Contextual Symmetries in Probabilistic Graphical Models

Ankit Anand
Indian Institute of Technology, Delhi
ankit.anand@cse.iitd.ac.in

Aditya Grover
Stanford University
adityag@cs.stanford.edu

Mausam and Parag Singla
Indian Institute of Technology, Delhi
{mausam,parags}@cse.iitd.ac.in

1 Introduction
An important approach for efficient inference in probabilistic graphical models exploits symmetries among objects in the domain. Symmetric variables (states) are collapsed into meta-variables (meta-states) and inference algorithms are run over the lifted graphical model instead of the flat one. Our paper extends existing definitions of symmetry by introducing the novel notion of contextual symmetry. Two states that are not globally symmetric, can be contextually symmetric under some specific assignment to a subset of variables, referred to as the context variables. Contextual symmetry subsumes previous symmetry definitions and can represent a large class of symmetries not representable earlier. We show how to compute contextual symmetries by reducing it to the problem of graph isomorphism. We extend previous work on exploiting symmetries in the MCMC framework to the case of contextual symmetries. Our experiments on several domains of interest demonstrate that exploiting contextual symmetries can result in significant computational gains.

Abstract
An important approach for efficient inference in probabilistic graphical models exploits symmetries among objects in the domain. Symmetric variables (states) are collapsed into meta-variables (meta-states) and inference algorithms are run over the lifted graphical model instead of the flat one. Our paper extends existing definitions of symmetry by introducing the novel notion of contextual symmetry. Two states that are not globally symmetric, can be contextually symmetric under some specific assignment to a subset of variables, referred to as the context variables. Contextual symmetry subsumes previous symmetry definitions and can represent a large class of symmetries not representable earlier. We show how to compute contextual symmetries by reducing it to the problem of graph isomorphism. We extend previous work on exploiting symmetries in the MCMC framework to the case of contextual symmetries. Our experiments on several domains of interest demonstrate that exploiting contextual symmetries can result in significant computational gains.

Finally, our paper extends the line of work on Orbital MCMC [Niepert, 2012] — a state-of-the-art approach to exploit unconditional symmetries in a generic MCMC framework. Orbital MCMC achieved reduced mixing times compared to Gibbs sampling in domains where such symmetries
exist. We design CON-MCMC, an algorithm that uses contextual symmetries within the MCMC framework. Our experiments demonstrate that on various interesting domains (relational and propositional), where contextual symmetries may be present, CON-MCMC can yield substantial gains compared to Orbital MCMC and Gibbs Sampling. We also release a reference implementation of CON-MCMC sampler for wider use.\footnote{https://github.com/dair-iitd/con-mcmc}

\section{Background}

Let $\mathcal{X} = \{X_1, \ldots, X_n\}$ be a finite set of discrete random variables. For ease of exposition, we consider Boolean random variables although our analysis extends more generally to $n$-ary random variables. We denote $s \in \{0, 1\}^n$ to represent a state. A graphical model $\mathcal{G}$ over $\mathcal{X}$ can be represented as the set of pairs $\{(f_k, w_k)\}_{k=1}^m$ where $f_k$ is a formula (feature) over a subset of variables in $\mathcal{X}$ and $w_k$ is the associated weight.\cite{KollerFriedman09}. This is the representation used in several existing models such as Markov logic networks\cite{DomingosL09}.

\subsection{Symmetries in Graphical Models}

Some states may be symmetric; thus, they will have the same joint probability. This fact can be exploited in inference algorithms. To define symmetries, we make use of the formalism of \textit{automorphism groups}, which are generic representations for symmetries between any set of objects. Automorphism groups over graphical models are defined using another algebraic structure called a \textit{permutation group}.

A permutation $\theta$ is a bijection from the set $\mathcal{X}$ onto itself. We use $\theta(X)$ to denote the application of $\theta$ to the element $X \in \mathcal{X}$. We also overload $\theta$ by denoting $\theta(s)$ as the permutation of a state $s$ in which each component random variable $X$ is permuted using $\theta(X)$.

A permutation group $\Theta$ is a set of permutations which contains the identity element, has a unique inverse for every element in the set, and is closed under composition operator.

Following previous work\cite{Niepert12} we define symmetries and automorphism groups in graphical models as follows:

\textbf{Definition 2.1.} A \textit{symmetry} of a graphical model $\mathcal{G}$ over the set $\mathcal{X}$ is represented as the permutation $\theta$ of the variables in $\mathcal{X}$ that maps $\mathcal{G}$ back on to itself, i.e. results in the same set of weighted formulas.

\textbf{Definition 2.2.} An \textit{automorphism group} of a graphical model $\mathcal{G}$ is defined as the permutation group ($\Theta$) over $\mathcal{G}$ such that each $\theta \in \Theta$ is a symmetry of $\mathcal{G}$.

This definition of an automorphism group of a graphical model is analogous to that of an automorphism group of an edge-colored graph, where variables in $\mathcal{G}$ act as vertices in graph, features act as edges (or hyperedges) and weights act as colors on edges. We next define an orbit of a state.

\textbf{Definition 2.3.} The \textit{orbit} ($\Gamma$) of a state $s$ under the automorphism group $\Theta$ is defined as all the states that can be reached by applying a symmetry $\theta \in \Theta$ on the variables of $s$, i.e., $\Gamma_\Theta(s) = \{s' \in \{0, 1\}^n \mid \exists \theta \in \Theta \text{ s.t. } \theta(s) = s'\}$.

Henceforth, we will refer to these unconditional symmetries of a graphical model as \textit{orbital symmetries}. Let $P(s)$ be the probability distribution defined by model $\mathcal{G}$ over the states.

\textbf{Theorem 2.1.} If $\Theta$ is an automorphism group of $\mathcal{G}$, then $\forall s' \in \Gamma_\Theta(s): P(s) = P(s')$, i.e. orbital symmetries of a graphical model are probability preserving transformations.

Therefore, the automorphism group $\Theta$ for a graphical model is also referred to as an automorphism group for the underlying probability distribution. The symmetries of a graphical models as defined above can be obtained via solving a graph automorphism problem. Though the problem is not known to be in P or NP-complete,\footnote{A quasipolynomial time algorithm\cite{Babai15} has been proposed recently for the related graph isomorphism problem which remains to be verified.} efficient solutions can be obtained using the software such as Saucy and Nauty\cite{Darga08,McKayP14}.

\subsection{Markov chain Monte Carlo}

Markov chain Monte Carlo (MCMC) is a popular approach for approximate inference in graphical models. A Markov chain is set up over the state space such that its stationary distribution is the same as the underlying probability distribution. An \textit{Orbital Markov chain}\cite{Niepert12} exploits the orbital symmetries of a model by setting up a Markov chain combining the original MCMC moves with orbital moves. Let $M_o$ denote an orbital Markov chain and $M$ be the corresponding original Markov chain. Then, given the current state $s^{(t)}$, the next state $s^{(t+1)}$ in $M_o$ is sampled as follows:

- \textbf{original move:} sample an intermediate state $s'_o^{(t)}$ from $s^{(t)}$ based on the transition probability in $M$.
- \textbf{orbital move:} sample a next state $s^{(t+1)}$ uniformly at random from the orbit $\Gamma_\Theta(s'_o^{(t+1)})$.

Orbital MCMC converges to the same stationary distribution as the original MCMC and is shown to have significantly faster convergence properties.
3 Contextual Symmetries

Our work proposes the novel notion of contextual symmetries – symmetries that only hold under a given context. We now extend the definitions of the previous section to their contextual counterparts. First we define a context.

Definition 3.1. A context $C$ is a partial assignment, i.e., a set of pairs $(X_i, x_i)$, where $X_i \in X$ and $x_i \in \{0, 1\}$, and no $X_i$ is repeated in the set.

For example, in Figure [1] we can define a context (Genre, “Romantic”). We refer to a context as a single variable context if there is only one element in the context set.

We say that a variable $X_i$ appears in a context if there is a pair $(X_i, x_i) \in C$. Given a context $C$, we will use $X_C$ to denote the subset of variables of $X$ which appear in $C$. We will use $X'_C$ to denote the complement of this set. Given a state $s$, we will use $\zeta_X(s)$ to denote the value of $X_i$ in state $s$. We say that a state $s$ is consistent with the context $C$ iff $\forall (X_i, x_i) \in C$ we have $x_i = \zeta_X(s)$.

In order to define contextual automorphism, we will need to define the notion of a reduced model.

Definition 3.2. Given a graphical model $G = \{f_k, w_k\}_{k=1}^n$, and a context $C$, the reduced model $G'_C$ is defined as the new graphical model obtained by substituting $X_i = x_i$ in each formula $f_k$ for all $(X_i, x_i) \in C$ and keeping the original weights $w_k$.

Note that $G'_C$ is defined over the set $X'_C$. As an example, if the model is represented by the formulas $\{(P \lor Q, w_1), (R \lor Q \lor S, w_2)\}$, the reduced model under the single variable context $\{(R, 0)\}$ will be $\{(P \lor Q, w_1)\}$ and $(Q \lor S, w_2)$. In the factored form representation, reduction by a context corresponds to fixing the values of the context variables in the potential table. E.g., in Figure [1] given the context “Romantic”, we reduce the factor to the bottom four rows of the potential table where Genre has value “Romantic”. We are now ready to define a contextual symmetry of a graphical model.

Definition 3.3. A contextual symmetry of a graphical model $G$ under context $C$ is represented as a permutation $\theta$ of variables in $X$ s.t. a) $\theta(X_i) = X_i$, $\forall X_i \in X_C$ i.e. variables in the context are mapped to themselves, and b) $\exists$ an orbital symmetry $\theta'$ of the reduced model $G'_C$ such that $\theta'(X_i) = \theta'(X_i)$, $\forall X_i \in X_C$, i.e. mapping of the remaining variables defines an orbital symmetry of the reduced graphical model under context $C$.

For example, in Figure [1] let a permutation $\theta^*$ be: $\theta^*(G) = G$, $\theta^*(A) = B$, $\theta^*(B) = A$. $\theta^*$ is a contextual symmetry under the context (Genre, “Romantic”), but not under the context (Genre, “Thriller”).

Definition 3.4. A contextual automorphism group of a graphical model $G$ under context $C$ is defined as a permutation group $(\Theta_C)$ over $G$, such that each $\theta \in \Theta_C$ is a contextual symmetry of $G$ under context $C$.

Definition 3.5. The contextual orbit of a state $s$ under the contextual automorphism group $\Theta_C$ (given the context $C$) is the set of those states which are consistent with $C$ and can be reached by applying $\theta \in \Theta_C$ to $s$, i.e., $\Gamma_{\Theta_C}(s) = \{s' \in \{0, 1\}^n \mid \exists \theta \in \Theta_C \text{ s.t. } \theta(s) = s' \land \forall (X_i, x_i) \in C, \zeta_X(s') = x_i\}$.

Note that $s$ must be consistent with $C$ for it to have a non-empty contextual orbit. Analogous to orbital symmetries, contextual symmetries are also probability preserving.

Theorem 3.1. A contextual symmetry $\theta$ of $G$ under context $C = \{(X_i, x_i)\}$ is probability preserving ($P(s) = P(\theta(s))$), as long as $s$ is consistent with $C$.

3.1 Relationship with Related Concepts

The set of contextual symmetries subsumes that of orbital symmetries – any orbital symmetry is a contextual symmetry under a null context $\emptyset$. The two notions are even more related, as the following two lemmas show. Let $X'_i$ be the set of variables that map onto itself in a permutation $\theta$, i.e. $\forall X \in X'_i : \theta(X) = X$.

Lemma 1. An orbital symmetry $\theta$ is a contextual symmetry under a context $C$ if $X_C \subseteq X'_i$.

Lemma 2. Let $V \subseteq X$. If a permutation $\theta$ is a contextual symmetry of $G$ under all possible contexts $C_i$ where $X_{C_i} = V$, then $\theta$ is an orbital symmetry of $G$.

We now distinguish the notions of context and contextual symmetries from two other related concepts. First, a context is different from evidence. External information in the form of evidence modifies the underlying distribution represented by the graphical model. In contrast, a context has no effect on the underlying distribution.

Second, it might be tempting to confuse contextually symmetric states with contextually independent states [Boutilier et al., 1996]. In the example of Figure [1] given Genre=“Thriller”, A and B are contextually independent, i.e., probability of $A$ does not change depending on $B$. For this context A and B are non-symmetric. For Genre=“Romantic” A and B are symmetric but not independent.

Finally, in Section [7] we discuss the relationship between contextual symmetries and the recent notion of conditional decomposability [Niepert and Van den Broeck, 2014].

3.2 Computing contextual symmetries

Computing contextual symmetries for $G$ under a context $C$ is equivalent to computing orbital symmetries on the reduced model $G'_C = \{f'_k, w'_k\}_{k=1}^n$. To compute orbital symmetries we adapt the procedure from Niepert [2012]. Following Niepert, we describe the construction when each $f'_k$ is a clause, though it can be extended to the more general case.

Niepert’s procedure creates a colored graph, with two nodes corresponding to every variable (one each for the positive and negative state), and one node for every formula $f'_k$. Edges exist between the positive and negative states of every variable, and also between the formula nodes and the variable nodes (either positive or negative) appearing in the formula. Finally, colors are assigned to nodes based on the following criteria: (a) every positive variable node is assigned a common color, (b) every negative variable node is assigned a different common color, and (c) every unique formula weight $w'_k$ is assigned a new color. The formula nodes $f'_k$ inherit the color associated with their weight $w'_k$.
This color graph is then passed through a graph isomorphism solver (e.g., Saucy), which computes the automorphism group for $G_C$. This is equivalent to computing contextual automorphism group for $G$ under $C$.

**Theorem 3.2.** The automorphism group for the color graph of the reduced graphical model $G_C$, along with an identity mapping of the context variables gives a contextual automorphism group for $G$ under $C$.

Note that in case we have any evidence $E$ available, the reduced model over which we induce a colored graph corresponds to $G_{C,E}$. This is in contrast with original Niepert’s procedure, where evidence nodes are not removed from the color graph and instead act as additional formulas for the original graphical model with infinity weights. This elimination of evidence nodes helps discover many more symmetries in the corresponding color graph while still preserving correctness. For example, if the model is represented by formulas $(P \vee R, w_1), (Q, w_2)$, and evidence is $(\neg R), P$ and $Q$ become symmetric only if $R$ is eliminated from the color graph, and not in Niepert’s procedure.

4 **Contextual MCMC**

We now extend the Orbital MCMC algorithm from Section 2.2 so that it can exploit contextual symmetries; our algorithm is named CON-MCMC, and is parameterized by $\alpha \in [0,1]$. Orbital MCMC reduces mixing times over original MCMC, because it can easily transition between high probability states falling in the same orbit, which may otherwise be separated by low probability regions. Unfortunately, as Figure 4 demonstrates, a domain may have little orbital symmetry, but still important contextual symmetry. CON-MCMC($\alpha$) exploits these for inference.

We are given a set of context variables $V \subset X$ (more on this later). Let $C_V$ denote the set of all possible contexts involving all the variables in $V$. Overloading the notation, we will use $C_V(s)$ to denote the (unique) context in $C_V$ consistent with state $s$. We compute contextual symmetries $\Theta_C$ under each context $C \in C_V$ using the algorithm from Section 3. We are also given an original regular Markov chain $M$ that samples a state $s(t+1)$ from $s(t)$ as follows:

1. **Gibbs-orig move**: Sample an intermediate state $s(t+1)$ from the current state $s(t)$ as:
   - (a) with probability $\alpha$ (Gibbs): flip a random context variable in $s(t)$ using Gibbs transition probability.
   - (b) with probability $1-\alpha$ (original): make the move from $s(t)$ based on the transition probability in $M$.

2. **con-orbital move**: Let $C = C_V(s(t+1))$ be the context consistent with $s(t+1)$. Let $\Gamma_{\Theta_C}(s(t+1))$ denote the contextual orbit of $s(t+1)$ under the context $C$. Sample a state $s(t+1)$ uniformly at random from $\Gamma_{\Theta_C}(s(t+1))$.

When $\alpha = 0$ our algorithm reduces to a direct extension of Orbital MCMC, where in second step, we sample uniformly from a contextual orbit instead of the original orbits. In the more interesting case of $\alpha > 0$, we enable the Markov chain to move more freely between different contexts using a Gibbs flip over the context variables. This Gibbs transition helps us carry over the effect of symmetries exploited under one context (via the orbital moves in step 2) to others. This can be especially useful when symmetries are unevenly distributed across multiple contexts (as also confirmed by our experiments).

In order to sample a state uniformly at random from a contextual orbit, we use the product replacement algorithm [Pak, 2000] as described and used by Niepert [2012]. Recall that since we are working with contextual permutations, the context variables are mapped to themselves and we are guaranteed to not change the context. Next, we show that CON-MCMC($\alpha$) converges to the desired stationary distribution $\pi(s)$. We need the following lemma.

**Lemma 3.** Let $M_1$ and $M_2$ be two Markov chains defined over a finite state space $S$ with transition probability functions $P_1$ and $P_2$, respectively, such that $\pi(s)$ is a stationary distribution for both $P_1$, $P_2$, i.e., $\pi(s) = \sum_{r \in S} P(r \rightarrow s)$, $i \in \{1, 2\}$. Further, let $M_2$ be regular. Then, the Markov chain $M'$ with the transition function $P'(s \rightarrow r) = \alpha \ast P_1(s \rightarrow r) + (1 - \alpha) \ast P_2(s \rightarrow r)$ is also regular and has a unique stationary distribution $\pi(s)$ for $\alpha \in [0, 1]$. Let $M^{\alpha}(\alpha)$ refer to the family of Markov chains constructed using only step 1 of our algorithm i.e. no orbital moves. $M$ is regular with the stationary distribution $\pi(s)$. Further, each individual Gibbs flip over a variable satisfies stationarity with respect to the underlying distribution $\pi(s)$ [Koller and Friedman, 2009]. Hence, using Lemma 3, $M^{\alpha}(\alpha)$ is regular with $\pi(s)$ as a stationary distribution.

**Theorem 4.** The family of contextual Markov chains $M^{\alpha}(\alpha)$ constructed using CON-MCMC($\alpha$) converges to the stationary distribution of original Markov chain $M$ for any choice of context variables $V$ and $\alpha \in [0, 1]$.

**Proof.** Let $\pi(s)$ be the stationary distribution of $M$. Since $M^{\alpha}(\alpha)$ is regular it is easy to see that $M^{\alpha}(\alpha)$ is also regular whenever $\alpha$ is always a non-zero probability of coming back to the same state in an orbital move). Therefore, $M^{\alpha}(\alpha)$ converges to a unique stationary distribution. Then, we only need to show that $\pi(s)$ is the stationary distribution of $M^{\alpha}(\alpha)$. Let $S = \{0,1\}^n$ denote the set all of all the states. For $r, s \in S$, let $P^{\alpha}(\alpha)(s \rightarrow r)$ and $P^{\alpha}(\alpha)(s \rightarrow r)$ represent the transition probability functions of $M^{\alpha}(\alpha)$ and $M^{\alpha}(\alpha)$, respectively. In order to show that $M^{\alpha}(\alpha)$ also converges to $\pi(s)$, we need to show that:

$$
\pi(s) = \sum_{r \in S} \pi(r) P^{\alpha}(\alpha)(r \rightarrow s)
$$

The RHS of the above equation can be written as:

$$
= \sum_{r \in S} \sum_{s' \in \Gamma_{\Theta_C}(s)} P^{\alpha}(\alpha)(r \rightarrow s') \frac{1}{|\Gamma_{\Theta_C}(s)|}
$$

$$
= \sum_{r \in S} \sum_{s' \in \Gamma_{\Theta_C}(s)} \pi(r) P^{\alpha}(\alpha)(r \rightarrow s') \frac{1}{|\Gamma_{\Theta_C}(s)|}
$$
we use training groups of subsets of students in a dorm (based on their training groups) independent of their training groups. In this domain, different the probability that they may also team up to play doubles in-participation of other students in her training group for that sport. The league for a given sport is (jointly) influenced by the participation of youngsters or older folks. For every person $X$ in the domain, we have predicates $\text{Smokes}(X)$, $\text{Cancer}(X)$ and $\text{EatsOut}(X)$. We also have the predicate $\text{Friends}(X,Y)$ for every pair of persons. We have rules stating that young persons are more likely to smoke and older people are less likely to smoke. Similarly, we have rules stating that young people are more likely to eat out and old people are less likely to eat out. When the population is young, everyone has the same weight for the smoking rule and slightly different weight (sampled from a Gaussian) for eating out. When the population is old, everyone has a slightly different weight (again sampled from a Gaussian) for smoking and the same weight for eating out. As in the original FS, we have rules stating smoking causes cancer and friends have similar smoking habits. We also have rules stating that cancer and friends variables have low prior probabilities. In this domain, smoking, cancer and friends variables are symmetric to each other when population is young, whereas all eating out variables are symmetric when the population is old. Clearly, $\text{IsYoung}$ is a natural choice for context in this domain.

An important property of both these domains is that different contextual symmetries exist for both assignments of the respective context variables. To test the robustness of CON-MCMC we further modify these domains so that contextual symmetries exist only on one of the two assignments of context variable. In Y&C (Single), we give (slightly) different weights to $\text{EatsOut}(X)$ variables when $\text{IsYoung}$ is false, i.e., symmetries exist only when $\text{IsYoung}$ is true. In Sports Network (Single), $S_X$ variables involved in a training group are symmetric only for tennis; for badminton, each $S_X$ in a group behaves (slightly) differently. We refer to these two variations as the Single side versions of the original domains.

For these four domains, we plot run time vs. the KL-divergence between approximate marginal probabilities computed by each algorithm and the true marginals\footnote{computed by running a Gibbs sampler for sufficiently long time.}. For both Orbital MCMC and CON-MCMC, the time to compute symmetries is included in the run time. For each problem we run 20 iterations of each algorithm and take the mean of the marginals to reduce variance of the measurements. We also plot the 95\% confidence intervals. We show CON-MCMC results for $\alpha = 0$ and 0.01, which was chosen based on performance on smaller problem sizes. We perform various control experiments by varying the size of domains, amount of available evidence, marginal posterior probability of the context variable and the value of $\alpha$ parameter. All the experiments are run on a quad-core Intel i-7 processor.

5.1 Domains and Methodology

**Sports Network:** This Markov network models a group of students who may enter a future sport league, which could be for one of two sports, badminton or tennis (modeled as the variable $\text{Sport}$). Each student belongs to one of the dorms on campus. The league accepts both singles as well as doubles. For each student $X$, the domain has a variable for playing singles, $S_X$. For each pairs of students $X$, $Y$ coming from the same dorm, we have a variable indicating that they will play doubles together, $D_{XY}$. Multiple students (in the same dorm) play together, $D_{XY}$. Multiple students in the league for a given sport is (jointly) influenced by the participation of other students in her training group for that sport. Moreover, if two students decide to play singles, it increases the probability that they may also team up to play doubles independent of their training groups. In this domain, different subsets of students in a dorm (based on their training groups) are symmetrical to each other depending upon $\text{Sport}$, which becomes a natural choice for the context. In our experiments, we use training groups of 5 students and dorms with 25 students each.

[https://code.google.com/archive/p/lifted-mcmc/](https://code.google.com/archive/p/lifted-mcmc/)
MCMC, but its performance can be closer to Gibbs or CON-MCMC(\(\alpha\)) depending upon the experimental setting. Orbital MCMC does not usually offer much advantage over Gibbs, primarily because these domains don’t have many orbital symmetries. For Sports Network, there are no orbital symmetries at all; Orbital MCMC avoids the overhead of the orbital move and performs at par with Gibbs. For Y\&O Orbital MCMC finds a few symmetries, which don’t particularly help in reducing mixing time. However, it still incurs the overhead of orbital moves, leading to a significantly worse performance compared to Gibbs.

**Variation with Domain Size:** Figure 2(a) compares the algorithms as we increase the domain size for the Sports network from 50 to 200 students. The overall trends remain similar, i.e., CON-MCMC algorithms outperform Gibbs and Orbital MCMC by huge margins. A closer look reveals that the y-axes are at different scales for the three curves – the relative edge of CON-MCMC algorithms increases substantially with larger domain sizes.

**Variation with Amount of Evidence:** Figure 2(c) compares the performance of the algorithms as we vary the amount of (random) evidence available from 0% to 60% in the Y\&O domain on predicates other than \(\text{Friends}(X,Y)\) using a domain size of 50. As earlier, CON-MCMC algorithms outperform others. We observe that the relative gain of CON-MCMC algorithms with Orbital MCMC decreases with increasing evidence (for 30% evidence Orbital MCMC overlaps with Gibbs, for 60% evidence, Orbital MCMC overlaps with CON-MCMC). We believe that this is due the fact that more evidence tends to disconnect the network introducing additional symmetries which can be exploited by Orbital MCMC. Nevertheless, CON-MCMC algorithms perform at least as well as Orbital MCMC for all values of evidence that we tested on.

**Variation across Versions of a Domain:** Figure 2(b) and 2(d) show the plots for the Single side versions of Sports network and Y\&O, respectively. We observe a significant difference in the performance of the two CON-MCMC algorithms. The reason is subtle. Since symmetries exist only on one side, that side mixes quickly for CON-MCMC(0); however, the other side does not mix as well, because of lack of symmetries. CON-MCMC(\(\alpha\)) for \(\alpha > 0\) mitigates this by upsampling the flip of the context variable. This enables the rapid mixing on symmetry side to regularly influence the non-symmetry side (via Gibbs move), which leads to a faster mixing on that side too. Nevertheless, CON-MCMC(0) is still able to outperform both Gibbs sampling as well as Orbital MCMC by exploiting the single sided symmetry. The posterior of symmetry side is 0.33 in Sports network (Figure 2(b)) and 0.37 in Y\&O (Figure 2(d)).

We also observe in the first graph of Figure 2(c), that CON-MCMC(0) performs somewhat worse than CON-MCMC(0.01). We believe that the reason for performance in this two-sided symmetry domain is similar to the single-sided case. In Y\&O, when \(\text{IsYoung} = \text{true}\), substantial symmetries may exist due to smoking, cancer and friends variables. However, on the other side, the symmetries are far less (only...
for eating out variables). This implies that Con-MCMC(0) will have much faster mixing on one side, but not on the other. On the other hand, Con-MCMC(0.01) will upsample context variable flips and allow the stronger symmetry side to influence the other. In general, Con-MCMC(α) performance is highly robust to varieties of symmetric and asymmetric domains.

Variation with Posterior of Context Variable: We investigate performance on Single-sided domains further by varying the posterior marginal probability of the context variable. Figure 3 shows the results for Sports network (Single) with marginal probability of Sport = tennis varying from 0.09 to 0.91. Note that Sport = tennis side is the side where symmetries exist.

The graphs show an interesting trend. Even for very low marginals, Con-MCMC(0.01) is able to benefit from one sided symmetries. Since the marginal is low we expect any MCMC algorithm to spend most of its time on the non-symmetry side. However, Con-MCMC(0.01) will still go back and forth several times between two sides; each flip to symmetry side and back will help in potentially reaching a different region of the state space leading to better mixing on the non-symmetry side.

Not surprisingly, Con-MCMC(0) does not perform as well for low marginals – it does not get to switch contexts as often, and ends up mixing slowly on the important, non symmetry side. As marginal of the context variable increases, the relative performance of Con-MCMC(0) improves substantially. As marginal becomes high (0.91), both Con-MCMC samplers end up sampling mostly on the symmetry side, and can reap benefits of symmetries similarly. We also conduct these experiments for the Y&O domain and observe a very similar behavior.

Variation with α Parameter: Figure 4 shows the performance of Con-MCMC(α) for different values of α in the range 0.001 to 0.5 for both Sports network (single) and Y&O (single) domains. Our algorithm is fairly robust for values of α between 0.01 and 0.1. Its performance starts to degrade for very low as well as very high values of α. For very low values of α, algorithm’s behavior approaches that of Con-MCMC(0). For very high values of α, the algorithm spends too much time flipping the context variable and not enough time exploring the state space, resulting in poor performance.

Overall, we conclude that Con-MCMC(0.01) is robust to various experimental settings and obtains the best results significantly outperforming Orbital MCMC and Gibbs. This underscores the importance of our contextual symmetry framework for probabilistic inference.

6 Discussion and Future Work

While our work extends the capability of lifted inference to a wider range of settings, it also raises important questions. In many cases, the set V of context variables is known from domain knowledge or domain description especially in relational models. An open question is how to automatically compute a good set V, since trying all possible sets can be prohibitive. We have designed a heuristic approach that greedily chooses the most useful context variable every iteration and adds it to the context set. It uses a few initial rounds of the color passing algorithm [Kersting et al., 2009] to approximate the amount of additional symmetry obtained by making a variable part of the context. More experiments are needed to assess the effectiveness of our approach.

Another important observation is that the set of contextual symmetries may not monotonically increase with increasing context size. This may happen if additional context variables break existing symmetries, since context variables are forced to undergo identity mapping. Then, how do we design algorithms so that their effectiveness monotonically increases with larger contexts in all cases? This is an important direction for future work.

Another question concerns the robustness of performance of symmetry-based inference algorithms. Over the course of our experiments, we tested our algorithms on several domain variations. While in most cases Con-MCMC(0.01) and Con-MCMC(0) performed much better than Gibbs, in rare cases, the performance was worse too. Further investigations revealed two main sources of lower performance.

The first and more prominent cause is the trade-off between mixing speed and sampling time. Because all symmetry-based algorithms run an expensive product replacement algorithm [Pak, 2000] to sample from an orbit, next samples for Con-MCMC (and Orbital MCMC) are generated much slower than Gibbs. In domains where symmetries are prevalent, this slower sampling is mitigated by rapid mixing, but in other domains, it could result in a worse performance. An intelligent wrapper that guesses whether to exploit symmetries or not in a given domain will be crucial for developing...
a robust inference algorithm. The second reason for lower performance is subtle. CON-MCMC(\(\alpha\)) is able to exploit contextual symmetries (even single-sided) in a wide variety of settings, but in one situation it can lose to other algorithms. This happens when the context variable has a huge Markov blanket, so much so that one Gibbs move that flips the context variable becomes overbearingly costly. Since CON-MCMC(\(\alpha\)) upsamples flips of context variables, this can cause significant loss to overall performance, even though the mixing is much faster with respect to number of samples.

Another observation relates to the effect of evidence in a domain. Evidence can both help and hurt symmetries in an inference problem. In some cases, evidence can break existing symmetries and reduce the relative gain of symmetry-based algorithms. In other cases, evidence can break edges and create new symmetries and help them. While in our experiments, we didn’t find \(\text{CON-MCMC}(0.01)\) to be ever worse than Gibbs due to additional evidence, such pathological cases can be constructed.

It would be interesting to see how algorithms other than MCMC can benefit from our contextual symmetry framework. In the future, we would also like to explore approximate contextual symmetries that could make our contribution applicable to several other domains, where exact contextual symmetries cannot be found. We would also like to theoretically analyze the mixing time of CON-MCMC.

7 Related Work

Some papers have discussed methods for computing symmetries under a given evidence [Van den Broeck and Darwiche, 2013; Venugopal and Gogate, 2014; Kopp et al., 2015]. As discussed in Section 3.2 the algorithm for computing contextual symmetries is closely related to computing evidence-based symmetries. The main difference is in the way we use these symmetries for downstream inference.

While our general notion of contextual symmetries is novel it has connections to a few recent works. The RockIt system [Noessner et al., 2013] identifies contextual symmetries in a very special case in which the domain theory has a set of disjunctive clauses of a specific kind \(g_i \lor c\) where each \(g_i\) is a single literal (or its negation). For this setting, \(c\) is a natural context and symmetries among \(g_i\)s can be exploited. RockIt does not provide any general notion beyond this special case. It constructs a reduced ILP for MAP inference instead of marginal inference, as in our case.

There is recent work on exploring connections between the concept of exchangeability of random variables and tractability of probabilistic inference [Niepert and Van den Broeck, 2014]. Our contextual symmetries can be seen as a generalization of their conditional decomposability to conditional partial decomposability where the sufficient statistics are precisely the contextual orbits. Whereas Niepert and Van den Broeck [2014] primarily focus on developing the theory for conditional decomposability, we propose and additionally connect this with the symmetries present in the structure of a graphical model. Further, unlike them we develop an algorithm to compute these conditional decompositions (contextual symmetries in our case) and show how they can be used in practice for efficient probabilistic inference.

As discussed in Section 1 our work builds upon the recent literature on lifted inference that pre-computes explicit domain symmetries using automorphism groups [Niepert, 2012; Bui et al., 2013; Van den Broeck and Niepert, 2015] and exploits them for efficient inference. Our work is most closely related to Orbital MCMC [Niepert, 2012]. Our experimental results show the value of CON-MCMC over Orbital MCMC that does not incorporate contextual symmetries.

Our contextual symmetries are also analogous to conditional symmetries in constraint satisfaction problems (CSPs) [Gent et al., 2005; Walsh, 2006; Gent et al., 2007]. CSP symmetries are called conditional if symmetry groups exist only in a sub-problem of the original CSP, i.e., in a CSP with one or more additional constraints. The CSP problem setting and their actual manifestation in algorithms are quite different from lifted inference, but their definition and use of conditional symmetries is in the same spirit as ours.

8 Conclusions

We present a novel framework for contextual symmetries in probabilistic graphical models. Contextual symmetries generalize and extend previous notions of orbital symmetry. Given any context, we can efficiently compute these symmetries by reducing it to the problem of colored graph isomorphism. While our framework is independent of any inference algorithm, we illustrate its applicability by proposing CON-MCMC, an MCMC approach that exploits contextual symmetries. Our experiments on several domains validate the efficacy of CON-MCMC, where it outperforms existing state-of-the-art techniques for symmetry-based MCMC by wide margins. Finally, we have released a reference implementation of CON-MCMC for wider use by the research community.

Acknowledgements

We are grateful to Mathias Niepert for sharing the code implementation of Orbital MCMC and for answering our queries on the code. We would also like to thank the anonymous reviewers for their comments and feedback. We thank Ritesh Noothigattu for discussions and comments on this research. Ankit Anand is supported by TCS Research Scholars Program. Mausam and Parag Singla are supported by Visvesvaraya faculty research awards by Govt. of India. Mausam is also supported by Google and Bloomberg research awards.

References

[Anand et al., 2016] Ankit Anand, Aditya Grover, Mausam, and Parag Singla. Contextual symmetries in probabilistic graphical models. In IJCAI, 2016.

[ Babai, 2015] László Babai. Graph isomorphism in quasipolynomial time. arXiv preprint arXiv:1512.03547, 2015.

[Boutilier et al., 1996] Craig Boutilier, Nir Friedman, Moises Goldszmidt, and Daphne Koller. Context-specific independence in Bayesian networks. In UAI, 1996.
[Bui et al., 2013] H. Bui, T. Huynh, and S. Riedel. Automorphism groups of graphical models and lifted variational inference. In *UAI*, 2013.

[Darga et al., 2008] Paul T Darga, Karem A Sakallah, and Igor L Markov. Faster symmetry discovery using sparsity of symmetries. In *Design Automation Conference*, 2008.

[de Salvo Braz et al., 2005] R. de Salvo Braz, E. Amir, and D. Roth. Lifted first-order probabilistic inference. In *IJCAI*, 2005.

[Domingos and Lindow, 2009] Pedro Domingos and Daniel Lindow. *Markov Logic: An Interface Layer for Artificial Intelligence*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2009.

[GAP, 2015] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.7.9*, 2015.

[Gent et al., 2005] Ian P Gent, Tom Kelsey, Steve A Linton, Iain McDonald, Ian Miguel, and Barbara M Smith. Conditional symmetry breaking. In *Principles and Practice of Constraint Programming*, 2005.

[Gent et al., 2007] Ian P Gent, Tom Kelsey, Stephen A Linton, Justin Pearson, and Colva M Roney-Dougal. Groupoids and conditional symmetry. In *Principles and Practice of Constraint Programming*, 2007.

[Gogate and Domingos, 2011] V. Gogate and P. Domingos. Probabilistic theorem proving. In *UAI*, 2011.

[Gogate et al., 2012] V. Gogate, A. Jha, and D. Venugopal. Advances in lifted importance sampling. In *AAAI*, 2012.

[Jha et al., 2010] Abhay Kumar Jha, Vibhav Gogate, Alexandra Meliou, and Dan Suciu. Lifted inference seen from the other side: The tractable features. In *NIPS*, 2010.

[Kersting et al., 2009] K. Kersting, B. Ahmadi, and S. Natarajan. Counting belief propagation. In *UAI*, 2009.

[Kimmig et al., 2015] A. Kimmig, L. Mihalkova, and L. Getoor. Lifted graphical models: A survey. *Machine Learning*, 99(1):1–45, 2015.

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

[Kopp et al., 2015] Timothy Kopp, Parag Singla, and Henry Kautz. Lifted symmetry detection and breaking for map inference. In *NIPS*, 2015.

[McKay and Piperno, 2014] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism. *Journal of Symbolic Computation*, 60(0):94 – 112, 2014.

[Mladenov et al., 2014] M. Mladenov, K. Kersting, and A. Globerson. Efficient lifting of MAP lp relaxations using k-locality. In *AISTATS*, 2014.

[Niepert and Van den Broeck, 2014] Mathias Niepert and Guy Van den Broeck. Tractability through exchangeability: A new perspective on efficient probabilistic inference. In *AAAI*, 2014.