TOWARDS STATIONARY TIME-VERTEX SIGNAL PROCESSING

Nathanaël Perraudin[†*], *Andreas Loukas*[†*], *Francesco Grassi*[‡] *and Pierre Vandergheynst*[†]

[†] LTS2, École Polytechnique Fédérale de Lausanne
[‡] DET, Politecnico di Torino

ABSTRACT

Graph-based methods for signal processing have shown promise for the analysis of data exhibiting irregular structure, such as those found in social, transportation, and sensor networks. Yet, though these systems are often dynamic, state-of-the-art methods for graph signal processing ignore the time dimension. To address this shortcoming, this paper considers the statistical analysis of time-varying graph signals. We introduce a novel definition of *joint (time-vertex) stationarity*, which generalizes the classical definition of time stationarity and the recent definition appropriate for graphs. This gives rise to a scalable Wiener optimization framework for denoising, semi-supervised learning, or more generally inverting a linear operator, that is provably optimal. Experimental results on real weather data demonstrate that taking into account graph and time dimensions jointly can yield significant accuracy improvements in the reconstruction effort.

Index Terms— Graph signal processing, time-vertex signal processing, joint stationarity, Wiener filter

1. INTRODUCTION

Whether examining opinion dichotomy in social networks [1], how traffic evolves in the roads of a city [2], or neuronal activation patterns in the brain [3], much of the high-dimensional data one encounters exhibit complex non-euclidean properties. Within the field of signal processing, one of the main research thrusts has been to extend harmonic analysis to graph signals, i.e., signals supported on the vertices of graphs. The key breakthrough has been the introduction of a notion of frequency appropriate for graph signals and of the associated graph Fourier transform (GFT), leading to advances in problems such as denoising [4] and semi-supervised learning [5, 6]. Yet, SoA graph frequency based methods often fail to produce useful results when applied to real datasets. One of the main reasons is that, with few recent exceptions [7, 8, 9, 10], they ignore time, treating successive signals independently or performing a global average [3, 11, 12].

In this paper we consider the *statistical analysis of time-varying graph signals*. Our results are inspired by the recent introduction of a joint temporal and graph Fourier transform (JFT), a generalization of GFT appropriate for time-varying graph signals [13], and the recent generalization of stationarity for graphs [11, 14, 15]. Our main contribution is a novel definition of time-vertex (wide-sense) stationarity, or *joint stationarity* for short, that generalizes stationarity in the time and vertex domains. We show that joint stationarity carries important properties classically associated with stationarity. Moreover, it leads to an optimal Wiener framework for solving denoising and interpolating time-varying graph signals, that is composed out of two key components: a scalable joint power spectral density estimation method, and an optimization framework suitable for deconvolution under additive error. Experiments with a real weather dataset illustrate the superior performance of our method, demonstrating that joint stationarity is a useful assumption in practice.

2. PRELIMINARIES

Our objective is to model the evolution of graph signals, i.e., signals supported on the vertices $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ of a undirected graph $G = (\mathcal{V}, \mathcal{E}, \boldsymbol{W}_G)$, with $\mathcal{E}$ the set of edges and $\boldsymbol{W}_G$ the weighted adjacency matrix. A more convenient representation of G is the (combinatorial[1]) Laplacian matrix $\boldsymbol{L}_G = \mathrm{diag}(\boldsymbol{W}_G \boldsymbol{1}_N) - \boldsymbol{W}_G$, where $\boldsymbol{1}_N$ is the all-ones vector of size N.We use sub/superscripts G, T, J to denote respectively the graph, time, and joint domains.

Harmonic vertex analysis. In the context of graph signal processing, the Laplacian matrix gives rise to a graph-specific notion of frequency. The GFT of a graph signal $\boldsymbol{x} \in \mathbb{R}^N$ is defined as $\mathrm{GFT}\{\boldsymbol{x}\} = \boldsymbol{U}_G^* \boldsymbol{x}$, where $\boldsymbol{U}_G$ is the eigenvectors matrix of $\boldsymbol{L}_G$ and thus $\boldsymbol{L}_G = \boldsymbol{U}_G \boldsymbol{\Lambda}_G \boldsymbol{U}_G^*$. The GFT allows us to extend filtering to graphs [17, 18, 19]. Filtering a signal $\boldsymbol{x}$ with a graph filter $h(\boldsymbol{L}_G)$ corresponds to element-wise multiplication in the spectral domain

$$h(\boldsymbol{L}_G)\boldsymbol{x} \triangleq \mathrm{GFT}^{-1}\{h(\boldsymbol{\Lambda}_G) \circ \mathrm{GFT}\{\boldsymbol{x}\}\} = \boldsymbol{U}_G h(\boldsymbol{\Lambda}_G) \boldsymbol{U}_G^* \boldsymbol{x},$$

with ($\circ$) being the element-wise multiplication (Hadamard product). The scalar function $h : \mathbb{R}_+ \mapsto \mathbb{R}$, referred to as the

*A. Loukas and N. Perraudin contributed equally to this work.

This work has been supported by the Swiss National Science Foundation research project *Towards Signal Processing on Graphs* (grant number: 2000_21/154350/1).

[1] Our results are applicable to any positive semi-definite matrix representation of a graph or to the recently introduced shift operator [16].

graph frequency response, is applied to each diagonal entry of $\boldsymbol{\Lambda}_G$. It is often convenient to represent the diagonal of matrix $h(\boldsymbol{\Lambda}_G)$ as a vector, in which case we write $\boldsymbol{h} = \mathrm{diag}(h(\boldsymbol{\Lambda}_G))$. $\boldsymbol{U}_G^*, \boldsymbol{U}_G^{\intercal}$ and $\bar{\boldsymbol{U}}_G$ denote respectively the transposed complex conjugate, the transpose and the complex conjugate of $\boldsymbol{U}_G$. We will also use the notion of *graph localization* [11, 19], a generalization of the translation operator for graphs[2]. The value at the vertex v_{i_2} of a filter with frequency response h localized onto vertex v_{i_1} is

$$\mathcal{T}_{i_1}^{G} h(i_2) \triangleq h(\boldsymbol{L}_G)\, \boldsymbol{\delta}_{i_1}(i_2) = \sum_{n=1}^{N} h(\lambda_n)\, \bar{\boldsymbol{u}}_n(i_1)\, \boldsymbol{u}_n(i_2), \quad (1)$$

where $\boldsymbol{\delta}_{i_1}$ is a Kronecker delta centered at vertex v_{i_1}. We use the notation $\boldsymbol{u}_n(i) = [\boldsymbol{U}_G]_{i,n}$ and $\lambda_n = [\boldsymbol{\Lambda}_G]_{n,n}$. For a sufficiently regular function h, this operation localizes the filter around v_i [19, Theorem 1 and Corollary 2].

The concept of localization is linked to that of *translation* in the time domain. If $\mathcal{T}_{\tau}^{T} h$ is the localization operator taken on a cycle graph of T vertices (representing time), localization is equivalent to translation

$$\mathcal{T}_{\tau}^{T} h(t) = \mathcal{T}_{0}^{T} h(t-\tau), \quad t, \tau = 1, \dots, T \text{ and } t > \tau. \quad (2)$$

Above $\mathcal{T}_{0}^{T} h = \boldsymbol{U}_T \boldsymbol{h}$ is the inverse Fourier transform of $\boldsymbol{h}$ and $\boldsymbol{U}_T$ is the orthonormal Fourier basis such that $\boldsymbol{L}_T = \boldsymbol{U}_T\, \boldsymbol{\Lambda}_T\, \boldsymbol{U}_T^*$. We also denote $\boldsymbol{\Omega}$ the diagonal matrix of angular frequencies (i.e., $\boldsymbol{\Omega}_{tt} = \omega_t = 2\pi t/T$). In the case of irregular graphs, localization differs further from translation because the shape of the localized filter adapts to the graph and varies as a function of its topology. Additional insights about the localization operator can be found in [11, 19, 17, 20].

Harmonic time-vertex analysis. Suppose that a graph signal $\boldsymbol{x}_t$ is sampled at T successive regular intervals of unit length. The time-varying graph signal $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_T] \in \mathbb{R}^{N\times T}$ is then the matrix having graph signal $\boldsymbol{x}_t$ as its t-th column. Equivalently, $\boldsymbol{X} = \left[\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^N\right]^{\intercal}$ holds N timeseries $\boldsymbol{x}^i \in \mathbb{R}^T$, one for each vertex v_i. Throughout this paper, we denote as $\boldsymbol{x} = \mathrm{vec}(\boldsymbol{X})$ the vectorized representation of the matrix $\boldsymbol{X}$. The frequency representation of $\boldsymbol{X}$ is given by the joint (time-vertex) Fourier transform [13]

$$\mathrm{JFT}\{\boldsymbol{X}\} = \boldsymbol{U}_G^* \boldsymbol{X} \bar{\boldsymbol{U}}_T. \quad (3)$$

In vector form the joint Fourier transform is $\mathrm{JFT}\{\boldsymbol{x}\} = \boldsymbol{U}_J^* \boldsymbol{x}$, where $\boldsymbol{U}_J = \boldsymbol{U}_T \otimes \boldsymbol{U}_G$ is unitary, and $(\otimes)$ denotes the kroneker product. The inverse JFT in matrix and vector form is, respectively, $\mathrm{JFT}^{-1}\{\boldsymbol{X}\} = \boldsymbol{U}_G \boldsymbol{X} \boldsymbol{U}_T^{\intercal}$ and $\mathrm{JFT}^{-1}\{\boldsymbol{x}\} = \boldsymbol{U}_J \boldsymbol{x}$.

Filtering and localization can also be extended to the joint (time-vertex) domain. A *joint filter* $h(\boldsymbol{L}_J)$ is a function defined in the joint spectral domain $h : \mathbb{R}_+ \times \mathbb{R} \mapsto \mathbb{R}$ that is evaluated at the graph eigenvalues λ_G and the angular frequencies ω. The output of a joint filter is

$$h(\boldsymbol{L}_J)\boldsymbol{x} \triangleq \boldsymbol{U}_J\, h(\boldsymbol{\Lambda}_G, \boldsymbol{\Omega})\, \boldsymbol{U}_J^* \boldsymbol{x}, \quad (4)$$

where $h(\boldsymbol{\Lambda}_G, \boldsymbol{\Omega})$ is a $NT \times NT$ diagonal matrix with $[h(\boldsymbol{\Lambda}_G, \boldsymbol{\Omega})]_{k,k} = h(\lambda_n, \omega_\tau)$ and $k = N(\tau - 1) + n$. Equivalently, if we define the matrix $\boldsymbol{H}_{n,\tau} = h(\lambda_n, \omega_\tau)$ of dimension $N \times T$ for every λ_n and ω_τ, we have

$$h(\boldsymbol{L}_J)\boldsymbol{x} = \mathrm{vec}\left(\mathrm{JFT}^{-1}\{\boldsymbol{H} \circ \mathrm{JFT}\{\boldsymbol{X}\}\}\right). \quad (5)$$

In analogy to (1), we define the *joint localization operator* as

$$\begin{aligned} \mathcal{T}_{i,t}^{J}\, h \quad &\triangleq \quad \mathrm{mat}(h(\boldsymbol{L}_J)\, (\boldsymbol{\delta}_t \otimes \boldsymbol{\delta}_i)) & (6)\\ &= \quad \mathrm{JFT}^{-1}\{\boldsymbol{H} \circ \mathrm{JFT}\{\boldsymbol{\delta}_i \boldsymbol{\delta}_t^{\intercal}\}\} & (7) \end{aligned}$$

where $\mathrm{mat}(\mathrm{vec}(\boldsymbol{X})) = \boldsymbol{X}$ is the matricization operator. To link (6) with graph localization (1) and the classical translation operator, we observe the following relations

$$\begin{aligned} &\mathcal{T}_{i_1,t_1}^{J}\, h(i_2, t_2) = \\ &\quad = \frac{1}{T} \sum_{\substack{n=1\\ \tau=1}}^{N,T} h(\lambda_n, \omega_\tau) \bar{\boldsymbol{u}}_n(i_1) \boldsymbol{u}_n(i_2) e^{2\pi j \frac{(\tau-1)(t_2-t_1)}{T}} \\ &\quad = \mathcal{T}_{i_1,0}^{J}\, h(i_2, t_2 - t_1) & (8)\\ &\quad = \sum_{n=1}^{N} \left[\mathcal{T}_{t_1}^{T} \boldsymbol{H}_{n,\cdot}\right](t_2)\, \bar{\boldsymbol{u}}_n(i_1) \boldsymbol{u}_n(i_2) & (9)\\ &\quad = \frac{1}{T} \sum_{\tau=1}^{T} \left[\mathcal{T}_{i_1}^{G} \boldsymbol{H}_{\cdot,\tau}\right](i_2)\, e^{2\pi j \frac{(\tau-1)(t_2-t_1)}{T}}. & (10) \end{aligned}$$

From (8) and (9), it follows that joint localization consists of first translating independently in time each line of the matrix $\boldsymbol{H}$ and then localizing independently on the graph each column of the resulting matrix. Joint localization is thus equivalent to a successive application of a graph and time localization operator. Furthermore, based on (10), the localization in time and graph can be performed in any order.

3. JOINT TIME-VERTEX STATIONARITY

Let $\boldsymbol{X}$ be a discrete multivariate stochastic process with finite number of time-steps T that is indexed by vertex v_i and time t. We refer to such processes as *joint time-vertex processes*, or *joint* processes. Let us review the established definitions of stationarity over time and vertex domains, respectively.

Definition 1 (Time stationarity)**.** *A joint process $\boldsymbol{X}$ is Time Wide-Sense Stationary (TWSS), if and only if, for each vertex v_i, the expected value is constant over the time domain $\mathbb{E}\left[\boldsymbol{x}^i\right] = c_i \boldsymbol{1}_T$ and there exists a function γ_i, for which*

$$[\boldsymbol{\Sigma}_{\boldsymbol{x}^i}]_{t,\cdot} = \left[\mathbb{E}\left[\boldsymbol{x}^i \boldsymbol{x}^{i*}\right] - \mathbb{E}\left[\boldsymbol{x}^i\right]\mathbb{E}\left[\boldsymbol{x}^{i*}\right]\right]_{t,\cdot} = \mathcal{T}_t^{T} \gamma_i$$

Function γ_i is the autocorrelation function of signal $\boldsymbol{x}^i$ in the Fourier domain, and is also referred to as Time Power Spectral Density (TPSD).

[2] Stationarity is classically defined as the invariance of statistical moments of a signal with respect to translation. This definition however cannot be directly generalized to graphs, which do not possess regular structure and thus lack of an isometric translation operator.

Thus, using (7) we recover the classical definition, where the autocorrelation function depends only on the time difference: $[\boldsymbol{\Sigma}_{\boldsymbol{x}^i}]_{t,\tau} = \mathcal{T}_0^{\,T}\gamma_i(t-\tau)$. We observe that the TPSD is the Fourier transform of the autocorrelation, agreeing with the Wiener-Khintchine Theorem [21]. This consideration allows us to generalize the concept of stationarity to graph signals [11, 15].

Definition 2 (Vertex stationarity)**.** *A joint process $\boldsymbol{X}$ is called Vertex Wide-Sense Stationary (VWSS), if and only if, for each time t, the expected value is in the null space of the Laplacian $\boldsymbol{L}_G\mathbb{E}[\boldsymbol{x}_t] = \boldsymbol{0}_N$ and a graph filter $s_t(\boldsymbol{L}_G)$ exists for which*

$$[\boldsymbol{\Sigma}_{\boldsymbol{x}_t}]_{i,\cdot} = [\mathbb{E}[\boldsymbol{x}_t\boldsymbol{x}_t^*] - \mathbb{E}[\boldsymbol{x}_t]\,\mathbb{E}[\boldsymbol{x}_t^*]]_{i,\cdot} = \mathcal{T}_i^{\,G} s_t.$$

Function s_t is the autocorrelation function of signal $\boldsymbol{x}_t$ in the graph Fourier domain and is also referred to as Vertex Power Spectral Density (VPSD).

Considering that the null space of $\boldsymbol{L}_T$ is the span of the constant eigenvector $\boldsymbol{1}_T$, the first condition is analogous to the first one of the time stationarity definition. Moreover, the condition for the second moment is a natural generalization of the second condition of time stationarity where we suppose invariance under the localization operator. This is in fact equivalent to a generalization of the Wiener-Khintchine theorem and implies that $\boldsymbol{\Sigma}_{\boldsymbol{x}_t}$ is jointly diagonalizable with $\boldsymbol{L}_G$.

We now unify the TWSS and GWSS in order to leverage both the time and vertex domain statistics.

Definition 3 (Joint stationarity)**.** *A process $\boldsymbol{X}$ is called Jointly (or time-vertex) Wide-Sense Stationary (JWSS), if and only if the expected value is in the null space of the joint Laplacian $\boldsymbol{L}_J\mathbb{E}[\boldsymbol{x}] = \boldsymbol{0}_{NT}$ and a joint filter $h(\boldsymbol{L}_J)$ exists, for which*

$$[\boldsymbol{\Sigma}_{\boldsymbol{x}}]_{k,\cdot} = [\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^*] - \mathbb{E}[\boldsymbol{x}]\,\mathbb{E}[\boldsymbol{x}^*]]_{k,\cdot} = vec\big(\mathcal{T}_{i,t}^{\,J}\,h\big)\,,$$

where $k = N(t-1)+i$. Function h is the autocorrelation function of signal $\boldsymbol{x}$ in the joint Fourier domain and is referred to as time-vertex power spectral density or Joint Power Spectral Density (JPSD).

The definition above is equivalent to stating that the mean is constant, and the covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ is jointly diagonalizable with $\boldsymbol{L}_J$. The latter statement is a generalization the Wiener-Khintchine theorem and is claimed next.

Theorem 1. *A process $\boldsymbol{X}$ is JWSS if and only if $\boldsymbol{L}_J\mathbb{E}[\boldsymbol{x}] = \boldsymbol{0}_{NT}$ and its covariance matrix is jointly diagonalizable by the joint Fourier basis $\boldsymbol{U}_J$.*

Interestingly, assuming joint stationarity is equivalent to assuming stationarity in both domains at the same time.

Theorem 2. *If a joint process $\boldsymbol{X}$ is JWSS, then it is both TWSS and VWSS.*

Moreover, white i.i.d. noise $\boldsymbol{w} \in \mathbb{R}^{NT}$ is JWSS for any graph. Indeed, the first moment $\mathbb{E}[\boldsymbol{w}]$ is constant for any time and vertex. $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ is diagonalized by the joint Fourier basis of any graph $\boldsymbol{\Sigma}_{\boldsymbol{w}} = \boldsymbol{I} = \boldsymbol{U}_J\boldsymbol{I}\boldsymbol{U}_J^*$.

An other interesting property of JWSS processes is that stationarity is preserved through a filtering operation.

Theorem 3. *When a joint filter $f(\boldsymbol{L}_J)$ is applied to a JWSS process $\boldsymbol{X}$, the result $\boldsymbol{Y}$ remains JWSS with mean $f(0,0)\mathbb{E}[\boldsymbol{X}]$ and JPSD that satisfies $h_{\boldsymbol{Y}}(\lambda,\omega) = f^2(\lambda,\omega)\cdot h_{\boldsymbol{X}}(\lambda,\omega)$.*

Theorem 3 provides a way to produce JWSS signals with a prescribed PSD f^2 by filtering white noise with the joint filter $f(\boldsymbol{L}_J)$. In the following, we will assume for simplicity $\mathbb{E}[\boldsymbol{x}] = 0\cdot\boldsymbol{1}_N$. For the proofs of the previous theorems we refer the reader to the extended version of this work [22].

4. JOINT PSD ESTIMATION

The basic idea behind our approach[3] stems from two established methods used to estimate the TPSD of a temporal signal, namely Bartlett's [24] and Welch's methods [25]. We can see the TPSD estimation of both methods as the averaging over time of the squared coefficients of a Short Time Fourier Transform (STFT). For a discrete signal $\boldsymbol{s}$ of length T, the circular discrete sampled STFT of $\boldsymbol{s}$ at the m-th (out of M) frequency band, and under window $\boldsymbol{g}$ is

$$\text{STFT}\{\boldsymbol{s}\}(k,m) \triangleq \sum_{t=1}^{T}\boldsymbol{s}(t)\,\overline{\boldsymbol{g}\,(t_k)}\,e^{-2\pi j\,\frac{(t-1)(m-1)}{M}},$$

where $t_k = \text{mod}(t - a(k-1), T) + 1$, scalar a is the shift in time between two successive windows [26, equation 1], and $\text{mod}(t,T)$ finds the remainder after division by T i.e., $\text{mod}(t,T) = t - T\lfloor\frac{t}{T}\rfloor$. Note that $k = 0,1,\ldots,\lfloor\frac{T}{a}\rfloor - 1$ is the time band centered at ka and that $m = 1,\ldots,M$ is the frequency band index. For additional insights about this transform, we refer the reader to [27, 28].

Joint PSD estimation. We propose to use the GFT of the STFT as a tool to estimate the joint PSD. Consider a time window $\boldsymbol{g}$ and a time-vertex signal $\boldsymbol{X}$. We define the coefficients' tensor as

$$\boldsymbol{C}_{n,k,m} \triangleq \sum_{i=1}^{N}[\boldsymbol{U}_G]_{i,n}\,\text{STFT}\{\boldsymbol{x}^i\}(k,m)$$

An usual parameter for M is the support size of $\boldsymbol{g}$. Then, for half-overlapping windows, we set a to $M/2$. For any λ_n and $\omega_m = 2\pi m/M$, our JPSD estimator is

$$\tilde{h}\,(\lambda_n,\omega_m) \triangleq \frac{a}{T\|\boldsymbol{g}\|_2^2}\sum_{k=0}^{\lfloor T/a\rfloor-1}\boldsymbol{C}_{n,k,m}^2. \tag{11}$$

[3]To estimate the statistics of a joint process one may use the covariance matrix. Since its size is $NT\times NT$, in many cases this matrix is not computable nor can be even stored. Moreover, the number of samples needed for obtaining a good estimate has been shown to be $O(NT\log(NT))$ [23].

In order to get an estimate of h at $\omega \neq \omega_m$, we interpolate between the known points. Alternatively, with sufficient computation power, one may set $M = T$.

5. OPTIMIZATION FRAMEWORK

We suppose that the measurements $\boldsymbol{y}$ are generated by a linear model $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$ where $\boldsymbol{A}$ is a general linear operator. Further, suppose that the JPSD of $\boldsymbol{x}$ is $h_{\boldsymbol{X}}$, whereas the noise $\boldsymbol{w}$ is zero mean, has JPSD $h_{\boldsymbol{W}}$ and may follow any distribution.

We propose to recover $\boldsymbol{x}$ as the solution of the Wiener optimization problem

$$\dot{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \|f(\boldsymbol{L}_J)(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])\|_2^2, \quad (12)$$

where $f(\lambda, \omega) \triangleq \left|\sqrt{h_{\boldsymbol{W}}(\lambda,\omega)/h_{\boldsymbol{X}}(\lambda,\omega)}\right|$ is the response of joint filter $f(\boldsymbol{L}_J)$ (see (4)). In the noise-less case, one alternatively solves the problem

$$\dot{\mathrm{x}} = \arg\min_{\boldsymbol{x}} \|h_{\boldsymbol{X}}^{-\frac{1}{2}}(\boldsymbol{L}_J)\,\boldsymbol{x}\|_2^2, \quad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}. \quad (13)$$

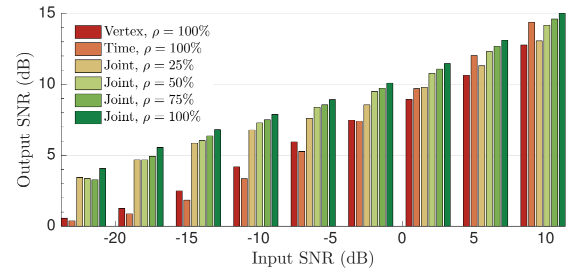
Intuitively, weight $f(\lambda, \omega)$ heavily penalizes frequencies associated with low SNR and vice-versa. Thanks to proximal splitting methods, we can derive an algorithm for solving Problem (12) that requires only the application of $\boldsymbol{A}$ and spectral graph filtering, thus scaling almost linearly (with $T \times |\mathcal{E}|$). Moreover, we can show that 1) if $\boldsymbol{X}$ is a Gaussian process, then the solution of Problem (12) coincides with a MAP estimator, 2) if $\boldsymbol{A}$ is a linear operator, then it coincides with the minimum mean square error linear estimator, 3) if $\boldsymbol{A} = a(\boldsymbol{L}_J)$ is a joint filter, then it is a joint Wiener filter [13]. Details can be found in the extended version [22].

6. EXPERIMENTS

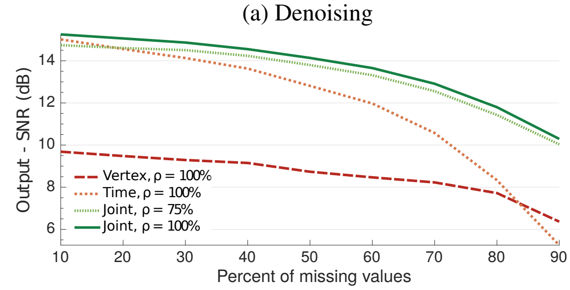
Our experiment aims to show that 1) joint stationarity is a useful model, even in datasets which may violate the strict conditions of our definition, and 2) it can yield a significant increase in denoising and recovery accuracy, as compared to time- or vertex-based methods, on a real dataset. We remark that our simulations were done using the GSPBOX [29], the UNLocBoX [30], and the LTFAT [26].

Experimental setup. The French national meteorological service has published in open access a dataset[4] with hourly weather observations collected in January 2014 in the region of Brest (France). The graph is built connecting the nearest weather stations. As sole pre-processing, we remove the mean (over time and stations) of the temperature. The dataset, which consisted of a total of $T = 744$ timesteps, was split into two parts of size ρT and $(1-\rho)T$, respectively. We use the first part of the dataset to estimate the PSD and the second to quantify the joint filter performance. We compare our method to the state-of-the-art Wiener filters for the disjoint time/vertex domains, which are known to outperform non-statistics based methods, such as graph/time Tikhonov and Total Variation. To highlight the benefit of the joint approach, in the disjoint cases we use the entire dataset to estimate the PSD (for $\rho = 1$ the same data are used for both training and testing).

[4] Access to the raw data is possible directly from https://donneespubliques.meteofrance.fr/donnees_libres/Hackathon/RADOMEH.tar.gz



(a) Denoising



(b) Recovery

Fig. 1: The joint stationarity approach becomes especially meaningful when the available data are very noisy or are few. The recovery performance is slightly improved when a larger percentage ρ of data are available for training.

Denoising. For this experiment, we add Gaussian noise to the data and perform denoising using Wiener filter ($\boldsymbol{A} = \boldsymbol{I}$ in problem (12)). The result is displayed in Figure 1. Joint stationarity outperforms time or vertex stationarity especially when the noise level is high. Indeed, joint stationarity allows the estimator to average over more samples. The effect of the dataset size can be observed through the parameter ρ, with larger ρ resulting in higher accuracy. Especially for large input SNR, the joint approach becomes particularly meaningful as it outperforms other approaches, even when a very small portion of the data is used for JPSD estimation.

Recovery. We also consider a recovery problem, where a given percentage of entries of matrix $\boldsymbol{X}$ is missing. Figure 1 depicts the recovery error obtained using problem (13). Again, we observe a significant improvement over competing methods. This improvement is achieved because the joint approach leverages the correlation both in the time and in the vertex domain: each random variable in a TWSS or VWSS process is dependent on only $T-1$ or $N-1$ other random variables, respectively (rather than $NT-1$ as in the joint case), implying a higher recovery variance.

7. REFERENCES

[1] Lada A Adamic and Natalie Glance, "The political blogosphere and the 2004 us election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*. ACM, 2005, pp. 36–43.

[2] Prashanth Mohan, Venkata N Padmanabhan, and Ramachandran Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 2008, pp. 323–336.

[3] Weiyu Huang, Leah Goldsberry, Nicholas F Wymbs, Scott T Grafton, Danielle S Bassett, and Alejandro Ribeiro, "Graph frequency analysis of brain signals," *arXiv preprint arXiv:1512.00037*, 2015.

[4] Fan Zhang and Edwin R Hancock, "Graph spectral image smoothing using the heat kernel," *Pattern Recognition*, vol. 41, no. 11, pp. 3328–3342, 2008.

[5] Alexander J Smola and Risi Kondor, "Kernels and regularization on graphs," in *Learning theory and kernel machines*, pp. 144–158. Springer, 2003.

[6] Mikhail Belkin and Partha Niyogi, "Semi-supervised learning on riemannian manifolds," *Machine learning*, vol. 56, no. 1-3, pp. 209–239, 2004.

[7] Elvin Isufi, Andreas Loukas, Andrea Simonetto, and Geert Leus, "Autoregressive moving average graph filtering," *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 274–288, 2017.

[8] Jonathan Mei and José M. F. Moura, "Signal processing on graphs: Modeling (causal) relations in big data," *CoRR*, vol. abs/1503.00173, 2015.

[9] Elvin Isufi, Andreas Loukas, Andrea Simonetto, and Geert Leus, "Separable autoregressive moving average graph-temporal filters," in *24th European Signal Processing Conference (EUSIPCO)*. IEEE, 2016, pp. 200–204.

[10] Elvin Isufi, Andrea Simonetto, Andreas Loukas, and Geert Leus, "Stochastic graph filtering on time-varying graphs," in *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec 2015, pp. 89–92.

[11] Nathanaël Perraudin and Pierre Vandergheynst, "Stationary signal processing on graphs," *arXiv preprint arXiv:1601.02522*, 2016.

[12] Vassilis Kalofolias, "How to learn a graph from smooth signals," in *th International Conference on Artificial Intelligence and Statistics AISTATS, Cadiz, Spain*, 2016.

[13] Andreas Loukas and Damien Foucard, "Frequency analysis of temporal graph signals," *arXiv preprint arXiv:1602.04434*, 2016.

[14] Benjamin Girault, "Stationary graph signals using an isometric graph translation," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*. IEEE, 2015, pp. 1516–1520.

[15] Antonio G Marques, Santiago Segarra, Geert Leus, and Alejandro Ribeiro, "Stationary graph processes and spectral estimation," *arXiv preprint arXiv:1603.04667*, 2016.

[16] Aliaksei Sandryhaila and José MF Moura, "Discrete signal processing on graphs," *IEEE transactions on signal processing*, vol. 61, pp. 1644–1656, 2013.

[17] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[18] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *Signal Processing Magazine, IEEE*, vol. 30, no. 3, pp. 83–98, 2013.

[19] David I Shuman, Benjamin Ricaud, and Pierre Vandergheynst, "Vertex-frequency analysis on graphs," *arXiv preprint arXiv:1307.5708*, 2013.

[20] Nathanael Perraudin, Benjamin Ricaud, David Shuman, and Pierre Vandergheynst, "Global and local uncertainty principles for signals on graphs," *arXiv preprint arXiv:1603.03030*, 2016.

[21] Norbert Wiener, "Generalized harmonic analysis," *Acta mathematica*, vol. 55, no. 1, pp. 117–258, 1930.

[22] Andreas Loukas and Nathanaël Perraudin, "Stationary time-vertex signal processing," *CoRR*, vol. abs/1611.00255, 2016.

[23] Mark Rudelson, "Random vectors in the isotropic position," *Journal of Functional Analysis*, vol. 164, no. 1, pp. 60–72, 1999.

[24] Maurice S Bartlett, "Periodogram analysis and continuous spectra," *Biometrika*, pp. 1–16, 1950.

[25] Peter Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on audio and electroacoustics*, pp. 70–73, 1967.

[26] Zdenek Prusa, Peter L. Sondergaard, Nicki Holighaus, Christoph Wiesmeyr, and Peter Balazs, "The Large Time-Frequency Analysis Toolbox 2.0," in *Sound, Music, and Motion*, Mitsuko Aramaki, Olivier Derrien, Richard Kronland-Martinet, and Solvi Ystad, Eds., Lecture Notes in Computer Science, pp. 419–442. Springer International Publishing, 2014.

[27] Karlheinz Gröchenig, *Foundations of time-frequency analysis*, Springer Science & Business Media, 2013.

[28] Hans G Feichtinger and Thomas Strohmer, *Gabor analysis and algorithms: Theory and applications*, Springer Science & Business Media, 2012.

[29] Nathanaël Perraudin, Johan Paratte, David Shuman, Vassilis Kalofolias, Pierre Vandergheynst, and David K. Hammond, "GSPBOX: A toolbox for signal processing on graphs," *ArXiv e-prints*, Aug. 2014.

[30] N. Perraudin, D. Shuman, G. Puy, and P. Vandergheynst, "UNLocBoX A matlab convex optimization toolbox using proximal splitting methods," *ArXiv e-prints*, Feb. 2014.