Distributed Stochastic Gradient Langevin Dynamics on MPI

Feynman Liang

CS 267, UC Berkeley

Introduction

- **Motivation**: Posterior samples $\theta_i \sim P(\theta \mid X)$ are useful for Bayesian statistics, e.g. Monte Carlo approximations

$$n^{-1} \sum_{i=1}^{n} f(\theta_i) \to E_{P(\cdot|X)} f(\theta_i)$$

- **Goal**: Generate posterior samples in a data-parallel manner which makes efficient use of available compute resources
- **Challenges**:
  - Typical MCMC require entire dataset to compute Metropolis-Hastings acceptance probability, limiting scalability
  - Communication costs are non-trivial in distributed computing environments
  - Worker speeds and data partitioning may be imbalanced, requiting non-trivial management to achieve good utilization

Our Method

- Partition data into shards $(X_s)_{s \in S}$
- Using MPI, perform distributed stochastic gradient Langevin dynamics (D-SGLD) [1]

$$\Delta\theta = \frac{\epsilon_t}{2} \left(\nabla \log P(\theta) + N g(\theta_t, X_s) \right) + \nu_t \qquad (1)$$

where $\nu_t \overset{\text{iid}}{\sim} N(0, \epsilon_t)$, $\epsilon_t \to 0$ sufficiently slowly, $g(\theta_t, X_s)$ unbiased estimator of $\nabla \log P(\theta \mid X_s)$.

In addition, we explored the following optimizations and extensions

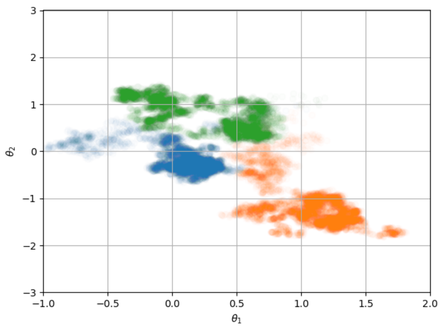
- Trajectory sampling to avoid short communication cycles and trade off between communication overhead and mixing rates [1]
- Trajectory length load balancing to improve utilization [1]
- Stochastic gradient Riemannian Langevin dynamics (SGRLD) to sample parameters constrained to the probability simplex [2]

Parallel chains and chain exchange

Gaussian mixture model (GMM)

- Priors $\theta_1 \sim \mathcal{N}(0, 10)$, $\theta_2 \sim \mathcal{N}(0, 1)$,
- Likelihood $x_i \overset{\text{iid}}{\sim} \frac{1}{2}\mathcal{N}(\theta_1, 2) + \frac{1}{2}\mathcal{N}(\theta_1 + \theta_2, 2)$.

Model has modes at $\theta = (0, 1)$ and $\theta = (-1, 1)$ with negatively correlated parameters



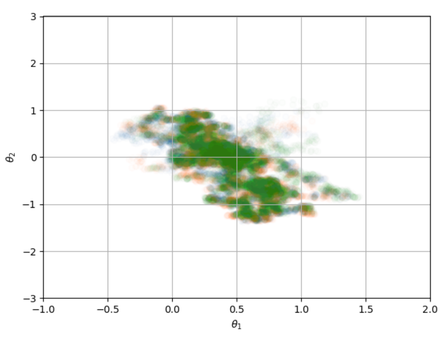


Figure 1: Samples drawn from local posteriors $P(\theta \mid X_s)$ (top, e.g. due to lack of chain exchange) may not well approximate the global posterior $P(\theta \mid \cup_s X_s)$ (bottom).

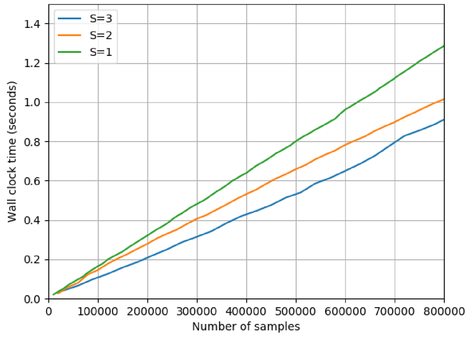


Figure 2: Speedups due to parallelism.

Trajectory sampling

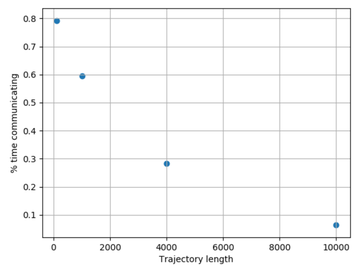


Figure 3: Increasing trajectory length reduces how often communication happens, resulting in less communication overhead.

Trajectory length load balancing

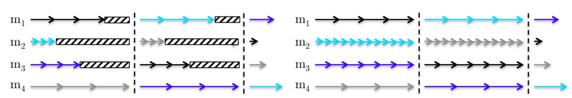


Figure 4: Figure from [1] illustrating improved worker utilization after load balancing.

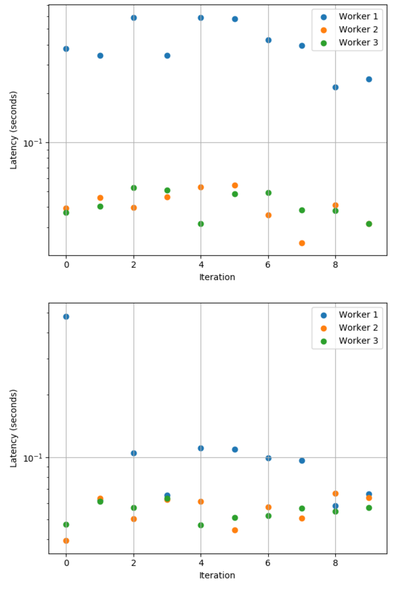


Figure 5: Before (top) and after (bottom) trajectory length load balancing. Experiments used the previous GMM with an imbalanced data partitioning which assigned 95% to worker 1.

Latent Dirichlet Allocation with Riemannian Langevin Dynamics

LDA requires sampling on probability simplex
$\implies$ Riemannian Langevin Dynamics [2]

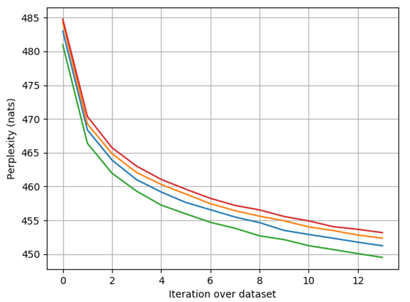


Figure 6: Distributed SGRLD training on NIPS corpus (1740 documents, 19889 unique words) of a 10 topic LDA model.

Conclusion

We provide an implementation of D-SGLD on top of MPI and investigate the performance characteristics of various optimizations. Our results demonstrate that D-SGLD enjoys significant speedups due to parallelism, can sample posteriors even when data is partitioned disjointedly across workers, and can be controlled using trajectory lengths to trade off between communication versus mixing time as well as load balance workers.

References

[1] Sungjin Ahn, Babak Shahbaba, and Max Welling. Distributed stochastic gradient mcmc. In *International conference on machine learning*, pages 1044–1052, 2014.

[2] Sam Patterson and Yee Whye Teh. Stochastic gradient riemannian langevin dynamics on the probability simplex. In *Advances in Neural Information Processing Systems*, pages 3102–3110, 2013.

[3] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 681–688, 2011.