Graph Signal Processing – A Probabilistic Framework

Cha Zhang, *Senior Member, IEEE,* Dinei Florêncio, *Senior Member, IEEE* and Philip A. Chou *Fellow, IEEE*

***Abstract*—This theoretical paper aims to provide a probabilistic framework for graph signal processing. By modeling signals on graphs as Gaussian Markov Random Fields, we present numerous important aspects of graph signal processing, including graph construction, graph transform, graph downsampling, graph prediction, and graph-based regularization, from a probabilistic point of view. As examples, we discuss a number of methods for constructing graphs based on statistics from input data sets; we show that the graph transform is the optimal linear transform to decorrelate the signal; we describe the optimality of the Kron reduction for graph downsampling in a probabilistic sense; and we derive the optimal predictive transform coding scheme applicable to both motion prediction and intra predictive coding.**

***Index Terms*—Graph signal processing, Gaussian Markov random field, graph transform, predictive graph transform, graph sampling, graph-based regularization**

I. Introduction

Historically, Digital Signal Processing (DSP) deals mostly with signals that exist in a continuous domain, and are then sampled to obtain a corresponding digital representation, which is then processed. Because these signals are, typically, acquired by a system with that exact and only purpose, the sampling grids are generally uniform. Thus, it is only natural that the bulk of signal processing research targets uniform grids. More recently, however, with the ever increasing reach of signal processing techniques, significant attention is being placed on signals that are either intrinsically digital (e.g., social signals, zip codes), or are sampled by a process that does not follow a regular sampling pattern. As such, many of the traditional DSP tools do not apply, thus creating a need for new tools. Graph Signal Processing (GSP), or processing signals that live on a graph (instead of on a regular sampling grid), has received a lot of attention as a promising research direction [30]. It essentially allows for a generalized "sampling grid" (the graph), and deals with the signal as samples on the graph nodes.

Many existing GSP works in recent literature attempt to bring the rich set of familiar and useful tools from DSP to graphs. For instance, graph spectral analysis is considered the counterpart of Fourier analysis in DSP, and thus well-known operations such as translation, convolution, modulation and filtering can be equivalently defined on the graph. These tools allow us to process graph signals in ways familiar to DSP researchers, and are thus invaluable to many real-world applications.

C. Zhang, D. Florêncio and P. A. Chou are with Microsoft Research, One Microsoft Way, Redmond 98052, USA. (email {chazhang,dinei,pachou}@microsoft.com).

On the other hand, historically graphs are also often the basis for probabilistic modeling. For example, graphical models such as Hidden Markov Models (HMM) have been widely used in speech signal processing [25] and bioinformatics [12]. Bayesian models on graphs are widely used to compute likelihoods or posterior probabilities for many pattern recognition problems [2]. Markov random fields have played an important role in image processing [20].

Nevertheless, the two research camps on graph signal processing do not often intersect with each other. We observe DSP researchers setting up graphs in an ad-hoc manner based on intuition, and Bayesians unaware of the powerful interpretations their probabilistic models can provide to signal processing. In this paper, we try to bridge that gap by providing a clear probabilistic explanation of some important topics in GSP, including graph construction, the graph transform, graph downsampling, graph prediction, and graph-based regularization. We believe that bridging this gap is the primary contribution of this paper, and that it helps to put the nascent field of GSP on a more solid foundation.

We start by reviewing the concept of a Gaussian Markov Random Field (GMRF) in Section II, and establishing a correspondence with a signal sitting on the graph nodes. That allows us to give graphs, including their nodes and the weights between nodes, a clear probabilistic interpretation. The analysis covers graphs with self-loops, which ensures a proper Gaussian distribution. With such a probabilistic interpretation in mind, we discuss how to construct a graph from a data set in Section III, and present three possible approaches: data-driven, intuitive model-based, and model-constrained data driven approaches. We further extend the correspondence, and show in Section IV that the well-known Graph Transform [29] is the optimal decorrelating transform of a signal obeying our GMRF model. Such a relationship leads to the important proof that a 2D discrete cosine transform (DCT) is one of the optimal linear transforms for a particular GMRF construction for graphs on a 2D regular grid, which has implications for image/video coding. In Section V, we discuss graph downsampling, and show that the well-known Kron reduction [11] is just a marginalization of the GMRF model. Furthermore, we study graph prediction in Section VI, and draw important conclusions on the optimal scheme for predictive transform coding, which gives new insights to both motion estimation and intra predictive coding. In Section VII, we show the probabilistic interpretation of regularization in GSP, which is essentially a prior probability model for the given application. Finally, conclusions are given in Section VIII.

II. The Gaussian Markov Random Field Model

A. The Gaussian Markov Random Field

We first introduce the concept of a GMRF model [27]. A GMRF is a restrictive multivariate Gaussian distribution that satisfies additional conditional independence assumptions. We often use a bi-directed graph [13] $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the conditional independence assumptions, where $\mathcal{V}$ represents the set of nodes in the graph, and $\mathcal{E}$ represents the set of edges.

Formally, a random vector $\mathbf{x} = (x_1, \cdots, x_n)^T$ is called a GMRF with respect to the bi-directed graph $\mathcal{G} = (\mathcal{V} = \{1, \cdots, n\}, \mathcal{E})$ with mean vector μ and a symmetric precision matrix $\mathbf{Q} > 0$ (positive definite), if and only if its density has the form [27]:

$$p(\mathbf{x}) = (2\pi)^{-\frac{n}{2}} |\mathbf{Q}|^{\frac{1}{2}} \exp\Big(-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{Q}(\mathbf{x}-\mu)\Big), \quad (1)$$

$$\text{and } Q_{ij} \neq 0 \Leftrightarrow \{i,j\} \in \mathcal{E} \text{ for all } i \neq j. \quad (2)$$

From the above definition, it is clear that a GMRF $\mathbf{x}$ is a multivariate Gaussian distribution with mean vector μ whose covariance matrix $\mathbf{\Sigma}$ is the inverse of $\mathbf{Q}$. A property of the precision matrix is that its elements have conditional interpretations [27]:

$$E(x_i|\mathbf{x}_{-i}) = \mu_i - \frac{1}{Q_{ii}} \sum_{j:j\sim i} Q_{ij}(x_j - \mu_j), \quad (3)$$

$$Prec(x_i|\mathbf{x}_{-i}) = Q_{ii}, \quad (4)$$

$$Corr(x_i, x_j|\mathbf{x}_{-ij}) = -\frac{Q_{ij}}{\sqrt{Q_{ii}Q_{jj}}}, i \neq j, \quad (5)$$

where $\mathbf{x}_{-i}$ represents all elements in $\mathbf{x}$ except x_i; $j : j \sim i$ represents all nodes j that are neighbors of i in the graph. The diagonal elements of $\mathbf{Q}$ are the conditional precisions of x_i given all other elements; while the off-diagonal elements, with a proper scaling, provide information about the conditional correlation between x_i and x_j given all other variables.

B. Equivalence of Weighted Graphs to GMRFs

Graph signal processing [30] begins with a *weighted* bi-directed graph $\langle \mathcal{G}, \mathbf{W} \rangle = \langle (\mathcal{V}, \mathcal{E}), \mathbf{W} \rangle$, where $\mathcal{V}$ is a of nodes, $\mathcal{E}$ is a set of edges, and $\mathbf{W}$ is a symmetric non-negative matrix of weights such that

$$W_{ij} > 0 \text{ if } \{i,j\} \in \mathcal{E} \text{ and } W_{ij} = 0 \text{ otherwise.} \quad (6)$$

In this section, we show that there is a one-to-one mapping from the set of symmetric non-negative weight matrices $\mathbf{W}$ satisfying (6) to the set of symmetric positive semi-definite precision matrices $\mathbf{Q}$ satisfying (2). This one-to-one mapping will establish the equivalence between the weighted bi-directed graphs used in graph signal processing to GMRFs.

Let us start with a GMRF model in the form of Eq. (1). Assume the signal is zero-mean, thus $\mu = \mathbf{0}$. We may construct a bi-directed graph $\mathcal{G} = (\mathcal{V} = \{1, \cdots, n\}, \mathcal{E})$, where each element of the random vector will form a node in the graph. When $i \neq j$, an edge between node i and j is created if and only if $Q_{ij} \neq 0$. In addition, we will add some self loops (edges that connect nodes to themselves) to the graph, as shown in Fig. 1. (Often a graph with self-loops is called a *pseudograph*, while a graph without self-loops is called a *simple graph*.)

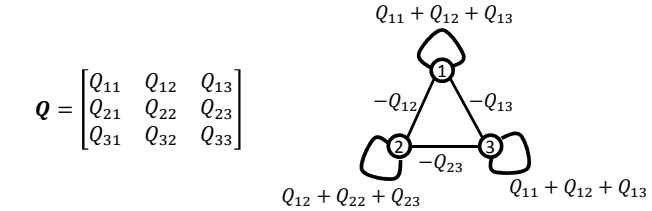

Fig. 1. Mapping between a precision matrix and a bi-directed graph that contains self-loops.

We now define a mapping from $n \times n$ matrices $\mathbf{Q}$ to $n \times n$ matrices $\mathbf{W}$. Specifically, let

$$W_{ij} = -Q_{ij}, \text{for all } i \neq j \quad (7)$$

and

$$W_{ii} = \sum_{j=1}^{n} Q_{ij}, \text{for all } i. \quad (8)$$

An example of this weighting for a three-node graph is given in Fig. 1.

A reverse mapping can be defined similarly. Specifically,

$$\hat{Q}_{ij} = -W_{ij}, \text{for all } i \neq j \quad (9)$$

and

$$\hat{Q}_{ii} = \sum_{j=1}^{n} W_{ij} \text{for all } i. \quad (10)$$

It is easy to verify that $\mathbf{Q} = \hat{\mathbf{Q}}$: clearly $Q_{ij} = -W_{ij} = \hat{Q}_{ij}$ for all $i \neq j$, and $Q_{ii} = W_{ii} - \sum_{j \neq i} Q_{ij} = \sum_j W_{ij} = \hat{Q}_{ii}$ for all i. Hence, the mapping is invertible and puts the set of $n \times n$ matrices $\mathbf{Q}$ in 1-1 correspondence with the set of $n \times n$ matrices $\mathbf{W}$.

Next consider the subset of matrices $\mathbf{W}$ that are symmetric and non-negative, satisfying (6). We now show that when such a matrix $\mathbf{W}$ is mapped to a matrix $\hat{\mathbf{Q}}$ via (9) and (10), the resulting matrix $\hat{\mathbf{Q}}$ is positive semi-definite. To see that, index the edges $\mathcal{E}$ by $e \in \{1, \ldots, |\mathcal{E}|\}$ and denote the eth edge by $\{i_e, j_e\}$, where $i_e \leq j_e$ are the vertices connected by edge e. Then construct the $|\mathcal{E}| \times |\mathcal{V}|$ matrix $\mathbf{R}$ whose eth row $\mathbf{r}_e^T = [R_{ei}]$ is given for $i_e < j_e$ by

$$R_{ei} = \begin{cases} \sqrt{W_{i_e j_e}} & \text{if } i = i_e \\ -\sqrt{W_{i_e j_e}} & \text{if } i = j_e \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

and for $i_e = j_e$ by

$$R_{ei} = \begin{cases} \sqrt{W_{i_e i_e}} & \text{if } i = i_e \\ 0 & \text{otherwise} \end{cases}. \quad (12)$$

Note that this construction is possible only if $\mathbf{W}$ is non-negative. Now it is clear that

$$\mathbf{R}^T\mathbf{R} = \sum_e \mathbf{r}_e^T \mathbf{r}_e \quad (13)$$

is the sum of symmetric $n \times n$ matrices $\mathbf{Q}^e = \mathbf{r}_e^T \mathbf{r}_e$, where for e such that $i_e < j_e$,

$$Q_{ij}^e = \begin{cases} W_{i_e j_e} & \text{if } i = i_e \text{ and } j = i_e \\ -W_{i_e j_e} & \text{if } i = i_e \text{ and } j = j_e \\ -W_{i_e j_e} & \text{if } i = j_e \text{ and } j = i_e \\ W_{i_e j_e} & \text{if } i = j_e \text{ and } j = j_e \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

and for e such that $i_e = j_e$,

$$Q_{ij}^e = \begin{cases} W_{i_e i_e} & \text{if } i = i_e \text{ and } j = i_e \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

Thus it can be seen from (9) and (10) that $\mathbf{R}^T\mathbf{R} = \hat{\mathbf{Q}}$, whence $\hat{\mathbf{Q}}$ must be positive semi-definite since $\mathbf{x}^T\mathbf{R}^T\mathbf{R}\mathbf{x} = ||\mathbf{R}\mathbf{x}||^2 \geq 0$ or alternatively

$$\begin{aligned} \mathbf{x}^T\hat{\mathbf{Q}}\mathbf{x} &= \sum_e \mathbf{x}^T\mathbf{Q}^e\mathbf{x} & (16) \\ &= \sum_i W_{ii}x_i^2 + \sum_{i<j} W_{ij}(x_i - x_j)^2 \geq 0 & (17) \end{aligned}$$

for any vector $\mathbf{x}$. Hence we have shown that if $\mathbf{W}$ is non-negative then $\hat{\mathbf{Q}}$ is positive semi-definite.

In light of the 1-1 correspondence between $\mathbf{W}$ and $\mathbf{Q}$ (= $\hat{\mathbf{Q}}$), this proves that there is an injective mapping from the set of symmetric non-negative weight matrices $\mathbf{W}$ satisfying (6) to the set of symmetric positive semi-definite precision matrices $\mathbf{Q}$ satisfying (2).

In general positive semi-definiteness of $\mathbf{Q}$ is required (as opposed to positive definiteness), since in general there are non-negative weight matrices $\mathbf{W}$ satisfying (6) corresponding to singular precision matrices $\mathbf{Q}$ satisfying (2). In particular, if $\mathbf{W}$ has no self-loops, i.e., if $W_{ii} = 0$ for all i, then it can be seen from (8) that all rows of $\mathbf{Q}$ sum to zero, hence $\mathbf{Q}$ is rank-deficient. We will show in Section II-D that the converse is also true: if $\mathbf{W}$ has at least one self-loop in every connected component, i.e., if $W_{ii} > 0$ for any i in every connected component, then the precision matrix has full rank. (This can also be proved from a different direction using [11, Lemma 3.1].) Thus in fact we can prove that there is a 1-1 correspondence between symmetric *positive definite* precision matrices $\mathbf{Q}$ that satisfy (2) and non-negative weight matrices $\mathbf{W}$ that both satisfy (6) and are sufficiently loopy: $W_{ii} > 0$ for at least one i in each connected component of $\mathcal{G}$.

In summary, each weighted bi-directed graph $\langle \mathcal{G}, \mathbf{W} \rangle$ used in graph signal processing corresponds uniquely to a zero-mean *intrinsic* GMRF with respect to $\mathcal{G}$. An intrinsic GMRF, as defined in the next section, is a generalization of a GMRF whose precision matrix may not be invertible. We need this generalization, since in many applications, the underlying graph contains no self-loops.

C. The Intrinsic Gaussian Markov Random Field

In this section, we define the intrinsic GMRF (IGMRF). Let $\mathbf{Q}$ be an $n \times n$ symmetric positive *semi-definite* matrix with rank $n - k$, which may be less than n. A vector $\mathbf{x} = (x_1, \cdots, x_n)^T$ is an *intrinsic GMRF* of order $k \geq 0$ with parameters $(\mu, \mathbf{Q})$ if it has density

$$p(\mathbf{x}) = (2\pi)^{-\frac{n-k}{2}} (|\mathbf{Q}|^*)^{\frac{1}{2}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \mu)^T\mathbf{Q}(\mathbf{x} - \mu)\Big), \tag{18}$$

where $|\cdot|^*$ denotes the generalized determinant (the product of the non-zero eigenvalues). An intrinsic GRMF of order $k > 0$ is also known as an *improper GMRF* of rank $n - k$.

For an improper GMRF, the density (18) is not integrable with respect to Lebesgue measure. Nevertheless, the density (18) well-defines a measure $P(X) = \int_X p(\mathbf{x})d\mathbf{x}$ on all Borel sets X in the sigma-algebra $\sigma(\mathbb{R}^n)$ on $\mathbb{R}^n$. This measure is the product of a Gaussian probability measure and Lebesgue measure. To see this, diagonalize $\mathbf{Q} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix} = \mathbf{V}_1\mathbf{\Lambda}_1\mathbf{V}_1^T, \tag{19}$$

where $\mathbf{V}$ is an orthonormal matrix whose columns are eigenvectors of $\mathbf{Q}$, $\mathbf{\Lambda}$ is the diagonal matrix of corresponding non-negative eigenvalues (without loss of generality sorted from highest to lowest), and $\mathbf{\Lambda}_1$ is the submatrix of the $n-k$ *positive* eigenvalues. Then with the change of variables

$$\mathbf{u} = \mathbf{V}^T\mathbf{x}, \tag{20}$$

we have

$$\mathbf{x} = \mathbf{V}\mathbf{u} = \begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} = \mathbf{V}_1\mathbf{u}_1 + \mathbf{V}_2\mathbf{u}_2. \tag{21}$$

Now assuming the transformed set $U = \mathbf{V}^T X$ is a "rectangle" $U = U_1 \times U_2$ in the new coordinate system, where U_i is is a measurable set in the column space of $\mathbf{V}_i$, $i = 1, 2$, we have

$$\begin{aligned} &P(X) \\ &= \int_X p(\mathbf{x})d\mathbf{x} & (22) \\ &= \det(\mathbf{V}) \int_{\mathbf{V}^T X} p(\mathbf{V}\mathbf{u})d\mathbf{u} & (23) \\ &= \int_{U_1 \times U_2} (2\pi)^{-\frac{n-k}{2}} (|\mathbf{Q}|^*)^{\frac{1}{2}} & (24) \\ &\quad \exp\Big(-\frac{1}{2}(\mathbf{u} - \mathbf{V}^T\mu)^T\mathbf{V}^T\mathbf{Q}\mathbf{V}(\mathbf{u} - \mathbf{V}^T\mu)\Big)d\mathbf{u} \\ &= \int_{U_2} d\mathbf{u}_2 \times \int_{U_1} (2\pi)^{-\frac{n-k}{2}} (|\mathbf{Q}|^*)^{\frac{1}{2}} & (25) \\ &\quad \exp\Big(-\frac{1}{2}(\mathbf{u}_1 - \mathbf{V}_1^T\mu)^T\mathbf{V}_1^T\mathbf{Q}\mathbf{V}_1(\mathbf{u}_1 - \mathbf{V}_1^T\mu)\Big)d\mathbf{u}_1. \end{aligned}$$

This measure on rectangles extends to a measure on arbitrary measurable sets $X \in \sigma(\mathbb{R})$ by the usual technique of approaching X from below by a sequence of unions of rectangles and taking limits [15].

Thus we see that the measure of an improper GMRF of rank $n - k$ with parameters $(\mu, \mathbf{Q})$ is the product of 1) a proper Gaussian measure on $\mathbb{R}^{n-k}$ with mean $\mathbf{V}_1^T\mu$ and precision $\mathbf{\Lambda}_1 = \mathbf{V}_1^T\mathbf{Q}\mathbf{V}_1$, and 2) an improper Lebesgue measure on $\mathbb{R}^k$. Moreover, it can be seen from (21) that an improper GMRF $\mathbf{x}$ is the sum of a Gaussian random vector $\mathbf{V}_1\mathbf{u}_1$ and an indeterminate (non-random, unknown) vector $\mathbf{V}_2\mathbf{u}_2$. The former lies in $\text{span}(\mathbf{V}_1) = \text{range}(\mathbf{Q})$, the subspace spanned

by the columns of $\mathbf{V}_1$, while the latter lies in the orthogonal subspace $\mathrm{span}(\mathbf{V}_2) = \mathrm{nullspace}(\mathbf{Q})$.

Since the indeterminate vector lies in $\mathrm{span}(\mathbf{V}_2)$, it can be killed by any linear combination $A\mathbf{x}$ where the columns of A^T are orthogonal to $\mathrm{span}(\mathbf{V}_2)$ (or equivalently, lie in $\mathrm{span}(\mathbf{V}_1)$), in which case $A\mathbf{V}_2 = 0$ and so

$$A\mathbf{x} = A\mathbf{V}_1\mathbf{u}_1 + A\mathbf{V}_2\mathbf{u}_2 = A\mathbf{V}_1\mathbf{u}_1. \tag{26}$$

Moreover, as a linear combination of the Gaussian random variables $\mathbf{u}_1$, $A\mathbf{x}$ is also Gaussian with mean $A\mathbf{V}_1\mathbf{V}_1^T\mu$ and covariance $A\mathbf{V}_1\mathbf{\Lambda}_1^{-1}\mathbf{V}_1^T A^T$.

As a special case, if $\mathrm{nullspace}(\mathbf{Q}) = \mathrm{span}(\mathbf{V}_2) = \{\alpha\mathbf{1}\}$, i.e., is the space spanned by the vector of all ones, then any n-vector $\mathbf{a}$ whose elements sum to zero is perpendicular to $\mathrm{nullspace}(\mathbf{Q})$, and hence the linear combination $\mathbf{a}^T\mathbf{x}$ is Gaussian. In particular, any difference of elements in $\mathbf{x}$, say $x_i - x_j$, and any linear combination of such differences, and any collection of such combinations, such as $\mathbf{Q}\mathbf{x}$ or $\mathbf{Q}^k\mathbf{x}$, is Gaussian. In the next section we show that this case is canonical.

D. IGMRF of First Order

In this section, we show that the IGMRF of first order is canonical, and we derive further results for this special case.

To see that the IGMRF of first order is canonical, observe from spectral graph theory [6, Lemma 1.7(iv)] that if $\mathcal{G}$ is loopless, then the rank k of $\mathrm{nullspace}(\mathbf{Q})$ is equal to the number of connected components of $\mathcal{G}$. This implies on the one hand, if $\mathcal{G}$ is connected then $k = 1$, corresponding to an IGMRF of first order (i.e., the rank of $\mathbf{Q}$ is $n-1$). On the other hand, if $\mathcal{G}$ is not connected then it can be decomposed into k connected components, corresponding to a collection of k independent IGMRFs each of first order. In this sense the IGMRF of first order is canonical; IGMRFs of greater order need not be considered.

We henceforth consider only the case where $\mathcal{G}$ is connected. Our results extend in the obvious way when $\mathcal{G}$ is a collection of connected components.

When $\mathcal{G}$ is connected and loopless, the null space of $\mathbf{Q}$ consists of the 1-dimensional subspace spanned by the vector of all ones. (This is the case examined at the end of the last section.) To see this, note from (8) and Fig. 1 that for $\mathcal{G}$ to be loopless (i.e., for it weight matrix to satisfy $W_{ii} = 0$ for all i) a necessary (and incidentally sufficient) condition is that all rows of $\mathbf{Q}$ must sum to zero:

$$\sum_j Q_{ij} = 0, \text{for all } i. \tag{27}$$

Also, note from the aforementioned spectral graph result that for $\mathcal{G}$ to be connected, we must have $k = 1$. Together these imply that (27) is the one and only way that $\mathbf{Q}$ can be deficient in rank. Hence the null space of $\mathbf{Q}$ consists precisely of the vectors spanned by the vector $\mathbf{1}$ of all ones.

We now show that the IGMRF of first order has the special property that its conditional distributions are proper GMRFs. To be precise, let $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2]$ be an IGMRF of first order with density $p(\mathbf{x})$ as in (18). (Here and in the sequel for readability we use the MATLAB notation $[\mathbf{x}_1; \mathbf{x}_2] = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T$.) We show that for any $\mathbf{a}$,

$$p(\mathbf{x}_1|\mathbf{x}_2 = \mathbf{a}) = \frac{p([\mathbf{x}_1; \mathbf{a}])}{\int p([\mathbf{x}_1; \mathbf{a}])d\mathbf{x}_1} \tag{28}$$

is the density of a proper GMRF as required in (1). It suffices to show that $p([\mathbf{x}_1; \mathbf{a}])$ is integrable, for if it is, then clearly (28) has a quadratic form as required in (1). But $p([\mathbf{x}_1; \mathbf{a}])$ is integrable if and only if $p([\mathbf{x}_1; \mathbf{0}])$ is integrable, since one is a non-zero multiple of the other. And $p([\mathbf{x}_1; \mathbf{0}])$ is integrable if and only if the subspace $\{[\mathbf{x}_1; \mathbf{0}]\}$ is contained entirely within $\mathrm{range}(\mathbf{Q})$, since otherwise there would exist a non-zero vector $\mathbf{v} = [\mathbf{x}_1; \mathbf{0}]$ in $\mathrm{nullspace}(\mathbf{Q})$ for which $p(\alpha\mathbf{v})$ is constant for all α.) Finally $\{[\mathbf{x}_1; \mathbf{0}]\}$ is indeed contained entirely within $\mathrm{range}(\mathbf{Q})$, since $\mathrm{nullspace}(\mathbf{Q})$ is spanned by the vector of all ones. Hence we have shown that the conditional density (28) is the density of a proper GMRF, and that the conditional distribution of an IGMRF of first order is the distribution of a proper GMRF.

At this point, it is worth observing the connection between self-loops and conditioning. From (17) it can be seen that an IGMRF $\mathbf{x} = (x_1, \ldots, x_n)$ with one or more self-loops is equivalent to an augmented IGMRF $\bar{\mathbf{x}} = (x_1, \ldots, x_n, x_{n+1})$ without self-loops, where $\bar{W}_{i,n+1} = W_{ii}$ for $i = 1, \ldots, n$ and x_{n+1} is pinned to zero. This is the reason that an IGMRF with at least one self-loop per connected component is a proper GMRF: each connected component is equivalent to an IGMRF of first order conditioned on one of its variables being equal to zero.

As a consequence of conditioning turning an IGMRF of first order into a proper GMRF, the conditional means, conditional precision matrices, and conditional covariance matrices of an IGMRF of first order are all well-defined as in (3)-(5).

An alternative way to see this is the following. Given any IGMRF, model the indeterminate vector $\mathbf{V}_2\mathbf{u}_2$ as Gaussian, where the k-dimensional vector $\mathbf{u}_2$ is Gaussian with mean $\mathbf{V}_2^T\mu$ and precision $\mathbf{\Lambda}_2 = \epsilon I_k$. Here I_k denotes the $k \times k$ identity matrix, and $\epsilon > 0$ is small. This turns the IGMRF with parameters $(\mu, \mathbf{Q})$ into an ordinary GMRF with parameters $(\mu, \mathbf{Q}_\epsilon)$, where

$$\begin{aligned} \mathbf{Q}_\epsilon &= \begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_1 & 0 \\ 0 & \epsilon I_k \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix} && (29) \\ &= \mathbf{V}_1\mathbf{\Lambda}_1\mathbf{V}_1^T + \epsilon\mathbf{V}_2\mathbf{V}_2^T. && (30) \end{aligned}$$

In the case of an IGMRF $\mathbf{x}$ of first order where $\mathbf{V}_2$ is the column vector of all ones, we have that $\epsilon\mathbf{V}_2\mathbf{V}_2^T$ is the $n \times n$ matrix of all ϵs. Hence $\mathbf{Q}_\epsilon = \mathbf{Q} + \epsilon$ and (3)-(5) become

$$E(x_i|\mathbf{x}_{-i}) = \mu_i - \frac{1}{(Q_{ii} + \epsilon)} \sum_{j:j\sim i} (Q_{ij} + \epsilon)(x_j - \mu_j), \tag{31}$$

$$Prec(x_i|\mathbf{x}_{-i}) = (Q_{ii} + \epsilon), \tag{32}$$

$$Corr(x_i, x_j|\mathbf{x}_{-ij}) = -\frac{(Q_{ij} + \epsilon)}{\sqrt{(Q_{ii} + \epsilon)(Q_{jj} + \epsilon)}}, i \neq j. \tag{33}$$

In the limit as $\epsilon \to 0$, these approach (3)-(5). That is, (3)-(5) hold even for the IGMRF $\mathbf{x}$, where $\mathbf{Q}$ is singular. In particular,

when $\mu = \mathbf{0}$, note that

$$E(x_i|\mathbf{x}_{-i}) = -\frac{1}{Q_{ii}} \sum_{j:j\sim i} Q_{ij} x_j. \tag{34}$$

Since $-\frac{\sum_{j:j\sim i} Q_{ij}}{Q_{ii}} = 1$ due to Eq. (27), the conditional mean of x_i is simply a weighted average of its neighbors. Such "local" behavior is desirable in many applications, as we will explain later.

IGMRFs of first order are closely related to a widely used model known as the first-order random walk. Furthermore, IGMRFs with higher orders can be defined similarly, and can be associated with higher order random walks on the graph. Interested readers are referred to [27] for more explanation of the relationship between the probabilistic models.

In the following sections, we will abuse terms and use GMRFs to refer to both proper and improper GMRFs. It shall be kept in mind that when a GMRF is improper, if necessary, with care, its properties may be interpreted as limits of properties of sequences of proper GMRFs.

III. Graph Construction

In any graph signal processing applications, the first and utmost important task is to construct the graph for the signal. This is, however, a non-trivial task. In the following, we examine numerous schemes to construct the signal graph with data statistics or heuristic models.

A. The Data-Driven Approach

Assume we are given a large number of observations from a high dimensional signal. Our goal is to construct a graph, including both its topology and edge weights, such that the graph can represent the signal in a principled manner. To this end, let us denote the high dimensional signal as a random vector $\mathbf{x} = (x_1, \cdots, x_n)^T$. Following Section II, we may model the signal's statistical distribution as a Gaussian Markov Random Field. The mean and covariance matrix of the GMRF can be easily estimated via sample mean and sample covariance of the observed examples. After removing the mean, the remaining signal can be fully described by its covariance matrix. According to Section II-B, if we take the inverse of the covariance matrix and obtain the precision matrix, we can construct a unique graph with or without self-loops to describe the signal.

While the above data-driven approach would be the most accurate if we have plenty of observations, it has some shortcomings. First, since the precision matrix is computed from the sample covariance matrix, most likely it is a full matrix consisting of few, if any, zero entries. Such highly connected graphs are usually inconvenient to analyze. (See [23], [22], and the references therein for approaches to learning sparse precision matrices.) Second, when the signal's dimensionality is high, estimating an accurate covariance matrix requires a lot of data in order to be statistically valid. Unfortunately, for many real world applications, data collection is expensive, and difficult to conduct.

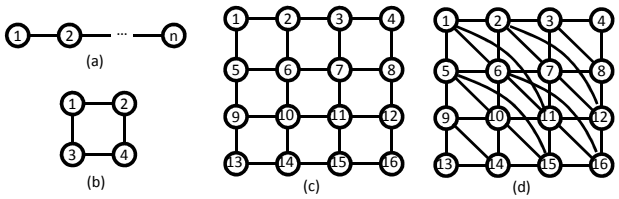

Fig. 2. A few intuitive graph models for 1D and 2D images. (a) 1D chain model, (b-c) simple 2D models for a 2×2 and a 4×4 block, where each pixel is only connected with its direct neighbors, (d) a more complex 2D model.

B. Intuitive Model-Based Approach

For certain types of signals such as 1D and 2D images, one can often construct the graph in an intuitive manner. For instance, in Fig. 2 (a)(b)(c), each pixel of the image is represented by a node in the graph. When weights (often unity) are assigned to edges in the graph, the graph corresponds to a GMRF, whose precision matrix is given as follows. Let the weights between two connected nodes i and j be $W_{ij} = W_{ji}$, and define

$$\mathbf{L} = \mathbf{D} - \mathbf{W}, \tag{35}$$

where $\mathbf{W}$ is the weight matrix, and $\mathbf{D} = diag(d_1, \cdots, d_n)$ is a diagonal degree matrix, where

$$d_i = 2W_{ii} + \sum_{j:j\neq i} W_{ij}. \tag{36}$$

The matrix $\mathbf{L}$ is often referred as the (loopy) graph Laplacian matrix, and it is a difference operator. Using (9) and (10) in Section II-B, it can be verified that $\mathbf{L} = \hat{\mathbf{Q}}$: clearly, $L_{ij} = -W_{ij}$ for all $i \neq j$, and $L_{ii} = \sum_j W_{ij} = \hat{Q}_{ii}$ for all i. Hence if $\mathbf{L}$ is positive semi-definite, then we may simply use the Laplacian matrix as the precision matrix,

$$\mathbf{Q} = \mathbf{L}, \tag{37}$$

which is consistent with the bijective mapping between the graph and the corresponding GMRF model. Such a GMRF model is sometimes called a *Laplacian GMRF* model. The Laplacian GMRF model has been widely adopted in the literature, such as in image reconstruction [19], texture modeling and discrimination [4], [8], image segmentation [17], etc.

If $\mathbf{x}$ is a signal on the graph, then its graph Laplacian quadratic form

$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \sum_i W_{ii}x_i^2 + \sum_i \sum_{j:j\sim i} W_{ij}(x_i - x_j)^2 \tag{38}$$

is often used as a measure for the signal's global smoothness [6]. The eigenvector matrix of the graph Laplacian is known as the *graph Fourier transform*, which can be used to define different notions of smoothness on the graph, leading to various applications for graph filter design and analysis [30].

Another popular option to define a difference operator on a graph is the normalized graph Laplacian. The normalized graph Laplacian is defined as:

$$\tilde{\mathbf{L}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{L}\mathbf{D}^{-\frac{1}{2}}. \tag{39}$$

The eigenvalues of the normalized graph Laplacian has the nice property that they are contained between the interval

$[0, 2]$. According to [6], the advantage of the normalized graph Laplacian definition is its consistency with the eigenvalues in spectral geometry and in stochastic processes. Many results which were only known for regular graphs can be generalized to all graphs.

One may also form a GMRF model using the normalized graph Laplacian:

$$\tilde{\mathbf{Q}} = \tilde{\mathbf{L}}. \tag{40}$$

Compared with the regular Laplacian GMRF in (37), we can observe that the conditional correlation between any two nodes remains unchanged (Eq. (5)). However, the conditional precision of each element has been normalized to unity. Since the resultant GMRF model can be inversely mapped to an infinite number of graphs with the same normalized graph Laplacian, the above mapping is not bijective.

C. Model-Constrained Data-Driven Approach

Using heuristic-based graph models is a convenient way for graph signal processing. Usually a node in the graph will be connected with limited number of neighbors; thus the final precision matrix is sparse. However, except for a few widely tested signals (such as images), it is difficult to predict how much error we are introducing by heuristically connecting nodes and assigning weights. Ideally, the model-based and data-driven approaches should be combined in order to achieve better performance.

Formally, given a zero-mean multi-dimensional random vector $\mathbf{x} = (x_1, \cdots, x_n)^T$ and a set of independently-drawn observations $\{\mathbf{x}_1, \cdots, \mathbf{x}_M\}$, our goal is to find the precision matrix $\hat{\mathbf{Q}}$ that maximizes the posterior,

$$\begin{aligned} \hat{\mathbf{Q}} &= \arg\max_{\mathbf{Q}} p(\mathbf{Q}|\mathbf{x}_1, \cdots, \mathbf{x}_M) \\ &= \arg\max_{\mathbf{Q}} p(\mathbf{Q}) \prod_{m=1}^{M} p(\mathbf{x}_m|\mathbf{Q}). \end{aligned} \tag{41}$$

Since we model the signal using GMRF models, the likelihood term $p(\mathbf{x}_m|\mathbf{Q})$ can be easily written

$$p(\mathbf{x}_m|\mathbf{Q}) = (2\pi)^{-\frac{n}{2}} |\mathbf{Q}|^{\frac{1}{2}} \exp\Big(-\frac{1}{2} \mathbf{x}_m^T \mathbf{Q} \mathbf{x}_m \Big). \tag{42}$$

The prior, $p(\mathbf{Q})$, encodes our knowledge about the target model, which is imposed artificially to encourage certain structure in the resultant graph.

The knowledge, certainly, is application dependent. It could be as simple as an impulse like prior that enforces all graph edge weights to have values 0 or 1 depending on the edge strength between pixels (based on the current image block), which was used in [29] without considering a probabilistic framework explicitly. A more sophisticated example that adopts the above probabilistic framework is the recent work by Dong *et al.* [10], and we refer the readers to their paper for more details.

IV. The Graph Transform

A. The Graph Transform

For a random signal residing on a graph, its elements are often highly correlated. A popular signal processing tool is to decorrelate the elements, making them easy to analyze or process. Let us decorrelate such a random graph signal $\mathbf{x}$. To begin, let us assume that $\mathbf{x}$ is a zero-mean, (proper) GMRF, with covariance matrix $\mathbf{\Sigma} = \mathbf{Q}^{-1}$, where $\mathbf{Q}$ is the precision matrix. The linear transform that decorrelates $\mathbf{x}$ is thus the Karhunen-Loève transform (KLT) $\mathbf{\Phi}^T$, where the columns of $\mathbf{\Phi}$ are the eigenvectors of $\mathbf{\Sigma}$. That is,

$$\mathbf{\Sigma\Phi} = \mathbf{\Phi\Gamma}, \tag{43}$$

where $\mathbf{\Gamma} = \mathrm{diag}(\gamma_1, \cdots, \gamma_n)$ is the diagonal matrix of eigenvalues of $\mathbf{\Sigma}$. Since

$$\mathbf{Q\Phi} = \mathbf{\Sigma}^{-1}\mathbf{\Phi} = (\mathbf{\Phi\Gamma\Phi}^T)^{-1}\mathbf{\Phi} = \mathbf{\Phi\Gamma}^{-1} = \mathbf{\Phi\Lambda}, \tag{44}$$

$\mathbf{\Phi}$ is also the eigenvector matrix of $\mathbf{Q}$, and the eigenvalues of $\mathbf{Q}$ are the inverses of the eigenvalues of $\mathbf{\Sigma}$.

In the event that $\mathbf{x} = \mathbf{V}_1\mathbf{u}_1 + \mathbf{V}_2\mathbf{u}_2$ is an (improper) IGMRF with precision matrix $\mathbf{Q}$ as in Section II-C, it can be seen that the transpose of the eigenvector matrix of $\mathbf{Q}$, $\mathbf{\Phi}^T = [\mathbf{V}_1, \mathbf{V}_2]$, "decorrelates" $\mathbf{x}$ into components $\mathbf{u}_1$ and $\mathbf{u}_2$, where $\mathbf{u}_1$ is a vector of independent Gaussian random variables and $\mathbf{u}_2$ is an indeterminate vector.

Therefore, whether $\mathbf{x}$ is a zero-mean GMRF or IGMRF with precision matrix $\mathbf{Q}$, the transpose of the eigenvector matrix of the precision matrix is a decorrelating transform. Since it is defined on a graph, we call it the *Graph Transform*. It can be shown that the various notions of optimality of the KLT extend naturally to the graph transform.

B. Graph Transform on Laplacian GMRF and Filtering

For signals that follow the Laplacian GMRF model, it is clear that the above defined graph transform is identical to the well-known graph Fourier transform. Therefore, the graph Fourier transform decorrelates the Laplacian GMRF signal.

Given a graph signal's Fourier transform, various operators can be defined on the signal, such as filtering, translation, modulation, dilation, etc. [30]. For example, let $\mathbf{x}$ be a zero-mean, Laplacian GMRF. The standard definition of filtering involves projecting the input signal with the Fourier transform matrix, applying a spectral filter, and then applying an inverse Fourier transform:

$$\mathbf{x}_f = \mathbf{\Phi H \Phi}^T\mathbf{x}, \tag{45}$$

where $\mathbf{x}_f$ is the filtered signal, and $\mathbf{H} = \mathrm{diag}(h_1, \cdots, h_n)$ is the filter matrix. Since $\mathbf{x}$ is Gaussian, and $\mathbf{H}$ and $\mathbf{\Phi}$ are both linear transforms, the filtered signal is also Gaussian, with covariance matrix

$$\mathbf{\Sigma}_f = E\mathbf{x}_f\mathbf{x}_f^T = (\mathbf{\Phi H \Phi}^T)\mathbf{\Sigma}(\mathbf{\Phi H \Phi}^T) \tag{46}$$

$$= (\mathbf{\Phi H \Phi}^T)\mathbf{\Phi\Gamma\Phi}^T(\mathbf{\Phi H \Phi}^T) = \mathbf{\Phi\Gamma}_f\mathbf{\Phi}^T \tag{47}$$

and precision matrix

$$\mathbf{Q}_f = \mathbf{\Sigma}_f^{-1} = \mathbf{\Phi\Gamma}_f^{-1}\mathbf{\Phi}^T = \mathbf{\Phi\Lambda}_f\mathbf{\Phi}^T, \tag{48}$$

where $\mathbf{\Gamma}_f = \mathbf{H\Gamma H} = \mathrm{diag}(h_1^2\gamma_1, \cdots, h_n^2\gamma_n)$ and $\mathbf{\Lambda}_f = \mathbf{H}^{-1}\mathbf{\Lambda H}^{-1} = \mathrm{diag}(\lambda_1/h_1^2, \cdots, \lambda_n/h_n^2)$. It can be seen that the filter operation essentially scales the multivariate Gaussian signal $\mathbf{x}$ along its principal axes by $h_1, \cdots, h_n$, effectively changing the correlations and precisions of the elements of $\mathbf{x}$.

In the event that $\mathbf{x}$ is an (improper) IGMRF, then it can be seen from the density (18) and a change of variables analogous to (22)–(25) that once again, $\mathbf{Q}_f = \mathbf{\Phi}\mathbf{\Lambda}_f\mathbf{\Phi}^T$.

Hence, provided with an arbitrary precision matrix $\mathbf{Q}$ (not necessarily derived from the graph Laplacian), filtering can be defined based on its own graph transform. From the probabilistic viewpoint of GMRF models, filtering on the graph signal is nothing but finding its principal axes and stretching them according to the needs of the applications. It shall be noted that filtering does not change the graph transform matrix $\mathbf{\Phi}^T$ of $\mathbf{Q}$.

C. Graph Transform and DCT

The graph transform is a signal-dependent transform, since it depends on the statistics of the signal, in particular, the precision matrix $\mathbf{Q}$. Nevertheless, for a certain family of signals, such as images on a 1D or 2D grid, researchers have applied data-independent GMRF models for various applications with great success [18][19][4][17]. In the following, we use a simple Laplacian GMRF model as the representation of 1D or 2D images, subsequently revealing the close relationship between graph transform and one of the most popular transforms for image processing: the discrete cosine transform (DCT).

Given a 1D or 2D image signal on a *regular lattice graph* $\mathcal{G}$ (Fig. 2(a)(b)(c)), let us define a weight matrix with $W_{ij} = W_{ji} = 1$ if nodes i and j are immediate neighbors connected by an edge. Otherwise, $W_{ij} = W_{ji} = 0$. Following the steps in Section III-B, we can easily define a Laplacian GMRF on the graph. It turns out that for the 1D signal graph as shown in Fig. 2(a), the above GMRF model is equivalent to a first order autoregressive signal model. The eigenvector matrix of the Laplacian matrix $\mathbf{L}$ has been shown to be identical to DCT (more specifically, DCT-2) [31], which is consistent with the conclusion in [7], where Clarke proved the optimality of the 1D DCT for an autoregressive signal model.

When the graph signal lies on a 2D lattice graph as Fig. 2(b) and (c), the Laplacian matrix $\mathbf{L}$ still has rank $n-1$, but it has duplicated eigenvalues. This means that we have an infinite number of optimal linear transforms that can fully decorrelate the signal. In our previous work [33], we showed that the 2D DCT is an eigenvector matrix for $\mathbf{L}$, and is thus one of these optimal linear transforms. In other words, although the 2D DCT is generally viewed simply as a computationally efficient extension of the 1D DCT into 2D, it is actually optimal for a very reasonable signal model: the Laplacian GMRF on the 2D lattice graph where the conditional mean of a pixel is the arithmetic mean of its four closest neighbors. This confirms the successful application of the 2D DCT in typical image coding algorithms such as JPEG.

D. Other Graph Transform Applications

The graph transform has received a lot of attention recently in various data compression tasks. For instance, a transform design referred to as an edge adaptive transform (EAT) was recently proposed in depth map coding [29], where a graph is defined on image blocks, and the correlation across depth edges in the graph is set to 0. The authors applied the eigenvector matrix of the Laplacian matrix $\mathbf{L}$ as the transform for the signal. Although they did not model the depth signal with a probabilistic model, our analysis above shows that an implicit Laplacian GMRF model was assumed on the depth data, and the EAT is indeed the optimal transform to decorrelate the signal under that assumption.

For less regular graph signals, the work in [34] applied graph transform on the compression of point cloud attributes, such as color and normal. There the weights between neighboring points are defined according to the distances between them. Still the graph Laplacian matrix is used as the precision matrix, and the corresponding graph transform demonstrated great performance against existing methods.

V. Graph Downsampling

Many multi-resolution signal processing schemes on graphs require successively generating coarser versions of the original graph and preserving properties of the original graph as much as possible. While there are many different algorithms that have been developed for graph coarsening [26], below we present a coarsening scheme from a probabilistic viewpoint.

Consider a random vector $\mathbf{x} = (x_1, \cdots, x_n)$ on a graph $\mathcal{G} = (\mathcal{V} = \{1, \cdots, n\}, \mathcal{E})$ that follows a (proper) GMRF model with mean μ and precision matrix $\mathbf{Q}$. In order to coarsen the graph, we plan to remove k nodes from the graph, reducing the input random vector's dimensionality from n to $n-k$. Note here we only consider the case where the remaining nodes are a subset of the original node set, which is often referred as *graph downsampling*. We have two main questions to address: how to choose the best k nodes such that we preserve as much information as possible, and once the k nodes are removed, how to construct the new graph such that it reflects the relationship between the remaining nodes?

A. Graph Reconstruction after Downsampling

We start by answering the second question, which is more straightforward. Knowing the k nodes to be eliminated, without loss of generality, we assume they are the last k elements of the input vector $\mathbf{x}$. Let $\mathbf{x}_1 = (x_1, \cdots, x_{n-k})^T$ be the nodes to be kept, and $\mathbf{x}_2 = (x_{n-k+1}, \cdots, x_n)^T$ be the nodes to be removed. We may partition the mean vector μ as

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \tag{49}$$

and the covariance matrix $\mathbf{\Sigma}$ and precision matrix $\mathbf{Q}$ as

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{pmatrix}, \mathbf{Q} = \mathbf{\Sigma}^{-1} = \begin{pmatrix} \mathbf{Q}_{11} & \mathbf{Q}_{12} \\ \mathbf{Q}_{21} & \mathbf{Q}_{22} \end{pmatrix}, \tag{50}$$

where μ_1 is the mean of $\mathbf{x}_1$, and μ_2 is the mean of $\mathbf{x}_2$. By removing the $\mathbf{x}_2$ elements from $\mathbf{x}$, we are left with a marginalized GMRF signal $\mathbf{x}_1$, with mean μ_1 and covariance matrix $\mathbf{\Sigma}_{11}$. Letting $\mathbf{Q}_1$ represent the precision matrix of $\mathbf{x}_1$, we have $\mathbf{Q}_1 = \mathbf{\Sigma}_{11}^{-1}$ and hence

$$\mathbf{Q}_1 = \mathbf{Q}_{11} - \mathbf{Q}_{12}\mathbf{Q}_{22}^{-1}\mathbf{Q}_{21}, \tag{51}$$

which can be easily derived based on block matrix inversion. $\mathbf{Q}_{22}$ is guaranteed to be invertible because $\mathbf{Q}$ is invertible and

is therefore positive definite. Therefore, the graph downsampling operation involves removing the k unwanted nodes from the graph, and reconnect all nodes based on the conditional correlation specified in $\mathbf{Q}_1$. Namely, if the entry at (i,j) for $\mathbf{Q}_1$ is non-zero, an edge shall be created on the downsampled graph connecting nodes i and j.

In the event that $\mathbf{x}$ is an (improper) IGMRF with a single connected component, it can be shown that $\mathbf{Q}_{22}$ is still invertible and hence (51) remains well-defined [11, Lemma 2.1(i)].

It turns out that the above graph downsampling process is known in the literature as the Kron reduction of a graph [11], which was originally derived from electrical networks. It is ubiquitous in classic circuit theory and many other disciplines such as sparse matrix algorithms, multi-grid solvers, finite-element analysis, etc. Our derivation above demonstrates that the Kron reduction has a simple yet profound probabilistic grounding, in particular for graph signal processing if the graph signals are modeled with GMRF models.

B. Progressive Graph Downsampling

The Kron reduction can be applied iteratively (one node by one node) to downsample the graph in a progressive manner. From the previous discussion, it is clear that if in the end only $n-k$ nodes are kept, the order of Kron reduction for the k eliminated nodes does not matter, since in the end we always reach the marginalized GMRF distribution of $\mathbf{x}_1$. However, if we were only told to remove k nodes from the graph, it remains unclear which k nodes shall be chosen in order to be optimal.

From a probabilistic viewpoint, we define the information loss by downsampling the original graph signal $\mathbf{x}$ to $\mathbf{x}_1$ as the entropy difference between $\mathbf{x}$ and $\mathbf{x}_1$. Since the distribution of $\mathbf{x}$ is fixed, we select $\mathbf{x}_1$ such that it will have the maximum differential entropy, which can be computed as:

$$\hat{\mathbf{x}}_1 = \arg\max_{\mathbf{x}_1} H(\mathbf{x}_1) = \arg\max_{\mathbf{x}_1} \frac{1}{2}\log\frac{(2\pi e)^{n-k}}{|\mathbf{Q}_1|}, \qquad (52)$$

where $\log$ is base 2 and the unit of entropy $H(\mathbf{x}_1)$ is in bits. It can be shown that selecting $\mathbf{x}_1$ according to this criterion in the same as finding the $\mathbf{x}_1$ that maximizes the mutual information with $\mathbf{x}$. Admittedly, an exhaustive search of the maximum of the remaining signal's entropy is still very expensive and NP-hard. In that regard, we refer the readers to [26][30] for more practical algorithms for graph downsampling.

VI. Graph Prediction

A. Prediction on the Graph

Consider a possibly improper IGMRF $\mathbf{x} = (x_1, \cdots, x_n, x_{n+1}, \cdots, x_m)^T$ with parameters $(\mu, \mathbf{Q})$, and assume that among the elements of $\mathbf{x}$, $\mathbf{x}_1 = (x_1, \cdots, x_n)^T$ is unknown, and $\mathbf{x}_2 = (x_{n+1}, \cdots, x_m)^T$ is known. As we have shown in Section II-D, the distribution of $\mathbf{x}_1$ conditioned on $\mathbf{x}_2$ is a proper GMRF. Specifically, it can be shown (see, e.g., [27]) that if μ and $\mathbf{Q}$) are correspondingly partitioned

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \mathbf{Q} = \begin{pmatrix} \mathbf{Q}_{11} & \mathbf{Q}_{12} \\ \mathbf{Q}_{21} & \mathbf{Q}_{22} \end{pmatrix}, \qquad (53)$$

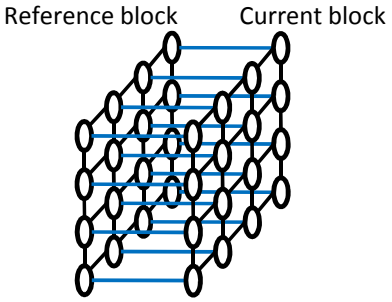

Fig. 3. An illustrative example for motion prediction. A 3D graph can be defined on the pixels to represent an example GMRF model.

then $\mathbf{x}_1|\mathbf{x}_2$ is a proper GMRF with mean $\mu_{\mathbf{x}_1|\mathbf{x}_2}$ and precision matrix $\mathbf{Q}_{\mathbf{x}_1|\mathbf{x}_2}$, where

$$\mu_{\mathbf{x}_1|\mathbf{x}_2} = \mu_1 - \mathbf{Q}_{11}^{-1}\mathbf{Q}_{12}(\mathbf{x}_2 - \mu_2), \qquad (54)$$

$$\mathbf{Q}_{\mathbf{x}_1|\mathbf{x}_2} = \mathbf{Q}_{11}. \qquad (55)$$

Again, it can be shown that if $\mathbf{x}$ is an IGMRF with a single connected component, then $\mathbf{Q}_{11}$ is invertible [11, Lemma 2.1(i)].

B. Graph Signal Interpolation

The graph prediction theory described above can be directly applied in graph signal interpolation. Given a graph and its associated weights, we first model the graph signal as a GMRF following Section II-A. We can then easily estimate the missing elements on the graph using (54), since it would represent the conditional mean of the missing elements.

C. Predictive Graph Transform

Graph prediction can also be very insightful in determining the best strategy for predictive transform coding. From Eq.(54) and (55), it is clear that in order to optimally decorrelate the variable $\mathbf{x}_1$ given $\mathbf{x}_2$, we can first subtract the conditional mean $\mu_{\mathbf{x}_1|\mathbf{x}_2}$ from $\mathbf{x}_1$, and then apply the eigenvector matrix of $\mathbf{Q}_{11}$ to transform the signal for further processing/compression. Such a procedure is optimal because $\mathbf{x}_1|\mathbf{x}_2$ is a GMRF, and we can reuse all the derivation on graph transform presented in Section IV. We term this the *Predictive Graph Transform* (PGT).

In the following, we discuss the application of PGT in motion prediction and intra-frame predictive coding. In both cases, we assume a generic GMRF model of the image is given during the analysis.

D. Motion Prediction

In motion prediction, a reference block is found through various motion estimation approaches, and is used to predict the block that is currently being encoded. Assume the two blocks are zero mean, and follow GMRF models described by precision matrix $\mathbf{Q}_{\text{ref}}$ and $\mathbf{Q}_{\text{C}}$, respectively. To this end, let us construct a GMRF model in 3D, as shown in Fig. 3. If we assume all "prediction" edges have weight one (since the reference block should be very similar to the current block

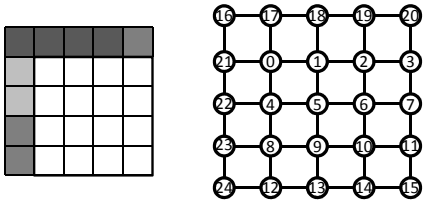

Fig. 4. An illustrative example for intra-frame prediction. The 4×4 image block will be predicted by the shaded known pixels on the top and left. The right figure is a typical graph defined on the image.

due to motion search), the precision matrix of the 3D GMRF model can be written

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_{11} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{Q}_{22} \end{pmatrix}, \tag{56}$$

where $\mathbf{I}$ is an identity matrix, and

$$\mathbf{Q}_{11} = \mathbf{Q}_c + \mathbf{I}, \mathbf{Q}_{22} = \mathbf{Q}_{\text{ref}} + \mathbf{I}. \tag{57}$$

Based on the derivation in Section VI-A, we may predict the current block $\mathbf{x}_1$ as

$$\mu_{\mathbf{x}_1|\mathbf{x}_2} = \mathbf{Q}_{11}^{-1}\mathbf{x}_2, \tag{58}$$

and then apply the eigenvector matrix of $\mathbf{Q}_{11}$ to decorrelate the signal.

The above analysis has interesting implications. For motion prediction, instead of directly copying the pixels from the reference block to the current block, Eq. (58) suggests that the optimal scheme is to first apply a filter on $\mathbf{x}_2$ before copying. In addition, since any orthogonal basis is an eigenvector matrix of the identity matrix $\mathbf{I}$, it can be shown that $\mathbf{Q}_{11}$ will share the same set of eigenvectors as $\mathbf{Q}_c$. Hence the optimal transform for the residue remains the same as when no motion prediction is performed.

For the special case that $\mathbf{Q}_c = \mathbf{L}$, since the Laplacian is essentially a high pass filter, $\mathbf{Q}_{11}^{-1}$ will be a low-pass filter. Therefore, one should *blur* the reference block and then copy it to the current block. Furthermore, from the analysis in Section IV-C, we can conclude that the 2D DCT transform is still optimal for encoding the prediction residual.

E. Intra Predictive Coding

In modern video codecs, the intra frames will also be predicted from neighboring known pixels to enhance coding efficiency. Again we may form a simple graph for the 2D block including the neighboring pixels, as shown in Fig. 4. Following Section VI-A, for a zero mean image, the optimal prediction would be:

$$\mu_{\mathbf{x}_1|\mathbf{x}_2} = -\mathbf{Q}_{11}^{-1}\mathbf{Q}_{12}\mathbf{x}_2, \tag{59}$$

where $\mathbf{x}_1$ is the list of pixels to be encoded, and $\mathbf{x}_2$ is the list of known neighbors. The optimal transform is the eigenvector matrix of $\mathbf{Q}_{11}$.

Noted that the optimal prediction for intra-frame predictive coding is related to both $\mathbf{Q}_{11}$ and $\mathbf{Q}_{12}$. That is, how the unknown pixels are correlated to themselves, and how they are correlated to the known pixels. In general the 2D DCT is no longer the eigenvector matrix for $\mathbf{Q}_{11}$ due to the connections between $\mathbf{x}_1$ and $\mathbf{x}_2$. Similar to the previous works [32], [16], [28], our analysis calls for different schemes of intra-prediction and transform coding. On the other hand, our derivation is rather general, and not limited to separable or first order signal models.

If we consider the special case that $\mathbf{Q} = \mathbf{L}$, both $\mathbf{Q}_{11}^{-1}\mathbf{Q}_{12}$ and the eigenvector matrix of $\mathbf{Q}_{11}$ can be pre-computed. In practice, however, the neighboring known pixels may suggest a better GMRF model, and it could certainly be adopted to improve the coding efficiency.

VII. Regularization

When graph signals are analyzed, it is important to impose constraints or regularization such as smoothness with respect to the graph structure. A popular global smoothness measure is the p-Dirichlet form of the graph signal, defined as:

$$S_p(\mathbf{x}) := \frac{1}{p}\Big[\sum_i \sum_{j:j\sim i} W_{ij}(x_i - x_j)^2\Big]^{\frac{p}{2}}. \tag{60}$$

When $p = 1$, $S_1(\mathbf{x})$ is the total variation of the signal with respect to the graph. When $p = 2$, we have:

$$S_2(\mathbf{x}) = \sum_i \sum_{j:j\sim i} W_{ij}(x_i - x_j)^2 = \mathbf{x}^T\mathbf{L}\mathbf{x}, \tag{61}$$

which is the graph Laplacian quadratic form mentioned in Section III-B.

In graph signal processing applications, a typical goal is to find an estimate of the signal on the graph, such that a certain cost function is minimized, subject to the smoothness constraint or regularization. That is, we solve:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} f(\mathbf{x}) + \lambda S_p(\mathbf{x}), \tag{62}$$

where λ is a Lagrange multiplier. The above minimization problem can have probabilistic interpretations. For instance, when the main target function $f(\mathbf{x})$ is the least square error between estimation and observation, it can be interpreted as the log likelihood of $\mathbf{x}$ under a Gaussian noise observation model. Similarly, the regularization term $S_p(\mathbf{x})$ can be considered as the (logarithm of) prior distribution of $\mathbf{x}$. It is immediately obvious that when the graph Laplacian quadratic form $S_2(\mathbf{x})$ is used for regularization, the prior distribution corresponds to a Laplacian GMRF model, with the Lagrange multiplier λ being a scale factor that defines the precision matrix $\mathbf{Q}$ as the λ-multiple of the graph Laplacian matrix.

It shall be noted that any regularization can be considered as imposing an implicit prior probability distribution of the unknown graph signal $\mathbf{x}$. We single out $S_2(\mathbf{x})$ because it is one of the most widely used and it is equivalent to having a GMRF model over the unknown graph signal [21][14][9]. When $p \neq 1$, the prior distribution is no longer Gaussian, but rather a generalized Gaussian distribution [5], or generalized GMRF (GGMRF) distribution in our context. Generalized Gaussian distribution has been shown to be a good model for natural images, and is thus also widely used in the literature [3][24][1].

VIII. Conclusion

In this paper, we proposed to use the Intrinsic Gaussian Markov Random Field as the underlying probabilistic model for graph signal processing. Such an approach allows us to draw a few important conclusions that were not obvious in the literature, such as the optimality of the graph transform and 2D DCT, the probabilistic implication of the Kron reduction, the optimal predictive transform coding, etc. We believe our analysis provides a novel angle to analyzing graph signal processing, and may inspire more important works in the future.

In the end, we shall point out that GMRF, after all, is a Gaussian distribution model for the signal, and may not be applicable to all real-world applications. Under circumstances where GMRF is not an ideal model, more sophisticated models could be used. As a researcher working in the field, one shall truly understand the probabilistic implications of the GSP algorithm he/she adopts, and make adaptations as necessary to better solve real-world problems.

References

[1] Y. Bazi, L. Bruzzone, and F. Melgani. An unsupervised approach based on the generalized gaussian model to automatic change detection in multitemporal sar images. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(4):874–887, 2005.

[2] C. M. Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.

[3] C. Bouman and K. Sauer. A generalized gaussian image model for edge-preserving map estimation. *Image Processing, IEEE Transactions on*, 2(3):296–310, 1993.

[4] R. Chellappa and S. Chatterjee. Classification of textures using Gaussian Markov random fields. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 33(4):959–963, 1985.

[5] M. Chmielewski. Elliptically symmetric distributions: a review and bibliography. *International Statistical Review/Revue Internationale de Statistique*, pages 67–74, 1981.

[6] F. R. Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[7] R. J. Clarke. *Transform Coding of Images*. Academic Press, 1985.

[8] F. S. Cohen, Z. Fan, and M. A. Patel. Classification of rotated and scaled textured images using Gaussian Markov random field models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(2):192–202, 1991.

[9] C. Couprie, L. Grady, L. Najman, and H. Talbot. Power watershed: A unifying graph-based optimization framework. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1384–1399, 2011.

[10] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst. Learning graphs from signal observations under smoothness prior. In *submitted to: IEEE Transacctions on Signal Processing*, 2014.

[11] F. Dörfler and F. Bullo. Kron reduction of graphs with applications to electrical networks. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 60(1):150–163, 2013.

[12] R. Durbin. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.

[13] J. Edmonds and E. L. Johnson. Matching: A well-solved class of integer linear programs. In *in: Combinatorial structures and their applications (Gordon and Breach)*. Citeseer, 1970.

[14] L. Grady. Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1768–1783, 2006.

[15] R. M. Gray. *Probability, Random Processes, and Ergodic Properties*. Springer-Verlag, 1988.

[16] J. Han, A. Saxena, and K. Rose. Towards jointly optimal spatial prediction and adaptive transform in video/image coding. In *ICASSP*, 2010.

[17] S. Krishnamachari and R. Chellappa. Multiresolution Gauss-Markov random field models for texture segmentation. *IEEE Trans. on Image Processing*, 6(2):251–267, 1997.

[18] E. Y. Lam and J. W. Goodman. A mathematical analysis of the dct coefficient distributions for images. *IEEE Trans. on Image Processing*, 9(10):1661–1666, Oct. 2000.

[19] E. Levitan and G. T. Herman. A maximum a posteriori probability expectation maximization algorithm for image reconstruction in emission tomography. *IEEE Trans. on Medical Imaging*, 6(3):185–192, 1987.

[20] S. Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag New York, Inc., 1995.

[21] B. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random field models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):478–482, 1991.

[22] G. Marjanovic and A. O. Hero. On lq estimation of spares inverse covariance. In *ICASSP*, 2014.

[23] Z. Meng, B. Erikson, and A. O. Hero. Learning latent variable Gaussian graphical models. In *ICASSP*, 2014.

[24] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized gaussian and complexity priors. *Information Theory, IEEE Transactions on*, 45(3):909–919, 1999.

[25] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[26] D. Ron, I. Safro, and A. Brandt. Relaxation-based coarsening and multiscale graph organization. *Multiscale Modeling & Simulation*, 9(1):407–423, 2011.

[27] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman and Hall/CRC, 2005.

[28] A. Saxena and F. C. A. Fernandes. Jointly optimal intra prediction and adaptive primary transform. *document JCTVC-C108, MPEG-H/JCT-VC*, 2010.

[29] G. Shen, W.-S. Kim, S. K. Narang, A. Ortega, J. Lee, and H. Wey. Edge-adaptive transforms for efficient depth map coding. In *PCS*, 2010.

[30] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *Signal Processing Magazine, IEEE*, 30(3):83–98, 2013.

[31] G. Strang. The discrete cosine transform. *SIAM Review*, 41(1):135–147, 1999.

[32] Y. Ye and M. Karczewicz. Improved H.264 intra coding based on bidirectional intra prediction, directional transform, and adaptive coefficient scanning. In *ICIP*, 2008.

[33] C. Zhang and D. Florêncio. Analyzing the optimality of predictive transform coding using graph-based models. *IEEE Signal Processing Letters*, 20(1), Jan. 2013.

[34] C. Zhang, D. Florêncio, and C. Loop. Point cloud attribute compression with graph transform. In *ICIP*, 2014.