DISCRETE SIGNAL PROCESSING ON GRAPHS: GRAPH FOURIER TRANSFORM

Aliaksei Sandryhaila and José M. F. Moura

Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

ABSTRACT

We propose a novel discrete signal processing framework for the representation and analysis of datasets with complex structure. Such datasets arise in many social, economic, biological, and physical networks. Our framework extends traditional discrete signal processing theory to structured datasets by viewing them as signals represented by graphs, so that signal coefficients are indexed by graph nodes and relations between them are represented by weighted graph edges. We discuss the notions of signals and filters on graphs, and define the concepts of the spectrum and Fourier transform for graph signals. We demonstrate their relation to the generalized eigenvector basis of the graph adjacency matrix and study their properties. As a potential application of the graph Fourier transform, we consider the efficient representation of structured data that utilizes the sparseness of graph signals in the frequency domain.

Index Terms— Graph signal processing, graph signal, graph filter, graph spectrum, graph Fourier transform, generalized eigenvectors, sparse representation.

1. INTRODUCTION

Recently we have been observing a growth of interest in the efficient techniques for representation, analysis and processing of large datasets emerging in various fields and applications, such as sensor and transportation networks, internet and world wide web, image and video databases, and social and economic networks. These datasets share a common trait: their elements are related to each other in a structured manner, for example, through similarities or dependencies between data elements. This relational structure is often represented with graphs, in which data elements correspond to nodes, relation between elements are represented by edges, and the strength or significance of relations is reflected in edge weights.

The analysis and processing of structured data has been studied in multiple ways. Graph properties, such as degree distributions, node centrality and betweenness, and clustering, are often used to infer the community structure and interaction in social and economic networks [1, 2]. Inference and learning of structured datasets can be performed using graphical models [3, 4] by viewing data elements as random variables and expressing their probabilistic dependencies between each other with graph edges. Data learning, clustering, and classification has been approached using spectral graph theory [5]. A common feature of these approaches, however, is that they analyze the graphs that represents the relational structure of datasets, rather than the datasets themselves. Another technique for the representation and spectral analysis of data based on the Laplacian matrix of the graph and its eigenvectors has become popular recently [6, 7]. This technique is more similar to existing signal processing approaches, and to our work in particular; however, it is restricted to undirected graphs with real, non-negative edge weights.

We propose a framework, called *discrete signal processing on graphs* (DSP_G), for the representation, processing, and analysis of structured datasets that can be represented by graphs. Our framework extends the traditional discrete signal processing (DSP) theory that studies signals with linear structure, such as time series and space signals. e.g. images, to signals with complex, non-linear structure. We discuss the notions of signals and filters on graphs, and then define the concepts of spectral decomposition, spectrum, and Fourier transform for graph signals. We identify their relation to the generalized eigenvectors of the adjacency matrices of representation graphs and study their properties. As a potential application of the graph Fourier transform, we consider efficient data representation and compression. In particular, we demonstrate that if a graph signal is sparsely represented in the spectral domain, i.e. its frequency content is dominated by few frequencies, then it can be efficiently approximated with only a few spectrum coefficients.

This work was supported in part by AFOSR grant FA95501210087.

2. SIGNALS AND FILTERS ON GRAPHS

In this section we discuss the notions of graph signals and filters. These concepts are defined and studied in [8].

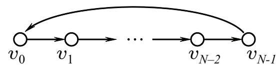
Graph signals. If we consider a quantitative dataset for which we are given information about the relationship between its elements, we can represent it as a numerical-valued signal indexed by a graph. For example, for a set of sensor measurements, the relation between measurements from different sensors can be expressed through the physical distance between sensors. For a collection of researchers and their publication records, the relation can be given by their collaborations and publication coauthoring. Assuming that the dataset is finite, we can treat it as a set of vectors

$$\mathcal{S} = \left\{ \mathbf{s} : \mathbf{s} = (s_0, \ldots, s_{N-1})^T, s_n \in \mathbb{C} \right\}. \tag{1}$$

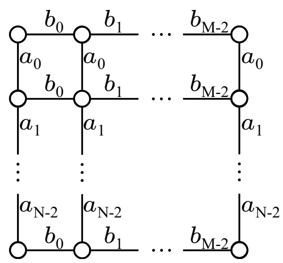
Then, we can represent the relation between coefficients s_n of $\mathbf{s}$ with a graph $G = (\mathcal{V}, \mathbf{A})$, so that $\mathcal{V} = \{v_0, \ldots, v_{N-1}\}$ is a set of N nodes, and $\mathbf{A}$ is a $N \times N$ weighted adjacency matrix. Each coefficient s_n corresponds to (is indexed by) node v_n, and the weight $\mathbf{A}_{n,m}$ of the directed edge from v_m to v_n expresses the degree of relation of s_n to s_m. Note that edge weights $\mathbf{A}_{n,m}$ can take arbitrary real or complex values (for example, if data elements are negatively correlated). We call a signal $\mathbf{s}$ indexed by a graph G a *graph signal*.

Graph signals, in general, can be complex-valued. Furthermore, they can be added together and scaled by constant coefficients. Hence, they form a vector space. If no additional assumptions are made on their values, the set $\mathcal{S}$ of graph signals corresponds to the N-dimensional complex vector space $\mathcal{S} = \mathbb{C}^N$.

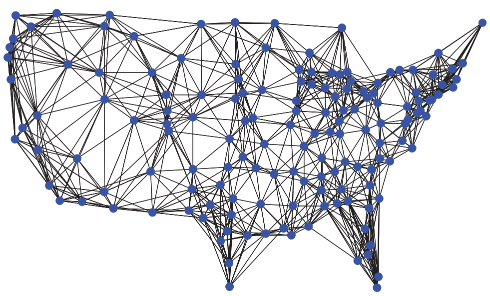
We illustrate representation graphs with several examples. The graph in Fig. 1(a) represents a finite, periodic discrete time series [9, 10]. It is a directed, cyclic graph, with directed edges of the same weight, reflecting the causality of a time series. The periodicity of the time series is captured by the edge from v_{N-1} to v_0. The two-dimensional rectangular lattice graph in Fig. 1(b) represents a digital image. Each pixel corresponds to a node that is connected to the nodes that index its four adjacent pixels. This relation is symmetric, hence all edges are undirected. If no additional information is available, all edge weights a_n and b_m are equal, with a possible exception of boundary nodes which may have directed edges and different



(a) Time series



(b) Digital image



(c) Weather stations across the U.S.

Fig. 1. Graph representations for different datasets.

edge weights due to imposed boundary conditions [9, 11]. Edge weights a_n and b_m, however, can be optimized for better image representation, as we show in Section 4. Finally, the graph in Fig. 1(c) represents temperature measurements from weather stations located across the United States. We represent the relations between temperature measurements of different sensors by geodesic distances between these sensors, and connect each node to nodes that correspond to several most closely located sensors. We discuss the latter example in more detail in Section 4.

Graph filters. In DSP, filters are systems that take a signal as an input and produce another signal as the output. In DSP_G, we define an equivalent concept of filters for the processing of graph signals. Given graph signals $\mathbf{s}$ indexed by a graph $G = (\mathcal{V}, \mathbf{A})$, the basic building block for graph filters on G is a *graph shift* that replaces each signal coefficient s_n indexed by node v_n with a linear combination of coefficients at other nodes weighted proportionally to the degree of their relation:

$$\tilde{s}_n = \sum_{m=0}^{N-1} \mathbf{A}_{n,m} s_m \Leftrightarrow \tilde{\mathbf{s}} = \mathbf{A}\mathbf{s}. \quad (2)$$

Similarly to traditional linear, time-invariant DSP theory, we consider linear, shift-invariant filters for graph signals. As demonstrated in [8], any linear, shift-invariant *graph filter* is necessarily a matrix polynomial in the adjacency matrix $\mathbf{A}$ of the form

$$h(\mathbf{A}) = h_0\,\mathbf{I} + h_1 \mathbf{A} + \ldots + h_L \mathbf{A}^L, \quad (3)$$

with possibly complex coefficients $h_\ell \in \mathbb{C}$. Furthermore, any graph filter (3) can be represented by at most N coefficients; and if it is invertible, its inverse also is a matrix polynomial in $\mathbf{A}$ of the form (3).

3. GRAPH FOURIER TRANSFORM

Having defined the concepts of graph signals and filters, we now discuss the spectral decomposition, spectrum, and Fourier transform for graph signals. These concepts are related to the Jordan normal form of the adjacency matrix $\mathbf{A}$; this topic is discussed in [12].

Spectral decomposition. The spectral decomposition of a signal space $\mathcal{S}$ corresponds to the identification of subspaces $\mathcal{S}_k$, $0 \le k < K$, of $\mathcal{S}$ that are *invariant* to filtering. For a signal $\mathbf{s}_k \in \mathcal{S}_k$ from a subspace $\mathcal{S}_k$, the output $\widetilde{\mathbf{s}}_k = h(\mathbf{A})\mathbf{s}_k$ of any filter $h(\mathbf{A})$ is also a signal from the same subspace $\mathcal{S}_k$. The signal $\mathbf{s} \in \mathcal{S}$ then can be represented as

$$\mathbf{s} = \mathbf{s}_0 + \mathbf{s}_1 + \ldots + \mathbf{s}_{K-1}, \quad (4)$$

where $\mathbf{s}_k \in \mathcal{S}_k$. Decomposition (4) is uniquely determined for every graph signal $\mathbf{s} \in \mathcal{S}$ if and only if 1) invariant subspaces $\mathcal{S}_k$ have zero intersection, i.e., $\mathcal{S}_k \cap \mathcal{S}_m = \{0\}$ for any $k \neq m$; 2) $\dim \mathcal{S}_0 + \ldots + \dim \mathcal{S}_{K-1} = \dim \mathcal{S} = N$; and 3) each $\mathcal{S}_k$ is *irreducible*, i.e., it cannot be decomposed into smaller invariant subspaces.

Consider the Jordan decomposition of $\mathbf{A}$:

$$\mathbf{A} = \mathbf{V}\,\mathbf{J}\,\mathbf{V}^{-1}. \quad (5)$$

Here, $\mathbf{J}$ is the Jordan normal form and $\mathbf{V}$ is the matrix of generalized eigenvectors. Let λ_m denote an arbitrary eigenvalue of $\mathbf{A}$, and $\mathbf{v}_{m,0}, \ldots, \mathbf{v}_{m,r}$ denote a Jordan chain of generalized eigenvectors corresponding to this eigenvalue. Then $\mathcal{S}_m = \mathbf{span}\{\mathbf{v}_{m,0}, \ldots, \mathbf{v}_{m,r}\}$ is a vector subspace of $\mathcal{S}$ with this Jordan chain as its basis. Any signal $\mathbf{s}_m \in \mathcal{S}_m$ has a unique expansion in this basis:

$$\begin{aligned} \mathbf{s}_m &= \hat{s}_{m,0}\mathbf{v}_{m,0} + \ldots + \hat{s}_{m,r}\mathbf{v}_{m,r} \\ &= \mathbf{V}_m \begin{pmatrix} \hat{s}_{m,0} & \ldots & \hat{s}_{m,r} \end{pmatrix}^T. \end{aligned}$$

Here, $\mathbf{V}_m$ is a matrix with columns given by $\mathbf{v}_{m,0}, \ldots, \mathbf{v}_{m,r}$. It follows from (5) that if we apply the graph shift (2) to $\mathbf{s}_m$, the output $\widehat{\mathbf{s}}_m = \mathbf{A}\mathbf{s}_m$ also belongs to the same subspace:

$$\begin{aligned} \widehat{\mathbf{s}}_m &= \mathbf{A}\mathbf{s}_m = \mathbf{A}\,\mathbf{V}_m \begin{pmatrix} \hat{s}_{m,0} & \ldots & \hat{s}_{m,r} \end{pmatrix}^T \\ &= \mathbf{V}_m\ \mathbf{J}(\lambda_m)\ \begin{pmatrix} \hat{s}_{m,d,0} & \ldots & \hat{s}_{m,r} \end{pmatrix}^T \\ &= \mathbf{V}_m \begin{pmatrix} \lambda_m \hat{s}_{m,0} + \hat{s}_{m,1} \\ \vdots \\ \lambda_m \hat{s}_{m,r} + \hat{s}_{m,r} \\ \lambda_m \hat{s}_{m,r} \end{pmatrix}, \end{aligned} \quad (6)$$

where $\mathbf{J}(\lambda_m)$ is a Jordan block corresponding to λ_m. Thus, the subspace $\mathcal{S}_m$ is invariant to shifting. It also follows from (6) that $\mathcal{S}_m$ is irreducible.

Furthermore, any graph filter (3) can be written as

$$\begin{aligned} h(\mathbf{A}) &= \sum_{\ell=0}^{L} h_\ell (\mathbf{V}\,\mathbf{J}\,\mathbf{V}^{-1})^\ell \ = \sum_{\ell=0}^{L} h_\ell\, \mathbf{V}\,\mathbf{J}^\ell\, \mathbf{V}^{-1} \\ &= \mathbf{V}\,\Big(\sum_{\ell=0}^{L} h_\ell\, \mathbf{J}^\ell\Big)\,\mathbf{V}^{-1} = \mathbf{V}\,h(\mathbf{J})\,\mathbf{V}^{-1}. \end{aligned} \quad (7)$$

Similarly to (6), filtering a signal $\mathbf{s}_m \in \mathcal{S}_m$ produces an output from the same subspace:

$$\begin{aligned} \widehat{\mathbf{s}}_m &= h(\mathbf{A})\mathbf{s}_m = h(\mathbf{A})\,\mathbf{V}_m \begin{pmatrix} \hat{s}_{m,0} & \ldots & \hat{s}_{m,r} \end{pmatrix}^T \\ &= \mathbf{V}_m \left[h(\mathbf{J}(\lambda_m)) \begin{pmatrix} \hat{s}_{m,0} \\ \vdots \\ \hat{s}_{m,r} \end{pmatrix} \right]. \end{aligned} \quad (8)$$

Since $\mathbf{A}$ has exactly N generalized eigenvectors, and they are linearly independent, all subspaces $\mathcal{S}_m$ are irreducible, have zero intersections, and their dimensions add up to N. Hence, they yield the *spectral decomposition* of the signal space $\mathcal{S}$.

Graph Fourier transform. The spectral decomposition of $\mathcal{S}$ expands each signal $\mathbf{s} \in \mathcal{S}$ in the basis given by the union of all generalized eigenvectors. This expansion can be written as

$$\mathbf{s} = \mathbf{V}\,\widehat{\mathbf{s}}, \tag{9}$$

where the vector of expansion coefficients is given by

$$\widehat{\mathbf{s}} = \mathbf{V}^{-1}\,\mathbf{s}. \tag{10}$$

We call the basis of generalized eigenvectors the *graph Fourier basis*, and the expansion (10) the *graph Fourier transform*. We denote the graph Fourier transform matrix as

$$\mathbf{F} = \mathbf{V}^{-1}\,. \tag{11}$$

Following the conventions of classical DSP, we call the coefficients $\hat{s}_n$ in (10) the *spectrum* of a signal $\mathbf{s}$. The *inverse graph Fourier transform* is given by (9); it reconstructs the signal from its spectrum.

Discussion. The connection (10) between the graph Fourier transform and the Jordan decomposition (5) highlights some desirable properties of representation graphs. For instance, graphs with orthogonal or unitary matrices $\mathbf{V}$ of generalized eigenvectors, including all undirected graphs, have orthogonal graph Fourier transforms: $\mathbf{F}^{-1} = \mathbf{F}^H$. This property has significant practical importance, since, for example, orthogonal transforms are well-suited for efficient signal representation, as we demonstrate in Section 4.

Also, observe that the definition (10) of graph Fourier transform is consistent with the traditional DSP theory. As we discussed in Section 2, finite discrete periodic time series are indexed by the directed graph in Fig. 1(a). The corresponding adjacency matrix is the $N \times N$ circulant matrix $\mathbf{A}$ for which $\mathbf{A}_{n,m}$ is 1 if $n - m = 1 \mod N$ and 0 otherwise. Its eigendecomposition (and hence, Jordan decomposition) is

$$\mathbf{A} = \mathbf{DFT}_N^H \begin{pmatrix} e^{-j\frac{2\pi\cdot 0}{N}} & & \\ & \ddots & \\ & & e^{-j\frac{2\pi\cdot(N-1)}{N}} \end{pmatrix} \mathbf{DFT}_N,$$

where $\mathbf{DFT}_N$ denotes the discrete Fourier transform matrix. Thus, as expected, the graph Fourier transform is $\mathbf{F} = \mathbf{DFT}_N$.

This example also illustrates an important difference between our proposed definition (10) of graph Fourier transform and a similar definition of graph Fourier transform used in [6, 7]. The latter one uses the eigenbasis of the Lagrangian matrix for the indexing graph, and assumes that the graph is undirected and has non-negative real weights. Thus, it is not applicable to traditional time DSP and the derivation of the standard discrete Fourier transform, or other datasets indexed by directed graphs, such as a set of documents linked by references [13], or graphs with negative weights [14].

In time DSP, the concepts of spectrum and Fourier transform have natural, physical interpretations. In $\mathrm{DSP_G}$these concepts may have drastically different and not immediately obvious interpretations. For example, if a graph signal represents measurements from multiple sensors and the indexing graph reflects their proximity in some metric (such as time, space, or geodesic distance), then filtering this graph signal linearly recombines related measurements. It can be viewed as a graph form of regression analysis with constant coefficients. The graph Fourier transform then decomposes the signal over equilibrium points of this regression. Alternatively, a graph signal may contain a characteristic of a social network, such as an opinion or a preference of individuals, and the indexing graph represents this social network. In this case filtering can be viewed as the diffusion of this characteristic along existing communication channels in the network, and the graph Fourier transform can be interpreted as the representation of the signal in terms of stable, unchangeable opinions or preferences.

4. SPECTRAL REPRESENTATION OF GRAPH SIGNALS

Efficient signal representation is required in multiple areas of signal processing, such as storage, compression, and transmission. Some widely-used techniques are based on expanding signals into suitable bases with the expectation that most information about the signal is captured with few basis functions. For example, some image compression standards, e.g. JPEG and JPEG 2000, expand images into cosine or wavelet bases, which yield high-quality approximations for smooth images [15].

If a representation basis corresponds to a Fourier basis in some signal model, we say that signals are sparse in the frequency domain if their spectrum is dominated by only a few frequencies, i.e. they are accurately approximated by a few Fourier basis functions. As we demonstrate in the following examples, graphs signals can be sparse in their respective frequency domain, which makes their Fourier bases useful for efficient signal representation and compression[1]. For simplicity of the discussion and calculations, we consider signals represented by undirected graphs. In this case, as discussed in Section 3, corresponding graph Fourier transforms are orthogonal matrices, and the Fourier bases are orthogonal. The advantage of an orthogonal basis is that selecting spectrum components with largest magnitudes minimizes the approximation error in the least-squares sense. The approach discussed here also extends to directed graphs with general Fourier bases.

Compression algorithm. Given an orthogonal graph Fourier basis, we compress a graph signal $\mathbf{s}$ by keeping only C of its spectrum coefficients (10) $\hat{\mathbf{s}}_n$ that have largest magnitudes. Without loss of generality, assume that $|\hat{\mathbf{s}}_0| \geq |\hat{\mathbf{s}}_1| \geq \ldots \geq |\hat{\mathbf{s}}_{N-1}|$. Then the signal reconstructed after compression is

$$\tilde{\mathbf{s}} = \mathbf{F}^{-1}\left(\hat{\mathbf{s}}_0, \ldots, \hat{\mathbf{s}}_{C-1}, 0, \ldots, 0\right)^T. \tag{12}$$

If for $0 \leq k < K$, signals $\mathbf{s}_k$ are approximated as $\tilde{\mathbf{s}}_k$, each with C largest-magnitude coefficients of their spectrum, we calculate the average approximation error as

$$err(C) = \frac{\sum_{k=0}^{K-1} ||\tilde{\mathbf{s}}_k - \mathbf{s}_k||_2}{\sum_{k=0}^{K-1} ||\mathbf{s}_k||_2}. \tag{13}$$

Image compression. As the first example, we consider the graph representation of images using the graph in Fig. 1(b). As follows from the figure, we make a simplifying assumptions that edge weights depend only on their row or column, as shown. Then, given a specific image, we determine the edge weights a_n and b_m by minimizing the distortion caused by the graph shift:

$$\left\{ \begin{array}{ccc} a_0, & \ldots, & a_{N-2}, \\ b_0, & \ldots, & b_{M-2} \end{array} \right\} = \underset{a_n, b_m \in \mathbb{C}}{\mathbf{argmin}} ||\mathbf{A}\mathbf{s} - \mathbf{s}||_2.$$

Here, $\mathbf{s}$ is a vectorized representation of the image. As demonstrated in [17], this is a least-squares minimization problem.

For the evaluation of this image representation technique, we consider $K = 4$ images shown in Fig. 2, all of size 256×256. Table 1 shows average errors (13) obtained for different fractions of spectrum coefficients used for approximation. For comparison, we also consider three standard orthogonal transform: the discrete Fourier (DFT), cosine (DCT), and wavelet (DWT) transforms. As can be observed from the results, the graph Fourier transform leads to smallest errors regardless of the number of spectrum coefficients used for approximation.

[1]Eigenvectors of the graph Laplacian matrix have also been considered for the compression of graph signals [16]. In contrast, our approach use the generalized eigenvectors of the graph adjacency matrix.

Fig. 2. Test images.

Transform	Fraction of coefficients used (C/N)					
	2%	5%	10%	15%	20%	30%
Graph FT	10%	5%	2%	1%	1%	0.5%
DFT	14%	8%	5%	3%	2%	1%
DCT	12%	6%	3%	2%	2%	1%
DWT	12%	6%	3%	2%	1%	1%

Table 1. Average approximation errors for digital images.

Compression of sensor measurements. Another example we consider is the representation of measurements from a non-uniformly distributed sensor field. In particular, we consider a set of daily temperature measurements from weather stations located near 150 major US cities [18]. Data from each sensor is a separate time series; however, compressing each time series separately requires buffering measurements from multiple days before they can be compressed for storage or transmission. Instead, we consider graph signals constructed from daily snapshots of all 150 measurements. We construct the representation graph, shown in Fig. 1(c), using geographical distances between sensors. Each sensor corresponds to a node v_n, $0 \leq n < 150$, and is connected to 8 nearest sensors with undirected edges weighted by the normalized inverse exponents of the squared distances: if d_{nm} denotes the distance between the nth and mth sensors and v_m is connected to v_n, then

$$\mathbf{A}_{n,m} = \frac{e^{-d_{nm}^2}}{\sqrt{\sum_{k \in \mathcal{N}_n} e^{-d_{nk}^2} \sum_{k \in \mathcal{N}_m} e^{-d_{mk}^2}}}. \tag{14}$$

We consider a full year of 365 daily measurements from each sensor, and evaluate the representation efficiency by calculating the average approximation error (13) over $K = 365$ days. For comparison, we also consider compressing each separate time series of measurements from each station with DFT and DCT, and calculating average errors over $K = 150$ stations. The results are shown in Table 2. The graph Fourier transform yields smallest errors for all fractions of spectrum coefficients used for approximation.

Transform	Fraction of coefficients used (C/N)					
	2%	5%	10%	15%	20%	30%
Graph FT	17%	9%	5%	4%	3%	1%
DFT	18%	14%	10%	7%	5%	3%
DCT	17%	12%	8%	5%	4%	2%

Table 2. Average approximation errors for temperature data.

5. CONCLUSIONS

We have proposed a framework for discrete signal processing of signals indexed by graphs. We discussed the notions of graph signals and filters, and defined the concepts of spectral decomposition, spectrum, and Fourier transform for graph signals. We identified their relation to the Jordan decomposition of the adjacency matrices of representation graphs. As a potential application of the graph Fourier transform, we demonstrated that graph signals can be sparsely represented in their frequency domain, and thus efficiently approximated using a few Fourier basis functions with little approximation error.

6. REFERENCES

[1] M. Jackson, *Social and Economic Networks*, Princeton Univ., 2008.

[2] M. Newman, *Networks: An Introduction*, Oxford Univ. Press, 2010.

[3] S. L. Lauritzen, *Graphical Models*, Oxford Univ. Press, 1996.

[4] M. I. Jordan, "Graphical models," *Statistical Science (Special Issue on Bayesian Statistics)*, vol. 19, no. 1, pp. 140–155, 2004.

[5] U. Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[6] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *J. Appl. Comp. Harm. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.

[7] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Trans. Signal Proc.*, vol. 60, no. 6, pp. 2786–2799, 2012.

[8] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Proc.*, accepted.

[9] M. Püschel and J. M. F. Moura, "Algebraic signal processing theory," `http://arxiv.org/abs/cs.IT/0612077`.

[10] M. Püschel and J. M. F. Moura, "Algebraic signal processing theory: Foundation and 1-D time," *IEEE Trans. Signal Proc.*, vol. 56, no. 8, pp. 3572–3585, 2008.

[11] M. Püschel and J. M. F. Moura, "Algebraic signal processing theory: 1-D space," *IEEE Trans. Signal Proc.*, vol. 56, no. 8, pp. 3586–3599, 2008.

[12] P. Lancaster and M. Tismenetsky, *The Theory of Matrices*, Academic Press, 2nd edition, 1985.

[13] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 U.S. election: Divided they blog," in *LinkKDD*, 2005.

[14] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media," in *Proc. of SIGCHI*, 2010, pp. 1361–1370.

[15] A. Bovik, *Handbook of Image and Video Processing*, Academic Press, 2nd edition, 2005.

[16] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *Proc. ICASSP*, 2012, pp. 3921–3924.

[17] A. Sandryhaila and J. M. F. Moura, "Nearest-neighbor image model," in *Proc. ICIP*, 2012, to appear.

[18] "National climatic data center," 2011, `ftp://ftp.ncdc.noaa.gov/pub/data/gsod`.