

## Massively parallel Monte Carlo simulations with population annealing

M. Weigel

<sup>a</sup>*Applied Mathematics Research Centre, Coventry University,  
Coventry, CV1 5FB, England*

While Moore's law of semiconductors has ensured for over forty years that the next generation of processors work significantly faster than the current one, for the last ten years or so *serial* code has not seen any speed-up from new hardware which, instead, achieves performance improvements only from packing more and more cores onto a single die. As a consequence, scientists working with computer simulations need to move away from intrinsically serial algorithms to find new approaches that can make good use of potentially millions of computational cores. *Population annealing*, that was initially suggested by Hukushima and Iba [1] more recently was studied more systematically by Machta [2], is a sequential Monte Carlo scheme that is potentially able to make use of such highly parallel computational resources. Additionally, it promises to allow for the accelerated simulation of systems with complex free-energy landscapes much alike to the much more well known replica-exchange or parallel tempering approach. The relative performance with respect to such more traditional techniques, the appropriate choice of population sizes temperature protocols and other parameters, the estimation of statistical and systematic errors and many other features, however, are essentially uncharted territory. Here, we use a systematic comparison of the new approach to Metropolis as well as parallel tempering simulations for the Ising model to gauge the potential of this new approach, and we suggest a range of heuristics for its application in more general circumstances.

## References

- [1] K. Hukushima and Y. Iba, in *The Monte Carlo Method in the Physical Sciences*, Vol. 690 of *American Institute of Physics Conference Series*, edited by J. E. Gubernatis (PUBLISHER, ADDRESS, 2003), pp. 200–206.
- [2] J. Machta, *Phys. Rev. E* **82**, 026704 (2010).