The Wang Landau parallel algorithm for the simple grids. Optimizing OpenMPI parallel implementation

A S Kussainov

1 Physics and Technology Department, al-Farabi Kazakh National University, Almaty, Republic of Kazakhstan
2 National Nanotechnology Laboratory of Open Type, al-Farabi Kazakh National University, Almaty, Republic of Kazakhstan
E-mail: arman.kussainov@gmail.com

Abstract. The Wang Landau Monte Carlo algorithm to calculate density of states for the different simple spin lattices was implemented. The energy space was split between the individual threads and balanced according to the expected runtime for the individual processes. Custom spin clustering mechanism, necessary for overcoming of the critical slowdown in the certain energy subspaces, was devised. Stable reconstruction of the density of states was of primary importance. Some data post-processing techniques were involved to produce the expected smooth density of states.

1. Introduction
Wang-Landau algorithm became a universal tool to study complex energy landscapes of the multidimensional quantum systems with multiple degrees of freedom and complex bonding mechanism between its members [1]. Numerous publications demonstrate successful application of this method to the wide range of phenomena. It guides itself through the plethora of accessible states by the explicit Hamiltonian of the system and by keeping track of the visited energy states and their accessibility. It could easily adopt or be a part of almost every effective modern algorithm simulating the behavior of the spin glasses. Complex optimization, graph theory and other state-of-the-art numerical methods greatly benefit from this algorithm. Wang-Landau algorithm is based on the Monte Carlo method and Metropolis algorithm [2] and provides an indispensable tool to study the complex systems. It could include in its fabric a cluster flip scheme to overcome critical slowdown around the transition points.

2. Methods
We implemented the all-inclusive universal version of the Wang-Landau Monte Carlo method. We were motivated by the need to get reasonable performance for a systems bigger than 8 by 8 points and also expected to provide a tool against the possible critical slowdown not only around the transition points but also during the regular thermalization stage at low temperatures. The single spin flip algorithm is straightforward and easily substantiated theoretically but generally inefficient. Convergence to the equilibrium values is usually slow and changes introduced by a single local spin flip propagate diffusively. Various algorithms have been proposed to speed up the process by flipping the clusters of spins at once. Swendsen-Wang [3] and Wolff [4] Monte
Carlo methods are among them. The Wolff algorithm is an improvement over the SwendsenWang algorithm since it has a larger probability of flipping the bigger clusters. It is not always possible to provide the exact value of the temperature to the algorithm and we have implemented the method of invasion cluster procedure elaborated by Machta et al [5].

Nowadays everyone should always consider and implement the code in parallel [6], [7] by distributing the tasks between the multiple walkers [8], [9] or split the energy range into pieces. In all cases the unique and custom made procedure to combine the results from a multiple sources is required and was built in our work.

Versatile Open MPI library [10] for the multithread coding in the multicore CPU (central processing unit) environment or in a network of computers connected in the computational cluster was used. OpenMPI is extensively documented in electronic resources and much easier to deploy on the Linux systems. Another, OpenMP project, is among the available alternatives though it works at its full power on the paid operational system platform and software [11].

The types of 2D grids we considered for our density of states simulation were originally studied by Domb [12] and represent a two-dimensional Euclidian $m \times n$ lattices hosting a single spin in every node of its discreet structure. The spin is taking $q$ possible values distributed around the circle with the following steps

$$\theta_n = \frac{2\pi n}{q},$$

where $n$ goes from 1 to $q$ and interaction Hamiltonian is defined as

$$\mathcal{H} = -J \sum_{i,j} \cos(\theta_{si} - \theta_{sj})$$

This model is known as vector Pott model or clock model [13]. Here $J$ is some spin-spin coupling constant and $\theta_{si}$ and $\theta_{sj}$ are the spins’ positions around the clock dial. The simpler standard Pott model has the Hamiltonian

$$\mathcal{H} = -J \sum_{i,j} \delta(s_i, s_j),$$

where $\delta$ is the Kroneker’s delta. When, in eq.(2), spin orientation takes only two values $s_i=\{+1, -1\}$ it is known as the Ising model [14].

To draw an energy landscape for the simple Hamiltonian, see eq.2-3, we need to consider interactions only between the closest neighbors which is a valid assumption considering the magnetic interaction strength is dropping as fast as $1/r^3$ with the distance. Periodic boundary conditions, known as toroidal, were used.

If the external magnetic field $B \neq 0$ is superimposed, the total interaction energy is modified by the presence of individual magnetic dipoles into the following expression

$$E = -J \sum_{i,j} s_is_j - B \sum_{i=1}^{m \times n} s_i,$$

where $J$ is again the spin-spin interaction strength and $B$ is the external magnetic field strength. If $J > 0$ the system is ferromagnetic, otherwise, if $J < 0$, it is paramagnetic. Indices $i$ and $j$ are combined into all accessible pairs of adjacent spins on the grid, excluding the double counts of $ij$ and $ji$ pairs. The upper bound for the first sum in eq.(4) is determined by the range of interaction, which is a simple pairwise interaction, and by the number of spins on the simulation grid.

The complete thermodynamical and statistical description of the system requires the knowledge of thermodynamic partition function and density of states. For canonical ensemble the partition function $Z$ is given by the following expression
\begin{equation}
Z = \sum_{\text{conf.space}} e^{-E_r/k_B T},
\end{equation}

where summation is run over all configuration space, \(k_B\) is the Boltzmann constant, \(E_r\) and \(T\) are the enumerated energy values and the temperature for this particular configuration \([15]\).

High degree of the system’s degeneracy, that is the same values of \(E_r\) are attained by the different system’s spin configurations, allows us to describe all possible configurations only in terms of energy, that is

\begin{equation}
Z = \sum_E g(E) e^{-E/k_B T},
\end{equation}

where \(g(E)\) is the density of states the Wang Landau algorithm is aimed to estimate.

For our analysis we need to keep track of the two values, that is of the \(H(E)\) - which is the number of times this particular energy value is measured at the single Monte Carlo run of the algorithm and \(\tilde{g}(E)\) which is the accumulated relative density of states. Proper normalization will give the true value of \(g(E)\). Probability of the system transition to the next configuration is given by an expression

\begin{equation}
p(E_1 \rightarrow E_2) = \min\left(\frac{\tilde{g}(E_1)}{\tilde{g}(E_2)}, 1\right)
\end{equation}

Process starts with the array of \(H(E)\) values filled with zeros everywhere. The unknown density of states \(\tilde{g}(E)\) is also uniform and taken to be 0 for any value of \(E\). Both values are modified every time any configuration with this particular value of \(E\) is visited. Due to the overwhelming multitude of the Monte Carlo steps it is better to keep track of \(g(E)\) changes through the following expression \(\ln(\tilde{g}(E)) \rightarrow \ln[\tilde{g}(E)] + \ln[f]\). For \(\ln[\tilde{g}(E)]\) the initial value of modification factor \(f\) is chosen to be bigger than 1 and equal \(e \approx 2.71828\) in this particular implementation.

As soon as the condition of flatness for \(H(E)\) is met we change modification factor to a new value \(f_{i+1} = \sqrt{f_i}\) and repeat the procedure starting with \(H(E)\) equals zero everywhere again. Measure of flatness is chosen to be \(\text{mean}(H(E))/\text{max}(H(E)) \times 100\%\). \(H(E)\) is incremented by 1 every time the state with particular \(E\) is visited or the system stays with the old value of \(E\). After multiple iterations, \(\tilde{g}(E)\) should converge to the unnormalized density of states for the simulated system.

Thus, to find the density of states the flat histogram random walk method of Wang and Landau goes through following steps:

1. Choose at random any spin \(s_i\) and flip its sign so \(s_i' = -s_i\)
2. Calculate the total energy according to eq.(2) or eq.(3).
3. Calculate the transition probability according to eq.(7) and keep the updated configuration if the criterion is met. If not, reverse the spin flip and keep the current configuration.
4. Repeat the previous steps to achieve the proper flatness of the histogram.

Every instance of visiting the different spin configuration with certain value of energy \(E\) or keeping the old one, when the transition rules in eq.(7) are not met, is tracked by an increase \(\ln[\tilde{g}(E)] \rightarrow \ln[\tilde{g}(E)] + \ln[f]\) and \(H(E) \rightarrow H(E) + 1\).

These steps, repeated multiple times, which is usually on the order of number of sites in the simulation grid or bigger, represent a single Monte Carlo step. To calculate any observable value for a given temperature or energy range, Monte Carlo steps have to be repeated numerous times.

Different physical properties, including magnetization, specific heat, density of states etc, could be calculated via similar tedious algorithms or directly from the partition function, see
the next three expressions for the thermodynamic total energy, heat capacity at constant volume and entropy

\[
\langle E \rangle = -\frac{\partial \ln Z}{\partial \beta},
\]
\[
C_v = \frac{1}{k_B T^2} \frac{\partial^2 \ln Z}{\partial \beta^2},
\]
\[
S = \frac{\partial}{\partial T} (k_B T \ln Z)
\]

(8)

Splitting the energy range between the processes require some additional thoughts and design. The different energy range pieces are meant to overlap to fit the latest development of the algorithm, see [16]. Each particular \( E \) value is enumerated and has its own slot in \( \tilde{g}(E) \) array to accumulate the number of visits \( H(E) \) from one Monte Carlo step to another. It is easy to show that the number of these slots, for the \( m \times n \) 2D grid, is \( 2mn + 1 \).

3. Results and Discussions
The eight-core desktop PC which was used for simulations runs on Intel Core i7 4790K, 4.0GHz/LGA-1150/22nm/Haswell/8Mb L3 Cache, DDR-3 16Gb/1866MHz PC14900. gcc C++ compiler and Octave packages were deployed on Debian Linux OS .

On Figure 1 we plotted the sample output data from our implementation of the algorithm for the Ising model. The grid size is \( 16^2 \) and the flatness criterion is 85%. For simplicity, magnetic field \( B \) is taken to be zero everywhere.

First subplot (a) on Figure 1 represents the \( H(E) \) - which is the number of visits per energy value accumulated for each separate energy range after the last modification of \( g(E) \), before flashing the data to the hard drive. The separate energy ranges and their overlap are clearly visible on the subplot. The second subplot (b) shows the densities of states \( g(E) \) obtained from the parallel processes before stitching them to a single function. For a better visual analysis the every data piece head is normalized. The thick line of zeros at the bottom represents the states included into \( 2mn + 1 \) range but not accessible due to the system configuration and its interaction Hamiltonian. The third subplot (c) is the final reconstructed density function.

The final graphical data postprocessing and display is done in Octave package. At this stage, depending on the number of artifacts in the data, each head of the consecutive data piece is aligned with the tail of the previous. The head of the first data piece is normalized to 2, because there are only two states when the spins are aligned in the same direction. If head-to-tail stitching at the particular point is unsuccessful we are automatically switching to the different but overlapping in the energy space points.

For the sake of simplicity, we studied only the region from \(-2\) to \(0\) \([E/N]\). In case of the Ising model the positive part is easily reconstructed by the mirror reflection. In all other cases the energy range could be expanded as far as it necessary.
One can see that the difference between the number of visits may be significant for the different pieces of the same model, see subplot (a), nonetheless the first derivative (the factor determining the quality of stitching) for the two adjacent \( g(E) \) pieces at the connection points, as could be understood from the process, is mainly dependent on the smoothness of \( H(E) \).

On Figure 2 we have plotted the data for the three various spin models, that is the Ising model on the subplot (a), the Pott model on the subplot (b) and the clock model on the subplot (c). For the Ising model we increment the grid size from 16 to 128 points. For the Pott model the \( q \) takes the value of 8, 9, 19 and for the clock model the \( q \) takes the value of 7, 9 and 11 correspondingly.

![Figure 2](image)

**Figure 2.** (a) the Ising, (b) the Pott , and (c) the clock model simulations for the different parameters sets, see text for the complete description.

We also implemented the cluster flip algorithm with the set of stopping rules effectively determined by the energy range of the particular thread in the parallel simulations, see Figure 3.

![Figure 3](image)

**Figure 3.** Invaded cluster algorithm implementation data with corresponding grid energy value \( E = -1.7256 \).

Test run and analysis have shown that this algorithm is still more suitable for the transition points but not for keeping the walker at the low energy range and requires a further study.

4. Conclusions

We have implemented multitool Wang-Landau algorithm to calculate density of states based on the Monte Carlo method with importance sampling in 2D spin glasses for the Ising, Domb and Pott models as an example of multithreading and performance optimization in scientific computing in general and in density of states calculations in particular. We reported results produced by a generalized set of rules represented in terms of parallel algorithms and programs to construct a density of states. The algorithm works fine for all type of simple models and only requires correct definition of the interaction Hamiltonian in C++ script. Significant speedup compare to the linear model on the multicore computer has been observed. The model and its development represent the computational basis for the whole generation of the quantum algorithms for approximating partition function. Matlab/Octave scripts are supplied to the code package to produce the final graphical output density of states. Knowing, that it is not always possible to provide the exact value of the
temperature to the algorithm we have implemented the invaded cluster procedure to supplement our single spin flip code.

Acknowledgments
This research was supported by grant 3824/GF4 provided by the Science Committee at the Ministry of Science and Education of Republic of Kazakhstan to the principal investigator at the National Nanotechnology Laboratory of Open Type, Physics and Technology Department, al-Farabi Kazakh National University.

References
[1] Wang F, Landau D P 2011 *Phys. Rev. Lett.* **86** 2050
[2] Metropolis N, Rosenbluth A W 1953 *J. Chem. Phys.* **21** 1087
[3] Wang J, Swendsen R H 1990 *Physica A: Statistical Mechanics and its Applications* **167**(3) 565
[4] Wolff U 1989 *Phys. Rev. Lett.* **62**(4) pp 361-364
[5] Machta J, Choi Y S, Lucke A and Schweizer T and Chayes L M 1996 *Phys. Rev. E* **54**(2) 1332
[6] Kussainov A S and Kussainov S G 2015 *J. Phys.: Conf. Series* **633** 012072
[7] Kussainov A S and Pyn A 2016 *J. Phys.: Conf. Series* **738** 012076
[8] Altevogt P, Linke A 1993 *Parallel Computing* **19**(9) 1041
[9] Heermann D W Burkitt A N 1991 *Parallel Algorithms in Computational Science* (New York:Springer-Verlag)
[10] Pješivac-Grbović Jelena, Angskun T et al. 2004 *Proceedings, 11th European PVM/MPI Users’ Group Meeting* 97
[11] Chapman B 2007 *Using OpenMP: Portable Shared Memory Parallel Programming* (Scientific and Engineering Computation) (Massachusetts:The MIT Press) 384
[12] Domb C, Green M S 1974 *Ising Model. Phase Transitions and Critical Phenomena* V3 357
[13] Potts R B 1952 *Proc. Cambridge Philos. Soc.* **48**(1) 106
[14] Ising E 1925 *Zeitschrift fr Physik* **31** 253
[15] Zhang C, Ma J 2007 *Phys. Rev. E.* **76** 036708
[16] Vogel T et al. *Phys. Rev.E* **90** 023302