Client-only Distributed Markov Chain Monte Carlo Sampling over a Network

Anonymous authors
Paper under double-blind review

Abstract

We aim to sample from a target $\exp(-\sum_{i=1}^{n} f_i(x|\mathcal{D}_i))$ where each client f_i only has access to local data $\mathcal{D}_i$. We present a fully distributed Markov Chain Monte Carlo (MCMC) sampler that operates through client-to-client communication, eliminating the need for additional centralized servers. Unlike MCMC algorithms that rely on server-client structures, our proposed sampler is entirely distributed, enhancing security and robustness through decentralized communication. In contrast to limited decentralized algorithms arising from Langevin dynamics, our sampler utilizes blocked Gibbs sampling on an augmented distribution. Furthermore, we establish a non-asymptotic analysis of our sampler, employing innovative techniques. This study contributes to one of the initial analyses of the non-asymptotic behavior of a fully distributed sampler arising from Gibbs sampling.

1 Introduction

In recent years, inspired by the rapid development in optimization, a lot of sampling algorithms have been proposed and then analyzed to obtain their non-asymptotic behaviors (Dalalyan, 2017b; Dalalyan & Riou-Durand, 2020; Zhang et al., 2023; Fan et al., 2023; Altschuler & Chewi, 2023). In this work, we are interested in sampling from a target distribution of $x \in \mathbb{R}^d$ defined as

$$\pi \propto \exp\left(-\sum_{i=1}^{n} f_i(x|\mathcal{D}_i)\right)$$

in a *distributed* manner. With this formulation, we implicitly have that each potential (negative log density) f_i only possesses access to its local data $\mathcal{D}_i$. Fully distributed systems, e.g., distributed sensor networks (Qi et al., 2001) and edge computation (Mach & Becvar, 2017), provide advantages over centralized ones by enabling computing agents to share limited information, reducing communication costs, and improving security and robustness. These factors make them potentially superior in terms of computational performance and data protection. These raise the need to develop and analyze a distributed sampling algorithm without centralized servers. Our target distribution is the straightforward generalization of the objective function $\sum_{i=1}^{n} f_i(x|\mathcal{D}_i)$ in distributed optimization, and this problem naturally arises in various applications of distributed optimization, e.g., power systems, sensor networks, and smart manufacturing (Yang et al., 2019).

In the machine learning community, our sampling problem is heavily related to Bayesian learning (Cao et al., 2023), an area carrying on Bayesian inference and uncertain quantification on learned models. Consider a learning setting where the goal is to train models with minimizing $\sum_{i=1}^{n} f_i(x|\mathcal{D}_i)$. As mentioned in Sun et al. (2022), one can view each f_i as a loss function on data collection $\mathcal{D}_i$. Compared with only searching for the minimizer of $\sum_i f_i$, Bayesian inference on the distribution π can give more detailed information on the loss landscape. Moreover, Bayesian inference is preferred when only limited data is available; applying prior distributions on model parameters can significantly enhance the performance of models trained on limited data (Sun et al., 2019). In what follows, we omit the dependency on $\mathcal{D}_i$ for brevity.

Many existing distributed Markov Chain Monte Carlo (MCMC) algorithms are deployed with server-client architectures: at each step, the server aggregates information from clients as shown in Figure 1. Distributed sampling via moment-sharing is proposed in Xu et al. (2014). More parallel MCMC algorithms are presented in Wang & Dunson (2013); Neiswanger et al. (2013); Wang et al. (2015); Chowdhury & Jermaine (2018); De Souza et al. (2022). Recently, several distributed MCMC algorithms (Vono et al., 2022b; Plassier et al., 2021; Kotelevskii et al., 2022) are proposed to address practical issues like safety, communication cost, and heterogeneity. These parallel and distributed MCMC algorithms either rely on client-server architectures or lack non-asymptotic convergence rates. To the best of our knowledge, only a handful of sampling literature (Kungurtsev, 2020; Parayil et al., 2020; Gürbüzbalaban et al., 2021; Kolesov & Kungurtsev, 2021; Kungurtsev et al., 2023) proposes fully distributed samplers and establishes the non-asymptotic results. These samplers arise from Langevin Monte Carlo or Hamiltonian Monte Carlo, leveraging the concept of gradient tracking in distributed optimization (Pu & Nedić, 2021). However, unsuitable hyperparameters of these methods may result in failure to sample from the target. As demonstrated in our experiments, samplers based on Langevin Monte Carlo diverge with unreasonably large step sizes, while our proposed sampler ensures convergence with the same step sizes.

Another well-known sampler is the Gibbs sampler, commonly used in asynchronous settings where multiple nodes can update their samples without waiting for each other (Terenin et al., 2020). However, the exploration of the non-asymptotic behaviors of a fully distributed sampler based on *Gibbs sampling* remains uncharted territory. Even though the parallel Gibbs sampler was already proposed in the distributed setting in Gonzalez et al. (2011), we give non-asymptotic results for the first time. Given the intricate nature of Gibbs sampling, which differs from Langevin dynamics, our main challenge is to develop innovative proof ideas to analyze non-asymptotic behaviors.

Figure 1: Comparison of server-client architectures with centralized servers (Left) and Trees that enable distributed sampling (Right). In the server-client structure, each client (small node) must communicate with a centralized server (big node).

We fill in the gap with a novel and fully distributed sampling framework. Our work represents one of the *first* endeavors to address the non-asymptotic convergence results of a fully distributed sampling algorithm on a Gibbs sampler. To utilize a specific Gibbs sampler, we assume all the clients are distributed on a *bipartite graph*. Note that every connected graph has a spanning tree. For a comparison of the server-client structures and trees, see Figure 1.

To sample from the target $\pi \propto \exp(-\sum_{i=1}^n f_i(x))$ where $x \in \mathbb{R}^d$, we instead consider sampling from an augmented distribution of $\{x_i\}_{i=1}^n \in \mathbb{R}^{nd}$, $\hat{\pi} \propto \exp(-\sum_{i=1}^n f_i(x_i) - \sum_{a\in A}\sum_{b\in B} {}^{\sigma_{ab}}/_{2\eta}\|x_a - x_b\|^2)$ where A and B stand for two disjoint node groups of a bipartite graph. Here, each $x_i \in \mathbb{R}^d$ stands for one node. We assume $\sigma_{ab} \in (0,1]$ if there is an edge between x_a and x_b, otherwise, $\sigma_{ab} = 0$. Furthermore, $\sum_a \sigma_{ab} \le 1, \sum_b \sigma_{ab} \le 1$ for any a, b. The nature of Gibbs sampling, iteratively sampling from conditional distributions, makes it a candidate for distributed sampling. With the specific design of $\hat{\pi}$, it can be observed that if η is small enough, the marginals of $\hat{\pi}$ and the distribution of the averaged samples can be arbitrarily close to π.

Our main theoretical contribution is the non-asymptotic bounds of the distance between the average of samples and the target in Theorem 5.4. Combining with improved convergence results of sampling from $\hat{\pi}$ in Yuan et al. (2023), we eventually give the overall iteration complexity in Proposition 5.5. We also demonstrate our sampler's superior performance under various settings.

In summary, our contributions are listed below.

- We propose a new distributed sampling algorithm (Algorithm 2) that does not require a server node and can be deployed on any bipartite graphs. Our sampler is applicable to any target distributions, provided that mild regularity conditions are met (Remark 5.6). Unlike previous works relying on Langevin Monte Carlo or Hamiltonian Monte Carlo, our sampler builds upon Gibbs sampling.
- We analyze its non-asymptotic results (Proposition 5.5) with the bound improved from(Yuan et al., 2023) and the non-trivial upper bound between marginals of $\hat{\pi}$ and π (Theorem 5.4) established in this work with novel techniques.
- We conduct both qualitative and quantitative experiments in Section 6 and Appendix A.5, demonstrating the efficiency of our sampler for large-scale distributed systems and the superior performance over the baselines.

2 Related works

Decentralized MCMC with theoretical guarantees In Kungurtsev (2020), convergence and consensus guarantee are provided for decentralized Langevin dynamics, with the necessity of a diminishing stepsize for ensuring asymptotic consensus. Gürbüzbalaban et al. (2021) independently investigates decentralized stochastic gradient Langevin dynamics and decentralized stochastic gradient Hamiltonian dynamics under the strong convexity assumption. Parayil et al. (2020) consider unadjusted Langevin algorithm with non-log-concave potentials, replacing convexity assumption with bounded conditions of disagreement on the gradient among the distributed agents. Kolesov & Kungurtsev (2021) considers directed graphs in their analysis. Kungurtsev et al. (2023) shows theoretical guarantees for decentralized Metropolis-adjusted Hamiltonian Monte Carlo, enabling a constant step size. It is worth pointing out that except for Parayil et al. (2020), the convergence rates are quantified with Wasserstein distance, while ours is Total Variation distance, which is due to innovative proof techniques used in this work. Recently, Bhar et al. (2023a;b) analyze decentralized Langevin algorithms under asynchronous and event-triggered conditions. Moreover, these works except for Kungurtsev et al. (2023) require diminishing step sizes, while our step size η is constant.

See Appendix A.1 for more related works.

3 Preliminaries

3.1 Problem setting

Throughout this paper, we aim to perform Bayesian inference on distributed models by sampling from a target distribution

$$\pi \propto \exp\left(-\sum_{i=1}^{n} f_i(x)\right) \tag{1}$$

where $x \in \mathbb{R}^d$. In these models, data is stored locally at each client. To uphold privacy, only estimated parameters are permitted to traverse across the clients.

Without loss of generality, we assume that all the clients are located on an undirected tree structure. This assumption can be satisfied as long as all the clients are on an undirected connected graph. To obtain the non-asymptotic convergence bounds, we utilize the standard assumptions on each potential: we assume each f_i is α_i-strongly-convex and β_i-smooth. This condition implies that for any x and y,

$$\nabla f_i(y)^T(x-y) + {}^{\alpha_i}\!/_2\|x-y\|^2 \leq f_i(x) - f_i(y),$$
$$\nabla f_i(y)^T(x-y) + {}^{\beta_i}\!/_2\|x-y\|^2 \geq f_i(x) - f_i(y).$$

3.2 Notation

For a joint distribution π^{XY} defined on (X, Y), the X-marginal distribution is represented by π^X, and the X-conditional distribution given y is $\pi^{X|Y=y}$. The iteration complexity is the required number of iterations of Gibbs samplers to achieve a given accuracy.

3.3 Sampling background

Here we list several useful tools and lemmas we utilize. The Total Variation (TV) distance and the Kullback-Leibler (KL) divergence are widely used to quantify the distance between two distributions. For two measures μ and ν with density function representations, we have $\mathrm{TV}(\mu, \nu) = \frac{1}{2}\|\mu - \nu\|_1$. The KL divergence of μ and ν is defined as $\mathrm{KL}(\mu||\nu) = \int \log \frac{\mu}{\nu}\mu$. Pinsker's inequality states that $\mathrm{TV}(\mu, \nu) \le \sqrt{\frac{1}{2}\mathrm{KL}(\mu||\nu)}$. We also quantify the convergence with W_2 distance in experiments. It is defined as $W_2(\mu, \nu) = \inf_\pi (\mathbf{E}_\pi \|x - y\|^2)^{1/2}$ where π is the coupling of μ and ν.

Recall that a probability measure ν satisfies Logarithmic Sobolev Inequality (LSI) with a constant $\alpha > 0$ (in short, LSI(α)), if for any distribution μ such than $\mu \ll \nu$, the KL divergence $\mathrm{KL}(\mu||\nu)$ and Fisher information $I(\mu||\nu) = \int \|\nabla \log \frac{\mu}{\nu}\|^2 \mu$ satisfy $\mathrm{KL}(\mu||\nu) \le \frac{1}{2\alpha} I(\mu||\nu)$. It is well-known that a strongly log-concave distribution ν with a constant α, equivalently its potential is α-strongly-convex, satisfies LSI with the same constant. Moreover, we utilize the following lemma in our proof.

Lemma 3.1 (Theorem 1 in Durmus & Moulines (2019))**.** *For a strongly log-concave distribution π with α defined on $\mathbb{R}^d$, one has $\mathbf{E}_{x\sim\pi}\|x - x^*\|^2 \le \frac{d}{\alpha}$ where x^* is the minimizer of the potential.*

4 Analysis of Composite Sampling

In this section, we aim to provide the analysis for composite sampling, i.e., sampling from a special case of equation 1,

$$\pi \propto \exp(-f(x) - g(x)). \tag{2}$$

We emphasize that in this distributed setting, each client symmetrically generates samples from conditional distributions, without the need for an additional server to aggregate parameters from the two clients. This two-node composite sampling is a special case of equation 1, and our main analysis of distributed sampling is essentially a generalization of ideas in this section. To proceed, we have the following assumption.

Assumption 4.1. Let $f(x)$ be α_f-strongly-convex and β_f-smooth. Similarly, let $g(x)$ be an α_g-strongly-convex and β_g-smooth function. It is also necessary to assume f and g share the same minimizer m.

Inspired by the success of proximal samplers, this sampling task can be performed by sampling from an augmented distribution

$$\pi^{XY} \propto \exp\left(-f(x) - g(y) - \frac{1}{2\eta}\|x - y\|^2\right)$$

and then control the TV distance between π and π^X with a sufficiently small η. The procedure is summarized in Algorithm 1.

The strong convexity and smoothness assumptions are standard and have been used in a series of sampling works (Dalalyan, 2017a; Dalalyan & Karagulyan, 2019; Fan et al., 2023; Altschuler & Chewi, 2023). Note that the other assumption that f and g have the same minimizer is required to establish the non-asymptotic bounds of $\mathrm{TV}(\pi, \pi^X)$. The same assumption has been discussed in Lee et al. (2021) and Yuan et al. (2023). In Lee et al. (2021), this assumption is necessary to analyze the non-asymptotic results of samplers for the same target equation 2. As mentioned in Lee et al. (2021), for two functions $\hat{f}$ and $\hat{g}$ that have different minimizers, one can easily find m such that $\nabla \hat{f}(m) + \nabla \hat{g}(m) = 0$ since both f and g are strongly convex. Then define $f(x) := \hat{f}(x) - \langle \nabla f(m), x\rangle$ and $g(x) := \hat{g}(x) - \langle \nabla g(m), x\rangle$. It is easy to verify that sampling from $\exp(-f(x) - g(x))$ is equivalent to the original target, and f and g share the same minimizer m. Furthermore, the necessity of this assumption has been illustrated in Example 3 in Yuan et al. (2023). It is worth emphasizing that Example 3 shows that, without the same minimizer assumption, the non-asymptotic bound also depends on other parameters: when f and g are quadratic, it also relies on the distance between the centers of the two functions.

Algorithm 1 A Sampler for Composite Potentials

```
Input: $\pi : \exp(-f(x) - g(x))$: target distribution
Output: $x^K$: Samplers within $\epsilon$ TV distance of $\pi$
Step 1: Construct a distribution $\pi^{XY} : \exp(-f(x) - g(y) - \frac{1}{2\eta}\|x - y\|^2)$
# The following is the Gibbs sampler for $\pi^{XY}$
Step 2: Generate an initial sample $x^0$ from $\nu_0^X$
for $k \leftarrow 0, \cdots, K-1$ do
    Step 3: Sample $y^k \sim \pi^{Y|X=x^k} \propto \exp(-g(y) - \frac{1}{2\eta}\|x^k - y\|^2)$
    Step 4: Sample $x^{k+1} \sim \pi^{X|Y=y^k} \propto \exp(-f(x) - \frac{1}{2\eta}\|x - y^k\|^2)$
end for
```

The convergence result of the Gibbs sampler in Algorithm 1 has been established in Theorem 5 in Yuan et al. (2023). For the sake of completeness, we restate it here.

Theorem 4.2 (Theorem 5 in Yuan et al. (2023))**.** *Assume Step 3 and Step 4 are exact without any bias, i.e., y^k and x^{k+1} follow the conditional distributions exactly. Denote the distribution of the k-th samples (x^k, y^k) by ν_k^{XY}. For the target distribution π^{XY}, under Assumption 4.1, we have*

$$\mathrm{KL}(\nu_{k+1}^X || \pi^X) \le \mathrm{KL}(\nu_k^X || \pi^X) \frac{1}{(1 + \eta\alpha_f)^2 (1 + \eta\alpha_g)^2},$$

where ν_k^X is the X-marginal of ν_k^{XY}.

To obtain the overall iteration complexity of Algorithm 1, it remains to select a suitable η based on $\mathrm{TV}(\pi, \pi^X)$, which can be seen from Theorem 4.3. Even though similar intuition has been discussed in Yuan et al. (2023), our work rigorously proves the non-asymptotic bound of $\mathrm{TV}(\pi, \pi^X)$ for the first time.

Theorem 4.3. *Under Assumption 4.1, we have*

$$\mathrm{TV}(\pi, \pi^X) \le \sqrt{\frac{\beta_g^2 d}{2(\alpha_f + \frac{\alpha_g}{\eta\alpha_g + 1})} \left(\frac{\eta^2 \beta_g^2}{\alpha_f + \alpha_g} + \frac{\eta}{\eta\alpha_g + 1} \right)} \tag{3}$$

Proof. As both f and g are strongly convex, by Theorem 3.8. in Saumard & Wellner (2014), the X-marginal distribution π^X is strongly-log-concave with a parameter $\alpha_f + \frac{\alpha_g}{\eta\alpha_g + 1}$. Hence, π^X satisfies $\mathrm{LSI}(\alpha_f + \frac{\alpha_g}{\eta\alpha_g + 1})$. This implies that

$$\mathrm{KL}(\pi || \pi^X) \le \frac{1}{2(\alpha_f + \frac{\alpha_g}{\eta\alpha_g + 1})} I(\pi || \pi^X) \tag{4}$$

By definition, the relative Fisher information $I(\pi || \pi^X)$ satisfies

$$\begin{aligned} I(\pi || \pi^X) &= \mathop{\mathbf{E}}_{x \sim \pi} \left\| \nabla \log \frac{\pi}{\pi^X} \right\|^2 \\ &= \mathop{\mathbf{E}}_{x \sim \pi} \|\nabla g(x) + \nabla \log G_\eta(x)\|^2, \end{aligned} \tag{5}$$

where $G_\eta(x) = \int \exp(-g(y) - \frac{1}{2\eta}\|x - y\|^2) dy$. To proceed, notice that

$$\nabla_x \log G_\eta(x) = \mathbf{E}_{y \sim \exp(-g(y) - \frac{1}{2\eta}\|x - y\|^2)} (-\nabla g(y)) \tag{6}$$

where we use the identity $\int\Big(\nabla_y \exp\Big(-g(y) - \frac{1}{2\eta}\|x-y\|^2\Big)\Big)\mathrm{d}y = 0$. Combining equation 5 and equation 6 yields that

$$\begin{aligned} &I(\pi||\pi^X) \\ &= \mathbf{E}_{x\sim\pi}\Big\|\mathbf{E}_{y\sim\exp(-g(y)-\frac{1}{2\eta}\|x-y\|^2)}(\nabla g(x) - \nabla g(y))\Big\|^2 \\ &\le \mathbf{E}_{x\sim\pi}\mathbf{E}_{y\sim\exp(-g(y)-\frac{1}{2\eta}\|x-y\|^2)}\|\nabla g(x) - \nabla g(y)\|^2 \\ &\le \beta_g^2\mathbf{E}_{x\sim\pi}\mathbf{E}_{y\sim\exp(-g(y)-\frac{1}{2\eta}\|x-y\|^2)}\|x-y\|^2 \end{aligned} \tag{7}$$

where we adopt Jensen's inequality and the smoothness of g. Then we estimate the upper bounds of two expectations separately. Denote the minimizer of $g(y) + \frac{1}{2\eta}\|x-y\|^2$ as y^*. For the inner expectation over y in equation 7, we have

$$\begin{aligned} &\mathbf{E}_{y\sim\exp(-g(y)-\frac{1}{2\eta}\|x-y\|^2)}\|x-y\|^2 \\ &\le 2\|x-y^*\|^2 + 2\mathbf{E}_{y\sim\exp(-g(y)-\frac{1}{2\eta}\|x-y\|^2)}\|y^*-y\|^2 \\ &\le 2\|x-y^*\|^2 + \frac{2d}{\alpha_g + 1/\eta}. \end{aligned} \tag{8}$$

Here, the last line is due to Lemma 3.1. Since $\nabla g(y^*) + \frac{1}{\eta}(y^* - x) = 0$, for the outer expectation over x in equation 7,

$$\mathbf{E}_{x\sim\pi}\|x-y^*\|^2 = \eta^2\mathbf{E}_{x\sim\pi}\|\nabla g(y^*)\|^2 \tag{9}$$

$$\le \eta^2\beta_g^2\mathbf{E}_{x\sim\pi}\|y^*-m\|^2$$

$$\le \eta^2\beta_g^2\mathbf{E}_{x\sim\pi}\|x-m\|^2 \le \eta^2\beta_g^2\frac{d}{\alpha_f+\alpha_g} \tag{10}$$

The second inequality is due to the smoothness of g and the assumption that $\nabla g(m) = 0$, and the third inequality is from the contraction property of the proximal method in optimization. For the last line, we use Lemma 3.1 and the assumption that f and g have the same minimizer. With equation 4, equation 7, equation 8 and equation 9, one has

$$\mathrm{KL}(\pi||\pi^X) \le \frac{\beta_g^2 d}{\alpha_f + \frac{\alpha_g}{\eta\alpha_g+1}}\left(\frac{\eta^2\beta_g^2}{\alpha_f+\alpha_g} + \frac{\eta}{\eta\alpha_g+1}\right) \tag{11}$$

By Pinsker's inequality, we obtain equation 3. □

Combining Theorem 4.2 and Theorem 4.3, we are ready to show the overall iteration complexity in Proposition 4.4.

Proposition 4.4. *For simplicity, we assume η is small enough that the second order term $\eta^2\beta_g^2/(\alpha_f+\alpha_g)$ in Theorem 4.3 can be ignored. Under Assumption 4.1, to achieve ϵ TV distance to π, with Algorithm 1 where $\eta = \boldsymbol{O}\Big(\frac{(\alpha_f+\alpha_g)\epsilon^2}{(\beta_f+\beta_g)^2 d}\Big)$, one needs iterations*

$$K = \boldsymbol{O}\bigg(\frac{(\beta_f+\beta_g)^2 d}{(\alpha_f+\alpha_g)^2\epsilon^2}\log\bigg(\frac{\mathrm{KL}(\nu_0^X||\pi^X)}{\epsilon^2}\bigg)\bigg). \tag{12}$$

Proof. Notice that the overall biased term ϵ comes from two components: the Gibbs sampler for π^{XY} and the distance between π and π^X.

By Theorem 4.2, the number of iterations to achieve $\mathrm{TV}(\nu_K^X, \pi^X) \le \epsilon/2$ is

$$K = \boldsymbol{O}\bigg(\log\bigg(\frac{\mathrm{KL}(\nu_0^X||\pi^X)}{\epsilon^2}\bigg)\frac{1}{\eta(\alpha_f+\alpha_g)}\bigg). \tag{13}$$

To ensure $\mathrm{TV}(\pi, \pi^X) \leq \epsilon/2$, combining the assumption that $\frac{\eta^2 \beta_g^2}{\alpha_f + \alpha_g}$ vanishes and Theorem 4.3, one needs

$$\eta = \boldsymbol{O}\left(\frac{(\alpha_f + \alpha_g)\epsilon^2}{\beta_g^2 d} \right). \tag{14}$$

Therefore, to have $\mathrm{TV}(\tilde{\nu}_T^X, \pi) \leq \epsilon$, with equation 13 and equation 14, η and K can be $\eta = \boldsymbol{O}\left(\frac{(\alpha_f+\alpha_g)\epsilon^2}{(\beta_f+\beta_g)^2 d} \right)$, and $K = \boldsymbol{O}\left(\frac{(\beta_f+\beta_g)^2 d}{(\alpha_f+\alpha_g)^2 \epsilon^2} \log\left(\frac{\mathrm{KL}(\nu_0^X \| \pi^X)}{\epsilon^2} \right) \right)$. ☐

Remark 4.5 (Weaker Assumptions). The result of Proposition 4.4 can be generalized to the case that g is still strongly convex and smooth, but f is convex and smooth. Then, with the same procedure, one can show the iteration complexity is $K = \boldsymbol{O}\left(\frac{(\beta_f+\beta_g)^2 d}{\alpha_g^2 \epsilon^2} \log\left(\frac{\mathrm{KL}(\nu_0^X \| \pi^X)}{\epsilon^2} \right) \right)$.

5 Analysis of Distributed Sampling over bipartite graphs

In this section, we present our main result: a distributed sampler and its non-asymptotic analysis. Recall that we are interested in sampling from

$$\pi \propto \exp\left(- \sum_{i=1}^{n} f_i(x) \right) \tag{15}$$

with the following Assumption 5.1.

Assumption 5.1. Without loss of generality, assume that each $f_i(x)$ is α-strongly convex and β-smooth. We also assume that each f_i shares the same minimizer m. As explained in Section 4, this assumption is necessary to have non-asymptotic results, and our sampler still converges asymptotically even without this assumption, as discussed in Remark 5.6.

It is worth noting that the bipartite graph assumption can be fulfilled on any connected graph. For example, every connected graph has a spanning tree (Serre, 2002), and a tree is always a bipartite graph (Bondy et al., 1976). In this setting, the sampler and its iteration complexity are established with steps similar to Section 4. To begin with, denote

$$\hat{\pi}^{AB} \propto \exp\left(- \sum_{i=1}^{n} f_i(x_i) - \sum_{a \in A} \sum_{b \in B} {}^{\sigma_{ab}}\!/_{2\eta} \|x_a - x_b\|^2 \right). \tag{16}$$

With a slight abuse of notation, we will interchangeably use $\hat{\pi}^{AB}$ and $\hat{\pi}$. We have the following assumption for the weights σ_{ab}.

Assumption 5.2. Let $\sigma_{ab} > 0$ stand for there is an edge between nodes a and b. Otherwise, $\sigma_{ab} = 0$. Moreover, we assume $\sum_a \sigma_{ab} \leq 1$ and $\sum_b \sigma_{ab} \leq 1$. Furthermore, we assume the graph is a fully connected bipartite graph. Define the n by n doubly stochastic matrix Σ whose element at the s-th row and t-th column is σ_{st}. This matrix satisfies $\sigma_{ts} = \sigma_{st} = \sigma_{ab} \geq 0$ if $s = a$ and $t = b$, and $\sigma_{ss} = 1 - \sum_a \sigma_{sa} \geq 0$. Let $(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^n$ be the eigenvalues of Σ organizing in the descending order, i.e., $\lambda_i \geq \lambda_{i+1}$,

A bipartite graph divides all nodes into two independent sets: A and B. One can view f_i as a local distribution on each node, and the quadratic terms $\|x_a - x_b\|^2$ correspond to Gaussian distributions on edges. With the same idea in Section 4, one can sample from $\hat{\pi}$ with Gibbs sampling and then reduce the distance between the target and the distribution of $\sum_{i=1}^{n} x_i/n$ with a small η. To sample from $\hat{\pi}$, it is natural to use a block Gibbs sampler where, at each iteration, each set of parameters is updated simultaneously as shown in Algorithm 2. Namely, in Step 3-4, one can generate the k-th samples, $\{x_b^k\}_{b \in B}$ and $\{x_a^k\}_{a \in A}$, *in parallel*. We emphasize that, thanks to the bipartite graph structure, our sampler is fully distributed without the need for server nodes. In practice, each node only receives information from its neighbors.

To obtain the overall iteration complexity of the sampler in Algorithm 2, we start with the non-asymptotic bounds associated with the block Gibbs sampler. Theorem 10 in Yuan et al. (2023) establishes the convergence rate of the block Gibbs sampler over bipartite graphs. We improve the original convergence rates after

Algorithm 2 Distributed Sampling over a bipartite graph

```
**Input:** $\pi$ as in equation 1: the target distribution
**Output:** $\{x_a^K\}_{a\in A}$ and $\{x_b^K\}_{b\in B}$
Step 1: Construct a distribution $\hat{\pi}^{AB}$ as in equation 16
# The following is the block Gibbs sampler for $\hat{\pi}^{AB}$
Step 2: Generate the initial samples $\{x_a^A\}_{a\in A}$ from $\nu_0^A$.
**for** $k \leftarrow 0, \cdots, K-1$ **do**
    Step 3: Sample $\{x_b^k\}_{b\in B} \sim \hat{\pi}^{\{x_b\}|\{x_a^k\}_{a\in A}}$
    which is proportional to $\exp(-\sum_{b\in B} f_b(x_b) - \sum_{a\in A}\sum_{b\in B} \frac{\sigma_{ab}}{2\eta}\|x_a^k - x_b\|^2))$
    Step 4: Sample $\{x_a^{k+1}\}_{a\in A} \sim \hat{\pi}^{\{x_a\}|\{x_b^k\}_{b\in B}}$
    which is proportional to $\exp(-\sum_{a\in A} f_a(x_a) - \sum_{a\in A}\sum_{b\in B} \frac{\sigma_{ab}}{2\eta}\|x_a - x_b^k\|^2))$
**end for**
```

reviewing its proof. The improvement comes from both the definition of $\hat{\pi}^{AB}$ and a tighter upper bound of c_t in Equation (12) in Yuan et al. (2023). Notice that in Yuan et al. (2023), $\sigma_{ab} \in [0,1]$ without the additional constrain that $\sum_a \sigma_{ab} \le 1, \sum_b \sigma_{ab} \le 1$ we add in this work. Hence, the lower bound of η_j used in the proof of Theorem 10 in Yuan et al. (2023), which is defined as ${}^{\eta}/_{\sum_a \sigma_{ab}}$ with our notation, can be improved to η. This leads to Lemma A.1. Directly applying our improved results yields the following convergence rate. The proof of Theorem 5.3 is in Appendix A.2.

Theorem 5.3 (Improved from Theorem 10 in Yuan et al. (2023))**.** *Assume Steps 3 and 4 in Algorithm 2 are exact without biased terms. Denote the distribution of the k-th samples by ν_k^{AB}. For the target distribution $\hat{\pi}^{AB}$ equation 16, under Assumption 5.1, we have*

$$\mathrm{KL}(\nu_k^{AB}||\hat{\pi}^{AB}) \le \exp(-kC)\mathrm{KL}(\nu_0^{A}||\hat{\pi}^{A}) \tag{17}$$

where C is

$$\begin{aligned} C = \eta \int_0^1 \Bigg(& \left[\frac{\eta t(1-t)}{\min_b \sum_a \sigma_{ab}} + \frac{(1-t)^2|B|}{\alpha} + \frac{t^2}{\alpha}\right]^{-1} \\ & + \left[\frac{\eta t(1-t)}{\min_a \sum_b \sigma_{ab}} + \frac{(1-t)^2|A|}{\alpha} + \frac{t^2}{\alpha}\right]^{-1}\Bigg)\mathrm{d}t. \end{aligned} \tag{18}$$

Here $|A|$ and $|B|$ are the number of nodes in the two disjoint groups, respectively.

Now we show the proof on bounding $\mathrm{TV}(\bar{\pi}, \pi)$ where $\bar{\pi}$ is the distribution of $\sum_i x_i/n$ with joint distribution being $\hat{\pi}$. The following Theorem 5.4 holds for any graph, so for the sake of presentation, we rewrite

$$\hat{\pi} \propto \exp\left(-\sum_{i=1}^n f_i(x_i) - \sum_{s=1}^n\sum_{t=1}^n {}^{\sigma_{st}}/_{2\eta}\|x_s - x_t\|^2\right). \tag{19}$$

Theorem 5.4. *Under Assumption 5.1 on the potentials and Assumption 5.2 on the weights, we have*

$$\mathrm{TV}(\bar{\pi}, \pi) = \boldsymbol{O}\left(\sqrt{\frac{n\beta^2 d}{\alpha}\left(\frac{(n-1)}{\alpha + \frac{n}{\eta}(1-\lambda_2)} + \frac{\eta^2\beta^2}{(1-\lambda_2)^2\alpha}\right)}\right). \tag{20}$$

where $\bar{\pi}$ is the distribution of $\sum_i x_i/n$ given the joint distribution $\hat{\pi}$ in equation 19, and λ_2 is the second largest eigenvalue of the doubly stochastic matrix Σ defined in Assumption 5.2

Proof. First, notice that $\sum_{s=1}^n \sum_{t=1}^n {}^{\sigma_{st}}/_{2\eta}\|x_s - x_t\|^2 = \frac{1}{2\eta}X^T(\mathbf{I}_{nd} - \Sigma \otimes \mathbf{I}_d)X$ where $X^T = (x_1^T, \ldots, x_n^T)$.

It is well-known that the eigenvalues of Σ, denoted as $(\lambda_1, \ldots, \lambda_n)$ in Assumption 5.2, are from -1 to 1, and $(1, \ldots, 1)$ is one eigenvector corresponding to eigenvalue 1. Define the eigendecomposition of $\mathbf{I}_{nd} - \Sigma \otimes \mathbf{I}_d$ as

$U\Lambda U^T$. Note that the eigenvalues of $\mathbf{I}_{nd} - \Sigma \otimes \mathbf{I}_d$ are $1 - \lambda_i$ for each λ_i in the set of eigenvalues of Σ. Since we assume the graph is fully connected, there is only one solution to ${}^1\!/_{2\eta} X^T(\mathbf{I}_{nd} - \Sigma \otimes \mathbf{I}_d)X = 0$, which is $x_1 = x_2 \ldots = x_n$. This implies that $1 - \lambda_1 = 0$ and $1 - \lambda_i > 0$ for $i = 2, \ldots, n$.

With the eigendecomposition, the potential of $\hat{\pi}$ is $F(\sqrt{n}U\hat{X}) + \frac{n}{2\eta}\hat{X}^T\Lambda\hat{X}$. Here, $F(X) := \sum_{i=1}^n f_i(x_i)$ and $\hat{X} := \frac{1}{\sqrt{n}}U^T X$. As the first d columns of $\sqrt{n}U$ are $(1, \ldots, 1)^T \otimes \mathbf{I}_d$, we have $\hat{x}_1 = \frac{1}{n}\sum_{i=1}^n x_i$. Then, the distribution over the average $\frac{1}{n}\sum_{i=1}^n x_i$ is

$$\hat{\pi}^{\hat{X}_1} \propto \int \exp(-F(\sqrt{n}U\hat{X}) - {}^n\!/_{2\eta}\hat{X}^T\Lambda\hat{X}) \prod_{i=2}^n \mathrm{d}\hat{x}_i. \tag{21}$$

Intuitively, when η is small enough, $\hat{X}^T\Lambda\hat{X}$ should be approximately zero. Since $1 - \lambda_i = 0$ if and only if $i = 1$, $\hat{\pi}^{\hat{X}_1}$ should converge to $\exp(-F(\sqrt{n}U\hat{X}))|_{\hat{X}^T = (\hat{x}_1^T, \mathbf{0}, \ldots, \mathbf{0})}$ as η goes to 0. With this intuition, we have the following rigorous proof.

When $\hat{X}^T = (\hat{x}_1^T, \mathbf{0}, \ldots, \mathbf{0})$, one has

$$F(\sqrt{n}U\hat{X}) = \sum_{i=1}^n f_i(\hat{x}_1), \tag{22}$$

which is exactly the potential of the target π. Hence, it is sufficient to bound the TV distance between distributions in equation 21 and equation 22. This is achieved following similar proof ideas used in the proof of Theorem 4.3.

Under Assumption 5.1, by Theorem 3.8. in Saumard & Wellner (2014), the $\hat{X}_1$-marginal distribution of $\hat{\pi}(\hat{X})$ is strongly-log-concave with a parameter $n\alpha$. Hence, $\pi^{\hat{X}_1}$ satisfies LSI($n\alpha$). This implies that

$$\mathrm{KL}(\pi||\hat{\pi}^{\hat{X}_1}) \leq \frac{1}{2n\alpha} I(\pi||\hat{\pi}^{\hat{X}_1}). \tag{23}$$

By definition and Jensen's inequality,

$$\begin{aligned}
I(\pi||\pi^{\hat{X}_1}) &= \mathbf{E}_\pi \left\| \nabla_{\hat{x}_1} \log \frac{\pi}{\hat{\pi}^{\hat{X}_1}} \right\|^2 = \mathbf{E}_\pi \left\| \nabla_{\hat{x}_1} \log \frac{\hat{\pi}^{\hat{X}_1}}{\pi} \right\|^2 \\
&= \mathbf{E}_\pi \| \nabla_{\hat{x}_1} F(\sqrt{n}U\hat{X})|_{\hat{X} = (\hat{x}_1, \mathbf{0}, \ldots, \mathbf{0})} \\
&\quad - \mathbf{E}_{\hat{\pi}^{\hat{X}_1 = \hat{x}_1}} \nabla_{\hat{x}_1} F(\sqrt{n}U\hat{X})|_{\hat{X} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)} \|^2 \\
&\leq \mathbf{E}_\pi \mathbf{E}_{\hat{\pi}^{\hat{X}_1 = \hat{x}_1}} \| \nabla_{\hat{x}_1} F(\sqrt{n}U\hat{X})|_{\hat{X} = (\hat{x}_1, \mathbf{0}, \ldots, \mathbf{0})} \\
&\quad - \nabla_{\hat{x}_1} F(\sqrt{n}U\hat{X})|_{\hat{X} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)} \|^2.
\end{aligned}$$

Here $\hat{\pi}^{\hat{X}_1 = \hat{x}_1}$ is the conditional distribution of $\hat{\pi}$ given $\hat{x}_1$, i.e.,

$$\exp\left(-F(\sqrt{n}U\hat{X})|_{\hat{X} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)} - \frac{n}{2\eta} \sum_{i=2}^n (1 - \lambda_i) \|\hat{x}_i\|^2 \right)$$

Then with the definition of $F(X)$, the assumption that $f_i(x)$ is β-smooth and Cauchy inequality, we have

$$I(\pi||\pi^{\hat{X}_1}) \leq n^2\beta^2 \mathbf{E}_\pi \mathbf{E}_{\hat{\pi}^{\hat{X}_1 = \hat{x}_1}} \|(\hat{x}_2^T, \ldots, \hat{x}_n^T)U_2^T\|^2.$$

Here $U_2 \in \mathbb{R}^{nd \times (n-1)d}$ consisting of the last $(n-1)d$ columns of U. It follows that

$$\begin{aligned}
&I(\pi||\pi^{\hat{X}_1}) \\
&\leq 2n^2\beta^2 \mathbf{E}_\pi \mathbf{E}_{\hat{\pi}^{\hat{X}_1 = \hat{x}_1}} \|(\hat{x}_2^T - \tilde{x}_2^T, \ldots, \hat{x}_n^T - \tilde{x}_n^T)U_2^T\|^2 \\
&\quad + 2n^2\beta^2 \mathbf{E}_\pi \mathbf{E}_{\hat{\pi}^{\hat{X}_1 = \hat{x}_1}} \|(\tilde{x}_2^T, \ldots, \tilde{x}_n^T)U_2^T\|^2 \\
&\leq 2n^2\beta^2 \left(\frac{(n-1)d}{\alpha + \frac{n}{\eta}(1 - \lambda_2)} + \mathbf{E}_\pi \|(\tilde{x}_2^T, \ldots, \tilde{x}_n^T)U_2^T\|^2 \right)
\end{aligned} \tag{24}$$

where $\{\tilde{x}_i\}$ is the minimizer of the potential of $\hat{\pi}^{\hat{X}_1=\hat{x}_1}$ that satisfies

$$\nabla_{\hat{x}_i} F(\sqrt{n}U\hat{X})|_{\hat{X}=(\hat{x}_1,\tilde{x}_2,\dots,\tilde{x}_n)} + \frac{n}{\eta}(1-\lambda_i)\tilde{x}_i = 0, \tag{25}$$

and the last line follows from Lemma 3.1 and the fact that the strong convexity constant of $\hat{\pi}^{\hat{X}_1=\hat{x}_1}$ is $\alpha + \frac{n}{\eta}(1-\lambda_2)$. Then, by equation 25, the smoothness of f_i, and each f_i has the minimizer m,

$$\begin{aligned} &\mathbf{E}_\pi \|(\tilde{x}_2^T,\dots,\tilde{x}_n^T)U_2^T\|^2 \\ &\le \frac{\eta^2}{n(1-\lambda_2)^2}\mathbf{E}_\pi\|U_2U_2^T\nabla F|_{\hat{X}=(\hat{x}_1,\tilde{x}_2,\dots,\tilde{x}_n)}\|^2 \\ &\le \frac{\eta^2\beta^2 n}{(1-\lambda_2)^2}\mathbf{E}_\pi\|\hat{x}_1 - m\|^2 \le \frac{\eta^2\beta^2 d}{(1-\lambda_2)^2\alpha} \end{aligned} \tag{26}$$

where $\nabla^T F = (\nabla f_1^T(x_1),\dots,\nabla f_n^T(x_n))$ and $(x_1^T,\dots,x_n^T)^T = \sqrt{n}U\hat{X}|_{\hat{X}=(\hat{x}_1,\tilde{x}_2,\dots,\tilde{x}_n)}$. The first line is given by equation 25. The last line is obtained by the assumption that each f_i has the minimizer m and the contraction property of the proximal method. With equation 23, equation 24, equation 26 and Pinsker's inequality, we complete the proof. ☐

Finally, we present the overall iteration complexity, assuming η is sufficiently small. In Proposition 5.5, the iteration complexity is affected by the structure of the bipartite graph from two aspects: the number of nodes n and the sparsity measured by the spectral gap $1-\lambda_2$. In Appendix A.5.2, we demonstrate that our sampler presents faster convergence than quadratic dependence on n for perfect binary trees and circular graphs. In Appendix A.4, we show that the same non-asymptotic complexity can be obtained under Wasserstein distance.

Proposition 5.5. *For simplicity, we assume η is small enough so that the quadratic term $\frac{\eta^2\beta^2}{(1-\lambda_2)^2\alpha}$ in Theorem 5.4 vanishes. Under Assumption 5.1 and 5.2, to achieve ϵ* TV *distance to π, with an exact sampler for Algorithm 2 and $\eta = \boldsymbol{O}\Big(\frac{\alpha\epsilon^2(1-\lambda_2)}{\beta^2 dn}\Big)$, one needs iterations $K = \tilde{\boldsymbol{O}}\Big(\frac{n^2\beta^2 d}{\alpha^2\epsilon^2(1-\lambda_2)}\Big)$. Here λ_2 is the second largest eigenvalue of the doubly stochastic matrix defined in Assumption 5.2.*

Proof. Please see the proof in Appendix A.3, which follows the same idea in the proof of Proposition 4.4. ☐

Remark 5.6 (Asymptotically convergence). We emphasize that our sampler still converges asymptotically after replacing Assumption 5.1 with mild regularity conditions. As shown in Proposition 5.5, to establish the convergence of the overall sampler, we only need to ensure the convergence of the Gibbs sampler and the convergence of $\bar{\pi}$ towards π with a sufficiently small η. Roberts & Smith (1994) gave the simple conditions under which Gibbs samplers converge asymptotically. The second convergence can be proved by Scheffè's lemma (Scheffé, 1947), as shown in proposition 5.1 in Yuan et al. (2023) and equation 6 in Vono et al. (2022a). Both asymptotic convergences require only mild regularity conditions. Moreover, we demonstrate the convergence in Figure 2b and 2c.

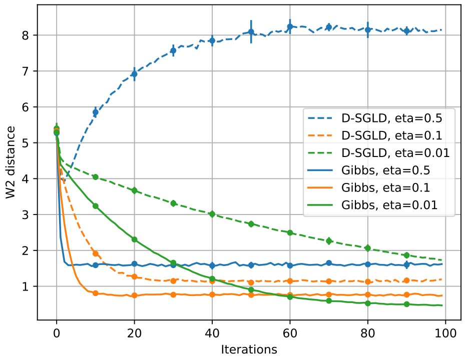
6 Experiments

We compare our sampler with two baselines developed from Langevin dynamics. D-SGLD (Gürbüzbalaban et al., 2021) is the decentralized version of stochastic gradient Langevin dynamics, which proceeds as $x_i^{k+1} = \sum_j \sigma_{ij}x_j^k + \eta\nabla\tilde{f}_i(x_i^k) + \sqrt{2\eta}w_i^{k+1}, w_i^{k+1} \sim \mathcal{N}(\mathbf{0},\mathbf{I}_d)$ where x_i^k is the k-th sample on node i. Note that here η is fixed, while another baseline, D-SGLD with diminishing step sizes (Parayil et al., 2020), utilizes diminishing step sizes and has been proved to operate under weaker assumptions.

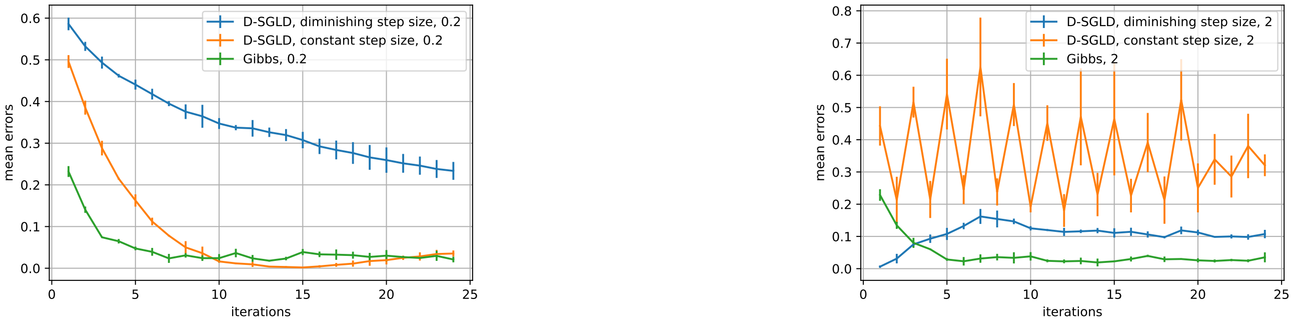
We compare our sampler against D-SGLD using Gaussian targets on perfect binary trees of depth 3. In this case, $\sigma_{ij} = 1/3$ if there is one edge connecting i and j, $\sigma_{ii} = 1 - \sum_j \sigma_{ij}$, and we replace the unbiased estimation of gradients $\nabla\tilde{f}_i(x_i^k)$ by the exact value. The performance is measured by the estimated 2-Wasserstein distance. We also conduct experiments on a more challenging target defined by $\exp(-\sum_{i=1}^4 f_i(x)) \propto$

$\exp(-\|x\|^{1.5} - \|x-0.5\|^{1.5} - \|x-1\|^{1.5} - \|x-1.5\|^{1.5})$ where each minimizer is an all-one vector multiplied by a scale. It is important to note that this target distribution does not satisfy any condition outlined in Assumptions 4.1 and 5.1. Each potential function has distinct minimizers and does not have properties such as strong convexity or smoothness. Due to potential bias when estimating the W_2 distance for this non-Gaussian target, we measure performance using the L2 distance between the estimated mean and the ground truth. We repeat each experiment three times and report the mean and 2-sigma error of measurements. Our sampler demonstrates robust performance even for distributions that do not satisfy the assumptions, and it shows greater stability in terms of initial step size compared to the two Langevin-based samplers.

We highlight the detailed observations. First, the performance of our Gibbs sampler aligns with our qualitative results in Section A.5.1: larger step size yields faster convergence but also leads to more bias. Furthermore, one advantage of our Gibbs sampler is that it converges with any step size, while D-SGLD is highly sensitive to the initial step size: a large step size can result in non-convergence (Figure 2c) or even divergence (Figure 2a). Also, D-SGLD with diminishing step sizes exhibits slow convergence when the initial step size is small or when a large number of iterations is required (Figure 2b).



(a) Gaussian target with varying η



(b) non-Gaussian target, $\eta = 0.2$

(c) non-Gaussian target, $\eta = 2$

Figure 2: Comparison against D-SGLD with fixed step sizes and diminishing step sizes

7 Conclusions

In conclusion, our study introduces a fully distributed sampler using blocked Gibbs sampling, adaptable to various bipartite graph structures without centralized servers. The non-asymptotic analysis provides a theoretical understanding of its behavior, contributing to the theoretical exploration of fully distributed samplers. Exploring the design and analysis of a more efficient augmented distribution represents a promising direction for future research. It is also interesting to explore how to modify our sampler to mitigate practical issues, e.g., communication cost and statistical heterogeneity.

References

Jason M Altschuler and Sinho Chewi. Faster high-accuracy log-concave sampling via algorithmic warm starts. *arXiv preprint arXiv:2302.10249*, 2023.

Krishna Balasubramanian, Sinho Chewi, Murat A Erdogdu, Adil Salim, and Shunshi Zhang. Towards a theory of non-log-concave sampling: first-order stationarity guarantees for Langevin Monte Carlo. In *Conference on Learning Theory*, pp. 2896–2923. PMLR, 2022.

Rémi Bardenet, Arnaud Doucet, and Chris Holmes. On Markov chain Monte Carlo methods for tall data. *Journal of Machine Learning Research*, 18(47), 2017.

Kinjal Bhar, He Bai, Jemin George, and Carl Busart. Asynchronous Local Computations in Distributed Bayesian Learning. *arXiv preprint arXiv:2311.03496*, 2023a.

Kinjal Bhar, He Bai, Jemin George, and Carl Busart. Distributed event-triggered unadjusted Langevin algorithm for Bayesian learning. *Automatica*, 156:111221, 2023b.

John Adrian Bondy, Uppaluri Siva Ramachandra Murty, et al. *Graph theory with applications*, volume 290. Macmillan London, 1976.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Longbing Cao, Hui Chen, Xuhui Fan, Joao Gama, Yew-Soon Ong, and Vipin Kumar. Bayesian Federated Learning: A Survey. *arXiv preprint arXiv:2304.13267*, 2023.

Yongxin Chen, Sinho Chewi, Adil Salim, and Andre Wibisono. Improved analysis for a proximal algorithm for sampling. In *Conference on Learning Theory*, pp. 2984–3014. PMLR, 2022.

Arkabandhu Chowdhury and Christopher Jermaine. Parallel and distributed MCMC via shepherding distributions. In *International Conference on Artificial Intelligence and Statistics*, pp. 1819–1827. PMLR, 2018.

Arnak Dalalyan. Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent. In *Conference on Learning Theory*, pp. 678–689. PMLR, 2017a.

Arnak S Dalalyan. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 79(3):651–676, 2017b.

Arnak S Dalalyan and Avetik Karagulyan. User-friendly guarantees for the Langevin Monte Carlo with inaccurate gradient. *Stochastic Processes and their Applications*, 129(12):5278–5311, 2019.

Arnak S Dalalyan and Lionel Riou-Durand. On sampling from a log-concave density using kinetic Langevin diffusions. 2020.

Daniel A De Souza, Diego Mesquita, Samuel Kaski, and Luigi Acerbi. Parallel MCMC without embarrassing failures. In *International Conference on Artificial Intelligence and Statistics*, pp. 1786–1804. PMLR, 2022.

Kumar Avinava Dubey, Sashank J Reddi, Sinead A Williamson, Barnabas Poczos, Alexander J Smola, and Eric P Xing. Variance reduction in stochastic gradient Langevin dynamics. *Advances in neural information processing systems*, 29, 2016.

Alain Durmus and Eric Moulines. High-dimensional Bayesian inference via the unadjusted Langevin algorithm. 2019.

Jiaojiao Fan, Bo Yuan, and Yongxin Chen. Improved dimension dependence of a proximal algorithm for sampling. In *Conference on Learning Theory*, pp. 1473–1521. PMLR, 2023.

Joseph Gonzalez, Yucheng Low, Arthur Gretton, and Carlos Guestrin. Parallel gibbs sampling: From colored fields to thin junction trees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 324–332. JMLR Workshop and Conference Proceedings, 2011.

Mert Gürbüzbalaban, Xuefeng Gao, Yuanhan Hu, and Lingjiong Zhu. Decentralized stochastic gradient Langevin dynamics and Hamiltonian Monte Carlo. *The Journal of Machine Learning Research*, 22(1): 10804–10872, 2021.

Trevor Halsted, Ola Shorinwa, Javier Yu, and Mac Schwager. A survey of distributed optimization methods for multi-robot systems. *arXiv preprint arXiv:2103.12840*, 2021.

Dušan Jakovetić, Joao Xavier, and José MF Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59(5):1131–1146, 2014.

Yuri Kinoshita and Taiji Suzuki. Improved Convergence Rate of Stochastic Gradient Langevin Dynamics with Variance Reduction and its Application to Optimization. *arXiv preprint arXiv:2203.16217*, 2022.

Alexander Kolesov and Vyacheslav Kungurtsev. Decentralized langevin dynamics over a directed graph. *arXiv preprint arXiv:2103.05444*, 2021.

Nikita Kotelevskii, Maxime Vono, Alain Durmus, and Eric Moulines. Fedpop: A bayesian approach for personalised federated learning. *Advances in Neural Information Processing Systems*, 35:8687–8701, 2022.

Vyacheslav Kungurtsev. Stochastic gradient Langevin dynamics on a distributed network. *arXiv preprint arXiv:2001.00665*, 2020.

Vyacheslav Kungurtsev, Adam Cobb, Tara Javidi, and Brian Jalaian. Decentralized Bayesian learning with Metropolis-adjusted Hamiltonian Monte Carlo. *Machine Learning*, pp. 1–29, 2023.

Yin Tat Lee, Ruoqi Shen, and Kevin Tian. Structured logconcave sampling with a restricted Gaussian oracle. In *Conference on Learning Theory*, pp. 2993–3050. PMLR, 2021.

Chunyuan Li, Changyou Chen, David Carlson, and Lawrence Carin. Preconditioned stochastic gradient Langevin dynamics for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016a.

Mufan Bill Li and Murat A Erdogdu. Riemannian Langevin algorithm for solving semidefinite programs. *arXiv preprint arXiv:2010.11176*, 2020.

Wenzhe Li, Sungjin Ahn, and Max Welling. Scalable MCMC for mixed membership stochastic blockmodels. In *Artificial Intelligence and Statistics*, pp. 723–731. PMLR, 2016b.

Pavel Mach and Zdenek Becvar. Mobile edge computing: A survey on architecture and computation offloading. *IEEE communications surveys & tutorials*, 19(3):1628–1656, 2017.

Angelia Nedic, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

Willie Neiswanger, Chong Wang, and Eric Xing. Asymptotically exact, embarrassingly parallel MCMC. *arXiv preprint arXiv:1311.4780*, 2013.

Felix Otto and Cédric Villani. Generalization of an inequality by Talagrand and links with the logarithmic Sobolev inequality. *Journal of Functional Analysis*, 173(2):361–400, 2000.

Anjaly Parayil, He Bai, Jemin George, and Prudhvi Gurram. Decentralized Langevin dynamics for Bayesian learning. *Advances in Neural Information Processing Systems*, 33:15978–15989, 2020.

Sam Patterson and Yee Whye Teh. Stochastic gradient Riemannian Langevin dynamics on the probability simplex. *Advances in neural information processing systems*, 26, 2013.

Vincent Plassier, Maxime Vono, Alain Durmus, and Eric Moulines. DG-LMC: a turn-key and scalable synchronous distributed MCMC algorithm via Langevin Monte Carlo within Gibbs. In *International Conference on Machine Learning*, pp. 8577–8587. PMLR, 2021.

Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, 187:409–457, 2021.

Hairong Qi, S Sitharama Iyengar, and Krishnendu Chakrabarty. Distributed sensor networks—a review of recent research. *Journal of the Franklin Institute*, 338(6):655–668, 2001.

Gareth O Roberts and Adrian FM Smith. Simple conditions for the convergence of the gibbs sampler and metropolis-hastings algorithms. *Stochastic processes and their applications*, 49(2):207–216, 1994.

Adrien Saumard and Jon A Wellner. Log-concavity and strong log-concavity: a review. *Statistics surveys*, 8: 45, 2014.

Henry Scheffé. A useful convergence theorem for probability distributions. *The Annals of Mathematical Statistics*, 18(3):434–438, 1947.

Jean-Pierre Serre. *Trees*. Springer Science & Business Media, 2002.

Lukang Sun, Adil Salim, and Peter Richtárik. Federated Learning with a Sampling Algorithm under Isoperimetry. *arXiv preprint arXiv:2206.00920*, 2022.

Shengyang Sun, Guodong Zhang, Jiaxin Shi, and Roger Grosse. Functional variational Bayesian neural networks. *arXiv preprint arXiv:1903.05779*, 2019.

Alexander Terenin, Daniel Simpson, and David Draper. Asynchronous gibbs sampling. In *International Conference on Artificial Intelligence and Statistics*, pp. 144–154. PMLR, 2020.

Maxime Vono, Daniel Paulin, and Arnaud Doucet. Efficient MCMC sampling with dimension-free convergence rate using ADMM-type splitting. *The Journal of Machine Learning Research*, 23(1):1100–1168, 2022a.

Maxime Vono, Vincent Plassier, Alain Durmus, Aymeric Dieuleveut, and Eric Moulines. QLSD: Quantised Langevin stochastic dynamics for Bayesian federated learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 6459–6500. PMLR, 2022b.

Xiangyu Wang and David B Dunson. Parallelizing MCMC via Weierstrass sampler. *arXiv preprint arXiv:1312.4605*, 2013.

Xiangyu Wang, Fangjian Guo, Katherine A Heller, and David B Dunson. Parallelizing MCMC with random partition trees. *Advances in neural information processing systems*, 28, 2015.

Chenguang Xi, Van Sy Mai, Ran Xin, Eyad H Abed, and Usman A Khan. Linear convergence in optimization over directed graphs with row-stochastic matrices. *IEEE Transactions on Automatic Control*, 63(10): 3558–3565, 2018.

Minjie Xu, Balaji Lakshminarayanan, Yee Whye Teh, Jun Zhu, and Bo Zhang. Distributed Bayesian posterior sampling via moment sharing. *Advances in neural information processing systems*, 27, 2014.

Tao Yang, Xinlei Yi, Junfeng Wu, Ye Yuan, Di Wu, Ziyang Meng, Yiguang Hong, Hong Wang, Zongli Lin, and Karl H Johansson. A survey of distributed optimization. *Annual Reviews in Control*, 47:278–305, 2019.

Bo Yuan, Jiaojiao Fan, Jiaming Liang, Andre Wibisono, and Yongxin Chen. On a class of gibbs sampling over networks. In Gergely Neu and Lorenzo Rosasco (eds.), *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 5754–5780. PMLR, 12–15 Jul 2023. URL `https://proceedings.mlr.press/v195/yuan23a.html`.

Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

Shunshi Zhang, Sinho Chewi, Mufan Li, Krishna Balasubramanian, and Murat A Erdogdu. Improved discretization analysis for underdamped Langevin Monte Carlo. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 36–71. PMLR, 2023.

Yanling Zheng and Qingshan Liu. A review of distributed optimization: Problems, models and algorithms. *Neurocomputing*, 483:446–459, 2022.

Difan Zou, Pan Xu, and Quanquan Gu. Stochastic gradient Hamiltonian Monte Carlo methods with recursive variance reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

Difan Zou, Pan Xu, and Quanquan Gu. Faster convergence of stochastic gradient Langevin dynamics for non-log-concave sampling. In *Uncertainty in Artificial Intelligence*, pp. 1152–1162. PMLR, 2021.

A Appendix

A.1 Other related works

Proximal Samplers: Recently, the proximal sampler (Lee et al., 2021; Chen et al., 2022) achieved the state-of-the-art probable dimension dependency under different assumptions. Its idea is as follows. To sample from $\exp(-f(x))$, one can sample from $\exp(-f(x) - \frac{1}{2\eta}\|x-y\|^2)$ alternatively between x and y with a Gibbs sampler. Our proposed algorithm can be seen as a generalization of proximal samplers into a distributed setting. Vono et al. (2022a) presents a MCMC algorithm using ADMM-type splitting. It considers sampling from $\exp(-\sum_{i=1}^n f_i(x_i)) - \frac{1}{2\eta}\|x_i - \theta\|^2)$ with a Gibbs sampler and then control the distance between marginals and the target with a small η. Despite having similar concepts to ours, this sampler requires a centralized server for the parameter θ. Combining with the proximal sampler framework, Lee et al. (2021) has proposed a high-accuracy approximate sampling algorithm for the same target $\exp(-\sum_{i=1}^n f_i(x))$. This approximate Metropolis-Hasting algorithm again requires a server node to gather information from every client.

Distributed MCMC with server-client structures: Embarrassingly parallel MCMC (Neiswanger et al., 2013) employs a two-step process where MCMC runs in parallel on data partitions, followed by a server combining the local results, but suffers from misrepresentation of low-density regions. To mitigate embarrassing failures, De Souza et al. (2022) proposes a new combination strategy, leveraging Gaussian Process surrogate modeling and active learning. To reach high-density regions faster, Chowdhury & Jermaine (2018) utilizes an auxiliary shepherding distribution (SD) to control parallel MCMC chains. Recently several works have focused on Bayesian federated learning. Vono et al. (2022b) extends SGLD to the federated learning setting, handling the communication bottleneck using gradient compression. Plassier et al. (2021) introduces "DG-LMC", a novel, scalable, synchronous distributed MCMC algorithm that integrates Langevin Monte Carlo and Gibbs sampling for efficient Bayesian inference in large-scale data analysis. Kotelevskii et al. (2022) addresses uncertainty quantification of personalized federated learning with federated stochastic optimization algorithms.

Distributed Optimization: The typical problem in distributed optimization is to minimize the target function, $\sum_{i=1}^n f_i(x)$ under certain assumptions on each f_i and the underlying communication graphs. A class of classical gradient descent based algorithms (Yuan et al., 2016) on undirected connected graphs can be represented as $x_{k+1}^i = \sum_{j=1}^n a_{ij} x_k^j - \eta_k \nabla f_i(x_k^i)$ where x_k^i is the local estimated variable on client i, $A = (a_{ij})$ is the connection matrix, and η_k is the step size at the k-th iteration. There exists a great number of distributed optimization algorithms that address unconstrained problems, e.g., the alternating direction method of multipliers (Boyd et al., 2011), the fast gradient method (Jakovetić et al., 2014), the push-sum algorithm (Nedic et al., 2017; Xi et al., 2018), and others. We recommend several surveys (Yang et al., 2019; Halsted et al., 2021; Zheng & Liu, 2022) for readers interested in more comprehensive discussions. We emphasize that a majority of distributed optimization algorithms do not rely on centralized servers.

Stochastic Gradient Langevin Dynamics: The centralized Stochastic Gradient Langevin Dynamics (SGLD) can be used to sample from $\exp(-\sum_{i=1}^n f_i(x))$ on the server-client architecture. It requires evaluating unbiased estimators of gradients by mini-batches. This method has sparked a huge number of associated works in sampling and other machine learning areas (Patterson & Teh, 2013; Li et al., 2016a; Dubey et al., 2016; Li et al., 2016b; Bardenet et al., 2017; Li & Erdogdu, 2020). Non-asymptotic bounds in terms of Wasserstein distance for strongly convex and smooth potentials have been established in Dalalyan (2017a); Dalalyan & Karagulyan (2019). It is then suggested to combine SGLD with variance reduction to reduce the variance of stochastic gradients (Zou et al., 2019; 2021). Recently, Kinoshita & Suzuki (2022); Balasubramanian et al. (2022) have established non-asymptotic bounds under weaker assumptions on potentials.

A.2 Proof of improved convergence rates in Theorem 5.3

To begin with, we rewrite the key results from Yuan et al. (2023). The following corresponds to the proof of Theorem 10 and Proposition 16 in it. As discussed, the following lemma directly comes from utilizing the additional condition $\sum_a \sigma_{ab} \le 1, \sum_b \sigma_{ab} \le 1$ we add in this work.

Lemma A.1. *Assume Steps 3 and 4 in Algorithm 2 are exact without biased terms. Denote the distribution of the k-th samples by* ν_k^{AB}*. For the target distribution* $\hat{\pi}^{AB}$ *equation 16, under Assumption 5.1, we have*

$$\mathrm{KL}(\nu_{k+1}^A || \hat{\pi}^A) \le \exp\left(-\eta \int_0^1 c_t \mathrm{d}t\right) \mathrm{KL}(\nu_k^B \| \hat{\pi}^B) \tag{27}$$

where

$$c_t = \left[\frac{\eta t(1-t)}{\min_b \sum_a \sigma_{ab}} + (1-t)^2 \frac{h}{\alpha} + \frac{t^2}{\alpha}\right]^{-1}.$$

Here h is the upper bound of eigenvalues of $H^T H$ *where the entry of H at the a-th row and b-th column is* $\sigma_{ab}/\sum_a \sigma_{ab}$*. In the proof of Proposition 16 in Yuan et al. (2023), this H matrix is denoted as A. We rewrite it to avoid conflicts between notations.*

Based on Lemma A.1, we have the proof of Theorem 5.3.

Proof. We consider the estimation of h in Lemma A.1. As a comparison, Yuan et al. (2023) bounds h by the squared operator norm of H, which follows that

$$\|H\|_{\mathrm{op}}^2 \le |A||B|\|H\|_{\max}^2 \le |A||B|.$$

In contrast, we notice

$$\|H\|_{\mathrm{op}}^2 \le \|H\|_F^2 = \sum_a \sum_b \frac{\sigma_{ab}^2}{(\sum_a \sigma_{ab})^2} \le \sum_a \sum_b \frac{\sigma_{ab}}{(\sum_a \sigma_{ab})} \le |B|.$$

$|A|$ and $|B|$ are the number of nodes in the group A and B, respectively.

This modification leads to the new lower bound for c_t and hence the improved convergence rate shown in Theorem 5.3. The contraction from $\mathrm{KL}(\nu_{k+1}^A || \hat{\pi}^A)$ to $\mathrm{KL}(\nu_{k+1}^B || \hat{\pi}^B)$ can be analyzed in the same way. ☐

A.3 Proof of Proposition 5.5

Proof. The proof is generalized from the proof of Proposition 4.4. With Theorem 5.3, the required iteration to ensure the convergence of Algorithm 2 is $K = \tilde{\boldsymbol{O}}\left(\frac{n}{\eta\alpha}\right)$. Apply our assumption that the quadratic term vanishes to Theorem 5.4, then we have $\mathrm{TV}(\bar{\pi}, \pi) = \boldsymbol{O}\left(\sqrt{\frac{n\beta^2 d\eta}{\alpha(1-\lambda_2)}}\right)$. This implies that to have $\mathrm{TV}(\bar{\pi}, \pi) \le \epsilon/2$, $\eta = \boldsymbol{O}\left(\frac{\alpha\epsilon^2(1-\lambda_2)}{\beta^2 dn}\right)$. Combining all the requirements, we conclude the proof. ☐

A.4 Iteration complexity of Algorithm 2 under Wasserstein distance

In proposition 5.5, we established the interaction complexity when the distance is measured by TV distance. The same complexity can be obtained under Wasserstein distance. Our complexity measured by TV distance is based on Theorem 5.3 and Theorem 5.4. Both theorems can be reformulated with KL divergence: on

Theorem 5.4, this can be done by omitting Pinsker's inequality used in the last step. Since we assume each potential is strongly convex, we can apply Talagrand inequality (Otto & Villani, 2000) on π and $\hat{\pi}$. This inequality claims that the Wasserstein distance is upper bounded by the KL divergence. Hence, the same iteration complexity is followed by the triangle inequality of Wasserstein distance.

A.5 Experiments

A.5.1 Qualitative analysis on a general Tree

To illustrate how our algorithm works for general trees, we conduct experiments on a tree with five nodes shown in Figure 3. Each f_i represents a 5-dimensional Gaussian distribution with a randomly generated mean and covariance matrix. We denote each one as $\boldsymbol{N}(\mu_i, \Sigma_i)$, and our target is $\exp(-\sum_{i=1}^5 f_i(x)) \propto \boldsymbol{N}(\tilde{\mu}, \tilde{\Sigma})$ where $\tilde{\mu} = \tilde{\Sigma}^{-1}(\sum_{i=1}^5 \Sigma_i^{-1}\mu_i)$ and $\tilde{\Sigma}^{-1} = \sum_{i=1}^5 \Sigma_i^{-1}$. All the Gaussian distributions have condition numbers (the ratio of smoothness and strong convexity) ranging from 2 to 5. The initial samples of X_1, X_4, and X_5 are drawn from a standard Gaussian $\boldsymbol{N}(0, \mathbf{I})$ independently. As explained in Algorithm 2, at each step, our sampler enables fully distributed sampling. For instance, clients 2 and 3 can sample from conditionals independently once x_1, x_4, and x_5 are given. Each sampler runs for 400,000 steps, with the first 100,000 steps being the burn-in stage. In Figure 4, we compare the histogram of generated samples and the true density along the first coordinate. In this experiment, we can demonstrate the following points.

- Our sampler can operate on general trees, accommodating scenarios where each node may possess unique strong convexity and smoothness parameters. We point out that a reasonable choice in this case is $\eta = 0.001$ and $K = 400,000$ (See the second column in Figure 4).
- All marginals converge to the same distribution. This observation holds even when all marginals converge to a biased distribution as shown in the first column of Figure 4.
- In practice, we recommend users identify the source of errors and then tune the step size η or iterations K accordingly: reducing η may require more iterations. More specifically, if the value of K is small, alternative diagnostic tools, such as trace plots, can be employed to assess the convergence of the blocked Gibbs sampler. This corresponds to the third column of Figure 4. If η is relatively big, it becomes apparent that all marginals converge to a common distribution, albeit one that is biased toward our target.

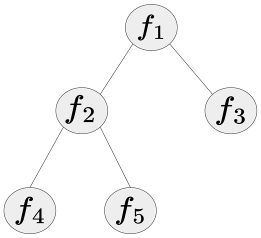


Figure 3: Each local distribution $\exp(-f_i(x))$ is a 5-dimensional Gaussian

A.5.2 Quantitative analysis

In this section, we assume each $\exp(-f_i(x))$ is $\mathcal{N}(\mathbf{1}, \mathbf{I}_d)$, and the initial distribution for each node is $\exp(-f_i(x))$ is $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. Also, following the Metropolis weights, we set $\sigma_{ab} = 1/(\max(h_a, h_b)+1)$ if there is an edge between node a and b. Here, h_a is the degree (the number of neighbors) of node a. For these two network structures considered below, perfect binary trees (with layers more than 2) and circular graphs (each node connects to its two neighbors), the non-zeros value of σ_{ab} is $1/3$. In this section, we repeat each of our experiments under five different random seeds and show the 2-sigma error bar in all figures and tables. It supports that our findings are statistically significant.

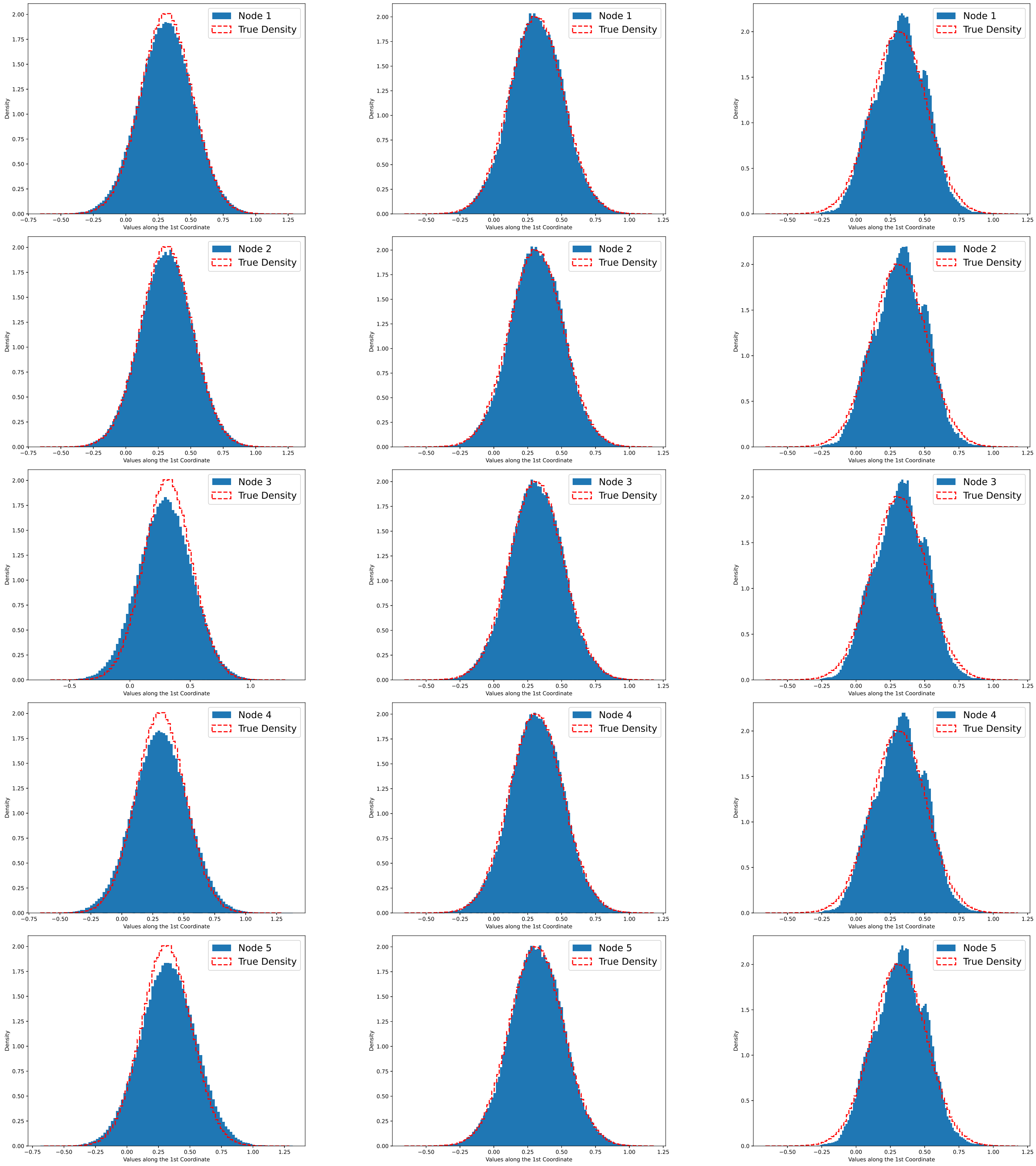
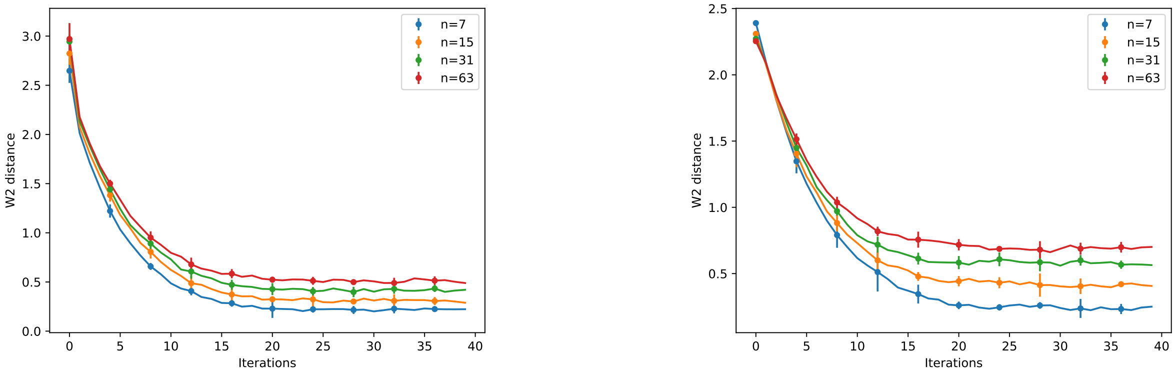


Figure 4: Each column in the figure represents the histogram of values on all five nodes with unique η values. The columns are arranged from left to right, where the η values are 0.01, 0.001, and 0.0001, respectively. In the first column, η is too large that all marginals converge to a biased distribution, while in the third column, η is too small that users need more iterations.

Dependency on the number of nodes n

In Proposition 5.5, we have proved the dependency of the mixing time K on the number of nodes n. Note that it is unclear if this upper bound is tight. To better understand the complexity of n, we conduct experiments



(a) Convergence on perfect binary trees

(b) Convergence on circular graphs

Figure 5: The mean and 2-sigma error bar of estimated W_2 distances under various nodes on trees (Left) and circular graphs (Right)

on two graph structures, perfect binary trees and circular graphs. We run 100 independent chains for 40 iterations and then estimate the empirical mean and covariance matrix at each iteration. Since the target distribution is Gaussian, we compute the W_2 distance with the explicit formula of W_2 between two Gaussian distributions.

In Figure 5, we run our sampler with layers ranging from 3 to 6 corresponding to $n = 7, 15, 31$, and 63, respectively. Then, we repeat our experiments on circular graphs with the same n. These figures show our sampler converges even when the graph is sparse and the number of nodes is relatively large. We further analyze the curves from two perspectives: the slope after the logarithm transformation and the minimal distance along a chain. By Theorem 5.3, the estimated slope after applying logarithm transformation is the convergence rate of the Gibbs sampler. In Table 1, we observe that the convergence rate is small for large n. Moreover, it may indicate a faster convergence rate than linear rates on n. We also estimate the biased term discussed in Theorem 5.4 by the minimal W_2 over each chain in Table 2. Again, we observe that the biased term is large for graphs with more nodes as indicated in Theorem 5.4. Furthermore, the coefficient of determination (R^2) for fitting the distances and $\sqrt{n}$ is approximately 0.940; this may indicate that the dependency in this setting aligns with our theoretical analysis in Theorem 5.4.

Table 1: Estimated slopes represented by the mean and 2-sigma error

Graphs	$n = 7$	$n = 15$	$n = 31$	$n = 63$
Perfect binary trees	-0.054 ± 0.003	-0.047 ± 0.002	-0.039 ± 0.002	-0.034 ± 0.001
Circular graphs	-0.056 ± 0.003	-0.038 ± 0.002	-0.028 ± 0.002	-0.024 ± 0.001

Table 2: Minimal distances represented by the mean and 2-sigma error

Graphs	$n = 7$	$n = 15$	$n = 31$	$n = 63$
Perfect binary trees	0.170 ± 0.043	0.270 ± 0.020	0.376 ± 0.024	0.457 ± 0.035
Circular graphs	0.187 ± 0.013	0.367 ± 0.016	0.526 ± 0.015	0.637 ± 0.025