graphs and the embedding \((4.8)\), we provided a theoretical explanation for the success of methods that analyze and process images based on graphs of patches. Our results establish that the embedding of the patch-graph of an image based on the commute time between vertices of the graph reveals the presence of patches containing rapid changes in the underlying signal or image by concentrating these patches close to one another while leaving the patches extracted from the slowly changing portions of the signal organized along low-dimensional structures.

7.1. Parameter selection.

7.1.1. Choosing the patch size. In this work we are interested in the local behavior of the image, and therefore \(d\) should remain of the order of what we consider to be the local scale. We also note that as \(d\) becomes large, the number of available patches \((N/d)\) becomes smaller, making the estimation of the geometry of the patch-set more difficult, since patches
now live in high-dimension. Another consequence of the “curse of dimensionality” is that the distance between patches becomes less informative for large values of $d$. If the original signal is oversampled with respect to the true physical processes at stake, then one can coarsen the sampling of the patch-set in the image domain. In practice, it would be more advisable to coarsen the underlying continuous patch-set, which is a nontrivial question.

7.1.2. Choosing edge weights. In general, two principles guide the choice of edge weights in the patch-graph. On the one hand, patches that are very close should be connected with a large weight (short distance), while patches that are faraway should have a very small weight along their mutual edge. This principle is equivalent to the idea of only trusting local distances in $\mathbb{R}^d$. Such a requirement is intuitively reasonable if we assume that the patch-set represents a discretization of a nonlinear manifold in $\mathbb{R}^d$. In this situation, we know that when the points on the manifold are very close to one another, the geodesic distance is well approximated by the Euclidean distance. Conversely, because of the presence of curvature, the Euclidean distance is a poor approximation to the geodesic distance on the manifold when points are far apart. Because the only information available to us is the Euclidean distance between

---

**Figure 20.** $\sqrt{\kappa}$ for slow (blue) and fast patches (orange) for the time-frequency model (left) and the local regularity model (right) as a function of the “roughness” of the fast patches. The slow patches were generated using $\beta_S = 8$ (left) and $H_S = 0.9$ (right).

**Figure 21.** Density of transition within the fast subgraph (orange), within the slow subgraph (blue), or from one subgraph to the other (green).
patches, we should not trust large Euclidean distances.

On the other hand, as observed in Section 3, the fast patches, which contain rapid changes, are all very far apart (large $\rho^2(x_n, x_m)$). Therefore the probability that the random walk escapes the fast patch $x_n$ and jumps to a different patch $x_m$, which is given by

$$w_{n,m} = \frac{e^{-\rho^2(x_n, x_m)/\sigma^2}}{\sum_l w_{n,l}},$$

is always much smaller than the probability of staying at $x_n$, which is given by $(\sum_l w_{n,l})^{-1}$.

In order to avoid that the random walk be trapped at each node $x_n$, we “saturate” the distance function by choosing $\sigma$ to be very large. In this case, for all the nearest neighbors $x_m$ of $x_n$, we have $w_{n,m} \approx 1$, and the transition probability is the same for all the neighbors, $P_{n,m} \approx 1/\nu$. This choice of $\sigma$ promotes a very fast diffusion of the random walk locally. We note that choosing a large $\sigma$ may be avoided if self-connections are not enforced (i.e. $w_{n,n} = 0$). However, self-connections are a necessary technical requirement to prove that the Markov process is aperiodic, which is required to prove the equality (4.4) [15].

We note that choosing $\sigma$ to be very large does not entirely obliterate the information provided by the mutual distance between patches, measured when patches are projected on the sphere with $\rho(x_n, x_m)$, (2.3). Indeed, $\rho(x_n, x_m)$ is used to select the nearest neighbors of each patch, and therefore allows us to define a notion of a local neighborhood around each patch. Choosing $\sigma$ to be very large forces a very fast diffusion within this neighborhood, irrespective of the actual distances $\rho(x_n, x_m)$. Alternatively, we could consider choosing $\sigma$ to vary adaptively from one neighborhood to another. The parameter $\sigma$ could be small when patches are extremely close to one another, while $\sigma$ could be large when the patches are at a large mutual distance of one another. This notion is the foundation of the self-tuning weight matrix, which adjusts its weights based on a point’s local neighborhood [31].

### 7.2. Extensions and generalizations

In general, the patch-set of an image consists of more than two homogeneous subsets. For example, one could partition an image patch-set into uniform patches, edge patches, and texture patches. Our experience [47] with a generalization of the time-frequency signal model (section 6) indicates that we can still separate the patches when the signal is composed of up to four different local behaviors (specified by four different values of the parameter in the autocorrelation function). Another extension of this work involves the embedding of a patch-set constructed from a library of images. Recent studies [27] indicate that high-contrast patches extracted from optical images organize themselves around 2-dimensional smooth sub-manifold [8]. This idea has also been exploited to construct dictionaries that lead to very sparse representations of images (e.g. [18], and references therein). Finally, we note that our results about the embedding of the slow (5.1), fast (5.2), and the fused graph (5.3) are very general and can be applied to datasets where the corresponding graph exhibits a similar structure [9]. For instance, one could imagine using this idea to study social networks, where the concept of cliques would correspond to fast subgraphs.

### 7.3. Related work

The concept of patches has proven extremely useful in many areas of image analysis: texture analysis/synthesis [29], image completion [32, 52], super-resolution
While these references do not explicitly construct a patch-graph, these works all compute distances between patches, and use the nearest neighbors of each patch to analyze and process patches. Recent works on the analysis of time-series also use patches and construct networks of patches [4, 26, 51]. All these references provide experimental evidence for the success of working on image (or signal) patches.

In this work, we provide a theoretical justification for this experimental success. We study the effect of the embedding $\Phi$ (4.8) on the organization of the patch-set. Our analysis assumes that there exists a natural partition of the patch-set into two classes: patches extracted from the smooth baseline and patches that contain sudden local changes of the image intensity or signal value. It is interesting to compare and contrast our work to the work of Singer, Shkolnisky, and Nadler [44] who provide a different theoretical explanation for the success of patch-based denoising algorithms in the limiting case where $\sigma \to 0$ and $N \to \infty$. The authors in [44] treat the matrix $P$ as a filter, which acts on an $N$-dimensional column-vector-representation of the signal or the image. Each multiplication of the probability distribution by $P$ is interpreted as the evolution of the diffusion process on the patch-graph over a time-step of duration $\sigma$. The results in [44] rely on the convergence of a properly normalized version of $P$ toward the backward Fokker-Planck operator. The authors can compute the eigenfunctions of the operator when the signal is either a one-dimensional constant function perturbed by Gaussian noise, or a one-dimensional step function also contaminated by Gaussian noise.

In contrast, our analysis is based on the analysis of the commute time on graphs that epitomize the patch-graph constructed from two classes of patches. In addition, we need not assume that the image is piece-wise constant. In fact, our experiments demonstrate that our analysis can be applied to detect many different types of anomalies: changes in the local frequency content, changes in local regularity, etc. Furthermore, our theoretical analysis holds for finite values of the number of patches $N$. It is interesting to note that Singer et al. study the mean first-passage time between patches extracted from the noisy step function. The mean first-passage time is derived from the hitting time, which is used to define the commute time. The authors in [44] use an energy argument to explain the existence of a large mean first-passage time between patches extracted from either side of the step function’s discontinuity. They argue that a high density of patches is associated with a lower potential energy, and consequently it will take longer for a random process to exit the well with such a low potential.

The energy argument in [44] adds an interesting interpretation to our analysis. Following this perspective, the slow patches can be interpreted as points sampled from a probability density function $P$ defined on $\mathbb{R}^d$ with a support that is defined along a low-dimensional manifold. This localization leads to a potential $U = -\log P$ with a deep and narrow well, from which the random walk cannot escape. This argument agrees with our findings that the average commute time between slow patches is very large, and thus, the random walk spends considerably more time in the slow subgraph before being able to reach a patch that is temporally faraway. We note that the density of edges in the graph models presented in this paper leads to graph that are not very dense ($p \sim \log(N)/N$. In contrast, the recent work in [50] (derived in the limiting case where $\sigma \to 0$ and $N \to \infty$) argues that when the density of edges becomes sufficiently large (e.g. $p \sim \log^2(N)/N$), and if the graph can be embedded into a $d$-dimensional cube, then the commute time can no longer be used as a useful metric: the random walk spends so much reaching its destination that it lost the memory of its origin.
This is clearly not the case for the graph of image patches presented in this paper, which are much sparser.

7.4. Open questions. While we obtained estimates for the average commute time on the fast and slow graph models considered separately, it would be desirable to obtain similar estimates on the fused graph. At the moment, our analysis of the fused graph relies on numerical simulations. We are also aware of a small discrepancy in the upper bound on $\kappa_F$: this bound is increasing with $L$. In fact we expect that the commute time on $\mathcal{F}(N, p)$ should decrease as $p$, and therefore $L$, increases. The reason for this apparent inconsistency is that the proof of (5.7) relies on a loose upper bound for the effective resistance between two vertices, which is provided by the geodesic distance on the graph [11]. This is not a tight inequality on a graph such as $\mathcal{F}(N, p)$. A more effective inequality, which could improve the upper bound (5.7), relies on the computation of the distribution of the number of paths $s$ of length at most $l$ between the two vertices. We could then use the fact that the commute time is bounded from above by a constant times the ratio $l/s$ [11], which would decrease the upper bound in (5.7).

Appendix A. The connectedness of the fast graph. It is necessary that the fast graph $\mathcal{F}(N, p)$ be connected to be able to apply the spectral decomposition of the commute time. Having a connected fast graph will also help us establish the bounds of Lemma (5.6). To ensure that the probability of $\mathcal{F}(N, p)$ being disconnected will vanish as $N$ gets large, we must choose $Np > \log N$ [17]. Since $p$ is defined as a function of $L$ in (5.3), any requirement on $p$ ultimately constrains $L$. First, because the maximum degree of a vertex in $\mathcal{S}(N, L)$ is $2L + 1$, according to (5.1), we require

$$2L + 1 \leq N.$$

Manipulation of this inequality leads to

$$\frac{L + 1}{N} \leq \frac{1}{2} + \frac{1}{2N}.$$

We assume that $N \geq 2$, so that

$$\frac{L + 1}{N} \leq \frac{3}{4}.$$

It follows that

$$\left(2 - \frac{L + 1}{N}\right) \geq \frac{5}{4} > 1.$$

Therefore, rewriting (5.3) and using the last inequality we have

$$p = \frac{L}{N - 1} \left(2 - \frac{L + 1}{N}\right) > \frac{L}{N} \left(2 - \frac{L + 1}{N}\right) > \frac{L}{N}.$$

Therefore, choosing $L = c \log N$ for some $c > 1$ ensures that $Np > \log N$, and consequently, the probability of $\mathcal{F}(N, p)$ being disconnected approaches zero as $N$ approaches infinity.

Appendix B. Bounding the commute times in the graph models.
B.1. Proof of the lower bound on the average commute time in the slow graph. In order to compute a lower bound on the average commute time, we consider a fixed pair of vertices in the slow graph, \( x_{n_0} \) and \( x_{m_0} \), and compute a lower bound on the commute time \( \kappa(x_{n_0}, x_{m_0}) \). We can then compute the average of this lower bound over all the pairs of vertices. To obtain the lower bound on \( \kappa(x_{n_0}, x_{m_0}) \) we use a standard tool to obtain lower bounds on commute time: the Nash-Williams inequality \( 30 \). The Nash-Williams inequality is usually formulated in terms of electrical networks. We prefer to present an equivalent formulation that is directly adapted to our problem. We first introduce the concept of edge-cutset.

**Definition B.1.** Let \( V_1 \) and \( V_2 \) be two disjoint sets of vertices. A set of edges \( E \) is an edge-cutset separating \( V_1 \) and \( V_2 \) if every path that connects a vertex in \( V_1 \) with a vertex in \( V_2 \) includes an edge in \( E \).

Given a weighted graph, which may contain loops, we define a random walk with the probability transition matrix \( P_{n,m} = \frac{w_{n,m}}{D_{n,n}} \). Let \( x_{m_0} \) and \( x_{n_0} \) be two vertices. The commute time between vertices \( x_{m_0} \) and \( x_{n_0} \), \( \kappa(x_{m_0}, x_{n_0}) \) satisfies the following lower bound.

**Lemma B.2 (Nash-Williams).** If \( x_{m_0} \) and \( x_{n_0} \) are distinct vertices in a graph that are separated by disjoint edge-cutsets \( E_k, k = 1, \ldots \), then

\[
V \sum_{k} \left[ \sum_{\{x_n, x_m\} \in E_k} w_{n,m} \right]^{-1} \leq \kappa(x_{m_0}, x_{n_0}),
\]

(B.1)

where \( \{x_m, x_n\} \) is an edge in the cutset \( E_k \) and where the volume of the graph is defined by

\[
V = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j}.
\]

Note that \( V \) can be written as

\[
V = \sum_{n=1}^{N} w_{n,n} + 2 \sum_{1 \leq m < l \leq N} w_{m,l}.
\]

(B.2)

In the case of the slow graph \( S(N, L) \) we have

\[
V = w_S N + 2w_S \left( NL - \frac{L(L+1)}{2} \right) = w_S (N(2L+1) - L(L+1)),
\]

where we used (5.2) in (B.2) to obtain the first equality above.

We now exhibit a sequence of edge-cutsets in the slow graph. We refer to Figure 22 for the construction of the cutsets. We define the first cutset \( E_1 \). If \( m_0 < L \), then \( E_1 \) needs a little more attention and is defined as the set of \( L \) edges \( \{x_i, x_j\} \), where \( i \) and \( j \) are defined by

\[
\begin{cases} 
i = 1, \ldots, m_0, \\
j = m_0 + 1, \ldots, L + i. 
\end{cases} \quad \text{(B.3)}
\]

The edge-cutset \( E_1 \) is shown in the Figure 22 for \( m_0 = 1 \) (left) and \( m_0 = 2 \) (center), for \( L = 3 \). The removal of this set of edges prevents \( x_{m_0} \) from being connected to \( x_{n_0} \). Indeed, the self
Figure 22. The small squares represent the nonzero entries in the upper triangular portion of the weight matrix $W$ of $S(N,L)$. The green entries on the diagonal are the self-loops. The edge-cutsets $E_k$ are shown in red for $m_0 = 1$ (left), $m_0 = 2$ (center), and for $m_0 \geq L$ (right). The submatrix $W(m_0 : n_0, m_0 : n_0)$ is also shown.

Figure 23. Top: edge-cutsets $E_1$ and $E_3$. Bottom: any path from $m_0$ to $n_0$ needs to use an edge of the edge-cutset $E_3$.

loop on the diagonal (green entry) does not allow the random walk to move toward $x_{n_0}$. This can be also be visualized in Figure 23, where $E_1$ is the leftmost set of edges that connect $x_{m_0}$ to that part of the graph that is connected to $x_{n_0}$. The sum of edge weights in $E_1$ is at most $L(L+1)w_S/2$. If $m_0 \geq L$, then $E_1$, is defined as the other generic edge-cutsets.

We now define the generic edge-cutsets $E_k$ as the set of $L(L+1)/2$ edges $\{x_i, x_j\}$ such that

$$
\begin{cases}
  i = m_0 + 1 + (k-2)L, \ldots, m_0 + (k-1)L, \\
  j = m_0 + 1 + (k-1)L, \ldots, L + i.
\end{cases}
$$

As seen in Figure 22-right for $k = 3$, setting the entries of $E_3$ to zero disconnects the upper and lower part of the submatrix $W(m_0 : n_0, m_0 : n_0)$, thereby isolating $x_{m_0}$ and $x_{n_0}$. Alternatively, we also see in Figure 23 that any path from $x_{m_0}$ to $x_{n_0}$ needs to go through $E_3$. Each edge-cutset $E_k, k \geq 2$ is a triangle with a height of size $L$. Therefore, after creating $E_1$, we can fit
\[
\left\lfloor \frac{n_0 - (m_0+1) + 1}{L} \right\rfloor \text{ such cutsets between } x_{m_0+1} \text{ and } x_{n_0}. \text{ The sum of the weights along the edges of each cutset } E_k, k = 2, \ldots \text{ is given by } L(L+1)w_s/2. \text{ In addition, the sum of edge weights in the first cutset } E_1 \text{ is at most } L(L+1)w_s/2. \text{ Putting everything together, the computation of the lower bound using the Nash-Williams Lemma yields}
\]
\[
V \sum_k \left[ \sum_{\{n,m\} \in E_k} w_{n,m} \right]^{-1} \geq \left[ N(2L+1) - L(L+1) \right] w_s \left( \left\lfloor \frac{n_0 - m_0}{L} \right\rfloor \frac{2}{L(L+1)w_s} + \frac{2}{L(L+1)w_s} \right) \\
\geq \left[ N(2L+1) - L(L+1) \right] \left( 2 \left( \frac{n_0 - m_0}{L} - 1 \right) + 2 \right) \\
\geq \left[ N(2L+1) - L(L+1) \right] \left( \frac{2n_0 - m_0}{L} \right).
\]

we can summarize this result in the following lemma.

**Lemma B.3.** The commute time between vertices \( x_{n_0} \) and \( x_{m_0} \) inside \( S(N, L) \) satisfies
\[
\kappa(x_{m_0}, x_{n_0}) \geq \frac{2 \left[ N(2L+1) - L(L+1) \right]}{L(L+1)} \left( \frac{n_0 - m_0}{L} \right). \tag{B.5}
\]

Finally, we bound the average commute time in the slow graph. Observe that the slow graph model \( S(N, L) \) has \( N - j \) pairs of vertices such that \( |m - n| = j \), for \( j = 1, \ldots, N - 1 \). Therefore, using the lower bound given in Lemma B.3 it follows that
\[
\sum_{1 \leq m < n \leq N} \kappa(x_m, x_n) \geq \frac{2 \left[ N(2L+1) - L(L+1) \right]}{L^2(L+1)} \sum_{j=1}^{N-1} (N - j)j.
\]

But
\[
\sum_{j=1}^{N-1} (N - j)j = \left( N \sum_{j=1}^{N-1} j - \sum_{j=1}^{N-1} j^2 \right) = \left( \frac{N^2(N-1)}{2} - \frac{N(N-1)(2N-1)}{3} \right) = \frac{N(N-1)N+1}{2}.
\]

Dividing both sides by \( N(N-1)/2 \) and simplifying yields (5.6).

**B.2. Proof of upper bound on the commute time in the fast graph.** Our approach relies on the relationship between electrical networks and random walks on graphs [16]. We begin by introducing the property of interest — the effective resistance — and its relationship to the commute time.
The electrical network perspective. For each pair of vertices $x_n$ and $x_m$ with a non zero weight $w_{n,m}$, we assign the resistance $r_{n,m} = 1/w_{n,m}$ to the edge $\{x_n, x_m\}$. We note that if $w_{n,m} = 0$, then there is no connection between $x_n$ and $x_m$, and no resistance to consider. Now, consider applying a potential difference, or voltage, across two randomly chosen vertices $x_{m_0}$ and $x_{n_0}$ in the giant component. As a result, some current flows across the resistors (edges) in the electrical network (graph). We may replace the set of resistors across which some current flows by an equivalent, effective resistance, $R_{m_0, n_0}$, that is connected between $x_{m_0}$ and $x_{n_0}$. The effective resistance $R_{m_0, n_0}$ is defined by the voltage necessary to maintain a one-unit current between $x_{m_0}$ and $x_{n_0}$. The main result in [11], is that the commute time between vertices $x_{m_0}$ and $x_{n_0}$ can be expressed as

$$\kappa(x_{m_0}, x_{n_0}) = VR_{m_0, n_0}.$$  \hfill (B.6)

Also, the effective resistance between two nodes of a network is at most the geodesic distance between them, $\delta(x_{m_0}, x_{n_0})$, scaled by $1/w_F$ [11]. It follows that

$$\kappa(x_{m_0}, x_{n_0}) \leq V \frac{\delta(x_{m_0}, x_{n_0})}{w_F}.$$  \hfill (B.7)

Using (B.2), $V$ in the fast graph can be expressed as

$$V = w_FN + 2w_FM,$$  \hfill (B.8)

where $M$ is a random variable representing the number of edges connecting distinct vertices in the giant component of the fast graph. Provided the realization is connected, which happens with probability approaching one according to appendix A, $M$ is distributed as a binomial random variable with parameters $(N(N-1)/2, p)$. Note that $M$ is also the sum of $N(N-1)/2$ Bernoulli random variables identifying the presence of an edge between the $N(N-1)/2$ pairs of distinct vertices in the fast graph. Therefore, the weak law of large numbers asserts that the ratio

$$\frac{M}{N(N-1)/2}$$

converges to $p$ in probability. Also, [17] shows that the distance

$$\delta(x_{m_0}, x_{n_0})$$

converges to $\frac{\log N}{\log(Np)}$ in probability. These results, combined with (B.8) show that the upper bound (B.7) converges to

$$(N + N(N-1)p) \frac{\log N}{\log(Np)}$$

in probability. Using (5.3) and the fact that the previous result is independent of the vertices chosen, we obtain the upper bound in (5.7).

Appendix C. Generating a random trigonometric polynomial with a specified autocorrelation. Let $z(t)$ represent a random trigonometric polynomial on $[0, 1)$ with an autocorrelation function given by

$$C(\tau) = 2(\cos(\pi\tau))^{2\beta} - 1 \quad \text{for } \tau \in [-0.5, 0.5),$$  \hfill (C.1)
for some nonnegative integer $\beta$. A Fourier expansion of $C(\tau)$ yields

$$C(\tau) = \sum_{j \in \mathbb{Z}} \hat{C}_j e^{2\pi i j \tau},$$  \hspace{1cm} (C.2)$$

with

$$\hat{C}_j = \int_0^1 C(\tau) e^{-2\pi i j \tau} d\tau = \int_0^1 \left( 2^{(1-2\beta)} \left( \sum_{k=0}^{2\beta} \left( \frac{2\beta}{k} \right) e^{2\pi i (\beta - k) \tau} \right) - 1 \right) e^{-2\pi i j \tau} d\tau$$

$$= 2^{(1-2\beta)} \sum_{k=0}^{2\beta} \frac{2\beta}{k} \int_0^1 e^{2\pi i (\beta - k - j) \tau} d\tau - \int_0^1 e^{-2\pi i j \tau} d\tau$$

$$= \begin{cases} 
2^{(1-2\beta)} \left( \frac{2\beta}{\beta} \right) - 1 & \text{if } j = 0, \\
2^{(1-2\beta)} \left( \frac{2\beta}{\beta - j} \right) & \text{if } |j| \leq \beta, \\
0 & \text{if } j > \beta, 
\end{cases}$$

Let $A_j$ and $B_j$ be independent and identically distributed Normal random variables with zero mean and unit variance. Define

$$\hat{z}_j = \sqrt{\frac{\hat{C}_j}{2}} (A_j + iB_j).$$

Finally, the signal $z(t)$ is defined as $z(t) = \sum_{j \in \mathbb{Z}} \hat{z}_j e^{2\pi i j t}$. To check that the signal $z(t)$ defined above has the correct autocorrelation, observe that linearity of the expectation, independence and zero mean of the random variables, and the fact that $\hat{C}_j = \hat{C}_{-j}$ together imply that

$$\mathbb{E}(z(t)z(t+\tau)) = \sum_{|j| \leq 2\beta} \sum_{|k| \leq 2\beta} \mathbb{E} \left( \hat{z}_j \overline{\hat{z}_k} \right) e^{-2\pi i k \tau} e^{2\pi i (j-k) t}$$

$$= \sum_{|j| \leq 2\beta} \sum_{|k| \leq 2\beta} \frac{\hat{C}_j \hat{C}_k}{2} \left[ \mathbb{E}(A_j A_k) - i\mathbb{E}(A_j B_k) + i\mathbb{E}(A_k B_j) + \mathbb{E}(B_j B_k) \right] e^{-2\pi i k \tau} e^{2\pi i (j-k) t}$$

$$= \sum_{|j| \leq 2\beta} \frac{\hat{C}_j}{2} \left[ \mathbb{E}(A_j^2) + \mathbb{E}(B_j)^2 \right] e^{-2\pi i j \tau} = \sum_{|j| \leq 2\beta} \hat{C}_j e^{2\pi i j \tau}.$$

Therefore, referencing (C.2), it follows that $\mathbb{E}(z(t)z(t+\tau)) = C(\tau)$.

**REFERENCES**


38

http://mypage.iu.edu/~rdlyons.


