Journal of Machine Learning Research 17 (2016) 1-29 Submitted 6/13; Revised 10/14; Published 3/16

Herded Gibbs Sampling

Yutian Chen YUTIAN.CHEN@UCI.EDU
7 Pancras Square, Kings Cross, London, N1C 4AG, United Kingdom

Luke Bornn BORNN@STAT.HARVARD.EDU
Statistics & Actuarial Science, Simon Fraser University, 8888 University Drive, Burnaby, BC, V5A1S6, Canada

Nando de Freitas NANDO@CS.OX.AC.UK
7 Pancras Square, Kings Cross, London, N1C 4AG, United Kingdom

Mareija Eskelin MAREIJA@CS.UBC.CA
Dept of Computer Science, University of British Columbia, 2366 Main Mall, Vancouver, BC, V6T1Z4, Canada

Jing Fang JINGF@CS.UBC.CA
1 Facebook Way, Menlo Park, CA, 94025, United States

Max Welling WELLING@ICS.UCI.EDU
Informatics Institute, Science Park 904, Postbus 94323, 1090 GH, Amsterdam, Netherlands

Editor: Aaron Courville, Rob Fergus, and Christopher Manning

Abstract

The Gibbs sampler is one of the most popular algorithms for inference in statistical models. In this paper, we introduce a herding variant of this algorithm, called herded Gibbs, that is entirely deterministic. We prove that herded Gibbs has an $O(1/T)$ convergence rate for models with independent variables and for fully connected probabilistic graphical models. Herded Gibbs is shown to outperform Gibbs in the tasks of image denoising with MRFs and named entity recognition with CRFs. However, the convergence for herded Gibbs for sparsely connected probabilistic graphical models is still an open problem.

Keywords: Gibbs sampling, herding, deterministic sampling

1. Introduction

Over the last 60 years, we have witnessed great progress in the design of randomized sampling algorithms; see for example Liu (2001); Doucet et al. (2001); Andrieu et al. (2003); Robert and Casella (2004) and the references therein. In contrast, the design of deterministic algorithms for "sampling" from distributions is still in its inception (Chen et al., 2010; Holroyd and Propp, 2010; Chen et al., 2011; Murray and Elliott, 2012). There are, however, many important reasons for pursuing this line of attack on the problem. From a theoretical perspective, this is a well defined mathematical challenge whose solution might have important consequences. It also brings us closer to reconciling the fact that we typically use pseudo-random number generators to run Monte Carlo algorithms on classical, Von Neumann architecture, computers. Moreover, the theory for some of the recently proposed deterministic sampling algorithms has taught us that they can achieve $O(1/T)$ convergence

rates (Chen et al., 2010; Holroyd and Propp, 2010), which are much faster than the standard Monte Carlo rates of $O(1/\sqrt{T})$ for computing ergodic averages. From a practical perspective, the design of deterministic sampling algorithms creates an opportunity for researchers to apply a great body of knowledge on optimization to the problem of sampling; see for example Bach et al. (2012) for an early example of this.

The domain of application of currently existing deterministic sampling algorithms is still very narrow. Importantly, the only available deterministic tool for sampling from unnormalized multivariate probability distributions is the Markov Chain Quasi-Monte Carlo method (Chen et al., 2011), but there is no theoretical result to show a better convergence rate than a standard MCMC method yet. This is very limiting because the problem of sampling from unnormalized distributions is at the heart of the field of Bayesian inference and the probabilistic programming approach to artificial intelligence (Lunn et al., 2000; Carbonetto et al., 2005; Milch and Russell, 2006; Goodman et al., 2008). At the same time, despite great progress in Monte Carlo simulation, the celebrated Gibbs sampler continues to be one of the most widely-used algorithms. For example, it is the inference engine behind popular statistics packages (Lunn et al., 2000), several tools for text analysis (Porteous et al., 2008), and Boltzmann machines (Ackley et al., 1985; Hinton and Salakhutdinov, 2006). The popularity of Gibbs stems from its generality and simplicity of implementation.

Without any doubt, it would be remarkable if we could design generic deterministic Gibbs samplers with fast (theoretical and empirical) rates of convergence. In this paper, we take steps toward achieving this goal by capitalizing on a recent idea for deterministic simulation known as herding. Herding (Welling, 2009a,b; Gelfand et al., 2010) is a deterministic procedure for generating samples $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$, such that the empirical moments $\boldsymbol{\mu}$ of the data are matched. The herding procedure, at iteration t, is as follows:

$$\begin{aligned} \mathbf{x}^{(t)} &= \arg\max_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{w}^{(t-1)}, \boldsymbol{\phi}(\mathbf{x}) \rangle, \\ \mathbf{w}^{(t)} &= \mathbf{w}^{(t-1)} + \boldsymbol{\mu} - \boldsymbol{\phi}(\mathbf{x}^{(t)}), \end{aligned} \tag{1}$$

where $\boldsymbol{\phi} : \mathcal{X} \to \mathcal{H}$ is a feature map (statistic) from $\mathcal{X}$ to a Hilbert space $\mathcal{H}$ with inner product $\langle \cdot, \cdot \rangle$, $\mathbf{w} \in \mathcal{H}$ is the vector of parameters, and $\boldsymbol{\mu} \in \mathcal{H}$ is the moment vector (expected value of $\boldsymbol{\phi}$ over the data) that we want to match. If we choose normalized features by making $\|\boldsymbol{\phi}(\mathbf{x})\|$ constant for all $\mathbf{x}$, then the update to generate samples $\mathbf{x}^{(t)}$ for $t = 1, 2, \ldots, T$ in Equation 1 is equivalent to minimizing the objective

$$J(\mathbf{x}_1, \ldots, \mathbf{x}_T) = \left\| \boldsymbol{\mu} - \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\phi}(\mathbf{x}^{(t)}) \right\|^2, \tag{2}$$

where T may have no prior known value and $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$ is the naturally defined norm based upon the inner product of the space $\mathcal{H}$ (Chen et al., 2010; Bach et al., 2012).

Herding can be used to produce samples from *normalized* probability distributions. This is done as follows. Let $\boldsymbol{\mu}$ denote a discrete, normalized probability distribution, with $\mu_i \in [0, 1]$ and $\sum_{i=1}^{n} \mu_i = 1$. A natural feature in this case is the vector $\boldsymbol{\phi}(x)$ that has all entries equal to zero, except for the entry at the position indicated by x. For instance, if $x = 2$ and $n = 5$, we have $\boldsymbol{\phi}(x) = (0, 1, 0, 0, 0)^T$. Hence, $\widehat{\boldsymbol{\mu}} = T^{-1} \sum_{t=1}^{T} \boldsymbol{\phi}(x^{(t)})$ is an empirical estimate of the distribution. In this case, one step of the herding algorithm

involves finding the largest component of the weight vector ($i^\star = \arg\max_{i\in\{1,2,\ldots,n\}} \mathbf{w}_i^{(t-1)}$), setting $x^{(t)} = i^\star$, fixing the $i^\star$-entry of $\boldsymbol{\phi}(x^{(t)})$ to one and all other entries to zero, and updating the weight vector: $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} + (\boldsymbol{\mu} - \boldsymbol{\phi}(x^{(t)}))$. The output is a set of samples $\{x^{(1)}, \ldots, x^{(T)}\}$ for which the empirical estimate $\widehat{\boldsymbol{\mu}}$ converges on the target distribution $\boldsymbol{\mu}$ as $O(1/T)$.

The herding method, as described thus far, only applies to normalized distributions or to problems where the objective is not to guarantee that the samples come from the right target, but to ensure that some moments are matched. An interpretation of herding in terms of Bayesian quadrature has been put forward recently by Huszar and Duvenaud (2012).

In this paper, we will show that it is possible to use herding to generate samples from more complex *unnormalized* probability distributions. In particular, we introduce a deterministic variant of the popular Gibbs sampling algorithm, which we refer to as *herded Gibbs*. While Gibbs relies on drawing samples from the *full-conditionals* at random, herded Gibbs generates the samples by matching the full-conditionals. That is, one simply applies herding to all the full-conditional distributions.

The experiments will demonstrate that the new algorithm outperforms Gibbs sampling and mean field methods in the domain of sampling from sparsely connected probabilistic graphical models, such as grid-lattice Markov random fields (MRFs) for image denoising and conditional random fields (CRFs) for natural language processing.

We advance the theory by proving that the deterministic Gibbs algorithm converges for distributions of independent variables and fully-connected probabilistic graphical models. However, a proof establishing suitable conditions that ensure convergence of herded Gibbs sampling for sparsely connected probabilistic graphical models is still unavailable.

2. Herded Gibbs Sampling

For a graph of discrete nodes $\mathcal{G} = (V, E)$, where the set of nodes are the random variables $V = \{X_i\}_{i=1}^N$, $X_i \in \mathcal{X}$, let π denote the *target distribution* defined on $\mathcal{G}$.

Gibbs sampling is one of the most popular methods to draw samples from π. Gibbs alternates (either systematically or randomly) the sampling of each variable X_i given $\mathbf{X}_{\mathcal{N}(i)} = \mathbf{x}_{\mathcal{N}(i)}$, where i is the index of the node, and $\mathcal{N}(i)$ denotes the neighbors of node i. That is, Gibbs generates each sample from its full-conditional distribution $p(X_i|\mathbf{x}_{\mathcal{N}(i)})$.

Herded Gibbs replaces the sampling from full-conditionals with herding at the level of the full-conditionals. That is, it alternates a process of matching the full-conditional distributions $p(X_i = x_i|\mathbf{X}_{\mathcal{N}(i)})$. To do this, herded Gibbs defines a set of auxiliary weights $\{w_{i,\mathbf{x}_{\mathcal{N}(i)}}\}$ for any value of $X_i = x_i$ and $\mathbf{X}_{\mathcal{N}(i)} = \mathbf{x}_{\mathcal{N}(i)}$. For ease of presentation, we assume the domain of X_i is binary, $\mathcal{X} = \{0, 1\}$, and we use one weight for every i and assignment to the neighbors $\mathbf{x}_{\mathcal{N}(i)}$. Herded Gibbs can be trivially generalized to the discrete setting by employing weight vectors in $\mathbb{R}^{|\mathcal{X}|}$ instead of scalars.

If the binary variable X_i has four binary neighbors $\mathbf{X}_{\mathcal{N}(i)}$, we must maintain $2^4 = 16$ weight vectors. Only the weight vector corresponding to the current instantiation of the neighbors is updated, as illustrated in Algorithm 1[1]. The memory complexity of herded

1. Code is available at `http://www.mareija.ca/research/code/`

Algorithm 1 Herded Gibbs Sampling

Input: T.

Step 1: Set $t = 0$. Initialize $\mathbf{X}^{(0)}$ in the support of π and $w^{(0)}_{i,\mathbf{x}_{\mathcal{N}(i)}}$ in $(\pi(X_i = 1|\mathbf{x}_{\mathcal{N}(i)}) - 1, \pi(X_i = 1|\mathbf{x}_{\mathcal{N}(i)}))$.

for $t = 1 \to T$ **do**

Step 2: Pick a node i according to some policy. Denote $w = w^{(t-1)}_{i,\mathbf{x}^{(t-1)}_{\mathcal{N}(i)}}$.

Step 3: If $w > 0$, set $x_i^{(t)} = 1$, otherwise set $x_i^{(t)} = 0$.

Step 4: Update weight $w^{(t)}_{i,\mathbf{x}^{(t)}_{\mathcal{N}(i)}} = w^{(t-1)}_{i,\mathbf{x}^{(t-1)}_{\mathcal{N}(i)}} + \pi(X_i = 1|\mathbf{x}^{(t-1)}_{\mathcal{N}(i)}) - x_i^{(t)}$.

Step 5: Keep the values of all the other nodes $x_j^{(t)} = x_j^{(t-1)}, \forall j \neq i$ and all the other weights $w^{(t)}_{j,\mathbf{x}_{\mathcal{N}(j)}} = w^{(t-1)}_{j,\mathbf{x}_{\mathcal{N}(j)}}, \forall j \neq i$ or $\mathbf{x}_{\mathcal{N}(j)} \neq \mathbf{x}^{(t-1)}_{\mathcal{N}(i)}$.

end for

Output: $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$

Gibbs is exponential in the maximum node degree. Note the algorithm is a deterministic Markov process with state $(\mathbf{X}, \mathbf{W})$.

The initialization in the first step of Algorithm 1 guarantees that $\mathbf{X}^{(t)}$ always remains in the support of π with the reason to be explained in Section 3.1. For a deterministic scan policy in step 2, we take the value of variables $\mathbf{x}^{(tN)}, t \in \mathbb{N}$ as a sample sequence. Throughout the paper all experiments employ a fixed variable traversal for sample generation. We call one such traversal of the variables a *sweep*.

3. Analysis

As herded Gibbs sampling is a deterministic algorithm, the probability distribution of the sample at any step t is a single point mass and there is no stationary distribution of states. Instead, we examine the average of the sample states over time and hypothesize that it converges to the joint distribution, our target distribution, π. To make the treatment precise, we need the following definition:

Definition 1 *For a graph of discrete nodes $\mathcal{G} = (V, E)$, where the set of nodes $V = \{X_i\}_{i=1}^N$, $X_i \in \mathcal{X}$, $P_T^{(\tau)}$ is the empirical estimate of the joint distribution obtained by averaging over T samples acquired from $\mathcal{G}$. $P_T^{(\tau)}$ is derived from T samples, collected at the end of every sweep over N variables, starting from the τ^{th} sweep:*

$$P_T^{(\tau)}(\mathbf{X} = \mathbf{x}) = \frac{1}{T} \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}^{(kN)} = \mathbf{x}). \tag{3}$$

The definition of $P_T^{(\tau)}$ is illustrated in Figure 1. Our goal is to prove that the limiting average sample distribution over time converges to the target distribution π. Specifically, we want to show the following:

$$\lim_{T\to\infty} P_T^{(\tau)}(\mathbf{x}) = \pi(\mathbf{x}), \forall \tau \geq 0. \tag{4}$$

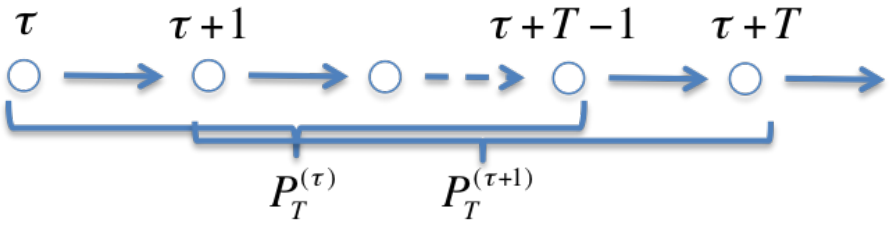


Figure 1: Sample distribution over T sweeps. Each node refers to the joint state at the end of one sweep.

If this holds, we also want to know what the convergence rate is.

3.1 Single Variable Models

We begin the theoretical analysis with a graph of one binary variable. For this graph, there is only one weight w and herded Gibbs is equivalent to the standard herding algorithm. Denote $\pi(X=1)$ as π for notational simplicity. The sequence of X is determined by the dynamics of w (shown in Figure 2a):

$$w^{(t)} = w^{(t-1)} + \pi - \mathbb{I}(w^{(t-1)} > 0), \quad X^{(t)} = \begin{cases} 1 & \text{if } w^{(t-1)} > 0 \\ 0 & \text{otherwise} \end{cases}. \tag{5}$$

The following lemma shows that $(\pi - 1, \pi]$ is the invariant interval of the dynamics.

Lemma 1 *If w is the weight of the herding dynamics of a single binary variable X with probability $P(X=1) = \pi$, and $w^{(s)} \in (\pi - 1, \pi]$ at some step $s \geq 0$, then $w^{(t)} \in (\pi - 1, \pi], \forall t \geq s$. Moreover, for $T \in \mathbb{N}$, we have:*

$$\sum_{t=s+1}^{s+T} \mathbb{I}[X^{(t)} = 1] \in [T\pi - 1, T\pi + 1], \tag{6}$$

$$\sum_{t=s+1}^{s+T} \mathbb{I}[X^{(t)} = 0] \in [T(1-\pi) - 1, T(1-\pi) + 1]. \tag{7}$$

See the proof in Appendix A. It follows immediately that the state $X = 1$ is visited at a frequency close to π with an error:

$$|P_T^{(\tau)}(X=1) - \pi| \leq \frac{1}{T}. \tag{8}$$

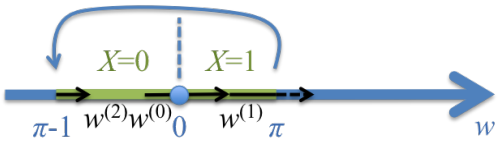
This is known as the fast moment matching property in Welling (2009a,b); Gelfand et al. (2010). When w is outside the invariant interval, it is easy to observe that w will move into it monotonically at a linear speed in a transient period. So we will always consider an initialization of $w \in (\pi - 1, \pi]$ from now on.

Another immediate consequence of Lemma 1 is that the initialization in Algorithm 1 ensures that $\mathbf{X}^{(t)}$ always remain in the support of π. That is because when we consider

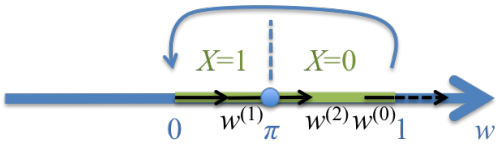
the set of iterations that involves a particular weight $w_{i,\mathbf{x}_{\mathcal{N}(i)}}$, the dynamics of that weight is equivalent to that of a single variable model with a probability $\pi(X_i = 1|\mathbf{x}_{\mathcal{N}(i)})$. If a joint state $\mathbf{X}$ is going to move outside the support at some iteration t, from e. g. $\mathbf{x}^{(t-1)} = (x_i = 0, \mathbf{x}_{-i})$ to $\mathbf{x}^{(t)} = (x_i = 1, \mathbf{x}_{-i})$ where $\mathbf{x}_{-i}$ denotes all the other variables but x_i, then the corresponding weight $w^{(t-1)}_{i,\mathbf{x}_{\mathcal{N}(i)}}$ must be positive according to the algorithm. However, the conditional probability $\pi(X_i = 1|\mathbf{x}_{\mathcal{N}(i)}) = 0$ because $\pi(\mathbf{x}^{(t-1)}) > 0$ and $\pi(\mathbf{x}^{(t)}) = 0$. Following Lemma 1 the weight $w^{(t-1)}_{i,\mathbf{x}_{\mathcal{N}(i)}} \in (-1, 0]$, leading to a contradiction. The same argument applies when $\mathbf{X}$ tries to move from $\mathbf{x}^{(t-1)} = (x_i = 1, \mathbf{x}_{-i})$ to $\mathbf{x}^{(t)} = (x_i = 0, \mathbf{x}_{-i})$. Therefore, once initialized inside the support, the samples of Algorithm 1 will remain in the support for any $t > 0$.

It will be useful for the next section to introduce an equivalent representation of the weight dynamics by taking a one-to-one mapping $w \leftarrow w \mod 1$ (we define $1 \mod 1 = 1$) with a little abuse of the symbol w. We think of the new variable w as updated by a constant translation vector in a circular unit interval $(0, 1]$ as shown in Figure 2b. That is,

$$w^{(t)} = (w^{(t-1)} + \pi) \mod 1, \quad X^{(t)} = \begin{cases} 1 & \text{if } w^{(t-1)} < \pi \\ 0 & \text{otherwise} \end{cases}. \tag{9}$$



(a) Representation in Equation 5.



(b) Equivalent representation.

Figure 2: Herding dynamics for a single variable. Black arrows show the trajectory of $w^{(t)}$ for 2 iterations.

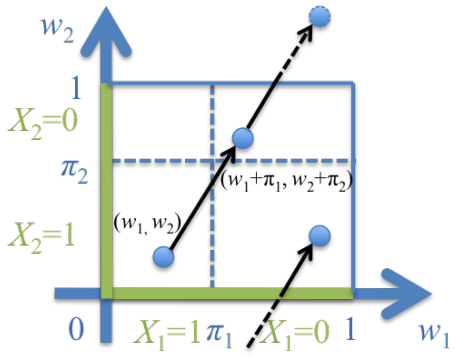


Figure 3: Herding dynamics for two independent variables in the equivalent representation.

3.2 Empty Graphs

The analysis of herded Gibbs for empty graphs is a natural extension of that for single variables. In an empty graph, all the variables are independent of each other and herded Gibbs reduces to running N one-variable chains in parallel. Denote the marginal distribution $\pi_i := \pi(X_i = 1)$.

Examples of failing convergence in the presence of rational ratios between the π_is were observed in Bach et al. (2012). There the need for further theoretical research on this matter

was pointed out. The following theorem provides a sufficient condition for convergence in the restricted domain of empty graphs.

Theorem 2 *For an empty graph, when herded Gibbs has a fixed scanning order, and* $\{1, \pi_1, \ldots, \pi_N\}$ *are rationally independent, the empirical distribution* $P_T^{(\tau)}$ *converges to the target distribution* π *as* $T \to \infty$ *for any* $\tau \geq 0$.

A set of n real numbers, $x_1, x_2, \ldots, x_n$, is said to be rationally independent if for any set of rational numbers, $a_1, a_2, \ldots, a_n$, we have $\sum_{i=1}^{n} a_i x_i = 0 \Leftrightarrow a_i = 0, \forall 1 \leq i \leq n$.

Proof For an empty graph of N independent vertices, the dynamics of the weight vector $\mathbf{w}$ after one sweep over all variables are equivalent to a constant translation mapping in an N-dimensional circular unit space $(0, 1]^N$, as shown in Figure 3:

$$\begin{aligned} \mathbf{w}^{(t)} &= (\mathbf{w}^{(t-1)} + \boldsymbol{\pi}) \mod 1 \\ &= (\mathbf{w}^{(0)} + t\boldsymbol{\pi}) \mod 1, \quad x_i^{(t)} = \begin{cases} 1 & \text{if } w_i^{(t-1)} < \pi_i \\ 0 & \text{otherwise} \end{cases}, \forall 1 \leq i \leq N. \end{aligned} \tag{10}$$

The Kronecker-Weyl theorem (Weyl, 1916) states that the sequence $\tilde{\mathbf{w}}^{(t)} = t\pi \mod 1, t \in \mathbb{Z}^+$ is equidistributed (or uniformly distributed) on $(0, 1]^N$ if and only if $(1, \pi_1, \ldots, \pi_N)$ is rationally independent. Intuitively, when $(1, \pi_1, \ldots, \pi_N)$ is rationally independent, the trajectory of $\tilde{\mathbf{w}}^{(t)}$ can not form a closed loop in the circular unit space and will thereby cover the entire space uniformly.

Since we can define a one-to-one volume preserving transformation between $\tilde{\mathbf{w}}^{(t)}$ and $\mathbf{w}^{(t)}$ as $(\tilde{\mathbf{w}}^{(t)} + \mathbf{w}^{(0)}) \mod 1 = \mathbf{w}^{(t)}$, the sequence of weights $\{\mathbf{w}^{(t)}\}$ is also uniformly distributed in $(0, 1]^N$.

Now define the mapping from a state value x_i to an interval of w_i as

$$A_i(x) = \begin{cases} (0, \pi_i] & \text{if } x = 1 \\ (\pi_i, 1] & \text{if } x = 0 \end{cases} \tag{11}$$

and let $|A_i|$ be its measure. We obtain the limiting distribution of the joint state as

$$\begin{aligned} \lim_{T\to\infty} P_T^{(\tau)}(\mathbf{X} = \mathbf{x}) &= \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{I}\left[\mathbf{w}^{(t-1)} \in \prod_{i=1}^{N} A_i(x_i)\right] \\ &= \prod_{i=1}^{N} |A_i(x_i)| = \prod_{i=1}^{N} \pi(X_i = x_i) = \pi(\mathbf{X} = \mathbf{x}). \end{aligned} \tag{12}$$

■

The rational independence condition in the application of the Kronecker-Weyl theorem ensures that the probabilistic independence between different variables will be respected by the herding dynamics. When the condition fails, the convergence of joint distribution is not guaranteed but the marginal distribution of each variables will still converge to their target distribution because the N one-variable chains run independently from each other.

3.3 Fully-Connected Graphs

When herded Gibbs is applied to fully-connected (complete) graphs, convergence is guaranteed even with rationally dependent conditional probabilities. In fact, herded Gibbs converges to the target joint distribution at a rate of $O(1/T)$ with a $O(\log(T))$ burn-in period. This statement is formalized in Theorem 3 and a corollary when we ignore the burn-in period, with proofs provided respectively in Appendix B.4 and B.5.

Theorem 3 *For a fully-connected graph, when herded Gibbs has a fixed scanning order and a Dobrushin coefficient of the corresponding Gibbs sampler $\eta < 1$, there exist constants $l > 0$, and $B > 0$ such that*

$$d_v(P_T^{(\tau)} - \pi) \leq \frac{\lambda}{T}, \forall T \geq T^*, \tau > \tau^*(T), \tag{13}$$

where $\lambda = \frac{2N(1+\eta)}{l(1-\eta)}$, $T^* = \frac{2B}{l}$, $\tau^*(T) = \log_{\frac{2}{1+\eta}}\left(\frac{(1-\eta)lT}{4N}\right)$, and $d_v(\delta\pi) := \frac{1}{2}||\delta\pi||_1$.

Corollary 4 *When the conditions of Theorem 3 hold, and we start collecting samples at the end of every sweep from the beginning, that is setting $\tau = 0$, the error of the sample distribution is bounded by:*

$$d_v(P_T^{(\tau=0)} - \pi) \leq \frac{\lambda + \tau^*(T)}{T} = O(\frac{\log(T)}{T}), \quad \forall T \geq T^* + \tau^*(T^*). \tag{14}$$

The Dobrushin ergodic coefficient (Brémaud, 1999) measures the geometric convergence rate of a Markov chain. The constant l in the convergence rate can be interpreted as a lower bound of the transition probability between any pair of states in the support of the target distribution. For a strictly positive distribution, the constants l and B are

$$l = \pi_{\min}^N, \quad B = \pi_{\min}^N + \frac{1 - (2\pi_{\min})^N}{1 - 2\pi_{\min}}. \tag{15}$$

where $\pi_{\min}$ is the minimal conditional probability $\pi_{\min} = \min_{1 \leq i \leq N, \mathbf{x}_{-i}} \pi(x_i|\mathbf{x}_{-i})$. We refer the readers to Equation 34 in Proposition 5 in the appendix for a general distribution. Notice that l has an exponential term, with N in the exponent, leading to an exponentially large constant. This is unavoidable for any sampling algorithm when considering the convergence to a joint distribution with 2^N states. As for the marginal distributions, it is obvious that the convergence rate of herded Gibbs is also $O(1/T)$ because marginal probabilities are linear functions of the joint distribution. In fact, we observe very rapid convergence results for the marginals in practice, so stronger theoretical results about the convergence of the marginal distributions seem plausible.

The proof proceeds by first bounding the discrepancy between the chain of empirical estimates of the joint obtained by averaging over T herded Gibbs samples, $\{P_T^{(s)}\}, s \geq \tau$, and a Gibbs chain initialized at $P_T^{(\tau)}$. After one sweep over N variables, this discrepancy is bounded above by $2N/lT$.

The Gibbs chain has geometric convergence to π and the distance between the Gibbs and herded Gibbs chains decreases as $O(1/T)$. When the distance between $P_T^{(\tau)}$ and π is

sufficiently large, the geometric convergence rate to π dominates the discrepancy of herded Gibbs and thus we infer that $P_T^{(\tau)}$ converges to a neighborhood of π geometrically in time τ for a fixed T. To round-off the proof, we must find a limiting value for τ. The proof concludes with an $O(\log(T))$ burn-in for τ.

When there exist conditional independencies in a distribution, we can still apply herded Gibbs with a fully connected graph and treat all the other variables as the Markov blanket of the variable to be sampled. Theorem 3 and its corollary still apply. Alternatively, we can apply herded Gibbs on a more compact representation with an incomplete graph. It requires less memory to run herded Gibbs because the number of weights depends exponentially on the neighborhood size. However, for a generic graph we have no mathematical guarantees for the convergence rate of herded Gibbs. In fact, one can easily construct synthetic examples for which herded Gibbs does not seem to converge to the true marginals and joint distribution. For the examples covered by our theorems and for examples with real data, herded Gibbs demonstrates good behaviour. The exact conditions under which herded Gibbs converges for sparsely connected graphs are still unknown.

4. Experiments

We illustrate the performance of herded Gibbs with two synthetic examples and two real experiments for image denoising and natural language processing respectively.

4.1 Simple Complete Graph

We begin with an illustration of how herded Gibbs substantially outperforms Gibbs and a deterministic Gibbs sampler based on MCQMC on a simple complete graph. The MCQMC algorithm replaces the random number generator of the regular Gibbs sampler with a completely uniformly distributed (CUD) sequence (Chen et al., 2011). We consider a fully-connected model of two variables, X_1 and X_2, as shown in Figure 4; the joint distribution of these variables is shown in Table 1. We run each sampler for 2.6×10^5 iterations. We use a small linear feedback shift registers (LFSR) described in (Chen et al., 2012) to generate the CUD sequence and choose the size of the LFSR so that the entire period of the sequence will be used for one run of the Markov chain. Both Gibbs and MCQMC-based Gibbs are run 100 times with different random seeds to assess their average performance. Herded Gibbs is run only once because different initialization does not show noticeable difference in its performance. Figure 5 shows the marginal distribution $P(X_1 = 1)$ and the joint distribution approximated by all the algorithms for different ϵ. As ϵ decreases, all the approaches require more iterations to converge, but herded Gibbs clearly outperforms the other two algorithms. Figure 5c also shows that herding does indeed exhibit a linear convergence rate. MCQMC-based Gibbs does not show any improvement on the error of the sample distribution compared to the standard Gibbs.

4.2 Simple Incomplete Graph

We also illustrate a simple counterexample where the sample distribution of herded Gibbs does not converge to the target distribution when the graph is incomplete. Figure 7 shows a four-variable graphical model with two missing edges, $X_1 - X_4$, $X_2 - X_3$. The unary

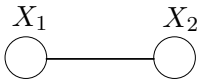


Figure 4: Two-variable model.

	$\mathbf{X_1 = 0}$	$\mathbf{X_1 = 1}$	$\mathbf{P(X_2)}$
$\mathbf{X_2 = 0}$	$1/4 - \epsilon$	ϵ	$1/4$
$\mathbf{X_2 = 1}$	ϵ	$3/4 - \epsilon$	$3/4$
$\mathbf{P(X_1)}$	$1/4$	$3/4$	1

Table 1: Joint distribution of the two-variable model.

and pairwise energies of existing edges are random sampled from a standard Gaussian distribution $\mathcal{N}(0, 1)$. We apply herded Gibbs for 10^8 iterations and compare the joint sample distribution with the true distribution. While the discrepancy is marginal as depicted in Figure 6b, the L_1 error plot in 6b does not show a tendency to converging to zero.

4.3 MRF for Image Denoising

Next, we consider the standard setting of a grid-lattice MRF for image denoising. Let us assume that we have a binary image corrupted by noise, and that we want to infer the original clean image. Let $X_i \in \{-1, +1\}$ denote the unknown true value of pixel i, and y_i the observed, noise-corrupted value of this pixel. We take advantage of the fact that neighboring pixels are likely to have the same label by defining an MRF with an Ising prior. That is, we specify a rectangular 2D lattice with the following pair-wise clique potentials:

$$\psi_{ij}(x_i, x_j) = \begin{pmatrix} e^{J_{ij}} & e^{-J_{ij}} \\ e^{-J_{ij}} & e^{J_{ij}} \end{pmatrix} \tag{16}$$

and joint distribution:

$$p(\mathbf{x}|\mathbf{J}) = \frac{1}{Z(\mathbf{J})} \prod_{i \sim j} \psi_{ij}(x_i, x_j) = \frac{1}{Z(\mathbf{J})} \exp\left(\frac{1}{2} \sum_{i \sim j} J_{ij} x_i x_j\right), \tag{17}$$

where $i \sim j$ is used to indicate that nodes i and j are connected. The known parameters J_{ij} establish the coupling strength between nodes i and j. Note that the matrix $\mathbf{J}$ is symmetric. If all the $J_{ij} > 0$, then neighboring pixels are likely to be in the same state.

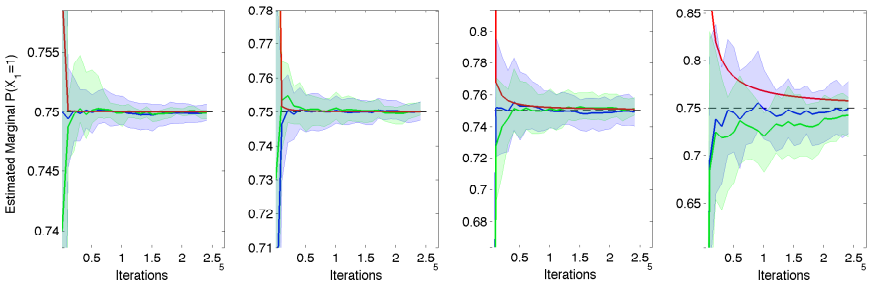
The MRF model combines the Ising prior with a likelihood model as follows:

$$p(\mathbf{x}, \mathbf{y}) \quad = \quad p(\mathbf{x})p(\mathbf{y}|\mathbf{x}) = \left[\frac{1}{Z} \prod_{i \sim j} \psi_{ij}(x_i, x_j)\right] \cdot \left[\prod_i p(y_i|x_i)\right]. \tag{18}$$

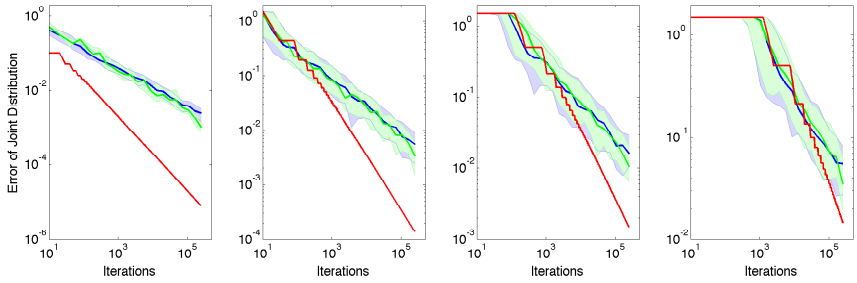
The potentials ψ_{ij} encourage label smoothness. The likelihood terms $p(y_i|x_i)$ are conditionally independent (e.g. Gaussians with known variance σ^2 and mean $\boldsymbol{\mu}$ centered at each value of x_i, denoted μ_{x_i}). In more precise terms,

$$p(\mathbf{x}, \mathbf{y}|\mathbf{J}, \boldsymbol{\mu}, \sigma) = \frac{1}{Z(\mathbf{J}, \boldsymbol{\mu}, \sigma)} \exp\left(\frac{1}{2} \sum_{i \sim j} J_{ij} x_i x_j - \frac{1}{2\sigma^2} \sum_i (y_i - \mu_{x_i})^2\right). \tag{19}$$

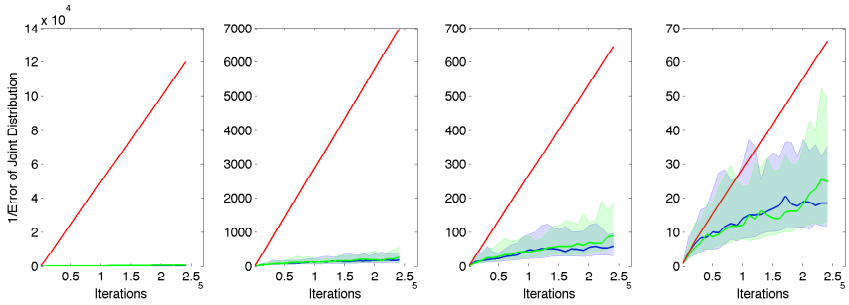
When the coupling parameters J_{ij} are identical, say $J_{ij} = J$, we have $\sum_{ij} J_{ij} f(x_i, x_j) = J \sum_{ij} f(x_i, x_j)$. Hence, different neighbor configurations result in the same value of $J \sum_{ij} f(x_i, x_j)$.



(a) Approximate marginals.

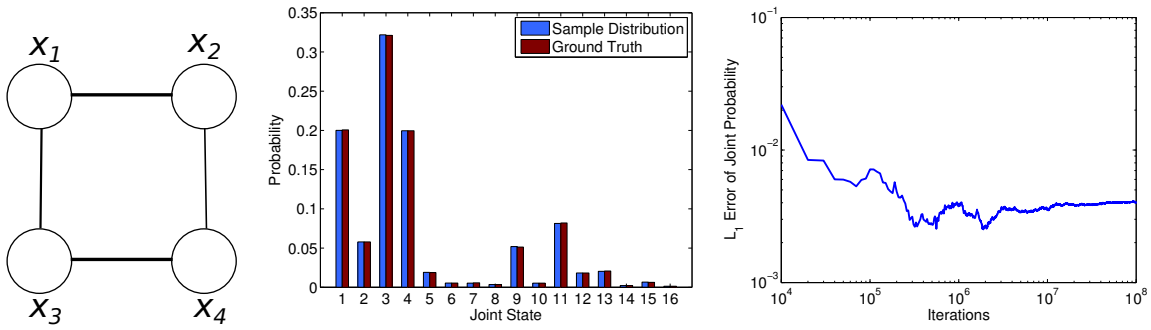


(b) Log-log plot of the L_1 error of the joint distribution.



(c) Inverse of the L_1 error of the joint distribution.

Figure 5: Approximating a marginal (a) and joint (b, c) distribution with Gibbs (blue), MCQMC-based Gibbs (green) and herded Gibbs (red) for an MRF of two variables, constructed so as to make the move from state $(0, 0)$ to $(1, 1)$ progressively more difficult as ϵ decreases. The four columns, from left to right, are for $\epsilon = 0.1$, $\epsilon = 0.01$, $\epsilon = 0.001$ and $\epsilon = 0.0001$. Table 1 provides the joint distribution for these variables. The shaded areas for Gibbs and MCQMC-based Gibbs correspond to 25%-75% quantile of 100 runs. Rows (b) and (c) illustrate that the empirical convergence rate of herded Gibbs matches the expected theoretical rate. As the error of herded Gibbs in (b) and (c) frequently drops to extremely small values for some iterations and jumps back, we plot the upper-bound (envelope) of the error for herded Gibbs defined as $\tilde{e}_t = \max_{\tau \geq t} e_\tau$ to remove the oscillating behavior for a better visualization.



(a) (b) (c)

Figure 7: Four-variable model represented as an incomplete graph. (a): Graphical Model. (b): Joint distribution of samples from herded Gibbs vs. the ground truth. (c): Log-log plot of the L_1 error of the joint sample distribution.

If we store the conditionals for configurations with the same sum together, we only need to store as many conditionals as different possible values that the sum could take. This enables us to develop a shared version of herded Gibbs that is more memory efficient where we only maintain and update weights for *distinct states* of the Markov blanket of each variable. The shared version of herded Gibbs also exhibits a different dynamics as the standard version as shown in the following result, and the convergence property of this algorithm remains an open problem.

In this exemplary image denoising experiment, noisy versions of the binary image, depicted in Figure 8 (left), were created through the addition of Gaussian noise, with varying σ. Figure 8 (right) shows a corrupted image with $\sigma = 4$. The L_2 reconstruction errors as a function of the number of iterations, for this example, are shown in Figure 9. The plot compares the herded Gibbs method against Gibbs and two versions of mean field with different damping factors (Murphy, 2012). The results demonstrate that the herded Gibbs techniques are among the best methods for solving this task.

A comparison for different values σ is presented in Table 2. As expected mean field does well in the low-noise scenario, but the performance of the shared version of herded Gibbs as the noise increases is significantly better.

4.4 CRF for Named Entity Recognition

Named entity recognition (NER) involves the identification of entities, such as people and locations, within a text sample. A conditional random field (CRF) for NER models the relationship between entity labels and sentences with a conditional probability distribution: $P(X|Y,\theta)$, where X is a labeling, Y is a sentence, and θ is a vector of coupling parameters. The parameters, θ, are feature weights and model relationships between variables X_i and Y_j or X_i and X_j. A chain CRF only employs relationships between adjacent variables, whereas a skip-chain CRF can employ relationships between variables where subscripts i and j differ dramatically. Skip-chain CRFs are important in language tasks, such as NER

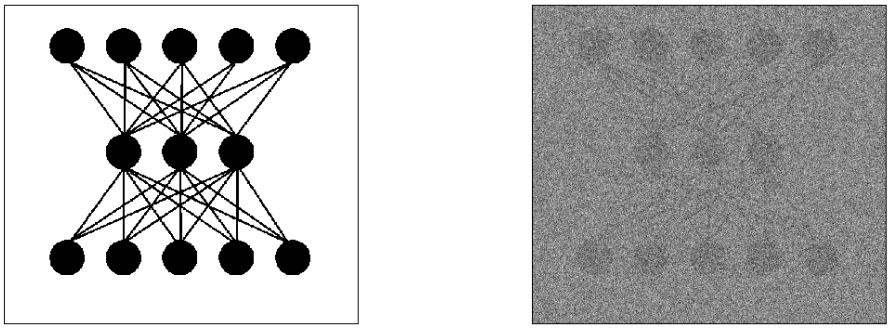


Figure 8: Original image (left) and its corrupted version (right), with noise parameter $\sigma = 4$.

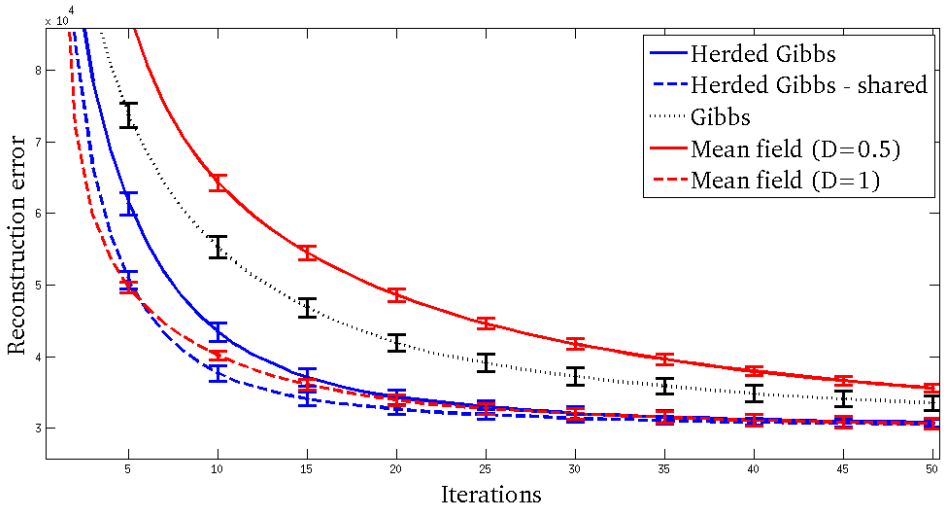


Figure 9: Reconstruction errors for the image denoising task. The results are averaged across 10 corrupted images with Gaussian noise $\mathcal{N}(0, 16)$. The error bars correspond to one standard deviation. Mean field requires the specification of the damping factor D.

and semantic role labeling, because they allow us to model long dependencies in a stream of words, see Figure 10.

Once the parameters have been learned, the CRF can be used for inference; a labeling for some sentence Y is found by maximizing the above probability. Inference for CRF models in the NER domain is typically carried out with the Viterbi algorithm. However, if we want to accommodate long term dependencies, thus resulting in the so called skip-chain CRFs,

Method \ σ	2	4	6	8
Herded Gibbs	21.58(0.26)	32.07(0.98)	47.52(1.64)	67.93(2.78)
Herded Gibbs - shared	22.24(0.29)	**31.40**(0.59)	**42.62**(1.98)	**58.49**(2.86)
Gibbs	21.63(0.28)	37.20(1.23)	63.78(2.41)	90.27(3.48)
Mean field (D=0.5)	**15.52**(0.30)	41.76(0.71)	76.24(1.65)	104.08(1.93)
Mean field (D=1)	17.67(0.40)	32.04(0.76)	51.19(1.44)	74.74(2.21)

Table 2: Errors of image denoising example after 30 iterations (all measurements have been scaled by $\times 10^{-3}$). We use an Ising prior with $J_{ij} = 1$ and four Gaussian noise models with different σ's. For each σ, we generated 10 corrupted images by adding Gaussian noise. The final results shown here are averages and standard deviations (in parentheses) across the 10 corrupted images. D denotes the damping factor in mean field.

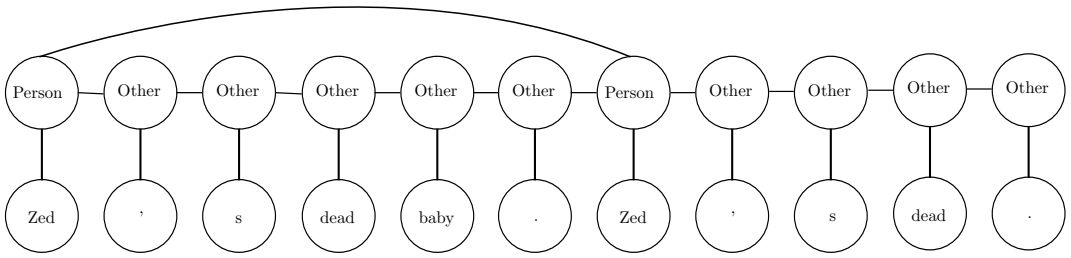


Figure 10: Typical skip-chain CRF model for named entity recognition.

Viterbi becomes prohibitively expensive. To surmount this problem, the Stanford named entity recognizer (J. R. Finkel and Manning, 2005) makes use of annealed Gibbs sampling.

To demonstrate herded Gibbs on a practical application of great interest in text mining, we modify the standard inference procedure of the Stanford named entity recognizer by replacing the annealed Gibbs sampler with the herded Gibbs sampler. The herded Gibbs sampler is not annealed. Notice that the label of a word X_i is a discrete variable with possibly multiple values. As discussed in Section 2 we generalize herded Gibbs for binary variables to discrete variables by assigning a different weight $w_{i,\mathbf{x}_{\mathcal{N}(i)}}$ for each value of X_i. To find the maximum a posteriori sequence X, we compute the joint discrete probability of every sample and choose the one with the highest probability as the prediction. The faster a sampler mixes in the state space, the more likely that a sample with high probability will be generated given a the same amount of time. In order to be able to compare against Viterbi, we have purposely chosen to use single-chain CRFs. We remind the reader, however, that the herded Gibbs algorithm could be used in cases where Viterbi inference is not possible.

We used the pre-trained 3-class CRF model in the Stanford NER package (J. R. Finkel and Manning, 2005). This model is a linear chain CRF with pre-defined features and pre-trained feature weights, θ. For the test set, we used the corpus for the NIST 1999 IE-ER Evaluation. Performance is measured in per-entity F_1 $\left(F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}\right)$. For all the

"Pumpkin" (*Tim Roth*) and "Honey Bunny" (*Amanda Plummer*) are having breakfast in a diner. They decide to rob it after realizing they could make money off the customers as well as the business, as they did during their previous heist. Moments after they initiate the hold-up, the scene breaks off and the title credits roll. As *Jules Winnfield* (*Samuel L. Jackson*) drives, *Vincent Vega* (*John Travolta*) talks about his experiences in **Europe**, from where he has just returned: the hash bars in **Amsterdam**, the French McDonald's and its "Royale with Cheese".

Figure 11: Results for the application of the NER CRF to a random Wikipedia sample (Wik, 2013). Entities are automatically classified as person (red italic), location (green boldfaced) and organization (orange underlined).

methods, except Viterbi, we show F_1 scores after 100, 400 and 800 iterations in Table 3. For Gibbs, the results shown are the averages and standard deviations over 5 random runs. We used a linear annealing schedule for Gibbs. As the results illustrate, herded Gibbs attains the same accuracy as Viterbi and it is faster than annealed Gibbs. Unlike Viterbi, herded Gibbs can be easily applied to skip-chain CRFs. After only 400 iterations (90.5 seconds), herded Gibbs already achieves an F_1 score of 84.75, while Gibbs, even after 800 iterations (115.9 seconds) only achieves an F_1 score of 84.61. The experiment thus clearly demonstrates that (i) herded Gibbs does no worse than the optimal solution, Viterbi, and (ii) herded Gibbs yields more accurate results for the same amount of computation than Gibbs sampling. Figure 11 provides a representative NER example of the performance of Gibbs, herded Gibbs and Viterbi (all methods produced the same annotation for this short example).

Method \ Iterations	100	400	800
Annealed Gibbs	84.36(0.16) [55.73s]	84.51(0.10) [83.49s]	84.61(0.05) [115.92s]
Herded Gibbs	84.70 [59.08s]	84.75 [90.48s]	84.81 [132.00s]
Viterbi			84.81[46.74s]

Table 3: F_1 scores for Gibbs, herded Gibbs and Viterbi on the NER task. The average computational time each approach took to do inference for the entire test set is listed (in square brackets). After only 400 iterations (90.48 seconds), herded Gibbs already achieves an F_1 score of 84.75, while Gibbs, even after 800 iterations (115.92 seconds) only achieves an F_1 score of 84.61. For the same computation, herded Gibbs is more accurate than Gibbs.

5. Conclusions and Future Work

In this paper, we introduced herded Gibbs, a deterministic variant of the popular Gibbs sampling algorithm. While Gibbs relies on drawing samples from the full-conditionals at random, herded Gibbs generates the samples by matching the full-conditionals. Importantly, the herded Gibbs algorithm is very close to the Gibbs algorithm and hence retains its simplicity of implementation.

The synthetic, denoising and named entity recognition experiments provided evidence that herded Gibbs outperforms Gibbs sampling. However, as discussed, herded Gibbs requires storage of the conditional distributions for all instantiations of the neighbors in the worst case. This storage requirement indicates that it is more suitable for sparse probabilistic graphical models, such as the CRFs used in information extraction. At the other extreme, the paper advanced the theory of deterministic sampling by showing that herded Gibbs converges with rate $O(1/T)$ for models with independent variables and fully-connected models. Thus, there is gap between theory and practice that needs to be narrowed. We do not anticipate that this will be an easy task, but it is certainly a key direction for future work.

We should mention that it is also possible to design parallel versions of herded Gibbs in an asynchronous Jacobi fashion. Preliminary study shows that these are less efficient than the synchronous Gauss-Seidel version of herded Gibbs discussed in this paper. However, if many cores are available, we strongly recommend the parallel implementation as it will likely outperform the current sequential implementation.

The design of efficient herding algorithms for densely connected probabilistic graphical models remains an important area for future research. Such algorithms, in conjunction with Rao Blackwellization, would enable us to attack many statistical inference tasks, including Bayesian variable selection and Dirichlet processes.

There are also interesting connections with other algorithms to explore. First, herding has ties to multicanonical sampling algorithms (Bornn et al., 2013), which while not deterministic, employ similar biasing/reweighting schemes. Second, if for a fully connected graphical model we build a new graph where every state is a node and directed connections exist between nodes that can be reached with a single herded Gibbs update, then herded Gibbs is very similar to the Rotor-Router model of Holroyd and Propp (2010)[2]. This deterministic analogue of a random walk has provably superior concentration rates for quantities such as normalized hitting frequencies, hitting times and occupation frequencies. In line with our own convergence results, it is shown that discrepancies in these quantities decrease as $O(1/T)$ instead of the usual $O(1/\sqrt{T})$. We expect that many of the results from this literature apply to herded Gibbs as well. The connection with the work of Art Owen and colleagues, see for example Chen et al. (2011), also needs to be explored further. Their work uses *completely uniformly distributed (CUD) sequences* to drive Markov chain Monte Carlo schemes. It is not clear, following discussions with Art Owen, that CUD sequences can be constructed in a greedy way as in herding.

2. We thank Art Owen for pointing out this connection.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0914783, 0928427, 1018433, 1216045, by NSERC, CIFAR's Neural Computation and Adaptive Perception Program, by DARPA under Grant No. FA8750-14-2-0117, by ARO under Grant No. W911NF-15-1-0172, by the Amazon AWS Research Grant, and by the Natural Sciences and Engineering Research Council of Canada.

Appendix A. Proof of Lemma 1

Proof We first show that $w \in (\pi - 1, \pi], \forall t \geq s$. This is easy to observe by induction as $w^{(s)} \in (\pi - 1, \pi]$ and if $w^{(t)} \in (\pi - 1, \pi]$ for some $t \geq s$, then, following Equation 5, we have:

$$w^{(t+1)} = \begin{cases} w^{(t)} + \pi - 1 \in (\pi - 1, 2\pi - 1] \subseteq (\pi - 1, \pi], & \text{if } w^{(t)} > 0, \\ w^{(t)} + \pi \in (2\pi - 1, \pi] \subseteq (\pi - 1, \pi], & \text{otherwise.} \end{cases} \tag{20}$$

Summing up both sides of Equation 5 over t immediately gives us the result of Equation 6 since:

$$T\pi - \sum_{t=s+1}^{s+T} \mathbb{I}[X^{(t)} = 1] = w^{(s+T)} - w^{(s)} \in [-1, 1]. \tag{21}$$

In addition, Equation 7 follows by observing that $\mathbb{I}[X^{(t)} = 0] = 1 - \mathbb{I}[X^{(t)} = 1]$. ■

Appendix B. Proof of Theorem 3

In this appendix, we give an upper bound for the convergence rate of the sampling distribution in fully connected graphs. As herded Gibbs sampling is deterministic, the distribution of a variable's state at every iteration degenerates to a single state. As such, we study here the empirical distribution of a collection of samples.

The structure of the proof is as follows (with notation defined in the next subsection): We study the distribution distance between the invariant distribution π and the empirical distribution of T samples collected starting from sweep τ, $P_T^{(\tau)}$. We show that the distance decreases as $\tau \Rightarrow \tau + 1$ with the help of an auxiliary regular Gibbs sampling Markov chain initialized at $\pi^{(0)} = P_T^{(\tau)}$, as shown in Figure 12. On the one hand, the distance between the regular Gibbs chain after one iteration, $\pi^{(1)}$, and π decreases according to the geometric convergence property of MCMC algorithms on compact state spaces. On the other hand, we show that in one step the distance between $P_T^{(\tau+1)}$ and $\pi^{(1)}$ increases by at most $O(1/T)$. Since the $O(1/T)$ distance term dominates the exponentially small distance term, the distance between $P_T^{(\tau+1)}$ and π is bounded by $O(1/T)$. Moreover, after a short burn-in period, $L = O(\log(T))$, the empirical distribution $P_T^{(\tau+L)}$ will have an approximation error in the order of $O(1/T)$.

B.1 Notation

Assume without loss of generality that in the systematic scanning policy, the variables are sampled in the order $1, 2, \cdots, N$.

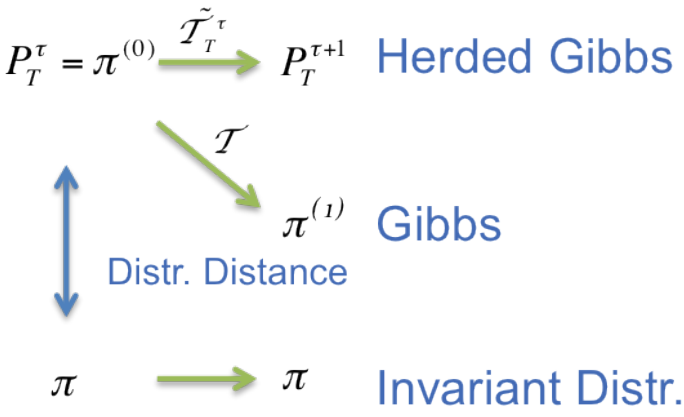


Figure 12: Transition kernels and relevant distances for the proof of Theorem 3.

B.1.1 State Distribution

- Denote by $\mathcal{X}_+$ the support of the distribution π, that is, the set of states with positive probability.
- We use τ to denote the time in terms of sweeps over all of the N variables, and t to denote the time in terms of steps where one step constitutes the updating of one variable. For example, at the end of τ sweeps, we have $t = \tau N$.
- Recall the sample/empirical distribution, $P_T^{(\tau)}$, presented in Definition 1.
- Denote the sample/empirical distribution at the i^{th} step within a sweep as $P_{T,i}^{(\tau)}, \tau \geq 0, T > 0, 0 \leq i \leq N$, as shown in Figure 13:

$$P_{T,i}^{(\tau)}(\mathbf{X} = \mathbf{x}) = \frac{1}{T} \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}^{(kN+i)} = \mathbf{x}).$$

This is the distribution of T samples collected at the i^{th} step of every sweep, starting from the τ^{th} sweep. Clearly, $P_T^{(\tau)} = P_{T,0}^{(\tau)} = P_{T,N}^{(\tau-1)}$.

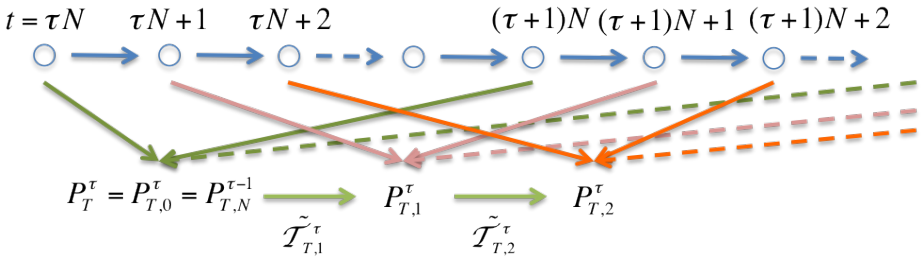


Figure 13: Distribution over time within a sweep.

- Denote the distribution of a regular Gibbs sampling Markov chain after L sweeps of updates over the N variables with $\pi^{(L)}, L \geq 0$.

  For a given time τ, we construct a Gibbs Markov chain with initial distribution $\pi^0 = P_T^{(\tau)}$ and the same scanning order of herded Gibbs, as shown in Figure 12.

B.1.2 Transition Kernel

- Denote the transition kernel of regular Gibbs for the step of updating a variable X_i with $\mathcal{T}_i$, and for a whole sweep with $\mathcal{T}$.

  By definition, $\pi^0\mathcal{T} = \pi^1$. The transition kernel for a single step can be represented as a $2^N \times 2^N$ matrix:

$$\mathcal{T}_i(\mathbf{x}, \mathbf{y}) = \begin{cases} 0, & \text{if } \mathbf{x}_{-i} \neq \mathbf{y}_{-i} \\ \pi(X_i = y_i | \mathbf{x}_{-i}), & \text{otherwise} \end{cases}, 1 \leq i \leq N, \mathbf{x}, \mathbf{y} \in \{0,1\}^N, \tag{22}$$

  where $\mathbf{x}$ is the current state vector of N variables, $\mathbf{y}$ is the state of the next step, and $\mathbf{x}_{-i}$ denotes all the components of $\mathbf{x}$ excluding the i^{th} component. If $\pi(\mathbf{x}_{-i}) = 0$, the conditional probability is undefined and we set it with an arbitrary distribution. Consequently, $\mathcal{T}$ can also be represented as:

$$\mathcal{T} = \mathcal{T}_1\mathcal{T}_2\cdots\mathcal{T}_N.$$

- Denote the Dobrushin ergodic coefficient (Brémaud, 1999) of the regular Gibbs kernel with $\eta \in [0,1]$. When $\eta < 1$, the regular Gibbs sampler has a geometric rate of convergence of

$$d_v(\pi^{(1)} - \pi) = d_v(\mathcal{T}\pi^{(0)} - \pi) \leq \eta d_v(\pi^{(0)} - \pi), \forall \pi^{(0)}. \tag{23}$$

  A common sufficient condition for $\eta < 1$ is that $\pi(\mathbf{X})$ is strictly positive.

- Consider the sequence of sample distributions $P_T^{(\tau)}, \tau = 0, 1, \cdots$ in Figures 1 and 13. We define the transition kernel of herded Gibbs for the step of updating variable X_i from $P_{T,i-1}^{(\tau)}$ to $P_{T,i}^{(\tau)}$ with $\tilde{\mathcal{T}}_{T,i}^{(\tau)}$, and for a whole sweep from $P_T^{(\tau)}$ to $P_T^{(\tau+1)}$ with $\tilde{\mathcal{T}}_T^{(\tau)}$. Unlike regular Gibbs, the transition kernel is not homogeneous. It depends on both the time τ and the sample size T. Nevertheless, we can still represent the single step transition kernel as a matrix:

$$\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x}, \mathbf{y}) = \begin{cases} 0, & \text{if } \mathbf{x}_{-i} \neq \mathbf{y}_{-i} \\ P_{T,i}^{(\tau)}(X_i = y_i | \mathbf{x}_{-i}), & \text{if } \mathbf{x}_{-i} = \mathbf{y}_{-i} \end{cases}, 1 \leq i \leq N, \mathbf{x}, \mathbf{y} \in \{0,1\}^N, \tag{24}$$

  where $P_{T,i}^{(\tau)}(X_i = y_i | \mathbf{x}_{-i})$ is defined as:

$$P_{T,i}^{(\tau)}(X_i = y_i | \mathbf{x}_{-i}) = \frac{N_{\text{num}}}{N_{\text{den}}},$$

$$N_{\text{num}} = TP_{T,i}^{(\tau)}(\mathbf{X}_{-i} = \mathbf{x}_{-i}, X_i = y_i) = \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}_{-i}^{(kN+i)} = \mathbf{x}_{-i}, X_i^{(kN+i)} = y_i),$$

$$N_{\text{den}} = TP_{T,i-1}^{(\tau)}(\mathbf{X}_{-i} = \mathbf{x}_{-i}) = \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}_{-i}^{(kN+i-1)} = \mathbf{x}_{-i}), \tag{25}$$

where N_{num} is the number of occurrences of a joint state, and N_{den} is the number of occurrences of a conditioning state in the previous step. When $\pi(\mathbf{x}_{-i}) = 0$, we know that $N_{\text{den}} = 0$ with a proper initialization of herded Gibbs, and we simply set $\tilde{\mathcal{T}}_{T,i}^{(\tau)} = \mathcal{T}_i$ for these entries. It is not hard to verify the following identity by expanding every term with its definition

$$P_{T,i}^{(\tau)} = P_{T,i-1}^{(\tau)} \tilde{\mathcal{T}}_{T,i}^{(\tau)},$$

and consequently,

$$P_{T}^{(\tau+1)} = P_{T}^{(\tau)} \tilde{\mathcal{T}}_{T}^{(\tau)},$$

with

$$\tilde{\mathcal{T}}_{T}^{(\tau)} = \tilde{\mathcal{T}}_{T,1}^{(\tau)} \tilde{\mathcal{T}}_{T,2}^{(\tau)} \cdots \tilde{\mathcal{T}}_{T,N}^{(\tau)}.$$

B.2 Linear Visiting Rate

We prove in this section that every joint state in the support of the target distribution is visited, at least, at a linear rate. This result will be used to measure the distance between the Gibbs and herded Gibbs transition kernels.

Proposition 5 *If a graph is fully connected, herded Gibbs sampling scans variables in a fixed order, and the corresponding Gibbs sampling Markov chain is irreducible, then for any state $\mathbf{x} \in \mathcal{X}_+$ and any index $i \in [1, N]$, the state is visited at least at a linear rate. Specifically,*

$$\begin{aligned} &\exists l > 0, B > 0, s.t., \forall i \in [1, N], \mathbf{x} \in \mathcal{X}_+, T \in \mathbb{N}, s \in \mathbb{N}, \\ &\sum_{k=s}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(t=Nk+i)} = \mathbf{x}\right] \geq lT - B. \end{aligned} \tag{26}$$

Denote the minimum nonzero conditional probability as

$$\pi_{\min} = \min_{1 \leq i \leq N, \pi(x_i|\mathbf{x}_{-i}) > 0} \pi(x_i|\mathbf{x}_{-i}).$$

The following lemma, which is needed to prove Proposition 5, gives an inequality between the number of visits of two sets of states in consecutive steps. Please refer to Figure 14 for an illustration of the two sets of states and the mapping defined in the lemma.

Lemma 6 *Given any integer $i \in [1, N]$, a set of states $\mathbb{X} \subseteq \mathcal{X}_+$, and a mapping $F : \mathbb{X} \to \mathcal{X}_+$ that corresponds to any possible state transition for a Gibbs step at variable X_i, that is, F is any mapping satisfying*

$$\mathbf{F}(\mathbf{x})_{-i} = \mathbf{x}_{-i} \text{ and } \pi(\mathbf{F}(\mathbf{x})_i|\mathbf{x}) > 0, \forall \mathbf{x} \in \mathbb{X}, \tag{27}$$

let $\mathbb{Y} = \cup_{\mathbf{x} \in \mathbb{X}} F(\mathbf{x})$. We have that, if the graph is fully connected, for any $s \geq 0$ and $T > 0$, the number of times any state in $\mathbb{Y}$ is visited in the set of all i'th step in sweeps $s, s+1, \ldots, s+T-1$, denoted as $C_i = \{t = kN + i : s \leq k \leq s+T-1\}$, is lower

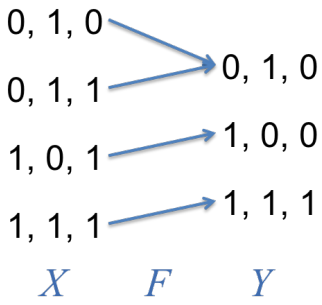


Figure 14: Example of the mapping defined in Lemma 6 with $i = 3$.

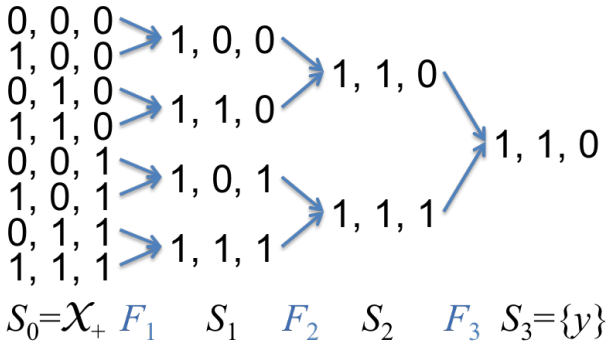


Figure 15: Example of the set of all paths from $\mathbf{x} \in \mathbb{X}_+$ to $\mathbf{y} = (1, 1, 0)$. Variables are updated in the order of $i = 1, 2, 3$. In this example, all the conditional distributions are positive and $\mathbf{y}$ can be reached from any state in $N = 3$ steps. Therefore, $t^* = 3$.

bounded by a function of the number of times any state in $\mathbb{X}$ *is visited in the previous steps* $C_{i-1} = \{t = kN + i - 1 : s \le k \le s + T - 1\}$ *as:*

$$\sum_{t \in C_i} \mathbb{I}\left[\mathbf{X}^{(t)} \in \mathbb{Y}\right] \ge \pi_{min} \sum_{t \in C_{i-1}} \mathbb{I}\left[\mathbf{X}^{(t)} \in \mathbb{X}\right] - |\mathbb{Y}|. \tag{28}$$

Proof As a complement to Condition 27, we can define F^{-1} as the inverse mapping from $\mathbb{Y}$ to subsets of $\mathbb{X}$ so that for any $\mathbf{y} \in \mathbb{Y}$, $\mathbf{x} \in F^{-1}(\mathbf{y})$, we have $\mathbf{x}_{-i} = \mathbf{y}_{-i}$, and $\cup_{\mathbf{y} \in \mathbb{Y}} F^{-1}(\mathbf{y}) = \mathbb{X}$. It's easy to observe that

$$\mathbb{I}\left[\mathbf{X}_{-i}^{(t)} = \mathbf{y}_{-i}\right] \ge \mathbb{I}\left[\mathbf{X}^{(t)} \in F^{-1}(\mathbf{y})\right], \forall t, \mathbf{y} \in \mathbb{Y}. \tag{29}$$

At every step of the herded Gibbs algorithm, only one weight is updated. Consider the sequence of steps when a weight $w_{i,\mathbf{x}_{N(i)}}$ is updated, we denote the segment of that sequence in sweeps $[s, s + T - 1]$ as $C_i(\mathbf{x}_{N(i)})$. By definition of the herded Gibbs algorithm, $C_i(\mathbf{x}_{N(i)}) = \{t : t \in C_i, \mathbf{X}_{N(i)}^{(t-1)} = \mathbf{x}_{N(i)}\} = \{t : t \in C_i, \mathbf{X}_{N(i)}^{(t)} = \mathbf{x}_{N(i)}\} \subseteq C_i$.

Because the graph is fully connected, $N(i) = -i$, the full conditional state $\mathbf{X}_{-i}^{(t)}$ with $t \in C_i(\mathbf{x}_{N(i)})$ is uniquely determined. Since the value $X_i^{(t)}$ is determined by $w_{i,\mathbf{x}_{-i}}$ for any $t \in C_i(\mathbf{x}_{-i})$, we can apply Lemma 1 and get that for any $\mathbf{y} \in \mathbb{Y}$,

$$\sum_{t \in C_i} \mathbb{I}\left[\mathbf{X}^{(t)} = \mathbf{y}\right] = \sum_{t \in C_i(\mathbf{y}_{-i})} \mathbb{I}\left[X_i^{(t)} = y_i\right] \ge \pi(y_i|\mathbf{y}_{-i})|C_i(\mathbf{y}_{-i})| - 1 \ge \pi_{\min}|C_i(\mathbf{y}_{-i})| - 1. \tag{30}$$

Since the variables $\mathbf{X}_{-i}$ are not changed at steps in C_i, combining with Equation 29 we can show

$$|C_i(\mathbf{y}_{-i})| = \sum_{t\in C_i} \mathbb{I}\left[\mathbf{X}_{-i}^{(t)} = \mathbf{y}_{-i}\right] = \sum_{t\in C_{i-1}} \mathbb{I}\left[\mathbf{X}_{-i}^{(t)} = \mathbf{y}_{-i}\right] \geq \sum_{t\in C_{i-1}} \mathbb{I}\left[\mathbf{X}^{(t)} \in F^{-1}(\mathbf{y})\right]. \quad (31)$$

Combining the fact that $\cup_{\mathbf{y}\in\mathbb{Y}} F^{-1}(\mathbf{y}) = \mathbb{X}$ and summing up both sides of Equation 30 over $\mathbb{Y}$ proves the lemma:

$$\sum_{t\in C_i} \mathbb{I}\left[\mathbf{X}^{(t)} \in \mathbb{Y}\right] \geq \sum_{\mathbf{y}\in\mathbb{Y}} \left(\pi_{\min} \sum_{t\in C_{i-1}} \mathbb{I}\left[\mathbf{X}^{(t)} \in F^{-1}(\mathbf{y})\right] - 1 \right) \geq \pi_{\min} \sum_{t\in C_{i-1}} \mathbb{I}\left[\mathbf{X}^{(t)} \in \mathbb{X}\right] - |\mathbb{Y}|. \quad (32)$$

■

Remark 7 *A fully connected graph is a necessary condition for the application of Lemma 1 in Equation 30. If a graph is not fully connected ($N(i) \neq -i$), a weight $w_{i,\mathbf{x}_{N(i)}}$ is associated with a partial conditional state and may be shared by multiple full conditional states. In that case $C_i(\mathbf{x}_{N(i)}) = \{t : t \in C_i, \mathbf{X}_{N(i)}^{(t)} = \mathbf{x}_{N(i)}\}$, and we can no longer use Lemma 1 to get a lower bound for the number of visits to a particular **joint** state, not to mention a linear visiting rate in Proposition 5.*

Now let us prove Proposition 5 by iteratively applying Lemma 6.

Proof [Proof of Proposition 5] Because the corresponding Gibbs sampler is irreducible and any Gibbs sampler is aperiodic, there exists a constant $t^* > 0$ such that for any state $\mathbf{y} \in \mathcal{X}_+$, and any step i in a sweep, we can find a path of length t^* from any state $\mathbf{x} \in \mathcal{X}_+$ with a positive transition probability, $Path(\mathbf{x}) = (\mathbf{x} = \mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(t^*) = \mathbf{y})$, where each step of the path follows the Gibbs updating scheme. For a strictly positive distribution, the minimum value of t^* is N. We show an example of the paths in a graph with 3 variables and a strictly positive distribution in Figure 15.

Denote $\tau^* = \lceil t^*/N \rceil$ and the j^{th} element of the path $Path(\mathbf{x})$ as $Path(\mathbf{x}, j)$. We can define $t^* + 1$ subsets $S_j \subseteq \mathcal{X}_+, 0 \leq j \leq t^*$ as the union of all the j^{th} states in the path from any state in $\mathcal{X}_+$:

$$S_j = \cup_{\mathbf{x}\in\mathcal{X}_+} Path(\mathbf{x}, j).$$

By definition of these paths, we know $S_0 = \mathcal{X}_+$ and $S_{t^*} = \{\mathbf{y}\}$, and there exits an integer $i(j)$ and a mapping $F_j : S_{j-1} \to S_j, \forall j$ that satisfy the condition in Lemma 6 ($i(j)$ is the index of the variable to be updated, and the mapping is defined by the transition paths). Also notice that any state in S_j can be different from $\mathbf{y}$ by at most $\min\{N, t^* - j\}$ variables, and therefore $|S_j| \leq 2^{\min\{N, t^*-j\}}$.

Let us apply Lemma 6 recursively from $j = t^*$ to 1 as

$$\begin{aligned}
&\sum_{k=s}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i)} = \mathbf{y}\right] \geq \sum_{k=s+\tau^*}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i)} = \mathbf{y}\right] \\
&= \sum_{k=s+\tau^*}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i)} \in S_{t^*}\right] && \text{(b.c. } S_{t^*} = \{\mathbf{y}\}\text{)} \\
&\geq \pi_{\min} \sum_{k=s+\tau^*}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i-1)} \in S_{t^*-1}\right] - |S_{t^*}| && \text{(Lemma 6, } \mathbb{X} = S_{t^*-1},\ \mathbb{Y} = S_{t^*}\text{)} \\
&\geq \pi_{\min}^2 \sum_{k=s+\tau^*}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i-2)} \in S_{t^*-2}\right] - \pi_{\min}|S_{t^*-1}| - |S_{t^*}| && \text{(Lemma 6, } \mathbb{X} = S_{t^*-2},\ \mathbb{Y} = S_{t^*-1}\text{)} \\
&\geq \cdots \\
&\geq \pi_{\min}^{t^*} \sum_{k=s+\tau^*}^{s+T-1} \mathbb{I}\left[\mathbf{X}^{(Nk+i-t^*)} \in S_0 = \mathcal{X}_+\right] - \sum_{j=0}^{t^*-1} \pi_{\min}^j |S_{t^*-j}| && \text{(Lemma 6, } \mathbb{X} = S_0,\ \mathbb{Y} = S_1\text{)} \\
&\geq \pi_{\min}^{t^*}(T - \tau^*) - \sum_{j=0}^{t^*-1} \pi_{\min}^j 2^{\min\{N,j\}}. && \text{(b.c. } \mathbb{I}\left[\mathbf{X}^{(t)} \in \mathcal{X}_+\right] = 1\text{)}
\end{aligned} \tag{33}$$

The proof is concluded by choosing the constants

$$l = \pi_{\min}^{t^*}, \quad B = \tau^* \pi_{\min}^{t^*} + \sum_{j=0}^{t^*-1} \pi_{\min}^j 2^{\min\{N,j\}}. \tag{34}$$

■

Remark 8 *For a strictly positive distribution, the constants reduce to*

$$l = \pi_{min}^N, \quad B = \pi_{min}^N + \sum_{j=0}^{N-1} (2\pi_{min})^j = \pi_{min}^N + \frac{1 - (2\pi_{min})^N}{1 - 2\pi_{min}}.$$

B.3 Herded Gibbs's Transition Kernel $\tilde{\mathcal{T}}_T^{(\tau)}$ is an Approximation to $\mathcal{T}$

The following proposition shows that $\tilde{\mathcal{T}}_T^{(\tau)}$ is an approximation to the regular Gibbs sampler's transition kernel $\mathcal{T}$ with an error of $O(1/T)$.

Proposition 9 *For a fully connected graph, if the herded Gibbs has a fixed scanning order and the corresponding Gibbs sampling Markov chain is irreducible, then for any $\tau \geq 0$, $T \geq T^* := \frac{2B}{l}$ where l and B are the constants in Proposition 5, the following inequality holds:*

$$\|\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T}\|_\infty \leq \frac{4N}{lT}. \tag{35}$$

Proof

When $\mathbf{x} \notin \mathcal{X}_+$, we have the equality $\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x},\mathbf{y}) = \mathcal{T}_i(\mathbf{x},\mathbf{y})$ by definition. When $\mathbf{x} \in \mathcal{X}_+$ but $\mathbf{y} \notin \mathcal{X}_+$, then $N_{\text{den}} = 0$ (see the notation of $\tilde{\mathcal{T}}_T^{(\tau)}$ for definition of N_{den}) as $\mathbf{y}$ will never be visited and thus $\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x},\mathbf{y}) = 0 = \mathcal{T}_i(\mathbf{x},\mathbf{y})$ also holds. Let us consider the entries in $\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x},\mathbf{y})$ with $\mathbf{x},\mathbf{y} \in \mathcal{X}_+$ in the following.

Because $\mathbf{X}_{-i}$ is not updated at i^{th} step of every sweep, we can replace $i-1$ in the definition of N_{den} by i and get

$$N_{\text{den}} = \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}_{-i}^{(kN+i)} = \mathbf{x}_{-i}).$$

Notice that the set of times $\{t = kN + i : \tau \le k \le \tau + T - 1, \mathbf{X}_{-i}^t = \mathbf{x}_{-i})\}$, whose size is N_{den}, is a consecutive set of times when $w_{i,\mathbf{x}_{-i}}$ is updated. By Lemma 1, we obtain a bound for the numerator

$$\begin{aligned} &N_{\text{num}} \in [N_{\text{den}}\pi(X_i = y_i|\mathbf{x}_{-i}) - 1, N_{\text{den}}\pi(X_i = y_i|\mathbf{x}_{-i}) + 1] \Leftrightarrow \\ &|P_{T,i}^{(\tau)}(X_i = y_i|\mathbf{x}_{-i}) - \pi(X_i = y_i|\mathbf{x}_{-i})| = |\frac{N_{\text{num}}}{N_{\text{den}}} - \pi(X_i = y_i|\mathbf{x}_{-i})| \le \frac{1}{N_{\text{den}}}. \end{aligned} \tag{36}$$

Also by Proposition 5, we know every state in $\mathcal{X}_+$ is visited at a linear rate, there hence exist constants $l > 0$ and $B > 0$, such that the number of occurrence of any conditioning state $\mathbf{x}_{-i}$, N_{den}, is bounded by

$$N_{\text{den}} \ge \sum_{k=\tau}^{\tau+T-1} \mathbb{I}(\mathbf{X}^{(kN+i)} = \mathbf{x}) \ge lT - B \ge \frac{l}{2}T, \quad \forall T \ge \frac{2B}{l}. \tag{37}$$

Combining equations (36) and (37), we obtain

$$|P_{T,i}^{(\tau)}(X_i = y_i|\mathbf{x}_{-i}) - \pi(X_i = y_i|\mathbf{x}_{-i})| \le \frac{2}{lT}, \quad \forall T \ge \frac{2B}{l}. \tag{38}$$

Since the matrix $\tilde{\mathcal{T}}_{T,i}^{(\tau)}$ and $\mathcal{T}_i$ differ only at those elements where $\mathbf{x}_{-i} = \mathbf{y}_{-i}$, we can bound the L_1 induced norm of the transposed matrix of their difference by

$$\begin{aligned} \|(\tilde{\mathcal{T}}_{T,i}^{(\tau)} - \mathcal{T}_i)^T\|_1 &= \max_{\mathbf{x}} \sum_{\mathbf{y}} |\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x},\mathbf{y}) - \mathcal{T}_i(\mathbf{x},\mathbf{y})| \\ &= \max_{\mathbf{x}} \sum_{y_i} |P_{T,i}^{(\tau)}(X_i = y_i|\mathbf{x}_{-i}) - \pi(X_i = y_i|\mathbf{x}_{-i})| \\ &\le \frac{4}{lT}, \quad \forall T \ge \frac{2B}{l}. \end{aligned} \tag{39}$$

Observing that both $\tilde{\mathcal{T}}_T^{(\tau)}$ and $\mathcal{T}$ are multiplications of N component transition matrices, and the transition matrices, $\tilde{\mathcal{T}}_T^{(\tau)}$ and $\mathcal{T}_i$, have a unit L_1 induced norm as:

$$\|(\tilde{\mathcal{T}}_{T,i}^{(\tau)})^T\|_1 = \max_{\mathbf{x}} \sum_{\mathbf{y}} |\tilde{\mathcal{T}}_{T,i}^{(\tau)}(\mathbf{x},\mathbf{y})| = \max_{\mathbf{x}} \sum_{\mathbf{y}} P_{T,i}^{(\tau)}(X_i = y_i|\mathbf{x}_{-i}) = 1, \tag{40}$$

$$\|(\mathcal{T}_i)^T\|_1 = \max_{\mathbf{x}} \sum_{\mathbf{y}} |\mathcal{T}_i(\mathbf{x},\mathbf{y})| = \max_{\mathbf{x}} \sum_{\mathbf{y}} P(X_i = y_i|\mathbf{x}_{-i}) = 1, \tag{41}$$

we can further bound the L_1 norm of the difference, $(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})^T$. Let $P \in \mathbb{R}^N$ be any vector with nonzero norm. Using the triangular inequality, the difference of the resulting vectors after applying $\tilde{\mathcal{T}}_T^{(\tau)}$ and $\mathcal{T}$ is bounded by

$$\begin{aligned}\|P(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})\|_1 =& \|P\tilde{\mathcal{T}}_{T,1}^{(\tau)} \dots \tilde{\mathcal{T}}_{T,N}^{(\tau)} - P\mathcal{T} \dots \mathcal{T}_N\|_1 \\ \leq& \|P(\tilde{\mathcal{T}}_{T,1}^{(\tau)} - \mathcal{T}_1)\tilde{\mathcal{T}}_{T,2}^{(\tau)} \dots \tilde{\mathcal{T}}_{T,N}^{(\tau)}\|_1 + \\ & \|P\mathcal{T}_1(\tilde{\mathcal{T}}_{T,2}^{(\tau)} - \mathcal{T}_2)\tilde{\mathcal{T}}_{T,3}^{(\tau)} \dots \tilde{\mathcal{T}}_{T,N}^{(\tau)}\|_1 + \\ & \dots \\ & \|P\mathcal{T}_1 \dots \mathcal{T}_{N-1}(\tilde{\mathcal{T}}_{T,N}^{(\tau)} - \mathcal{T}_N)\|_1, \end{aligned} \tag{42}$$

where the i'th term is

$$\begin{aligned}\|P\mathcal{T}_1 \dots \mathcal{T}_{i-1}(\tilde{\mathcal{T}}_{T,i}^{(\tau)} - \mathcal{T}_i)\tilde{\mathcal{T}}_{T,i+1}^{(\tau)} \dots \tilde{\mathcal{T}}_{T,N}^{(\tau)}\|_1 &\leq \|P\mathcal{T}_1 \dots \mathcal{T}_{i-1}(\tilde{\mathcal{T}}_{T,i}^{(\tau)} - \mathcal{T}_i)\|_1 && \text{(Unit } L_1 \text{ norm, Eqn. 40)} \\ &\leq \|P\mathcal{T}_1 \dots \mathcal{T}_{i-1}\|_1 \frac{4}{lT} && \text{(Eqn. 39)} \\ &\leq \|P\|_1 \frac{4}{lT}. && \text{(Unit } L_1 \text{ norm, Eqn. 41)} \end{aligned} \tag{43}$$

Consequently, we get the L_1 induced norm of $(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})^T$ as

$$\|(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})^T\| = \max_P \frac{\|P(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})\|_1}{\|P\|_1} \leq \frac{4N}{lT}, \quad \forall T \geq \frac{2B}{l}. \tag{44}$$

■

B.4 Proof of Theorem 3

When we initialize the herded Gibbs and regular Gibbs with the same distribution (see Figure 12), since the transition kernel of herded Gibbs is an approximation to regular Gibbs and the distribution of regular Gibbs converges to the invariant distribution, we expect that herded Gibbs also approaches the invariant distribution.

Proof [Proof of Theorem 3] Construct an auxiliary regular Gibbs sampling Markov chain initialized with $\pi^{(0)}(\mathbf{X}) = P_T^{(\tau)}(\mathbf{X})$ and the same scanning order as herded Gibbs. As $\eta < 1$, the Gibbs Markov chain has uniform geometric convergence rate as shown in Equation (23).

Also, the Gibbs Markov chain must be irreducible due to $\eta < 1$ and therefore Proposition 9 applies here. We can bound the distance between the distributions of herded Gibbs after one sweep of all variables, $P_T^{(\tau+1)}$, and the distribution after one sweep of regular Gibbs sampling, $\pi^{(1)}$ by

$$\begin{aligned} & d_v(P_T^{(\tau+1)} - \pi^{(1)}) = d_v(\pi^{(0)}(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})) = \frac{1}{2}\|\pi^{(0)}(\tilde{\mathcal{T}}_T^{(\tau)} - \mathcal{T})\|_1 \\ & \leq \frac{2N}{lT}\|\pi^{(0)}\|_1 = \frac{2N}{lT}, \quad \forall T \geq T^*, \tau \geq 0. \end{aligned} \tag{45}$$

Now we study the change of discrepancy between $P_T^{(\tau)}$ and π as a function as τ. Applying the triangle inequality of d_v:

$$d_v(P_T^{(\tau+1)} - \pi) = d_v(P_T^{(\tau+1)} - \pi^{(1)} + \pi^{(1)} - \pi) \leq d_v(P_T^{(\tau+1)} - \pi^{(1)}) + d_v(\pi^{(1)} - \pi)$$
$$\leq \frac{2N}{lT} + \eta d_v(P_T^{(\tau)} - \pi), \quad \forall T \geq T^*, \tau \geq 0. \tag{46}$$

The last inequality follows Equations (23) and (45). When the sample distribution is outside a neighborhood of π, $\mathcal{B}_{\epsilon_1}(\pi)$, with $\epsilon_1 = \frac{4N}{(1-\eta)lT}$, i.e.

$$d_v(P_T^{(\tau)} - \pi) \geq \frac{4N}{(1-\eta)lT}, \tag{47}$$

we get a geometric convergence rate toward the invariant distribution by combining the two equations above:

$$d_v(P_T^{(\tau+1)} - \pi) \leq \frac{1-\eta}{2} d_v(P_T^{(\tau)} - \pi) + \eta d_v(P_T^{(\tau)} - \pi) = \frac{1+\eta}{2} d_v(P_T^{(\tau)} - \pi). \tag{48}$$

So starting from $\tau = 0$, we have a burn-in period for herded Gibbs to enter $\mathcal{B}_{\epsilon_1}(\pi)$ in a finite number of rounds. Denote the first time it enters the neighborhood by τ'. According to the geometric convergence rate in Equations 48 and $d_v(P_T^{(0)} - \pi) \leq 1$

$$\tau' \leq \left\lceil \log_{\frac{1+\eta}{2}} \left(\frac{\epsilon_1}{d_v(P_T^{(0)} - \pi)} \right) \right\rceil \leq \left\lceil \log_{\frac{1+\eta}{2}} (\epsilon_1) \right\rceil = \lceil \tau^*(T) \rceil. \tag{49}$$

After that burn-in period, the herded Gibbs sampler will stay within a smaller neighborhood, $\mathcal{B}_{\epsilon_2}(\pi)$, with $\epsilon_2 = \frac{1+\eta}{1-\eta} \frac{2N}{lT}$, i.e.

$$d_v(P_T^{(\tau)} - \pi) \leq \frac{1+\eta}{1-\eta} \frac{2N}{lT}, \quad \forall \tau > \tau'. \tag{50}$$

This is proved by induction:

1. Equation (50) holds at $\tau = \tau' + 1$. This is because $P_T^{(\tau')} \in \mathcal{B}_{\epsilon_1}(\pi)$ and following Eqn. 46 we get
$$d_v(P_T^{(\tau'+1)} - \pi) \leq \frac{2N}{lT} + \eta\epsilon_1 = \epsilon_2. \tag{51}$$
2. For any $\tau \geq \tau' + 2$, assume $P_T^{(\tau-1)} \in \mathcal{B}_{\epsilon_2}(\pi)$. Since $\epsilon_2 < \epsilon_1$, $P_T^{(\tau-1)}$ is also in the ball $\mathcal{B}_{\epsilon_1}(\pi)$. We can apply the same computation as when $\tau = \tau' + 1$ to prove $d_v(P_T^{(\tau)} - \pi) \leq \epsilon_2$. So inequality (50) is always satisfied by induction.

Consequently, Theorem 3 is proved when combining (50) with the inequality $\tau' \leq \lceil \tau^*(T) \rceil$ in Equation(49). ■

Remark 10 *Similarly to the regular Gibbs sampler, the herded Gibbs sampler also has a burn-in period with geometric convergence rate. After that, the distribution discrepancy is in the order of $O(1/T)$, which is faster than the regular Gibbs sampler. Notice that the length of the burn-in period depends on T, specifically as a function of* $\log(T)$.

Remark 11 *Irrationality is not required to prove the convergence on a fully-connected graph.*

B.5 Proof of Corollary 4

Proof Since $\tau^*(T)$ is a monotonically increasing function of T, for any $T \geq T^* + \tau^*(T^*)$, we can find a number t so that

$$T = t + \tau^*(t), t \geq T^*.$$

Partition the sample sequence $S_{0,T} = \{\mathbf{X}^{(kN)} : 0 \leq k < T\}$ into two parts: the burn-in period $S_{0,\tau^*(t)}$ and the stable period $S_{\tau^*(t),T}$. The discrepancy in the burn-in period is bounded by 1 and according to Theorem 3, the discrepancy in the stable period is bounded by

$$d_v(\tilde{P}(S_{t,T}) - \pi) \leq \frac{\lambda}{t}.$$

Hence, the discrepancy of the whole set $S_{0,T}$ is bounded by

$$\begin{aligned}
&d_v(\tilde{P}(S_{0,T}) - \pi) = d_v\left(\frac{\tau^*(t)}{T}\tilde{P}(S_{0,\tau^*(t)}) + \frac{t}{T}\tilde{P}(S_{\tau^*(t),T}) - \pi\right) \\
&\leq d_v\left(\frac{\tau^*(t)}{T}(\tilde{P}(S_{0,\tau^*(t)}) - \pi\right) + d_v\left(\frac{t}{T}(\tilde{P}(S_{\tau^*(t),T}) - \pi\right) \\
&\leq \frac{\tau^*(t)}{T} d_v(\tilde{P}(S_{0,\tau^*(t)}) - \pi) + \frac{t}{T} d_v(\tilde{P}(S_{\tau^*(t),T}) - \pi) \\
&\leq \frac{\tau^*(t)}{T} \cdot 1 + \frac{t}{T}\frac{\lambda}{t} \leq \frac{\tau^*(T) + \lambda}{T}.
\end{aligned} \tag{52}$$

■

References

Pulp Fiction - Wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/Pulp_Fiction`, 2013.

D. H. Ackley, G. Hinton, and T. Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9:147–169, 1985.

C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50(1):5–43, 2003.

F. Bach, S. Lacoste-Julien, and G. Obozinski. On the equivalence between herding and conditional gradient algorithms. In *International Conference on Machine Learning*, 2012.

L. Bornn, P. E. Jacob, P. Del Moral, and A. Doucet. An adaptive interacting wang–landau algorithm for automatic density exploration. *Journal of Computational and Graphical Statistics*, 22(3):749–773, 2013.

P. Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues*, volume 31. Springer, 1999.

P. Carbonetto, J. Kisynski, N. de Freitas, and D. Poole. Nonparametric Bayesian logic. In *Uncertainty in Artificial Intelligence*, pages 85–93, 2005.

S. Chen, J. Dick, and A. B. Owen. Consistency of Markov chain quasi-Monte Carlo on continuous state spaces. *The Annals of Statistics*, 39(2):673–701, 04 2011. doi: 10.1214/10-AOS831. URL `http://dx.doi.org/10.1214/10-AOS831`.

S. Chen, M. Matsumoto, T. Nishimura, and A. Owen. New inputs and methods for Markov chain quasi-Monte Carlo. In L. Plaskota and H. Wozniakowski, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*, volume 23 of *Springer Proceedings in Mathematics & Statistics*, pages 313–327. Springer Berlin Heidelberg, 2012. URL `http://dx.doi.org/10.1007/978-3-642-27440-4_15`.

Y. Chen, M. Welling, and A.J. Smola. Supersamples from kernel-herding. In *Uncertainty in Artificial Intelligence*, pages 109–116, 2010.

A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, 2001.

A. Gelfand, Y. Chen, L. van der Maaten, and M. Welling. On herding and the perceptron cycling theorem. In *Advances in Neural Information Processing Systems*, pages 694–702, 2010.

N. D. Goodman, V. K. Mansinghka, D. M. Roy, K. Bonawitz, and J. B. Tenenbaum. Church: a language for generative models. *Uncertainty in Artificial Intelligence*, 2008.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

A. E. Holroyd and J. Propp. Rotor walks and Markov chains. *Algorithmic Probability and Combinatorics*, 520:105–126, 2010.

F. Huszar and D. Duvenaud. Optimally-weighted herding is Bayesian quadrature. In *Proceedings of the Twenty-Eighth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-12)*, pages 377–386, Corvallis, Oregon, 2012. AUAI Press.

T. Grenager J. R. Finkel and C. Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219885. URL `http://dx.doi.org/10.3115/1219840.1219885`.

J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2001.

D. J. Lunn, A. Thomas, N. Best, and D. Spiegelhalter. WinBUGS a Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, 10(4):325–337, 2000.

B. Milch and S. Russell. General-purpose MCMC inference over relational structures. In *Uncertainty in Artificial Intelligence*, pages 349–358, 2006.

K. P. Murphy. *Machine Learning: a Probabilistic Perspective*. MIT Press, 2012.

I. Murray and L. T. Elliott. Driving Markov chain Monte Carlo with a dependent random stream. *arXiv preprint arXiv:1204.3187*, 2012.

I. Porteous, D. Newman, A. Ihler, A. Asuncion, P. Smyth, and M. Welling. Fast collapsed Gibbs sampling for latent Dirichlet allocation. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 569–577, 2008.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 2nd edition, 2004.

M. Welling. Herding dynamical weights to learn. In *International Conference on Machine Learning*, pages 1121–1128, 2009a.

M. Welling. Herding dynamic weights for partially observed random field models. In *Uncertainty in Artificial Intelligence*, pages 599–606, 2009b.

H. Weyl. Über die gleichverteilung von zahlen mod. eins. *Mathematische Annalen*, 77: 313–352, 1916. ISSN 0025-5831. doi: 10.1007/BF01475864. URL `http://dx.doi.org/10.1007/BF01475864`.