Adaptive Rao-Blackwellisation in Gibbs Sampling for Probabilistic Graphical Models

Craig Kelly
University of Memphis
craig.kelly@gmail.com

Somdeb Sarkhel
Adobe Research
sarkhel@adobe.com

Deepak Venugopal
University of Memphis
dvngopal@memphis.edu

Abstract

Rao-Blackwellisation is a technique that provably improves the performance of Gibbs sampling by summing-out variables from the PGM. However, collapsing variables is computationally expensive, since it changes the PGM structure introducing factors whose size is dependent upon the Markov blanket of the variable. Therefore, collapsing out several variables jointly is typically intractable in arbitrary PGM structures. In this paper, we propose an adaptive approach for Rao-Blackwellisation, where we add parallel Markov chains defined over different collapsed PGM structures. The collapsed variables are chosen based on their convergence diagnostics. However, adding a new chain requires burn-in, thus wasting samples. To address this, we initialize the new chains from a mean field approximation for the distribution, that improves over time, thus reducing the burn-in period. Our experiments on several UAI benchmarks shows that our approach is more accurate than state-of-the-art inference systems such as Merlin that implements algorithms that have previously won the UAI inference challenge.

1 Introduction

Probabilistic graphical models (PGMs) [15] are routinely used in several practical problems such as computer vision [22], computational biology [6], medical diagnosis [23], natural language processing [19], etc. A core problem in PGMs is probabilistic inference, which is required both for learning graphical models as well as for prediction. However, exact probabilistic inference is typically intractable for most PGM structures. Therefore, approximate inference methods such as sampling or belief propagation [25] are almost always used for practical problems. Gibbs sampling [8] is arguably one of the most popular MCMC sampling-based approaches for approximate inference in PGMs.

However, despite its widespread use, Gibbs sampling is well-known to have difficulties when the distribution has highly-correlated variables, since the sampler tends to get struck in local modes of the distribution. Rao-Blackwellisation is a strategy that significantly improves the convergence of of Gibbs sampling on hard-to-sample multimodal distributions with correlated variables. However, though Rao-Blackwellised Gibbs sampling is provably better than ordinary Gibbs sampling [16], performing Rao-Blackwellisation effectively and in a scalable manner is a challenging problem. Specifically, we consider Rao-Blackwellisation (or collapsing) in discrete PGMs. In this case, we need to sum-out variables from the PGM, and this implicitly changes the structure of the PGM. More specifically, collapsing a variable creates a factor over all variables in its Markov blanket. Therefore, it is intractable to collapse variables arbitrarily from the PGM. At the same time, collapsing specific variables in the distribution could be more beneficial w.r.t convergence of the sampler. Previous approaches [12, 20, 2] have utilized the structure of the PGM to perform Rao-Blackwellisation tractably. For example, Hamze and Defreitas [12] partition checkerboard Markov Random Fields (MRFs) into disjoint trees, where they sample one tree and conditioned on this, estimate the joint distribution tractbly over the other using belief propagation. Similarly, Bidyuk and Dechter [2] sample a cutset of the PGM such that the induced width on the remaining variables is bounded by a constant. However, though these approaches leverage structural properties of the PGM, they do not exploit sampler history to determine the optimal variables to collapse. At the same time, adaptive sampling approaches [1] have been

Proceedings of the 22[nd] International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

successful in improving convergence in MCMC-based samplers by exploiting sampler history. In this paper, we present an adaptive sampler for Rao-Blackwellised Gibbs sampling that collapses variables in parallel based on their convergence properties.

Our main idea is quite straightforward. We adapt the sampler to collapse slowly converging variables. However, since it may be intractable to collapse all such variables sequentially, we collapse them in parallel. Specifically, we alternate between two steps. In the first step, we choose the optimal variables to collapse based on their convergence statistics, given tractability constraints. In the second step, we add parallel Gibbs samplers with the selected variables collapsed out, and re-compute the convergence statistics. Our final sampler is a mixture of parallel Markov chains where each Markov chain is constructed around a different collapsed PGM structure, but with all marginals converging to the same invariant distribution. A key issue with parallel sampling is that every time we add a new sampler, we need to initialize its state. Typically, Gibbs samplers initialize parallel chains by sampling from a uniform distribution over its state space. However, in such a case, the benefit of adding new collapsed samplers is offset by time spent in the *burn-in* period of the new chain before useful samples can be generated from the collapsed sampler. To address this, we initialize the state of the sampler based on a mean field approximation of the distribution. The parameters of this approximation are improved progressively resulting in better initialization points of the sampler and smaller mixing times.

We evaluate our approach with UAI benchmarks on marginal inference tasks. Our comparison with Merlin [17], a PGM inference system clearly shows that our proposed approach is more accurate than state-of-the-art marginal inference solvers.

2 Background

2.1 Discrete Probabilistic Graphical Models

Probabilistic Graphical models (PGMs) [21, 4, 15] unify graph theory with probabilistic reasoning. The two main categories of PGMs are Bayesian networks which are directed models and Markov networks which are undirected models. Below, we give a brief overview of Markov networks, since from an inference perspective, they are both equivalent to each other [4, 15], and similar algorithms are used in both networks.

A (discrete) PGM or a Markov network, denoted by $\mathcal{M}$ is a pair $\langle \mathbf{X}, \Phi \rangle$ where $\mathbf{X} = \{X_1, \ldots, X_n\}$ is a set of discrete variables (i.e., they take values from a finite domain) and $\Phi = \{\phi_1, \ldots, \phi_m\}$ is a set of positive real-valued functions (or potentials). Each function is defined over one or more variables and the scope of a function is the union of all the variables occurring in ϕ. $\mathcal{M}$ represents a probability distribution called the Gibbs distribution which is the normalized product of all its potentials as given by the following equation.

$$P(\overline{\mathbf{x}}) = \frac{1}{Z} \prod_{\phi \in \Phi} \phi(\overline{\mathbf{x}}) \tag{1}$$

where $\overline{\mathbf{x}}$ is an assignment of values to all variables in $\mathbf{X}$, $\phi(\overline{\mathbf{x}})$ evaluates the factor ϕ with the values to variables in its scope specified by $\overline{\mathbf{x}}$, and Z is a normalization constant called the partition function.

2.2 Gibbs Sampling

Given a PGM $\mathcal{M} = \langle \mathbf{X}, \Phi \rangle$ and observed evidence $\mathbf{E}$, Gibbs sampling begins by initializing all non-evidence variables randomly. In each iteration, we generate a new sample from the previous sample by selecting exactly one variable X and sampling it from its conditional distribution $P(X|\mathbf{X} \setminus X)$. Note that in Markov networks, it is typically easy to compute this conditional probability since given its Markov blanket, a variable is conditionally independent of all other variables in the PGM. Marginal probabilities can be estimated from Gibbs samples using standard Monte Carlo estimators.

Rao-Blackwellisation or collapsing is a technique for improving the accuracy of Gibbs sampling. Collapsing operates by summing-out or marginalizing a subset of variables, say $\mathbf{X}'$, from the PGM. Gibbs sampling is then performed on this smaller PGM, and this reduces variance of estimates because only a sub-space is sampled. Collapsing a variable in a PGM is typically computationally expensive since it creates a new factor (or in terms of the graph, a clique) over all its neighboring variables.

3 Related Work

Liu [16] showed that Rao-Blackwellised Gibbs is provably better than random scan Gibbs sampling, and therefore samplers should always try to collapse variables when possible. Thus, several previous approaches have been proposed for Rao-Blackwellised Gibbs sampling in discrete PGMs. Most of these approaches are non-adaptive samplers that only exploit the structure of the Markov network to collapse a subset of variables tractably. Specifically, Paskin [20] proposed sample propagation, an efficient Rao-Blackwellisation technique for PGMs. Specifically, sample propagation builds samples some vari-

ables and performs exact inference over the others using a junction tree by passing messages efficiently that does not require running the full junction tree (over the collapsed variables) for each sample. Hamze and Defreitas [12] build a Rao-Blackwellised sampler over Ising models, where the PGM is partitioned into two parts, one of which is sampled, and conditioned on this, the exact marginal is computed over the the remaining tree-structured induced graph. Bidyuk and Dechter [2] proposed cutset-sampling, where they sampled a cutset such that the induced width of the remaining network is bounded, and can be solved exactly. Adaptive approaches include the splash Gibbs sampler by Gonzalez et al. [11], where they use a likelihood estimator to compute the optimal variables to sample jointly (blocking), also known as splashes. They run parallel chains to sample from different splashes where the splashes constructed to maintain ergodicity of the sampler. Venugopal and Gogate [24] and Islam et al. [13] proposed approaches to learn correlations, and used these to collapse the sampler efficiently. However, they considered a single collapsed PGM structure which is inherently limited since jointly collapsing several variables quickly becomes intractable. Whereas, in our approach, we sample from multiple collapsed structures in parallel.

4 Adaptive Rao-Blackwellisation

We motivate our approach using a simple example. Consider a pairwise Markov network shown in Fig. 1 (a). Let the variables X_2, X_5 and X_8 are correlated with (X_1, X_3), (X_4, X_6) and (X_7, X_9) respectively. In this case, collapsing X_2, X_5 and X_8 is likely to improve the mixing time of the sampler. However, collapsing all three variables in the PGM yields a factor with six variables. Specifically, after collapsing, there is a clique over the variables X_1, X_3, X_4, X_6, X_7 and X_9. Now, suppose we add a constraint that we can only construct factors of four variables or less for computational reasons, then, clearly, X_2, X_5 and X_8 cannot be collapsed in the same chain. An alternate strategy is to then spawn two parallel Gibbs samplers corresponding to the Markov networks shown in Fig. 1 (b) and (c). Specifically, in Fig. 1 (b), X_2 and X_8 is collapsed, and in Fig. 1 (c) the variable X_5 is collapsed. Our final sampler is now a mixture of the two parallel collapsed samplers.

Formally, given a PGM $\mathcal{M} = \langle \mathbf{X}, \Phi \rangle$, we will partition the variables $\mathbf{X}$ into two sets, one set is collapsed or marginalized-out of $\mathcal{M}$, and the other set of variables are sampled. However, we need to choose the variables to collapse carefully such that it improves mixing time of the sampler. We can formulate this as follows.

$$\min_{\mathbf{X}' \subseteq \mathbf{X}} t_{mix}(P_{-\mathbf{X}'}, \epsilon)$$

where $t_{mix}(P_{-\mathbf{X}'}, \epsilon)$ is the mixing time of the Gibbs sampler where variables $\mathbf{X}'$ have been collapsed out of $\mathcal{M}$. In general, we can define the mixing time as the minimum number of time steps before the total variational distance between the true and approximate distribution is less than a constant. Specifically,

$$t_{mix}(P_{-\mathbf{X}'}, \epsilon) = \min \left\{ t : \max_{\mu} ||P^{(t)}_{-\mathbf{X}'}\mu - P_{\mathcal{M}}||_{TV} \leq \epsilon \right\}$$

where ϵ is a constant, $P^{(t)}$ is the transition matrix after t time steps of the sampler defined on the PGM obtained by collapsing variables $\mathbf{X}'$ from $\mathcal{M}$, $||P^{(t)}_{-\mathbf{X}'}\mu - P_{\mathcal{M}}||_{TV}$ is the total variational distance between the approximate distribution estimated by the sampler after t time steps, given by $P^{(t)}_{-\mathbf{X}'}\mu$ and the true stationary distribution $P_{\mathcal{M}}$. Thus, our task is to select the optimal subset of collapsed variables that minimizes mixing time of a sampler, that samples the remaining variables in the PGM.

However, it is notoriously hard to analytically derive the mixing time for arbitrary PGM structures. Therefore, a standard approach used to analyze whether a sampler has mixed or not is to use convergence diagnostics derived from the drawn samples. In principle, though any convergence diagnostic can be used, in this paper, we adopt a popular approach known as the Gelman and Rubin (GR) diagnostic [7]. Specifically, given multiple chains from different starting points, for a specific parameter θ that is estimated by the sampler, we estimate the between-chain as well as the within-chain variances, and combine the variances together. For mixed chains, the within as well as in-between variances for θ would be small. In our case, we use the diagnostic to determine whether the single-variable marginal probability estimates for each variable in the PGM has converged. However, since we are assuming a discrete PGM, it turns out that the standard GR diagnostic does not work effectively with limited sample-sizes [3, 5]. Since our aim is to get an accurate estimate of the convergence diagnostic as quickly as possible, we use a variant of the GR scheme based on the methods proposed by Deonovic and Smith [5]. Specifically, for each variable, we compute convergence of its marginal probability within and between Markov chains using the Hellinger distance, which is a popular symmetric distance measure for discrete distributions. We then compute the PSRF (potential scale reduction factor) as in the GR diagnostic. Larger values will mean that the chain has not converged. Using this, we search for a subset of variables to collapse such that the sum of the GR diagnostic scores computed for the marginal

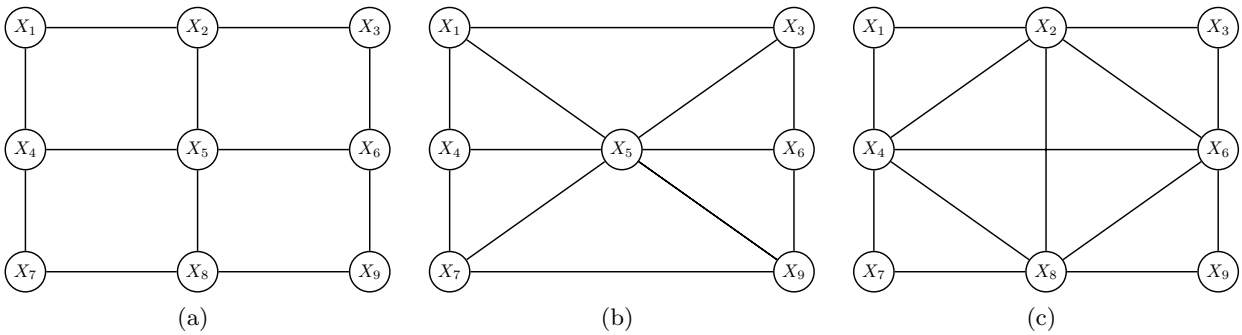

Figure 1: (a) Original PGM (b) X_2 and X_8 collapsed (c) X_5 collapsed

estimates on the un-collapsed variables is minimized. Specifically,

$$\min_{\mathbf{X}' \subseteq \mathbf{X}} \sum_{X \in \mathbf{X} \setminus \mathbf{X}'} GR(P_{-\mathbf{X}'}(X)) \tag{2}$$

where $P_{-\mathbf{X}'}(X)$ is the marginal probability estimate for variable X obtained by running the sampler on the PGM where $\mathbf{X}'$ has been collapsed.

Solving the above optimization problem is clearly computationally hard. Specifically, we need to enumerate over all possible subsets of $\mathbf{X}$, and corresponding to each of the subsets, we need to first collapse the PGM and then compute the GR statistics for un-collapsed variables after running the sampler for a fixed number of time steps. Instead, we develop a more efficient coordinate descent approach to optimize Eq. (2), where we alternate between the following steps. We fix the GR statistics for all variables and choose the optimal subset of variables ($\mathbf{X}'$) to collapse. We marginalize variables $\mathbf{X}'$ from the PGM, and spawn a new Gibbs sampler that estimates the marginals $P_{-\mathbf{X}'}(X)$, where X is an un-collapsed variable. After running the sampler for a fixed number of time steps, we recompute the GR statistics and repeat the aforementioned steps.

However, note that the unconstrained problem specified in Eqn. (2) can lead to a trivial solution where all the variables are selected for collapsing. That is, the mixing time of the sampler is obviously minimzed when there are no variables to sample. However, clearly in arbitrary PGM structures, it is computationally intractable to marginalize all variables. Therefore, we introduce a tractability constraint where the minimum *width* of the collapsed variables should be bounded by a constant. Specifically, given $\mathbf{X}' \subseteq \mathbf{X}$, the width is defined over an ordering of $\mathbf{X}'$, $\pi(\mathbf{X}')$ as the maximum factor-size that is obtained by collapsing variables sequantially in the order $\pi(\mathbf{X}')$. The minimum width is the smallest width over all possible orderings of $\mathbf{X}'$. We add a constraint to Eq. (2) that the minimum width of the variables chosen for collapsing must be bounded by a constant. That is, $w(\mathbf{X}') \leq \alpha$. However, adding this constraint implies that we need to compute the minimum width by enumerating all possible orderings of the subset in the worst case. In general, computing the minumum-width is known to be computationally intractable. Instead, we use well-known heuristics that yield an upper bound to $w(\mathbf{X}')$, such as the *min-degree* or the *min-fill* heuristic. Note that more accurate branch-and-bound based estimates [9] for $w(\mathbf{X}')$ can also be computed, however, they are computationally more expensive.

Our approach to solve Eq. (2) with the tractability constraint proceeds as follows. To start with, we assume that every variable in $\mathcal{M}$ converges at the same rate, i.e., the GR statistics for each marginal probability in $\mathbf{X}$ are equal. In iteration t, conditioned on the GR statistics, we partition $\mathbf{X}$ into the variables that are sampled, $\mathbf{X}_s^{(t)}$, and the variables that are collapsed, $\mathbf{X}_c^{(t)}$, using a greedy approach. Specifically, we select the variable X that has maximum GR value and where $w(X) \leq \alpha$, and add it to $\mathbf{X}_c^{(t)}$, until we can add no more variables. We then spawn Gibbs samplers to sample from $P_{-\mathbf{X}_c^{(t)}}$, and, using samples from them, we update the GR statistics for all variables in $\mathbf{X}_s^{(t)}$. Based on the new GR statistics, we recompute $\mathbf{X}_c^{(t+1)}$ and $\mathbf{X}_s^{(t+1)}$.

4.1 Estimating Marginals

Let $\bar{\mathbf{x}}_{ij}^{(t)}$ represent the j-th sample generated from the i-th sampler spawned in iteration t after the sampler is assumed to have converged. The marginals for $\mathbf{X}_c^{(t)}$ are computed exactly, while the marginals for $\mathbf{X}_s^{(t)}$ are estimated from $\bar{\mathbf{x}}_{ij}^{(t)}$. Note that once we can collapse a variable in one of our iterations, we have access to its exact marginals and do not need to use a sampling-based estimator for that variable. For variables that cannot be collapsed in any iteration, we estimate the marginal probabilities for the sampled variables using

a mixture estimator as,

$$\hat{P}(X) = \frac{1}{T * K * M} \sum_{t=1}^{T} \sum_{i=1}^{K} \sum_{j=1}^{M} P_{-\mathbf{X}_c^{(t)}}(X | \bar{\mathbf{x}}_{ij}^{(t)} \setminus X)$$

where K is the number of parallel chains in each iteration, M is the number of samples collected per chain and T is the number of iterations.

Note that our sampler modifies the transition kernel each time by changing the variables that are collapsed. Specifically, $P_{-\mathbf{X}_c^{(t)}}()$ changes as t changes. This results in the overall sampler being non-ergodic. Specifically, Gonzalez et al. [11] showed that continuous adaptation of the transition probability based on the previous state of the sampler yields a non-ergodic Gibbs sampler. Further, they showed that using *vanishing adaptation*, i.e., stopping the adaptation over time preserves ergodicity of the sampler. To use this result, we stop adaptation after a finite number of steps. Specifically, let $GR(X)^{(t)}$ be the GR statistic for variable X after iteration t, we update this in iteration $t+1$ as β $GR(X)^{(t+1)}$ + $(1-\beta)$ $GR(X)^{(t)}$, and β is decreased in each iteration.

Proposition 1. *As $T \to \infty$, for each variable X in $\mathcal{M}$, the estimated marginal $\hat{P}(X) \to P_{\mathcal{M}}(X)$, where $P_{\mathcal{M}}(X)$ is the true marginal distribution for X.*

Proof. (Sketch) Let $P_{\mathcal{M}}$ be the distribution represented by $\mathcal{M}$. In iteration t, the Gibbs sampler is constructed on the PGM with $\mathbf{X}_c^{(t)}$ collapsed. That is, we draw samples from $P(\mathbf{X}_s^{(t)} = \sum_{\bar{\mathbf{x}} \in \mathbf{X}_c^{(t)}} P(\mathbf{X}_s^{(t)}, \bar{\mathbf{x}})$. There are two possible cases. Let X be a variable that is collapsed in some iteration. This means, we compute the exact marginal for X. Therefore, $\hat{P}(X) = P_{\mathcal{M}}(X)$. Suppose X is not collapsed in any iteration. Then, the estimate for X is derived by the samples from $P(\mathbf{X}_s^{(1)}$, $\ldots$ $P(\mathbf{X}_s^{(T)}$. Since each collapsed distribution leaves the marginal distribution for X invariant, and as $T \to \infty$, $\mathbf{X}_s^{(t-1)} = \mathbf{X}_s^{(t)}$ (vanishing adaptation), from Gonzalez et al. [11], it follows that $\hat{P}(X) \to P_{\mathcal{M}}(X)$.

□

Algorithm 1: Adaptive RB

Input: $\mathcal{M}$ ($\mathbf{X}$,Φ), α
Output: Marginal probabilities for each variable in the PGM

```
// Initialize GR stats
for each X_i in X do
    GR^(1)(X_i) = C
Initialize burn-in B
Initialize adaptation parameter β
for t = 1 to T do
    // Compute the optimal collapsed variables
    X_c^(t), X_s^(t) = Greedy-select based on GR^(t) where
      w(X_c^(t)) ≤ α
    Collapse M to represent P_{-X_c^(t)}(X_s^(t))
    Initialize K parallel Gibbs samplers by sampling from
      the current marginal estimates for P̂(X_i) ... P̂(X_n)
    // Estimate the marginals
    for i = 1 to M do
        if i ≥ B then
            for X ∈ X_s^(t) do
                Update P̂(X)
    for X ∈ X_c^(t) do
        Compute exact marginals and store in P̂(X)
    for X ∈ X_s^(t) do
        GR^(t)(X) = β GR^(t)(X) + (1 − β)GR^(t−1)(X)
    Decrease β and B
Return P̂(X_i) ... P̂(X_n)
```

4.2 Initializing the Markov Chains

In each iteration, we add Gibbs samplers for the collapsed PGM. The common approach to initialize the samplers is to initialize assignments to the variables randomly. However, such an initialization would require a full burn-in before samples can used for estimating the marginal probabilities. Thus, as we add more parallel samplers, we are effectively wasting more samples during the burn-in period, which affects scalability of the sampler. To address this problem, we use an importance distribution to initialize the sampler where we learn the parameters of the importance distribution based on results obtained from the previous samplers. Specifically, we assume that our importance function is fully factorized over the estimated marginal probabilities, i.e., we assume a *mean-field* approximation of the joint distribution.

$$P_{\mathcal{M}}(\mathbf{X}) \approx Q_{\mathcal{M}}(\mathbf{X}) = \prod_{X_i \in \mathbf{X}} \hat{P}(X_i)$$

In each iteration, we initialize the Gibbs sampler by sampling assignments from $\prod_{X \in \mathbf{X}} \hat{P}(X)$. As the marginal estimates become more accurate, the mean-field approximation for the joint distribution improves. Specifically, the KL-distance between the original distribution and the mean field approximation is given by,

$$KL(Q_{\mathcal{M}} || P_{\mathcal{M}}) = \sum_{\mathbf{X}} Q_{\mathcal{M}}(\mathbf{X}) \log \frac{P_{\mathcal{M}}(\mathbf{X})}{Q_{\mathcal{M}}(\mathbf{X})}$$

Using variational inference [14], the optimal param-

eters for $Q_{\mathcal{M}}$ can be obtained by maximizing the Evidence-lower bound (ELBO) which is equivalent to minimizing the KL-divergence between the mean-field approximation and the original distribution. Typically, this is done using a co-ordinate descent procedure where we compute the optimal distribution for each variable in the mean-field approximation independent of the distributions over the other variables. Analytically computing the mean field parameters for a variable in the PGM requires the multiplication of all factors that the variable is involved in. Specifically, if a variable has N variables in its Markov blanket, and is involved in F factors and can take K values, computing the distribution analytically has a complexity $O(KFN^K)$. For PGMs where variables have large Markov blankets, this can be an expensive operation. Therefore, we can think of our approach as a way to estimate the parameters of the approximation $Q_{\mathcal{M}}$ from the samples.

Formally, let $Q*$ be a locally optimal solution for the minimization problem, $\min_{Q \in \mathcal{F}} KL(Q||P_{\mathcal{M}})$, where $\mathcal{F}$ is the family of all possible mean-field approximations for $P_{\mathcal{M}}$. We can show that

Proposition 2. *As $T \rightarrow \infty$, $Q_{\mathcal{M}} \rightarrow Q*$*

Proof. Let $Q*(\mathbf{X}) = q_1(X_1) \dots q_n(X_n)$. The ELBO optimization problem is given by,

$$\min_{q_1 \dots q_n} \mathbb{E}_{Q*(\mathbf{X})}[\log \bar{P}_{\mathcal{M}}(\mathbf{X}) - \log Q*(\mathbf{X})]$$

where $\bar{P}_{\mathcal{M}}(\mathbf{X})$ is the un-normalized probability, which is product of all factors, i.e., $\prod_{i=1}^{m} \phi_i(\mathbf{X})$. Using the closed form solution (cf. [14]) to the above problem, we have,

$$\log q_j(X_j) \propto \mathbb{E}_{q_{-j}}[log \bar{P}_{\mathcal{M}}(X_j)]$$

where $\mathbb{E}_{q_{-j}}$ denotes that the expectation is computed while keeping all the distributions except q_j fixed. $\bar{P}_{\mathcal{M}}(x_j)$ is the product of all factors containing variable x_j. From Proposition 1, the sampled marginals approach the true marginals as $T \rightarrow \infty$. Thus, as $T \rightarrow \infty$, $\forall i$, $\hat{P}(X_i) \rightarrow P(X_i) \propto q_i(X_i)$. Therefore, $Q_{\mathcal{M}} \rightarrow Q*$. □

Thus, after each iteration of our algorithm, the KL divergence between the true distribution and the mean-field approximation is guaranteed to decrease. Thus, as we converge towards more accurate marginal estimates, we reduce the burn-in time in each iteration and waste fewer samples in parallel chains added to the sampler. Note that, even though we may converge to a locally optimal mean-field approximation, recall that we only initialize the un-collapsed variables. Collapsing out different subsets of variables should help the sampler jump across local modes. Our experiments confirm this, where we are able to converge to the true estimates even though we use very small burn-in as we add parallel samplers. A more detailed theoretical analysis of this is part of our future work. To summarize, Algorithm 1 illustrates our complete sampler.

5 Experiments

5.1 Setup

We evaluated our approach on the UAI 2014 marginal inference competetion tasks [10]. We experimented with several problems and present a subset of our results here. Specifically, we present results from sample problems in Alchemy, CSP, Grids, Pedigree and Promedus. Alchemy specifies statistical relational models, CSP specifies constraint satisfaction problems, Grids are Ising models, Pedigree models are used in linkage analysis in biology and Promedus are models used for medical diagnosis. Each of these benchmarks have PGMs of different structures. The true marginal probabilities for these tasks are available in the UAI repository. We measure performance of each algorithm using the Hellinger distance between the approximate marginal and the true marginal. Specifically, given a variable X_i, where the marginal distribution $P(X_i)$ is given by the values $(p_1, \cdots, p_m)$, and the marginal estimate for X_i, $\hat{P}(X_i)$ is given by $(\hat{p}_1, \cdots, \hat{p}_m)$, the Hellinger distance between the two marginal distributions is,

$$H(\hat{P}(X_i), P(X_i)) = \frac{1}{\sqrt{2}} \sqrt{\sum_{j=1}^{m} (\sqrt{\hat{p}_j} - \sqrt{p_j})^2}$$

Note that we divide by $\sqrt{2}$ so that the error metric is $0 \leq H(\cdot, \cdot) \leq 1$. Then for the n variables in the PGM denoted by $\mathbf{X}$, the Maximum Hellinger metric $E = \max_{\hat{X}_i, X_i \in \mathbf{X}} H(\hat{P}(X_i), P(X_i))$. For ease of comparison, we used $-log_2(E)$. The negative log of the Maximum Hellinger distance is presented in all our tables and figures. Thus, higher values are better.

We implemented our Adaptive Rao-Blackwellisation (ARB) algorithm in the Go programming language due to its support for parallelization. Note that since all the benchmarks can be solved by exact inference (to obtain the true marginals), all variables can be collapsed jointly which would simply give us the exact marginal results. Therefore, we use the following process to design the collapsed chains.

After a specified number of iterations (c), we calculate a convergence score per variable, select a variables to collapse, and create new chains with those variables

collapsed. This adaptive process continues for $\frac{s}{2}$ seconds, where s is the maximum time in seconds for sampling. From that time until s seconds (or until the sampler converges), the chains are not altered and sampling continues normally with no adaptive steps. In our tests we used the settings $b = 2$, $a = 4$, $c = 2000$, and $s = 300$: so we began with 2 uncollapsed samplers and added 4 collapsed samplers after every 2000 iterations. This continued for at most 150 seconds ($s/2$), after which we continued to draw samples for at most 150 seconds.

We compared our results to Merlin [17] which implements Iterative Join-Graph Propagation (IJGP, see [18]). It should be noted that IJGP is a state-of-the-art system for marginal inference and has in fact won the UAI inference competition in several categories. We used the Merlin library's default setting for IJGP. In all cases, Merlin converged extremely quickly, and therefore, their results do not show changes over time. We also implemented a non-adaptive Rao-Blackwellised Gibbs sampler (RC) that tractably collapses a subset of variables beforehand. This approach is similar to the one taken by several existing methods for Rao-Blackwellisation [12, 2]. Once again, in order to ensure that we do not collapse all variables, and for a fair comparison, we collapsed the same number of variables as in ARB. Further, we generated the same number of parallel chains as in ARB (but without adaptation) for a fair comparison. We allowed our samplers to run for a maximum of 600 seconds or until convergence. We ran our experiments in an Ubuntu 18.04 environment on a physical machine with 16GB of RAM and 6 CPU's. The CPU's each have 2 hardware threads, for a total of 12 hardware threads available to our implementation.

5.2 Results

Our results are shown in Fig. 2. As shown here, in a majority of the cases, ARB is the best performing method. The results are also summarized in Table 1 that specifies the numerical value for the maximum Hellinger error. The benchmark problems in Grids, CSP and Alchemy (shown only in table for lack of space) converged very fast for all algorithms. On CSP the performance of ARB and RC were more or less similar, while on the others ARB outperformed the other methods. On the Promedas and Pedigree benchmarks, ARB was the best performer followed by RC. Merlin converged much faster but could not improve on its results, and therefore had lower accuracy.

In addition, the relative difficulty of the larger benchmarks (Promedas) allowed us to evaluate our use of mean-field approximation for chain initialization. As mentioned above, during our adaptive phase, new chains are created by collapsed variables chosen with a convergence diagnostic score. However, the burn-in is reduced since the initialization is performed with a mean-field approximation using the marginal estimates. The Promedas results shows that on larger benchmarks, error was continuously decreasing as more chains were added even as the burn-in period was dropping. If the new chains had started in poorly chosen areas of the sample space, then on decreasing burn-in as iterations progressed, we should have had poorer samples which would have caused consistent dips in our accuracy whenever a parallel chain was added, followed by a recovery period where the accuracy would improve as the sample quality improves.

Table 1: UAI Benchmark Results, negative log of Maximum Hellinger (higher is better). RC is Random Collapsed Gibbs Sampling, and ARB is our Adaptive Rao-Blackwellisation technique. Best result is in bold.

Model	Merlin	RC	ARB
Alchemy 11	2.683	4.018	**4.018**
CSP 11	1.128	1.870	**1.870**
CSP 12	0.746	**1.910**	1.860
CSP 13	1.009	**1.883**	1.785
Grids 11	0.039	0.583	**1.253**
Grids 12	0.043	0.707	**1.211**
Grids 13	0.010	0.596	**0.879**
Pedigree 11	0.128	0.680	**0.795**
Pedigree 12	0.459	0.742	**0.984**
Pedigree 13	0.001	0.278	**0.711**
Promedus 11	0.093	1.132	**1.421**
Promedus 12	0.185	0.643	**1.483**
Promedus 13	0.230	1.087	**1.211**

6 Conclusion

Collapsing variables in a discrete PGM is expensive since it changes the PGM structure. In this paper, we presented an adaptive Rao-Blackwellised sampler, where, instead of collapsing several variables sequentially in a single Gibbs sampler (which may be intractable), we construct parallel chains for tractable collapsed structures by identifying the optimal variables to collapse based on convergence diagnostics. However, adding parallel chains wastes samples since the samplers need to be burned-in. We initialized new samplers using the marginal estimates that corresponds to an increasingly accurate mean-field approximation of the distribution, which reduces burn-in time. Our experiments on UAI benchmarks clearly showed that our approach is more accurate than state-of-the-art methods. Future work includes applying our approach to continuous PGMs and other inference tasks.

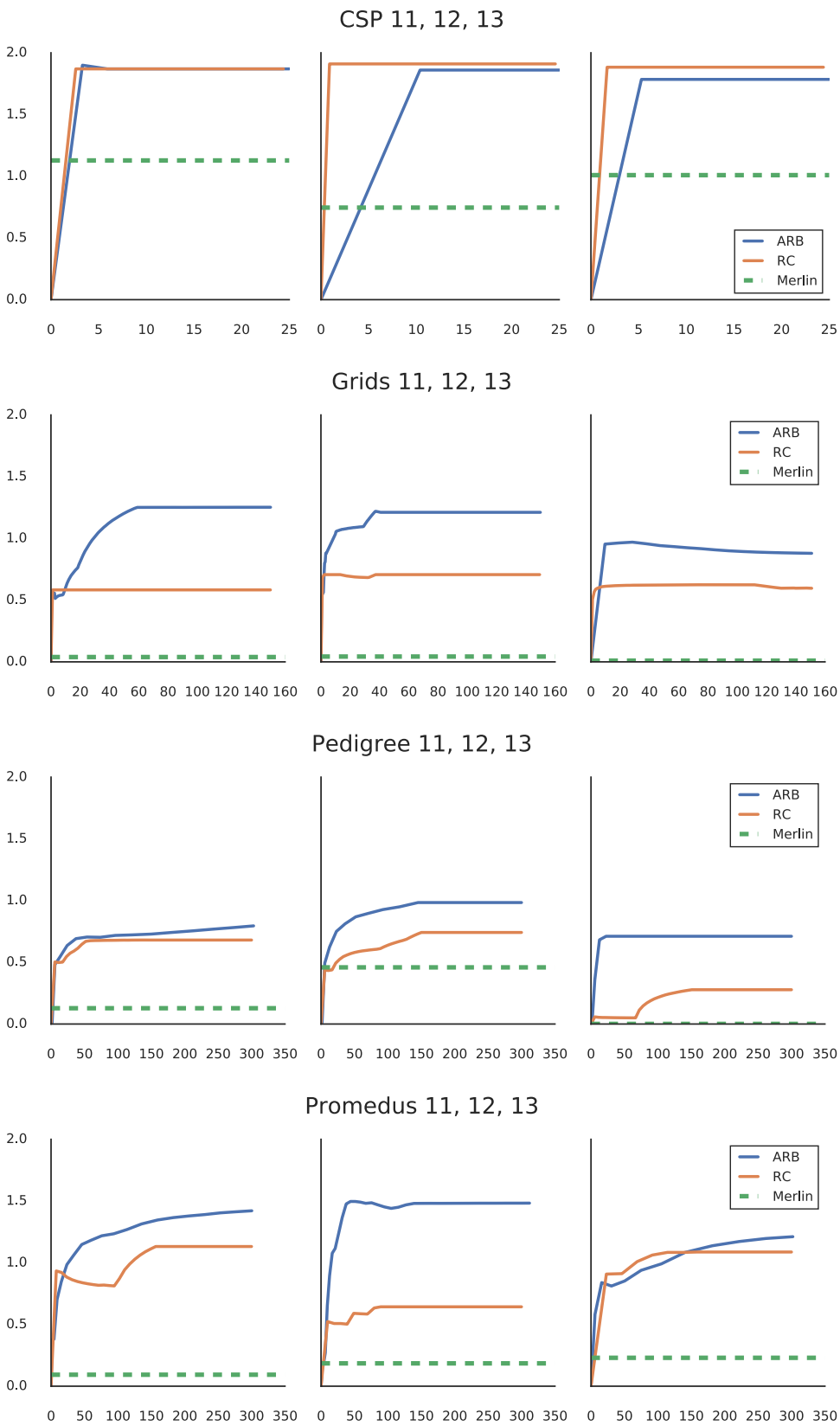

Figure 2: Comparison of ARB, Merlin and RC. X axis is time in seconds; Y axis is negative log of Maximum Hellinger distance (bigger is better). Note that X axes vary by model.

References

[1] Christophe Andrieu and Johannes Thoms. A tutorial on adaptive mcmc. *Statistics and Computing*, 18(4):343–373, Dec 2008.

[2] B. Bidyuk and R. Dechter. Cutset Sampling for Bayesian Networks. *Journal of Artificial Intelligence Research*, 28:1–48, 2007.

[3] Mary Kathryn Cowles and Bradley P. Carlin. Markov chain monte carlo convergence diagnostics: A comparative review. *Journal of the American Statistical Association*, 91:883–904, 1996.

[4] A. Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.

[5] Benjamin Deonovic and Brian Smith. Convergence diagnostics for MCMC draws of a categorical variable. *CoRR*, 2017.

[6] M. Fishelson and D. Geiger. Optimizing Exact Genetic Linkage Computations. *Journal of Computational Biology*, 11(2/3):263–275, 2004.

[7] A. Gelman and D. B. Rubin. Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4):457–472, 1992.

[8] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[9] V. Gogate and R. Dechter. A Complete Anytime Algorithm for Treewidth. In David Maxwell Chickering and Joseph Y. Halpern, editors, *Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 201–208. AUAI Press, 2004.

[10] Vibhav Gogate. Uai inference competetion. Technical report, UT-Dallas, 2014. http://www.hlt.utdallas.edu/ vgogate/uai14-competition/index.html.

[11] J. Gonzalez, Y. Low, A. Gretton, and C. Guestrin. Parallel gibbs sampling: From colored fields to thin junction trees. In *In Artificial Intelligence and Statistics*, May 2011.

[12] F. Hamze and N. de Freitas. From Fields to Trees. In *Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 243–250, 2004.

[13] Mohammad Maminur Islam, Mohammad Khan Al Farabi, and Deepak Venugopal. Adaptive blocked gibbs sampling for inference in probabilistic graphical models. In *International Joint Conference on Neural Networks*, pages 262–269, 2017.

[14] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An Introduction to Variational Methods for Graphical Models. *Machine Learning*, 37(2):183–233, 1999.

[15] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[16] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Publishing Company, Incorporated, 2001.

[17] Radu Marinescu. Merlin:an extensible c++ library for probabilistic inference in graphical models. Technical report, UCI. https://github.com/radum2275/merlin.

[18] R. Mateescu, K. Kask, V. Gogate, and R. Dechter. Iterative Join Graph Propagation algorithms. *Journal of Artificial Intelligence Research*, 37:279–328, 2010.

[19] A. McCallum and B. Wellner. Conditional models of identity uncertainty with application to noun coreference. In *Proceedings of the 18th Annual Conference on Neural Information Processing Systems (NIPS)*, 2004.

[20] M. A. Paskin. Sample Propagation. In *Advances in Neural Information Processing Systems*, pages 425–432, 2003.

[21] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco, CA, 1988.

[22] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal on Computer Vision*, 47(1-3):7–42, April 2002.

[23] M.A. Shwe, B. Middleton, D.E. Heckerman, M. Henrion, E.J. Horvitz, H.P. Lehmann, and G.F. Cooper. Probabilistic diagnosis using a reformulation of the internist-1/qmr knowledge base. i. the probabilistic model and inference algorithms. *Methods of Information in Medicine*, 30(4):241–55, 1991.

[24] Deepak Venugopal and Vibhav Gogate. Dynamic Blocking and Collapsing for Gibbs Sampling. In *Uncertainty In Artificial Intelligence*, 2013.

[25] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized Belief Propagation. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 689–695. MIT Press, Cambridge, MA, 2001.