Approximate Inference in Collective Graphical Models

Daniel Sheldon[1] SHELDON@CS.UMASS.EDU
Tao Sun[1] TAOSUN@CS.UMASS.EDU
Akshat Kumar[2] AKSHAT.KUMAR@GMAIL.COM
Thomas G. Dietterich[3] TGD@EECS.OREGONSTATE.EDU
[1]University of Massachusetts, Amherst, MA 01002, USA
[2]IBM Research, New Delhi 110070, India
[3]Oregon State University, Corvallis, OR 97331, USA

Abstract

We study the problem of approximate inference in collective graphical models (CGMs), which were recently introduced to model the problem of learning and inference with noisy aggregate observations. We first analyze the complexity of inference in CGMs: unlike inference in conventional graphical models, exact inference in CGMs is NP-hard even for tree-structured models. We then develop a tractable convex approximation to the NP-hard MAP inference problem in CGMs, and show how to use MAP inference for approximate *marginal* inference within the EM framework. We demonstrate empirically that these approximation techniques can reduce the computational cost of inference by two orders of magnitude and the cost of learning by at least an order of magnitude while providing solutions of equal or better quality.

1. Introduction

Sheldon & Dietterich (2011) introduced *collective graphical models* (CGMs) to model the problem of learning and inference with noisy aggregate data. CGMs are motivated by the growing number of applications where data about individuals are not available, but aggregate population-level data in the form of counts or contingency tables are available. For example, the US Census Bureau cannot release individual records for privacy reasons, so they commonly release low-dimensional contingency tables that classify each member of the population according to a few demographic variables. In ecology, survey data provide counts of animals in different locations, but they cannot identify individuals.

CGMs are generative models that serve as a link between individual behavior and aggregate data. As a concrete example, consider the model illustrated in Figure 1(a) for modeling bird migration from observational data collected by citizen scientists through the eBird project (Sheldon et al., 2008; Sheldon, 2009; Sullivan et al., 2009). Inside the plate, an independent Markov chain describes the migration of each bird among a discrete set of locations: X_t^m represents the location of the mth bird at time t. Outside the plate, *aggregate* observations are made about the spatial distribution of the population: the variable $\mathbf{n}_t$ is a vector whose ith entry counts the number of birds in location i at time t. By observing temporal changes in the vectors $\{\mathbf{n}_t\}$, one can make inferences about migratory routes without tracking individual birds.

In general CGMs, any discrete graphical model can appear inside the plate to model individuals in a population, and observations are made in the form of (noisy) low-dimensional contingency tables (Sheldon & Dietterich, 2011). A key problem we would like to solve is learning the model pararameters (of the individual model) from the aggregate data, for which inference is the key subroutine. Unfortunately, standard inference techniques applied to CGMs quickly become computationally intractable as the population size increases, due to the large number of hidden individual-level variables that are all connected by the aggregate counts.

A key to efficient inference in CGMs is the fact that, when *only* aggregate data is being modeled, the same data-generating mechanism can be described much more compactly by analytically marginalizing away the individual variables to obtain a direct probabilistic model for the sufficient statistics (Sundberg, 1975;

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

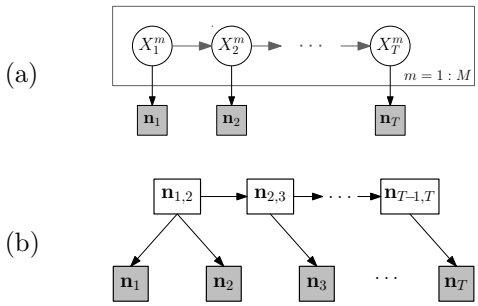

Figure 1. CGM example: (a) Individuals are explicitly modeled. (b) After marginalization, the hidden variables are the sufficient statistics of the individual model.

Sheldon & Dietterich, 2011). Figure 1(b) illustrates the resulting model for the bird migration example. The new hidden variables $\mathbf{n}_{t,t+1}$ are tables of sufficient statistics: the entry $n_{t,t+1}(i,j)$ is the number of birds that fly from location i to location j at time t. For large populations, the resulting model is much more amenable to inference, because it has many fewer variables and it retains a graphical structure analogous to that of the individual model. However, the reduction in the number of variables comes at a cost: the new variables are tables of integer counts, which can take on many more values than the original discrete variables in the individual model, and this adversely affects the running time of inference algorithms.

The first contribution of this paper is to characterize the computational complexity of exact inference in CGMs. For tree-structured graphical models, the running time of exact inference (MAP or marginal) by message passing in the CGM is polynomial in *either* the population size *or* the cardinality of the variables in the individual model (when the other parameter is fixed). However, there is no algorithm that is polynomial in both parameters unless P=NP. This is a striking difference from inference in standard tree-structured graphical models, for which the running time of message passing is always polynomial in the variable cardinality. We also analyze the running time of message passing in a junction tree for general (non-tree-structured) CGMs to draw out another difference between CGMs and standard graphical models: the dependence on clique-width is doubly-exponential instead of singly-exponential for some parameter regimes.

Our second main contribution is an approximate algorithm for MAP inference in CGMs that is based on a continuous and convex approximation of the MAP optimization problem. The algorithm finds a fractional approximation of the most likely sufficient statistics, which are integer counts, given the observations, and it can be used for reconstruction and data exploration. For large populations, which is the regime where exact inference is most difficult, replacing integer counts by continuous values will result in little loss of utility. One can interpret fractional contingency tables as describing percentages of the population instead of absolute counts.

Our final main contribution is to show that approximate MAP inference in CGMs is an excellent approximation for a very important *marginal* inference problem. In particular, to learn the model parameters via the EM algorithm, one must compute the posterior mean—the expected value of the sufficient statistics given the observations—which is a marginal inference problem. Our approximate MAP algorithm computes a fractional approximation to the posterior mode, and we show empirically that it is also an excellent approximation of the posterior mean. For a fixed time budget, it is usually significantly *more* accurate than approximate marginal inference via Gibbs sampling. We show that our approach can be used within an EM algorithm to dramatically accelerate parameter learning while still achieving less than 1% error.

2. Related Work

Sheldon et al. (2008) solved a related MAP inference problem on a chain-structured CGM using linear programming and network flow techniques. Sheldon (2009) extended those algorithms to the case when observations are corrupted by log-concave noise models. The MAP problem in those papers is slightly different from ours: it seeks the most likely setting of all of the individual variables, while we seek the most likely setting of the sufficient statistics, which adds together the probability of *all* possible settings of the individual variables that give the same counts. This gives rise to combinatorial terms in the probability model (see Equation (2) of Section 3) and leads to harder non-linear optimization problems.

Sheldon & Dietterich (2011) generalized the previous ideas from chain-structured models to arbitrary discrete graphical models and developed the first algorithms for marginal inference in CGMs, which were based on Gibbs sampling and *Markov bases* (Diaconis & Sturmfels, 1998; Dobra, 2003). They showed empirically that Gibbs sampling is much faster than exact inference: for some tasks, the running time to achieve a fixed error level is independent of the population size. However, no analogous approximate method was developed for MAP inference.

Inference in CGMs is related to lifted inference in relational models (Getoor & Taskar, 2007). A CGM can be viewed as a simple relational model with only one logical variable to describe the repetition over individuals. A unique feature of CGMs is that *all evidence occurs at the aggregate level*. The most related ideas from lifted inference are counting elimination (de Salvo Braz et al., 2007) and counting conversion (Milch et al., 2008), which perform aggregation operations similar those performed by a CGM. See also (Apsel & Brafman, 2011). Fierens & Kersting (2012) recently proposed to "lift" probabilistic models instead of inference algorithms, which is very similar in spirit to CGMs, but they did not present a general approach to do so. CGMs lift any model with a single logical variable when all evidence is at the population level. In general, while lifted inference algorithms use counting arguments similar to those that appear in CGMs, they cannot reproduce the CGM model, and none of our results are consequences of any of the results in those papers.

Multiple authors in econometrics and statistics have considered the problem of fitting the transition probabilities of a Markov chain from aggregate data by a conditional least squares approach (Lee et al., 1970; MacRae, 1977; Kalbfleisch et al., 1983). Van Der Plas (1983) showed that the conditional least squares estimator is consistent and asymptotically normal under weak assumptions about the Markov model. This problem can be viewed as learning in the CGM in Figure 1, where the graphical model is chain-structured and single-variable contingency tables are observed *exactly* for each node. In our work, the CGM may take on a more general graph structure, some nodes may be unobserved, and the observations may be noisy, so the conditional least squares estimator is not applicable.

3. Problem Statement

In this section, we describe the generative model for aggregate data, introduce collective graphical models, and state the problems of (collective) marginal and MAP inference.

Generative Model for Aggregate Data. The probability model starts with a tree-structured graphical model over random variables $X_1, \ldots, X_N$. Let $\mathbf{x} = (x_1, \ldots, x_N)$ be a particular assignment to the variables (for simplicity, assume that each takes values in $[L] = \{1, \ldots, L\}$), and let $G = (V, E)$ be the independence graph. The probability model is

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{(i,j)\in E} \phi_{ij}(x_i, x_j). \tag{1}$$

Here Z is the normalization constant and $\phi_{ij} : [L]^2 \to \mathbb{R}^+$ are edge potentials. We refer to this as the *individual model*. For the remainder of the paper, we assume that G is a tree to develop the important ideas while keeping the exposition manageable. For a graph with cycles, the methods of this paper can be applied to perform inference on a junction tree derived from G.

To generate the aggregate data, first assume that M independent vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(M)}$ are drawn from the individual probability model to represent the individuals in a population. Aggregate observations are then made in the form of *contingency tables* on small sets of variables, which count the number of times that each possible combination of those variables appears in the population. Specifically, a contingency table $\mathbf{n}_A$ on variable set A is a table with entries

$$n_A(\mathbf{x}_A) = \sum_{m=1}^{M} \mathbf{1}\{\mathbf{x}_A^{(m)} = \mathbf{x}_A\}, \quad \mathbf{x}_A \in [L]^{|A|},$$

where $\mathbf{x}_A^{(m)}$ is the subvector corresponding to the variable subset A. In this section, we will focus first on observed tables $\mathbf{n}_i := \mathbf{n}_{\{i\}}$ over *single* variables, which we refer to as *node tables*. Later, the tables $\mathbf{n}_{ij} := \mathbf{n}_{\{i,j\}}$ for edges (i, j), which are the sufficient statistics of the individual model, will play a prominent role. We will refer to these as *edge tables*.

We will consider two different observation models. Let U be an observed subset of nodes. For each $i \in U$ we observe a table $\mathbf{y}_i$ of the same dimension as $\mathbf{n}_i$, where the entry $y_i(x_i)$ has one of the following distributions:

Exact observations: $y_i(x_i) = n_i(x_i)$.

Noisy observations: $y_i(x_i) \,|\, n_i(x_i) \sim \text{Pois}(\alpha \cdot n_i(x_i))$

The Poisson model is motivated by the bird migration problem, and models birds being counted at a rate proportional to their true density. While it is helpful to focus on these two observation models, considerable variation is possible without significantly changing the results: (1) observations of the different types can be mixed, (2) higher-order contingency tables, such as the edge tables $\mathbf{n}_{ij}$, may be observed, either exactly or noisily, (3) some table entries may be unobserved while others are observed, or, in the noisy model, they may have multiple independent observations, and (4) the Poisson model can be replaced by any other noise model $p(y \,|\, n)$ that is log-concave in n. Sheldon & Dietterich (2011) discuss some of these extensions further.

In this paper, the exact observation model is used to prove the hardness results, while the Poisson model is used in the algorithms and experiments. Since the exact model can be obtained as the limiting case of a log-concave noise model (e.g., $y \mid n \sim \text{Normal}(n, \sigma^2)$

as $\sigma^2 \to 0$), we do not expect that noisy observations lead to a more tractable problem.

Inference Problems. Suppose we wish to learn the parameters of the individual model from these aggregate observations. To do this, we need to know the values of the sufficient statistics of the individual model—namely, $\mathbf{n}_{ij}$ for all edges $(i,j) \in E$. Our observation models do not directly observe these. Fortunately, we can apply the EM algorithm, in which case we need to know the expected values of the sufficient statistics given the observations: $E[\mathbf{n}_{ij} \mid \mathbf{y}]$.

This leads us to define two inference problems: marginal inference and MAP inference. The *aggregate marginal inference* problem is to compute the conditional distributions $p(\mathbf{n}_{ij} \mid \mathbf{y})$ for all $(i,j) \in E$. Although these are finite discrete distributions, the variable $\mathbf{n}_{ij}$ can take a very large number of values,[1] so we typically don't want to represent its distribution explicitly in tabular form. An important special case of marginal inference that does not require a large tabular potential is to compute the $L \times L$ tables of expected values $E[\mathbf{n}_{ij} \mid \mathbf{y}]$, which are the quantities needed for the E step of the EM algorithm in an exponential family. Sheldon & Dietterich (2011) also showed how to generate samples from $p(\mathbf{n}_{ij} \mid \mathbf{y})$, which is an alternative way to query the posterior distribution without storing a huge tabular potential.

The *aggregate MAP inference* problem is to find the tables $\mathbf{n} = (\mathbf{n}_{ij})_{(i,j)\in E}$ that jointly maximize $p(\mathbf{n} \mid \mathbf{y})$. A primary focus of this paper is approximate algorithms for aggregate MAP inference. One reason for conducting MAP inference is the usual one: to reconstruct the most likely value of $\mathbf{n}$ given the evidence as a way of "reconstruction" (e.g., for data exploration). However, a second and important motivation is the fact that the *posterior mode* $p(\mathbf{n} \mid \mathbf{y})$ *is an excellent approximation for the posterior mean* $E[\mathbf{n} \mid \mathbf{y}]$ in this model, so approximate MAP inference also gives an approximate algorithm for the important marginal inference problem needed for the EM algorithm.

Collective Graphical Models. Notice that in the setting we are considering, our observations and queries only concern aggregate quantities. The observations are (noisy) counts and the queries are MAP or marginal probabilities over sufficient statistics (which are also counts). In this setting, we don't care about the values of the individual variables $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(M)}\}$, so we can marginalize them away. This marginalization can be performed analytically to obtain a probability model with many fewer variables. It results in a model whose random variables are the vector $\mathbf{n}$ of sufficient statistics and the vector $\mathbf{y}$ of observations.

For trees (and, more generally, for junction trees), the distribution of $\mathbf{n}$ can be written in closed form given the marginal probabilities $\mu_i(x_i) = \Pr(X_i = x_i)$ and $\mu_{ij}(x_i, x_j) = \Pr(X_i = x_i, X_j = x_j)$.[2] The following expression is originally due to Sundberg (1975):

$$p(\mathbf{n}) = M! \prod_{i\in V} \prod_{x_i} \left(\frac{n_i(x_i)!}{\mu_i(x_i)^{n_i(x_i)}} \right)^{\nu(i)-1} \prod_{(i,j)\in E} \prod_{x_i,x_j} \frac{\mu_{ij}(x_i,x_j)^{n_{ij}(x_i,x_j)}}{n_{ij}(x_i,x_j)!}, \quad (2)$$

subject to:

$$n_i(x_i) = \textstyle\sum_{x_j} n_{ij}(x_i, x_j), \quad \forall i,\ x_i,\ j \sim i \quad (3)$$

$$\textstyle\sum_{x_i} n_i(x_i) = M, \quad \forall i. \quad (4)$$

Here, $\nu(i)$ is the degree of node i, and the notation $j \sim i$ means that j is a neighbor of i. By "subject to" we mean that the probability is zero if the constraints are not satisfied.

The distribution $p(\mathbf{n})$ is the *collective graphical model*, which is defined over the random variables $\{\mathbf{n}_i\}$ and $\{\mathbf{n}_{ij}\}$. The CGM distribution satisfies a *hyper Markov property*: it has conditional independence properties that follow the same essential structure as the original graphical model (Dawid & Lauritzen, 1993). (To see this, note that Eq. (2) factors into separate terms for each node and edge contingency table; when the hard constraints of Eq. (3) are also included as factors, this has the effect of connecting the tables for edges incident on the same node, as illustrated in Figure 1.)

Likelihood. We can combine the CGM with the likelihood term to derive an explicit expression for the (unnormalized) posterior $p(\mathbf{n} \mid \mathbf{y}) \propto p(\mathbf{n})p(\mathbf{y} \mid \mathbf{n})$. Under the Poisson observation model, the likelihood has the form

$$p(\mathbf{y} \mid \mathbf{n}) = \prod_{i\in U} \prod_{x_i} \frac{(\alpha\, n_i(x_i))^{y_i(x_i)} e^{-\alpha n_i(x_i)}}{y_i(x_i)!}.$$

Generalizations. For general graph structures, Sheldon & Dietterich (2011) give a probability model analogous to Eq. (2) defined over a junction tree for the graphical model. In that case, the vector $\mathbf{n}$ includes contingency tables $\mathbf{n}_C$ for each clique C of the junction

[1] There are $\binom{M+L^2-1}{L^2-1} = O(M^{L^2-1})$ different $L \times L$ tables of non-negative integers that sum to M.

[2] If marginal probabilities are not given, they can be computed by performing inference in the individual model.

tree. Different junction trees may be chosen for a particular model, which will lead to different definitions of the hidden variables $\mathbf{n}$ and thus slightly different inference problems, but always the same marginal distribution $p(\mathbf{y})$ over observed variables. Higher-order contingency tables may be observed as long as each observed table $\mathbf{n}_A$ satisfies $A \subseteq C$ for *some* clique C, so it can be expressed using marginalization constraints such as Eq. 3. The approximate inference algorithms in Section 5 extend to these more general models in a straightforward way by making the same two approximations presented in that section for the expression $\log p(\mathbf{n} \mid \mathbf{y})$ to derive a convex optimization problem.

4. Computational Complexity

There are a number of natural parameters quantifying the difficulty of inference in CGMs: the population size M; the number of variables N; the variable cardinality L; and the clique-width K (largest clique size) of the junction tree used to perform inference, which is bounded below by the tree-width of G plus one. The input size is $\text{poly}(N, L, \log M)$. The inputs are: the vector $\mathbf{y}$, the integer M, and the CGM probability model. The vector $\mathbf{y}$ has at most NL entries of magnitude $O(M)$, so each can be represented in $\log M$ bits, and the CGM is fully specified by the potential functions in Equation (1), which have size $O(NL^2)$.

We first describe the best known running time for exact inference in trees ($K = 2$), which are the focus of this paper.

Theorem 1. *When G is a tree, message passing in the CGM solves the aggregate MAP or marginal inference problems in time $O(N \cdot \min(M^{L^2-1}, L^{2M}))$.*

Proof sketch. Because of the hyper-Markov property, the CGM also has the form of a tree-structured graphical model. Message passing gives an exact solution to the MAP or marginal inference problem in two passes through the tree, which takes $O(N)$ messages (Koller & Friedman, 2009). In a standard implementation of message passing, the time per message is bounded by the maximum over all factors of the product of the cardinalities of the variables in that factor. However, due to the nature of the hard constraints in the CGM, it is possible to bound the time per message by a smaller number, which is the number of values for the random variable $\mathbf{n}_{ij}$ (details omitted). The number of contingency tables with c entries that sum to M is $\binom{M+c-1}{c-1}$, which is the number of ways of placing M identical balls in c distinct bins. This number is bounded above by M^{c-1} and by $(c-1)^M$. In a CGM, the table $\mathbf{n}_{ij}$ has L^2 entries, so the number of values for $\mathbf{n}_{ij}$ is $O(\min(M^{L^2-1}, L^{2M}))$, which gives the desired upper bound. □

For general graphical models, message passing on junction trees can be implemented in a similar fashion. For a clique of size K, the contingency table will have L^K entries, so there are $O(\min(M^{L^K-1}, L^{KM}))$ possible values of the contingency table. This gives us the following result.

Theorem 2. *Message passing on a junction tree with maximum clique size K and maximum variable cardinality L takes time $O(N \cdot \min(M^{L^K-1}, L^{KM}))$.*

Thus, if either L or M is fixed, message passing runs in time polynomial in the other parameter. When M is constant, then the running time is exponential in the clique-width, which captures the familiar case of discrete graphical models. When L is constant, however, the running time is not only exponential in L but *doubly-exponential* in the clique-width. Thus, despite being polynomial in one of the parameters, message passing is unlikely to give satisfactory performance on real problems. Finally, the next result tells us that we should not expect to find an algorithm that is polynomial in both parameters.

Theorem 3. *Unless P=NP, there is no algorithm for MAP or marginal inference in a CGM that is polynomial in both M and L. This remains true when G is a tree and $N = 4$.*

Proof of Theorem 3. The proof is by reduction from exact 3-dimensional (3D) matching. An instance of exact 3D matching consists of finite sets A_1, A_2, and A_3, each of size M, and a set of hyperedges $T \subseteq A_1 \times A_2 \times A_3$. A hyperedge $e = (a_1, a_2, a_3)$ is said to *cover* a_1, a_2, and a_3. The problem of determining whether there is a subset $S \subseteq T$ of size M that covers each element is NP-complete (Karp, 1972).

To reduce exact 3D matching to inference in collective graphical models, define a graphical model with random variables X_0, X_1, X_2, X_3 such that $X_0 \in \{1, \ldots, |T|\}$ is a hyperedge chosen uniformly at random, and X_1, X_2, and X_3 are the elements covered by X_0 (see Figure 2). Define the observed counts to be $n_i(a) = 1$ for all $a \in A_i$, $i = 1, 2, 3$, which specify that M hyperedges are selected that cover each element exactly once. These counts have nonzero probability if and only if there is an exact 3D matching. Thus, MAP or marginal inference can be used to decide exact 3D matching. For MAP, there exist tables $\mathbf{n}_{0i}$ such that $p(\{\mathbf{n}_{0i}\}, \{\mathbf{n}_i\}) > 0$ if and only if there is a 3D matching. For marginal inference $p(\{\mathbf{n}_i\}) > 0$ if and only if there is a 3D matching.

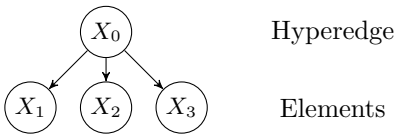

Figure 2. Reduction from 3-dimensional matching.

Because the model used in the reduction is a tree with only four variables, the hardness result clearly holds under that restricted case. □

5. Approximate MAP Inference

In this section, we address the problem of MAP inference in CGMs under the noisy observation model from Section 3. That is, the node tables for an observed set U are corrupted independently by noise, which is conditionally Poisson:

$$p(\mathbf{y} \mid \mathbf{n}) = \prod_{i \in U} \prod_{x_i} p\big(y_i(x_i) \mid n_i(x_i)\big), \tag{5}$$

$$p\big(y_i(x_i) \mid n_i(x_i)\big) = \frac{(\alpha n_i(x_i))^{y_i(x_i)} e^{-\alpha n_i(x_i)}}{y_i(x_i)!}. \tag{6}$$

Our goal is to maximize the following objective:

$$\log p(\mathbf{n} \mid \mathbf{y}) = \log p(\mathbf{n}) + \log p(\mathbf{y} \mid \mathbf{n}) + \text{ constant} \tag{7}$$

As highlighted earlier, it is computationally intractable to directly optimize $\log p(\mathbf{n}|\mathbf{y})$. Therefore, we introduce two approximations. First, we relax the constraint that the entries of $\mathbf{n}$ be integers. For large sample size M, the effect of allowing fractional values is minimal. Second, as it is hard to incorporate factorial terms $\log n!$ directly into an optimization framework, we employ Stirling's approximation: $\log n! \approx n \log n - n$.

Using these two approximations, we arrive at the following optimization problem:

$$\begin{aligned}
\max_{\mathbf{n}} & \sum_{i \sim j} \sum_{x_i, x_j} \big(\log \mu_{ij}(x_i, x_j) + 1\big) n_{ij}(x_i, x_j) - \sum_{i \in U, x_i} \alpha n_i(x_i) \\
& + \sum_{i \in V} \big(1 - \nu(i)\big) \sum_{x_i} \big(\log \mu_i(x_i) + 1\big) n_i(x_i) \\
& + \sum_{i \in U} \sum_{x_i} y_i(x_i) \log n_i(x_i) - \sum_{i \sim j} \sum_{x_i, x_j} n_{ij}(x_i, x_j) \log n_{ij}(x_i, x_j) \\
& - \sum_{i \in V} \big(1 - \nu(i)\big) \sum_{x_i} n_i(x_i) \log n_i(x_i) + \text{const.},
\end{aligned} \tag{8}$$

subject to (3) and (4), for $n_i(x_i), n_{ij}(x_i, x_j) \in \mathbb{R}^+$.

Theorem 4. *The optimization problem (8) for approximate MAP inference in tree-structured CGMs is convex.*

Proof. The constraints are linear and thus the feasible set is convex. Since this is a maximization problem, we must show that the objective is concave in $\mathbf{n}$, which is clearly true for each term but the last one: the first three terms are linear, and the functions $\log n$ and $-n \log n$ are concave. The last term is convex. However, the *sum* of the last two terms:

$$\begin{aligned}
& \sum_{i \sim j} \sum_{x_i, x_j} -n_{ij}(x_i, x_j) \log n_{ij}(x_i, x_j) \\
& + \sum_{i \in V} \big(1 - \nu(i)\big) \sum_{x_i} -n_i(x_i) \log n_i(x_i)
\end{aligned} \tag{9}$$

is concave over the feasible set. Indeed, this is exactly the expression for the Bethe entropy of a graphical model, and the constraints (3) and (4) are identical to the constraints for pairwise and node marginals used in Bethe entropy. Bethe entropy is concave over the constraint set of a tree-structured graphical model (Heskes, 2006). The only difference between this and the conventional Bethe entropy is that the variables are normalized to sum to M instead of 1, but scaling in this way does not affect concavity. Therefore, the problem is convex. □

MAP Inference for EM. We now describe how MAP inference for CGMs can be used to significantly accelerate the E-step of the EM algorithm for learning the parameters of the *individual model*. Let θ denote the unknown parameter to be optimized, let $\mathbf{y}$ be the observed variables, and let $\mathbf{x} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(M)})$ be the hidden variables for all individuals in the population. The EM algorithm iteratively finds parameters $\theta^\star$ that maximize the following expected log-likelihood:

$$Q(\theta^\star, \theta) = \sum_{\mathbf{x}} p(\mathbf{x} \mid \mathbf{y}; \theta) \log p(\mathbf{x}, \mathbf{y}; \theta^\star)$$

where $\theta^\star$ denotes the parameters to optimize and θ denotes the previous iteration's parameters. When the joint distribution $p(\cdot)$ is from an exponential family, as in our case, then the problem simplifies to maximizing $\log p(\bar{\mathbf{n}}, \mathbf{y}; \theta^\star)$, where $\bar{\mathbf{n}} = E_\theta[\mathbf{n} \mid \mathbf{y}]$ is the expected value of the sufficient statistics $\mathbf{n} = \mathbf{n}(\mathbf{x}, \mathbf{y})$ used to define the model; these are exactly the hidden variables in the CGM. In general, this expectation is difficult to compute and requires the specialized sampling approach of Sheldon & Dietterich (2011).

Instead, we will show that the approximate mode $\mathbf{n}^\star \approx \operatorname{argmax}_{\mathbf{n}} p(\mathbf{n} \mid \mathbf{y}; \theta)$ of the distribution $p(\mathbf{n} \mid \mathbf{y}; \theta)$ is an excellent approximation for its mean $E_\theta[\mathbf{n} \mid \mathbf{y}]$. While this may seem surprising at first, recall that the random variables in question take values that are relatively large non-negative integers. A good analogy is

the Binomial distribution (a CGM with only one variable), for which the mode is very close to the mean, and the mode of the continuous extension of the Binomial pmf is even closer to the mean. Our experiments show that by using the convex optimization approach of Section 5, the approximate mode can be computed extremely quickly and is an excellent substitute for the mean. It is typically a much *better* approximation of the mean than the one found by Gibbs sampling for reasonable time budgets, and this makes the overall EM procedure many times faster.

6. Evaluation

We evaluated our approximate MAP inference algorithm by measuring its accuracy against exact solutions for small problem instances and by comparing it with Gibbs sampling for marginal inference within the E step of the EM algorithm. For all experiments, we generated data from a chain-structured CGM to simulate wind-dependent migration of a population of M birds from the bottom-left to the top-right corner of an $\ell \times \ell$ grid to mimic the seasonal migration of a migratory songbird from a known winter range to a known breeding range. Thus, the variables X_t of the individual model are the grid locations of the individual birds at times $t = 1, \ldots, T$, and have cardinality $L = \ell^2$. The transition probabilities between grid cells were determined by a log-linear model with four parameters that controlled the effect of features such as direction, distance, and wind on the transition probability. The parameters $\boldsymbol{\theta}_{\text{true}}$ were selected manually to generate realistic migration trajectories. After generating data for a population of M birds, we computed node contingency tables and generated observations from the Poisson model ($\alpha = 1$) for every node. Unless specified otherwise, the marginal probabilities $\mu_{ij}(\cdot,\cdot)$ and $\mu_i(\cdot)$ used to define the CGM are those determined by $\boldsymbol{\theta}_{\text{true}}$—that is, we perform inference in the same model used to generate the data.

We solved the approximate MAP convex optimization problem using MATLAB's interior point solver. For the comparisons with Gibbs sampling, we developed an optimized C implementation of the algorithm of (Sheldon & Dietterich, 2011) and developed an adaptive rejection sampler for discrete distributions to perform the log-concave sampling required by that algorithm, based on the ideas of Gilks & Wild (1992).

Accuracy of Approximate MAP Solutions. To evaluate the impact of the two approximations in our approximate MAP algorithm, we first compare its solutions to exact solutions obtained by message passing for small models ($L = 4$, $T = 6$). We expect the fractional relaxation and Stirling's approximation to be more accurate as M increases. Because the running time of message passing in this model is $O(M^{15})$, we are limited to tiny populations. Nevertheless, Figure 3 shows that the relative error $\|\mathbf{n}^*_{\text{MAP}} - \mathbf{n}^*_{\text{exact}}\|_1 / \|\mathbf{n}^*_{\text{exact}}\|_1$ is already less than 1% for $M = 7$. For all M, approximate MAP take less than 0.2 seconds, while the running time of exact MAP scales very poorly.

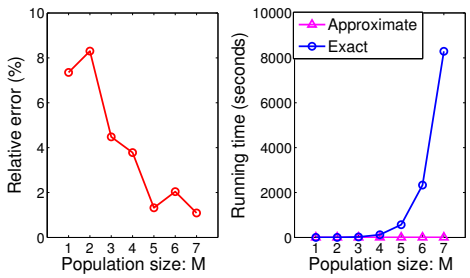

Figure 3. The effect of population size M on accuracy and running time of approximate MAP ($L = 4, T = 6$). Left: relative error vs. M. Right: running time vs. M.

Marginal Inference. We next evaluated the approximate MAP inference algorithm to show that it can solve the EM marginal inference problem more quickly and accurately than Gibbs sampling. For these experiments, we fixed $T = 20$ and varied L by increasing the map size. The largest models ($L = 49$) result in a hidden vector with $(T-1)L^2 \approx 46$K entries. The goal is to approximate $E_\theta[\mathbf{n} \mid \mathbf{y}]$. Since we cannot compute the exact answer for non-trivial problems, we run ten very long runs of Gibbs sampling (10 million iterations), and then compare each Gibbs run, as well as approximate MAP, to the reference solution obtained by averaging the nine remaining Gibbs runs; this yields ten evaluations for each method.

Figure 4 shows that the solver quickly finds an optimal solution to the approximate MAP problem, and it takes Gibbs sampling nearly 100 times longer to reach a solution that is as close to the reference as the one found by approximate MAP. Table 1 shows that the same pattern holds as the problem size increases: for increasing values of L, Gibbs consistently takes 50 to 100 times longer to find a solution as close to the reference solution as the one found by MAP.

We conjecture that the approximate MAP solution may be extremely close to the ground truth. In Figure 4, each Gibbs solution has a relative difference of about 0.09 from the reference solution computed using the other nine Gibbs runs, which suggests that there

Table 1. Comparison of Gibbs vs. MAP: seconds to achieve the same relative error compared with reference solution.

L	9	16	25	36	49
MAP TIME	0.9	1.9	3.4	9.7	17.2
GIBBS TIME	161.8	251.6	354.0	768.1	1115.5

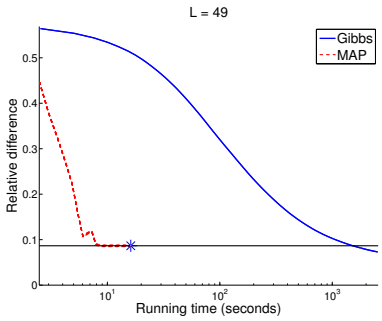

Figure 4. Relative difference from reference solution versus time (log scale) for MAP and Gibbs ($M = 1000$, $L = 49$). Gibbs and MAP: average over 10 trials; 95% confidence intervals are negligible.

is still substantial error in the Gibbs runs after 10 million iterations. Furthermore, each time we increased the number of Gibbs iterations used to compute a reference solution, the MAP relative difference decreased, meaning that the reference solution was getting closer to the MAP solution.

Learning. Finally, we evaluated approximate MAP as a substitute for Gibbs within the full EM learning procedure. We initialized the parameter vector $\boldsymbol{\theta}$ randomly and then ran three variants of the EM algorithm: MAP-EM uses approximate MAP within the E step; Monte Carlo EM (MCEM) uses a fixed number of 100K Gibbs iterations per E step; and stochastic approximation EM (SAEM) uses a smaller number of 10K Gibbs iterations per E step, but it combines those with samples from previous iterations. SAEM usually has better convergence properties than MCEM (Delyon et al., 1999). We employed step sizes of $\gamma_t = 1/t^{0.75}$ within SAEM. In the M step, we applied a gradient-based solver to update the parameters $\boldsymbol{\theta}$ of the log-linear model for transition probabilities. For each algorithm we measured the relative error of the final parameter vector from $\boldsymbol{\theta}_{\text{true}}$.

Figure 5 shows that MAP-EM dramatically outperforms the other two algorithms, especially as the problem size increases. SAEM has better long-term convergence behavior than MCEM, but MAP-EM finds

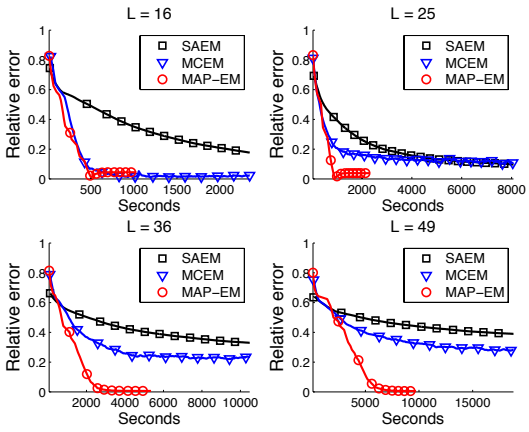

Figure 5. Relative error of learned parameters versus running time for different EM algorithms (MAP: 100 EM iterations, MCEM and SAEM: 400 EM iterations, $M = 1000$).

parameters that are within 1% relative error of $\boldsymbol{\theta}_{\text{true}}$, while SAEM still has relative error greater than 40% after a much longer running time for the larger problems. We conclude that approximate MAP is an excellent substitute for marginal inference with the EM algorithm, both in terms of accuracy and running time.

7. Conclusion

We presented hardness results for the problem of marginal inference and an approximate algorithm for the problem of MAP inference in collective graphical models. We showed that exact marginal inference by message passing runs in time that is polynomial either in the population size or the cardinality of the variables, but there is no algorithm that is polynomial in both of these parameters unless P=NP. We then showed that the MAP problem can be formulated approximately as a non-linear convex optimization problem. We demonstrated empirically that this approximation is very accurate even for modest sized populations and that approximate MAP inference is an excellent substitute for marginal inference for computing the E step of the EM algorithm. Our approximate MAP inference algorithm leads to a learning procedure that is much more accurate and runs in a fraction of the time of the only known alternatives.

Acknowledgments. We thank the anonymous reviewers for constructive comments that improved the manuscript. This material is based upon work supported by the National Science Foundation under Grant No. IIS-1125228.

References

Apsel, U. and Brafman, R. Extended lifted inference with joint formulas. In *Proceedings of the Proceedings of the Twenty-Seventh Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-11)*, pp. 11–18, Corvallis, Oregon, 2011. AUAI Press.

Dawid, A. P. and Lauritzen, S. L. Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics*, 21(3):1272–1317, 1993.

de Salvo Braz, R., Amir, E., and Roth, D. Lifted first-order probabilistic inference. *Introduction to Statistical Relational Learning*, pp. 433, 2007.

Delyon, B., Lavielle, M., and Moulines, E. Convergence of a stochastic approximation version of the EM algorithm. *Annals of Statistics*, 27(1):94–128, 1999.

Diaconis, P. and Sturmfels, B. Algebraic algorithms for sampling from conditional distributions. *The Annals of statistics*, 26(1):363–397, 1998.

Dobra, A. Markov bases for decomposable graphical models. *Bernoulli*, 9(6):1093–1108, 2003.

Fierens, D. and Kersting, K. From lifted inference to lifted models. Second International Workshop on Statistical Relational AI, 2012.

Getoor, L. and Taskar, B. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2007.

Gilks, W.R. and Wild, P. Adaptive Rejection sampling for Gibbs Sampling. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 41(2): 337–348, 1992.

Heskes, T. Convexity arguments for efficient minimization of the Bethe and Kikuchi free energies. *J. Artif. Int. Res.*, 26(1):153–190, 2006.

Kalbfleisch, J. D., Lawless, J. F., and Vollmer, W. M. Estimation in Markov models from aggregate data. *Biometrics*, 1983.

Karp, RM. Reducibility among combinatorial problems. *Complexity of Computer Computations*, 1972.

Koller, D. and Friedman, N. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Lee, T. C., Judge, G. G., and Zellner, A. *Estimating the parameters of the Markov probability model from aggregate time series data*. North-Holland Pub. Co., 1970.

MacRae, E. C. Estimation of time-varying markov processes with aggregate data. *Econometrica: Journal of the Econometric Society*, 1977.

Milch, B., Zettlemoyer, L.S., Kersting, K., Haimes, M., and Kaelbling, L.P. Lifted probabilistic inference with counting formulas. *Proc. 23rd AAAI*, pp. 1062–1068, 2008.

Sheldon, D. and Dietterich, T. G. Collective graphical models. *Advances in Neural Information Processing Systems*, 2011.

Sheldon, D., Elmohamed, M. A. S., and Kozen, D. Collective inference on Markov models for modeling bird migration. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. (eds.), *Advances in Neural Information Processing Systems 20*, pp. 1321–1328. MIT Press, Cambridge, MA, 2008.

Sheldon, Daniel. *Manipulation of PageRank and Collective Hidden Markov Models*. PhD thesis, Cornell University, 2009.

Sullivan, B. L., Wood, C. L., Iliff, M. J., Bonney, R. E., Fink, D., and Kelling, S. eBird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*, 142(10):2282 – 2292, 2009.

Sundberg, R. Some results about decomposable (or Markov-type) models for multidimensional contingency tables: distribution of marginals and partitioning of tests. *Scandinavian Journal of Statistics*, 2(2):71–79, 1975.

Van Der Plas, A. P. On the estimation of the parameters of Markov probability models using macro data. *The Annals of Statistics*, 1983.