Parallelising Glauber Dynamics

Holden Lee

Johns Hopkins University

Joint Math Meetings 2024
http://www.arxiv.org/abs/2307.07131

http://tiny.cc/pigd

Parallel Ising Glauber Dynamics

Holden Lee

Johns Hopkins University

Joint Math Meetings 2024

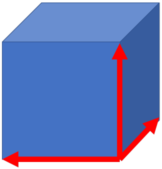
http://www.arxiv.org/abs/2307.07131

http://tiny.cc/pigd

Glauber dynamics

Problem: Approximately sample from

$$\mu(x) \propto e^{f(x)} \text{ on } \prod_{i=1}^{n} \Omega_i \quad (\text{e.g., } \{\pm 1\}^n).$$



Method (MCMC): Run **Glauber dynamics**. Given X_t,

- Select coordinate: $i \sim \text{Uniform}([n])$.
- Resample coordinate: $X_i | X_{-i}$, i.e.,

$$X_{t+1,i} = z \text{ with probability } \mu(X_i = z | X_{\sim i} = x_{\sim i}) = \frac{e^{f(x_{i \leftarrow z})}}{\sum_{z' \in \Omega_i} e^{f(x_{i \leftarrow z'})}}.$$

- μ is **stationary distribution** (preserved under Markov chain).

For "nice" μ, we have **rapid mixing**: letting $\mu_T = \text{Law}(X_T)$,

$$t_{\text{mix}}(\varepsilon) := \min \{ T : \mathrm{TV}(\mu_T, \mu) \leq \varepsilon \} = O\left(n \ln \left(\frac{n}{\varepsilon} \right) \right).$$

Parallelising Glauber dynamics

Question: Glauber dynamics is sequential. Can we do better with parallel computation?

Natural idea: Resample k coordinates at a time.

1. k-Glauber dynamics mixes k times as fast.

Problem: How can we implement one step of k-Glauber?

2. Parallel algorithm for sampling from Ising (& p-spin) model.

Related work

1. Various algorithms in $\mathbb{R}^n$ for sampling from log-concave distributions using the gradient take $o(n)$ steps; randomized midpoint method is fully parallelisable (RNC) [Shen and Lee, 2019]
2. Fast parallel algorithms under Dobrushin conditions [Feng et al., 2021, Liu and Yin, 2022]
   - Gives an alternate approach for the Ising model, but not the p-spin model
3. Fast parallel algorithms (RNC) when fast counting algorithms exist [Anari et al., 2023a, Anari et al., 2023b]

Our focus: Get a fast parallel algorithm

1. in discrete setting
2. under general conditions for mixing
3. without fast parallel counting algorithms

Outline

1. k-Glauber mixes k times as fast

2. Parallel algorithm for Ising model

k-Glauber dynamics

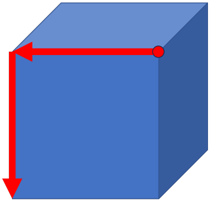
k-Glauber dynamics: Given x_t,

- Select random *subset* $S \subset [n]$, $|S| = k$.
- Resample subset $x_S | x_{S^c}$. $X_{t+1,S} = z$ with probability

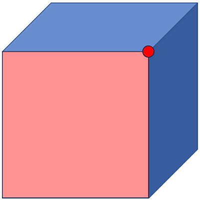
$$\mu(X_S = z | X_{S^c} = x_{S^c}) = \frac{e^{f(x_{S\leftarrow z})}}{\sum_{z' \in \{\pm 1\}^S} e^{f(x_{S\leftarrow z'})}}.$$

k-Glauber $\neq$ k steps of Glauber...

2 steps of Glauber



2-Glauber



...but intuitively it should only be better!

Down-up walks

A **Markov kernel** $K : A \rightsquigarrow B$ is given by a transition matrix $\mathbb{R}^{A\times B}$ with

$$K_{ab} = \text{probability of going from } a \text{ to } b.$$

For a measure μ on A, μK is the measure on B after applying K.
Given a distribution μ on $\binom{[n]}{k}$, $\ell < k$, define...

Down operator	**Up operator**
$D_{k\to\ell} : \binom{[n]}{k} \rightsquigarrow \binom{[n]}{\ell}$	$U_{\ell\to k} : \binom{[n]}{\ell} \rightsquigarrow \binom{[n]}{k}$
Choose a uniform random subset of A of size ℓ	Choose a random superset of B of size k, with probability $\mu(A\vert A \supset B)$.
$D_{k\to\ell}(A, B) = \mathbb{1}_{B\subset A} \frac{1}{\binom{k}{\ell}}$	$U_{\ell\to k}(B, A) = \mathbb{1}_{B\subset A} \frac{\mu(A)}{\sum_{A'\supset B} \mu(A')}$

Given $\mu = \mu_k$, let $\mu_\ell = \mu_k D_{k\to\ell}$.

Realizing Glauber dynamics as down-up walk

Let μ be a measure on $\{\pm 1\}^n$.
The **homogenization** of μ is the measure over $\binom{[n]\times\{\pm 1\}}{n}$ where

$$(x_1, \ldots, x_n) \text{ is identified with } \{(x_1, 1) \ldots, (x_n, n)\}.$$

1	-1	1	1	-1

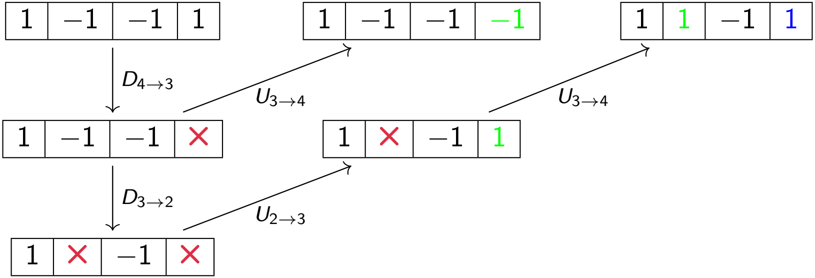
1	1	1	1	1
-1	-1	-1	-1	-1

Realizing Glauber dynamics as down-up walk

- $D_{k\to\ell}$: Erase $k-\ell$ coordinates.
- $U_{\ell\to k}$: Restore $k-\ell$ coordinates, using conditional distribution.

Lemma

$$P_{Glauber} = P^{\nabla}_{n\leftrightarrow n-1} = D_{n\to n-1}U_{n-1\to n}$$
$$P_{k\text{-}Glauber} = P^{\nabla}_{n\leftrightarrow n-k} = D_{n\to n-1}\cdots D_{n-k+1\to n-k}U_{n-k\to n-k+1}\cdots U_{n-1\to n}.$$



General framework: Diffusion models

A general way to form a Markov chain is by *adding noise* and *denoising* (using the posterior) [Montanari, 2023].

- **Discrete setting** $\{\pm 1\}^n$:
  Noise: Erase coordinates.

1 step	k steps
Glauber	k-Glauber

- **Continuous setting** $\mathbb{R}^n$:
  Noise: Add Gaussian $N(0, t)$.

$t \to 0$	$t > 0$
Langevin dynamics	Proximal sampler [Lee et al., 2021]

Proximal sampler gives best known rates ($\widetilde{O}(\sqrt{n})$) for high-accuracy log-concave sampling [Fan et al., 2023].

Denoising process is also used in machine learning for generative modeling [Sohl-Dickstein et al., 2015, Song and Ermon, 2019, Song et al., 2020].

Mixing for Markov chains

Often measure closeness of distributions in χ^2 or $\mathcal{D}_{\mathrm{KL}}$-divergence:

$$\chi^2(\nu\|\mu) = \int \left(\frac{d\nu}{d\mu} - 1\right)^2 d\mu \qquad \mathcal{D}_{\mathrm{KL}}(\nu\|\mu) = \int \frac{d\nu}{d\mu} \ln \frac{d\nu}{d\mu}\, d\mu.$$

These are examples of f-divergences $D_f(\nu\|\mu) = \int f\left(\frac{d\nu}{d\mu}\right) d\mu$.

Definition

$P : A \rightsquigarrow B$ satisfies $(1-\kappa)$-**contraction in f-divergence** w.r.t. μ if

$$D_f(\nu P\|\mu P) \le (1-\kappa) D_f(\nu\|\mu).$$

If $P : A \rightsquigarrow A$ with stationary distribution μ, we have exponential convergence of Markov chain:

$$D_f(\nu P^t\|\mu) \le (1-\kappa)^t D_f(\nu\|\mu).$$

Mixing for Markov chains

Definition

$P : A \rightsquigarrow B$ satisfies $(1-\kappa)$-**contraction in** f-**divergence** w.r.t. μ if

$$D_f(\nu P \| \mu P) \le (1-\kappa) D_f(\nu \| \mu).$$

If $P : A \rightsquigarrow A$ with stationary distribution μ, we have exponential convergence of Markov chain:

$$D_f(\nu P^t \| \mu) \le (1-\kappa)^t D_f(\nu \| \mu).$$

- For χ^2, equivalent to **spectral gap** (κ) or **Poincaré inequality** $(1/\kappa)$.
- For $\mathcal{D}_{\mathsf{KL}}$, for $D_{n \to n-1}$ called **approximate tensorization of entropy**, closely related to **log-Sobolev inequality**.

For Glauber dynamics, **rapid mixing** when $\kappa = \Omega\left(\frac{1}{n}\right)$: get constant contraction after $O(n)$ steps.

Main theorem

Theorem (k-Glauber mixes k times faster)

Let μ be a distribution on $\Omega = \prod_{i=1}^n \Omega'_i$, $1 \le k \le n$, and $C \ge 1$. If $D_{n \to n-1}$ satisfies $(1 - \frac{1}{Cn})$-contraction in χ^2 or $\mathcal{D}_{\mathrm{KL}}$-divergence, then $P_{k\text{-Glauber}}$ satisfies $(1 - \Omega\left(\frac{k}{Cn}\right))$-contraction in χ^2 or $\mathcal{D}_{\mathrm{KL}}$-divergence.

Alternative phrasing:

1. If P_{Glauber} has Poincaré constant Cn then $P_{k\text{-Glauber}}$ has Poincaré constant $O\left(\frac{Cn}{k}\right)$.
2. If μ satisfies C-approximate tensorization of entropy, then μ satisfies $O\left(\frac{C}{k}\right)$-approximate k-uniform block factorization of entropy.

Main theorem

Theorem (k-Glauber mixes k times faster)

Let μ be a distribution on $\Omega = \prod_{i=1}^n \Omega_i'$, $1 \le k \le n$, and $C \ge 1$.
If $D_{n\to n-1}$ satisfies $(1 - \frac{1}{Cn})$-contraction in χ^2 or $\mathcal{D}_{\mathrm{KL}}$-divergence, then $P_{k\text{-Glauber}}$ satisfies $(1 - \Omega\left(\frac{k}{Cn}\right))$-contraction in χ^2 or $\mathcal{D}_{\mathrm{KL}}$-divergence.

Given contraction of $D_{n\to n-1}$ in

$$P_{\mathrm{Glauber}} = P^{\triangledown}_{n\leftrightarrow n-1} = D_{n\to n-1} U_{n-1\to n},$$

how to show (k times as much) contraction of

$$P_{k\text{-Glauber}} = P^{\triangledown}_{n\leftrightarrow n-k} = \underbrace{D_{n\to n-1} \cdots D_{n-k+1\to n-k}}_{k} U_{n-k\to n-k+1} \cdots U_{n-1\to n}?$$

Sufficient to show: for any $m < n$, $D_{m\to m-1}$ is "at least as contractive" as $D_{n\to n-1}$.

Proof sketch

Need to show: for any $m < n$, $D_{m\to m-1}$ is "at least as contractive" as $D_{n\to n-1}$.

Idea: Write $D_{m\to m-1}$ as projection of $D_{n\to n-1}$ tensorized with noise!

- **Tensorization** with noise slightly degrades contraction.
- **Projection** only improves contraction.

Bernoulli-Laplace model: For $\mu = \text{Uniform}\binom{[n]}{k}$, suppose $D_{k\to k-1}$ has contraction $\kappa_{\text{BL},k}$ (known, [Salez, 2021]). Suffices to show:

Lemma

Let κ_k be the contraction of $D_{k\to k-1}$ w.r.t. μ_k. Then $\kappa_k \geq \frac{n\kappa_n\kappa_{\text{BL},k}}{n\kappa_n + k\kappa_{\text{BL},k}}$.

Tensorization

Proposition

Let $\kappa(\cdot)$ denote contraction in χ^2 or $\mathcal{D}_{\mathrm{KL}}$. Given kernels

$$P_1 : \Omega_1 \rightsquigarrow \Omega_1' \qquad\qquad P_2 : \Omega_2 \rightsquigarrow \Omega_2',$$

define

$$P = p(P_1 \otimes I_2) + (1-p)(I_1 \otimes P_2) : \Omega_1 \times \Omega_2 \rightsquigarrow \Omega_1' \times \Omega_2 \sqcup \Omega_1 \times \Omega_2'$$

$$\begin{tikzcd}
 & \Omega_1 \times \Omega_2 \arrow[dl, squiggly, "p"'] \arrow[dr, squiggly, "1-p"] & \\
\Omega_1' \times \Omega_2 & & \Omega_1 \times \Omega_2'
\end{tikzcd}$$

Then

$$\kappa(P) \ge \min\{p\kappa_1, (1-p)\kappa_2\}.$$

When $\Omega_i = \Omega_i'$ this is the product Markov chain.

Projection

Proposition

Given kernels making the following diagram commute:

$$\begin{tikzcd} (\Omega_1, \mu_1) \arrow[r, squiggly, "P"] \arrow[d, "\pi_1"] & (\Omega_2, \mu_2) \arrow[d, "\pi_2"] \\ (\Omega_1', \mu_1') \arrow[r, squiggly, "P'"] & (\Omega_2', \mu_2') \end{tikzcd}$$

Then

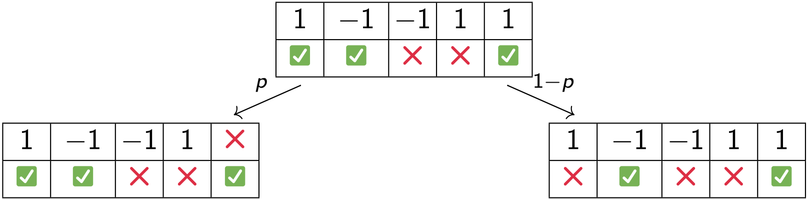
$$\kappa(P') \geq \kappa(P).$$

- **Tensorization.** Let $\Omega = \{\pm 1\} \times [n]$, $p = \frac{\kappa_{\mathrm{BL},k}}{\kappa_n + \kappa_{\mathrm{BL},k}}$

$$P = p(D_{n\to n-1} \otimes I_{\binom{[n]}{k}}) + (1-p)(I_{\binom{\Omega}{n}} \otimes D_{k\to k-1})$$

$$\binom{\Omega}{n} \times \binom{[n]}{k} \rightsquigarrow \binom{\Omega}{n-1} \times \binom{[n]}{k} \cup \binom{\Omega}{k} \times \binom{[n]}{k-1}$$

$$\kappa(P)^{-1} \le \kappa_n^{-1} + \kappa_{\mathrm{BL},k}^{-1}.$$



- **Projection** (only keep ✅ coordinates) is

$$P' = \left(1 - \frac{p(n-k)}{n}\right) D_{k\to k-1} + \frac{p(n-k)}{n} I$$

$$\binom{\Omega}{k} \rightsquigarrow \binom{\Omega}{k-1} \cup \binom{\Omega}{k}.$$

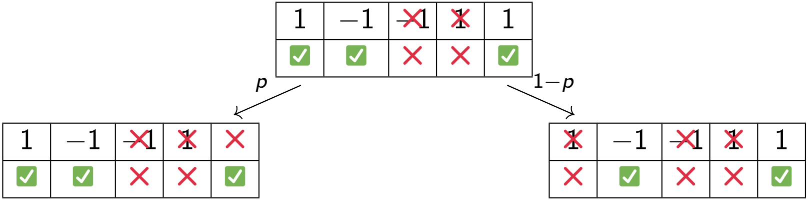
  - Probability $\frac{p(n-k)}{n}$ of trying to remove already-removed coordinate.
  - Solve for κ_k.

- **Tensorization.** Let $\Omega = \{\pm 1\} \times [n]$, $p = \frac{\kappa_{\mathrm{BL},k}}{\kappa_n + \kappa_{\mathrm{BL},k}}$

$$P = p(D_{n\to n-1} \otimes I_{\binom{[n]}{k}}) + (1-p)(I_{\binom{\Omega}{n}} \otimes D_{k\to k-1})$$

$$\binom{\Omega}{n} \times \binom{[n]}{k} \rightsquigarrow \binom{\Omega}{n-1} \times \binom{[n]}{k} \cup \binom{\Omega}{k} \times \binom{[n]}{k-1}$$

$$\kappa(P)^{-1} \le \kappa_n^{-1} + \kappa_{\mathrm{BL},k}^{-1}.$$



- **Projection** (only keep ✅ coordinates) is

$$P' = \left(1 - \frac{p(n-k)}{n}\right) D_{k\to k-1} + \frac{p(n-k)}{n} I$$

$$\binom{\Omega}{k} \rightsquigarrow \binom{\Omega}{k-1} \cup \binom{\Omega}{k}.$$

  - Probability $\frac{p(n-k)}{n}$ of trying to remove already-removed coordinate.
  - Solve for κ_k.

Outline

1. k-Glauber mixes k times as fast
2. Parallel algorithm for Ising model

Sampling from the Ising model

Definition

The **Ising model** with interaction matrix $J \in \mathbb{R}^{n\times n}$ and external field $h \in \mathbb{R}^n$ is the probability distribution on $\{\pm 1\}^n$ given by

$$\mu_{J,h}(x) \propto \exp\left(\frac{1}{2}\langle x, Jx\rangle + \langle h, x\rangle\right), \quad x \in \{\pm 1\}^n.$$

- Model for interacting particles in statistical physics.
  - $J_{ij} > 0$ incentivizes $x_i = x_j$: **ferromagnetic** interaction.
  - $J_{ij} < 0$ incentivizes $x_i = -x_j$: **antiferromagnetic** interaction.
- Connections to TCS, (Bayesian) statistics/machine learning,...
- When J is *dense*, cannot update different coordinates independently.

Sampling from the Ising model

Glauber dynamics mixes rapidly under weak interactions.

Theorem ([Anari et al., 2021])

Let $J \in \mathbb{R}^{n\times n}$ be a symmetric matrix satisfying $0 \preceq J \prec I_n$. Then for any $\varepsilon > 0$, Glauber dynamics mixes to ε in total variation distance in

$$O\left(\frac{n\log\left(\frac{n}{\varepsilon}\right)}{1-\|J\|_{op}}\right) \textit{ steps.}$$

- Their techniques can be used to show **approximate tensorization of entropy**.
- So if we could apply k-Glauber, then we get factor of k parallel speedup.
  - We will take $k = \widetilde{\Theta}\left(\frac{n}{\|J\|_F}\right)$.

Parallel Sampling from the Ising model

Theorem

Fix $c > 0$. With appropriate choice of constants depending only on c, if J is symmetric PSD with $\|J\| \le 1 - c$, then **ParallelIsingSampler** *outputs a sample ε-close in TV distance from $\mu_{J,h}$ and, with probability $\ge 1 - \varepsilon$, runs in time*

$$O\left(\max\{\|J\|_F, 1\} \operatorname{poly}\log\left(\frac{n}{\varepsilon}\right)\right)$$

on a parallel machine with $\operatorname{poly}(n)$ *processors.*

Since $\|J\|_F \le \sqrt{n}$, this is a $\widetilde{\Theta}\left(\frac{n}{\|J\|_F}\right) = \widetilde{\Omega}(\sqrt{n})$ speedup.

Algorithm 1 Parallel Ising Sampler (**ParallelIsingSampler**)

Input: $J \in \mathbb{R}^{R\times R}$ ($|R| \le n$), $h \in \mathbb{R}^R$, $\varepsilon \in (0, \frac{1}{2})$.
if $\left\|J^{\obslash}\right\|_F \le c_3 / \ln\left(\frac{n}{\varepsilon}\right)$ **then** ($\obslash$ means zero out diagonal)
    Approximate rejection sample (ARS) using $\nu(x) \propto e^{\langle h+\widehat{h}, x\rangle}$ where $\widehat{h}$ solves $\mathbb{E}_{\mu_{h+\widehat{h}}} J^{\obslash} x = \widehat{h}$.
else
    Initialize $y \sim \nu_0(x) \propto e^{\langle h,x\rangle}$.
    for t from 1 to $\Theta\left(\operatorname{poly}\log\left(\frac{n}{\varepsilon}\right)\|J\|_F\right)$ **do**
        Choose $S \subseteq R$ a random subset of size $\widetilde{\Theta}\left(\frac{|R|}{\|J\|_F}\right)$.
        Let $y_S \hookleftarrow$ **ParallelIsingSampler**$(J_S, J_{S\times R\setminus S}\, y_{R\setminus S} + h_S, \varepsilon)$.
    end for
end if

If ARS works perfectly, then achieves good error by **rapid mixing for Ising model** and **speedup of k-Glauber**. Remains to show:

1. Approximate rejection sampling has small error when $\|J_{R\times R}\|_F$ small.
2. Recursion is subcritical.

1. Approximate rejection sampling

Algorithm 2 Approximate rejection sampler (**ApproxRejectionSampler**)

1: **Input:** Oracle sampler for Q, function g s.t. $\frac{dP}{dQ} \propto e^g$, parameter c.
2: **repeat**
3: Draw $X, Z \sim Q$.
4: Let $R = \exp(g(X) - g(Z))$.
5: **until** $U \le \frac{1}{c} R$ where $U \sim \text{Uniform}([0, 1])$

Theorem (Hanson-Wright Inequality)

For X with independent, mean-0, K-subgaussian coordinates, $A \in \mathbb{R}^{n \times n}$,

$$\mathbb{P}\left(\left| \langle X, AX \rangle - \mathbb{E}\langle X, AX \rangle \right| \ge t\right) \le 2 \exp\left[-c \min\left\{ \frac{t^2}{K^4 \|A\|_F^2}, \frac{t}{K^2 \|A\|} \right\}\right].$$

When $\left\|J^{\not\ni}\right\|$ small, take $Q(x) \propto e^{\langle h, x \rangle}$, $g(x) = \frac{1}{2}\langle x, Jx \rangle$?

1. Approximate rejection sampling

Algorithm 2 Approximate rejection sampler (**ApproxRejectionSampler**)

Input: Oracle sampler for Q, function g s.t. $\frac{dP}{dQ} \propto e^g$, parameter c.
repeat
  Draw $X, Z \sim Q$.
  Let $R = \exp(g(X) - g(Z))$.
until $U \le \frac{1}{c} R$ where $U \sim \text{Uniform}([0,1])$

Theorem (Hanson-Wright Inequality)

For X with independent, mean-0, K-subgaussian coordinates, $A \in \mathbb{R}^{n\times n}$,

$$\mathbb{P}\left(\left|\langle X, AX\rangle - \mathbb{E}\langle X, AX\rangle\right| \ge t\right) \le 2\exp\left[-c\min\left\{\frac{t^2}{K^4\|A\|_F^2}, \frac{t}{K^2\|A\|}\right\}\right].$$

When $\left\|J^{\oslash}\right\|$ small, take $Q(x) \propto e^{\langle h,x\rangle}$, $g(x) = \frac{1}{2}\langle x, Jx\rangle$? ✗

1. Approximate rejection sampling

Theorem ([Sambale and Sinulis, 2019])

For $X \in \{\pm 1\}^n$ with independent coordinates and $g : \{\pm 1\}^n \to \mathbb{R}$,

$$\mathbb{P}(|g - \mathbb{E}g| \geq t) \leq 2\exp\left[-c\min\left\{\frac{t^2}{\mathbb{E}[\|\nabla g\|^2]}, \frac{t}{\max_{x\in\{\pm 1\}^n}\|\nabla^2 g\|_F}\right\}\right].$$

The discrete gradient is defined by $(\nabla g)_i(x) = \frac{1}{2}[g(x_{i\leftarrow 1}) - g(x_{i\leftarrow -1})]$.

$$Q = \mu_{h+\widehat{h}} = e^{\langle h+\widehat{h}, x\rangle}$$

$$g = \ln\frac{dP}{dQ}(+\text{const.}) = \frac{1}{2}\langle x, Jx\rangle - \left\langle \widehat{h}, x\right\rangle$$

To make $\mathbb{E}[\|\nabla g\|^2]$ small, need ∇g to be centered.

$$\mathbb{E}_Q \nabla g = 0 \iff \mathbb{E}_{\mu_{h+\widehat{h}}} J^{\not\circ} x = \widehat{h} \iff J^{\not\circ}\tanh(h+\widehat{h}) = \widehat{h}.$$

Can solve approximately with fixed point iteration.

1. Approximate rejection sampling

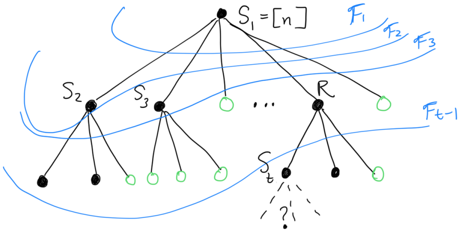
Algorithm 2 Approximate rejection sampler (**ApproxRejectionSampler**)

1: **Input:** Oracle sampler for Q, function g s.t. $\frac{dP}{dQ} \propto e^g$, parameter c.
2: **repeat**
3: Draw $X, Z \sim Q$.
4: Let $R = \exp(g(X) - g(Z))$.
5: **until** $U \le \frac{1}{c} R$ where $U \sim \text{Uniform}([0,1])$

- For small enough $\|A\| \le \|A\|_F = O(1)$,
  - $g(X) = \langle X, AX \rangle - \left\langle \widehat{h}, X \right\rangle$ has exponential tail (with large enough constant).
  - $R = e^{g(X) - g(Z)}$ has power-law tail (large enough power).
- If $\widehat{P}$ is output of **ApproxRejectionSampler**,
  - $\mathrm{TV}(\widehat{P}, P) \le \frac{\mathbb{E}[(R-c)\mathbb{1}_{R \ge c}]}{\mathbb{E}R}$ (tail of expectation)
  - acceptance probability is $\ge \frac{1}{2c}$.
- Taking power to be $\ln\left(\frac{n}{\varepsilon}\right)$, we get $\frac{\varepsilon}{n}$ error.

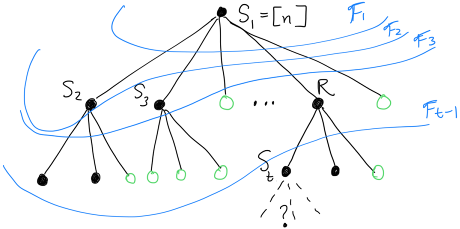
2. Analyzing the recursion

Need to bound total number of vertices in recursion tree.
Node: Call of **ParallelIsingSampler**.
Leaf: Call to **ApproxRejectionSampler**.



- Tree depends on subsets chosen, not visited states $y \in \{\pm 1\}^n$.
- Expand one node at a time. Let $\mathcal{F}_t$ = information revealed up to time t.
- Label nodes with subset S_t. Let D_t = number of children of S_t.
- Fix t. Let R be parent of S_t, $|R| = m$, $|S_t| = s = \frac{c_1 m}{\ln\left(\frac{n}{\varepsilon}\right)\|J_{R\times R}\|_F}$. If we recurse further, # children is $T = O(\ln\left(\frac{n}{\varepsilon}\right)\frac{m}{s})$.

Analyzing the recursion



$|R| = m$, $|S_t| = s = \frac{c_1 m}{\ln\left(\frac{n}{\varepsilon}\right)\|J_{R\times R}\|_F}$, $T = O(\ln\left(\frac{n}{\varepsilon}\right)\frac{m}{s})$, $D_t = \#$ children of S_t.

$$\mathbb{E}[D_t|\mathcal{F}_{t-1}] \lesssim \mathbb{E}\left[\frac{\ln\left(\frac{n}{\varepsilon}\right)^2}{c_1}\|J_{S_t\times S_t}\|_F \mathbb{1}[\|J_{S_t\times S_t}\|_F > c|\mathcal{F}_{t-1}]\right]$$

$$= O\left(\frac{1}{c_1}\ln\left(\frac{n}{\varepsilon}\right)^2\mathbb{E}[\|J_{S_t\times S_t}\|_F^2|\mathcal{F}_{t-1}]\right) \quad \text{(Cauchy-Schwarz)}$$

$$\mathbb{E}\left[\|J_{S_t\times S_t}\|_F^2|\mathcal{F}_{t-1}\right] = \mathbb{E}_{S\sim\mathrm{Uniform}\binom{R}{s}}\left[\|J_{S\times S}\|_F^2\right] = \left(\frac{s}{m}\right)^2\|J_{S\times S}\|_F^2 \le \frac{c_1^2}{\ln\left(\frac{n}{\varepsilon}\right)^2}$$

$$\mathbb{E}[D_t|\mathcal{F}_{t-1}] = O(c_1).$$

Analyzing the recursion

$|R| = m$, $|S_t| = s = \frac{c_1 m}{\ln\left(\frac{n}{\varepsilon}\right)\|J_{R\times R}\|_F}$, $T = O(\ln\left(\frac{n}{\varepsilon}\right)\frac{m}{s})$, $D_t = \#$ children of S_t.

$$\mathbb{E}[D_t|\mathcal{F}_{t-1}] \lesssim \mathbb{E}\left[\frac{\ln\left(\frac{n}{\varepsilon}\right)^2}{c_1}\|J_{S_t\times S_t}\|_F \mathbb{1}[\|J_{S_t\times S_t}\|_F > c|\mathcal{F}_{t-1}]\right]$$

$$= O\left(\frac{1}{c_1}\ln\left(\frac{n}{\varepsilon}\right)^2 \mathbb{E}[\|J_{S_t\times S_t}\|_F^2 |\mathcal{F}_{t-1}]\right) \quad \text{(Cauchy-Schwarz)}$$

$$\mathbb{E}\left[\|J_{S_t\times S_t}\|_F^2 |\mathcal{F}_{t-1}\right] = \mathbb{E}_{S\sim\text{Uniform}\binom{R}{s}}\left[\|J_{S\times S}\|_F^2\right] = \left(\frac{s}{m}\right)^2 \|J_{S\times S}\|_F^2 \le \frac{c_1^2}{\ln\left(\frac{n}{\varepsilon}\right)^2}$$

$$\mathbb{E}[D_t|\mathcal{F}_{t-1}] = O(c_1).$$

- **Key:** When take subsets of R of size $p|R|$, expected Frobenius norm is $O(p^2)$ but number of steps is only $O\left(\frac{1}{p}\right)$.
- For $c_1 \ll 1$, $\mathbb{E}[D_t|\mathcal{F}_{t-1}] < 1$, this is a **subcritical branching process**.
- **Martingale concentration**: w.h.p. number of vertices (runtime) is bounded by poly log $\left(\frac{n}{\varepsilon}\right)\|J\|_F$.

Extension: Mixed p-spin model

Definition

The **mixed p-spin model** with coefficients $\beta_2, \beta_3, \ldots$ is the random measure

$$\mu(x) \propto \exp\left(\sum_{p=2}^{\infty} \frac{\beta_p}{n^{\frac{p-1}{2}}} \sum_{1\le i_1\le\cdots\le i_p} g_{i_1,\cdots,i_p} x_{i_1}\cdots x_{i_p} + \sum_{i=1}^{n} h_i x_i\right), \quad x \in \{\pm 1\}^n$$

where $g_{i_1,\cdots,i_p} \sim N(0,1)$.

Theorem ([Adhikari et al., 2022, Anari et al., 2023c])

There is an absolute constant A such that if $\sum_{p=2}^{\infty} \sqrt{p^3 \ln p} \cdot \beta_p \le A$ and $\sum_{p=2}^{\infty} \sqrt{2^p p^3 \ln p} \cdot \beta_p = B < \infty$, when with probability $\ge 1 - \exp(-\Omega(n))$ over g, μ satisfies approximate tensorization of entropy with constant $O_B(1)$.

Extension: Mixed p-spin model

Theorem ([Adhikari et al., 2022, Anari et al., 2023c])

There is an absolute constant A such that if $\sum_{p=2}^{\infty} \sqrt{p^3 \ln p} \cdot \beta_p \le A$ and $\sum_{p=2}^{\infty} \sqrt{2^p p^3 \ln p} \cdot \beta_p = B < \infty$, when with probability $\ge 1 - \exp(-\Omega(n))$ over g, μ satisfies approximate tensorization of entropy with constant $O_B(1)$.

By using concentration of polynomials rather than quadratics, we get:

Theorem

In the above setting, w.h.p. there is an algorithm which outputs a sample ε-close in TV distance from μ and, with probability $\ge 1 - \varepsilon$, runs in time

$$O\left(\sqrt{n}\,\mathrm{poly}\log\left(\frac{n}{\varepsilon}\right)\right)$$

on a parallel machine with $\mathrm{poly}\left(\frac{n}{\varepsilon}\right)$ *processors.*

Conclusion and open questions

1. k-Glauber dynamics gives a generic parallel speedup for discrete Markov chains, *if* we can implement each step.
2. Implemented for Ising (& p-spin) model in the regime of rapid mixing.

Open questions

- Fast parallel sampling under generic smoothness assumptions?

Theorem ([Anari et al., 2023c])

There is an absolute constant $A > 0$ such that for $\mu(x) \propto e^{H(x)}$, $\beta := \max_{x\in\{\pm 1\}^n} \left\|\nabla^2 H(x)\right\|_2 \le A$, then μ has spectral gap $\ge \frac{1}{(1+O(\beta))n}$.

- Analysis of "gradient-based" discrete sampling algorithms [Grathwohl et al., 2021, Zhang et al., 2022, Rhodes and Gutmann, 2022].
- Other settings where k-Glauber dynamics can be efficiently approximated?

Bibliography I

Adhikari, A., Brennecke, C., Xu, C., and Yau, H.-T. (2022).
Spectral gap estimates for mixed p-spin models at high temperature.
arXiv preprint arXiv:2208.07844.

Anari, N., Burgess, C., Tian, K., and Vuong, T.-D. (2023a).
Quadratic speedups in parallel sampling from determinantal distributions.
In *Proceedings of the 35th ACM Symposium on Parallelism in Algorithms and Architectures*, pages 367–377.

Anari, N., Huang, Y., Liu, T., Vuong, T.-D., Xu, B., and Yu, K. (2023b).
Parallel discrete sampling via continuous walks.
In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 103–116.

Anari, N., Jain, V., Koehler, F., Pham, H. T., and Vuong, T.-D. (2021).
Entropic independence i: Modified log-sobolev inequalities for fractionally log-concave distributions and high-temperature ising models.
arXiv preprint arXiv:2106.04105.

Anari, N., Jain, V., Koehler, F., Pham, H. T., and Vuong, T.-D. (2023c).
Universality of spectral independence with applications to fast mixing in spin glasses.
arXiv preprint arXiv:2307.10466.

Fan, J., Yuan, B., and Chen, Y. (2023).
Improved dimension dependence of a proximal algorithm for sampling.
arXiv preprint arXiv:2302.10081.

Feng, W., Hayes, T. P., and Yin, Y. (2021).
Distributed metropolis sampler with optimal parallelism.
In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2121–2140. SIAM.

Bibliography II

Grathwohl, W., Swersky, K., Hashemi, M., Duvenaud, D., and Maddison, C. (2021).
Oops i took a gradient: Scalable sampling for discrete distributions.
In *International Conference on Machine Learning*, pages 3831–3841. PMLR.

Lee, Y. T., Shen, R., and Tian, K. (2021).
Structured logconcave sampling with a restricted gaussian oracle.
In *Conference on Learning Theory*, pages 2993–3050. PMLR.

Liu, H. and Yin, Y. (2022).
Simple parallel algorithms for single-site dynamics.
In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1431–1444.

Montanari, A. (2023).
Sampling, diffusions, and stochastic localization.
arXiv preprint arXiv:2305.10690.

Rhodes, B. and Gutmann, M. (2022).
Enhanced gradient-based mcmc in discrete spaces.
arXiv preprint arXiv:2208.00040.

Salez, J. (2021).
A sharp log-Sobolev inequality for the multislice.
Ann. H. Lebesgue, 4:1143–1161.

Sambale, H. and Sinulis, A. (2019).
Modified log-sobolev inequalities and two-level concentration.
arXiv preprint arXiv:1905.06137.

Bibliography III

Shen, R. and Lee, Y. T. (2019).
The randomized midpoint method for log-concave sampling.
Advances in Neural Information Processing Systems, 32.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. (2015).
Deep unsupervised learning using nonequilibrium thermodynamics.
In *International Conference on Machine Learning*, pages 2256–2265. PMLR.

Song, Y. and Ermon, S. (2019).
Generative modeling by estimating gradients of the data distribution.
Advances in neural information processing systems, 32.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2020).
Score-based generative modeling through stochastic differential equations.
arXiv preprint arXiv:2011.13456.

Zhang, R., Liu, X., and Liu, Q. (2022).
A langevin-like sampler for discrete distributions.
In *International Conference on Machine Learning*, pages 26375–26396. PMLR.

Inspiration from the continuous analogue

Lemma

Suppose that $X_1 \sim \mu_1, X_2 \sim \mu_2$ are distributions on $\mathbb{R}^n$ with Poincaré constants C_1, C_2. Then $X_1 + X_2 \sim \mu_1 * \mu_2$ has Poincaré constant bounded by $C_1 + C_2$.

Example. $\mu_2 = N(0, C_2)$.

Proof.

1. **Scaling.** $m_i X_i$ has Poincaré constant $m_i^2 C_i$.
2. **Tensorization.** $(m_1 X_1, m_2 X_2)$ has Poincaré constant $\le C = \max\{m_1^2 C_1, m_2^2 C_2\}$.
3. **Projection.** $X + Y = (m_1 X_1, m_2 X_2) \cdot \left(\frac{1}{m_1}, \frac{1}{m_2}\right)$ has Poincaré constant $\le C$ when $\left\|\left(\frac{1}{m_1}, \frac{1}{m_2}\right)\right\| = 1$.

Choose $\frac{1}{m_1^2} = \frac{C_1}{C_1 + C_2}$, $\frac{1}{m_2^2} = \frac{C_2}{C_1 + C_2}$.