A MASSIVELY PARALLEL EVOLUTIONARY MARKOV CHAIN MONTE CARLO ALGORITHM FOR SAMPLING SPATIAL STATE SPACES

Allocation: Illinois/100 Knh
PI: Wendy K. Tam Cho[1]
Co-PI: Yan Liu[2]
Collaborator: Simon Rubinstein–Salzedo[3]

[1]University of Illinois at Urbana–Champaign
[2]National Center for Supercomputing Applications
[3]Euler Circle

EXECUTIVE SUMMARY

The research team has developed an Evolutionary Markov chain Monte Carlo algorithm for sampling from large, idiosyncratic, and multimodal state spaces. This algorithm combines the advantages of evolutionary algorithms (EA) as optimization heuristics for state space traversal and the theoretical convergence properties of Markov chain Monte Carlo algorithms for sampling from unknown distributions. The team encompassed these two algorithms within the framework of a multiple-try Metropolis Markov chain with a generalized Metropolis–Hastings ratio. Further, the group harnessed the computational power of massively parallel architecture by integrating a parallel EA framework that guides Markov chains running in parallel. Because the algorithm is a sampling algorithm, it is applicable to any field that samples, which is essentially all fields of science.

RESEARCH CHALLENGE

The challenge is to create a method that randomly samples from unstructured spatial data with stringent spatial constraints.

METHODS & CODES

The researchers' approach is two-pronged. The first prong involves developing an optimization algorithm that explores the solution landscape in a spatially aware manner. The second prong involves integrating the optimization algorithm into the sampling framework of a Markov chain Monte Carlo (MCMC) technique. The algorithm is implemented in ANSI C and can be compiled on Linux and OS X as a standard *makefile* project. It uses MPI nonblocking functions for asynchronous migration for load balancing and efficiency. In addition, the C SPRNG 2.0 library provides a unique random number sequence for each MPI process, which is necessary for running a large number of parallel MCMC chains.

RESULTS & IMPACT

MCMC methods are used to sample from unknown distributions. While the theory ensures sampling from unknown distributions, this theoretical result is asymptotic (approaching a value or curve arbitrarily closely). For large applications, the time required before the theoretical convergence is realized may be prohibitively long. Hence, while MCMC methods are theoretically attractive, successful implementation for complex applications can be quite challenging.

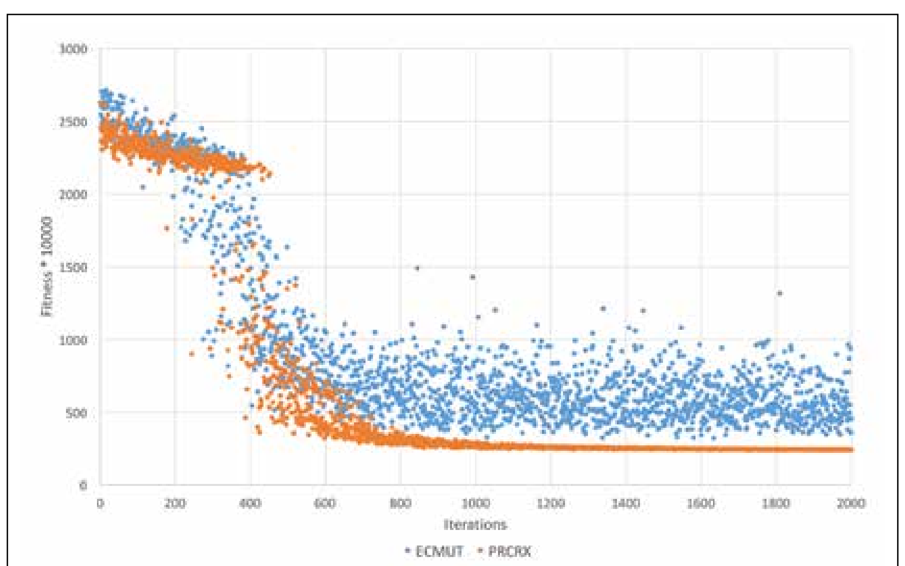

Figure 1: Performance of the spatial path relinking crossover operator. This operator was designed in the context of an evolutionary algorithm crossover. It has been adapted via a multiple-try framework for a Markov chain Monte Carlo algorithm. The figure shows the effectiveness of the operator in identifying increasingly optimal states.

Figure 2: The algorithm produced over three million electoral maps for the State of Ohio by partitioning voter tabulation units into 16 districts. These maps were produced for the partisan gerrymandering case, APRI v. Householder. A three-judge panel found the evidence compelling and ordered a redrawing of the Ohio congressional map [1].

A common MCMC strategy is to define a Markov transition function that amounts to a small or local random change in the current state. Small local changes are attractive for two reasons. First, they are conceptually and operationally simple. Second, the Metropolis–Hastings ratio needs to lead regularly to accepted transition proposals. Since a small change likely results in a large Metropolis–Hastings ratio, the movement of the Markov chain is then fairly fluid. At the same time, because these are small movements in a very large state space, the resulting Markov chain converges slowly and is, moreover, likely to become trapped in localized regions. Hence, for large or complex applications, it is not likely to converge rapidly enough to be practically useful.

To improve performance and hasten convergence, one strategy is to define larger steps for the Markov chain. If designed well, larger moves have the ability to more efficiently and effectively traverse a large state space, leading to faster convergence of the chain. At the same time, devising large and effective movements intelligently is not straightforward. "Large" movements often result in small Metropolis–Hastings ratios, which lead to rejected proposals, and thus to a nonfluid and ineffective Markov chain.

To devise a chain that is able to traverse the state space in both an effective and efficient manner, the team integrated movements from optimization heuristics. A central task in marrying these two techniques is to fit the mechanics of the optimization search within the theoretical framework that enables sampling in an MCMC algorithm. One cannot use optimization operators directly, but one can adapt the operators to an MCMC framework to provide the proposal set for directional sampling according to the structure of the multiple-try Metropolis Markov chain model.

In short, the project has resulted in an evolutionary Markov chain Monte Carlo that uses evolutionary algorithm operators to guide a large number of parallel Markov chains. Statistical evidence generated by this algorithm was used in a lawsuit filed in federal court in Ohio that resulted in a three-judge panel finding that the current congressional map in that state was a partisan gerrymander [1].

WHY BLUE WATERS:

These methods scale to all of the processor cores on Blue Waters through nonblocking MPI communication calls. The computational approach the team implemented in their solution requires generating a very large number of solutions that comprise a representative sample. Generating a large number of statistically independent draws is only feasible on a leadership-class supercomputer such as Blue Waters.

PUBLICATIONS & DATA SETS

W. K. T. Cho, "Technology-enabled coin flips for judging partisan gerrymandering," *South. Calif. Law Rev. Postscript*, vol. 93, pp. 11-27, 2019.

W. K. T. Cho and S. Rubinstein-Salzedo, "Rejoinder to 'understanding our Markov chain significance test,'" *Stat. and Publ. Pol.*, vol. 6, no. 1, p. 54, 2019.

W. K. T. Cho and S. Rubinstein-Salzedo, "Understanding significance tests from a non-mixing Markov chain for partisan gerrymandering claims," *Stat. and Publ. Pol.*, vol. 6, no. 1, pp. 44-49, 2019.

W. K. T. Cho and Y. Y. Liu, "A massively parallel Evolutionary Markov Chain Monte Carlo algorithm for sampling complicated multimodal state spaces," extended abstract in *SC18: The International Conference for High Performance Computing, Networking, Storage, and Analysis*, Dallas, TX, USA, Nov. 11–16, 2018.

W. K. T. Cho, "Algorithms can foster a more democratic society," *Nature*, vol. 558, p. 487, June 28, 2018. Available: https://www.nature.com/articles/d41586-018-05498-y

W. K. T. Cho and Y. Y. Liu, "Sampling from complicated and unknown distributions: Monte Carlo and Markov Chain Monte Carlo methods for redistricting," *Physica A*, vol. 506, pp. 170–178, Sept. 2018.