Interacting Ensemble MCMC
and
Fast Entropically Regularized SDP

Michael Lindsey

UC Berkeley

November 14, 2023

1. Interacting Ensemble MCMC

- ML, Jonathan Weare, and Anna Zhang,
  Ensemble Markov chain Monte Carlo with teleporting walkers,
  SIAM/ASA JUQ 10, 860 (2022) [arXiv:2106.02686]

2. Fast Entropically Regularized SDP

- ML, Fast randomized entropically regularized semidefinite programming,
  preprint [arXiv:2303.12133].

1. Interacting Ensemble MCMC
   - ML, Jonathan Weare, and Anna Zhang,
     Ensemble Markov chain Monte Carlo with teleporting walkers,
     SIAM/ASA JUQ 10, 860 (2022) [arXiv:2106.02686]

2. Fast Entropically Regularized SDP
   - ML, Fast randomized entropically regularized semidefinite programming,
     preprint [arXiv:2303.12133].

1. Interacting Ensemble MCMC
   - ML, Jonathan Weare, and Anna Zhang,
     Ensemble Markov chain Monte Carlo with teleporting walkers,
     SIAM/ASA JUQ 10, 860 (2022) [arXiv:2106.02686]

2. Fast Entropically Regularized SDP
   - ML, Fast randomized entropically regularized semidefinite programming,
     preprint [arXiv:2303.12133].

Sampling

- Given the ability to evaluate $U(x)$, for $x \in \mathcal{X}$, we want to draw **independent samples** from

$$\pi(x) = \frac{1}{Z} e^{-U(x)},$$

  where Z is a suitable (unknown) normalizing constant
- Ubiquitous applications in scientific computing:
  - ***Bayesian inference* (sampling from posterior distribution)**
    - Data science, computational astronomy, inverse problems, etc.
    - Often low-to-moderate dimensional but too high-dimensional for quadrature
  - *Computational chemistry* (sampling molecular configurations)
  - *Quantum Monte Carlo* (many methods)
  - *Optimization* (simulated annealing, etc.)
  - *...and many more!*

Markov chain Monte Carlo

- Most widely-used approach is **Markov chain Monte Carlo (MCMC)**
- **Idea:** construct a Markov chain $g(y \mid x)$ whose invariant/equilibrium measure is π
- Major generic frameworks for MCMC (*can be combined!*)—
  - **Metropolis-Hastings:** let q be *any* Markov chain, propose move $x \to x'$ according to q and accept with probability

$$A = \min\left(1, \frac{\pi(x')}{\pi(x)} \frac{q(x \mid x')}{q(x' \mid x)}\right)$$

  - **Integrator-based methods:** chain defined by discrete-time integration of an SDE
    - For example **overdamped Langevin** dynamics are defined by

$$dX_t = -\nabla U(X_t)\, dt + \sqrt{2}\, dB_t,$$

      and can be Metropolis-adjusted (**MALA**)
    - See also **underdamped Langevin** and **Hamiltonian Monte Carlo (HMC)**

Metastability

- All of these generic approaches can suffer from **metastability**
- In this case, the autocorrelation time (i.e., number of steps required to get an effectively independent sample) can be arbitrarily long

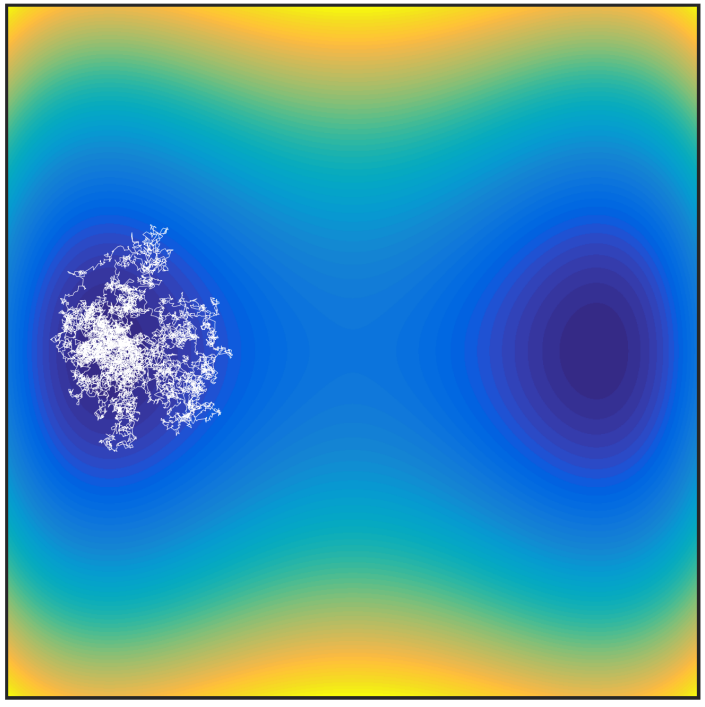

Figure: Illustration of metastability for Langevin dynamics. Colors indicate values of U. (Recall $\pi \propto e^{-U}$.)

Interacting ensemble MCMC

- **Idea:** instead of sampling $x \sim \pi(x)$ directly, we consider an ensemble

$$\mathrm{x} = (x_1, \ldots, x_N) \in \mathcal{X}^N$$

of N walkers and sample

$$\mathrm{x} \sim \Pi(\mathrm{x}) = \prod_{i=1}^{N} \pi(x_i)$$

- Then each walker individually samples from π, as originally desired
- Although the walkers are distributed independently, our **proposal** will allow interaction between them
  - We will Metropolize our interacting-walker proposal to preserve the joint distribution Π

Teleporting proposal

- Assume we are given any proposal $q(y \mid x)$
- Uniformly select walker index $j \in \{1, \ldots, N\}$
  - The j-th walker will be cloned and then moved according to q
  - Specifically, sample $z \sim q(\cdot \mid x_j)$

Teleporting proposal

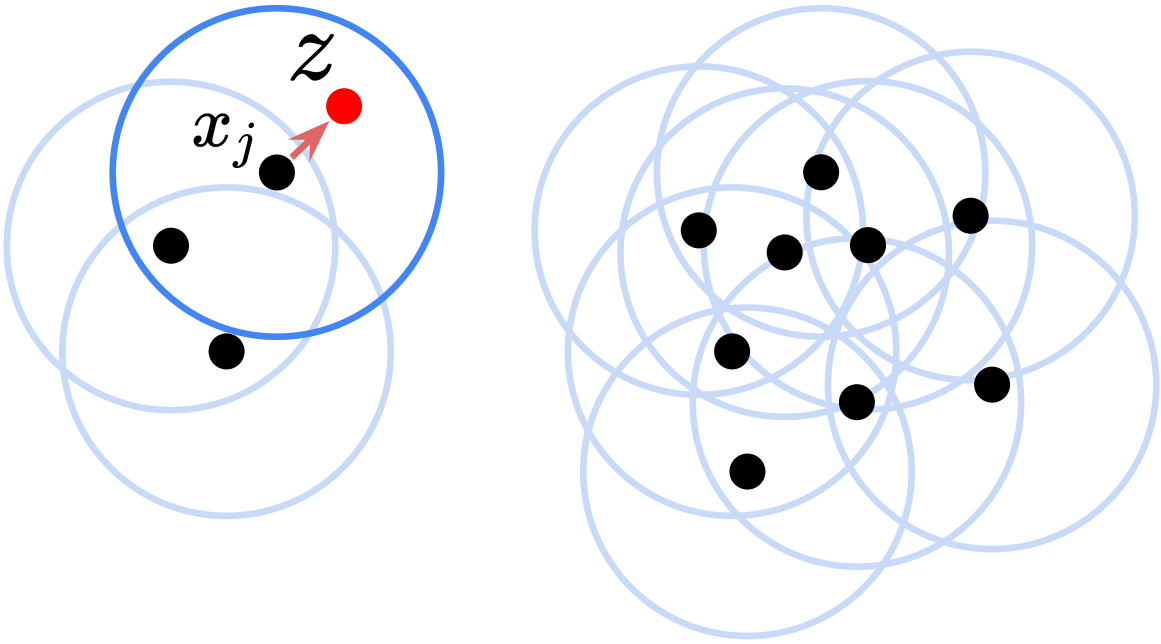

Figure: Illustration of teleporting proposal. Think of the target density π as uniform for simplicity.

Teleporting proposal

- Then we will sample an index i (possibly $i = j$) for deletion
- i.e., we will propose $x_i \leftarrow z$
  - If $i \neq j$, it's as if we have proposed teleporting walker i to be near walker j
- Specifically, i is sampled according to the importance weights

$$w_i \propto \frac{q(x_i \mid z) + \sum_{k \neq i}^{N} q(x_i \mid x_k)}{\pi(x_i)},$$

  where the weights are normalized to sum to one
  - Choice guarantees acceptance probability of 1 in the infinite-walker limit $N \to \infty$
  - We do not delete any walker that is 'lonely,' unless we have just cloned that walker

Teleporting proposal

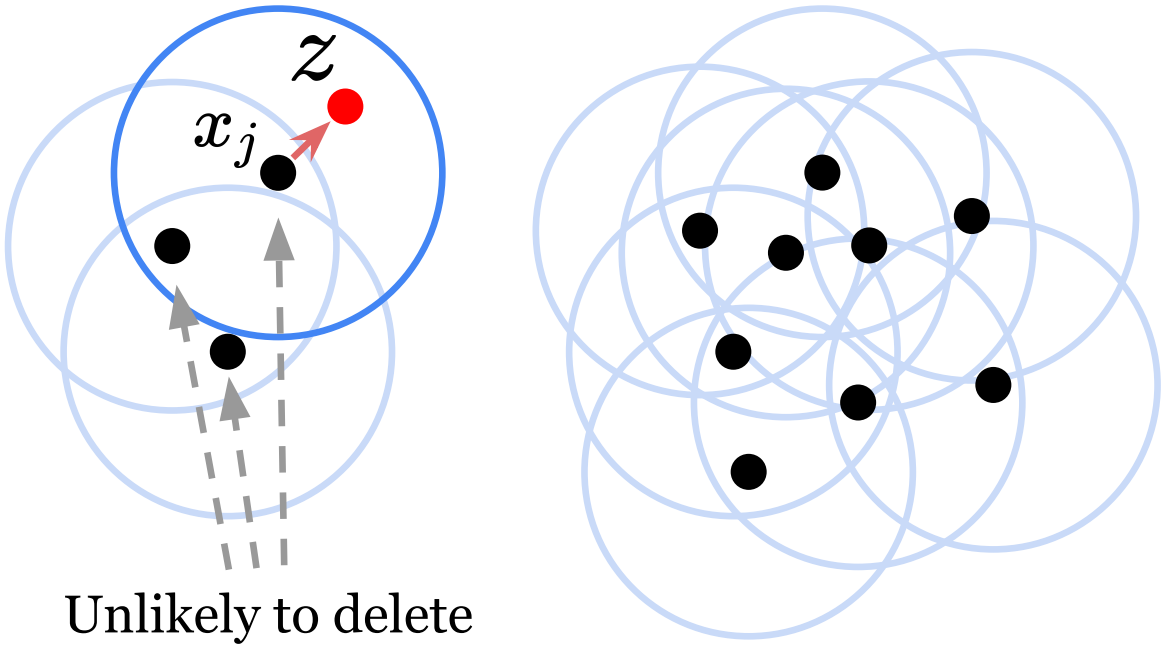

Figure: Illustration of teleporting proposal. Think of the target density π as uniform for simplicity.

Mean-field limit

- Can try to get theoretical understanding via **mean-field limit**
- In the limit of large N we can approximate the empirical measure of the walker positions

$$\mu := \frac{1}{N} \sum_{i=1}^{N} \delta_{x_i}$$

by a continuous density function, i.e.,

$$d\mu(x) \approx \rho(x)\, dx$$

Mean-field limit

- Obtain mean-field evolution for the density

$$\partial_t \rho(x) = \left[1 - \frac{1}{Z_\rho} \frac{\rho(x)}{\pi(x)}\right] \mathcal{Q}\rho(x)$$

- Z_ρ is the constant that guarantees conservation of total probability

$$\int \partial_t \rho \, dx = 0$$

- $\mathcal{Q}$ is the Markov transition kernel operator

$$\mathcal{Q}\rho(x) = \int q(x \,|\, y) \, \rho(y) \, dy$$

Mean-field limit

- The mean-field dynamics enjoy convergence to π that is monotone in the Pearson χ^2-divergence

$$\chi^2(\pi \,\|\, \rho) = \int \left(1 - \frac{\pi}{\rho}\right)^2 \rho \, dx$$

  - **Aside:** dynamics also admit interpretation as a gradient flow for the *reversed* χ^2-divergence

Theorem (ML, Weare, Zhang)

Under suitable technical conditions, $\chi^2(\pi \,\|\, \rho_t)$ is monotone decreasing. Moreover, there exists C independent of t such that

$$\chi^2(\pi \,\|\, \rho_t) \leq Ce^{-t/\gamma_\infty},$$

where $\gamma_\infty := \frac{1}{2}\|\pi/\mathcal{Q}\pi\|_\infty$.

Mean-field limit

Theorem (ML, Weare, Zhang)

Under suitable technical conditions, $\chi^2(\pi \,\|\, \rho_t)$ is monotone decreasing. Moreover, there exists C independent of t such that

$$\chi^2(\pi \,\|\, \rho_t) \leq Ce^{-t/\gamma_\infty},$$

where $\gamma_\infty := \frac{1}{2}\|\pi/\mathcal{Q}\pi\|_\infty$.

- Asymptotic rate of convergence for the ***non-interacting ensemble*** is controlled by the spectral gap of $\mathcal{Q}$
- Asymptotic rate of convergence for our scheme is ***not*** limited by the spectral gap
  - For a narrow proposal, rate ≈ 2 independent of proposal

Illustrative example

- Consider the double-well potential

$$U(x) = \beta(x^4 - x^2),$$

where β is an inverse temperature parameter controlling the depth of the two wells

- Use Gaussian proposal $q(y \mid x) \propto e^{-\frac{1}{2\sigma^2}(y-x)^2}$

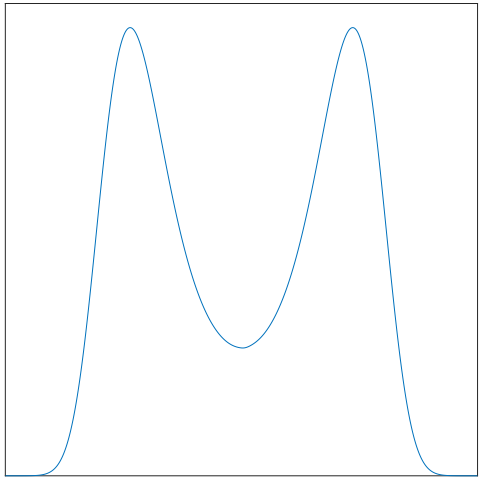

Figure: Graph of $\pi(x)$ for $\beta = 5$.

Illustrative example

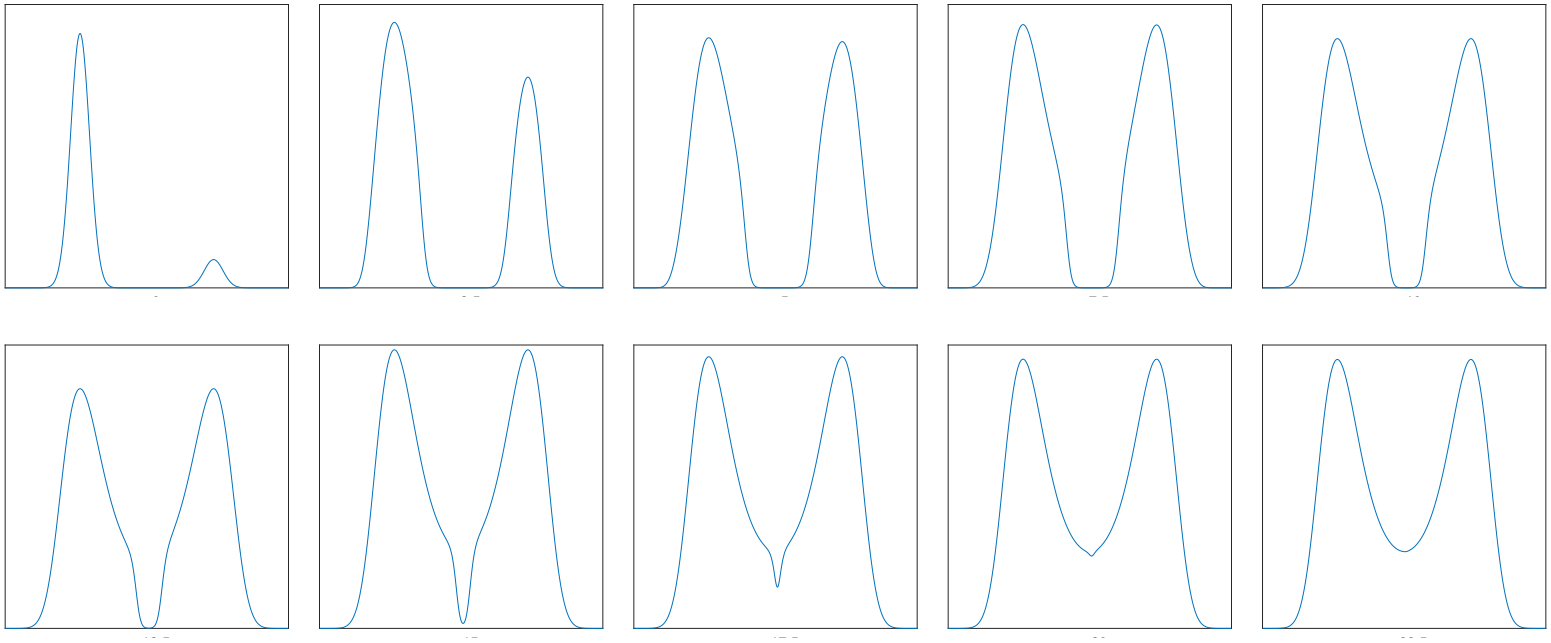

Figure: ρ_t according to the mean-field dynamics with $\beta = 5$, $\sigma = 0.0125$ at times $t = 0$, 2.5, 5, 7.5, 10, 12.5, 15, 17.5, 20 , 22.5, ordered left-to-right, then bottom-to-top.

Illustrative example

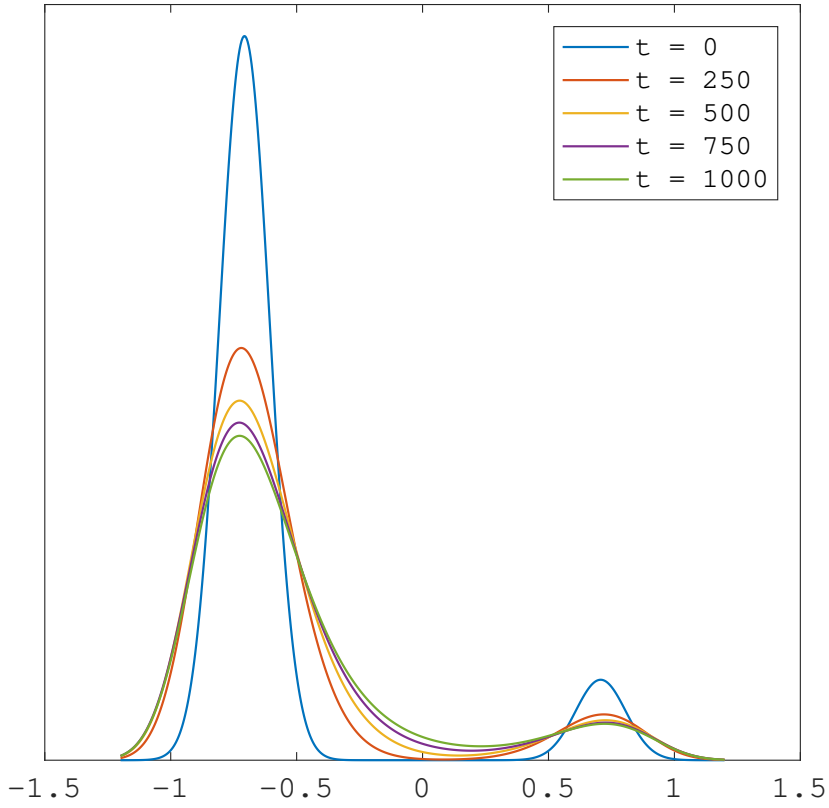

Figure: ρ_t according to the mean-field dynamics for a **non-interacting** ensemble with $\beta = 5$, $\sigma = 0.0125$ at several different times. Note that even by time $t = 1000$, the dynamics are far from convergence.

Illustrative example

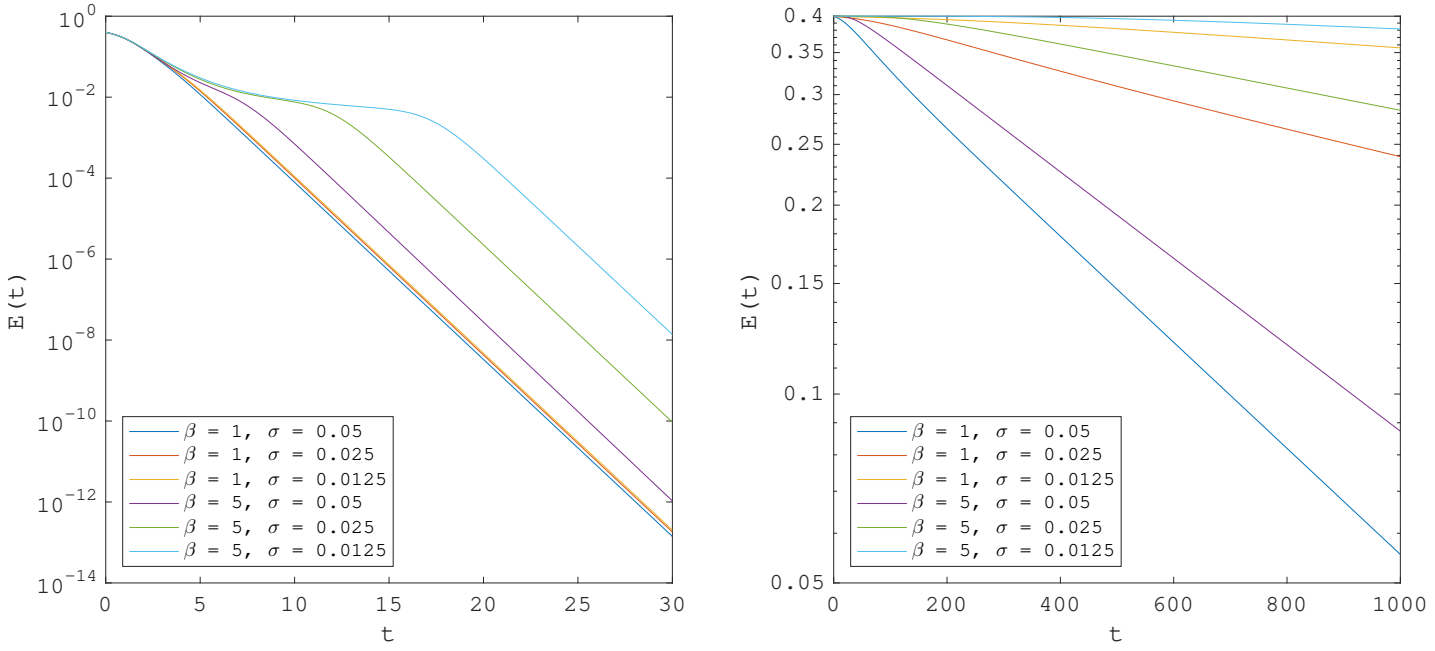

Figure: Convergence of $\mathbb{P}_{X\sim\rho}(X \geq 0)$ for **interacting dynamics (left)** and **non-interacting dynamics (right)**, for several different values of β, σ. Note the different horizontal and vertical axis scales at left and right.

Bayesian hyperparameter estimation

- We consider a multimodal Bayesian posterior sampling problem introduced in [Yao, Vehtari, and Gelman (2020)]
- Goal is to **estimate hyperparameters in Gaussian process regression**
- Observe data (x_i, y_i) and assume that

$$y_i = f(x_i) + \varepsilon_i,$$

where $f \sim \mathcal{GP}(0, \Sigma)$, $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ i.i.d. noise, and

$$\Sigma(x_1, x_2) = \alpha^2 \left(-\frac{(x_1 - x_2)^2}{\rho^2} \right)$$

- Place indepedent Cauchy priors on our hyperparameters $\theta = (\alpha, \rho, \sigma)$
- Sample from posterior distribution $p(\theta \,|\, y)$

Bayesian hyperparameter estimation

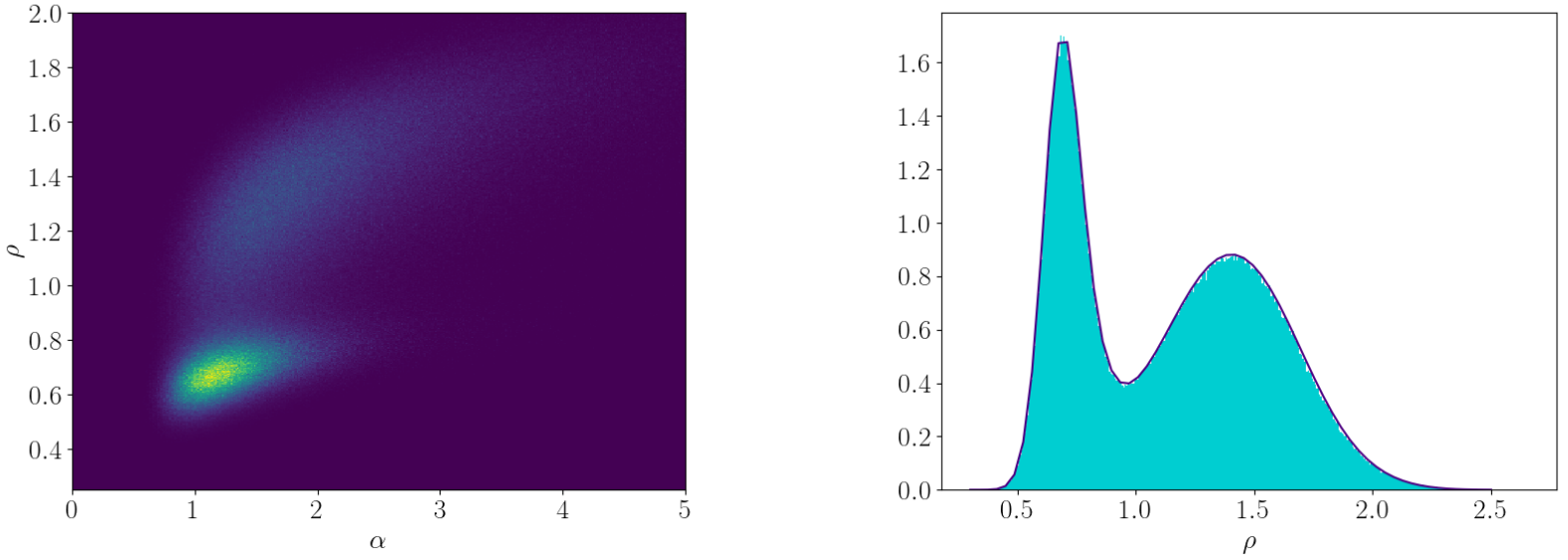

Figure: Posterior (α, ρ) marginal **(left)** and ρ marginal **(right)**.

Bayesian hyperparameter estimation

N	1	10	50
IAT	2111	867	97

Table: Integrated autocorrelation time. We see an **over 20-fold efficiency gain** by considering an interacting scheme with $N = 50$ instead of a single walker, assuming cost is dominated by the the **number of density evaluations** (usually the bottleneck).

Results not shown

- Similar results for the case of multivariate Gaussian process, where the data $x_i \in \mathbb{R}^3$ (9-dimensional hyperparamter)
  - 16-fold efficiency gain of $N = 100$ over $N = 1$
- Paper also considers extension for interaction of a subset of variables
- **Ongoing work:**
  - Push the advantage to larger ensemble scales ($N \sim 10^5$-10^6) with a modified scheme allowing **fast kernel operations** for walker interaction and **parallel density evaluations**
  - Nonlocal proposals can still provide huge computational speedup ***even for unimodal densities!***

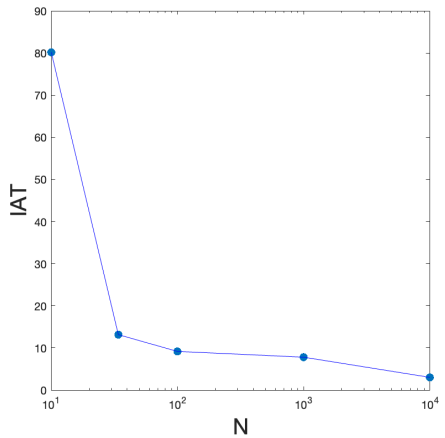

Figure: IAT as a function of ensemble size for GPR hyperparameter posterior sampling.

Related work

- Related work arises by starting with a Fokker-Planck equation for a birth-death stochastic process, then considering a discrete-time particle approximation
  - Y. Lu, J. Lu, and J. Nolen [arXiv:1905.09863]
  - G. Rostkoff, S. Jelassi, J. Bruna, and E. Vanden-Eijnden [arXiv:1902.01843]
  - M. Gabrie, G. Rostkoff, and E. Vanden-Eijnden [arXiv:2105.12603]

1. Interacting Ensemble MCMC
   - ML, Jonathan Weare, and Anna Zhang,
     Ensemble Markov chain Monte Carlo with teleporting walkers,
     SIAM/ASA JUQ 10, 860 (2022) [arXiv:2106.02686]

2. Fast Entropically Regularized SDP
   - ML, Fast randomized entropically regularized semidefinite programming,
     preprint [arXiv:2303.12133].

1. Interacting Ensemble MCMC
   - ML, Jonathan Weare, and Anna Zhang,
     Ensemble Markov chain Monte Carlo with teleporting walkers,
     SIAM/ASA JUQ 10, 860 (2022) [arXiv:2106.02686]

2. Fast Entropically Regularized SDP
   - ML, Fast randomized entropically regularized semidefinite programming,
     preprint [arXiv:2303.12133].

Why SDP?

- Semidefinite programs arise in many settings, often from the relaxation of an underlying difficult problem
- The most fundamental examples are from 0-1 **combinatorial optimization**, starting from [Goemans and Williamson (1995)]
  - See also the **Lassere Hierarchy**, starting with [Lasserre (2001)], as well as [Wainright and Jordan (2008)], for systematic approaches
- My interest in SDP comes from ***marginal relaxations*** for scientific computing problems:
  - **Density functional theory** [Khoo, Lin, ML, and Ying (2020)]
  - **Continuous global optimization** [Chen, Khoo, and ML (2020)]
  - **Quantum many-body problems** [Lin and ML (2022)], [Khoo and ML (2022)]

Max-Cut problem

- Given a graph (V, E), want to color the vertices white/black so that as many edges as possible connect unlike colors

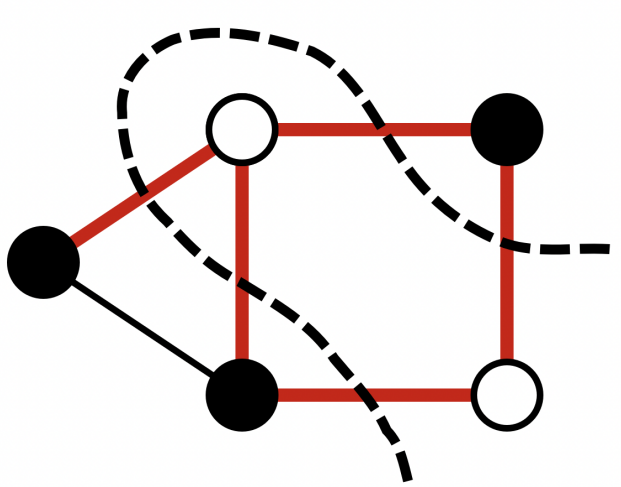

Figure: Maximum cut of a small graph

- Formally, if the vertices are indexed by $i = 1, \ldots, n$, want to solve:

$$\min_{\{-1,1\}^n} \sum_{i,j\,:\,(i,j)\in E} x_i x_j$$

- ***NP-complete!***
  - Hard to do much better than enumerating all 2^n possibilities for $x = (x_i)$

Goemans-Williamson relaxation

- On the domain $x \in \{-1, 1\}^N$, rewrite the objective:

$$\sum_{i,j\,:\,(i,j)\in E} x_i x_j = \sum_{i,j=1}^{N} A_{ij} x_i x_j = x^\top A x = \mathrm{Tr}[A x x^\top],$$

  where A is the adjacency matrix of the graph
- Let

$$X = x x^\top$$

  and observe that $\mathrm{diag}(X) = 1$, $X \succeq 0$
- If we optimize $X \in \mathbb{R}^{n \times n}$ subject only to these constraints, we obtain a ***relaxation*** of the original problem, providing a ***lower bound*** on the optimal value

Goemans-Williamson relaxation

- Specifically

$$\begin{aligned} \underset{X \in \mathbb{R}^{n \times n}}{\text{minimize}} \quad & \text{Tr}[AX] \\ \text{subject to} \quad & X \succeq 0, \\ & \text{diag}(X) = 1 \end{aligned}$$

- This relaxation is due to **Goemans and Williamson (1995)**
  - They also provide a randomized rounding procedure from the solution X to an element $x \in \{-1, 1\}^n$
  - Plugging in x yields an upper bound guaranteed to yield an upper bound achieving an **approximation ratio** of

$$\alpha \approx 0.878$$

- In fact it is conjectured [Khot et al (2007)] to be the best possible guaranteed approximation ratio, and it is known [Trevisan et al (2000)] that ≥ 0.941 is NP-hard
- Aside from interest in Max-Cut per se, the GW relaxation is the **prototypical** semidefinite relaxation and SDP

General SDP

- A general SDP can be written

$$\begin{aligned} \underset{X \in \mathbb{R}^{n \times n}}{\text{minimize}} \quad & \mathrm{Tr}[CX] \\ \text{subject to} \quad & X \succeq 0, \\ & \mathrm{Tr}[A_k X] = b_k, \quad k = 1, \ldots, m, \end{aligned}$$

  though sometimes alternative equivalent presentations may be preferred based on structure

Review of optimization approaches

- How to solve an SDP? There are several categories of methods:
  - **Interior point methods:** strong convergence guarantees but very poor scaling per iteration
  - **Augmented Lagrangian / ADMM-type methods:** weaker convergence guarantees but optimal-in-general $O(n^3)$ scaling per iteration (e.g., SDPNAL, cf. Toh et al)
  - **Low-rank methods:** exploit low-rank assumption on solution, e.g., **SDPLR** [Burer and Monteiro (2003)] and **SketchyCGAL** [Yurtsever et al (2021)]
  - **TCS-style algorithms:** e.g., **MMWU** [Arora and Kale (2007)]
- We want linear scaling (assuming sparsity of the cost and constraint matrices) ***without*** a low-rank assumption
  - The MMWU has nice theoretical scaling guarantees but impractical to implement!
- **This work:** first practical linear-scaling algorithm achieving a fixed approximation ratio for Max-Cut (though applies more broadly)

Entropic regularization

- Inspired by the success of the **entropic regularization of optimal transport** [Cuturi 2013], which is a ***linear program***, we are motivated to consider the entropic regularization of ***SDP*** as a general computational tool
  - In addition to quantum statistical mechanics literature, see [Krechetov (2019)], [Lin and ML (2022)], and [Pavlov et al (2022)] for other uses
- For a positive definite matrix X define the **von Neumann entropy**

$$S(X) = \mathrm{Tr}[X \log X] - \mathrm{Tr}[X]$$

  - Can be viewed as a quantum analog of Shannon's classical entropy, fundamental in quantum information theory, cf. [Nielsen and Chuang]

Entropic regularization

- Von Neumann entropy noncommutative/quantum analog of classical entropy, appearing in quantum information theory
- Consider the regularized problem, where $\beta \in 0, \infty)$ is a regularization parameter ('*inverse temperature*'):

$$\begin{array}{ll} \underset{X \succ 0}{\text{minimize}} & \text{Tr}[CX] + \beta^{-1} S(X) \\ \text{subject to} & \text{Tr}[A_k X] = b_k, \quad k = 1, \ldots, m \end{array}$$

- Note that the entropy acts as a ***barrier*** to the boundary of $\{X \succeq 0\}$ and also makes the problem ***strictly convex***

Dual problem

- Restrict to case of diagonal constraint $\mathrm{diag}(X) = b$ for simplicity
- The dual problem is unconstrained:

$$\max_{\lambda \in \mathbb{R}^n} \quad b \cdot \lambda - \beta^{-1} \mathrm{Tr}\left[e^{-\beta(C - \mathrm{diag}(\lambda))}\right]$$

- In quantum statistical mechanics interpretation:
  - $C_\lambda := C - \mathrm{diag}(\lambda)$ is an effective **Hamiltonian**
  - β is the **inverse temperature**
  - $Z_{\beta,\lambda} := \mathrm{Tr}[e^{-\beta C_\lambda}]$ is the **partition function**
  - $X_{\beta,\lambda} := e^{-\beta C_\lambda}$ is the (unnormalized) **density operator**
  - $F_{\beta,\lambda} := -\beta^{-1} Z_{\beta,\lambda}$ is (kind of) the **free energy**
- The **gradient** of the dual objective

$$b - \mathrm{diag}(X_{\beta,\lambda})$$

- Hence we want to find a **dual solution** $\lambda^\star$ such that $\mathrm{diag}(X_{\beta,\lambda^\star}) = b$, and this $X_{\beta,\lambda^\star}$ is in fact the **primal solution** of the regularized problem

Stochastic diagonal estimation

- How to compute $\mathrm{diag}(X_{\beta,\lambda}) = \mathrm{diag}(e^{-\beta C_\lambda})$?
  - Forming the matrix exponential, even if C_λ is sparse, consumes $O(n^3)$ cost
- Randomized approach which has appeared in the GPR literature [Mathur et al (2021)]:

$$\mathrm{diag}(X) = \mathbb{E}_{z\sim\mathcal{N}}\left[(X^{1/2}z)\odot(X^{1/2}z)\right],$$

  where z is a standard Gaussian random vector
- We prove **concentration bounds** for the corresponding estimator
  - Relative error of estimator is essentially problem-independent
- Hence we only need to compute matrix-vector multiplications

$$X_{\beta,\lambda}^{1/2}z = e^{-\frac{\beta}{2}C_\lambda}z$$

  - Fast matrix-free algorithm available [Al-Mohy and Higham (2011)], or alternatively matrix-free Chebyshev expansion [Driscoll, Hale, and Trefethen (2014)]

Additional comments

- In fact for Max-Cut we do not apply dual gradient ascent, but introduce a specialized **noncommutative matrix scaling** approach
  - But still makes use of the same estimator
- **Not discussed:** *'fermionic' entropic regularization* allows us to solve other problems, such as the **spectral embedding** of a graph with n vertices into $\mathbb{R}^k$
  - Improve $O(nk^2)$ of standard eigensolver approach to $O(nk)$ randomized approximate algorithm

Max-Cut results: convergence profile

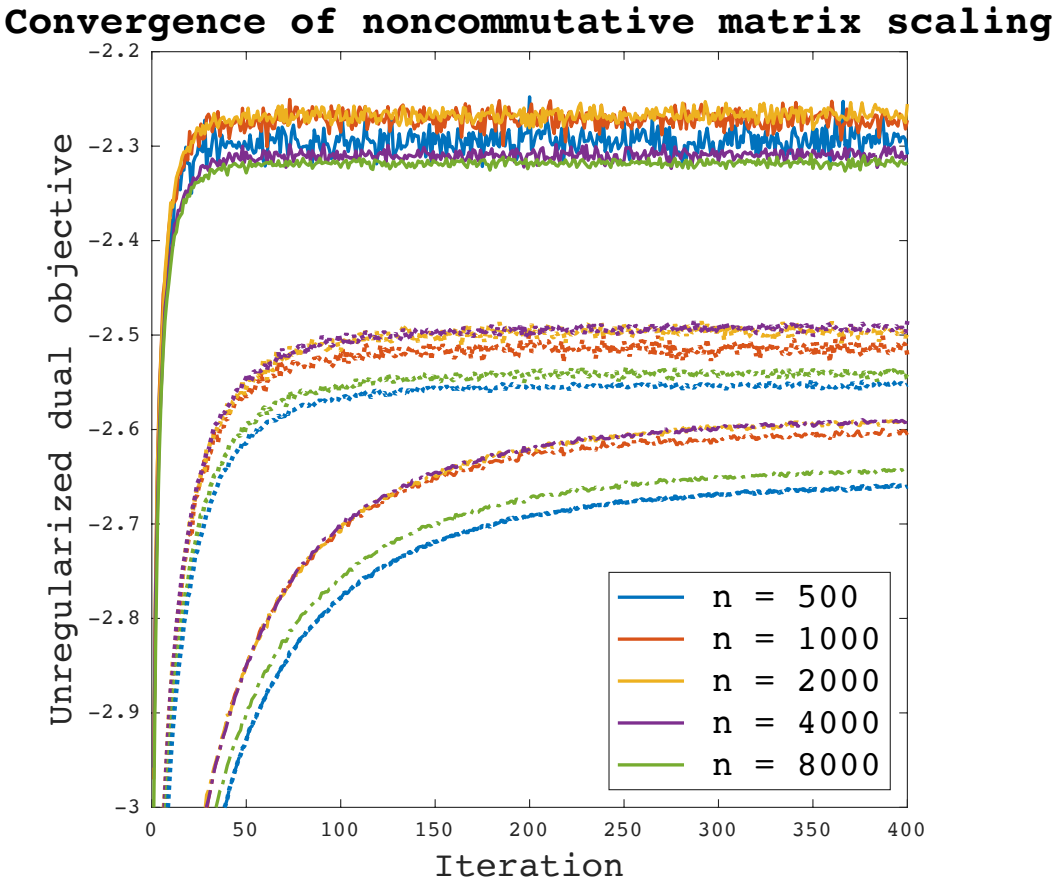

Figure: Convergence profile for various system sizes n and regularization parameters β (solid lines for $\beta = 10$, dotted lines for $\beta = 32$, dashed lines for $\beta = 100$)

Max-Cut results: approximation ratio

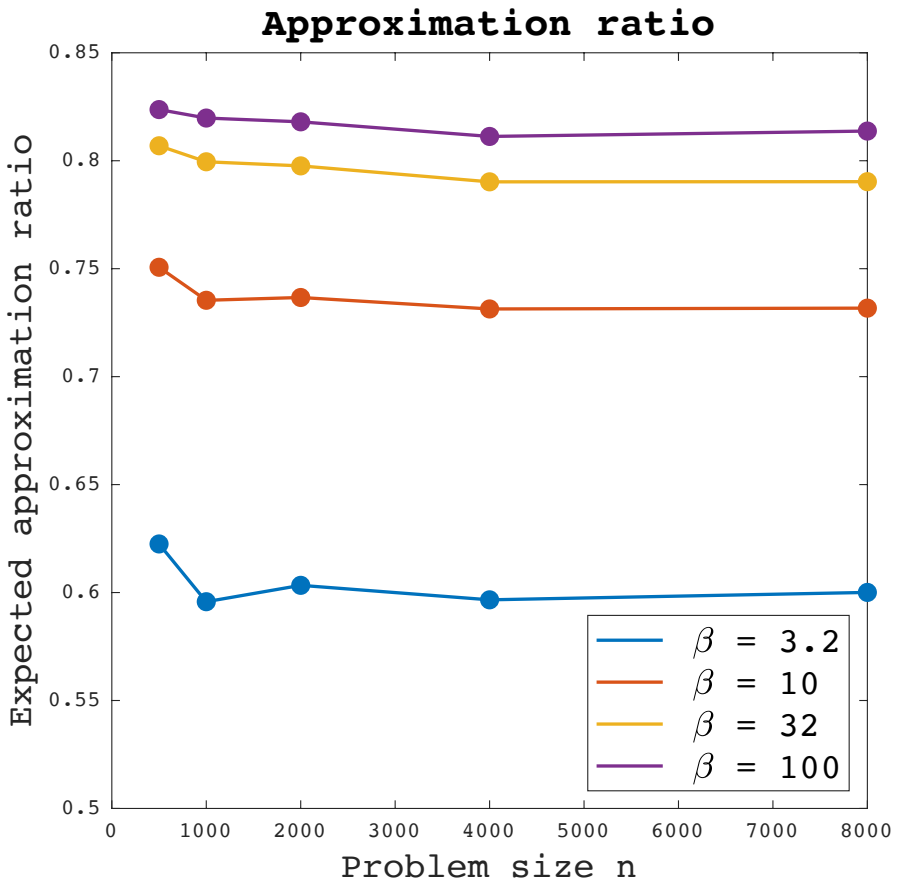

Figure: Approximation ratio obtained as a function of system size, for various regularization parameters β

References

MCMC:

- **ML**, J. Weare, and A. Zhang, Ensemble Markov chain Monte Carlo with teleporting walkers, *submitted (anticipated SIAM/ASA Journal on Uncertainty Quantification)*

SDP:

- **ML**, Fast randomized entropically regularized semidefinite programming, preprint [arXiv:2303.12133]

Thank you very much for your attention!