A Recursive Model-Reduction Method for Approximate Inference in Gaussian Markov Random Fields

Jason K. Johnson and Alan S. Willsky

Abstract

This paper presents *recursive cavity modeling*—a principled, tractable approach to approximate, near-optimal inference for large Gauss-Markov random fields. The main idea is to subdivide the random field into smaller subfields, constructing *cavity models* which approximate these subfields. Each cavity model is a concise yet faithful model for the surface of one subfield sufficient for near-optimal inference in adjacent subfields. This basic idea leads to a tree-structured algorithm which recursively builds a hierarchy of cavity models during an "upward pass" and then builds a complementary set of *blanket models* during a reverse "downward pass." The marginal statistics of individual variables can then be approximated using their blanket models. *Model thinning* plays an important role, allowing us to develop thinned cavity and blanket models thereby providing tractable approximate inference. We develop a *maximum-entropy* approach that exploits certain tractable representations of Fisher information on thin chordal graphs. Given the resulting set of thinned cavity models, we also develop a *fast preconditioner*, which provides a simple iterative method to compute optimal estimates. Thus, our overall approach combines recursive inference, variational learning and iterative estimation. We demonstrate the accuracy and scalability of this approach in several challenging, large-scale remote sensing problems.

I. Introduction

Markov random fields (MRFs) play an important role for modeling and estimation in a wide variety of contexts including physics [1], [2], communication and coding [3], signal and image processing [4], [5], [6], [7], [8], [9], pattern recognition [10] remote sensing [11], [12], [13], sensor networks [14], and localization and mapping [15]. Their importance can be traced in some cases to underlying physics of the phenomenon being modeled, in others to the spatially distributed nature of the sensors and computational resources, and in essentially all cases to the expressiveness of this model class. MRFs are *graphical models* [16], [17], that is, collections of random variables, indexed by nodes of graphs, which satisfy certain graph-structured conditional independence relations: Conditioned on the values of the variables on any set of nodes that separate the graph into two or more disconnected components, the sets of values on those disconnected components are mutually independent. An implication of this Markov property—thanks to the Hammersley-Clifford Theorem [18], [1]—is that the joint distribution of the variables at all nodes can be compactly described in terms of "local" interactions among variables at small, completely connected subsets of nodes (the *cliques* of the graph).

MRFs have another well recognized characteristic, namely that performing optimal inference on such models can be prohibitively complex because of the implicit coding of the global distribution in terms of many local interactions. For this reason, most applications of MRFs involve the use of suboptimal or approximate inference methods, and many such methods have been developed [19], [20], [21], [22]. In this paper we describe a new, systematic approach, to approximately optimal inference for MRFs that focuses explicitly on propagating local approximate models for subfields of the overall graphical model that are close (in a sense to be made precise) to the exact models for these subfields but are far simpler and, in fact allow computationally tractable exact inference with respect to these approximate models.

The building blocks for our approach—variable elimination, information projections, and inference on cycle-free graphs (that is, graphs that are *trees*)—are well-known in the graphical model community. What is new here is their synthesis into a systematic procedure for computationally tractable inference that focuses on recursive reduced-order *modeling* (based on information-theoretic principles) and exact inference on the resulting set of approximate models. The resulting algorithms also have attractive structure that is of potential value for distributed implementations such as in sensor networks. To be sure our approach has connections with work of others—perhaps most significantly with [23], [24], [25], [12], [26], [27], and we discuss these relationships as we proceed.

The authors are with the Laboratory for Information and Decision Systems at the Massachusetts Institute of Technology, 77 Mass. Ave., Cambridge, MA 02139 (email: {jasonj,willsky}@mit.edu). This research was funded by the Air Force Office of Scientific Research under Grant FA9550-04-1-0351 and by a grant from Shell International Exploration and Production, Inc.

While the principles for our approach apply to general MRFs, we focus our development on the important class of Gaussian MRFs (GMRFs). In the next section we introduce this class, discuss the challenges in solving estimation problems for such models, briefly review methods and literature relevant to these challenges and to our approach, and provide a conceptual overview of our approach that also explains its name: *Recursive Cavity Modeling* (RCM). In Section III we develop the model-reduction techniques required by RCM. In particular, we develop a tractable maximum-entropy method to compute information projections using convex optimization methods and tractable representations of Fisher information for models defined on chordal graphs. In Section IV we provide the details of the RCM methodology, which consists of a two-pass procedure for building cavity and blanket models and a corresponding hierarchical preconditioner for iterative estimation. Section V demonstrates the effectiveness of RCM with its application to several remote sensing problems. We conclude in Section VI with a discussion of RCM and further directions that it suggests.[1]

II. Preliminaries

A. Gaussian Markov Random Fields

Let $\mathcal{G} = (V, \mathcal{E})$ denote a graph with node (or vertex) set V and edge set $\mathcal{E} \subset V \times V$. Let x_v denote a random variable associated with node $v \in V$, and let x denote the vector of all of the x_v. If x is Gaussian with mean $\hat{\mathrm{x}}$ and invertible covariance P, its probability density can be written as

$$p(\mathrm{x}) \propto \exp\{-\frac{1}{2}(\mathrm{x}-\hat{\mathrm{x}})^T\mathrm{P}^{-1}(\mathrm{x}-\hat{\mathrm{x}})\} \propto \exp\{-\frac{1}{2}\mathrm{x}^T\mathrm{J}\mathrm{x} + \mathrm{h}^T\mathrm{x}\} \tag{1}$$

$$\begin{aligned} \mathrm{J}\hat{\mathrm{x}} &= \mathrm{h} \\ \mathrm{P} &= \mathrm{J}^{-1} \end{aligned} \tag{2}$$

The form on the right-hand side of (1) is often referred to as the *information form* of the statistics of a Gaussian process where J is the *information matrix*. The fill pattern of J provides the Markov structure [28]: x is Markov with respect to $\mathcal{G}$ if and only if $\mathrm{J}_{u,v} = 0$ for all $\{u, v\} \notin \mathcal{E}$.

In applications, $\hat{\mathrm{x}}$ and P typically specify *posterior* statistics of x after conditioning on some set of observations. The most common example is one in which we have an original GMRF with respect to $\mathcal{G}$, together with measurements, corrupted by independent Gaussian noise, at some or all of the nodes of the graph. Since an independent measurement at node v simply modifies the values of h_v and $\mathrm{J}_{v,v}$, the resulting field conditioned on all such measurements is also Markov with respect to $\mathcal{G}$.

B. The Estimation Problem

Given J and h, we wish to compute $\hat{\mathrm{x}}$ and (at least) the diagonal elements of P—thus providing the marginal distributions for each of the x_v. However solving the linear equations in (2) and inverting J can run into scalability problems. For example, methods that take no advantage of graphical structure require $\mathcal{O}(n^3)$ computations for graphs with n nodes. If the graph $\mathcal{G}$ has particularly nice structure, however, very efficient algorithms do exist. In particular, if $\mathcal{G}$ is a tree there are a variety of algorithms which compute $\hat{\mathrm{x}}$ and the diagonal of P with total complexity that is linear in n and that also allow distributed computation corresponding to "messages" being passed along edges of the graph. For example, if J is tri-diagonal, the variables x_v form a Markov chain, and efficient solution of (2) can be obtained by Gaussian elimination of variables from one end of the chain to the other followed by back-substitution—corresponding to a forward Kalman filtering sweep followed by a backward Rauch-Tung-Striebel smoothing sweep [14].

Because of the abundance of applications involving MRFs on graphs with cycles, there is considerable interest and a growing body of literature on computationally tractable inference algorithms. For example, the generalization of the Rauch-Tung-Striebel smoother to trees can in principle be applied to graphs with cycles by aggregating nodes of the original graph—using so-called junction tree algorithms [17] to form an equivalent model on a tree. However, the dimensions of variables at nodes in such a tree model depend on the so-called *tree-width* of the original graph [29], with overall inference complexity in GMRFs that grows as the cube of this tree-width. Thus,

[1] A C++ implementation of the algorithms described in this paper is available at `http://ssg.mit.edu/group/jasonj`.

these algorithms are tractable only for graphs with small tree-width, precluding use for many graphs of practical importance such as a 2D $s \times s$ lattice (with $n = s^2$ nodes) for which the tree-width is s (so that the cube of the tree-width is $n^{3/2}$, resulting in complexity that grows faster than linearly with graph size) or a 3D $s \times s \times s$ lattice for which the tree-width is s^2 (so that the cube of the tree-width is n^2).

Since exact inference is only feasible for very particular graphs, there is great interest in algorithms that yield approximations to the correct means and covariances and that have tractable complexity. One well-known algorithm is loopy belief propagation (LBP) [30], [20] which take the local message-passing rules which yield the exact solution on trees, and apply them unchanged and iteratively to graphs with cycles. There has been recent progress [31], [20] in understanding how such algorithms behave, and for GMRFs it is now known [31], [22] that if LBP converges, it yields the correct value for $\hat{\mathrm{x}}$ but not the correct values for the $\mathrm{P}_{v,v}$. Although some sufficient conditions for convergence are known [31], [32], LBP does not always converge and may converge slowly in large GMRFs.

There are several other classes of approximate algorithms that are more closely related to our approach, and we discuss these connections in the next subsection. As we now describe, RCM can be viewed as a direct, recursive approximation of an exact (and hence intractable) inference algorithm created by aggregating nodes of the original graph into a tree. In particular, by employing an information-theoretic approach to reduced-order modeling, together with a particular strategy for aggregating nodes, we construct tractable, near-optimal algorithms that can be applied successfully to very large graphs.

C. The Basic Elements of RCM

As with exact methods based on junction trees, RCM makes use of *separators*—that is, sets of nodes which, if removed from the graph, result in two or more disconnected components. By Markovianity, the sets of variables in each of these disconnected components are mutually independent conditioned on the set of values on the separator. This suggests a "divide and conquer" approach to describing the overall statistics of the MRF on a hierarchically-organized tree. Each node in this tree corresponds to a separator at a different "scale" in the field. For example, the root node of this tree might correspond to a separator that separates the entire graph into, say k, disconnected subgraphs. The root node then has k children—one corresponding to each of these disconnected subgraphs—and the node for each of these children would then correspond to a separator that further dissects that subgraph. This continues to some finest level at which exact inference computations on the subgraphs at that level are manageable. The problem with this approach, as suggested in Section II-B, is that the dimensionality associated with the larger separators in our hierarchical tree can be quite high—for instance, $\sqrt{n}$ in square grids. This problem has led several researchers [24], [7], [33], [34], [14] to develop approaches for GMRFs based on *dimensionality reduction*—that is, replacing the high-dimensional vector of values along an entire separator by a lower-dimensional vector. While approaches such as [33], [34] use statistically-motivated criteria for choosing these approximations, there are significant limitations of this idea. The first is that the use of low-dimensional approximations can lead to artifacts (that is, modeling errors which expose the underlying approximation), both across and along these separators. The second is that performing such a dimensionality reduction requires that we have available the exact mean and covariance for the vector of variables whose dimension we wish to reduce, which is precisely the intractable computation we wish to approximate! The third limitation is that these approaches are strictly top-down approaches—that is, they require establishing the hierarchical decomposition from the root node on down to the leaf nodes *a priori*, an approach often referred to as *nested dissection*. We also employ nested dissection in our examples, but the RCM approach also offers the possibility of bottom-up organization of computations, beginning at nodes located close to each other and working outward—a capability that is particularly appealing for distributed sensor networks.

The key to RCM is the use of the implicit, information form, corresponding to *models* for the variables along separators, allowing us to consider *model-order*—rather than dimensionality—reduction. In this way, we still retain full dimensionality of the variables along each separator, overcoming the problem of artifacts. Of course, we still have to deal with the computational complexity of obtaining the information form of the statistics along each separator. Doing that in a computationally and statistically principled fashion is one of the major components of RCM. Consider a GMRF on the graph depicted in Fig. 4(a) so that the information matrix for this field has a sparsity pattern defined by this graph. Suppose now that we consider solving for $\hat{\mathrm{x}}$ and the diagonal of P from (2) by *variable elimination*. In particular, suppose that we eliminate all of the variables within the dashed region

in Fig. 4(a) except for those right at the boundary. Doing this in general will lead to *fill* in the information matrix for the set of variables that remain after variable elimination. As depicted in Fig. 4(b), this fill is completely concentrated within the dashed region—that is, within this *cavity*. That is, if we ignore the connections outside the cavity, we have a model for the variables along the boundary that is generally very densely connected. This suggests approximating this high-order exact model for the boundary by a reduced or *thinned* model as in Fig. 4(c) with the sparsity suggested by this figure. Indeed, if we thin the model sufficiently, we can then continue the process of alternating variable elimination and model thinning in a computationally tractable manner.

Suppose next that there are a number of disjoint cavities as in Fig. 4 in each of which we have performed alternating steps of variable elimination to enlarge the cavity followed by model thinning to maintain tractability. Eventually, two or more of these cavities will reach a point at which they are adjacent to each other, as in Fig. 5(a). At this point, the next step is one of merging these cavities into a larger one (Fig. 5(b)), eliminating the nodes that are interior to the new, larger cavity (Fig. 5(c)), and then thinning this new model. If each step does sufficient thinning, computational tractability can be maintained. Eventually, this "outwards" elimination ends, and a reverse "inwards" elimination procedure commences, again done in information form. This inwards procedure eliminates all of the variables *outside* of each subfield, except for the variables adjacent to the subfield, producing what is known as a *blanket model*. Eliminating these variables involves computations that recursively produce blanket models for smaller and smaller subfields, as illustrated in Fig. 6, which shows that, once again, model thinning (going from Fig. 6(c) to Fig. 6(d)) plays a central role. Finally, once this inward sweep has been completed, we have information forms for the marginal statistics for each of the subfields that were used to initialize the outwards elimination procedure. Inverting these many smaller, now-localized models to obtain means and variances is then, by construction, computationally tractable.

RCM has some relationships to other work as well as some substantive differences. The general conceptual form we have outlined is closely related to the nested dissection approach [23], [12] to solving large linear systems. The approach to model thinning in [23], [12], however, is simply zeroing a set of elements (retaining just those elements which couple nearby nodes along the boundary). The statistical interpretation of this approach and its extensibility to less regular lattices and fields, however, are problematic. In particular, zeroing elements can lead to indefinite (and hence meaningless) information matrices, and even if this is not the case, such an operation in general will modify *all* of the elements of the covariance matrix (including the variances of individual variables). In contrast, we adopt a principled, statistical approach to model thinning, using so-called information projections, which guarantee that the means, variances and edgewise correlations in the thinned model are unchanged by the thinning process.

Information-theoretic approaches to approximating graphical models have a significant literature [25], [35], [29], [36], most of which focuses on doing this for a single, overall graphical model and not in the context of a recursive procedure such as we develop. One effort that has considered a recursive approach is [26] which examines time-recursive inference for *Dynamic Bayes' Nets (DBNs)*—that is, for graphical models that evolve in time, so that we can view the overall graphical model as a set of coupled temporal "stages." Causal recursive filtering then corresponds to propagating "frontiers"—that is, a particular choice of what we would call cavity boundaries corresponding to the values at all nodes at a single point in time. The method in [26] projects each frontier model into a family of factored models so that the projection is given by a product of marginals on disjoint subsets of nodes. Such an operation can be viewed as a special case of the "outward" propagation of cavity models where nodes in the boundary are required to be mutually independent. In our approach, we instead adaptively thin the graphical model by identifying and removing edges that correspond to weak *conditional* dependencies so that the thinned models typically do not become disconnected. Also, the other two elements of our approach—specifically, the hierarchical structure which requires merging operations as in Fig. 5 and the inward recursion for blanket models as in Fig. 6—don't arise in the consideration of DBNs [26]. This distinction is important for large-scale computation because the hierarchical, tree structure of RCM is highly favorable for parallel computing[2], whereas frontier propagation methods require serial computations. There are also parallels to the group renormalization method using decimation [37], [13], which constructs a multi-scale cascade of coarse-scale MRFs by a combination of node-elimination and edge-thinning, and estimates the most probable configuration at each scale using iterative methods.

[2]In exact inference methods using junction trees, the benefits of parallel computing are limited by the predominant computations on the largest separators, a limitation that RCM avoids through the use of thinned boundary models.

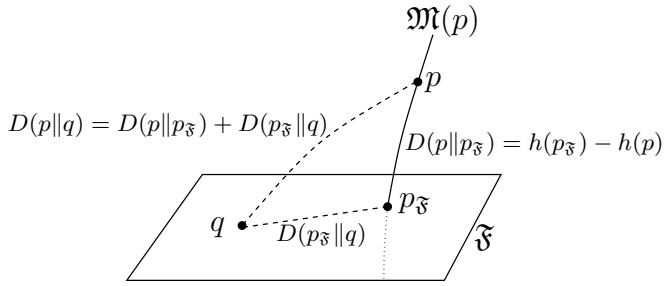

Fig. 1. Illustration of information projection and the Pythagorean relation.

III. Model Reduction

In this section we focus on the problem of model reduction, the solution of which RCM employs in the recursive thinning of cavity and blanket models. In Section III-A, we pose model reduction as information projection to a family of GMRFs and develop a tractable maximum-entropy method to compute these projections. In Section III-C we present a greedy algorithm that uses conditional mutual information to select which edges to remove. In our development we assume that the model being thinned is tractable. This is consistent with RCM in which we thin models, propagate them to larger ones that are still tractable and then thin again to maintain tractability.

A. Information Projection and Maximum Entropy

Suppose that we wish to approximate a probability distribution $p(\mathrm{x})$ by a GMRF defined on a graph $\mathcal{G} = (V, \mathcal{E})$. Over the family $\mathfrak{F}$ of GMRFs on $\mathcal{G}$ we select $q(\mathrm{x})$ to minimize the *information divergence* (relative entropy [38]) relative to p:

$$D(p\|q) = \int p(\mathrm{x}) \log \frac{p(\mathrm{x})}{q(\mathrm{x})} d\mathrm{x} \geq 0 \tag{3}$$

As depicted in Fig. 1, minimizing (3) can be viewed as a "projection" of p onto $\mathfrak{F}$. Many researchers have adopted (3) as a natural measure of modeling error [25], [36], [29]. The problem of minimizing information divergence takes on an especially simple characterization when the approximating family $\mathfrak{F}$ is an *exponential family* [39], [35], [40], that is, a family of the form $p_\theta(\mathrm{x}) \propto \exp\{\theta \cdot \phi(\mathrm{x})\}$ where $\theta \in \mathbb{R}^d$ are the *exponential parameters* and $\phi : \mathbb{R}^n \to \mathbb{R}^d$ is a vector of linearly independent *sufficient statistics*. The family is defined by the set $\theta \in \Theta$ for which $\int \exp\{\theta \cdot \phi(\mathrm{x})\} d\mathrm{x} < \infty$. The vector of *moments* $\eta = \Lambda(\theta) \triangleq \mathbb{E}_\theta\{\phi(\mathrm{x})\}$ plays a central role in minimizing (3). In particular, it can be shown that $\theta \in \Theta$ minimizes $D(p\|p_\theta)$ if and only if we have *moment matching relative to* $\mathfrak{F}$; that is if and only if the expected values of the sufficient statistics that define $\mathfrak{F}$ are the same under p_θ and the original density p. This optimizing element, which we refer to as the *information projection* of p to $\mathfrak{F}$ and denote by $p_{\mathfrak{F}}$, is the unique member of the family $\mathfrak{F}$ for which the following Pythagorean relation holds: $D(p\|q) = D(p\|p_{\mathfrak{F}}) + D(p_{\mathfrak{F}}\|q)$ for any $q \in \mathfrak{F}$ (see [35] and Fig. 1). Information projections also have a maximum entropy interpretation [41], [35] in that among all densities $q \in \mathfrak{M}(p)$ that match the moments of p relative to $\mathfrak{F}$, $p_{\mathfrak{F}}$ is the one which maximizes the entropy $h(q) = -\int q(\mathrm{x}) \log q(\mathrm{x}) d\mathrm{x}$. Moreover, the increase in entropy from p to $p_{\mathfrak{F}}$ is precisely the value of the *information loss* $D(p\|p_{\mathfrak{F}}) = h(p_{\mathfrak{F}}) - h(p)$.

The family of GMRFs on a graph $\mathcal{G}$ is represented by an exponential family with sufficient statistics $\phi_{\mathcal{G}}$, exponential parameters $\theta_{\mathcal{G}}$ and moment parameter $\eta_{\mathcal{G}}$ given by:

$$\begin{aligned} \phi_{\mathcal{G}}(\mathrm{x}) &\triangleq (\mathrm{x}_v)_{v\in V} \cup (\mathrm{x}_v^2)_{v\in V} \cup (\mathrm{x}_u\mathrm{x}_v)_{\{u,v\}\in\mathcal{E}} \\ \theta_{\mathcal{G}} &= (\mathrm{h}_v)_{v\in V} \cup (-\tfrac{1}{2}\mathrm{J}_{v,v})_{v\in V} \cup (-\mathrm{J}_{u,v})_{\{u,v\}\in\mathcal{E}} \\ \eta_{\mathcal{G}} &= (\hat{\mathrm{x}}_v)_{v\in V} \cup (\mathrm{P}_{v,v} + \hat{\mathrm{x}}_v^2)_{v\in V} \cup (\mathrm{P}_{u,v} + \hat{\mathrm{x}}_u\hat{\mathrm{x}}_v)_{\{u,v\}\in\mathcal{E}} \end{aligned} \tag{4}$$

Note that $\theta_{\mathcal{G}}$ specifies h and the non-zero elements of J while $\eta_{\mathcal{G}}$ specifies $\hat{x}$ and the corresponding subset of elements in P. These parameters are related by a one-to-one map $\eta_{\mathcal{G}} = \Lambda(\theta_{\mathcal{G}})$, defined by $\mathrm{P} = \mathrm{J}^{-1}$ and $\hat{\mathrm{x}} = \mathrm{J}^{-1}\mathrm{h}$, which is bijective on the image of realizable moments $\mathcal{M}(\mathcal{G})$.

Given a distribution $p(x)$, the projection to $\mathfrak{F}(\mathcal{G})$ is given as follows. Using the distribution p, we compute the moments relative to $\mathcal{G}$, or equivalently, the means $\hat{\mathrm{x}}_v$, variances $\mathrm{P}_{v,v}$ and edge-wise cross-covariances $\mathrm{P}_{u,v}$ on $\mathcal{G}$.

The information matrix J of the projection is then uniquely determined by the following complementary sets of constraints [42], [28]:

$$(\mathrm{J}^{-1})_{u,v} = \mathrm{P}_{u,v}, \ \ \forall (u,v) \in \mathcal{E}^* \tag{5}$$

$$\mathrm{J}_{u,v} = 0, \ \ \forall (u,v) \notin \mathcal{E}^* \tag{6}$$

where $\mathcal{E}^* = \mathcal{E} \cup \{(v,v), \ v \in V\}$. Eq. (5) imposes covariance-matching conditions over $\mathcal{G}$ while (6) imposes Markovianity with respect to $\mathcal{G}$. Also, by the maximum-entropy principle, an equivalent characterization of J is that $\mathrm{P} \triangleq \mathrm{J}^{-1}$ is the *maximum entropy completion* [43] of the partial covariance specification $\mathrm{P}_{\mathcal{G}} = (\mathrm{P}_{u,v}, (u,v) \in \mathcal{E}^*)$. Given J, the remaining moment constraints are satisfied by setting $\mathrm{h} = \mathrm{J}\hat{\mathrm{x}}$. Then, (h, J) is the information form of the projection to $\mathcal{G}$. Hence, projection to general GMRFs may be solved by a "shifted" projection to the zero-mean GMRFs and we may focus on this zero-mean case without any loss of generality. The family of zero-mean GMRFs is described as in (4) but without the linear-statistics x and corresponding parameters h and moments $\hat{\mathrm{x}}$.

B. Maximum-Entropy Relative to a Chordal Super-Graph

We now develop a method to compute the projection to a graph by embedding this graph within a chordal super-graph and maximizing entropy of the chordal GMRF subject to moment constraints over the embedded sub-graph. This approach allows us to exploit certain tractable calculations on chordal graphs to efficiently compute the projection to a non-chordal graph.

1) Chordal GMRFs: A graph is *chordal* if, for every cycle of four or more nodes, there exists an edge (a *chord*) connecting two non-consecutive nodes of the cycle. Let $\mathcal{C}(\mathcal{G})$ denote the set of *cliques* of $\mathcal{G}$: the maximal subsets $C \subset V$ for which the induced subgraph $\mathcal{G}_C$ is complete, that is, every pair of nodes in C is an edge of the graph. A useful result of graph theory states that a graph is chordal if and only if there exists a *junction tree*: a tree $T = (\Gamma, \mathcal{E}_\Gamma)$ whose nodes $\gamma \in \Gamma$ are identified with cliques $C_\gamma \in \mathcal{C}(\mathcal{G})$ and where for every pair of nodes $\alpha, \beta \in \Gamma$ we have $C_\alpha \cap C_\beta \subset C_\gamma$ for all γ along the path from α to β. Then, each edge $(\alpha, \beta) \in \mathcal{E}_\Gamma$ determines a minimal separator $S = C_\alpha \cap C_\beta$ of the graph. Moreover, *any* junction tree of a chordal graph $\mathcal{G}$ yields the *same* collection of edge-wise separators, which we denote by $\mathcal{S}(\mathcal{G})$. The importance of chordal graphs is shown by the following well-known result: Any strictly-positive probability distribution $p_{\mathcal{G}}(x)$ that is Markov on a chordal graph $\mathcal{G}$ can be represented in terms of its marginal distributions on the cliques $C \in \mathcal{C}(\mathcal{G})$ and separators $S \in \mathcal{S}(\mathcal{G})$ of the graph as

$$p_{\mathcal{G}}(x) = \frac{\prod_C p_C(x_C)}{\prod_S p_S(x_S)}. \tag{7}$$

In chordal GMRFs, this leads to the following formula for the sparse information matrix in terms of marginal covariances:

$$\mathrm{J} = \sum_C [\mathrm{P}_C^{-1}]_V - \sum_S [\mathrm{P}_S^{-1}]_V. \tag{8}$$

Here, $[\ldots]_V$ denotes zero-padding to a $|V| \times |V|$ matrix indexed by V. In the exponential family, this provides an efficient method to compute $\theta_{\mathcal{G}} = \Lambda^{-1}(\eta_{\mathcal{G}})$. Also, given the marginal covariances of an arbitrary distribution $p(x)$, not necessarily Markov on $\mathcal{G}$, (8) describes the projection of p to $\mathfrak{F}(\mathcal{G})$. The complexity of this calculation is $\mathcal{O}(nw^3)$ where w is the size of the largest clique.[3]

2) Entropy and Fisher Information in Chordal GMRFs: Based on (7), it follows that the entropy of a chordal MRF likewise decomposes in terms of marginal entropy on the cliques and separators of $\mathcal{G}$. In the moment parameters of the GMRF, we have

$$h(\eta_{\mathcal{G}}) = \sum_C h_C(\eta_{\mathcal{G}_C}) - \sum_S h_S(\eta_{\mathcal{G}_S}), \tag{9}$$

where h_C and h_S denote marginal entropy of cliques and separators, computed using

$$h_U(\eta_{\mathcal{G}_U}) = \tfrac{1}{2}(\log\det \mathrm{P}_U(\eta_{\mathcal{G}_U}) + |U| \log 2\pi e). \tag{10}$$

For exponential families, it is well-known that $\nabla h(\eta) = -\Lambda^{-1}(\eta)$ so that, for GMRFs, differentiating (9) reduces to performing the conversion (8). Thus, both $h(\eta_{\mathcal{G}})$ and $\nabla h(\eta_{\mathcal{G}})$ can be computed with $\mathcal{O}(nw^3)$ complexity.

[3]This complexity bound follows from the fact that, in chordal graphs, the number of maximal cliques is at most $n-1$ and, in GMRFs, the computations we perform on each clique are cubic in the size of the clique.

Next, we recall that the *Fisher information* with respect to parameters $\eta_{\mathcal{G}}$ is defined

$$\mathrm{G}(\eta_{\mathcal{G}}) \triangleq \mathbb{E}_{\eta_{\mathcal{G}}}\{\nabla \log p(x;\eta_{\mathcal{G}})\nabla^T \log p(x;\eta_{\mathcal{G}})\} = -\nabla\nabla^T h(\eta_{\mathcal{G}}),$$

where ∇ denotes gradient with respect to $\eta_{\mathcal{G}}$ and the expectation is with respect to the unique element $p \in \mathfrak{F}(\mathcal{G})$ with moments $\eta_{\mathcal{G}}$. Then, $\mathrm{G}(\eta_{\mathcal{G}})$ is a symmetric, positive-definite matrix and also describes the negative Hessian of entropy in exponential families. By twice differentiating (9), it follows that, in chordal GMRFs, the Fisher information matrix has a sparse representation in terms of marginal Fisher information defined on the cliques and separators of the graph:

$$\mathrm{G}(\eta_{\mathcal{G}}) = \sum_C [\mathrm{G}_C(\eta_{\mathcal{G}_C})]_{\mathcal{G}} - \sum_S [\mathrm{G}_S(\eta_{\mathcal{G}_S})]_{\mathcal{G}}, \tag{11}$$

where $\mathrm{G}_U(\eta_{\mathcal{G}_U}) \triangleq -\nabla\nabla^T h_U(\eta_{\mathcal{G}_U})$ is the marginal Fisher information on U and $[\dots]_{\mathcal{G}}$ denotes zero-padding to a matrix indexed by nodes and edges of $\mathcal{G}$. From (11), we observe that the fill pattern of $\mathrm{G}(\eta_{\mathcal{G}})$ defines another chordal graph with the same junction tree as $\mathcal{G}$, but where each clique $C \in \mathcal{C}(\mathcal{G}_c)$ maps to a larger clique with $\mathcal{O}(|C|^2)$ nodes (corresponding to a full sub-matrix of $\mathrm{G}(\eta_{\mathcal{G}})$ indexed by nodes and edges of $\mathcal{G}_C$). For this reason, direct use of $\mathrm{G}(\eta_{\mathcal{G}})$, viewed simply as a sparse matrix, is undesirable if $\mathcal{G}$ contains larger cliques. However, we can specify *implicit* methods that exploit the special structure of G to implement multiplication by either G or G^{-1} with $\mathcal{O}(nw^3)$ complexity. Observing that $\mathrm{G}(\eta_{\mathcal{G}}) = \frac{\partial \theta_{\mathcal{G}}}{\partial \eta_{\mathcal{G}}}$ represents the Jacobian of the mapping from $\eta_{\mathcal{G}}$ to $\theta_{\mathcal{G}}$, we can compute matrix-vector products $d\theta_{\mathcal{G}} = \mathrm{G} \cdot d\eta_{\mathcal{G}}$ for an arbitrary input $d\eta_{\mathcal{G}}$ (viewed as a change in moment coordinates). Differentiating (8) using $d(\mathrm{P}_U^{-1}) = -\mathrm{P}_U^{-1} d\mathrm{P}_U \mathrm{P}_U^{-1}$ we obtain:

$$d\mathrm{J} = -\sum_S [\mathrm{P}_C^{-1} d\mathrm{P}_C \mathrm{P}_C^{-1}]_V + \sum_C [\mathrm{P}_S^{-1} d\mathrm{P}_S \mathrm{P}_S^{-1}]_V. \tag{12}$$

Similarly, we can compute $d\eta_{\mathcal{G}} = \mathrm{G}^{-1} \cdot d\theta_{\mathcal{G}}$ by differentiating $\eta_{\mathcal{G}} = \Lambda(\theta_{\mathcal{G}})$. In appendix A, we summarize a recursive inference algorithm, defined relative to a junction tree of $\mathcal{G}$, that computes $\eta_{\mathcal{G}}$ given $\theta_{\mathcal{G}}$ and derive a corresponding differential form of the algorithm that computes $d\eta_{\mathcal{G}}$ given $d\theta_{\mathcal{G}}$. These methods are used to efficiently implement the variational method described next.

3) Maximum-Entropy Optimization: Given a GMRF p on $\mathcal{G}$, we develop a maximum-entropy (ME) method to compute the projection to an arbitrary (non-chordal) sub-graph $\mathcal{S}$. Let $\mathcal{G}'$ be a chordal super-graph of $\mathcal{G}$ and let $\mathcal{R} = \mathcal{E}(\mathcal{G}') \setminus \mathcal{E}(\mathcal{S})$ such that $\eta_{\mathcal{G}'} = (\eta_{\mathcal{S}}, \eta_{\mathcal{R}})$. We may compute $\eta_{\mathcal{G}'}(p)$ using recursive inference on a junction tree of $\mathcal{G}'$ (see Appendix A). To compute the projection to $\mathcal{S}$, we maximize entropy in the chordal GMRF subject to moment constraints over the sub-graph $\mathcal{S}$. This may be formulated as a convex optimization problem:

$$\begin{aligned} \min_{\eta_{\mathcal{R}}} \quad & f(\eta_{\mathcal{R}}) \triangleq -h(\eta_{\mathcal{S}}, \eta_{\mathcal{R}}) \\ \text{s.t.} \quad & (\eta_{\mathcal{S}}, \eta_{\mathcal{R}}) \in \mathfrak{M}(\mathcal{G}') \end{aligned} \tag{13}$$

Here, $\eta_{\mathcal{G}'} \in \mathfrak{M}(\mathcal{G}')$ are the realizable moments of the GMRF defined on $\mathcal{G}'$.[4] Starting from $\eta_{\mathcal{G}'}^{(0)} = \eta_{\mathcal{G}'}(p)$, we compute a sequence $\eta_{\mathcal{G}'}^{(k)} = (\eta_{\mathcal{S}}, \eta_{\mathcal{R}}^{(k)})$ using Newton's method. For each k, this requires solving the linear system

$$\mathrm{G}_{\mathcal{R}}^{(k)} \cdot \Delta\eta_{\mathcal{R}}^{(k)} = -\theta_{\mathcal{R}}^{(k)} \tag{14}$$

where $\mathrm{G}_{\mathcal{R}}^{(k)} = \nabla\nabla^T f$ is the principle sub-matrix of $\mathrm{G}(\eta_{\mathcal{G}'}^{(k)})$ corresponding to $\mathcal{R}$ and $\theta_{\mathcal{R}}^{(k)} = \nabla f$ is the corresponding sub-vector of $\theta_{\mathcal{G}'}^{(k)} = \Lambda^{-1}(\eta_{\mathcal{G}'}^{(k)})$ computed using (8). We then set $\eta_{\mathcal{R}}^{(k+1)} = \eta_{\mathcal{R}}^{(k)} + \lambda\Delta\eta_{\mathcal{R}}^{(k)}$, where $\lambda \in (0, 1]$ is determined by back-tracking line search to stay within $\mathfrak{M}(\mathcal{G}')$ and to insure that entropy is increased. This procedure converges to the optimal $\eta_{\mathcal{G}'}^* = (\eta_{\mathcal{S}}, \eta_{\mathcal{R}}^*)$, for which the corresponding exponential parameters satisfy $\theta_{\mathcal{R}}^* = 0$. Then, $\theta_{\mathcal{S}}^*$ is the information projection to $\mathcal{S}$.

Finally, we discuss an efficient method to compute the Newton step: If the width w of the chordal graph is very small, say $w < 10$, we could explicitly form the sparse matrix $\mathrm{G}_{\mathcal{R}}$ and efficiently solve (14) using direct methods. However, this approach has $\mathcal{O}(nw^6)$ complexity, which is undesirable for larger values of w. Instead, we use an *inexact* Newton's step, obtained by *approximate* solution of (14) using standard iterative methods, for instance,

[4] In the chordal graph $\mathcal{G}'$, the condition that $\eta_{\mathcal{G}'}$ is realizable is equivalent to $\mathrm{P}_C(\eta_{\mathcal{G}'_C}) \succeq 0$ for all $C \in \mathcal{C}(\mathcal{G}')$, which can be verified with complexity $\mathcal{O}(nw^3)$. Thus, $\mathfrak{M}(\mathcal{G}')$ is convex because the set of positive-definite matrices on each clique is convex.

preconditioned conjugate gradients (PCG). Such methods generally require an efficient method to compute matrix-vector products $\mathrm{G}_{\mathcal{R}} \cdot \Delta\eta_{\mathcal{R}}$, which we can provide using the implicit method, based on (12), for multiplication by G. Also, to obtain rapid convergence, it is important to provide an efficient *preconditioner*, which approximates $(\mathrm{G}_{\mathcal{R}})^{-1}$. For our preconditioner, we use $(\mathrm{G}^{-1})_{\mathcal{R}}$,[5] implemented using an implicit method for multiplication by G^{-1} described in Appendix A. In this way, we obtain iterative methods that have $\mathcal{O}(nw^3)$ complexity per iteration. Using the PCG method, we find that a small number of iterations (typically, 3-12) is sufficient to obtain a good approximation to each Newton step, leading to rapid convergence in Newton's method, but with significantly less overall computation for larger values of w than is required using direct methods.

C. Greedy Model Thinning

In this section, we propose a simple greedy strategy for *thinning* a GMRF model. This entails selecting edges of the graph which correspond to weak statistical interactions between variables and pruning these edges from the GMRF by information projection. The quantity we use to measure the strength of interaction between x_u and x_v is the *conditional mutual information* [38],

$$I_{u,v}(p) \triangleq \mathbb{E}_p \left\{ \log \frac{p(\mathrm{x}_u, \mathrm{x}_v | \mathrm{x}_{\setminus\{u,v\}})}{p(\mathrm{x}_u|\mathrm{x}_{\setminus\{u,v\}})p(\mathrm{x}_v|\mathrm{x}_{\setminus\{u,v\}})} \right\} = -\frac{1}{2}\log\left(1 - \frac{\mathrm{J}_{u,v}^2}{\mathrm{J}_{u,u}\mathrm{J}_{v,v}}\right) \geq 0,$$

which is the average mutual information between x_u and x_v after conditioning on the other variables $\mathrm{x}_{\setminus\{u,v\}}$. In GMRFs, we can omit edge $\{u,v\}$ from $\mathcal{G}$, without any modeling error, if and only if x_u and x_v are conditionally independent given $\mathrm{x}_{\setminus\{u,v\}}$, that is, if and only if $I_{u,v}(p) = 0$. This suggests using the *value* of $I_{u,v}(p)$, which is tractable to compute, to select edges $\{u,v\} \in \mathcal{E}$ to remove. To motivate this idea further, we note that $I_{u,v}(p)$ is closely related to the information loss resulting from removing edge $\{u,v\}$ from $\mathcal{G}$ by information projection. Let $\mathcal{G} \setminus \{u,v\} = (V, \mathcal{E} \setminus \{u,v\})$ denote the sub-graph of $\mathcal{G}$ with edge $\{u,v\}$ removed and let $\mathcal{K}$ denote the complete graph on V. Then, observing that $\mathcal{G} \setminus \{u,v\}$ is a sub-graph of $\mathcal{K} \setminus \{u,v\}$, we have, by the Pythagorean relation with respect to $p_{\mathcal{K}\setminus\{u,v\}}$,

$$\begin{aligned} D(p\|p_{\mathcal{G}\setminus\{u,v\}}) &= D(p\|p_{\mathcal{K}\setminus\{u,v\}}) + D(p_{\mathcal{K}\setminus\{u,v\}}\|p_{\mathcal{G}\setminus\{u,v\}}) \\ &= I_{u,v}(p) + D(p_{\mathcal{K}\setminus\{u,v\}}\|p_{\mathcal{G}\setminus\{u,v\}}) \\ &\geq I_{u,v}(p) \end{aligned}$$

where we have used $D(p\|p_{\mathcal{K}\setminus\{u,v\}}) = I_{u,v}(p)$ and $D(p_{\mathcal{K}\setminus\{u,v\}}\|p_{\mathcal{G}\setminus\{u,v\}}) \geq 0$. Thus, $I_{u,v}(p)$ is a lower-bound on the information loss $D(p\|p_{\mathcal{G}\setminus\{u,v\}})$. Moreover, for $p \in \mathcal{F}(\mathcal{G})$ having a small value of $I_{u,v}(p)$, we find that $D(p_{\mathcal{K}\setminus\{u,v\}}\|p_{\mathcal{G}\setminus\{u,v\}})$ tends to be small relative to $I_{u,v}(p)$ so that $I_{u,v}(p)$ then provides a good estimate of $D(p\|p_{\mathcal{G}\setminus\{u,v\}})$. In other words, removing edges with small conditional mutual information is roughly equivalent to picking those edges to remove that result in the least modeling error.

We use the following greedy approach to thin a GMRF defined on $\mathcal{G}$. Let $\delta > 0$ specify the tolerance on conditional mutual information for removal of an edge. We compute $I_{u,v}(p)$ for all edges $\{u,v\} \in \mathcal{E}$ and select a subset of edges $\mathcal{R} \subset \mathcal{E}$ with $I_{u,v}(p) < \delta$ to remove. The information projection to the sub-graph $\mathcal{S} = (V, \mathcal{E} \setminus \mathcal{R})$ is then computed using our ME method as described in Section III-B (relative to a chordal super-graph of $\mathcal{G}$). Because the values of $I_{u,v}$ in this information projection will generally differ from their prior values, we may continue to thin the GMRF until all the remaining edges have $I_{u,v} > \delta$. Also, by limiting the number of edges removed at each step, it is possible to take into account the effect of removing the weakest edges before selecting which other edges to remove, which can help reduce the overall information loss.

IV. Recursive Cavity Modeling

We now flesh out the details of RCM. In Section IV-A, we specify the hierarchical tree representation of the GMRF that we use, and in Section IV-B, we define *information forms* and the three basic operations we use: composition, elimination and model reduction. These forms and operators are the components we use to build our

[5]To motivate this preconditioner, we note that $(\mathrm{G}_{\mathcal{R}})^{-1}$ is given by the *Schur complement* $\mathrm{H}_{\mathcal{R}} - \mathrm{H}_{\mathcal{R},\mathcal{S}}\mathrm{H}_{\mathcal{S}}^{-1}\mathrm{H}_{\mathcal{S},\mathcal{R}}$ with respect to $\mathrm{H} \triangleq \mathrm{G}^{-1} = \mathrm{cov}\{\phi(\mathrm{x})\}$. Hence, our preconditioner $\mathrm{H}_{\mathcal{R}} = (\mathrm{G}^{-1})_{\mathcal{R}}$ arises by neglecting the intractable term $\mathrm{H}_{\mathcal{R},\mathcal{S}}\mathrm{H}_{\mathcal{S}}^{-1}\mathrm{H}_{\mathcal{S},\mathcal{R}}$, which is a good approximation if the correlation $\mathrm{H}_{\mathcal{S},\mathcal{R}}$ is weak relative to $\mathrm{H}_{\mathcal{R}}$ and $\mathrm{H}_{\mathcal{S}}$.

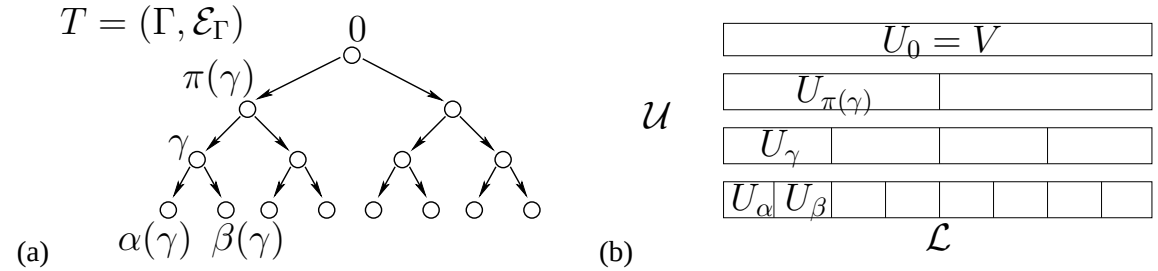

Fig. 2. (a) The tree $T = (\Gamma, \mathcal{E}_\Gamma)$ based on (b) the collection $\mathcal{U}$ of nested subsets of vertices V of the underlying graph $\mathcal{G}$.

two-pass, recursive, message-passing inference algorithm on the hierarchical tree. First, as described in Section IV-C, we perform an upward pass on the tree which constructs cavity models. Next, as described in Section IV-D, we perform a downward pass on the tree which constructs blanket models and also estimates marginal variances and edge-wise covariances in the GMRF. Lastly, in Section IV-E, we describe a hierarchical preconditioner, using the cavity models computed by RCM, and an iterative estimation algorithm that computes the means for all vertices of the GMRF.

Before we proceed, we define some basic notation with respect to the graph $\mathcal{G} = (V, \mathcal{E})$ describing the Markov structure of x. Given $U \subset V$, let $U' \triangleq V \backslash U$ denote the set complement of U in V and let $\partial U \triangleq \{v \in U' | (u, v) \in \mathcal{E}\}$ denote the *blanket* of U in $\mathcal{G}$. Also, $\partial U' \triangleq \partial(U')$ is the *surface* of U and $U^\circ \triangleq U \setminus \partial U'$ is its *interior*. These definitions are illustrated in Fig. 3(a), (b) and (c).

A. Hierarchical Tree Structure

We begin by requiring that the graphical model is recursively dissected into a hierarchy of nested subfields as indicated in Fig. 2. First, we describe a "bottom-up" construction. Let the set V be partitioned into a collection $\mathcal{L}$ of many small, disjoint subsets chosen so as to induce low-diameter, connected subgraphs in $\mathcal{G}$ over which exact inference is tractable. These small sets of vertices are recursively *merged* into larger and larger subfields until only the entire set V remains. Only adjacent subfields are merged so as to induce connected subgraphs. Also, merging should (ideally) keep the diameter of these connected subgraphs as small as possible. To simplify presentation only, we assume that subfields are merged two at a time. This generates a collection $\mathcal{U} \subset 2^V$ containing the smallest sets in $\mathcal{L}$ as well as each of the merged sets up to and including V. Alternatively, such a dissection can be constructed in a "top-down" fashion by recursively splitting the graph, and resulting sub-graphs, into roughly equal parts chosen so as to minimize the number of cut edges at each step. For instance, in 2D lattices this is simply achieved by performing an alternating series of vertical and horizontal cuts.

In any case, this recursive dissection of the graph defines a tree $T = (\Gamma, \mathcal{E}_\Gamma)$, in which each node $\gamma \in \Gamma$ corresponds to a subset $U_\gamma \in \mathcal{U}$ and with directed edges $\mathcal{E}_\Gamma$ linking each dissection cell to it immediate sub-cells. We let $\pi(\gamma)$ denote the *parent* of node γ in this tree. Also, the *children* of γ are denoted $\pi^{-1}(\gamma) = \{\alpha(\gamma), \beta(\gamma)\}$, or more simply $\{\alpha, \beta\}$ where γ has been explicitly specified. The following vertex sets are defined for each $U_\gamma \in \mathcal{U}$ relative to the graph $\mathcal{G}$:

$$B_\gamma \triangleq \partial U_\gamma, \quad R_\gamma \triangleq \partial U'_\gamma. \tag{15}$$

As seen in Figs. 3(a), (b) and (c), the blanket B_γ is the "outer" boundary of U_γ while the surface R_γ is its "inner" boundary, and either serves as a separator between U_γ and U'_γ. Also, the following separators are used in RCM:

$$S^\gamma \triangleq R_\alpha \cup R_\beta, \quad S_\alpha \triangleq B_\gamma \cup R_\beta, \quad S_\beta \triangleq B_\gamma \cup R_\alpha. \tag{16}$$

The separator S^γ, used in the RCM upward pass, is the union of the surfaces of the two children of a subfield (see Fig. 3(d) and (e)). The separators S_α and S_β, used in the RCM downward pass, are each the union of its parent's blanket and its sibling's surface (see Fig. 3(d) and (f)).

These separators define a Markov tree representation, with respect to T, of the original GMRF defined on $\mathcal{G}$ [24]: For each leaf γ of T define the state vector $\mathrm{x}_\gamma \triangleq \mathrm{x}_{U_\gamma}$. For each non-leaf γ let $\mathrm{x}_\gamma \triangleq \mathrm{x}_{S^\gamma}$. By construction, each S^γ is a separator of the graph, that is, the subfields U_α, U_β and U'_γ are mutually separated by S^γ. Hence, all conditional independence relations required by the Markov tree are satisfied by the underlying GMRF. However, we are interested in the large class of models for which exact inference on such a Markov tree representation is not feasible because of the large size of some of the separators. As discussed in Section II-C, we instead perform reduced-order *modeling* of these variables, corresponding to a thinned, tractable graphical model on each separator.

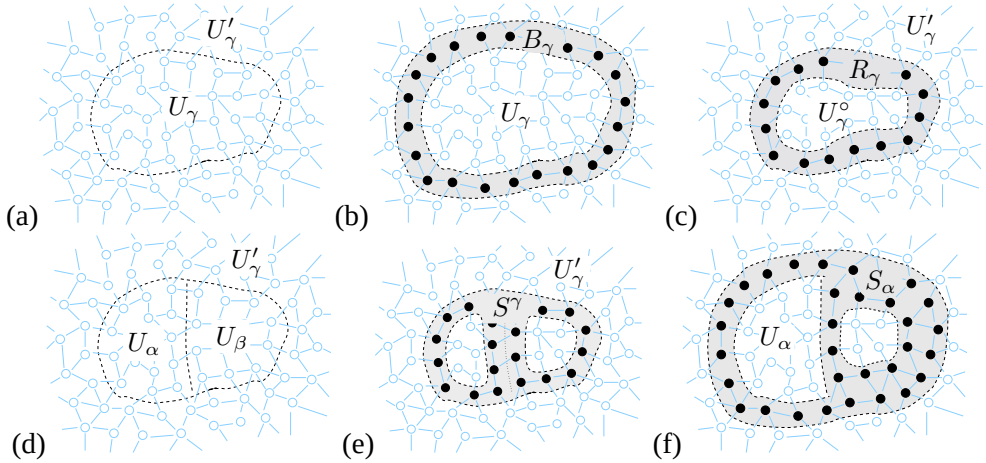

Fig. 3. Illustrations of the graph $\mathcal{G}$ of a GMRF and of our notation used to indicate subfields: (a) the subfield U_γ and its complement U'_γ; (b) the blanket $B_\gamma = \partial U_\gamma$; (b) the interior U°_γ and surface $R_\gamma = \partial U'_\gamma$; (d) partitioning of U_γ into sub-cells U_α and U_β; (e) separator $S^\gamma = R_\alpha \cup R_\beta$; (f) separator $S_\alpha = R_\beta \cup B_\gamma$.

B. Information Kernels

In the sequel, we let $(\mathrm{h}_U, \mathrm{J}_U)$, where $U \subset V$, $\mathrm{h}_U \in \mathbb{R}^{|U|}$ and $\mathrm{J}_U \in \mathbb{R}^{|U|\times|U|}$ is symmetric positive definite, represent the *information kernel* $f_U : \mathbb{R}^{|U|} \to \mathbb{R}_+$ defined by:

$$f_U(\mathrm{x}_U; \mathrm{h}_U, \mathrm{J}_U) = \exp\{-\frac{1}{2}\mathrm{x}_U^T \mathrm{J}_U \mathrm{x}_U + \mathrm{h}_U^T \mathrm{x}_U\} \tag{17}$$

The subscript U indicates the support of the information kernel, and of the matrices h_U and J_U. Generally, f_U corresponds (after normalization) to a density over the variables x_U parameterized by h_U and J_U. In RCM, the set U is typically a separator of the graph, and h_U and J_U are approximations to the exact distribution in question so that J_U is sparse. We also use matrices $\mathrm{J}_{U,W}$, where $U, W \subset V$ and $\mathrm{J}_{U,W} \in \mathbb{R}^{|U|\times|W|}$, to represent the function

$$f_{U,W}(\mathrm{x}_U, \mathrm{x}_W; \mathrm{J}_{U,W}) = \exp\{-\mathrm{x}_U^T \mathrm{J}_{U,W} \mathrm{x}_W\}, \tag{18}$$

which describes the interaction between subfields U, W. We adopt the following notations: Let $\mathrm{h}_U[W]$ denote the sub-vector of h_U indexed by $W \subset U$. Likewise, $\mathrm{J}_U[W_1, W_2]$ denotes the sub-matrix of J_U indexed by $W_1 \times W_2$ and we write $\mathrm{J}_U[W] = \mathrm{J}_U[W, W]$ to indicate a principle sub-matrix.

Given two disjoint subfield models $(\mathrm{h}_{U_1}, \mathrm{J}_{U_1})$ and $(\mathrm{h}_{U_2}, \mathrm{J}_{U_2})$ and the interaction J_{U_1,U_2} we let $(\mathrm{h}_U, \mathrm{J}_U) = (\mathrm{h}_{U_1}, \mathrm{J}_{U_1}) \oplus \mathrm{J}_{U_1,U_2} \oplus (\mathrm{h}_{U_2}, \mathrm{J}_{U_2})$ denote the joint model on $U = U_1 \cup U_2$ defined by

$$\mathrm{h}_U = \begin{pmatrix} \mathrm{h}_{U_1} \\ \mathrm{h}_{U_2} \end{pmatrix}, \quad \mathrm{J}_U = \begin{pmatrix} \mathrm{J}_{U_1} & \mathrm{J}_{U_1,U_2} \\ \mathrm{J}_{U_1,U_2}^T & \mathrm{J}_{U_2} \end{pmatrix} \tag{19}$$

which corresponds to multiplication of information kernels or addition of their information forms.

Given an information form $(\mathrm{h}_U, \mathrm{J}_U)$ and $D \subset U$ to be eliminated, we let $(\hat{\mathrm{h}}_S, \hat{\mathrm{J}}_S) = \hat{\Pi}_S(\mathrm{h}_U, \mathrm{J}_U) \equiv \hat{\Pi}_{\setminus D}(\mathrm{h}_U, \mathrm{J}_U)$ denote[6] the operation of *Gaussian Elimination* (GE) defined by $S = U \setminus D$ and

$$\begin{aligned} \hat{\mathrm{h}}_S &= \mathrm{h}_U[S] - \mathrm{J}_U[S, D]\mathrm{J}_U[D]^{-1}\mathrm{h}_U[D] \\ \hat{\mathrm{J}}_S &= \mathrm{J}_U[S] - \mathrm{J}_U[S, D]\mathrm{J}_U[D]^{-1}\mathrm{J}_U[D, S] \end{aligned} \tag{20}$$

The matrix $\hat{\mathrm{J}}_S$ is the *Schur complement* of the sub-matrix $\mathrm{J}_U[D]$ in J_U. Straightforward manipulations lead to the following well-known result:

$$(\hat{\mathrm{J}}_S)^{-1} = (\mathrm{J}_U^{-1})[S] \;\text{ and }\; (\hat{\mathrm{J}}_S)^{-1}\hat{\mathrm{h}}_S = (\mathrm{J}_U^{-1}\mathrm{h}_U)[S]. \tag{21}$$

Thus, the information form $(\hat{\mathrm{h}}_S, \hat{\mathrm{J}}_S)$ corresponds to the marginal on S with respect to the model $(\mathrm{h}_U, \mathrm{J}_U)$. Also, GE may be implemented *recursively* as follows: given an *elimination order* $(d_1, \ldots, d_n)$ of the elements in D, compute (20) as $\hat{\Pi}_{\setminus d_n} \cdots \hat{\Pi}_{\setminus d_1}(U, \mathrm{h}_U, \mathrm{J}_U)$, that is, by eliminating one variable at a time. Note also that only those

[6]Two notations are introduced, as in some cases $S = U \setminus D$ is given explicitly, while in others it is only implicitly specified in terms of U and D.

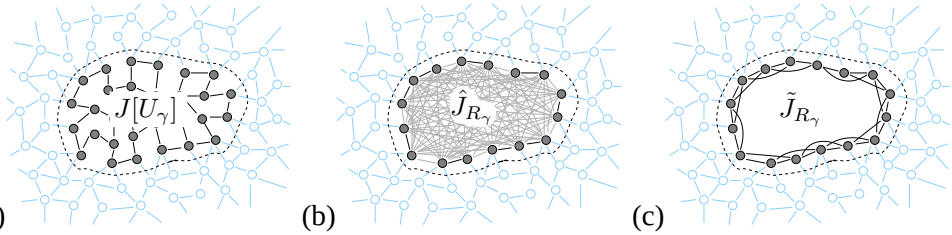

Fig. 4. Initialization of a cavity model for a small subfield $U_\gamma \in \mathcal{L}$, corresponding to a leaf of T: (a) the initial subfield model $J[U_\gamma]$, a sub-matrix of J; (b) the cavity model $\hat{J}_{R_\gamma} = \hat{\Pi} J[U_\gamma]$ after Gaussian elimination of the interior variables $U_\gamma^\circ = U_\gamma \setminus R_\gamma$; (c) the final thinned cavity model $\tilde{J}_{R_\gamma} = \tilde{\Pi}_\delta \hat{J}_{R_\gamma}$ defined on the surface R_γ of subfield U_γ.

entries of h_U and J_U indexed by ∂D are modified by GE. Hence, GE is a *localized* operation within the graphical representation of the GMRF as suggested by Figs. 4(a) and 4(b). However, eliminating D typically has the effect of causing $\hat{\mathrm{J}}_S[\partial D]$ to become full as shown in Fig. 4(b). This creation of *fill* can spoil the graphical model so that recursive GE becomes intractable with worst-case cubic complexity in dense graphs.

Given an information matrix J_U we denote the result of *model-order reduction* by $\tilde{\mathrm{J}}_U = \tilde{\Pi}_\delta \mathrm{J}_U$. The model reduction algorithm in Section III requires specifying a parameter δ which controls the tolerance on conditional mutual information for the removal of an edge. The procedure then determines which edges in the graph corresponding to J_U to remove and determines the projection to this thinned graph. This projection preserves variances and edge-wise cross-covariances on the thinned graph, which is equivalent to $\hat{\Pi}_C \tilde{\mathrm{J}}_U = \hat{\Pi}_C \mathrm{J}_U$ for each clique $C \subset U$ of the thinned graph.

In the following sections, we first develop our two-pass approximate inference procedure, focusing on calculation of just the information matrices, which are all independent of h. Then, we provide additional calculations involving h and $\hat{\mathrm{x}}$, presented as a separate two-pass procedure which then serves as a preconditioner in an iterative method.

C. Upward Pass: Cavity Model Propagation

In this first step, messages are passed from the leaves of the tree $\mathcal{L}$ up towards the root V. These upward messages take the form of *cavity models*, encoding conditional statistics of variables lying in the surfaces of given subfields. To be precise, each cavity model, represented by an information matrix $\tilde{\mathrm{J}}_{R_\gamma}$, approximates a conditional density $p(\mathrm{x}_{R_\gamma} | \mathrm{x}_{B_\gamma} = 0)$ so that $\tilde{\mathrm{J}}_{R_\gamma}$ is a tractable, thin matrix.

1) Leaf-Node Initialization: For each $U_\gamma \in \mathcal{L}$ we initialize a cavity model as follows: We begin with the local information matrix $\mathrm{J}[U_\gamma]$ as depicted in Fig. 4(a). This specifies the conditional density $p(\mathrm{x}_{U_\gamma} | \mathrm{x}_{B_\gamma} = 0) \propto f(\mathrm{x}_{U_\gamma}; 0, \mathrm{J}[U_\gamma])$. We then eliminate all variables within the interior of U_γ by Gaussian elimination: $\hat{\mathrm{J}}_{R_\gamma} = \hat{\Pi}_{R_\gamma} \mathrm{J}[U_\gamma]$. This has the effect of deleting all nodes in the interior of U_γ and updating the matrix parameters on the surface. As indicated in Fig. 4(b), this also induces fill within the information matrix. To ensure tractable computations in later stages, we thin this model: $\tilde{\mathrm{J}}_{R_\gamma} = \tilde{\Pi}_\delta \hat{\mathrm{J}}_{R_\gamma}$, yielding a reduced-order cavity model, as shown if Fig. 4(c), for each subfield $U_\gamma \in \mathcal{L}$. Then we are ready to proceed up the tree growing larger cavity models from smaller ones.

2) Growing Cavity Models: Let $U_\gamma \subsetneq V$ be a subfield in $\mathcal{U}$ where we have already constructed the two cavity models for R_α and R_β as depicted in Fig. 5(a). Then, we construct the cavity model for U_γ as follows:

a) Join Cavity Models: First, we form the composition of the two sub-cavity models as indicated in Fig. 5(b): $\tilde{\mathrm{J}}_{S^\gamma} = \tilde{\mathrm{J}}_{R_\alpha} \oplus \mathrm{J}[R_\alpha, R_\beta] \oplus \tilde{\mathrm{J}}_{R_\beta}$. Note that $S^\gamma = R_\alpha \cup R_\beta$ is a superset of R_γ.

b) Variable Elimination: Next, we must eliminate the extra variables $D^\gamma \triangleq S^\gamma \setminus R_\gamma$, to obtain the marginal information matrix $\hat{\mathrm{J}}_{R_\gamma} = \hat{\Pi}_{R_\gamma} \tilde{\mathrm{J}}_{S^\gamma}$. To ensure scalability, rather than eliminating all variables at once, we eliminate variables recursively beginning with those farthest from the surface and working our way towards the surface. This is an efficient computation thanks to model reductions performed previously in U_α and U_β.

c) Model Thinning: This preceding elimination step induces fill "across" the cavity (Fig. 5(c)). Hence, to maintain tractability as we continue, we perform model-order reduction yielding $\tilde{\mathrm{J}}_{R_\gamma} = \tilde{\Pi}_\delta \hat{\mathrm{J}}_{R_\gamma}$ which is the desired reduced-order cavity model represented in Fig. 5(d). This projection step requires that we compute moments of the graphical model specified by $\hat{\mathrm{J}}_{R_\gamma}$. Thanks to model thinning in the subtree of T rooted at γ, these moments can be computed efficiently.

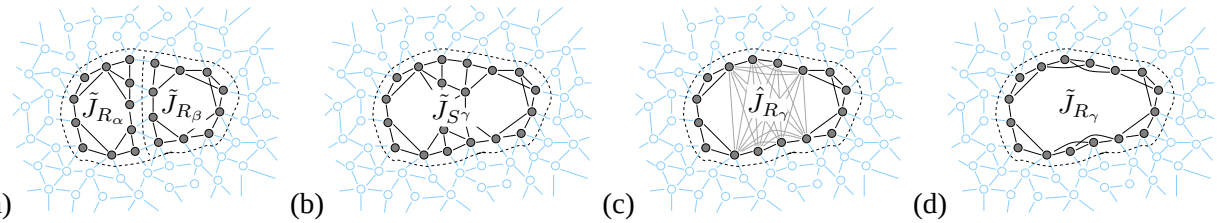

Fig. 5. Recursive construction of a cavity model: (a) cavity models $\tilde{J}_{R_\alpha}, \tilde{J}_{R_\beta}$ of sub-cells U_α, U_β; (b) joined cavity model $\tilde{J}_{S^\gamma} = \tilde{J}_{R_\alpha} \oplus J[R_\alpha, R_\beta] \oplus \tilde{J}_{R_\beta}$ defined on separator $S^\gamma = R_\alpha \cup R_\beta$; (c) the cavity model $\hat{J}_{R_\gamma} = \hat{\Pi} \tilde{J}_{S^\gamma}$ after Gaussian elimination of variables $S^\gamma \setminus R_\gamma$; (d) the final thinned cavity model $\tilde{J}_{R_\gamma} = \tilde{\Pi}_\delta \hat{J}_{R_\gamma}$ defined on the surface R_γ of subfield U_γ.

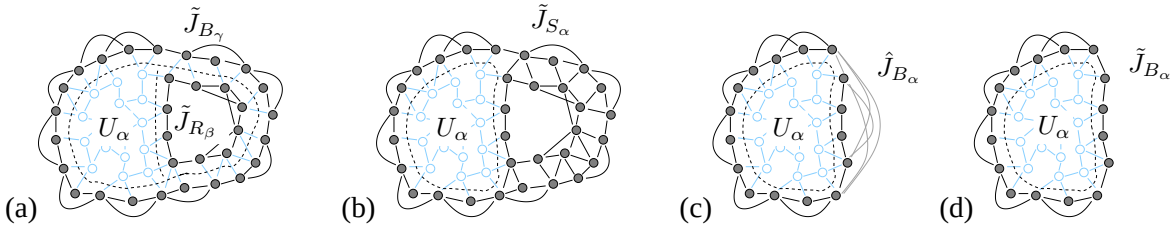

Fig. 6. Recursive construction of a blanket model: (a) the cavity model $\tilde{J}_{R_\beta}$ of the sibling subfield U_β and the blanket model $\tilde{J}_{B_\gamma}$ of the parent; (b) joined cavity/blanket model $\tilde{J}_{S_\alpha} = \tilde{J}_{R_\beta} \oplus J[R_\beta, B_\gamma] \oplus \tilde{J}_{B_\gamma}$ defined on the separator $S_\alpha = R_\beta \cup B_\gamma$; (c) the blanket model $\hat{J}_{B_\gamma} = \hat{\Pi} \tilde{J}_{S_\alpha}$ after Gaussian elimination of variables $S_\alpha \setminus B_\gamma$; (d) the final thinned blanket model $\tilde{J}_{B_\alpha} = \tilde{\Pi}_\delta \hat{J}_{B_\alpha}$ defined on the blanket B_α of subfield U_α.

D. Downward Pass: Blanket Model Propagation

The next, downward pass on the tree T involves messages in the form of *blanket models*, that is, graphical models encoding the conditional statistics of variables lying in the blanket of some subfield. Each subfield's blanket model is a concise summary of the complement of that subfield sufficient for near-optimal inference within the subfield. Specifically, the blanket model $\tilde{\mathrm{J}}_{B_\gamma}$ is a tractable approximation of the conditional model $p(\mathrm{x}_{B_\gamma} | \mathrm{x}_{R_\gamma} = 0)$.

1) Root-Node Initialization: Note that the blanket of V is the empty set so that a blanket model is not required for the root of T. As we move down to the children $U_{\alpha(0)}$ and $U_{\beta(0)}$, we note that $B_{\alpha(0)} = R_{\beta(0)}$ and, hence, a blanket model for $U_{\alpha(0)}$ is given by the cavity model for $U_{\beta(0)}$, which was computed in the upward pass. Hence, we already have blanket models for $U_{\alpha(0)}$ and $U_{\beta(0)}$ and are ready to build blanket models for their descendents.

2) Shrinking Blanket Models: Suppose that we already have the blanket model for U_γ as represented in Fig. 6(a). Then, we can construct the blanket model for the child U_α as follows:

a) Joining Blanket and Sub-Cavity Model: First, we form the composition of the blanket model defined on B_γ with the cavity model defined on R_β (from the sibling of α) as shown in Fig. 6(b): $\tilde{\mathrm{J}}_{S_\alpha} = \tilde{\mathrm{J}}_{B_\gamma} \oplus \mathrm{J}[B_\gamma, R_\beta] \oplus \tilde{\mathrm{J}}_{R_\beta}$. Note that $S_\alpha = B_\gamma \cup R_\beta$ is a superset of B_α.

b) Variable Elimination: Next, we eliminate all variables in $D_\gamma \triangleq S_\alpha \setminus B_\alpha$, yielding $\hat{\mathrm{J}}_{B_\alpha} = \hat{\Pi}_{B_\alpha} \tilde{\mathrm{J}}_{S_\alpha}$. To ensure scalable computations, we again perform variable elimination recursively, starting with vertices farthest from the blanket and working our way towards U_α. The result is depicted in Fig. 6(c).

c) Model Thinning: Lastly, we thin this resulting blanket model: $\tilde{\mathrm{J}}_{B_\alpha} = \tilde{\Pi}_\delta \hat{\mathrm{J}}_{B_\alpha}$, yielding our reduced-order blanket model for subfield U_α (Fig. 6(d)). The blanket model for U_β is computed in an identical manner with the roles of α and β reversed.

3) Leaf-Node Marginalization: Once we have constructed a blanket model for each of the smallest subfields $U_\gamma \in \mathcal{L}$, we can join this model with the conditional model for the enclosed subfield (that is the model used to seed the upwards pass), to obtain a graphical model approximation of the (zero-mean) marginal density $p(\mathrm{x}_{\bar{U}_\gamma})$ on $\bar{U}_\gamma \triangleq U_\gamma \cup B_\gamma$, given in information form by $\tilde{\mathrm{J}}_{\bar{U}_\gamma} = \mathrm{J}[U_\gamma] \oplus \mathrm{J}[U_\gamma, B_\gamma] \oplus \tilde{\mathrm{J}}_{B_\gamma}$. Inverting each of these localized models, that is, computing $\tilde{\mathrm{P}}_{\bar{U}_\gamma} = (\tilde{\mathrm{J}}_{\bar{U}_\gamma})^{-1}$, yields variances of all variables and covariances for each edge of $\mathcal{G}$.

E. An RCM-Preconditioner for Iterative Estimation

In the preceding sections, we have described a recursive algorithm for constructing a hierarchical collection of cavity and blanket models, described by thin information matrices. In this section, we describe how to extend these computations to compute the estimates $\hat{\mathrm{x}}$ solving $\mathrm{J}\hat{\mathrm{x}} = \mathrm{h}$. We begin by describing a two-pass algorithm, based on the cavity models computed previously, which computes an approximation of $\hat{\mathrm{x}}$, and then describe an iterative procedure, using the two-pass algorithm as a *preconditioner*, that iteratively refines the estimate.

1) Upward-Pass: We specify a recursive algorithm that works its way up the tree, computing a potential vector h_{R_γ} at each node $\gamma \in \Gamma$ of the dissection tree. Let $\tilde{\mathrm{J}}_{R_\gamma}$ denote the cavity models computed previously by the RCM upward pass. For each leaf-node, we solve $J[U_\gamma] \cdot \mathrm{x}_{U_\gamma} = \mathrm{h}[U_\gamma]$ for x_{U_γ} and then compute $\mathrm{h}_{R_\gamma} = \tilde{\mathrm{J}}_{R_\gamma} \cdot \mathrm{x}_{U_\gamma}$. At each non-leaf node, we compute h_{R_γ} as follows:

a) Join: Form the composite model $(\mathrm{h}_{S^\gamma}, \mathrm{J}_{S^\gamma}) = (\mathrm{h}_{R_\alpha}, \tilde{\mathrm{J}}_{R_\alpha}) \oplus \mathrm{J}[R_\alpha, R_\beta] \oplus (\mathrm{h}_{R_\beta}, \tilde{\mathrm{J}}_{R_\beta})$, where $S^\gamma = R_\alpha \cup R_\beta$, by joining the two cavity models from the children.

b) Sparse Solve: Given this joint model, we solve $\mathrm{J}_{S^\gamma} \cdot \mathrm{x}_{S^\gamma} = \mathrm{h}_{S^\gamma}$ using direct methods, which is tractable because J_{S^γ} is a thin, sparse matrix.[7]

c) Sparse Multiply: Finally, we compute the potential vector $\mathrm{h}_{R_\gamma} = \tilde{\mathrm{J}}_{R_\gamma} \cdot \mathrm{x}_{S^\gamma}[R_\gamma]$, which is a tractable computation because $\tilde{\mathrm{J}}_{R_\gamma}$ is sparse.

2) Downward-Pass (Back-Substitution): Once the root node is reached, we have the information form $(\mathrm{h}_{S^0}, \mathrm{J}_{S^0}) = (\mathrm{h}_{R_{\alpha(0)}}, \tilde{\mathrm{J}}_{R_{\alpha(0)}}) \oplus \mathrm{J}[R_{\alpha(0)}, R_{\beta(0)}] \oplus (\mathrm{h}_{R_{\beta(0)}}, \tilde{\mathrm{J}}_{R_{\beta(0)}})$ at the top-level separator of the dissection tree, which is an approximate model for the marginal distribution $p(\mathrm{x}_{S^0})$. Hence, we can compute an approximation for the means $\hat{\mathrm{x}}_{S^0}$ by solving $\mathrm{J}_{S^0} \cdot \hat{\mathrm{x}}_{S^0} = \mathrm{h}_{S^0}$. Conditioning on this estimate, we can then recurse back down the tree filling in the missing values of $\hat{\mathrm{x}}$ along each separator, thereby propagating estimates down the tree. In this downward pass, each node below the root of the tree receives an estimate $\hat{\mathrm{x}}_{R_\gamma}$ of the variables in the surface R_γ of the corresponding subfield. Again using the model $(\mathrm{h}_{S^\gamma}, \mathrm{J}_{S^\gamma})$, formed by the upward computations, we interpolate into the subfield, computing $\hat{\mathrm{x}}_{D^\gamma}$ where $D^\gamma = S^\gamma \setminus R_\gamma$, by solution of the linear system of equations

$$\begin{aligned} \mathrm{J}_{D^\gamma} \cdot \hat{\mathrm{x}}_{D^\gamma} &= \mathrm{h}_{D^\gamma} \\ \mathrm{J}_{D^\gamma} &\triangleq \mathrm{J}_{S^\gamma}[D^\gamma] \\ \mathrm{h}_{D^\gamma} &\triangleq \mathrm{h}_{S^\gamma}[D^\gamma] - \mathrm{J}_{S^\gamma}[D^\gamma, R_\gamma] \cdot \hat{\mathrm{x}}_{R_\gamma}. \end{aligned} \tag{22}$$

The estimate $\hat{\mathrm{x}}_{D^\gamma}$ is computed with respect to the approximation of $p(\mathrm{x}_{D^\gamma} | \hat{\mathrm{x}}_{R_\gamma}) \approx f(\mathrm{x}_{D^\gamma}; \mathrm{h}_{D^\gamma}, \mathrm{J}_{D^\gamma})$ (after normalization), which is approximate because of the model thinning steps in RCM. Once the leaves of the tree are reached, the interior of each subfield is interpolated similarly, thus yielding a complete estimate $\hat{\mathrm{x}}$.

3) Richardson Iteration: The preceding two-pass algorithm may be used to compute an approximate solution of $\mathrm{J}\mathrm{x} = \mathrm{b}$ for an arbitrary right-hand side b. The resulting estimate is linear in b and we denote this linear operator by M. Using M as a preconditioner[8], we compute a sequence of estimates $\{\hat{\mathrm{x}}^{(n)}\}$ defined by $\hat{\mathrm{x}}^{(0)} = 0$ and

$$\hat{\mathrm{x}}^{(n+1)} = \hat{\mathrm{x}}^{(n)} + \mathrm{M} \cdot (\mathrm{h} - \mathrm{J} \cdot \hat{\mathrm{x}}^{(n)}). \tag{23}$$

Let ρ denote the spectral radius of $\mathrm{I} - \mathrm{M}\mathrm{J}$. If $\rho < 1$ then $\hat{\mathrm{x}}^{(n)}$ converges to $\hat{\mathrm{x}} \triangleq \mathrm{J}^{-1}\mathrm{h}$ with $\|\hat{\mathrm{x}}^{(n)} - \hat{\mathrm{x}}\| \leq \rho^n \|\hat{\mathrm{x}}\|$. For small δ, this condition is met and we achieve rapid convergence to the correct means.

V. Applications in Remote Sensing

In this section we develop two applications of RCM in remote sensing: (1) interpolation of satellite altimetry measurements of sea-surface height, and (2) estimation of the surface of a large salt-deposit beneath the Gulf of Mexico. The purpose of these examples is to demonstrate that RCM scales well to very large problems while yielding estimates and error covariances that are close to those that would have resulted if exact optimal estimation had been performed instead. Although the specific statistical models used in these examples are perhaps over-simplified, the results that follow (which include space-varying measurement densities and hence space-variant estimation errors) do serve to demonstrate the applicability of RCM to very large spatial estimation problems.

[7]We use a sparse Cholesky factorization of $\tilde{\mathrm{J}}_{S^\gamma}$ and back-substitution based on h_{S^γ}. Also, some computation can be saved if we use an elimination order beginning with $S^\gamma \setminus R_\gamma$ because we only need to compute $\mathrm{x}_{S^\gamma}[R_\gamma]$ in the back-substitution.

[8]To implement $\mathrm{M} \cdot \mathrm{b}$ efficiently, we pre-compute and store calculations that do not depend on b. For instance, we compute a sparse Cholesky factorization for each $\tilde{J}_{R_\gamma}$ using a low-fill elimination order. This leads to an extremely fast preconditioner because only back-substitution steps are required each time we apply M to a different b vector.

A. Model Specifications

Throughout this section, we consider GMRFs of the form

$$p(\mathrm{x},\mathrm{y}) \propto \exp\left\{-\frac{1}{2}\left(\frac{\|\mathrm{Dx}\|^2}{\sigma_r^2}+\frac{\|\mathrm{y}-\mathrm{Cx}\|^2}{\sigma_d^2}\right)\right\} \tag{24}$$

where $\mathrm{x}\in\mathbb{R}^n$ represents the vector of field values at the vertices of a regular 2D lattice and $\mathrm{y}\in\mathbb{R}^m$ is a vector of local, noisy measurements of the underlying field at an irregular set of points scattered throughout the field. Here, $\|\mathrm{Dx}\|^2$ represent our prior for x, which serves to regularize the field, and the data-fidelity term $\|\mathrm{y}-\mathrm{Cx}\|^2$ represents our measurement model. We consider two prior models commonly used in image-processing. The thin-membrane (TM) model is defined such that each row d_k corresponds to an edge $\{u,v\}\in\mathcal{E}$, and has two non-zero components: $\mathrm{d}_{k,u}=+1$ and $\mathrm{d}_{k,v}=-1$. This gives a regularization term

$$\|\mathrm{Dx}\|^2=\sum_{\{u,v\}\in\mathcal{E}}(x_u-x_v)^2 \tag{25}$$

that penalizes gradients, favoring level surfaces. The thin-plate (TP) model is defined such that each row d_v corresponds to a vertex $v\in V$ and has non-zero components $\mathrm{d}_{v,v}=1$ and $\mathrm{d}_{v,u}=-\frac{1}{|N(v)|}$ for adjacent vertices $u\in N(v)$. This gives a regularization term

$$\|\mathrm{Dx}\|^2=\sum_{v\in V}\left(x_v-\tfrac{1}{|N(v)|}\sum_{u\in N(v)}x_u\right)^2 \tag{26}$$

that penalizes curvature, favoring flat surfaces. In general, the locations of the measurements y defines an irregular pattern with respect to the grid defined for x. Moreover, the location of individual measurements may fall between these grid points. For this reason each measurement y_t is modeled as the bilinear interpolation $\mathrm{c}_t\cdot\mathrm{x}$ of the four nearest grid points to the actual measurement location corrupted by zero-mean, white Gaussian noise: $y_t=\mathrm{c}_t\cdot\mathrm{x}+v_t$ where $v_t\sim N(0,\sigma_d^2)$. The posterior density $p(x|y)$ may be expressed in information form with parameters

$$\mathrm{J}=\frac{\mathrm{D}^T\mathrm{D}}{\sigma_r^2}+\frac{\mathrm{C}^T\mathrm{C}}{\sigma_d^2},\quad \mathrm{h}=\frac{\mathrm{C}^T\mathrm{y}}{\sigma_d^2}.$$

Thus, the fill-pattern of J (and hence the posterior Markov structure of x) is determined both by $\mathrm{D}^T\mathrm{D}$ and $\mathrm{C}^T\mathrm{C}$. In the TM model, $\mathrm{D}^T\mathrm{D}$ has non-zero off-diagonal entries only at those locations corresponding to nearest neighbors in the lattice. In the TP model, there are also additional connections between pairs of vertices that are two steps away in the square lattice, including diagonal edges. Finally, for each measurement y_k there is a contribution of $\mathrm{c}_k c_k^T$ to J, which creates edges between those four grid points closest to the location of measurement k. This results in a sparse J matrix where all edges are between nearby points in the lattice. Hence, we can apply RCM to the information model (h,J) to calculate approximations of the estimates $\hat{\mathrm{x}}_v(\mathrm{y})=\mathbb{E}\{\mathrm{x}_v|\mathrm{y}\}$ and error variances $\hat{\sigma}_v^2=\mathbb{E}\{(\mathrm{x}_v-\hat{\mathrm{x}}_v(\mathrm{y}))^2|\mathrm{y}\}$ for all vertices $v\in V$ and error covariances $\mathbb{E}\{(\mathrm{x}_u-\hat{\mathrm{x}}_u(\mathrm{y}))(\mathrm{x}_v-\hat{\mathrm{x}}_v(\mathrm{y}))\}$ for all edges $\{u,v\}\in\mathcal{E}$. In Appendix B, we also describe an iterative algorithm to estimate the model parameters σ_r and σ_d.

B. Sea-Surface Height Estimation

First, we consider the problem of performing near-optimal interpolation of satellite altimetry of sea-surface height anomaly (SSHA), measured relative to seasonal, space-variant mean-sea level.[9] We model SSHA by the thin-membrane model, which seems an appropriate choice as it favors a level sea-surface. We estimate SSHA at the vertices of an 800×2400 lattice covering latitudes between $\pm 60^\circ$ and a full 360° of longitude, which yields a resolution of $\frac{1}{5}^\circ$ in both latitude and longitude. The final world-wide estimates and associated error variances, obtained using RCM with $\delta=10^{-4}$ and model parameters $\sigma_r\approx 1\mathrm{cm}$ and $\sigma_d\approx 3.5\mathrm{cm}$, are displayed in Fig. 7. In this example, RCM requires about three minutes to execute, including run-time of both the cavity and blanket modeling procedures as well as the total run-time of the iterative procedure to compute the means. About 30 iterations are required to obtain a residual error $\|h-J\hat{\mathrm{x}}^{(k)}\|$ less than 10^{-4}, where each iteration takes 2-3 seconds.

[9]This data was collected by the Jason-1 satellite over a ten day period beginning 12/1/2004 and is available from the Jet Propulsion Laboratory `http://poodaac.jpl.nasa.gov`.

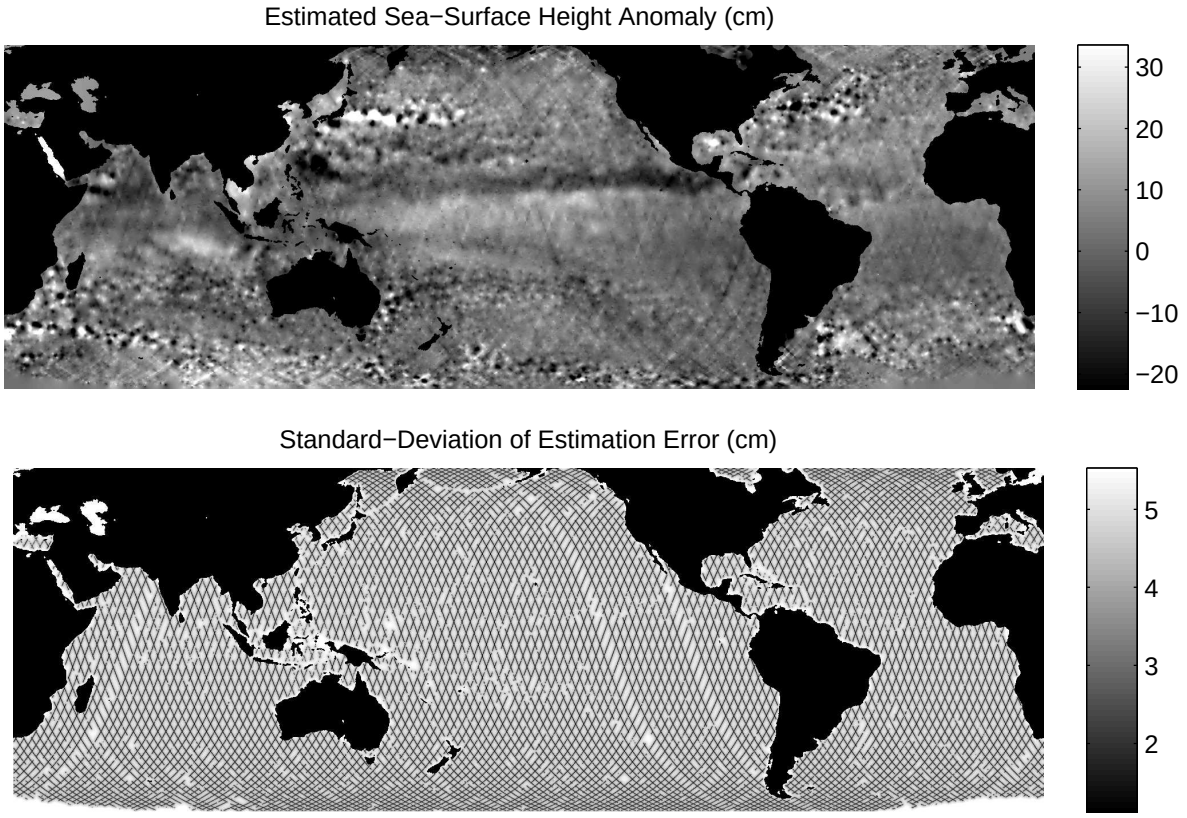

Fig. 7. Estimated sea-surface height anomaly and space-variant standard deviation of estimation error computing using RCM.

C. Salt-Top Estimation

Next, we consider the problem of estimating the "salt-top", that is, the top surface of a large salt deposit located several kilometers beneath the sea-floor somewhere in the Gulf of Mexico. The data for this estimation problem, provided courtesy of Shell International Exploration, Inc., consists of a large set of "picks" chosen by analysts while viewing cross-sections of seismic sounding data. Hence, these picks fall along straight line segments in latitude and longitude, and it is our goal to interpolate between these points. For this problem, we use the thin-plate model for the surface of the salt-deposit, which allows for undulations typically seen in the salt-top, with a 800×800 lattice at a resolution of 60 feet and with model parameters $\sigma_r \approx 12$ feet and $\sigma_d \approx 35$ feet. The final estimates and error variances are shown in Fig. 8. These results were obtained using RCM with a tolerance of $\delta = 10^{-4}$, which required about five minutes to run, including the total time required for iterative computation of the means. The run-times for the TP model are somewhat slower than for the TM model because the Markov blankets arising in the TP model are twice as wide as in the TM model, so the cavity and blanket models are more complex.

VI. Conclusion

We have presented a new, principled approach to approximate inference in very large GMRFs employing a recursive model reduction strategy based on information theoretic principles and have applied this method to perform near-optimal interpolation of sea-surface satellite altimetry. These results show the practical utility of the method for near-optimal, large-scale estimation. Several possible directions for further research are suggested by this work. First, the accuracy of RCM in applications such as that illustrated here provides considerable motivation for the development of a better theoretical understanding of its accuracy and stability. For instance, if it were possible to compute and propagate upper-bounds on the information divergence in RCM this would be very useful and may lead to a robust formulation. Although we have focused on examples using Gaussian prior models, we expect RCM will also prove useful in non-linear edge-preserving methods such as [44]. Although these methods use a non-Gaussian prior, their solution generally involves solving a sequence of Gaussian problems with an adaptive, space-variant process noise. Hence, RCM could be used as a fast computational engine in these methods. We also are interested to apply RCM to higher-dimensional GMRFs, such as arise in seismic and tomographic 3D estimation problems or

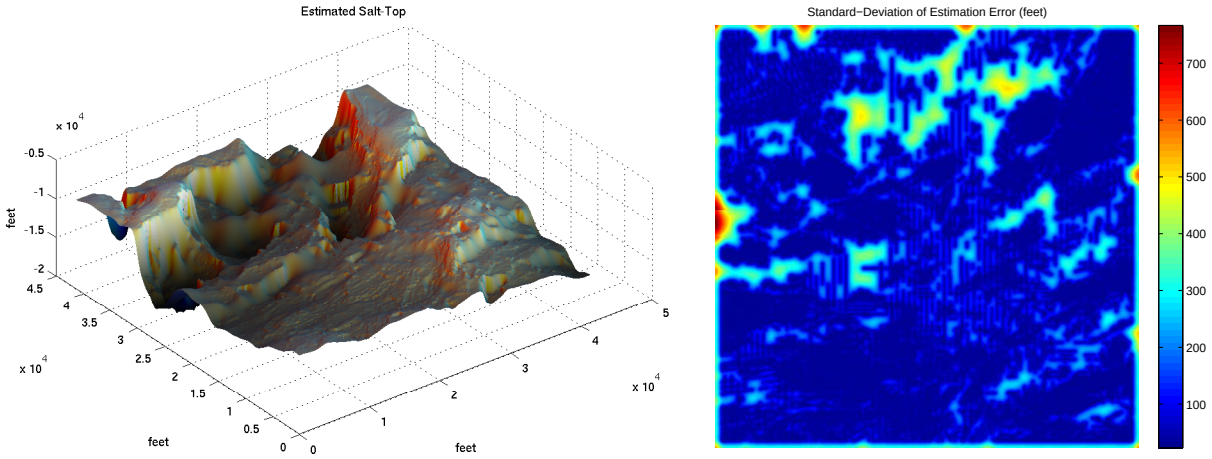

Fig. 8. Estimated salt-top and space-variant standard deviation of estimation error computing using RCM.

for filtering of dynamic GMRFs. We anticipate that it will be important to take advantage of the inherently parallel nature of RCM to address these computationally intensive applications. Another direction to explore is based on the rich class of multi-scale models, such as models having multi-grid or pyramidal structure. For example, the work in [11] demonstrates the utility and drawbacks of using multi-resolution models defined on *trees* to estimation of ocean height from satellite data. Such models allow one to capture long-distance correlations much more efficiently than a single-resolution nearest-neighbor model, but the tree structure used in [11] leads to artifacts at tree boundaries, something that RCM is able to avoid. This suggests the idea of enhancing models as in [11] by including new edges that eliminate these artifacts but that introduce cycles into these multi-resolution graphical models. However, if such models can be developed, RCM offers a principled, scalable approximate inference algorithm well-suited for solution of such hierarchical, multi-resolution models. Finally, while the specifics of this paper concern Gaussian models, the general framework we have outlined should apply more generally. This is especially pertinent for inference in discrete MRFs where computation of either the marginal distributions or the mode grows exponentially in the width of the graph [6], [10], which suggests developing counterparts to RCM for these problems.

APPENDIX

A. Recursive Inference Algorithm

In this appendix we summarize a recursive algorithm for computing the moments $\eta_{\mathcal{G}} = \Lambda(\theta_{\mathcal{G}})$ of a zero-mean, chordal GMRF. Also, by differentiating each step of this procedure, we obtain an algorithm to compute the first-order change in moment parameters $d\eta_{\mathcal{G}}$ due to a perturbation $d\theta_{\mathcal{G}}$. The complexity of both algorithms is $\mathcal{O}(nw^3)$, where n is the number of variables and w is the size of the largest clique. These algorithms are used as sub-routines in the model-reduction procedure described in Section III. In RCM, these methods are only used for thin cavity and blanket models and are tractable in that context.

Let $T = (\Gamma, \mathcal{E}_\Gamma)$ be a junction tree of $\mathcal{G}$. We obtain a directed version of T by selecting an arbitrary clique to be the root node and orienting the edges away from the root. For each non-root node γ, let $\pi(\gamma)$ denote its parent. We split each clique C_γ into a separator $S_\gamma = C_\gamma \cap C_{\pi(\gamma)}$ and the residual set $R_\gamma = C_\gamma \setminus C_{\pi(\gamma)}$. At the root, these are defined $S_\gamma = \emptyset$ and $R_\gamma = C_\gamma$. Now, we specify our recursive inference procedure. The input to this procedure is the sparse matrix J, which is defined over a chordal graph and parameterized by $\theta_{\mathcal{G}}$. The output is a sparse matrix P, defined on the same chordal graph, with elements specified by $\eta_{\mathcal{G}}$. In the differential form of the algorithm, we also have a sparse input dJ and sparse output dJ, corresponding to $d\theta_{\mathcal{G}}$ and $d\eta_{\mathcal{G}}$.

1) Upward Pass: For each node $\gamma \in \Gamma$ of the junction tree, starting from the the leaves of the tree and working upwards, we perform the following computations in the order shown:

$$\begin{aligned} \mathrm{Q}_\gamma &= (\mathrm{J}[R_\gamma])^{-1} \\ \mathrm{A}_\gamma &= -\mathrm{Q}_\gamma \cdot \mathrm{J}[R_\gamma, S_\gamma] \\ \mathrm{J}[S_\gamma] &\leftarrow \mathrm{J}[S_\gamma] + \mathrm{J}[S_\gamma, R_\gamma] \cdot \mathrm{A}_\gamma \end{aligned}$$

In the differential form of the algorithm, we also compute:

$$\begin{aligned} \mathrm{dJ}_\gamma &= -\mathrm{Q}_\gamma \cdot \mathrm{dJ}[R_\gamma] \cdot \mathrm{Q}_\gamma \\ \mathrm{dJ}_\gamma &= -(\mathrm{Q}_\gamma \cdot \mathrm{dJ}[R_\gamma] + \mathrm{dJ}_\gamma \cdot \mathrm{J}[R_\gamma]) \\ \mathrm{dJ}[S_\gamma] &\leftarrow \mathrm{dJ}[S_\gamma] + \mathrm{dJ}[S_\gamma, R_\gamma] \cdot \mathrm{A}_\gamma + \mathrm{J}[S_\gamma, R_\gamma] \cdot \mathrm{dJ}_\gamma \end{aligned}$$

The upward pass performs Gaussian elimination in J. At each step, the principle sub-matrices of J and dJ indexed by S_γ are overwritten, which propagates information to the ancestor's of node γ in the junction tree. Also, A_γ and Q_γ specify an equivalent downward model: $x[R_\gamma] = \mathrm{A}_\gamma \cdot x[S_\gamma] + w_\gamma$ where $w_\gamma \sim \mathcal{N}(0, \mathrm{Q}_\gamma)$. This downward model is re-used in the downward pass.

2) Downward Pass: For each node $\gamma \in \Gamma$ of the junction tree, starting from the root node and working down the tree, we perform the following calculations at each node γ of the dissection tree:

$$\begin{aligned} \mathrm{P}[R_\gamma, S_\gamma] &\leftarrow \mathrm{A}_\gamma \cdot \mathrm{P}[S_\gamma] \\ \mathrm{P}[S_\gamma, R_\gamma] &\leftarrow \mathrm{P}^T[R_\gamma, S_\gamma] \\ \mathrm{P}[R_\gamma] &\leftarrow \mathrm{P}[R_\gamma, S_\gamma] \cdot \mathrm{A}_\gamma^T + \mathrm{Q}_\gamma \end{aligned}$$

In the differential form of the algorithm, we also compute:

$$\begin{aligned} \mathrm{dJ}[R_\gamma, S_\gamma] &\leftarrow \mathrm{dJ}_\gamma \cdot \mathrm{P}[S_\gamma] + \mathrm{A}_\gamma \cdot \mathrm{dJ}[S_\gamma] \\ \mathrm{dJ}[S_\gamma, R_\gamma] &\leftarrow \mathrm{dJ}^T[R_\gamma, S_\gamma] \\ \mathrm{dJ}[R_\gamma] &\leftarrow \mathrm{dJ}[R_\gamma, S_\gamma] \cdot \mathrm{A}_\gamma^T + \mathrm{P}[R_\gamma, S_\gamma] \cdot \mathrm{dJ}_\gamma^T + \mathrm{dJ}_\gamma \end{aligned}$$

B. Parameter Estimation

We describe the expectation-maximization (EM) algorithm [45] we use for parameter estimation in both models described in Section V. These are exponential family models of the form

$$\begin{aligned} p_\theta(\mathrm{x}, \mathrm{y}) &\propto \exp\{\theta_1 \phi_1(\mathrm{x}) + \theta_2 \phi_2(\mathrm{x}, \mathrm{y})\} \\ \phi_1(\mathrm{x}) &= \tfrac{1}{2}\left(\|\mathrm{Dx}\|^2 + \epsilon \|\mathrm{x}\|^2\right) \\ \phi_2(\mathrm{x}, \mathrm{y}) &= \tfrac{1}{2}\|\mathrm{y} - \mathrm{Cx}\|^2 \end{aligned} \tag{27}$$

where ϕ_1 is the regularization term[10] and ϕ_2 is the data-fidelity term. We wish to select the parameters $\theta = (\theta_1, \theta_2) = -(\frac{1}{\sigma_r^2}, \frac{1}{\sigma_d^2})$ to maximize $\ell(\theta) = \int p_\theta(\mathrm{x}, \mathrm{y}) d\mathrm{x}$ for a given set of observations y. The EM algorithm is an iterative procedure that converges to a local maxima of $\ell(\theta)$ starting from an initial guess $\theta^{(0)}$. For $t = 1, 2, \ldots$, we alternate between (E-step) computing the conditional moments $\eta^{(t)} \triangleq \mathbb{E}_{\theta^{(t-1)}}\{\phi | \mathrm{y}\}$ given y and $\theta^{(t-1)}$, and (M-step) selecting the next parameter estimate $\theta^{(t)}$ to solve the equations $\mathbb{E}_{\theta^{(t)}}\{\phi\} = \eta^{(t)}$. In our model, the conditional moments are

$$\begin{aligned} \eta_1^{(t)} &= \phi_1(\hat{\mathrm{x}}) + \tfrac{1}{2}\left(\mathrm{tr}(\mathrm{DP}^{(t)}\mathrm{D}^T) + \epsilon \mathrm{tr}(\mathrm{P}^{(t)})\right) \\ \eta_2^{(t)} &= \phi_2(\hat{\mathrm{x}}, \mathrm{y}) + \tfrac{1}{2}\mathrm{tr}(\mathrm{CP}^{(t)}\mathrm{C}^T) \end{aligned} \tag{28}$$

where $\hat{\mathrm{x}}^{(t)} = \mathbb{E}\{\mathrm{x}|\mathrm{y}\}$ and $\mathrm{P}^{(t)} = \mathrm{cov}(\mathrm{x}|\mathrm{y})$ are computed given y and $\theta^{(t-1)}$. Due to sparsity of D, it is tractable to compute $\mathrm{D}\hat{\mathrm{x}}^{(t)}$ and only certain sub-matrices of $\mathrm{P}^{(t)}$ are needed to compute $\mathrm{tr}(\mathrm{DP}^{(t)}\mathrm{D}^T)$. For instance, in the TM model we have

$$\mathrm{tr}(\mathrm{DP}^{(t)}\mathrm{D}^T) = \sum_k \mathrm{d}_k^T \mathrm{P}^{(t)} \mathrm{d}_k = \sum_{\{u,v\} \in \mathcal{E}} (\mathrm{P}_{u,u}^{(t)} + \mathrm{P}_{v,v}^{(t)} - 2\mathrm{P}_{u,v}^{(t)}),$$

[10]Here, to simplify analysis, we add an additional regularization term $\|\mathrm{x}\|^2$ with relative weight $\epsilon > 0$, which can be made arbitrarily small. This insures that $p(\mathrm{x})$ is non-singular, with invertible information matrix $-\theta_1(\mathrm{D}^T\mathrm{D} + \epsilon \mathrm{I}_n)$.

which only requires computation of variances and edge-wise covariances. Similarly, because each measurement only depends on a few components of x, the matrix C is sparse and it is tractable to compute $\mathrm{C}\hat{\mathrm{x}}^{(t)}$ and $\mathrm{tr}(\mathrm{C}\mathrm{P}^{(t)}\mathrm{C}^T)$. To solve the M-step, we note that

$$\begin{aligned} E\{\phi_1\} &= \mathrm{tr}((\mathrm{D}^T\mathrm{D} + \epsilon I_n)\mathrm{cov}(x)) \\ &= \mathrm{tr}((\mathrm{D}^T\mathrm{D} + \epsilon I_n)(-\theta_1(\mathrm{D}^T\mathrm{D} + \epsilon I_n))^{-1}) \\ &= -n\theta_1^{-1} \end{aligned}$$

where n is the dimension of x. By similar analysis, $E\{\phi_2\} = -m\theta_2^{-1}$ where m is the number of measurements. Thus, the solution for the M-step is

$$\theta_1^{(t)} = -\frac{n}{\eta_1^{(t)}} \text{ and } \theta_2^{(t)} = -\frac{m}{\eta_2^{(t)}}, \tag{29}$$

which, together with (28), specifies the EM algorithm. The EM algorithm requires computation of conditional variances and edge-wise covariances at each iteration. Hence, simple estimation methods that only compute the means $\hat{\mathrm{x}}$ are inadequate for parameter estimation. RCM also computes approximate variances and edge-wise covariances and is therefore well-suited for implementing an *approximate* EM procedure for models where direct methods are intractable. This approach can be used to obtain parameter estimates in the applications considered in Section V.

REFERENCES

[1] H.-O. Georgii, *Gibbs measures and phase transitions*. Berlin: de Gruyter, 1988.
[2] B. Simon, *The Statistical Mechanics of Lattice Gases*. Princeton Univ. Press, 1993.
[3] B. Frey, Ed., *Graphical models for machine learning and digital communication*. MIT Press, 1998.
[4] J. Woods, "Markov image modeling," *IEEE Trans. Automat. Contr.*, vol. 23, no. 5, pp. 846–850, Oct. 1978.
[5] J. Besag, "On the statistical analysis of dirty pictures," *J. Royal Stat. Soc.*, vol. 48, no. 3, pp. 259–302, 1986.
[6] P. Pérez, A. Chardin, and J.-M. Laferté, "Noniterative manipulation of discrete energy-based models for image analysis," *Pattern Recognition*, vol. 33, pp. 573–586, 2000.
[7] J.-M. Laferté, P. Pérez, and F. Heitz, "Discrete Markov image modeling and inference on the quadtree," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 390–404, Mar. 2000.
[8] E. Monfrini, J. Lecomte, F. Desbouvries, and W. Pieczynski, "Image and signal restoration using pairwise Markov trees," in *IEEE Workshop on Signal Processing*, Oct. 2003, pp. 174–177.
[9] W. Pieczynski, "Hidden evidential Markov trees and image segmentation," in *Inter. Conf. Image Processing*, 1999, pp. 338–342.
[10] S. Chevalier, E. Geoffrois, F. Prêteux, and M. Lemaître, "A generic 2d approach to handwriting recognition," in *Proc. 8th Inter. Conf. Document Analysis and Recognition*, Sept. 2005, pp. 489–493.
[11] P. Fieguth, W. Karl, A. Willsky, and C. Wunsch, "Multiresolution optimal interpolation & statistical analysis of topex/poseidon satellite altimetry," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, no. 2, pp. 280–292, Mar. 1995.
[12] M. Daniel and A. Willsky, "A multiresolution methodology for signal-level fusion and data assimilation with applications in remote sensing," *Proc. IEEE*, vol. 85, pp. 164–183, July 1997.
[13] X. Descombes, M. Sigelle, and F. Préteux, "Estimating GMRF parameters in a nonstationary framework: application to remote sensing," *IEEE Trans. Image Processing*, vol. 8, no. 4, pp. 490–503, Apr. 1999.
[14] A. Willsky, "Multiresolution Markov models for signal and image processing," *Proc. IEEE*, vol. 90, no. 8, pp. 1396–1458, Aug. 2002.
[15] S. Thrun, D. Koller, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *Int. J. Robotics Research*, vol. 23, no. 8, pp. 693–716, Aug. 2004.
[16] S. Lauritzen, *Graphical Models*. Oxford Univ. Press, 1996.
[17] R. Cowell, A. Dawid, S. Lauritzen, and D. Spiegelhalter, *Probabilistic Networks and Expert Systems*. Springer, 1999.
[18] G. Grimmett, "A theorem about random fields," *Bull. London Math. Soc.*, vol. 5, pp. 81–84, 1973.
[19] L. Saul and M. Jordan, "Exploiting tractable substructures in intractable networks," in *Adv. Neural Inf. Proc. Systems*, 1995.
[20] J. Yedidia, W. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *Proc. Conf. Uncertainty in AI*, pp. 239–269, 2003.
[21] H. Kappen and W. Wiegerinck, "A novel iteration scheme for the cluster variation method," in *Adv. Neural Inf. Proc. Systems*, 2002.
[22] M. Wainwright, T. Jaakkola, and A. Willsky, "Tree-based reparameterization framework for analysis of sum-product and related algorithms," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1120–1130, May 2003.
[23] D. Taylor, "Parallel estimation of one and two dimensional systems," Ph.D. dissertation, MIT, Feb. 1992.
[24] M. Luettgen, W. Karl, A. Willsky, and R. Tenney, "Multiscale representations of Markov random fields," *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3377–3395, 1993.
[25] U. Kjærulff, "Reduction of computational complexity in Bayesian networks through removal of weak dependences," in *Proc. Conf. Uncertainty in AI*, vol. 10, 1994, pp. 374–384.
[26] X. Boyen and D. Koller, "Tractable inference for complex stoch. proc." in *Adv. Neural Inf. Proc. Systems*, vol. 14, 1998, pp. 33–42.
[27] T. Heskes and O. Zoeter, "Expectation propagation for approximate inference in dynamic Bayesian networks," in *Proc. Conf. Uncertainty in AI*, 2002.

[28] T. Speed and H. Kiiveri, "Gaussian Markov distributions over finite graphs," *Annals Stat.*, vol. 14, no. 1, pp. 138–150, Mar. 1986.
[29] D. Karger and N. Srebro, "Learning Markov networks: maximum bounded treewidth graphs," in *Proc. 12th ACM-SIAM Symp. on Discrete Algorihtms*, Jan. 2001.
[30] J. Pearl, *Probabilistic inference in intelligent systems*. Morgan Kaufmann, 1988.
[31] Y. Weiss and W. Freeman, "Correctness of belief propagation in Gaussian graphical models," *Neural Comp.*, vol. 13, pp. 2173–2200, 2001.
[32] D. Malioutov, J. Johnson, and A. Willsky, "Walk-sums and belief propagation in Gaussian graphical models," *J. Machine Learning Research*, vol. 7, pp. 2031–2064, Oct. 2006.
[33] W. Irving and A. Willsky, "Multiscale stochastic realization using canonical correlations," *IEEE Trans. Automat. Contr.*, Sept. 2001.
[34] A. Frakt and A. Willsky, "A scale-recursive method for constructing multiscale stochastic models," *Multidim. Sig. Proc.*, vol. 12, pp. 109–142, 2001.
[35] S. Amari, "Information geometry on hierarchy of probability distributions," *IEEE Trans. Inform. Theory*, vol. 47, no. 5, pp. 1701–1711, July 2001.
[36] F. Bach and M. Jordan, "Thin junction trees," in *Adv. Neural Inf. Proc. Systems*, 2001.
[37] B. Gidas, "A renormalization group approach to image processing problems," *IEEE Trans. Image Processing*, vol. 11, no. 2, pp. 164–180, Feb. 1989.
[38] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley, 1991.
[39] O. Barndorff-Nielsen, *Information and Exponential Families*. John Wiley, 1978.
[40] M. Wainwright and M. Jordan, "Graphical models, exponential families, and variational inference," Dept. of Stat., UC Berkeley, Tech. Rep. 649, 2003.
[41] E. Jaynes, "Information theory and statistical mechanics," *Phys. Rev.*, 1957.
[42] A. Dempster, "Covariance selection," *Biometrics*, vol. 28, no. 1, pp. 157–175, Mar. 1972.
[43] A. Frakt, H. Lev-Ari, and A. Willsky, "A generalized Levinson algorithm for covariance extension with application to multiscale autoregressive modeling," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, Feb. 2003.
[44] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Image Processing*, vol. 4, no. 7, pp. 932–946, July 1995.
[45] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete date via the EM algorithm," *J. Royal Stat. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.