

Message Quantization in Belief Propagation: Structural Results in the Low-Rate Regime

O. Patrick Kreidl and Alan S. Willsky, *Fellow, IEEE*

Abstract—Motivated by distributed inference applications in unreliable communication networks, we adapt the popular (sum-product) belief propagation (BP) algorithm under the constraint of discrete-valued messages. We show that, in contrast to conventional BP, the optimal message-generation rules are node-dependent and iteration-dependent, each rule making explicit use of local memory from *all* past iterations. These results expose both the intractability of optimal design and an inherent structure that can be exploited for tractable approximate design. We propose one such approximation and demonstrate its efficacy on canonical examples. We also discuss extensions to communication networks with lossy links (e.g., erasures) or topologies that differ from the graph underlying the probabilistic model.

I. INTRODUCTION

A. Motivation

Inference problems, typically posed as the computation of summarizing statistics (e.g., marginals, modes) given a multivariate probability distribution, arise in a variety of scientific fields and engineering applications. Probabilistic graphical models provide a scalable framework for developing efficient inference methods, such as message-passing algorithms (e.g., belief propagation) that exploit the conditional independencies among subsets of random variables as encoded by the given graph [1]. Assuming a network of distributed sensors, application of the graphical model formalism may at first seem trivial, as there already exists a natural graph defined by the sensor nodes and the inter-sensor communications. However, modern networks can involve resource constraints beyond those satisfied by existing message-passing algorithms e.g., a fixed small number of iterations, low-rate or unreliable links, a topology that differs from the probabilistic graph. Such issues have already inspired inquiries into the robustness of existing message-passing algorithms to unmodeled resource constraints [2]–[7], demonstrating limits to their reliability and motivating alternative distributed solutions that degrade gracefully even as network constraints become severe [8]–[10].

This paper focuses on a rich class of *network-constrained* inference problems, namely those for which the unconstrained counterparts are popularly addressed via the (sum-product) belief propagation (BP) algorithm. The distinguishing assumption in our formulation is the non-ideal commu-

nication model, rendering conventional BP infeasible and fundamentally altering the character of satisfactory solutions. For example, in the special case where the probability graph and the network topology are identical, it is well known that BP requires communication overhead of *at least* two real-valued messages per edge. In contrast, our class of problems mandates having to compress, or quantize, these messages such that communication overhead is *at most* a fixed number of discrete-valued messages (e.g., two “bits” per edge). Moreover, assuming only a fixed number of iterations, the canonical inference challenge of finding efficient yet convergent message-passing approximations for “loopy” graphical models is met trivially by constraint. The key algorithmic challenges rather arise in the need to redesign the message-passing rules subject to the communication constraints, taking into account the goals of processing (e.g., decisions to be made by some or all sensor nodes) in order to make best use of these limited bits. The necessary departure from BP only becomes more pronounced when the network topology may differ from the probability graph.

B. Related Work

The explicit consideration of binding communication constraints enters our work into the realm of approximate inference, where existing graph-based methods primarily address the case of limited computation resources. Variational methods for approximate inference start by expressing the intractable solution as the minimizing (or maximizing) argument of a mathematical optimization problem [11], [12]. One can often recover existing algorithms from different specializations of such an optimization problem. More importantly, by relaxing or otherwise modifying this optimization problem to render it amenable to mathematical programming techniques, one can obtain tractable yet effective approximations to the original inference problem and, ideally, an analysis of error bounds or other fundamental limits associated with alternative approximations. Variational methods have recently been the vehicle towards an improved understanding of the so-called *loopy* belief propagation (BP) algorithm [1], uncovering its links to LDPC codes in information theory [13], [14] as well as to entropy-based Bethe free energy approximations in statistical physics [15]–[18].

The variational methods in this paper sharply depart from these contemporary perspectives on belief propagation (BP). Firstly, originally motivated by sensor network applications, we return to BP’s traditional message-passing view, assuming that the nodes in the graph physically correspond to spa-

This work was supported in part by ARO MURI W911NF-06-1-0076 and in part by AFOSR MURI FA9550-06-1-0324.

The authors are with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, USA. {opk,willsky}@mit.edu

tially distributed sensors/processors. Secondly, our need for approximation is dominated by constraints on the available *communication* resources—efficient computation remains a concern as well: in particular, we essentially bypass the key technical issue of convergence by allowing from the start only a fixed small number of message-passing iterations. Also in contrast to other variational methods, our approximation is driven by decision-based objectives (as opposed to entropy-based objectives) that can also capture costs associated to communication decisions.

Other recent works in approximate inference similarly look towards distributed sensing applications. An experimental implementation of BP within an actual sensor network concludes that reliable communications are indeed the dominant drain on battery power, with overhead varying substantially over different message schedules and network topologies [2]. A modification of the exact junction-tree algorithm introduces redundant representations to compensate for anticipated packet losses and node dropouts inherent to wireless sensor networks [4]. Some theoretical impacts of finite-rate links in loopy BP have also been addressed [5], essentially proving that “small-enough” quantization errors do not alter the behavior of BP. A similar robustness property is observed empirically in a distributed object tracking application, where “occasionally” suppressing the transmission of a message is shown to have negligible impact on performance and, in some cases, can even speed up convergence [6]. These views on communication issues relate closely to the general problem of BP message approximation [3], [5], [19], [20], which generically arises due to the infinite-dimensional messages implied by BP in the case of (non-Gaussian) continuous-variable graphical models.

This paper considers more severe network constraints than those discussed above. As such, in contrast to proposing modifications directly to the BP algorithms, Section II explicitly models the network constraints inside an otherwise unconstrained formulation by which BP can be derived. In Section III, via analysis of the resulting constrained optimization problem, we examine the extent to which alternative message-passing rules mitigate the *loss* from optimal performance subject to the network constraints. Experiments on canonical examples (Section IV) suggest that our network-constrained solutions perform competitively with (unconstrained) BP if it converges, and can even outperform BP when its convergence is in question (i.e., in so-called “frustrated” models). While conceptually related to the problem of BP message approximation, especially when the network topology and the probability graph coincide, our applicability to the “low-rate” quantization regime is unique.

II. PROBLEM DEFINITION

Our problem definition has two main parts: the probabilistic model and the communication model. The former (Subsection II-A) belongs in the class of graphical models for which belief propagation (BP) algorithms are applicable, while the latter (Subsection II-B) specifies the explicit network constraints that render conventional BP infeasible

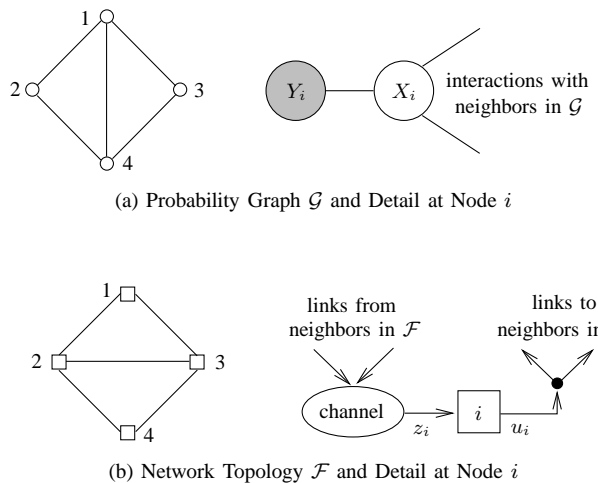


Fig. 1. Two different graph-based models in our problem definition and the meaning of their respective edge sets.

or unreliable. These two graph-based models, taken alongside a decision-theoretic penalty function (Subsection II-C), comprise the constrained minimization problem we analyze in the next section. It is worth noting here that each model involves an n -node graph, but we impose no restrictions on how the respective edge sets are related.

A. Probabilistic Graphical Model

Let random vectors X and Y denote a hidden state process and noisy observation process, respectively, taking values x in a discrete product space $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_n$ and y in a Euclidean product space $\mathcal{Y} = \mathcal{Y}_1 \times \dots \times \mathcal{Y}_n$. We focus on joint distributions $p(x, y)$ represented compactly by a (pairwise) Markov random field, which is a typical assumption in BP. Specifically, given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V} = \{1, \dots, n\}$ and (undirected) edge set $\mathcal{E} \subset \{(i, j) \in \mathcal{V} \times \mathcal{V} | i \neq j\}$,

$$p(x, y) \propto \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(x_i, x_j) \prod_{i \in \mathcal{V}} p(y_i | x_i), \quad (1)$$

where the so-called compatibility functions $\psi_{i,j} : \mathcal{X}_i \times \mathcal{X}_j \rightarrow (0, \infty)$ collectively specify (up to normalization) prior probabilities $p(x)$, while each conditional distribution $p(y_i | x_i)$ specifies the (perhaps) noisy observation process local to node i . Figure 1(a) illustrates a four-node probability graph \mathcal{G} having the structure expressed in (1).

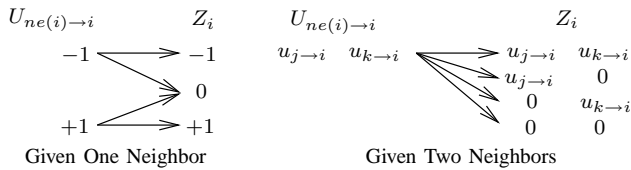
B. Communication Network Model

Our non-ideal communication model is inspired by the iterative (or parallel) BP algorithms that exist for probabilistic graphical models of the form in (1), while managing the twists that (i) the network topology may have different links than those implied by edge set \mathcal{E} and (ii) every such link is low-rate and perhaps also unreliable. More formally, let graph $\mathcal{F} = (\mathcal{V}, \mathcal{D})$ define the network topology, each (undirected) edge (i, j) in \mathcal{D} indicating a bidirectional communication link between nodes i and j . Assign to each

such link the integers $d_{i \rightarrow j}$ and $d_{j \rightarrow i}$, each greater than unity, denoting the size of the (direction-dependent) symbol set supported by that link within each iteration, or *stage*, of communication. (i.e., the link rate in the direction from node i to node j is $\log_2 d_{i \rightarrow j}$ bits per stage). Thus, in every communication stage, the *symbol(s)* transmitted by node i can take at most $\prod_{j \in ne(i)} d_{i \rightarrow j}$ distinct values, where $ne(i) = \{j \in \mathcal{V} \mid \text{edge}(i, j) \text{ in } \mathcal{D}\}$ denotes the neighbors of node i (in \mathcal{F}). For example, if node i may transmit a different symbol to each neighbor, it selects one of $\prod_{j \in ne(i)} d_{i \rightarrow j}$ distinct alternatives; if node i transmits the same symbol to every neighbor, it selects one of only $\min_{j \in ne(i)} d_{i \rightarrow j}$ distinct alternatives.

No matter the transmission scheme, let \mathcal{U}_i denote the finite set from which each node i selects the symbol(s) to send to its neighbors in each communication stage. We similarly assume the symbol(s) received by node i in the subsequent stage take their values in a given finite set \mathcal{Z}_i . The cardinality of \mathcal{Z}_i will certainly reflect the joint cardinality $|\mathcal{U}_{ne(i)}| = \prod_{j \in ne(i)} |\mathcal{U}_j|$ of its neighbors' transmissions, but the exact relation is determined by a given multipoint-to-point noisy *channel* into each node i i.e., a conditional distribution $p(z_i | u_{ne(i)})$ that characterizes the information Z_i received by node i based on the collective symbols $u_{ne(i)} = \{u_j \in \mathcal{U}_j \mid j \in ne(i)\}$ transmitted by its neighbors. Figure 1(b) illustrates a network topology \mathcal{F} and its implications on each stage of nearest-neighbor communications; the following example describes an illustrative special case.

Example 1: (Peer-to-Peer Binary Comms with Erasures). Consider a network of bidirectional unit-rate links, meaning $d_{i \rightarrow j} = d_{j \rightarrow i} = 2$ for every edge (i, j) in \mathcal{F} . Let $u_{i \rightarrow j} \in \{-1, +1\}$ denote the actual symbol transmitted by node i to its neighbor $j \in ne(i)$. It follows that the collective symbol $u_i = \{u_{i \rightarrow j} \mid j \in ne(i)\}$ transmitted by node i takes its values in the set $\mathcal{U}_i = \{-1, +1\}^{|ne(i)|}$. On the receiving end, let $z_{j \rightarrow i} \in \{-1, 0, +1\}$ denote the actual symbol received by node i from its neighbor $j \in ne(i)$, where the value “0” indicates an erasure and otherwise $z_{j \rightarrow i} = u_{j \rightarrow i}$. It follows that the collective symbol $z_i = \{z_{j \rightarrow i} \mid j \in ne(i)\}$ received by node i takes values in $\mathcal{Z}_i = \{-1, 0, +1\}^{|ne(i)|}$. Note that the received information Z_i depends statistically only on those symbols transmitted to node i by its neighbors, which we denote by $u_{ne(i) \rightarrow i} = \{u_{j \rightarrow i} \mid j \in ne(i)\}$.



C. Decision-Theoretic Variational Formulation

We first describe the formulation in the absence of communication network constraints, seeking a function $\gamma : \mathcal{Y} \rightarrow \mathcal{X}$ by which to decide upon the value of the hidden random vector X based on the observable random vector Y . Letting $\hat{X} = \gamma(Y)$ denote the induced decision process and

associating a numeric “cost” $c(\hat{x}, x)$ to every possible joint realization of (\hat{X}, X) , the expected cost

$$J(\gamma) = E \left[c(\hat{X}, X) \right] = E \left[E \left[c(\gamma(Y), X) \mid Y \right] \right] \quad (2)$$

is a well-defined measure of performance. Expanding the inner expectation and recognizing $p(x|y)$ to be proportional to $p(x)p(y|x)$ for every y such that $p(y) > 0$, the minimizer γ^* of (2) satisfies

$$\bar{\gamma}(Y) = \arg \min_{\hat{x} \in \mathcal{X}} \sum_{x \in \mathcal{X}} c(\hat{x}, x) p(x|Y) \quad \text{with probability one.} \quad (3)$$

The (sum-product) BP algorithm can be motivated by a special case of the cost function $c(\hat{x}, x)$.

Example 2: (Maximum-Posterior-Marginal Estimation). Assume the cost function satisfies $c(\hat{x}, x) = \sum_{i=1}^n c(\hat{x}_i, x_i)$ with

$$c(\hat{x}_i, x_i) = \begin{cases} 1 & , \hat{x}_i \neq x_i \\ 0 & , \text{otherwise} \end{cases} \quad \text{for } i = 1, \dots, n.$$

Then (2) specializes to the *sum-error-rate*, or the expected number of component errors between vectors \hat{X} and X , and (3) specializes to the the *Maximum-Posterior-Marginal* (MPM) estimator, or $\bar{\gamma}(Y) = (\bar{\gamma}_1(Y), \dots, \bar{\gamma}_n(Y))$ with

$$\bar{\gamma}_i(Y) = \arg \max_{x_i \in \mathcal{X}_i} p(x_i|Y) \quad \text{for } i = 1, \dots, n.$$

This estimator is easy to implement given the posterior marginal $p(x_i|y)$ local to every node i , which BP strives to compute efficiently (per observation $Y = y$) for probabilistic models that satisfy (1).

We now introduce into the formulation the communication network model described in Subsection II-B, manifesting itself as explicit constraints on the function space Γ over which the expected cost $J(\gamma)$ in (2) is minimized. Let us allow $t \geq 1$ stages of communication, each taken to be a parallel symbol exchange between every node i and its neighbors $ne(i)$ in the network topology \mathcal{F} . In the initial stage, node i generates its communication decision u_i^1 as a function of only the local observation y_i . In each subsequent communication stage $k = 2, 3, \dots, t$, let z_i^k denote the symbol(s) received by node i before it then makes its next communication decision u_i^k . After the t th communication stage, upon receiving z_i^{t+1} , each node i makes its final state-related decision \hat{x}_i .

A key opportunity associated with this communication scheme is the use of *memory*, which local to each node can include the symbols both transmitted and received in all preceding stages. We denote by \mathcal{M}_i^k the set of all stage- k communication rules local to node i , each of the form

$$\mu_i^k : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \times \mathcal{U}_i^2 \times \mathcal{Z}_i^3 \times \dots \times \mathcal{U}_i^{k-1} \times \mathcal{Z}_i^k \rightarrow \mathcal{U}_i^k$$

for $k = 1, 2, \dots, t$. Similarly, we denote by Δ_i the set of all final-stage decision rules local to node i , each of the form

$$\delta_i : \mathcal{Y}_i \times \mathcal{U}_i^1 \times \mathcal{Z}_i^2 \times \mathcal{U}_i^2 \times \mathcal{Z}_i^3 \times \dots \times \mathcal{U}_i^t \times \mathcal{Z}_i^{t+1} \rightarrow \mathcal{X}_i.$$

It follows that the set of all *multi-stage rules* local to node i , each a particular sequence of single-stage rules $\gamma_i =$

$(\mu_i^1, \dots, \mu_i^t, \delta_i)$, is defined by $\Gamma_i = \mathcal{M}_i^1 \times \dots \times \mathcal{M}_i^t \times \Delta_i$. In turn, the set of all *network-constrained strategies*, each a particular collection of multi-stage rules $\gamma = (\gamma_1, \dots, \gamma_n)$, is defined by $\Gamma = \Gamma_1 \times \dots \times \Gamma_n$. It is clear that these functional constraints, prohibiting all n components of \hat{X} from being decided jointly, render the most general unconstrained minimizer in (3) infeasible; the MPM estimator in Example 2 is rendered infeasible by the finite-rate links, particularly evident in the “low-rate” regime where $\prod_{k=1}^t |\mathcal{U}_i^k \times \mathcal{Z}_i^{k+1}|$ is much smaller than $\prod_{j \neq i} |\mathcal{Y}_j|$.

As in the unconstrained formulation, any function $\gamma \in \Gamma$ induces a state-related decision process \hat{X} based on the observable random vector Y . Every network-constrained strategy will also induce two communication-related processes, namely all transmitted information $U = (U_1, \dots, U_n)$ and all received information $Z = (Z_1, \dots, Z_n)$, where the components local to node i are comprised of the sequences $U_i = (U_i^1, \dots, U_i^t)$ and $Z_i = (Z_i^2, \dots, Z_i^{t+1})$, respectively. It follows that the strategy-dependent distribution underlying the expectation in (2) generalizes to

$$p(\hat{x}, x; \gamma) = p(x) \int_{y \in \mathcal{Y}} p(y|x) p(\hat{x}|y; \gamma) dy \quad (4)$$

with

$$p(\hat{x}|y; \gamma) = \sum_{u \in \mathcal{U}} \sum_{z \in \mathcal{Z}} p(u, z, \hat{x}|y; \gamma),$$

which makes (2) impractical to evaluate (and thus to minimize) directly. The analysis in the next section reveals a special structure in (4), stemming from the factorization and causality implied by the respective graph-based models and exposing opportunities for effective approximations.

Example 3: (BP with Binary-Valued Messages). Consider the special case of our variational formulation where the cost function is as described in Example 2 and the communication model is as described in Example 1, in the latter also assuming that the network topology \mathcal{F} is identical to the probability graph \mathcal{G} in (1) and all erasure probabilities are zero i.e., at every node i , we have $\mathcal{Z}_i = \mathcal{U}_{ne(i) \rightarrow i}$ and the channel is simply

$$p(z_i^{k+1} | u_{ne(i) \rightarrow i}^k) = \begin{cases} 1 & , \quad z_i^{k+1} = u_{ne(i) \rightarrow i}^k \\ 0 & , \quad \text{otherwise} \end{cases}$$

for every stage $k = 1, 2, \dots, t$. Here, the channel serves only to capture the convention in BP that every node in each stage may transmit different symbols to different neighbors, so in the subsequent stage it receives only its unique subset of its neighbors’ transmitted symbols. However, BP assumes that nodes transmit (non-negative) real-valued vectors in each stage, which in our model corresponds to the (infeasible) case of an infinite-rate network i.e., $u_{i \rightarrow j}^s \in [0, \infty)^{|\mathcal{X}_j|}$ and $u_{j \rightarrow i}^s \in [0, \infty)^{|\mathcal{X}_i|}$ per edge (i, j) in \mathcal{G} . It is also worth noting that the rules implied by conventional BP do *not* use memory, since so doing is known to be superfluous if \mathcal{G} is a tree.

III. SUMMARY OF STRUCTURAL RESULTS

The variational problem defined in the preceding section is, at the highest level, to minimize the penalty function $J(\gamma)$

in (2) subject to functional constraints $\gamma \in \Gamma_1 \times \dots \times \Gamma_n$. Details of the probabilistic graphical model are captured primarily within the penalty function, whereas details of the communication network model are captured primarily within the functional constraints. Our analysis follows essentially the same steps taken for the single-stage instance formulated in [9], proceeding from a simple premise: if strategy $\gamma^* = (\gamma_1^*, \dots, \gamma_n^*)$ is optimal over Γ , then for each i and assuming all other component functions are fixed at $\gamma_{\setminus i}^* = \{\gamma_j^* | j \neq i\}$, the function γ_i^* must be optimal over Γ_i . In the multi-stage problem formulated here, each component optimization over Γ_i is similarly decomposed over its single-stage rules. Note that this relaxation neglects the opportunity for optimizing over multiple component functions simultaneously, and thus the associated analysis leads to necessary (but *not* sufficient) optimality conditions for the original problem. Nonetheless, our results expose a number of structural properties that the optimal network-constrained strategy should satisfy, including how each node should use its local memory (i.e., *all* previously communicated symbols to or from that node) in a most informative way. For brevity, all formal proofs and certain lower-level details must be omitted here—we refer the interested reader to [21].

Let us first introduce notation to allow a more concise representation for the multi-stage rules defined in Subsection II-B. Firstly, for each node i , define \mathcal{Z}_i^1 as the empty set (i.e., no received symbols exist before the first communication stage); then, view the expanding memory at each node i as the sequential realization of a (local) *information vector*,

$$I_i^k = \begin{cases} \emptyset & , \quad k = 1 \\ (I_i^{k-1}, z_i^{k-1}, u_i^{k-1}) & , \quad k = 2, 3, \dots, t+1 \end{cases} .$$

We may then denote each k th communication rule by $U_i^k = \mu_i^k(Y_i, I_i^k, Z_i^k)$ and each final-stage decision rule by $\hat{X}_i = \delta_i(Y_i, I_i^{t+1}, Z_i^{t+1})$. Note that, by construction, fixing a stage- k rule $\mu_i^k \in \mathcal{M}_i^k$ is equivalent to specifying a distribution $p(u_i^k | y_i, I_i^k, z_i^k; \mu_i^k) = 1$ if $u_i^k = \mu_i^k(y_i, I_i^k, z_i^k)$ and zero otherwise; similarly, fixing a final-stage rule specifies the analogous distribution $p(\hat{x}_i | y_i, I_i^{t+1}, z_i^{t+1}; \delta_i)$. In turn, fixing a multi-stage rule $\gamma_i \in \Gamma_i$ is equivalent to specifying

$$p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) = p(\hat{x}_i | y_i, I_i^{t+1}, z_i^{t+1}; \delta_i) \prod_{k=1}^t p(u_i^k | y_i, I_i^k, z_i^k; \mu_i^k). \quad (5)$$

The factorization in (5) is a direct consequence of the constraints that every node may communicate only with its nearest-neighbors $ne(i)$ in the network \mathcal{F} . The following lemma states when this factorization leads to special structure in the global distribution $p(\hat{x}|x; \gamma)$ in (4), which requires an assumption already satisfied by the probabilistic and communication models defined in Section II.

Assumption 1: (Spatially-Independent Noise and Memoryless Channels). Conditioned on the hidden process X_i , the observation Y_i and received symbols Z_i local to node i are mutually independent as well as independent of all non-local processes in the network, namely the hidden processes $X_{\setminus i}$,

$$p(u_i, \hat{x}_i | x_i, u_{ne(i)}; \gamma_i) = \sum_{z_i \in \mathcal{Z}_i} p(z_i | x_i, u_{ne(i)}) \int_{y_i \in \mathcal{Y}_i} p(y_i | x_i) p(u_i, \hat{x}_i | y_i, z_i; \gamma_i) dy_i. \quad (7)$$

$$p(y_i | x_i, I_i^k; \gamma_i) \propto \begin{cases} p(y_i | x_i) & , \text{ if } y_i \text{ such that } u_i^m = \mu_i^m(y_i, I_i^m, z_i^m) \text{ for } m = 1, 2, \dots, k-1 \\ 0 & , \text{ otherwise} \end{cases} \quad (8)$$

observations $Y_{\setminus i}$ and the channels of all other nodes i.e., we have, for every node i ,

$$\begin{aligned} p(y_i, z_i | x, y_{\setminus i}, z_{\setminus i}, u_{\setminus i}) &= p(y_i | x_i) p(z_i | x_i, u_{ne(i)}) \\ &= p(y_i | x_i) \prod_{k=1}^t p(z_i^{k+1} | x_i, u_{ne(i)}^k). \end{aligned}$$

Lemma 1: (Network-Constrained Global Factorization).

Let Assumption 1 hold. For every network-constrained strategy $\gamma \in \Gamma$, the distribution in (4) specializes to

$$p(\hat{x}, x; \gamma) = p(x) \sum_{u \in \mathcal{U}} p(u, \hat{x} | x; \gamma) \quad (6)$$

with

$$p(u, \hat{x} | x; \gamma) = \prod_{i=1}^n p(u_i, \hat{x}_i | x_i, u_{ne(i)}; \gamma_i)$$

and each i th factor is given by (7).

It may seem counter-intuitive, in light of the sequential communication model, that (6) does not also exhibit a factorization with respect to stages $k = 1, \dots, t$. The caveat is that these successive stages collectively operate on the same observation vector $Y = y$. It is rather the side information local to each node i that grows over successive stages, providing an increasingly global context in which to *reprocess* the local observation $Y_i = y_i$. However, the sequential communication model can be exploited to simplify the local marginalizations in (7). In particular, each node i may firstly decompose the integral over \mathcal{Y}_i into a finite collection of integrals over memory-dependent subregions of \mathcal{Y}_i and secondly evaluate the sum over \mathcal{Z}_i in a recursive fashion. These simplifications are developed formally in [21], but the upshot is a precise characterization of how each node interprets its communicated information to successively pare down its local likelihood (see Figure 2): that is, with every stage k , each node i hones in on a smaller support set for its local likelihood function $p(Y_i | x_i)$ as a function of its information vector I_i^k and its preceding communication rules $\mu_i^{1:k-1}$, expressed mathematically in (8).

Assumption 2: (Separable Costs). The global cost function is additive across nodes, each i th term independent of all non-local decision and hidden variables i.e., we have $c(\hat{x}, x) = \sum_{i=1}^n c(\hat{x}_i, x_i)$.

Proposition 1: (Optimal Parameterization of Final Decision Stage). Let Assumptions 1 & 2 both hold. Assume that all nodes' multi-stage communication rules are fixed at their optimal values, denoted by $\mu_j^* \in \mathcal{M}_j^1 \times \dots \times \mathcal{M}_j^t$ for every node j . Then, there exists a likelihood statistic $P_i^*(u_i, z_i | x_i)$ for each node i such that its optimal final-stage rule over all

Δ_i reduces to

$$\delta_i^*(Y_i, U_i, Z_i) = \arg \min_{\hat{x}_i \in \mathcal{X}_i} \sum_{x_i \in \mathcal{X}_i} b_i(\hat{x}_i, x_i; U_i, Z_i) p(Y_i | x_i, I_i^{t+1}; \mu_i^*)$$

with probability one, where real-valued parameters b_i satisfy

$$b_i(\hat{x}_i, x_i; u_i, z_i) \propto p(x_i) P_i^*(u_i, z_i | x_i) c(\hat{x}_i, x_i).$$

Proposition 1 carries a number of important implications. Foremost, it clarifies the conditions under which the optimal final-stage strategy δ^* lies in a finitely parameterized subset of $\Delta_1 \times \dots \times \Delta_n$, where the associated parameter vector $b = (b_1, \dots, b_n)$ scales linearly with the number of nodes n . Each component rule δ_i^* is also seen to make two different uses of its memory I_i^{t+1} . The first was highlighted in Figure 2, while the second is in interpreting the symbols z_i received over the preceding t stages of communication. We see that each likelihood statistic P_i^* , encompassing all that node i needs to know about the (fixed) communication rules of *all* other nodes in the network, depends *jointly* on the entire information vector $(u_i, z_i) = (I_i^{t+1}, z_i^{t+1})$. This joint dependence on information (U_i, Z_i) carries over to the local parameterization b_i and, in turn, implies that the parameter vector b scales exponentially with the number of stages t . We also see the appearance of the marginals $p(x_i)$ for every hidden variable X_i , which defines the extent to which our network-constrained solution restricts the probability graph

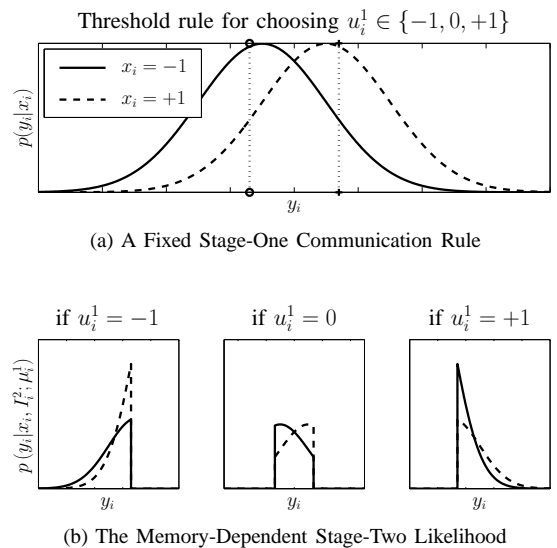


Fig. 2. Illustration of the first stage of the memory-dependent likelihood evolution in (8) for the case of a binary-valued hidden process X_i and a real-valued observation process Y_i corrupted by additive Gaussian noise. The trend to smaller support regions continues with each additional stage.

\mathcal{G} i.e., we assume the global prior $p(x)$ permits these local marginals to be computed or well-approximated “offline,” or before actual observations are processed.

Substituting the specific costs $c(\hat{x}_i, x_i)$ of Example 2 into Proposition 1 also reveals interesting ties to BP and MPM estimation. Each rule δ_i^* reduces to selecting the mode of the *network-constrained* posterior marginal $p(x_i|y_i, I_i^{t+1}, z_i^{t+1}; \mu^*)$. In other words, the role of the optimal communication strategy μ^* is to map the global observation vector y into the sequence of symbols (u, z) such that every node i may use the accessible portion of those symbols, namely (u_i, z_i) , alongside its local observation y_i to best approximate its (unconstrained) sufficient statistic $p(x_i|y)$. This interpretation (and identities within the proof to Proposition 1) yields the following network-constrained analog to the “belief update” equation for each stage k ,

$$M_i^k(x_i) \propto p(x_i)p(y_i|x_i, I_i^k; \mu_i^*) \times \sum_{\{(u_i^m, z_i^{m+1}) | m=k, k+1, \dots, t\}} P_i^*(u_i, z_i|x_i). \quad (9)$$

Observe that, before any online communication occurs, (9) specializes to $M_i^1(x_i) \propto p(x_i)p(y_i|x_i)$, or the MPM sufficient statistic at node i if the edge set of graph \mathcal{G} is empty (i.e. if random variables X_1, \dots, X_n are mutually independent).

The question of whether the optimal communication strategy μ^* similarly admits a finite parameterization is open. The special case of a single communication stage provably does [9]; the distinct complication with multiple communication stages is that each node’s earlier transmissions can impact information it will receive in later communication stages, affording an opportunity for every node to adapt to each new symbol of information, knowing every other node can do the same. Extrapolating from the single-stage solution and folding in the structure exposed by Proposition 1 motivates the following conjecture—its formal proof (or disproof) remains for future work. In any case, network-constrained strategies that are generated by a design algorithm developed on this conjecture appear to perform well empirically, as will be highlighted in the next section.

Conjecture 1: (Optimal Parameterization of Communication Stages). Let Assumptions 1 & 2 both hold. Assume all rules except for the stage- k communication rule local to node i are fixed at their optimal values. Then, there exist both a likelihood statistic $P_i^k(I_i^k, z_i^k|x_i)$ and a cost-to-go statistic $C_i^k(I_i^{k+1}, x_i)$ such that the optimal communication rule over all \mathcal{M}_i^k reduces to

$$\mu_i^k(Y_i, I_i^k, Z_i^k) = \arg \min_{u_i^k \in \mathcal{U}_i^k} \sum_{x_i \in \mathcal{X}_i} a_i^k(u_i^k, x_i; I_i^k, Z_i^k) p(Y_i|x_i, I_i^k, \mu_i^{1:k-1})$$

with probability one, where parameters a_i^k satisfy¹

$$a_i^k(u_i^k, x_i; I_i^k, z_i^k) \propto p(x_i) P_i^k(I_i^k, z_i^k|x_i) C_i^k(I_i^{k+1}, x_i).$$

¹Arguably the most optimistic part of Conjecture 1 is the lack of explicit dependence on Y_i in the cost function C_i^k . With such dependence, the rule μ_i^k does not necessarily lie in a finitely-parameterized subset of \mathcal{M}_i^k .

IV. AN APPROXIMATE OFFLINE ALGORITHM

The analysis of the preceding section reveals a number of barriers to tractably optimizing multi-stage network-constrained decision strategies that do not arise in the single-stage counterpart [9]. On the positive side, Proposition 1 establishes the minimal assumptions under which online computation (in the final-stage strategy δ^*) scales linearly with the number of nodes n . These assumptions, namely Assumption 1 and sparsity of the network topology \mathcal{F} , are seen to coincide with those needed to guarantee online efficiency in the single-stage case. However, in contrast to the single-stage case, the addition of Assumption 2 is not enough to also guarantee that the associated *offline* computation scales linearly in n . Moreover, we were unable to derive analogous structural results for the multi-stage communication strategy μ^* , offering instead Conjecture 1 that only proposes that it enjoys the analogous online efficiency of its single-stage counterpart. Indeed, we expect the offline computation associated with μ^* to be no easier than that of the final-stage strategy δ^* , considering the latter need only account for the receivers’ perspectives of any multi-stage signaling incentives whereas the former should also account for the transmitters’ perspectives.

Supposing Assumptions 1 & 2 are in effect, Subsection IV-A describes an approximate offline algorithm for generating multi-stage network-constrained decision strategies. This approximation is suited only for a small number of online communication stages t , as it respects the parameterization suggested in Section III and, hence, assumes the exponential growth in t is not yet a barrier. In this light, the approximation is most useful for exploring what performance benefits are achievable when moving from a single-stage to two-stages of communication, from two-stages to three-stages of communication, and so on as long as t is small enough such that local memory requirements remain manageable.² Initial experiments of this nature are discussed in Subsection IV-B.

A. Overview and Intuition

We lack the space to describe the approximation in detail (see [21]), but at a highest level there are two main steps.

- 1) Find a particular t -stage communication strategy $\tilde{\mu} \in \mathcal{M} = \mathcal{M}_1 \times \dots \times \mathcal{M}_n$ by making repeated use of known single-stage solutions [9], [10].
- 2) Find a particular final-stage strategy $\tilde{\delta} \in \Delta = \Delta_1 \times \dots \times \Delta_n$ via Proposition 1, employing a Monte-Carlo method to obtain the statistics $P_i^{\tilde{\mu}}$ for each node i .

Recall that Proposition 1 characterizes the optimal final-stage strategy assuming the multi-stage communication strategy is fixed, so the dominant approximations are introduced within the algorithm by which we first generate $\tilde{\mu}$ based on Conjecture 1. The key to preserving performance despite these approximations is to uphold the memory-dependent paring down of all nodes’ local likelihoods expressed in (8).

²Of equal interest is the question of finding good limited-memory approximations for problems that merit large t , a pursuit for future work.

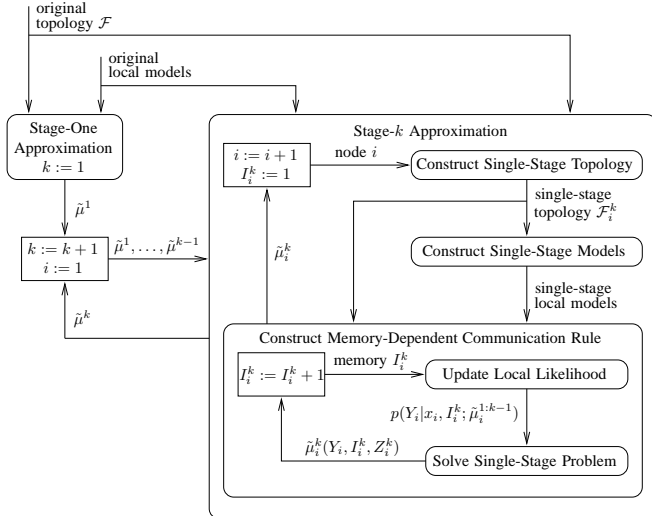


Fig. 3. A high-level flowchart of an algorithm for constructing an approximate multi-stage communication strategy $\tilde{\mu}$.

1) *Approximating the Communication Strategy*: Our algorithm for constructing an approximate multi-stage communication strategy $\tilde{\mu} \in \mathcal{M}$ combines the probabilistic structure exposed by Lemma 1, the finite parameterization proposed by Conjecture 1, and repeated application of the known single-stage solutions [9], [10]. A high-level flowchart of this algorithm is shown in Figure 3. In stage $k = 1$, every node's information vector is empty and the single-stage approximation operates directly on the given network topology \mathcal{F} , yielding all nodes' initial communication rules $\tilde{\mu}^1$. The outer loop of the algorithm then proceeds over increasing stages, each stage $k > 1$ involving an inner loop over all nodes and, for each node i , an inner-most loop over all possible values of local memory I_i^k , crafting a series of single-stage problems whose solutions collectively determine a particular local communication rule $\tilde{\mu}_i^k \in \mathcal{M}_i^k$.

For each node-stage pair (i, k) , the manner in which the series of single-stage problems is constructed, including how the local models are crafted from the original multi-stage models, involves a number of subtle yet significant approximations. These are described in detail in [21], but the main ideas are illustrated in Figure 4 by way of an example. Approximation of the stage-one communication rules $\tilde{\mu}^1 = (\tilde{\mu}_1^1, \dots, \tilde{\mu}_n^1)$ is straightforward, as no node has yet to account for local memory so the communication rule $\tilde{\mu}_i^1$ for every node i obtained from the single-stage solution is already a member of the stage-one function space \mathcal{M}_i^1 . Of course, this single-stage solution fails to capture any incentives for impacting the value of later-stage communications. Indeed, this side of the multi-stage signaling incentives is neglected throughout our approximation, as we repeatedly use the single-stage solutions without any look-ahead to future rounds of communication. Furthermore, in each subsequent stage $k > 1$, the rule $\tilde{\mu}_i^k$ for each node i is generated without consideration of the communication rules being generated in parallel at other nodes. Doing so

clearly neglects the fact that the true likelihood statistic $P_i^k(I_i^k, z_i^k | x_i)$ is a function of all nodes' communication rules $\tilde{\mu}^{1:k-1}$ from previous stages. More specifically, our construction of each single-stage problem makes no attempt to account for the exact statistical dependence between side information Z_i^k and local memory I_i^k . We do, however, properly account for the local memory I_i^k (and the local rules $\tilde{\mu}_i^{1:k-1}$ from preceding stages) inside of the measurement likelihood $p(y_i | x_i, I_i^k, \tilde{\mu}^{1:k-1})$ in accordance with (8).

2) *Approximating the Final-Stage Strategy*: Given Assumptions 1 & 2 hold and the multi-stage communication strategy is fixed to some member $\tilde{\mu}$ of the set \mathcal{M} , Proposition 1 implies that the search for a best final-stage strategy $\tilde{\delta}$ boils down to computing (offline) the likelihood statistics $P_i^{\tilde{\mu}}(u_i, z_i | x_i) \equiv p(u_i, z_i | x_i; \tilde{\mu})$ for every node i . However, as is evident in the proof of Proposition 1, exact computation of each such statistic at node i scales exponentially with the size of the node's t -step neighborhood (which is all n nodes once the number of stages exceeds the diameter of graph \mathcal{F}).

We employ a Monte-Carlo method (offline) to approximate the desired likelihood statistics $P_i^{\tilde{\mu}}$. Specifically, we draw independent samples from the joint distribution $p(x, y)$, and for each such sample apply both the multi-stage communication strategy $\tilde{\mu}$ and sample from the local channel models to yield a specific sequence of transmitted/received symbols (u_i, z_i) local to every node i . The statistic $P_i^{\tilde{\mu}}$ is then taken to be the empirical (conditional) distribution formed by all such samples of the triplet (X_i, U_i, Z_i) . A practical caveat

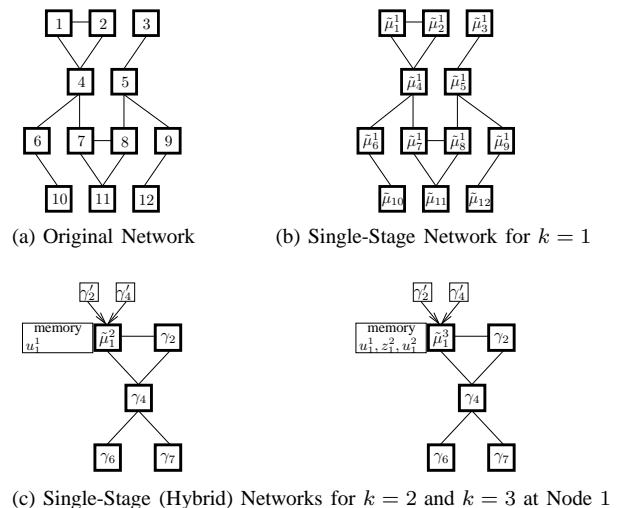
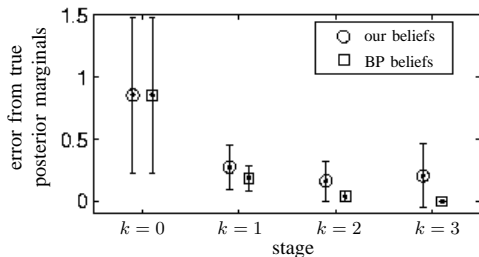
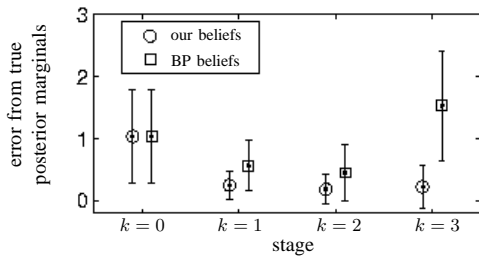


Fig. 4. (a) A specific undirected network topology \mathcal{F} in a multi-stage problem and (b) & (c) the sequence of single-stage hybrid topologies [10] used to approximate the multi-stage rule of node $i = 1$. All first-stage rules (including that of node i) can be approximated by just one single-stage solution, whereas the advent of memory in subsequent communication stages requires a single-stage solution per value of the local memory I_i^k . In (c), because we extract only the communication rule local to node i from each single-stage solution, we need not include the nodes that lie beyond its two-step neighborhood in \mathcal{F} [9]. In addition, rules γ_j for $j \neq i$ represent functions that are optimized within every single-stage solution for node i , but otherwise play no role in the approximation. Also note the phantom nodes (small boxes) serving as a placeholder for the stage- k side information Z_i^k local to node i , which in the stage- k rule results from the neighbors' decisions $U_{ne(i)}^{k-1}$ but in the single-stage solutions is optimized from scratch.



(a) A Four-Node Hidden Markov Model



(b) A Four-Node "Frustrated" Loopy Model

Fig. 5. Comparison of the sequence of “beliefs” (i.e., approximation of the true posterior marginals) in our network-constrained strategy via (9) with those in the (unconstrained) BP algorithm—shown are the “two-sigma” error bars based on 1000 samples from the joint process (X, Y) . In (a), BP always converges to the true posterior marginal $p(x_i|y)$ at every node i by the third stage, and our beliefs remain within statistical significance. In (b), BP typically diverges or oscillates while our beliefs stabilize, with significant performance benefits already apparent by the third stage.

is that, with only a finite number of samples, it is possible that probable triplets are never actually generated, so zeros in the empirical distribution must be handled with care [21].

B. Initial Experiments

One question we have addressed empirically is how well our network-constrained beliefs, generated via (9) using our approximate strategy $\tilde{\gamma} = (\tilde{\mu}, \tilde{\delta})$, compare to those generated by the (unconstrained) BP algorithm. Figure 5 shows results for two different probabilistic models, one with \mathcal{G} a simple four-node chain and the other with \mathcal{G} as shown in Figure 1(a): in both cases, each compatibility function $\psi_{i,j}(x_i, x_j)$ has value 0.1 if $x_i = x_j$ and 0.9 otherwise (which makes the loopy model “frustrated”), while each observation likelihood $p(y_i|x_i)$ is as shown in Figure 2(a) with (conditional) means at $\pm\frac{1}{2}$ and unit-variance. The network-constrained setup assumes a communication model as described in Example 3. The error of the stage- k beliefs is measured by the symmetric Kullback-Liebler distance summed over all nodes, or $\frac{1}{2} \sum_{i=1}^n D(M_i^k(x_i)||p(x_i|y)) + D(p(x_i|y)||M_i^k(x_i))$.

V. CONCLUSION

We close with some forward-looking speculation on our methods as an alternative message-passing paradigm in complex graphical models. The discussion neglects the differences in communication overhead, in which our methods are superior by design. A related advantage in our methods is that online efficiency is tied to sparsity of the given communication graph, which need not bear any relation to

the underlying probability graph; in contrast, loopy BP loses online efficiency for probabilistic models defined on densely-connected graphs. From the performance perspective, in our methods Proposition 1 guarantees improvement over the stage-1 initialization, while loopy BP in the absence of convergence typically fails catastrophically, performing worse than its initialization; on the other hand, when BP does converge, its performance is typically better than that of our network-constrained methods. From the computational perspective, a clear disadvantage of our methods is offline design, an issue entirely absent in BP; on the other hand, per online observation, our processing terminates in only a few iterations (by constraint), whereas BP in even small loopy models is seen to take an order of magnitude more iterations to converge (if it converges). Altogether, in applications where convergence is difficult to guarantee over all probable observations and online computation is far more expensive than offline computation, our methods become an attractive alternative. Even so, whether our methods can scale comparably to the scalability of BP, while preserving the appealing performance trends demonstrated in Figure 5 on only small graphical models, remains to be seen.

REFERENCES

- [1] J Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, 1988.
- [2] C Crick and A Pfeffer, “Loopy BP as a basis for communication in sensor networks,” *UAI*, 19, 2003.
- [3] E Sudderth, et al, “Nonparametric belief propagation,” *CVPR*, 2003.
- [4] M Paskin and C Guestrin, “Robust probabilistic inference in distributed systems,” *UAI*, 20, 2004.
- [5] A Ihler, et al, “Loopy BP: Convergence and effects of message errors,” *JMLR*, 6(11), 2005.
- [6] L Chen, et al, “Data association based on optimization in graphical models with application to sensor networks,” *Mathematical and Computer Modeling*, 43(9-10), May 2006.
- [7] V Saligrama, et al, “Distributed detection with packet losses and finite capacity links,” *IEEE Trans. on SP*, 54(11), 2006.
- [8] O Kreidl and A Willsky, “Inference with minimal communication,” in *NIPS*, 18, 2006.
- [9] O Kreidl and A Willsky, “Decentralized detection in undirected network topologies,” *IEEE SSP Workshop*, 2007.
- [10] O Kreidl and A Willsky, “Decentralized detection with long-distance communication,” *Asilomar SSC*, 2008.
- [11] M Jordan, et al, “An introduction to variational methods for graphical models,” in *Learning in Graphical Models*, MIT Press, 1999.
- [12] M Wainwright and M Jordan, “Graphical models, exponential families and variational inference,” Tech Report 649, UC-Berkeley Department of Statistics, Sep 2003.
- [13] R McEliece, et al, “Turbo decoding as an instance of Pearl’s BP algorithm,” *IEEE J. SAC*, 16(2), 1998.
- [14] T Richardson, “The geometry of turbo-decoding dynamics,” *IEEE Trans. on IT*, 46(1), 2000.
- [15] S Tatikonda and M Jordan, “Loopy belief propagation and Gibbs measures,” *UAI*, 18, 2002.
- [16] T Heskes, “Stable fixed points of loopy BP are minima of the Bethe free energy,” *NIPS*, 16, 2003.
- [17] M Welling and Y Teh, “Approximate inference in Boltzmann machines,” *AI*, 143(1), 2003.
- [18] J Yedidia, et al, “Constructing free-energy approximations and generalized BP algorithms,” *IEEE Trans. on IT*, 51(7), 2005.
- [19] D Koller, et al, “A general algorithm for approximate inference in hybrid Bayes nets,” *UAI*, 15, 1999.
- [20] T Minka, “Expectation propagation for approximate Bayesian inference,” *UAI*, 17, 2001.
- [21] O Kreidl, *Graphical Models and Message-Passing Algorithms for Network-Constrained Decision Problems*. Ph.D. Dissertation, MIT EECS, Cambridge, MA, 2008.