Parallelization of Markov Chain Monte Carlo Methods
Zeyang Ye1 and Yuefan Deng2 (Advisor)

1IACS Jr. Researcher Award Winner; Department of Applied Mathematics & Statistics, Stony Brook University, NY 11794-3600, United States
2IACS Affiliate Faculty; Department of Applied Mathematics & Statistics, Stony Brook University, NY 11794-3600, United States

Abstract
We analyze a massive search space for finding optimal scalable parallelization strategies, including parallel temperature schedule, mixing strategy, and mixing frequency, for rapid convergence of the parallel Markov Chain Monte Carlo methods. We expect to find the optimal time and ways for multiple Markov chains to communicate. Also, we adjust the sequential parameter, temperature, to fit for the parallel method. It is impossible to design a general theory applicable to arbitrary objective functions at this stage of our research. We examine the performance of our strategies by testing the optimization of the mobile route recommendation problem. We find that with the careful selection of the parallel strategies, nearly 100% speedup can be achieved. We believe there exists a scheme for these strategies leading to the optimal parallelization.

Introduction
Markov Chain Monte Carlo (MCMC) method is a randomized heuristic approach to locate a global minimum or a near global minimum state given an objective function. Because of its high computing time, it is effective in solving optimization problems within limited sizes. To solve large and realistic problems, parallel MCMC methods are designed by letting multiple Markov Chains run sequential MCMC method independently and communicating in certain amount of time. By adjusting parallel schemes including mixing frequency and mixing strategies, we hope to speed up MCMC method significantly.

Methodology
Sequential MCMC Methods
Parallel MCMC Methods
Start
Start
Generate an initial solution x
Master core generates and broadcasts an initial solution x

Move x to a*?
exp{(E – E*)} > rand[0,1)?
No
Yes

Keep x

Update T

Converge?

Yes

No

Gather the results and output the best

End

Temperature

Parallel temperature schedule

\[ T_0 = \begin{cases} 1 \text{ (if } x < a^*) \\ (E - E*) \text{ if } (E - E*) > \text{rand}[0,1] \end{cases} \]

Mixing Frequency

Mixing Strategy

MCMC Method
Higher Dimension
Parallelization

Figure 1(a) Comparisons of Computational Efficiency with Different Dimensions

Figure 1(b) Performance curve of the proposed approaches with dimension 5000

Figure 1(c) Decrease of the computing time with increase of the number of cores. The percentage in the figures are the relative different of \( E \) compared to the sequential one.

Conclusions
1. Basic parallel scheme for parallel MCMC method obtains moderate speedup.
2. Using mixing strategies, mixing frequencies, and parallel temperature adjusted, parallel MCMC method achieves nearly 100% speedup.
3. In an optimal parallel scheme, mixing frequency decreases while the computing time increases.
4. 1-to-all mixing strategy can speed up parallel MCMC method as long as it has a suitable parallel temperature schedule.

References