Novel, parallel Monte Carlo simulations of systems of interacting classical particles

by

Junqi Yin

(Under the direction of David P. Landau)

Abstract

This dissertation consists of two parts describing the use of Graphical Processing Unit (GPU) accelerated Monte Carlo simulations to attack problems in statistical physics. In Part (1) we use the parallel tempering algorithm on a GPU and perform large-scale Monte Carlo simulations of two different Ising models on square lattices subject to uniform magnetic fields with competing interactions: (a) antiferromagnetic (repulsive) nearest-neighbor (NN) and next-nearest-neighbor (NNN) interactions of the same strength and (b) two-body and three-body interactions. In both cases, phase diagrams are obtained and critical behaviors are studied. In part (2) we study the temperature dependence of structural properties and thermodynamic behavior of water clusters using Wang-Landau sampling. Five different empirical water models are compared, base on which the suggestion for the choice of the water model in Monte Carlo simulations for water clusters and/or inclusions is provided. We also present an implementation for the massively parallel Wang-Landau sampling on multiple GPUs. For GPU applications in both parts, we obtain a $50 \sim 200$ times speedup comparing to corresponding codes on the current generation Intel Central Processing Unit (CPU).

Index words: Ising model, lattice gas, water cluster, competing interaction, water model, phase transition, critical phenomena, Monte Carlo simulation, parallel tempering, Wang-Landau sampling, MPI, GPU, CUDA

Novel, parallel Monte Carlo simulations of systems of interacting classical particles

by

Junqi Yin

B.S., Zhejiang University, China, 2004

M.S., Zhejiang University, China, 2006

M.S., University of Georgia, United States, 2010

A Dissertation Submitted to the Graduate Faculty

of The University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2011

Novel, parallel Monte Carlo simulations of systems of interacting classical particles

by

Junqi Yin

Approved:

Major Professor: David P. Landau

Committee: Steven P. Lewis
Phillip Stancil
Shan-Ho Tsai

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
May 2011

Dedication

To my grandma and grandpa.

Acknowledgments

I am deeply indebted to my advisor, Prof. David P. Landau, for his invaluable guidance and encouragement throughout my entire dissertational research. I learned not only the research methodology but also the scientific attitude.

I would like to thank Prof. Kurt Binder and Prof. Peter T. Cummings for illuminating discussions and insightful comments about my research. I thank Dr. Shan-Ho Tsai for her patient help and support at the Research Computing Center. I also thank Dr. Tobias Preis for introducing me to the general purpose computing on GPU.

I am grateful to Prof. Steven P. Lewis, Prof. Phillip Stancil and Dr. Shan-Ho Tsai for serving on my advisory committee, and for their advice and guidance. I thank Dr. Daniel Seaton and Dr. Stephan Schnabel for fruitful discussions. I should also thank all the members of Center for Simulational Physics for their help and support during my graduate life in Athens.

This research was supported by NSF grant DMR-0810223 and $^\sharp OCI - 0904685$.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

In the study of low-dimensional magnets [1], in addition to the nearest neighbor interaction, usually the next-nearest neighbor interaction also comes into play. If both interactions are ferromagnetic, the system will only undergo a transition to a ferromagnetic state and there are seldom complexities. However, if both interactions are antiferromagnetic, the system is frustrated and there may be multiple configurations of quite similar free energy, which can significantly complicate the phase diagram and the critical behavior. Similar effects can be seen by introducing multiple spin interactions, where one can observe tricritical phenomena or the terminating critical point in the suitable range of relative interaction strengths.

On the other hand, such competing interactions also bring complications to simulations and analyses. Even with efficient sampling methods, relatively large lattice sizes are often necessary to approach enough resolution to study the nature of these transitions. In the first half of this dissertation, we try to tackle these problems of the two dimensional Ising model with competing interactions, which are not only of theoretical interest, but also have possible applications in Graphene and adiabatic quantum computing [2].

The second half of this dissertation is a part of a collaborative project with Oak Ridge National Lab and Vanderbilt University to develop a petaFLOPS application framework for the simulation of biomolecular systems. We start our exploration with water clusters, and develop a generic scheme for massively parallel Wang-Landau sampling.

Wang-Landau sampling [3] as a novel uniform Monte Carlo sampling method has been drawing increasing interest in simulations [4] of both discrete and continuous systems with complex free energy landscapes. The advantage over traditional sampling methods, such as

Metropolis sampling, lies in the fact that in Wang-Landau sampling the system behaves like a random walker in energy and/or other parameter space, instead of being constrained by the Boltzmann distribution. Therefore, a broad parameter space can be effectively sampled in a single simulation. By directly estimating the density of states, one can obtain thermal quantities, such as the free energy, as continuous functions of the temperature and/or other conjugated field variables. Furthermore, the algorithm can be easily implemented on multiple processors and the parallelization is also straightforward.

As the computing power of a Graphical Processing Unit (GPU) surpassed that of a Central Processing Unit (CPU) by orders of magnitude in the last decade, a GPU device becomes an ideal platform for massively parallel applications. For Monte Carlo simulations, there are several studies reported for lattice magnetic models [5], since a regular lattice with short range interactions can be naturally divided into blocks and hence run simultaneously following a checkerboard scheme. For off-lattice models, the parallelization usually obtained with an appropriate algorithm, such as parallel tempering [6] (see Chapter 3) or the Wang-Landau method (see Chapter 5), and it does not depend on the specific system.

In the next Chapter, we will briefly review the methods that are applied in the simulation and analysis. Results for the Ising model with antiferromagnetic nearest- and next-nearest-neighbor interactions are presented in Chapter 3, and the Ising model with two- and three-body interactions are presented in Chapter 4. In the last Chapter, we will discuss our results for water clusters with five different empirical water potentials, and present a multi-GPU implementation for Wang-Landau sampling.

Chapter 2

Simulation methods and Analyses

2.1 Parallel Tempering Method

The Metropolis algorithm [12] is the best known importance sampling Monte Carlo scheme that generates configurations of the system according to a predefined distribution. For a Boltzmann distribution, the transition probability at a given temperature T is given by $min[1, \exp(-\Delta E/k_B T)]$, where k_B is the Boltzmann constant and ΔE is the change of energy before and after the transition move. It works well for simple systems, such as the nearest-neighbor Ising square lattice. However, for a system with a complicated free energy landscape, it can easily get stuck at some metastable state, especially at low temperature, and hence give the wrong statistical average. Luckily, the parallel tempering method [13, 14] provides an easy and effective way to overcome this problem.

The basic idea of the parallel tempering method is similar to simulated annealing [15] which enables the system to escape from the metastable state by increasing the temperature and then relaxing to equilibrium again. In a sense, parallel tempering is like a super-simulated annealing. Many replicas of the same system at different temperatures are simulated simultaneously, and after a fixed number of Monte Carlo steps, a swap trial of configurations between neighboring replicas is performed with a Metropolis-like probability which satisfies the detailed balance condition. In our simulations, one Monte Carlo step is defined as an update trial for every spin, and an extra swap trial in the case of parallel tempering. The transition probability from a configuration X_m simulated at temperature β_m ($\beta = \frac{1}{k_B T}$) to a configuration X_n simulated at temperature β_n would be

$$W(X_m, \beta_m | X_n, \beta_n) = min[1, \exp(-\triangle)], \tag{2.1}$$

$$\triangle = (\beta_n - \beta_m)(\mathcal{H}_m - \mathcal{H}_n). \tag{2.2}$$

In general, such replica exchange not only applies to the temperature set, but also can apply to any other sets of fields, such as the external magnetic field. Following the same argument, one can get the transition probability from $\{X_m, H_m\}$ to $\{X_n, H_n\}$

$$W(X_m, H_m|X_n, H_n) = min[1, \exp(-\triangle)], \tag{2.3}$$

$$\triangle = \beta(H_n - H_m)(M_m - M_n). \tag{2.4}$$

where M_m, M_n are the uniform magnetizations of replica m and n, respectively.

2.2 Wang-Landau Method

Unlike conventional Metropolis methods that generate a canonical distribution at a given temperature T, the Wang-Landau method estimates the density of states g(E) directly and accurately via a random walk that produces a flat histogram $H(E)$ in energy space. Since the partition function can be calculated as

$$Z = \sum_{configurations} e^{-E/k_B T} = \sum_{E} g(E) e^{-E/k_B T} \tag{2.5}$$

where k_B is the Boltzman factor, once $g(E)$ is estimated, one can calculate Z at any T, hence any thermodynamic quantities can be evaluated as continuous functions of the temperature.

The simulation procedure follows several stages:

```
1. initialize g(E), H(E) and modification factor f

2. randomly pick a particle, and perform a trial move
   that drives the system from E to E‘

3. accept the trial move with probability min[1,g(E)/g(E‘)];
      update g(E) = g(E) x f and H(E) = H(E) + 1

4. repeat 2 and 3 until H(E) is flat according to a predefined criterion

5. reset H(E) = 0 and f = sqrt(f)

6. repeat 2-5 until f is no bigger than a predefined minimum value
```

We choose the initial modification factor to be $e \simeq 2.71828$ which allows us to quickly reach all possible energy levels. Too big or too small an initial value would cause either more interactive stages or more Monte Carlo steps for each stage. In order to obtain an accurate estimation of $g(E)$, the final f often should be smaller than $\exp(10^{-7}) \simeq 1.0000001$. As for the flatness criterion for $H(E)$, usually, making sure the minimum entry is no less than 60% of the average of all entries is stringent enough for relatively large systems.

2.3 General Purpose Computing on GPU

In Fig. 2.1, we show the typical architecture of a Graphic Processing Unit (GPU). A single GPU card contains several multiprocessors, and each multiprocessor contains many cores (processors). For each core, there is a very small private memory space for registers, and cores on the same multiprocessor share a public memory space. All the multiprocessors then share a bigger memory space, called global device memory. Cores can communicate with each other through the global memory, and neighboring cores on a multiprocessor can also exchange information by the shared memory, which is about two orders of magnitude faster than the global memory. There is a part of the device memory that only allows read access from a device function, called the constant memory, the speed of which is between the speeds of shared memory and the global memory. Currently, to fully utilize the computing power of a GPU, one has to explicitly deal with optimizing memory access.

For non-graphical applications, Compute Unified Device Architecture (CUDA) provides the Application Programming Interface (API) that drives the general purpose computing on a GPU. Therefore, on the programmer's end, the coding style is exactly C (or C++, Fortran) language with simple extension. The challenging part comes with the parallel algorithm and the optimization of the thread and memory hierarchy. Threads are grouped into blocks, and blocks into grids. A GPU device function must specify the block and grid configuration, which could affect the performance depending on resources of a specific GPU card. For more detail, please refer to the CUDA manual [16].

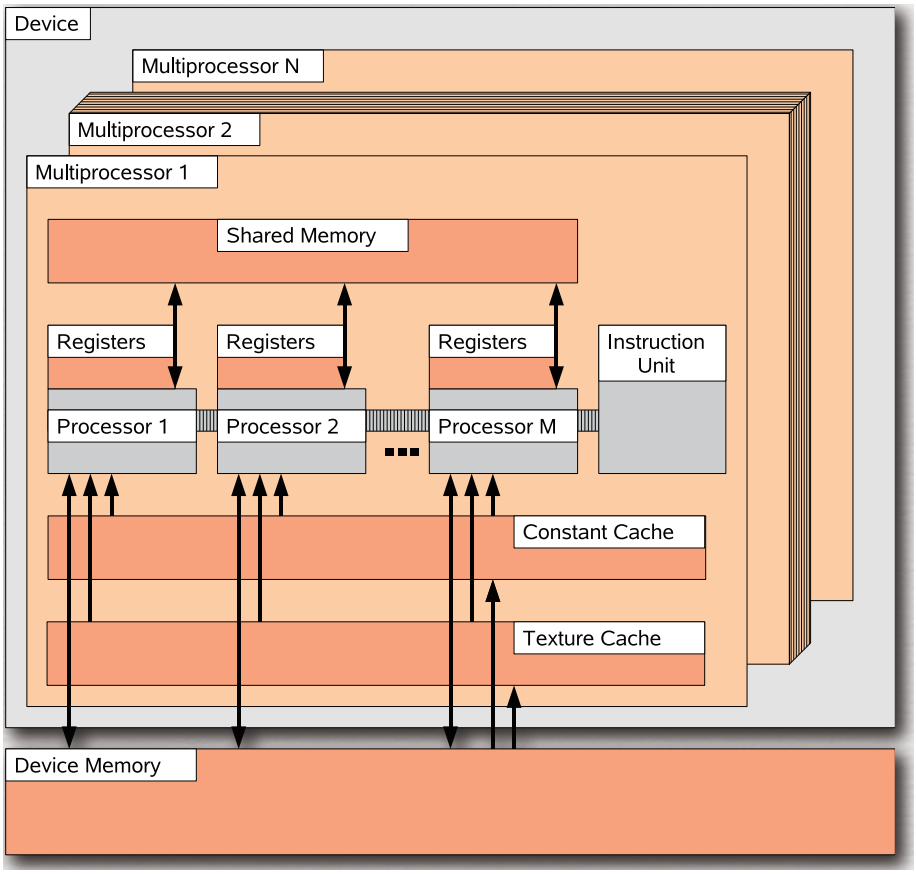

Figure 2.1: Architecture of a Graphic Processing Unit device. (Graph by T. Preis)

Table 2.1: Comparisons of the current generation CPU and GPUs.

Device	Peak Performance(GFLOPS)		Peak Memory Bandwidth (GB/s)
	Single Precision	Double Precision	
i7 930	66	33	25.6
GTX285	1063	86	159
Tesla S1070	1035	86	102
Tesla C2050	1030	515	144

To present the advantage of GPU computing, we list the peak performance and the peak memory bandwidth for a CPU and three GPUs that are used in the simulations throughout this dissertation in Table 2.1. In single precision, a GPU outperforms a CPU by more than an order of magnitude. This is also the case in double precision for the latest GPU with the Fermi architecture (Tesla C2050). As for the speed of memory access which is specially important for a parallel application, a GPU is also four to six times faster. Overall, it is possible for a GPU application to gain a speedup of almost two orders of magnitude.

2.4 Finite-size scaling analyses

In the computer simulation of lattice magnetic models (or equivalently, lattice gas models), we can estimate quantities such as magnetization M (first moment), susceptibility χ (second moment), fourth-order cumulant U_4 and specific heat C from finite size systems. Ultimately, we aim to study the critical behavior in the thermodynamic limit (infinite size systems), and the finite-size scaling theory [7] provides analyses to extract critical exponents $(\alpha, \beta, \gamma, \nu)$ from the data near the critical point.

The maximum slope of the fourth-order cumulant U_4 follows [8]

$$\left(\frac{dU_4}{dK}\right)_{max} = a' L^{\frac{1}{\nu}}(1 + b' L^{-\omega}), \tag{2.6}$$

where $K = \frac{J_{NN}}{k_B T}$ (J_{NN} is the nearest-neighbor coupling constant and T is the temperature) and the $L^{-\omega}$ term indicates the correction to scaling. The correlation length exponent ν can be estimated directly.

With the exponent ν and critical temperature T_c at hand, the critical exponent β and γ can be extracted from the data collapsing of the finite-size scaling forms,

$$M = L^{-\frac{\beta}{\nu}} \overline{X}(tL^{\frac{1}{\nu}}) \tag{2.7}$$

$$\chi T = L^{\frac{\gamma}{\nu}} \overline{Y}(tL^{\frac{1}{\nu}}) \tag{2.8}$$

where $t = |1 - \frac{T}{T_c}|$, and $\overline{X}$ and $\overline{Y}$ are universal functions whose analytical forms are not known. One can also estimate the exponent α from the relation of the peak position with lattice size for the specific heat

$$C_{max} = cL^{\frac{\alpha}{\nu}} + C_0 \tag{2.9}$$

where C_0 is the "background" contribution. In some cases when the appropriate paths, i.e. which are perpendicular to the phase boundary, are ones of constant temperatures, then the critical behavior would be expressed in terms of reduced field $h = |1 - \frac{H}{H_c}|$, and all the above analysis still applies.

Actually, these critical exponents are not completely independent of each other. Instead, they are bound by the hyper-scaling laws [9], which can be used as a way to check the self-consistency of the evaluated critical exponents.

2.5 Field Mixing Analyses

In general, the relevant scaling fields τ and h that characterize the critical behavior are comprised of linear combinations of the thermodynamic fields T and H as [10]

$$\tau = T - T_c + s(H - H_c) \tag{2.10}$$

$$h = H - H_c + r(T - T_c) \tag{2.11}$$

where s and r are parameters controlling the extent of field mixing. As a result, the associated conjugate scaling operators $\mathcal{E}$ and $\mathcal{M}$ are also linear combinations of the spin-spin interaction energy density u and the magnetization m as

$$\mathcal{E} = \frac{u - rm}{1 - rs} \tag{2.12}$$

$$\mathcal{M} = \frac{m - su}{1 - rs} \tag{2.13}$$

$$m = \frac{1}{N} \sum_i \sigma_i \tag{2.14}$$

$$u = \frac{1}{N} \left(\sum_{i \neq j} \sigma_i \sigma_j + R \sum_{i \neq k} \sigma_i \sigma_k + R_t \sum_{i \neq j \neq k} \sigma_i \sigma_j \sigma_k \right) \tag{2.15}$$

where N is the total number of spins. According to finite-size scaling [11], the joint probability distribution $p_L(\mathcal{E}, \mathcal{M})$ near criticality should obey the following scaling ansatz:

$$p_L(\mathcal{E}, \mathcal{M}) \simeq \Lambda_{\mathcal{M}}^{+} \Lambda_{\mathcal{E}}^{+} \tilde{p}_{\mathcal{E},\mathcal{M}}(\Lambda_{\mathcal{M}}^{+} \delta\mathcal{M}, \Lambda_{\mathcal{E}}^{+} \delta\mathcal{E}, \Lambda_{\mathcal{M}} h, \Lambda_{\mathcal{E}} \tau) \tag{2.16}$$

where,

$$\Lambda_{\mathcal{E}} = a_{\mathcal{E}} L^{1/\nu}, \quad \Lambda_{\mathcal{M}} = a_{\mathcal{M}} L^{d-\beta/\nu} \tag{2.17}$$

$$\Lambda_{\mathcal{E}} \Lambda_{\mathcal{E}}^{+} = \Lambda_{\mathcal{M}} \Lambda_{\mathcal{M}}^{+} = L^d \tag{2.18}$$

$$\delta\mathcal{M} = \mathcal{M} - < \mathcal{M} >_c, \quad \delta\mathcal{E} = \mathcal{E} - < \mathcal{E} >_c \tag{2.19}$$

The subscript c denotes that the averages are taken at criticality. For appropriate choices of the nonuniversal factors $a_{\mathcal{E}}$ and $a_{\mathcal{M}}$, the function $\tilde{p}_{\mathcal{E},\mathcal{M}}$ would be universal. After integration over $\mathcal{E}$, exactly at criticality, where $h = \tau = 0$, one has

$$p_L(\mathcal{M}) \simeq a_{\mathcal{M}}^{-1} L^{\beta/\nu} \tilde{p}_{\mathcal{M}}^{*}(L^{\beta/\nu} a_{\mathcal{M}}^{-1} \delta\mathcal{M}) \tag{2.20}$$

where the function $\tilde{p}_{\mathcal{M}}^{*}$ characterizes the universality class, the form of which has been well established for the two-dimensional Ising model.

Chapter 3

Ising model with antiferromagnetic nearest-neighbor and next-nearest-neighbor interactions

3.1 Background and Motivation

For the nearest-neighbor (NN) Ising antiferromagnet on the square lattice in a uniform magnetic field, the low temperature ordered phase is separated from the paramagnetic phase by a simple, 2nd order phase boundary. (Within the context of the lattice gas model this system could be described as having repulsive NN-coupling and forming a $c(2 \times 2)$ ordered state.) With the addition of repulsive (antiferromagnetic) next-nearest-neighbor (NNN) interactions the situation becomes more complicated. Early Monte Carlo simulations suggested that a single, super-antiferromagnetic, or (2×1), phase existed, separated from the paramagnetic phase by a single phase boundary [17, 18, 19]. A degenerate, row-shifted (2×2) state was also predicted at zero temperature. (See Fig. 3.1 for a schematic representation of these states.) On the other hand, symmetry arguments based on Landau theory [20] predict the order-disorder transitions of (2×1) and (2×2) structures belong to the XY model with cubic anisotropy.

The original motivation of this study was to investigate the possibility of XY-like behavior of the Ising spins on the square lattice, since there is numerical evidence [21] that an Ising antiferromaget with attractive NNN interaction on the triangular lattice has an XY-like intermediate state between the low temperature ordered state and high temperature disordered state. In fact, the present model has already been studied by many authors using different approaches. An early Monte Carlo study [19] comprehensively showed the phase diagrams for several different interaction ratios (R) of NNN to NN interaction. But due to

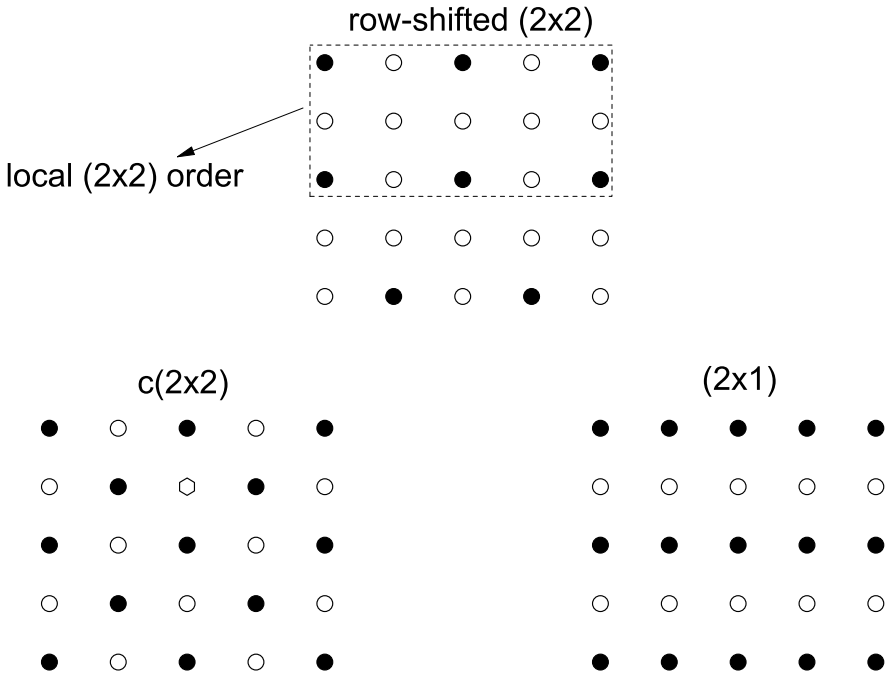

Figure 3.1: Schematic plots of $c(2 \times 2)$, (2×1) and row-shifted (2×2) ordered structures within the context of the lattice gas model. (In magnetic language, filled circles correspond to up spins and empty circles correspond to down spins.)

the deficiencies of computer resources at that time, for the $R = 1$ case, a disordered region was missed between the two ordered phases which was pointed out in a later interfacial free energy study [22].

Meanwhile, transfer matrix studies [23, 24] found reentrant behavior for the (2×1) transition lines. While a study using the cluster variation method [25, 26] concluded that for a range of R (0.5∼1.2) the system undergoes a first order transition, a recent Monte Carlo study [28] using a variant of the Wang-Landau method [3] focused on the $R = 1$ case without external field and found the phase transition is of second order. For external field $H = 4$ (in the unit of NN interaction constant) the two ordered phases, namely (2×1) and

row-shifted (2×2), are degenerate at zero temperature so it is tempting to think that the cubic anisotropy would be zero for this field and that there could be a Kosterlitz-Thouless transition.

In this chapter, we will present comprehensive results from large-scale Monte Carlo simulations, and try to put an end to this debate. In Sec. 3.2 to 3.4 the model, relevant methods and analysis techniques are reviewed. Our results are presented in Sec. 3.5 to 3.7, along with finite size scaling analyses, and we summarize and conclude in Sec. 3.8.

3.2 Model hamiltonian and order parameter

The Ising model with NNN interaction is described by the Hamiltonian

$$\mathcal{H} = J_{NN} \sum_{<i,j>_{NN}} \sigma_i \sigma_j + J_{NNN} \sum_{<i,j>_{NNN}} \sigma_i \sigma_j + H \sum \sigma_i, \tag{3.1}$$

where $\sigma_i, \sigma_j = \pm 1$, J_{NN} and J_{NNN} are NN and NNN interaction constants, respectively, H is an external magnetic field, and the sums in the first two terms run over indicated pairs of neighbors on a square lattice with periodic boundary conditions. Both J_{NN} and J_{NNN} are positive (antiferromagnetic) and the ratio $R = J_{NNN}/J_{NN}$.

For the $R = 1$ case, the ground states would be the (2×1) state, also known as the super-antiferromagnetic state, in small magnetic fields; and at higher fields it would be a row-shifted (2×2) state, which differs from the (2×2) state in the sense that the antiferromagnetic chains in the former state can slide freely without energy cost (See Fig. 3.1). Locally, the structure may appear to be (2×2), but for large enough lattices the equilibrium structure always shows row shifting. As a result, such a row-shifted (2×2) state is highly degenerate, and the antiferromagnetic sublattice exhibits only one dimensional long range order. The change of ground states with the external field for all possible R value is plotted in Fig. 3.2. In terms of the sublattice magnetizations

$$M_\lambda = \frac{4}{N} \sum_{i \in \lambda} \sigma_i, \quad \lambda = 1, 2, 3, 4 \quad , \tag{3.2}$$

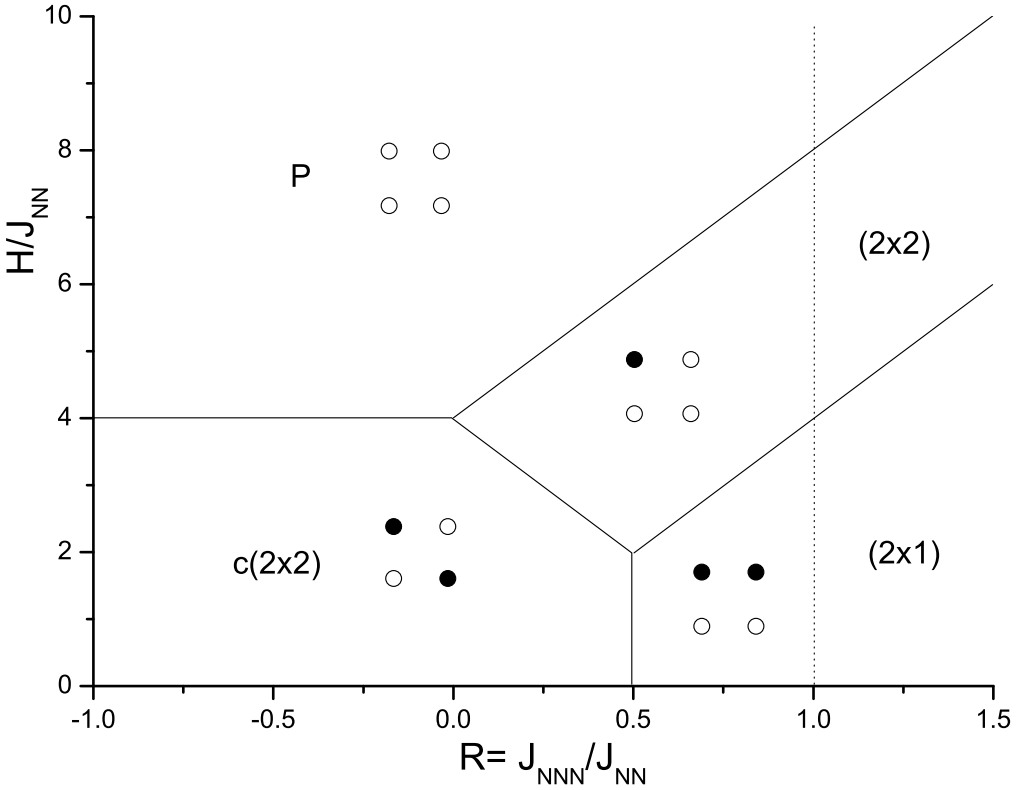

Figure 3.2: Ground states for different external fields H and interaction ratios R.

we can define two components of the order parameter for the (2×1) state

$$M^a = [M_1 + M_2 - (M_3 + M_4)]/4, \tag{3.3}$$

$$M^b = [M_1 + M_4 - (M_2 + M_3)]/4, \tag{3.4}$$

with a computationally convenient root-mean-square order parameter

$$M^{rms} = \sqrt{(M^a)^2 + (M^b)^2}. \tag{3.5}$$

Since M^{rms} would have a limiting value of $\frac{1}{2}$ for the row-shifted (2×2) state and be zero for the disordered state, it can also be used as an order parameter for the row-shifted (2×2) state.

Other observables, such as the finite lattice ordering susceptibility χ and fourth-order cumulant U_4, are defined in terms of the order parameter M^{rms} as

$$\chi = \frac{N}{T}\left[< (M^{rms})^2 > - < M^{rms} >^2\right] \tag{3.6}$$

$$U_4 = 1 - \frac{< (M^{rms})^4 >}{3 < (M^{rms})^2 >^2} \tag{3.7}$$

where N is the total number of spins and T is the simulation temperature. In some cases, the true ordering susceptibility χ^+, which is $\frac{N}{T} < (M^{rms})^2 >$, is used to eliminate simulation errors resulting from $< M^{rms} >$, where the order parameter is known to be zero for the infinite lattice.

3.3 Preliminary study

For small lattice sizes, Wang-Landau sampling [3] was used to obtain a quick overview of the thermodynamic behavior of our model. A two-dimensional random walk in energy E and magnetization M space was performed so that the density of states $g(E, M)$ could be used to determine all thermodynamic quantities (derived from the partition function) for any value of temperatures and external field. Consequently, "freezing" problems are avoided at

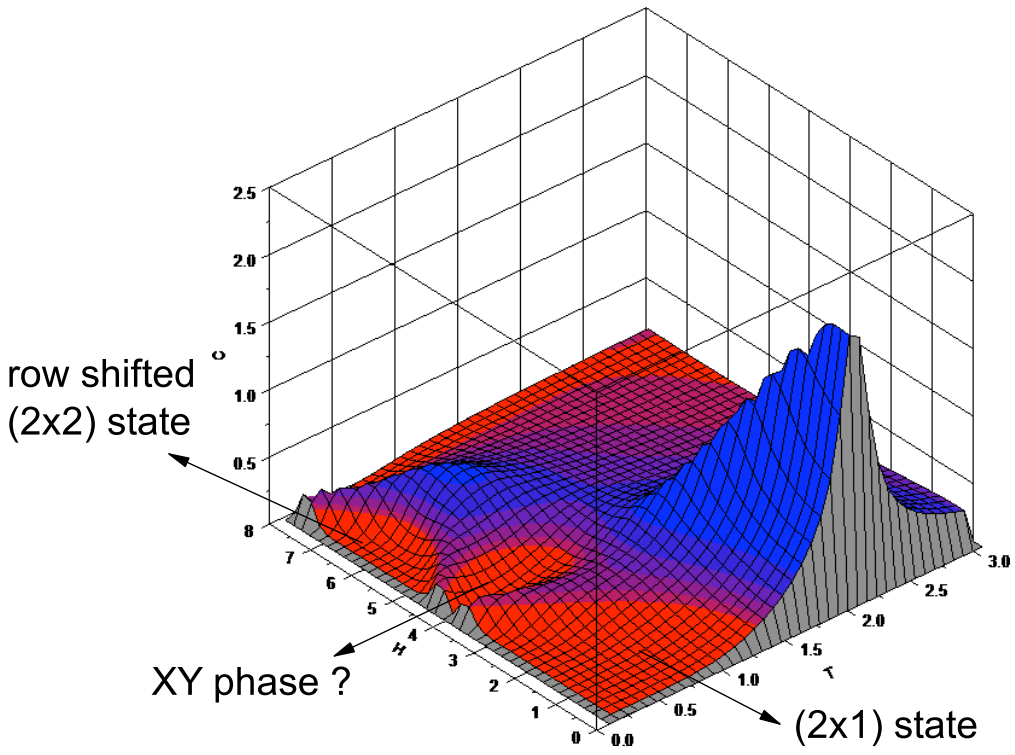

Figure 3.3: 3D plot of the specific heat c versus the external field H and the temperature T with lattice size $L = 20$ for interaction ratio $R = 1$.

extremely low temperatures. This allowed us to determine the "interesting" regions of field-temperature space. In Fig. 3.3, we show the 3D plot of specific heat for both the external field and temperature. We expect a low field (2×1) state and a high field row-shifted (2×2) state according to the ground state analyses (See Fig. 3.2). Interestingly, there appears to be an intermediate region in between. We then calculate the fourth order cumulant of (2×1) order parameter with temperatures across this region (See Fig. 3.4). For lattice sizes we considered and the resolution we had, it appears that all the curves merge into a single line below a certain temperature, which could indicate a XY transition for the system. To confirm that, we need much larger lattice sizes and higher resolution.

However, it quickly became apparent that, because of subtle finite size effects, quite large lattices would be needed. Unfortunately, as L increases, the number of entries in the histogram in Wang-Landau sampling explodes as L^4 and it proved to be more efficient to use parallel tempering instead.

Since a large portion of interesting phase boundary is at relatively low temperatures and many local energy minima exist which makes the relaxation time rather long, the parallel tempering method described in Section 2.1, is a good choice for simulating our model.

We chose the temperatures for the replicas to be in a geometric progression [29], which would make acceptance rates relatively constant among neighboring temperature pairs, and the total number of temperatures was chosen to make the average acceptance rate above 20%. As shown in Fig. 3.5, the overlapping of the energy histogram of the neighboring replica is essential for the parallel tempering method to work. In Fig. 3.6, we plot the time series of M^{rms} for parallel tempering and Metropolis simulation with ordered or disordered initial states, respectively. At low temperatures, parallel tempering can quickly approach equilibrium, even with disordered initial states, and has much shorter equilibrium relaxation time. We calculate the linear relaxation function $\phi_A(t) = \frac{<A(0)A(t)> - <A>^2}{<A^2> - <A>^2}$, and plot it in Fig. 3.7. It indicates that with the same amount of Monte Carlo steps, parallel tempering can provide many more independent configurations of the system in equilibrium.

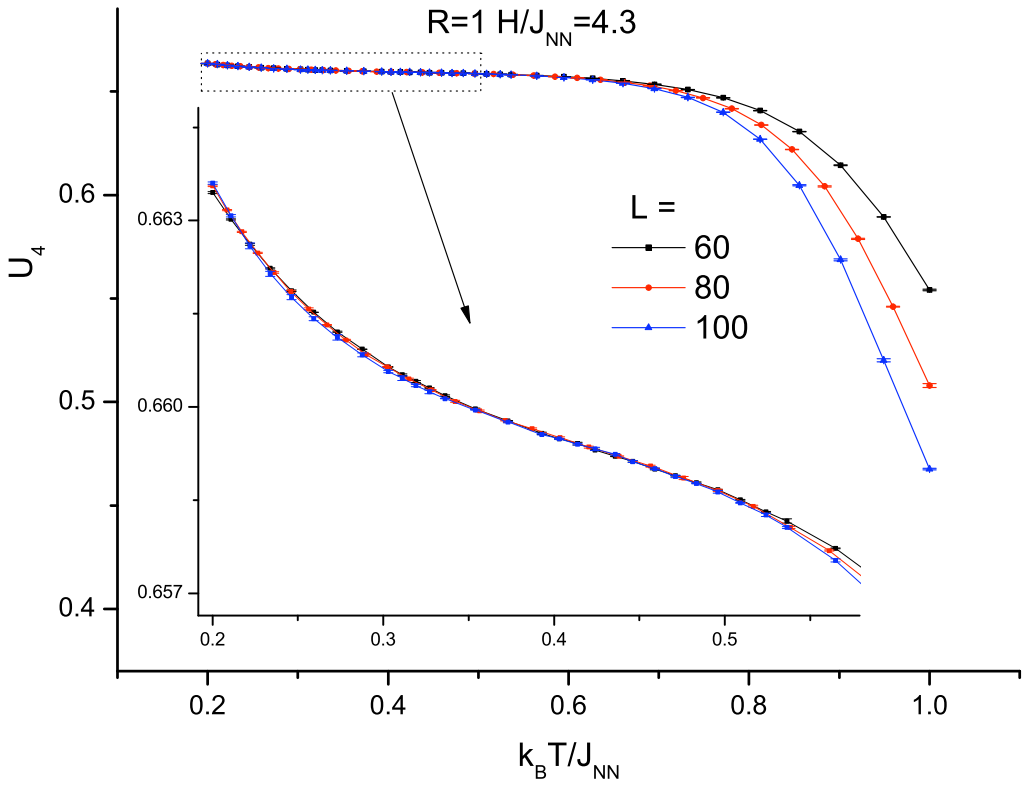

Figure 3.4: Fourth order cumulant U_4 for the (2×1) order parameter for lattice size $L = 60, 80, 100$.

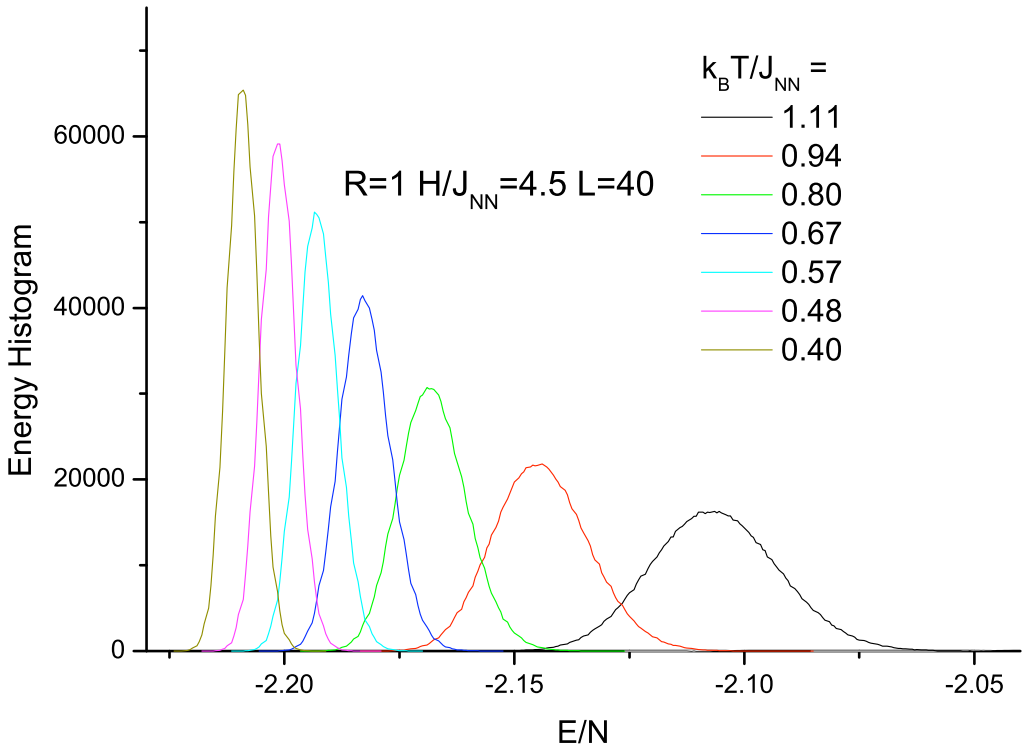

Figure 3.5: The energy histogram for a temperature set in one of the parallel tempering simulation.

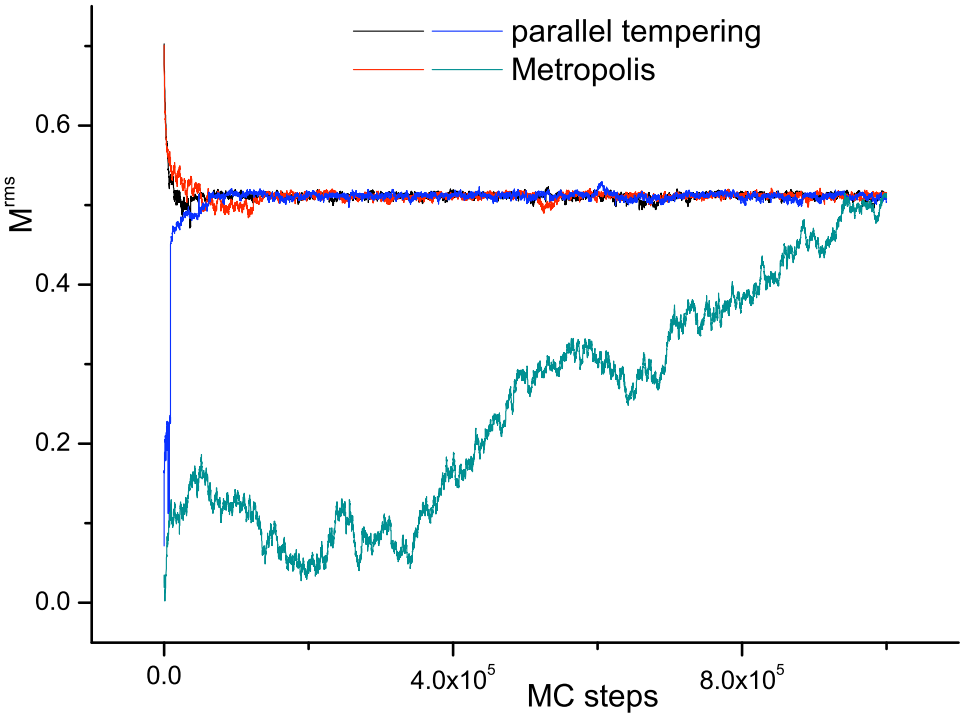

Figure 3.6: Time series of order parameter M^{rms} for parallel tempering and Metropolis simulation from disordered and ordered initial states, respectively. For the definition of Monte Carlo (MC) steps, see Section 2.1.

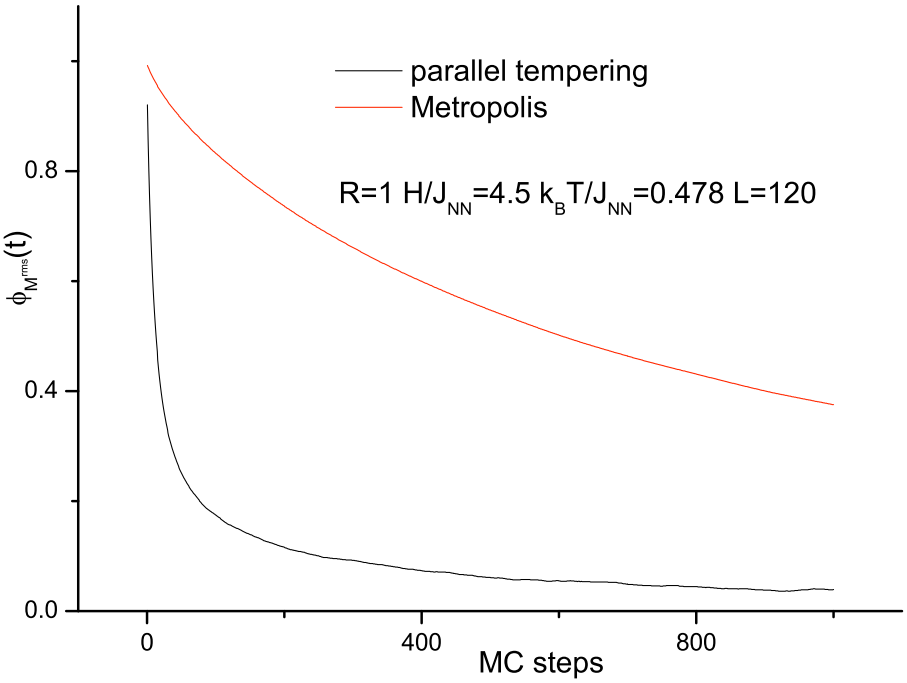

Figure 3.7: Equilibrium linear relaxation $\phi_{M^{rms}}(t)$ for parallel tempering and Metropolis simulation.

The multiplicative, congruential random number generator RANECU was used [30, 31], and some results were also obtained using the Mersenne Twister [32] for comparison. No difference was observed to within the error bars.

Typically, data from 10^6 to 10^7 MC steps were kept for each run and 3 to 6 independent runs are taken to calculate standard statistical error bars. For parallel tempering, the swap trial was attempted after every MC steps. In all the plots of data and analysis shown in following sections, if error bars are not shown they are always smaller than the size of the symbols.

3.4 GPU acceleration

General purpose computing on graphics processing units (GPU) attracts steadily increasing interest in simulational physics [33, 34, 5], since the computational power of a recent GPU exceeds that of a central processing unit (CPU) by orders of magnitude. The advantage continues to grow as the performance of GPU's doubles every year. Recently, a GPU accelerated Monte Carlo simulation of Ising models [5] was performed. Compared to traditional CPU calculations, the speedup was about 60 times.

The idea behind the implementation in Ref. [5] can be easily extended to our model, and the parallel tempering algorithm is naturally realized. Initially, all the replicas are loaded into the global memory of the GPU. For each replica, the entire lattice is divided into four sublattices, then spins in the same sublattice can be updated simultaneously by the GPU using a Metropolis scheme, and the swap of configurations of replica pairs can also be achieved in parallel by using thousands of threads to exchange spin values at the same time.

There are two levels of parallelism: on system-specific level, since the interactions are short-ranged and the spins are fixed on the lattice, it is natural to divide the lattice into blocks and use a checkerboard scheme to update different blocks simultaneously; on an algorithm level which is system-independent, all replicas can perform sampling at the same time and exchange information at certain Monte Carlo steps. Such an implementation is

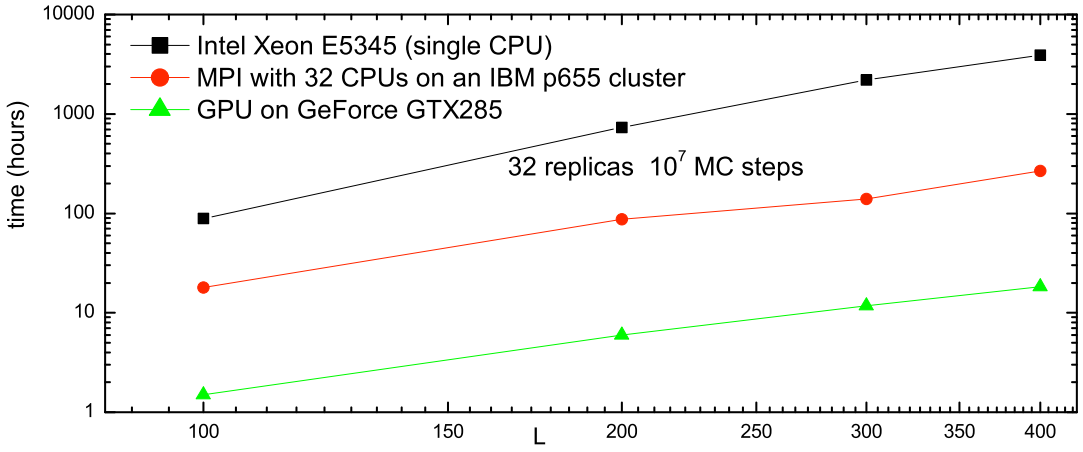

Figure 3.8: Computing time versus lattice size for an Intel Xeon E5345 CPU, 32 CPUs with Message Passing Interface (MPI) on an IBM p655 cluster, and a Nvidia GeForce GTX285 GPU for the parallel tempering simulation of 32 replicas up to 10^7 Monte Carlo steps. Data points for the Xeon CPU are linearly extrapolated from simulations with thousands of Monte Carlo steps.

extremely efficient. On the GeForce GTX285 graphics unit, our code runs about 10 times faster than on 32 CPUs of an IBM p655 cluster using Message Passing Interface (MPI) for parallel computation (See Fig. 3.8).

3.5 Phase diagram and short range ordering

From the ground state analysis for sufficiently small magnetic field, the ordered state would be the superantiferromagnetic, or (2×1), structure. As the external field increases to $4 < H/J_{NN} < 8$, more spins align in the opposite field direction, and the ordered structure would be the row-shifted (2×2). With even stronger fields, the system becomes paramagnetic. In the region near $H/J_{NN} = 4$, a mixture of (2×1) and row-shifted (2×2) is visible.

For finite temperatures, we found that the fourth-order cumulant is always a good quantity to use to locate the transition points, while the data for other quantities, such as the

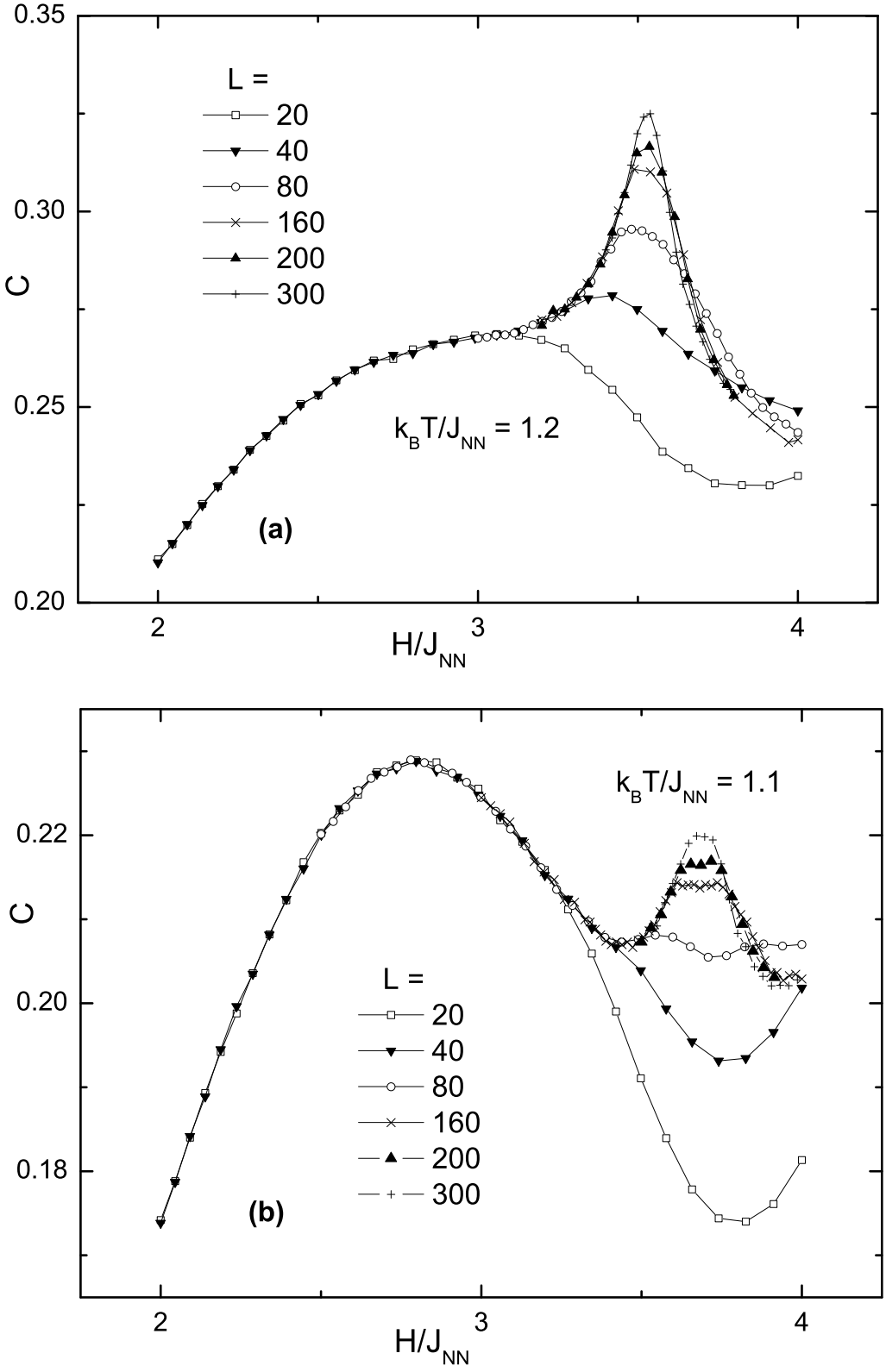

Figure 3.9: Variation of the specific heat C versus field H with lattice sizes $L = 20, 40, 80, 160, 200, 300$ for paths of constant: (a) $k_BT/J_{NN} = 1.2$ and (b) $k_BT/J_{NN} = 1.1$.

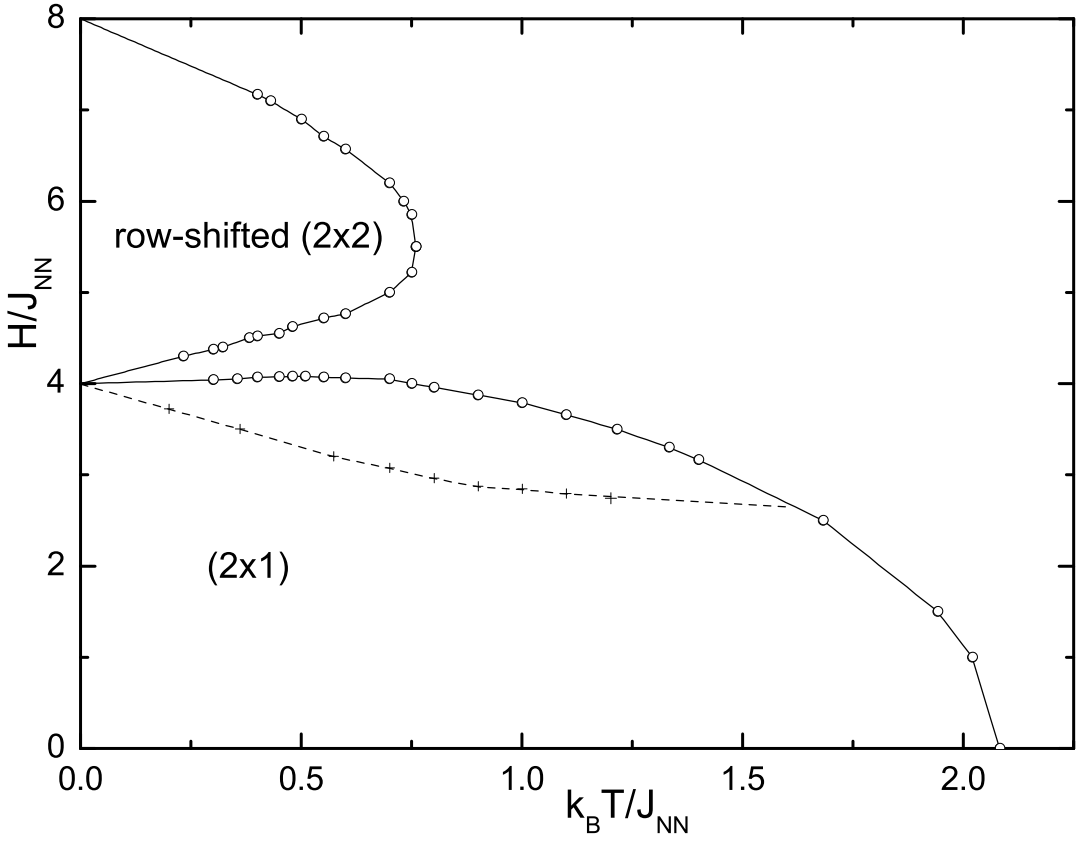

Figure 3.10: The phase diagram for the Ising square lattice with antiferromagnetic nearest- and next-nearest-neighbor interactions in a magnetic field for $R = 1$. Open circles and pluses denote points where we perform simulations. The solid lines are second order transition lines, while the dashed line indicates the "short range ordering" line.

specific heat or susceptibility, may look "strange". As shown in Fig. 3.9, for paths of constant temperature $k_BT/J_{NN} = 1.1$ and 1.2, there are two peaks for the specific heat, and the one that increases with lattice size corresponds to the critical point. Due to the effect of neighboring critical points, this peak can not be observed for small lattice sizes. To help the reader understand the observed thermodynamic properties, the final phase diagram for $R = 1$ is plotted in Fig. 3.10: The solid lines are the phase boundaries, all of which are continuous. The dashed line inside the (2×1) ordered phase indicates a "short range ordering" line, which was located from the peak position of the specific heat.

An indication of the complexity of the finite size behavior is clearly seen in the bottom portion of Fig. 3.9 in which the small lattices actually have minima in the specific heat for field values that eventually show phase transitions for sufficiently large systems. The round-shaped size-independent peak is due to the short range ordering of the (2×1) "clusters" of different orientation from the ordered background. No corresponding behavior was observed from the susceptibility or the fourth-order cumulant.

In order to confirm the above argument, we also calculated the NN and NNN pair correlation function, that is $< \sigma_i \sigma_j >$, for paths of different fields crossing this line. The correlation function data are plotted in Fig. 3.11. The NN pair correlation decreases from zero to a minimum and then increases to positive values, while the NNN pair correlation increases monotonically from -1.

3.6 Critical behavior

The data for the specific heat and susceptibility for three different values of H are plotted in Fig. 3.12.

Without the field, they both show very sharp peaks, and from the magnitudes of the peak values of the specific heat, as shown in Fig. 3.13(b), we can obtain a rather accurate estimate of the exponent ratio $\alpha/\nu = 0.357(8)$, which is obviously not zero. In Fig. 3.13(a),

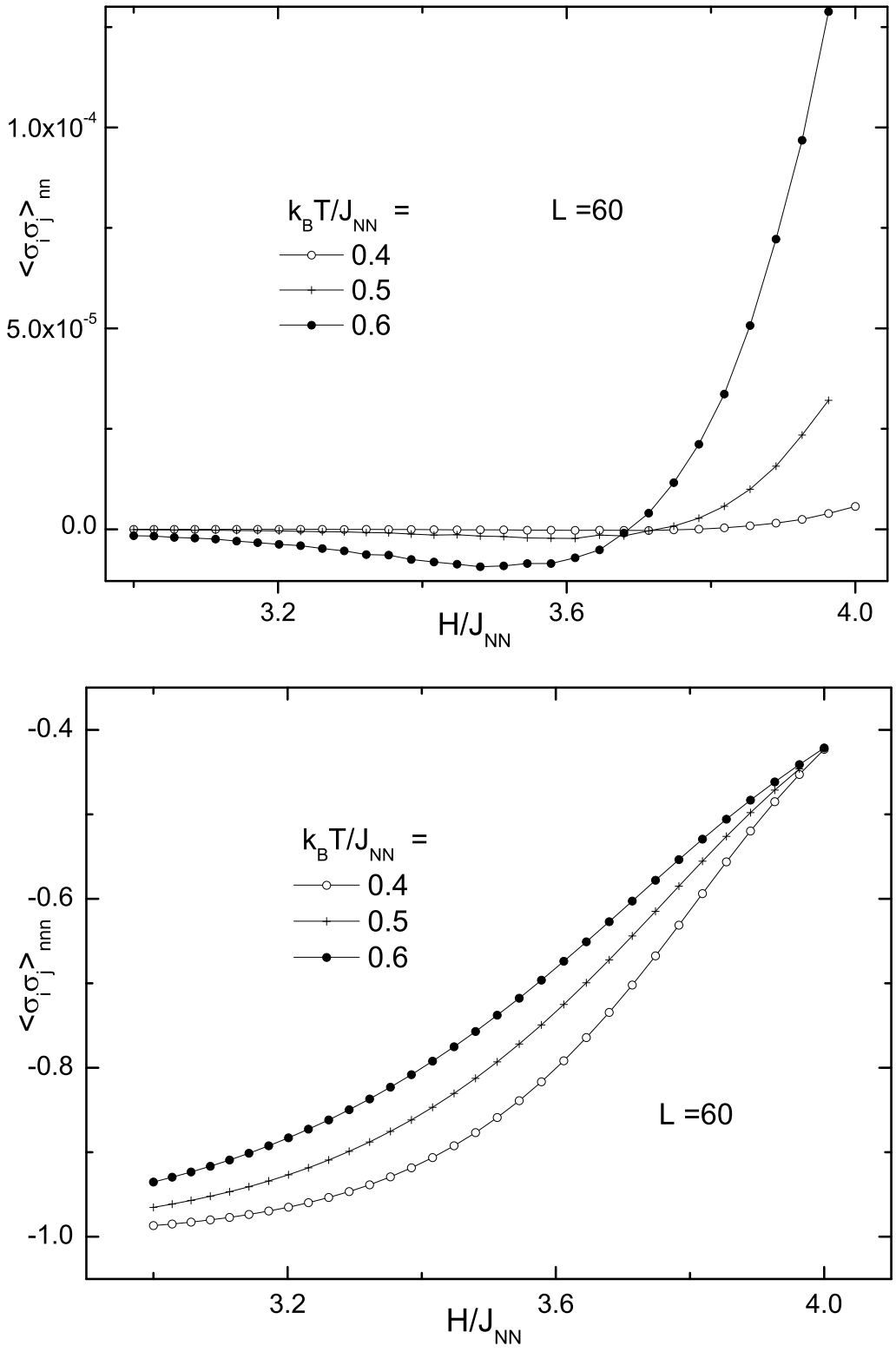

Figure 3.11: Nearest- and next-nearest-neighbor correlation functions. The field is varied for paths of 3 different temperatures: $k_BT/J_{NN} = 0.4, 0.5$ and 0.6 across the short range ordering line.

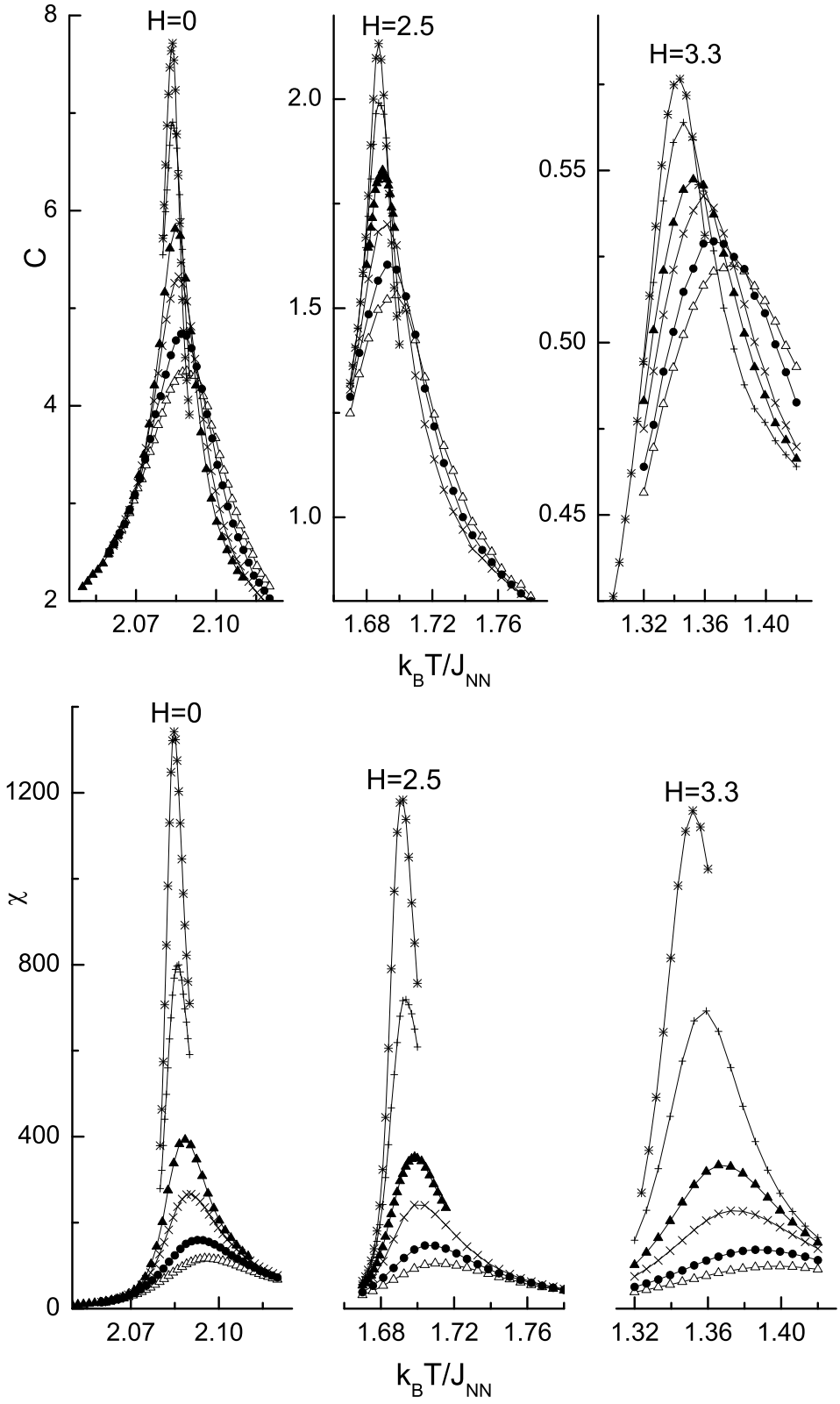

Figure 3.12: Specific heat and susceptibility for 3 different fields across the phase boundary. Data are for: $L = 100$, $\triangle$; $L = 120$, $\bullet$; $L = 160$, $\times$; $L = 200$ $\blacktriangle$; $L = 300$ $+$; $L = 400$ $*$.

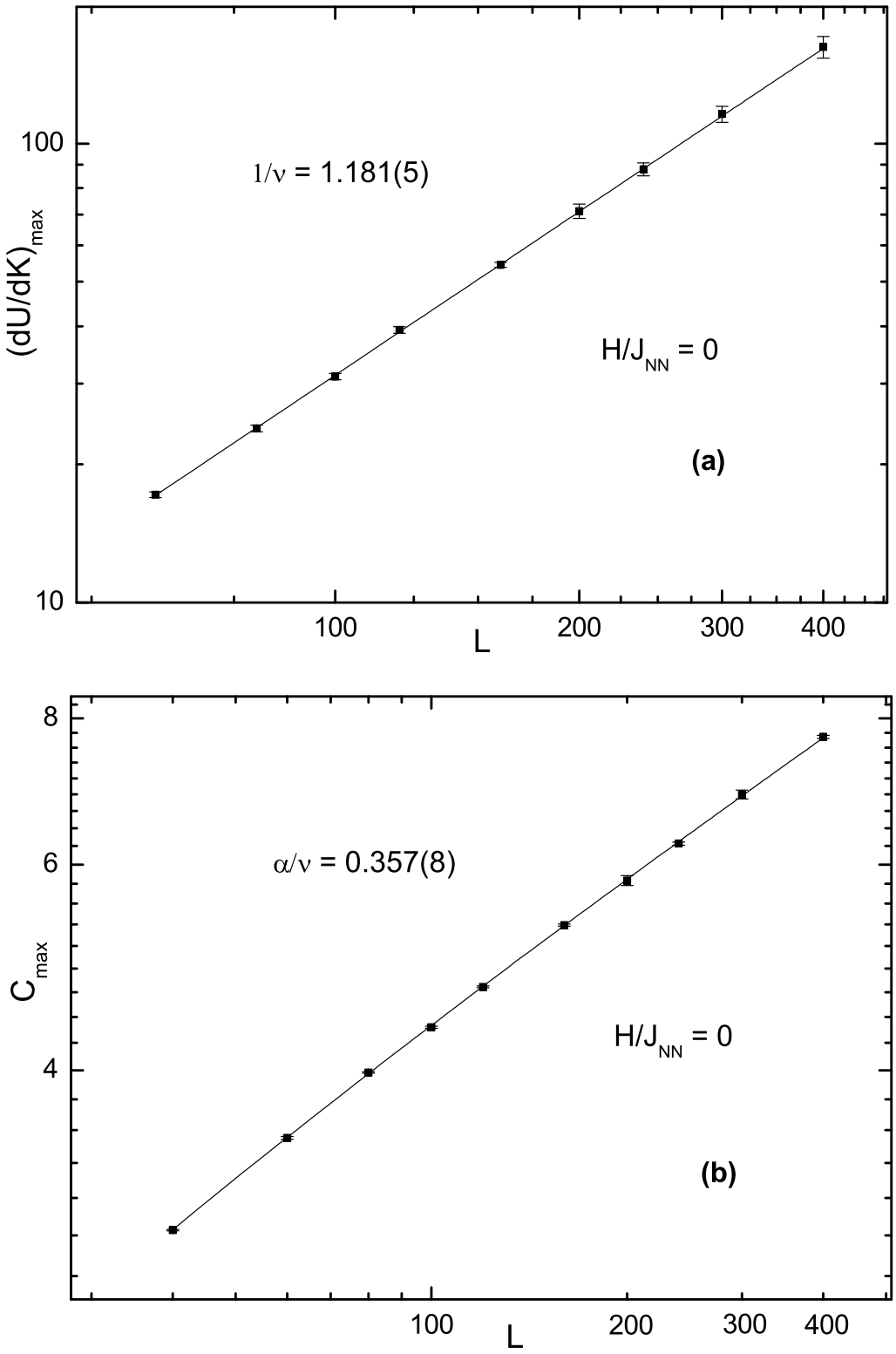

Figure 3.13: Curve fits using the leading terms of equations 2.6 and 2.9 for the maximum slopes of $\frac{dU}{dK}$ (a) and peak values of specific heat (b), respectively.

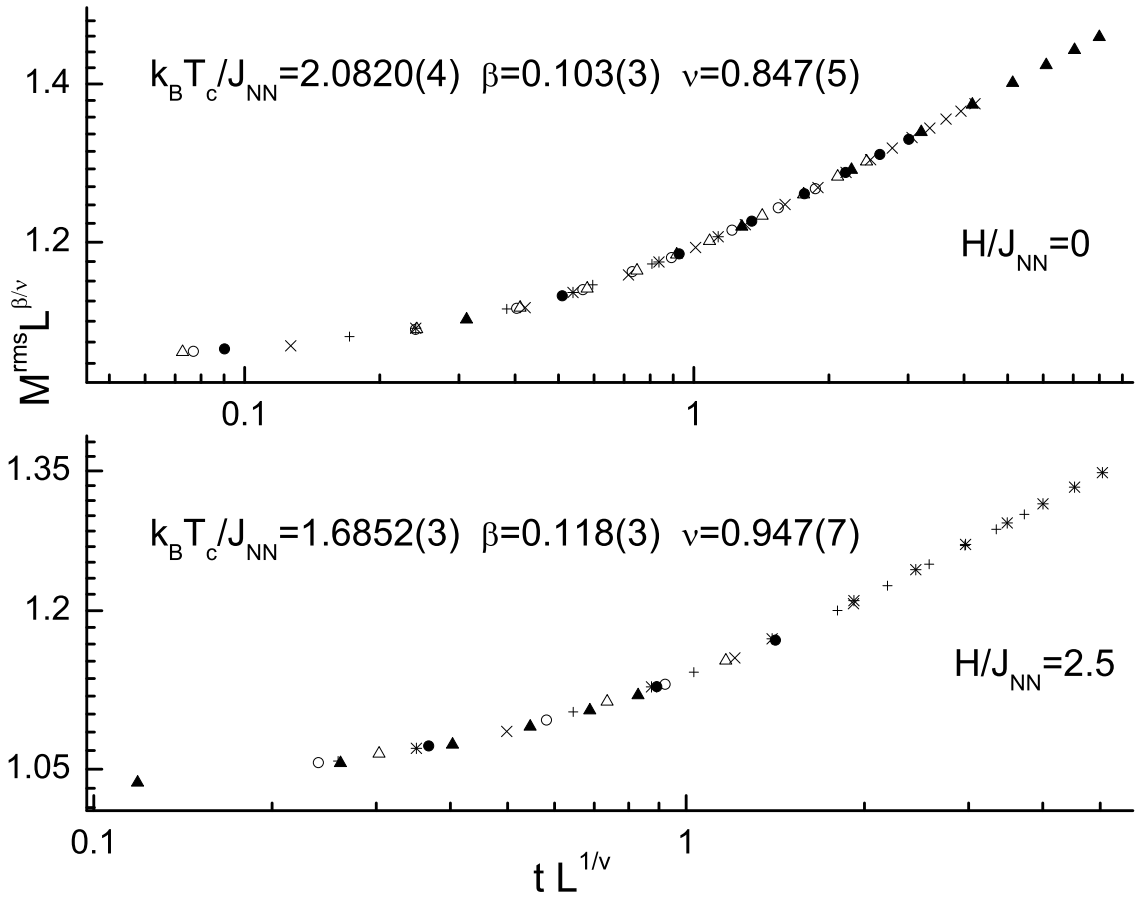

Figure 3.14: Finite size scaling data collapsing along paths of constant $H/J_{NN} = 0$ and 2.5 for root-mean-square order parameter. $t = (1 - \frac{T}{Tc})$. Data are for: $L = 80$, $\circ$; $L = 100$, $\triangle$; $L = 120$, $\bullet$; $L = 160$, $\times$; $L = 200$ $\blacktriangle$; $L = 300$, $+$; $L = 400$, $*$.

we also show the curve-fit for the maximum slope of $\frac{dU}{dK}$ for $H = 0$, and extract the exponent $\nu = 0.847(4)$ directly.

Both values of α and ν are quite consistent with the early estimates in Ref. [19], and the value of α/ν is different from Ref. [28], in which the $1/L$ correction term was assumed up to lattice size $L = 160$.

The same procedure was repeated for $H/J_{NN} = 2.5$ and 3.3, however, as shown in Fig. 3.12, the peaks of the specific heat become increasingly rounded as the field increases, which makes it rather difficult to get a direct estimate of the exponent α. Because of this it was necessary to obtain data for much larger lattice sizes, a task that was only possible with the use of GPU computing. In fact, as the value of ν increases with the field, for $H/J_{NN} = 3.3$, according the hyper-scaling law $\alpha = 2 - d\nu$, where $d = 2$ is the dimension of the system, α should be negative which implies a non-diverging specific heat. Although the curve-fit is not stable, given the value of α we can get a fit within error bars. Such continuous increasing of the exponent ν up to values much greater than one is actually consistent with the findings of an early transfer-matrix study [24].

To estimate the critical exponents β and γ, we performed data collapsing with a large range of lattice sizes for the order parameter and its susceptibility. As shown in Fig. 3.14 and Fig. 3.15, the data in both finite size scaling plots collapse very nicely onto single curves, and the ratio β/ν and γ/ν agree with values of Suzuki's weak universality [27] within error bars.

Hence, although the individual exponents are non-universal, Suzuki's weak universality holds quite well. Another set of data collapsing along the path of constant $H/J_{NN} = 6$ across the phase boundary of the row-shifted 2×2 state is shown in Fig. 3.16. The quality of the data collapsing is also excellent, and again, the exponents are non-universal. The estimate for β/ν is about 25% low, but γ/ν agrees well with the prediction of weak universality.

In Table. 3.1, the critical points and exponents α, β, ν and γ for several typical paths of constant H or T across the phase boundary are listed.

3.7 Reentrance behavior

Close to the region between the two ordered phases the correlation length exponent ν turns out to be quite large, and correspondingly the location of the critical points becomes very difficult to determine. In addition, the specific heat curves look "strange", see Fig. 3.9, and

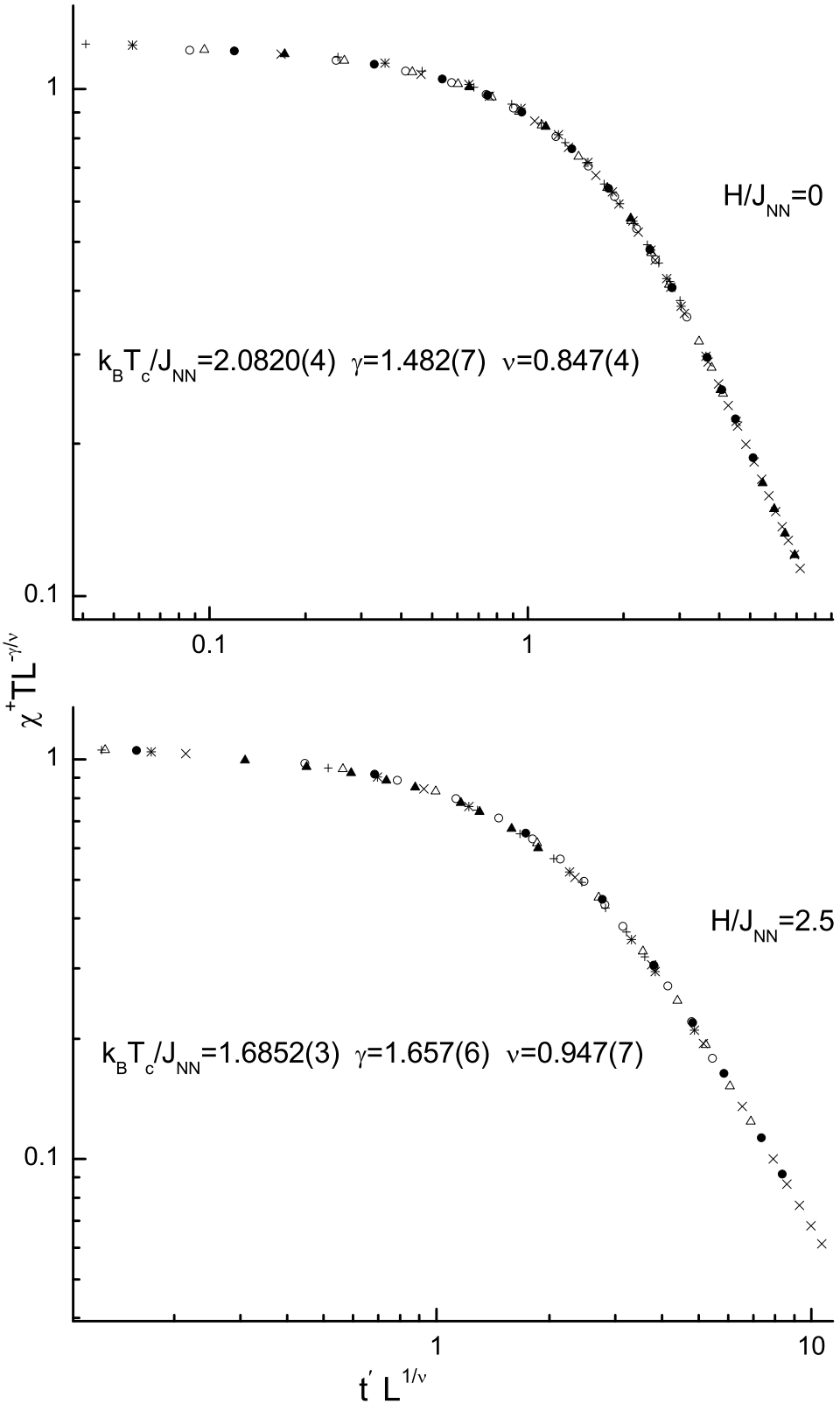

Figure 3.15: Finite size scaling data collapsing along paths of constant $H/J_{NN} = 0$ and 2.5 for the ordering susceptibility. $t' = |1 - \frac{T_c}{T}|$. Data are for: $L = 80$, $\circ$; $L = 100$, $\triangle$; $L = 120$, $\bullet$; $L = 160$, $\times$; $L = 200$ $\blacktriangle$; $L = 300$, $+$; $L = 400$, $*$.

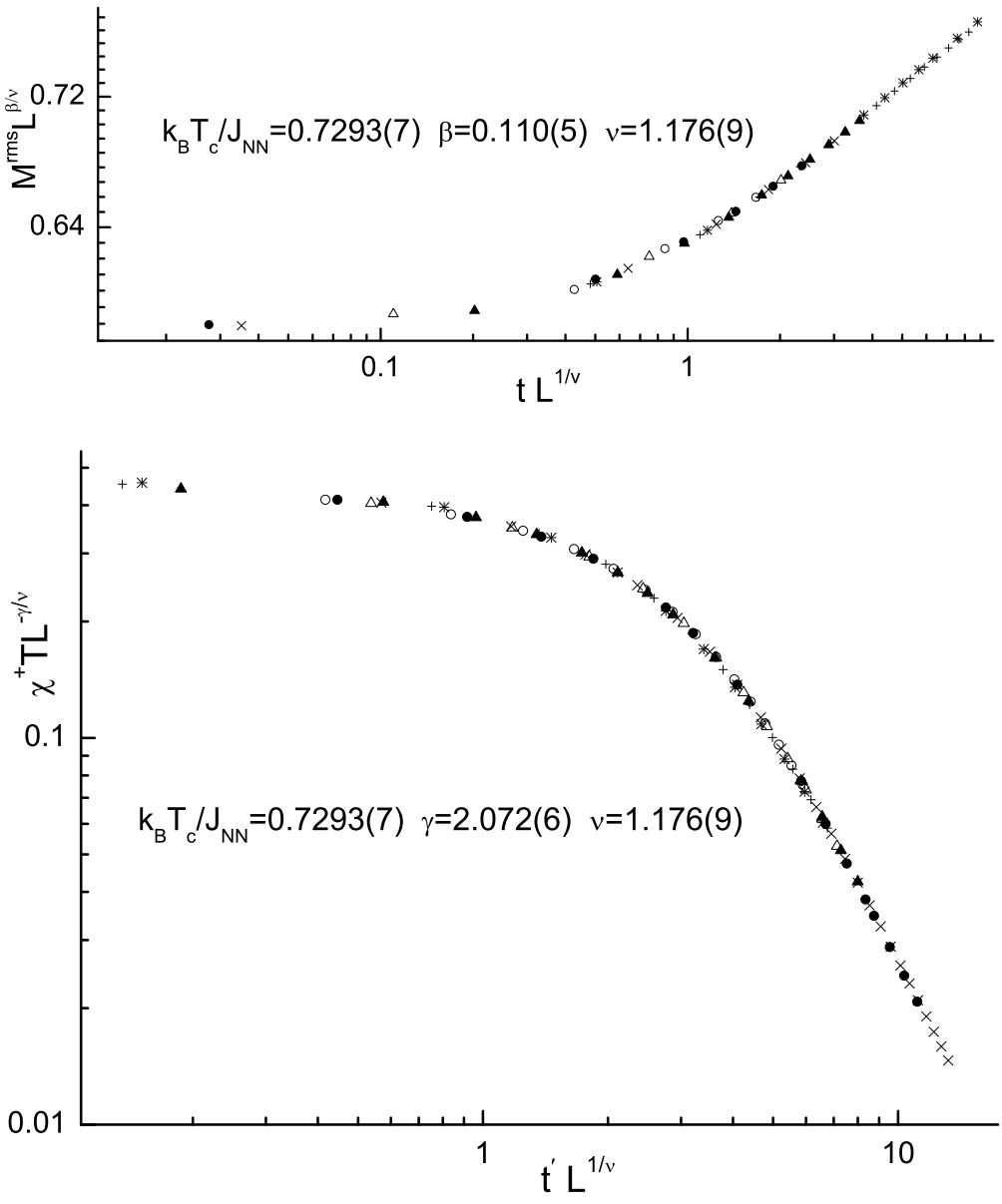

Figure 3.16: Data collapsing along the path of constant $H/J_{NN} = 6$ for root-mean-square order parameter and its ordering susceptibility, respectively. Data are for: $L = 80$, ○; $L = 100$, △; $L = 120$, ●; $L = 160$, ×; $L = 200$ ▲; $L = 300$, +; $L = 400$, ∗.

order	*path*	T_c or H_c	α	β	γ	ν	β/ν	γ/ν
	$H = 0$	2.0820(4)	0.302(7)	0.103(3)	1.482(7)	0.847(4)	0.122(4)	1.750(12)
2×1	$H = 2.5$	1.6852(3)	0.104(19)	0.118(3)	1.657(6)	0.947(7)	0.125(3)	1.750(14)
	$H = 3.3$	1.3335(6)		0.130(5)	1.930(6)	1.102(8)	0.118(5)	1.751(14)
$2 \times 2^*$	$H = 6$	0.7293(7)		0.110(5)	2.072(6)	1.176(9)	0.094(4)	1.762(14)
	$T = 0.5$	6.848(5)		0.126(4)	1.775(5)	1.02(2)	0.124(5)	1.740(34)

Table 3.1: Values of critical point temperatures or magnetic fields and corresponding critical exponents for several paths of constant temperature or field across the phase boundary of the (2×1) and *row-shifted (2×2) ordered phases.

since the specific heat exponent α would have a negative value, a limiting peak value is expected. However, it seems larger lattice sizes are needed. Since Suzuki's weak universality seems to hold along the transition line, we fixed the values of $\beta/\nu = 0.125$ and $\gamma/\nu = 1.75$ for the data collapsing analysis to get a better estimate of the critical point and the exponent ν.

As shown in Fig. 3.17, the crossing point of the fourth order cumulant curves for a path of constant $k_BT/J_{NN} = 0.7$ is slightly above $H/J_{NN} = 4$, and from the data collapsing analysis, see Fig. 3.18, we obtained an estimate of the critical point to be $H_c/J_{NN} = 4.052(7)$. Hence, we confirm the reentrant behavior of the (2×1) transition line, which could result from the appearance of (2×2) "clusters" that help to sustain the (2×1) order at low temperatures even when the external field is slightly bigger than 4.

For the paths of constant $k_BT/J_{NN} = 0.45$, the curves of the fourth order cumulant shows two crossing points and the finite size effect is quite obvious (See Fig. 3.19).

For the larger lattice size, the two crossing points move towards lower fields but they do not approach each other. Thus, a region of disorder remains between the two different ordered states, even down to quite low temperatures. (If however, small lattices are used with insufficient data precision, it looks as though the curves for different lattice sizes coincide. Such behavior would indicate, erroneously, the presence of an XY-like region.) In Fig. 3.20,

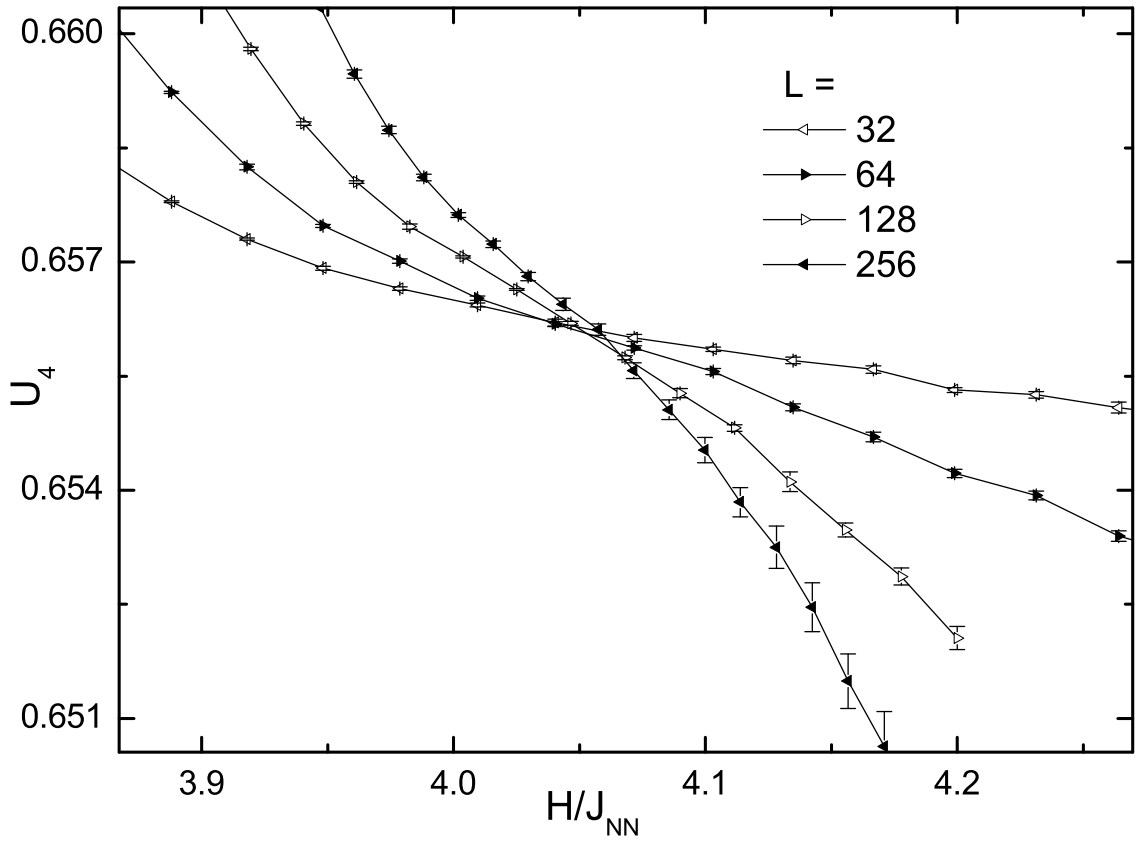

Figure 3.17: Fourth order cumulant U_4 versus field H along the path of constant $k_BT/J_{NN} = 0.7$ for lattice size $L = 32, 64, 128, 256$.

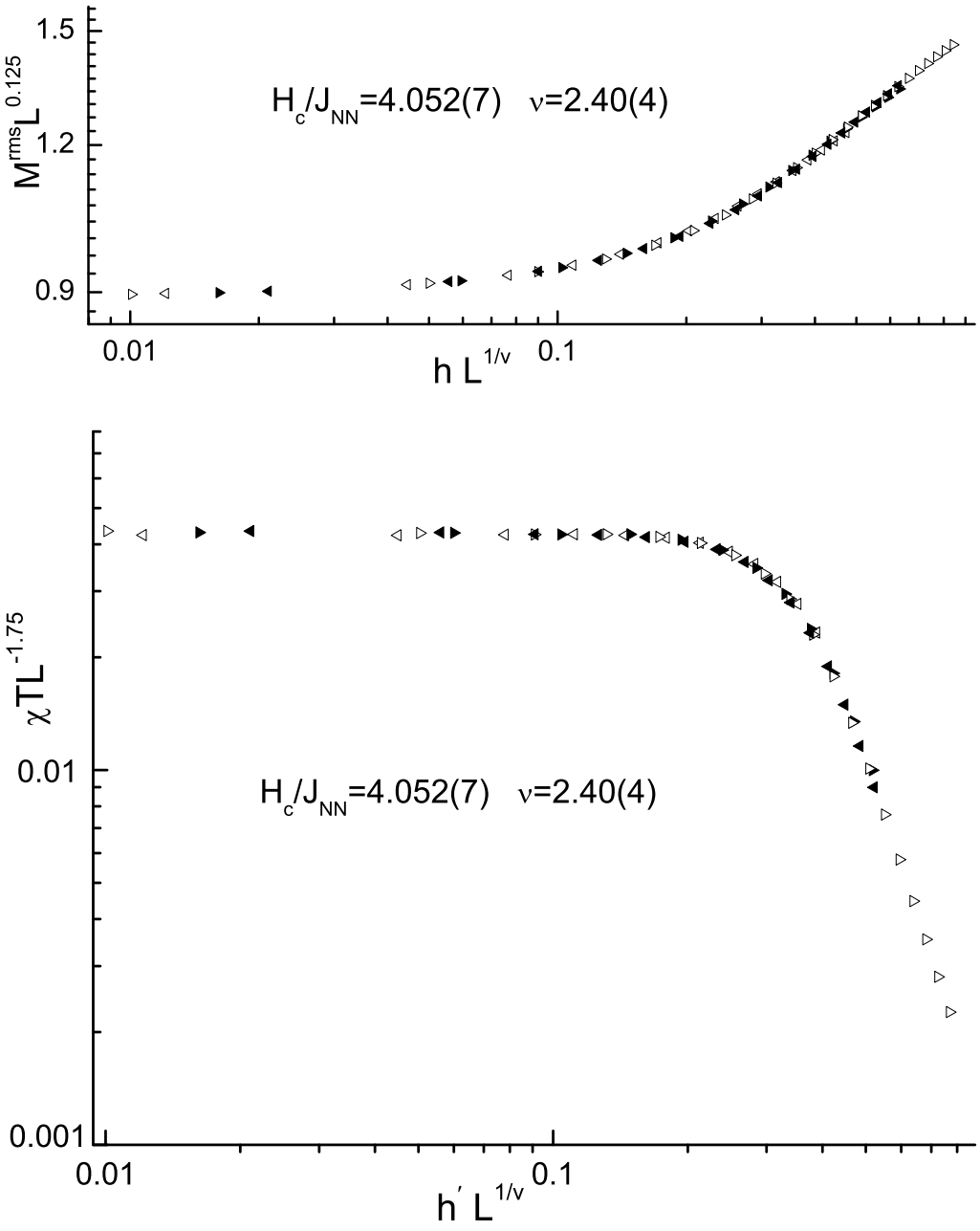

Figure 3.18: Finite size scaling data collapsing along the path of constant $k_BT/J_{NN} = 0.7$ for root-mean-square order parameter and its susceptibility, respectively. $h' = |1 - \frac{H_c}{H}|$ and $h = (1 - \frac{H}{Hc})$. Data are for: $L = 32$, ◁; $L = 64$, ▶; $L = 128$, ▷; $L = 256$, ◀.

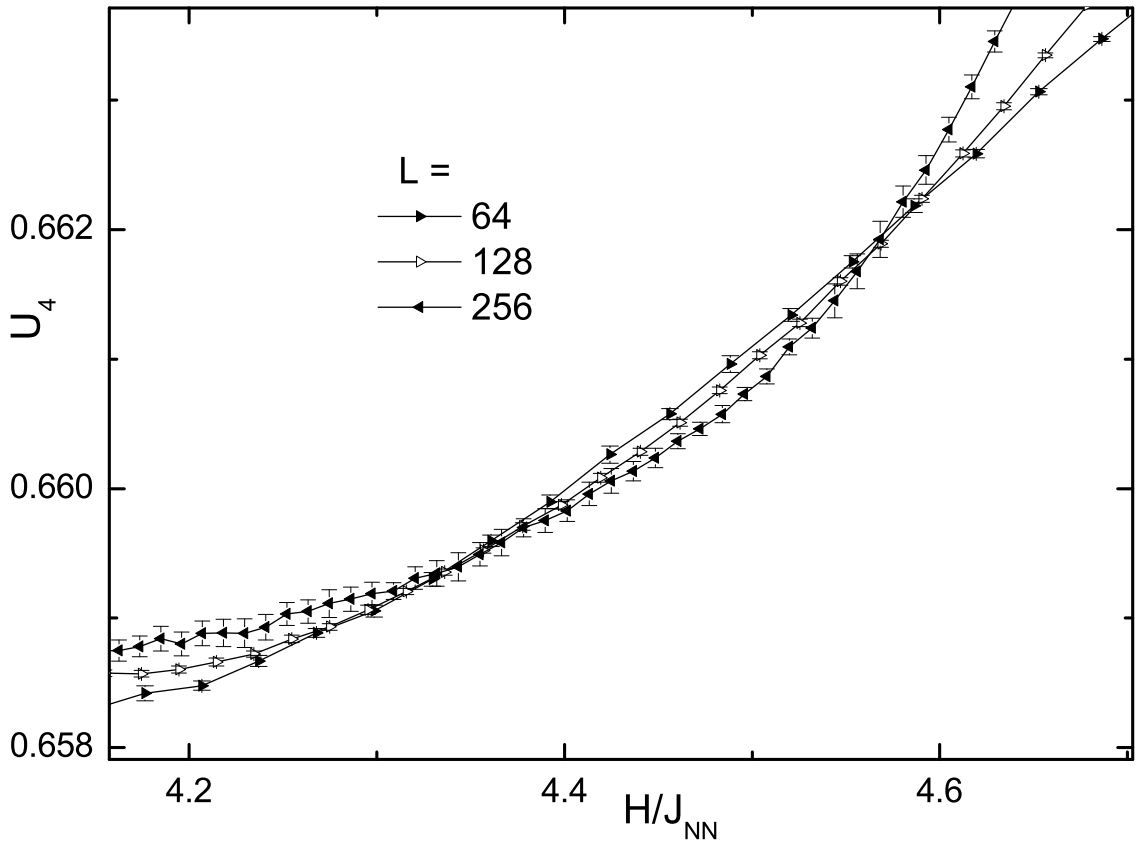

Figure 3.19: Fourth order cumulant U_4 versus field H along the path of constant $k_BT/J_{NN} = 0.45$ for lattice size $L = 64, 128, 256$.

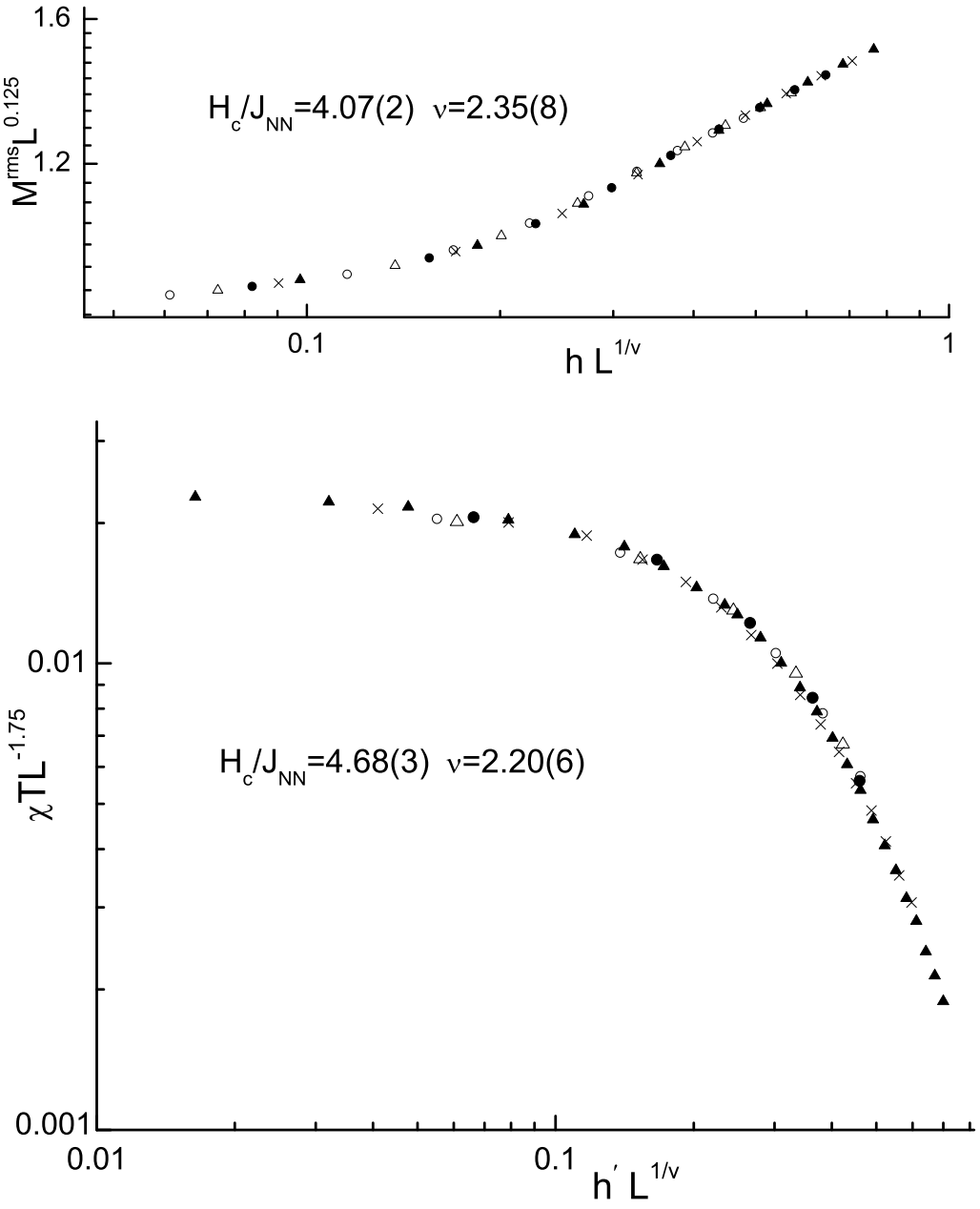

Figure 3.20: Finite size scaling data collapsing along paths of constant $k_BT/J_{NN} = 0.5$ for root-mean-square order parameter and its susceptibility, respectively. Data are for: $L = 80$, ○; $L = 100$, △; $L = 120$, ●; $L = 160$, ×; $L = 200$ ▲.

we show data collapsing fits along the path of constant $k_BT/J_{NN} = 0.5$ (and excellent data collapsing is also found along the path of constant $k_BT/J_{NN} = 0.3$), which confirms that there is a disordered region in between the two ordered states.

We thus conclude that there is no XY-like region and that the two phase boundaries probably only come together at a bicritical point at $T = 0$, although we cannot exclude the possibility of a bicritical point at very low, but non-zero, temperature. However, the data do not yield any hint of such a bicritical point; but the lack of data points for $T < 0.2$ in Fig. 3.10 precludes us from making a definitive statement about this issue. (Moreover, the reentrant behavior of the (2×1) phase makes it very difficult to study the approach to $T = 0$.) The variation of the critical exponents is consistent with the changing magnetic field producing different effective anisotropies which, in turn, is expected to yield non-universal behavior [35]. Due to the large values of ν near the bicritical point (and correspondingly strongly negative values of α), we consider it also extremely unlikely that tricritical points could be found along these transition lines as T becomes small.

3.8 Summary

We have carried out large-scale Monte Carlo simulations for the square-lattice Ising model with repulsive (antiferromagnetic) nearest- and next-nearest-neighbor interactions. From the finite size scaling analysis, both transitions from (2×1) and row-shifted (2×2) ordered states to disordered states turn out to be continuous and non-universal. The reentrance behavior of the (2×1) transition line is confirmed, and the proximity to the transition line to the (2×2) state makes it difficult to untangle the low temperature behavior unless quite large lattices are used. No evidence for XY-like behavior is found, and we conclude that there is probably a zero temperature bicritical point. Although the exponent ν varies along the transition line, the exponent ratio β/ν and γ/ν seem to agree with that of the 2D Ising universality class.

Chapter 4

Lattice-gas Ising model with two-body and three-body interactions

4.1 Introduction

Experimental studies of phase transitions in adsorbed monolayers have resulted in examination of order-disorder transitions in lattice gas Ising models which represent the occupation of the periodic minima in the substrate potential [36, 37]. Such models, usually containing two or more competing two-body interactions, have been studied by Monte Carlo simulations [38, 39] which have determined the location and nature of the resultant phase boundaries. Typical ordered phases which are found for square lattice models with near-neighbor coupling are shown in Fig. 4.1 along with low density and high density disordered states termed lattice gas (L.G.) and lattice liquid (L.L.), respectively. Actually, in Chapter 3, we studied the interesting behavior of this row-shifted (2×2) phase, where we concluded that the transition to the disordered phase is of second order although it appears to be XY-like for small lattice sizes. Experimentally observed asymmetries in phase boundaries as a function of coverage can be explained by lattice gas models if three-body interactions are introduced as well. Theoretical studies of adatom-adatom interactions find surface mediated many-body couplings [40]. A square lattice model with first- and second-neighbor two-body coupling and weak three-body interactions was investigated [41] in an attempt to clarify the phase transitions for H on Pd(001), and predictions were made for the case of even larger three-body coupling. Monte Carlo simulations showed that the inclusion of three-body interactions did make the transition asymmetric for about 50 percent coverage and could even force the tricritical point on one side of the phase boundary to zero temperature. Such an asymmetry

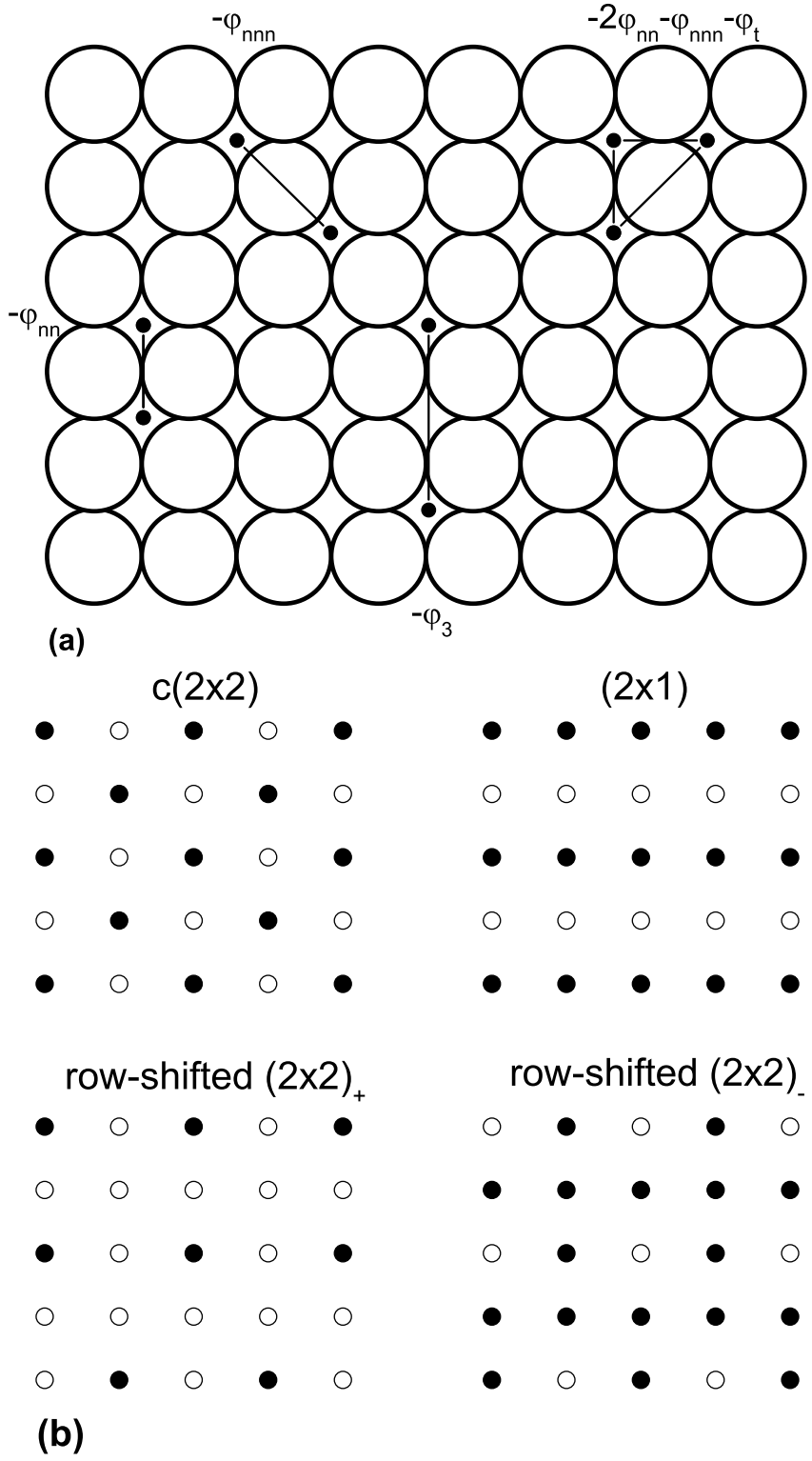

Figure 4.1: (a) Schematic view of the (100) surface of a substrate whose periodic potential provides a square lattice of preferred adsorption sites. The interactions used in this study are shown schematically as straight lines between adatoms which are represented by the filled circles. (b) Unit cells of the ordered overlayer structures discussed in the text.

can also be interpreted in another form of non-additive interactions [42], and a recent Monte Carlo study shows the effect of varying the interaction strength [43].

In this chapter, we will study interaction parameters in a range that has not been previously considered, and hence complete phase diagrams for all four possible scenarios (see next section and Fig. 4.2). As presented in Sec. 4.3 and 4.4, our results for two different interaction strengths do yield qualitatively different phase diagrams.

4.2 Lattice-gas Ising model

A lattice gas model is a collection of atoms whose positions may take on only discrete positions which form a periodic array, in this case a simple square lattice. A configuration of this lattice is defined by site occupation variables c_i where $c_i = 1$ if site i is occupied and $c_i = 0$ if the site is empty. The Hamiltonian which we use includes interaction φ_{nn} between nearest-neighbors, φ_{nnn} between next-nearest-neighbors, and φ_t between neighbors on a triangle inscribed within a square made up of nearest-neighbors:

$$\begin{aligned} \mathcal{H} - \mu N_a = & - (\epsilon + \mu) \sum_i c_i - \varphi_{nn} \sum_{i \neq j} c_i c_j \\ & - \varphi_{nnn} \sum_{i \neq k} c_i c_k - \varphi_t \sum_{i \neq j \neq k} c_i c_j c_k, \end{aligned} \tag{4.1}$$

and the coverage of the lattice of size L is given by

$$\theta = \frac{1}{L^2} \sum_i c_i. \tag{4.2}$$

This model may be transcribed to the Ising model by the transformation to spin variables $\sigma = 1 - 2c_i$, thus giving rise to the Hamiltonian

$$\begin{aligned} \mathcal{H} = & - H \sum_i \sigma_i - J_{nn} \sum_{i \neq j} \sigma_i \sigma_j \\ & - J_{nnn} \sum_{i \neq k} \sigma_i \sigma_k - J_t \sum_{i \neq j \neq k} \sigma_i \sigma_j \sigma_k, \end{aligned} \tag{4.3}$$

where the Ising model interaction parameters are related to the lattice gas couplings by

$$J_{nn} = \frac{1}{4}\varphi_{nn} + \frac{1}{2}\varphi_t \tag{4.4}$$

$$J_{nnn} = \frac{1}{4}\varphi_{nnn} + \frac{1}{4}\varphi_t \tag{4.5}$$

$$J_t = -\frac{1}{8}\varphi_t \tag{4.6}$$

$$H = -\frac{1}{2}(\epsilon + \mu) - \varphi_{nn} - \varphi_{nnn} - \frac{3}{2}\varphi_t \tag{4.7}$$

The magnetization is then related trivially to the coverage

$$M = 1 - 2\theta. \tag{4.8}$$

Because of technical considerations, it is generally easier to carry out simulations in the magnetic (Ising) representation, and additional symmetries often become obvious in this approach. For example, in the Ising representation it is easy to see that the phase diagram must be symmetric in the absence of three-body interactions. Throughout the remainder of this chapter we shall normalize all quantities by the nearest-neighbor coupling J_{nn} and define $R = J_{nnn}/J_{nn}$ and $R_t = J_t/J_{nn}$. As a function of the relative strength of the three-body interactions, Binder and Landau suggested four different schematic phase diagrams (shown in Fig. 4.2) to describe the range of possible behavior due to the inclusion of three-body coupling, and using Monte Carlo simulations verified the nature of the phase diagrams with weak three-spin coupling shown in the upper portion of Fig. 4.2. The remaining two diagrams were not based on any explicit calculation but were guessed as extensions of what was then believed to be the field-dependent behavior in the absence of three-spin interactions. Our views of the correct behavior in this latter case have changed, and given the

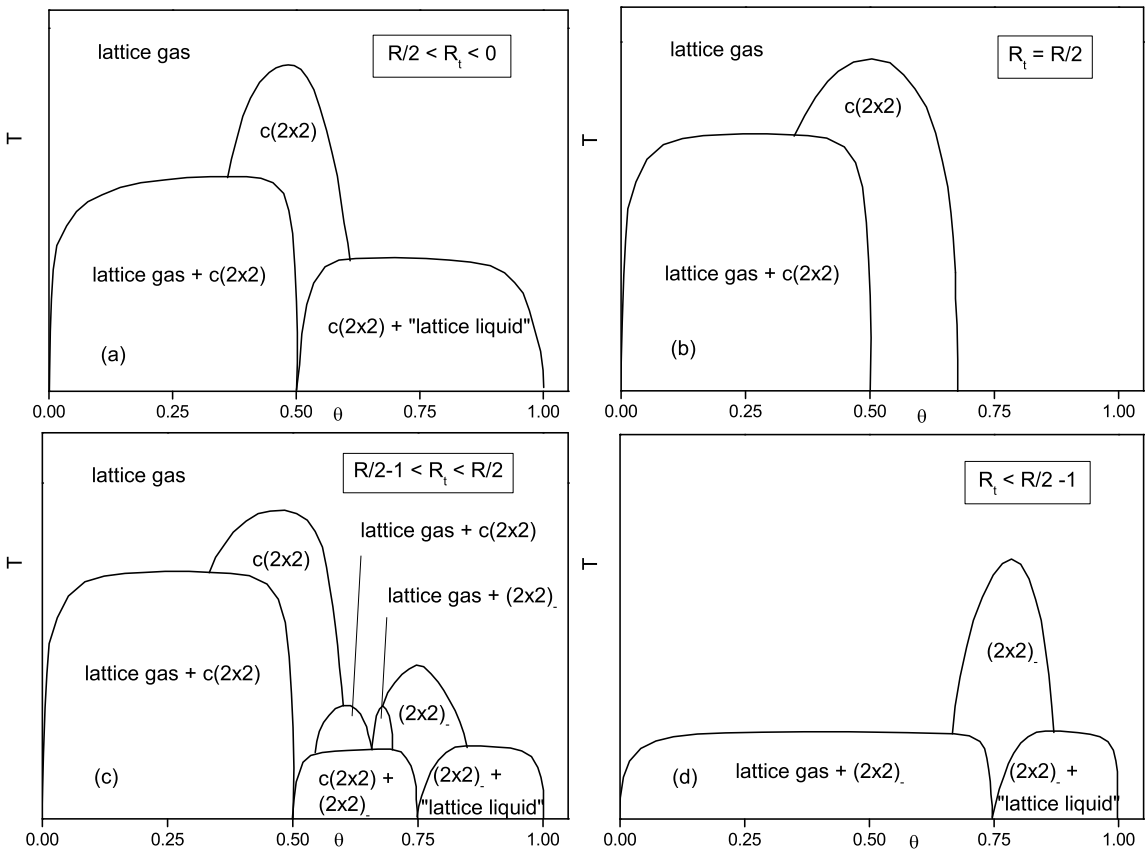

Figure 4.2: Possible schematic temperature-coverage phase diagrams for various choices of R_t (from Ref. [41]). The lattice gas and lattice liquid corresponds to the low density and high density disordered states, respectively.

complexity of phase diagrams in other two-dimensional systems with competing interactions, the predictions should be regarded with care.

At $T = 0$ the energies of each ordered state as well as the lattice gas and lattice liquid phases may be calculated without difficulty, and the intersections of the energy vs. chemical potential lines locate the transitions between neighboring phases. Of course, this is also valid in the Ising representation, and in Fig. 4.3 we show the ground state phase diagrams as a function of R_t and H which we obtain for two different values of R which are in the parameter region discussed by Binder and Landau [41] but not actually simulated.

One interesting feature of this diagram is that the (2×2) state is actually degenerate in that either alternate rows or alternate columns may be shifted randomly by one lattice constant without any cost in energy[38, 6]. This "row-shifted (2×2)" state has been seen before and the nature of the finite temperature transition to the disordered state is a matter of some disagreement. In Chapter 3, we performed large-scale Monte Carlo simulations, and hopefully put this debate to an end concluding that the transition is of second order. We believe that as long as the ground states do not change, the specific choice of parameters is not important and we have simply chosen values for which the $c(2 \times 2)$ and row-shifted (2×2) phases are stable over relatively large ranges of fields in Fig. 4.3 at which multiple phases become degenerate.

We have used the parallel tempering algorithm with GPU accelerated techniques[6] to study the thermodynamic properties of this model from which phase diagrams can be deduced. Spins were placed on $L \times L$ square lattices with periodic boundary conditions and were flipped using a Metropolis transition probability. Typically, 10^6 to 10^7 Monte Carlo steps are used to collect data for each run and 3 to 6 independent runs are taken to calculate standard statistical error bars, and in all the plots of data and analysis in the following sections, if error bars are not shown they are always smaller than the size of the symbols. Lattice sizes from $L = 32$ to $L = 300$ were simulated and the data were interpreted within the context of finite-size scaling [44]. Most of the simulations were carried out on a GeForce

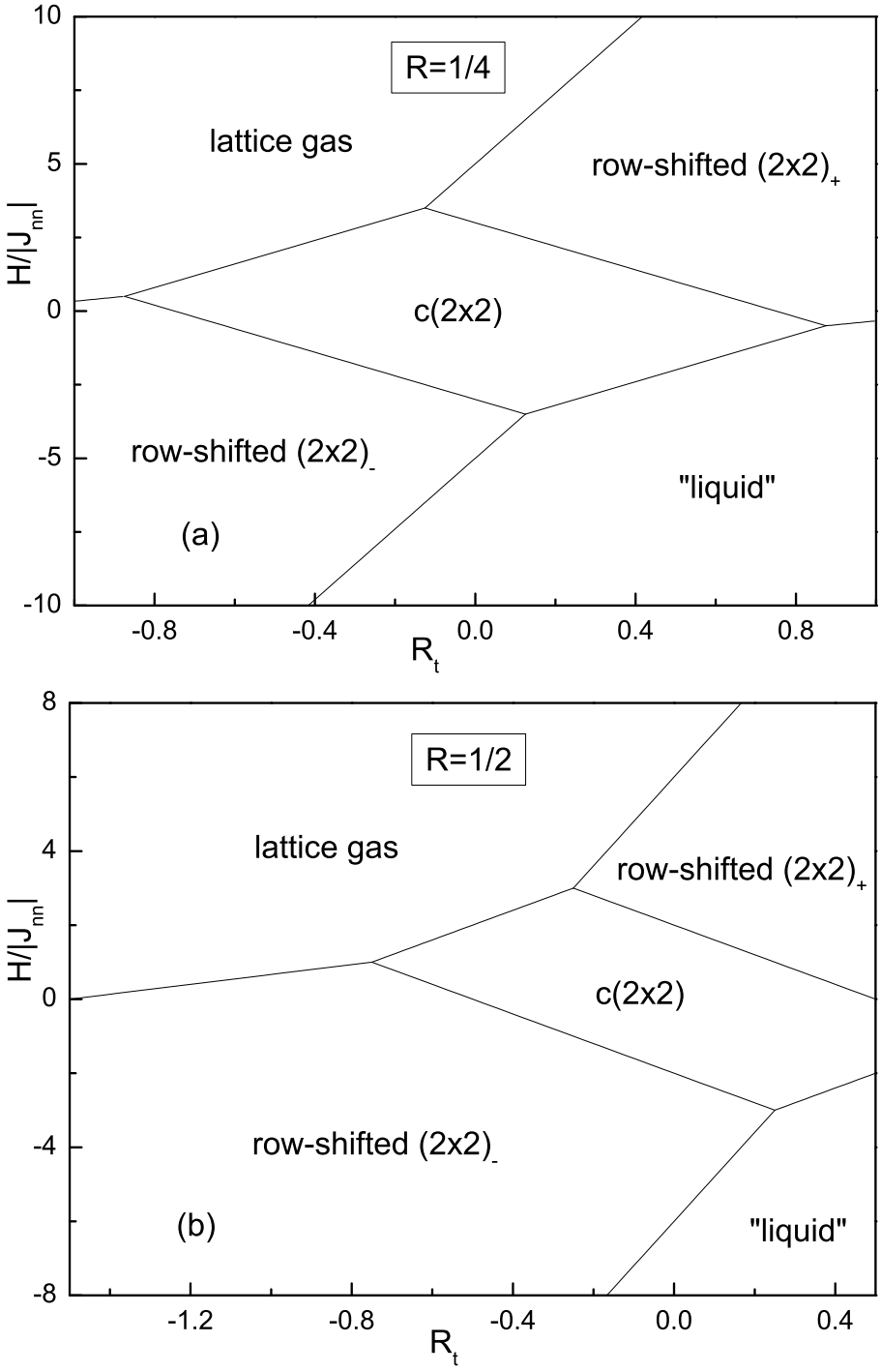

Figure 4.3: Groundstate phase diagrams for the Ising square lattice with pairwise and three-spin interactions: (a) $R = 1/4$; (b) $R = 1/2$.

GTX285 graphics unit. In addition to internal energy, specific heat, and magnetization, we calculated order parameters, e.g., $M_{c(2\times2)}$ ($[M_1 + M_3 - (M_2 + M_4)]/4$, where $M_i, i = 1\ 4$ are sublattice magnetizations defined in the previous Chapter), M^{rms} (see previous Chapter), etc., for the various ordered states shown in Fig. 4.1, and 4th order cumulant U_4 is defined in terms of the order parameter accordingly, i.e., $U_4 = 1 - \frac{<(M_o)^4>}{3<(M_o)^2>^2}$, where M_o is the order parameter for the corresponding state.

4.3 Results and Analysis: $R = 1/4, R_t = -1/4$

Bulk properties such as the specific heat peak, temperature dependence of the 4th order cumulant of the order parameter, etc., were used to determine the location of phase transitions.

Sample data for the 4th order cumulant and specific heat are shown in Fig. 4.4; the specific heat peaks diverge with increasing lattice sizes for fields above $H/|J_{nn}| = -2.0$, but for fields more negative and close to the $c(2 \times 2)$ phase boundary, the specific heat peaks first decrease and then diverge again with larger lattice sizes. Similar behavior was observed for the 4th order cumulant: for the small lattice, there is more than one diverging correlation length; but if the lattice sizes are big enough, only one dominates. Therefore, in certain range of small lattice sizes, the behavior is easy to confuse with XY-like [45], and GPU accelerated simulations of large lattice sizes is essential. For positive fields and low temperatures there is hysteresis in the M vs. H data indicating the presence of first-order transitions as shown in Fig. 4.5, but at higher temperature the data obtained for increasing and decreasing fields are essentially identical.

The resultant phase diagram in field-temperature space is shown in Fig. 4.6. The $c(2 \times 2)$ phase is separated from the disordered phase on the high field side by a phase boundary which contains a tricritical point but on the low field side the transition appears to stay second-order down to the lowest temperature studied, and the row-shifted (2×2) state is also bounded by a line of second order transitions. Since the transition from the $c(2 \times 2)$ ordered

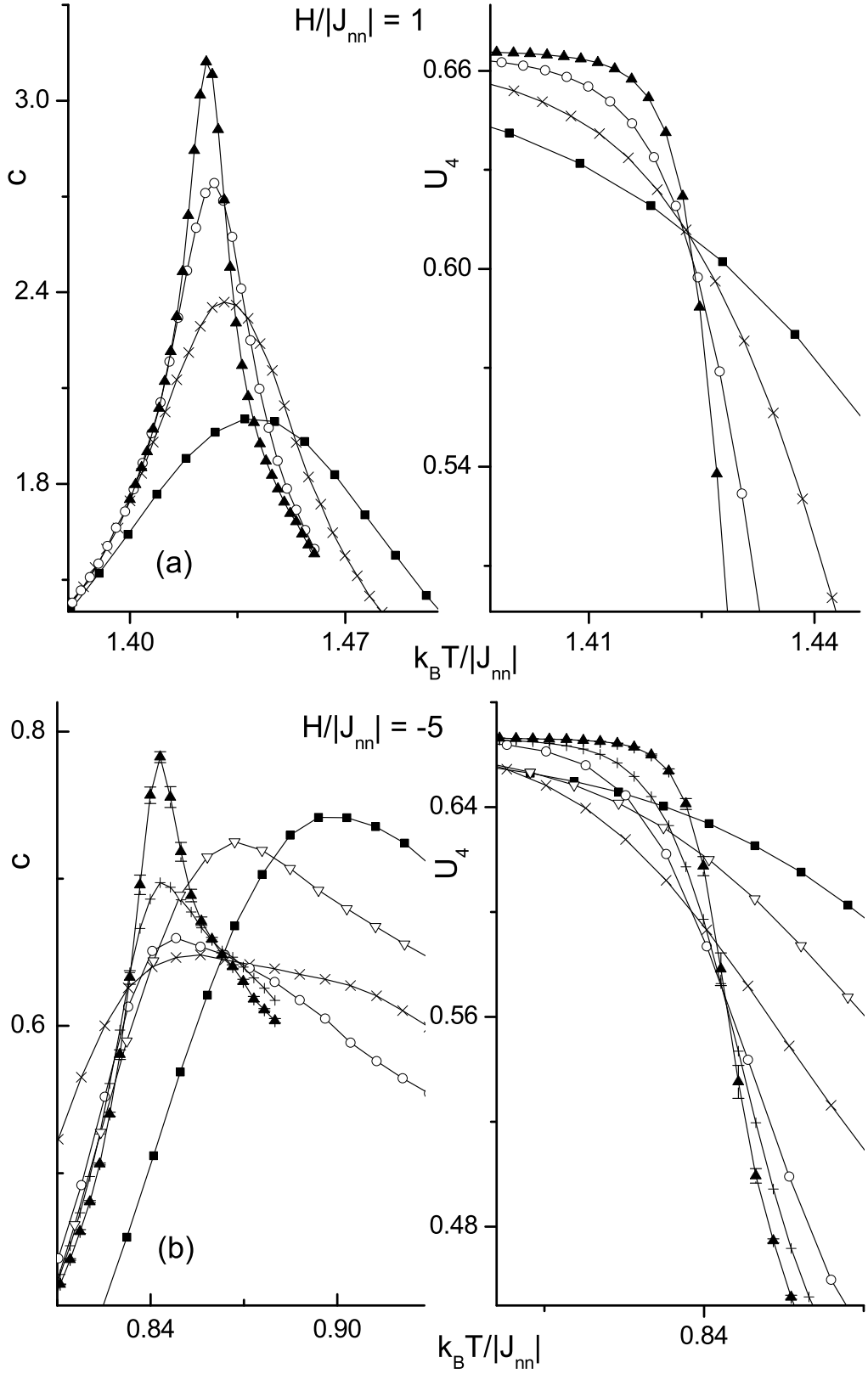

Figure 4.4: Bulk properties for $R = 1/4, R_t = -1/4$: Specific heat c and 4th order cumulant U_4 of the corresponding order parameter for (a) $H/|J_{nn}| = 1$. Data are for: $L = 32$, ■; $L = 64$, ×; $L = 128$, ○; $L = 256$, ▲. (b) $H/|J_{nn}| = -5$. Data are for $L = 30$, ■; $L = 40$, ▽;

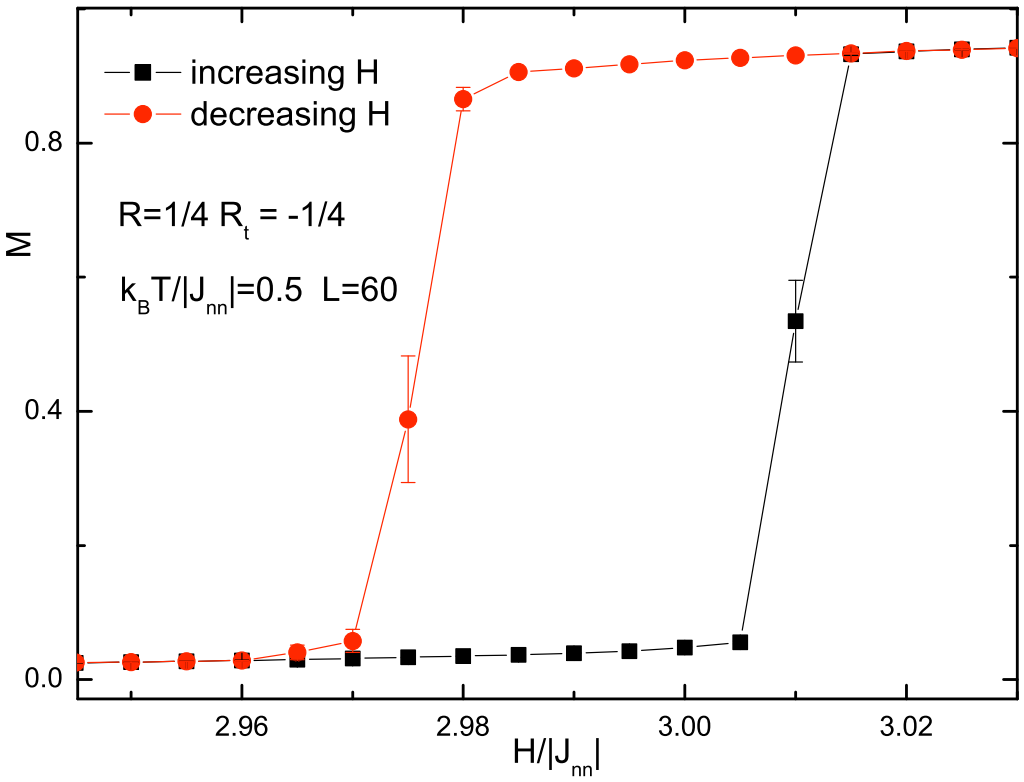

Figure 4.5: Hysteresis for $R = 1/4, R_t = -1/4$ at temperature $k_BT/|J_{nn}| = 0.5$ with lattice size $L = 60$. The large error bars of two central points are due to the fluctuation of the system between two phases.

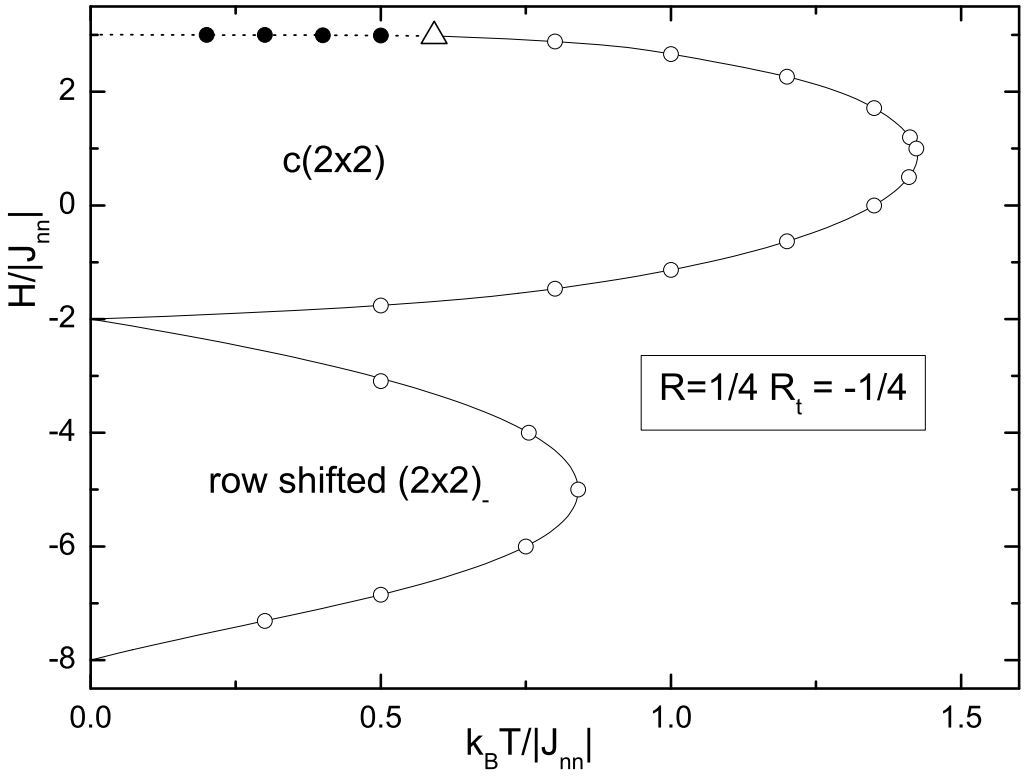

Figure 4.6: Phase diagram in magnetic field-temperature space for $R = 1/4, R_t = -1/4$. The solid curves are second-order phase boundaries and the dashed line indicates first-order transitions. The open triangle indicates the location of a tricritical point.

phase to the disordered phase should belong to the Ising universality, we expect $\alpha/\nu = 0$ for a second-order phase transition since the specific heat shows a logarithmic singularity at the critical point, and $\alpha/\nu = 2$ for a first-order phase transition since the logarithmic singularity is replaced by a peak of height proportional to L^d according to the renormalization-group theory. At the tricritical point, the exact (conjectured) value for the exponent α/ν is $\frac{8}{5}$, which is supported by many renormalization group calculations [46]. Therefore, we estimate the exponent from finite size behavior of specific heat peaks, $c_{max} \sim L^{\alpha/\nu}$, near the connecting section of the first- and second-order transition line, and found the tricritical point is close to $k_BT/|J_{nn}| = 0.592$. To confirm our estimation and get more accurate location, we also calculate the density distribution of the order parameter, as shown in Fig. 4.7. The three-peak structure is observed: two side peaks correspond to the two degenerated $c(2\times 2)$ phases, and the central peak is due to the lattice liquid phase. The final estimation of the tricritical point is $k_BT/|J_{nn}| = 0.5915(4), H/|J_{nn}| = 2.98073(8)$, and the evaluated exponent $\alpha/\nu = 1.59(2)$ agrees nicely with the predicted value of 1.60.

The corresponding phase diagram in coverage-temperature space is shown in Fig. 4.8. Here we see that the $c(2 \times 2)$ phase and the L.G.+$c(2 \times 2)$ coexistence phase, which is present below the tricritical point, appear over substantial ranges of θ and T, whereas the row-shifted (2×2) phase is actually confined to a very narrow range of coverage. This phase diagram is substantially different from that postulated in Fig. 4.2c, and in particular there is no triple point. However, if third-nearest- neighbor two-body interactions are added, the ground state degeneracy for the row-shifted (2×2) state will be removed. Then, tricritical points involving the (2×2) phase could occur, and triple points in the region of the first order transition perhaps as well, i.e., the predicted phase diagram in Fig. 4.2c could then be valid.

In Fig. 4.9 we show adsorption isotherms which are obtained for several different temperatures and for comparison include the Langmuir isotherm which would be correct for a non-interacting lattice gas. The jump in the low temperature data shown by the dotted

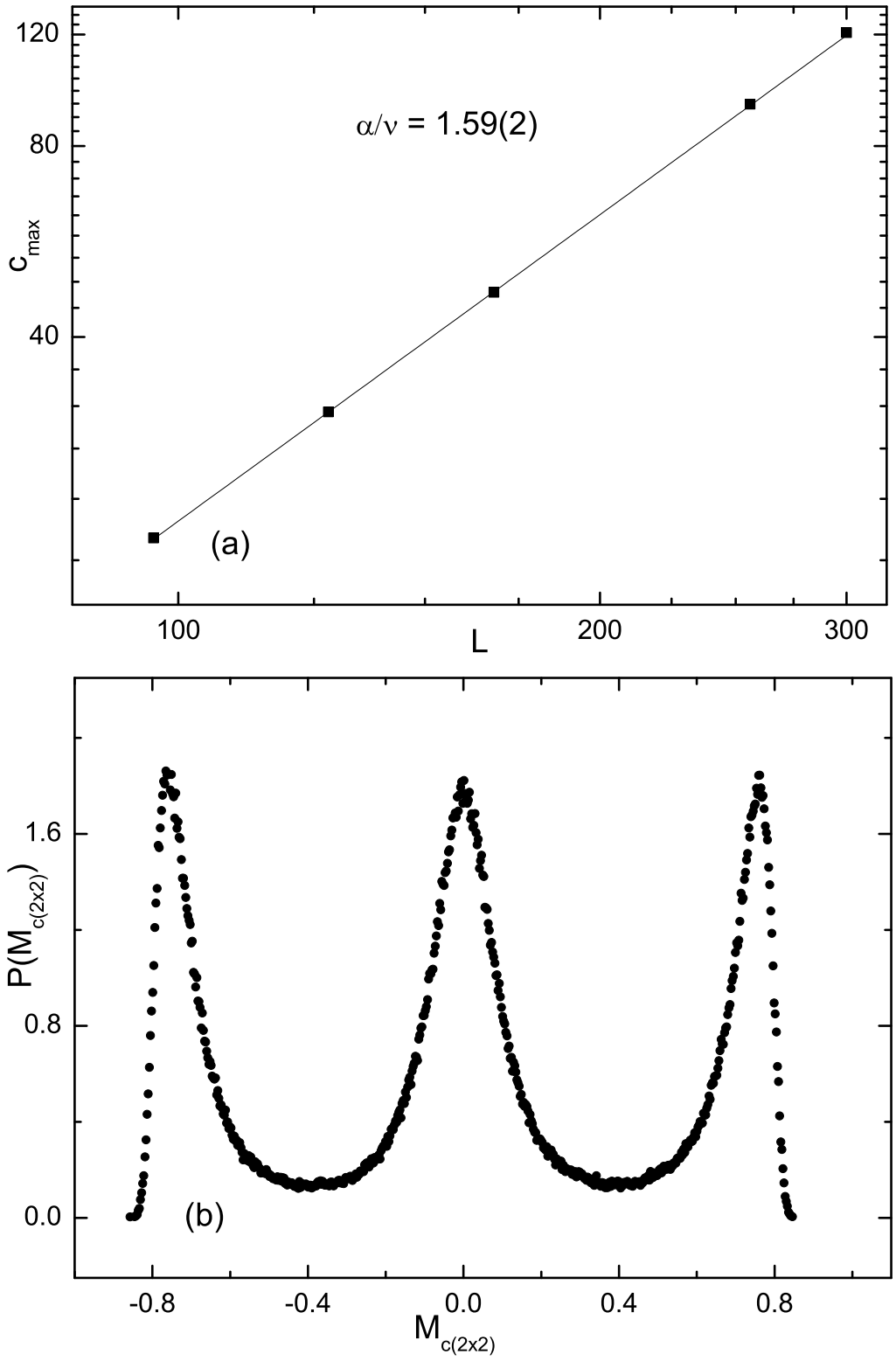

Figure 4.7: Tricritical point(T=0.5915(4), H=2.98073(8)) for $R = 1/4, R_t = -1/4$: (a) Curve fit of the specific heat peaks. (b) The density distribution of order parameter $M_{c(2\times2)}$ for $L = 256$.

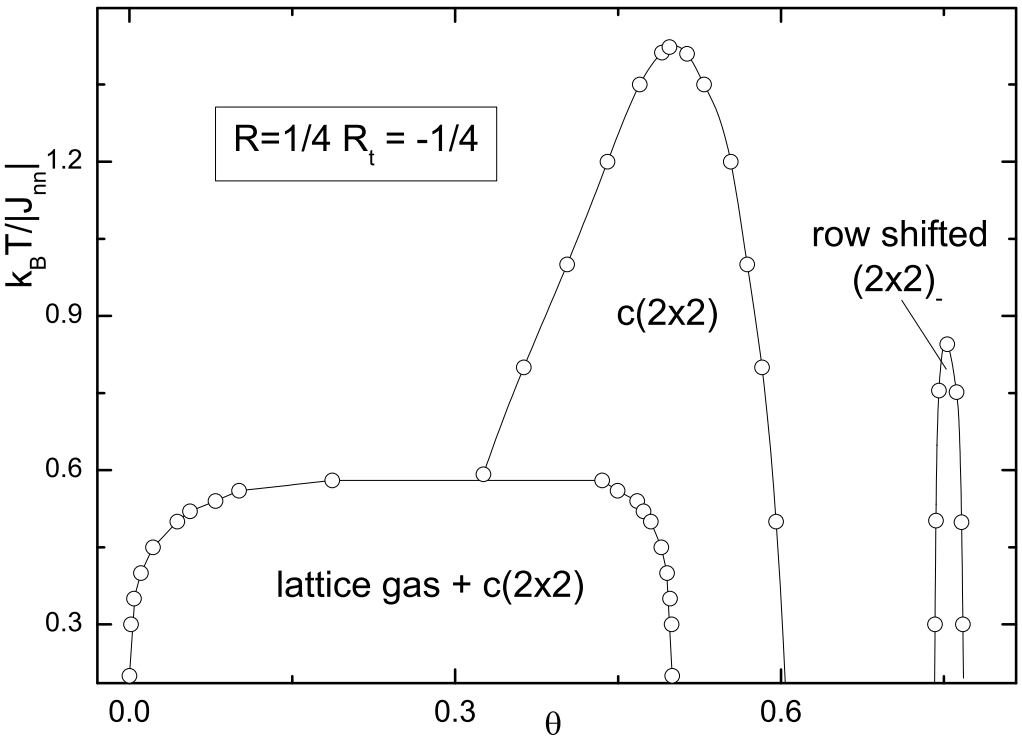

Figure 4.8: Temperature-coverage phase diagram for $R = 1/4, R_t = -1/4$.

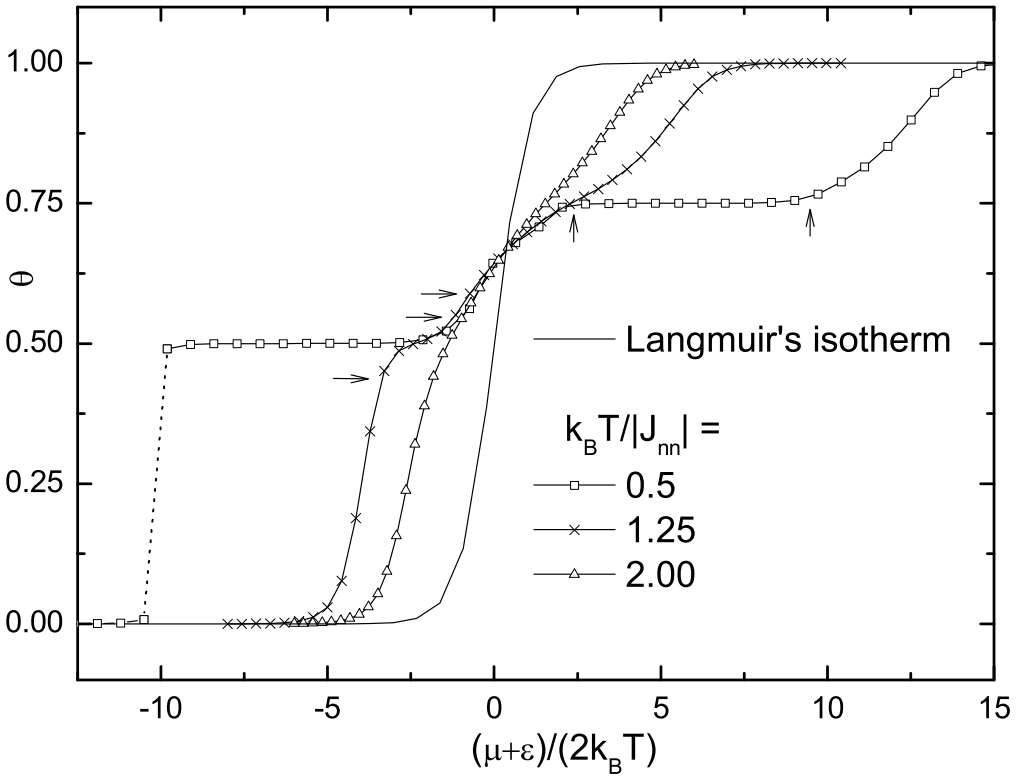

Figure 4.9: Adsorption isotherms of the lattice gas model with $R = 1/4, R_t = -1/4$. The arrows mark the second-order phase transitions.

line clearly locates the first-order transition; but the second-order transitions, indicated by the arrows, are extremely difficult to identify from the adsorption isotherms. The step-like behavior of the lowest temperature adsorption isotherm shown in this figure is not dissimilar to the multiple risers which are seen for multilayer adsorption, but here it merely represents multiple transitions within a single layer!

4.4 Results and Analysis: $R = 1/2, R_t = -1$

The same thermodynamic properties were determined as were described in Sec. 4.3, and since there were no significant differences in the nature of the results, we shall not show any raw data for this case. The resultant phase diagram in $H - T$ space is shown in Fig. 4.10. A line of first-order transitions, terminating in a critical point, separates a lattice liquid from a lattice gas state, and a line of second-order transitions bounds a row-shifted (2×2) phase.

Since there is a lack of symmetry among the two different phases at the critical point that terminates the first order line, we have to apply the field mixing analysis described in Section 2.5. In Fig. 4.11, we plot the density distribution function at estimated criticality $k_B T/|J_{nn}| = 2.2738(4), H/|J_{nn}| = 1.24925(13)$ with the controlling parameter $s = -0.30(2)$ for $L = 64$ and 128. The superimposed curve is the corresponding distribution for the two-dimensional Ising model for $L = 400$. The nonuniversal factors $a_{\mathcal{M}}$ for each lattice sizes is chosen in such a way that the variable $a_{\mathcal{M}}^{-1} L^{\beta/\nu}(\mathcal{M} - < \mathcal{M} >_c)$ has unit variance so the term $a_{\mathcal{M}}^{-1} L^{\beta/\nu}$ is just the inverse of the variance of $\mathcal{M}$.

As for the critical exponents for the continuous transition from the row-shifted (2×2) phase to the paramagnetic phase for both cases in Sec. 4.3 and 4.4, the correlation length exponent ν changes along the transition line, but the reduced exponents γ/ν and β/ν have Ising values, similar to what we found in Ref [6]. The effect of such competing interactions has also been studied by several authors using a cluster variation method (CVM) [47] or entropic Wang-Landau sampling [48]. Although CVM studies predict first-order behavior in

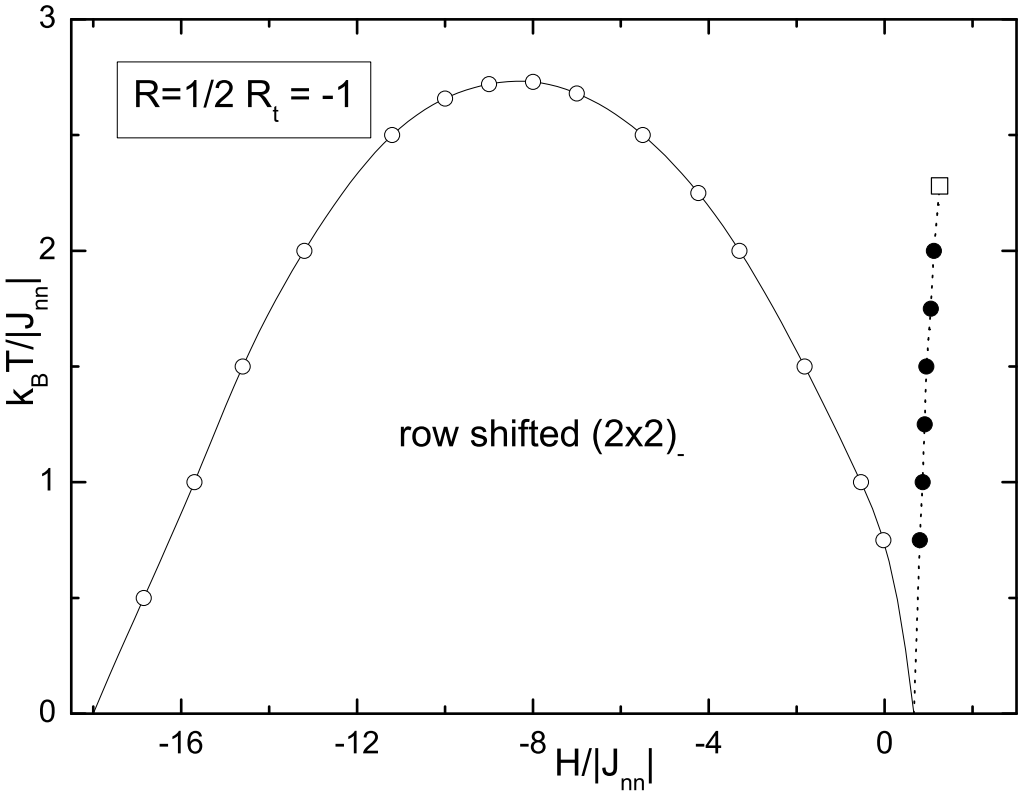

Figure 4.10: Field-temperature phase diagram for $R = 1/2, R_t = -1$. The solid curves are second-order transitions and the dashed lines show first-order phase boundaries. The open square indicates the location of the terminating critical point.

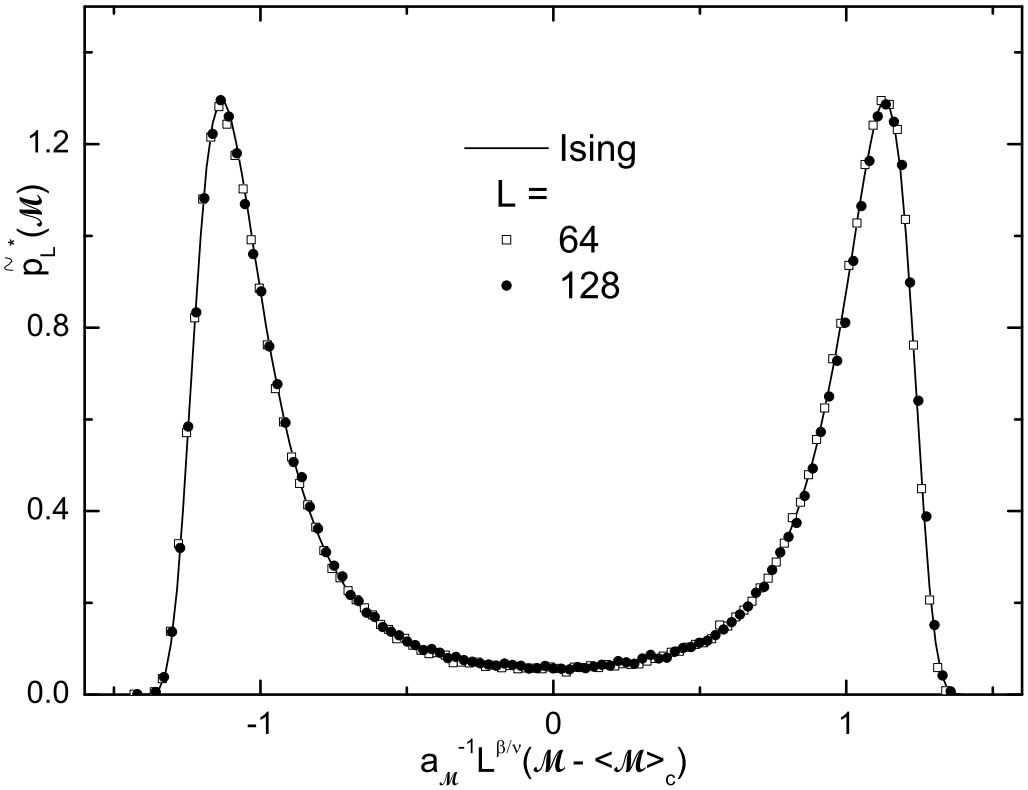

Figure 4.11: The density distributions for $L = 64$ and 128 at the critical point $k_B T/|J_{nn}| = 2.2738(4), H/|J_{nn}| = 1.24925(13)$. The full curve is the corresponding distribution for the two-dimensional Ising model for $L = 400$.

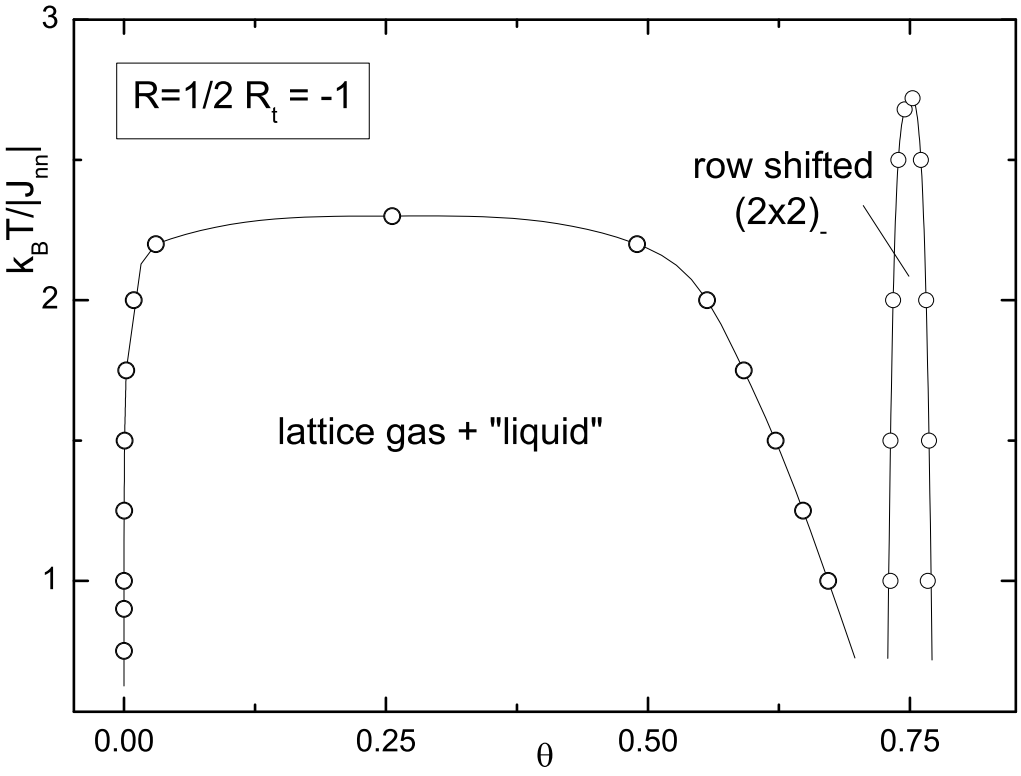

Figure 4.12: Temperature-coverage phase diagram for $R = 1/2, R_t = -1$.

a certain range of R ($0.5 \sim 1.2$), our results in this chapter and Ref. [6] agree with Ref. [48], i.e., a continuous transition with reduced exponents that obey Suzuki's weak universality.

The phase diagram is replotted in coverage-temperature space in Fig. 4.12. Here, too, we see that the row-shifted (2×2) phase is present only over a relatively narrow range of coverages, but the L.G. + L.L. coexistence phase is stable over a much larger region of $T - \theta$ space. Comparing with Fig. 4.2d, we see that there are qualitative differences between the actual behavior and the phase diagrams which had previously been postulated; but again, if the degeneracy allowing for row-shifted structures were removed, the predicted phase diagram in Fig. 4.2d might hold.

4.5 2D Ising Tricritical universality

In Sec. 4.3, we obtained the phase diagram for the interaction ratio $R = 1/4, R_t = -1/4$, and accurately locate the tricritical point for this model. Although we plotted the density distribution of the order parameter and showed the three peak structure, we did not perform scaling analyses and present the universal behavior. For reference in the future, we carefully analyze the data and simulate larger lattices to make a record for the tricritical density distribution of 2D Ising universality. To this end, we also simulate another Ising model with antiferromagnetic nearest-neighbor and ferromagnetic next-nearest-neighbor interactions ($R = -1/2, R_t = 0$), whose tricritical point was located by Landau and Swendsen [18] using the Monte Carlo renormalization-group method.

In Fig. 4.13, we plot the order parameter (antiferromagnetic magnetization) density distribution for both models with different lattice sizes. Due to the finite size effect, the effective tricritical points for different sizes are not exactly the same, but the trend to converge with increasing size is clearly seen. In fact, within the resolution of this simulation, there are other scalable distribution shapes that can be found in the nearby range. To confidently determine the universal distribution, more accurate estimation of the tricritical point is essential, but

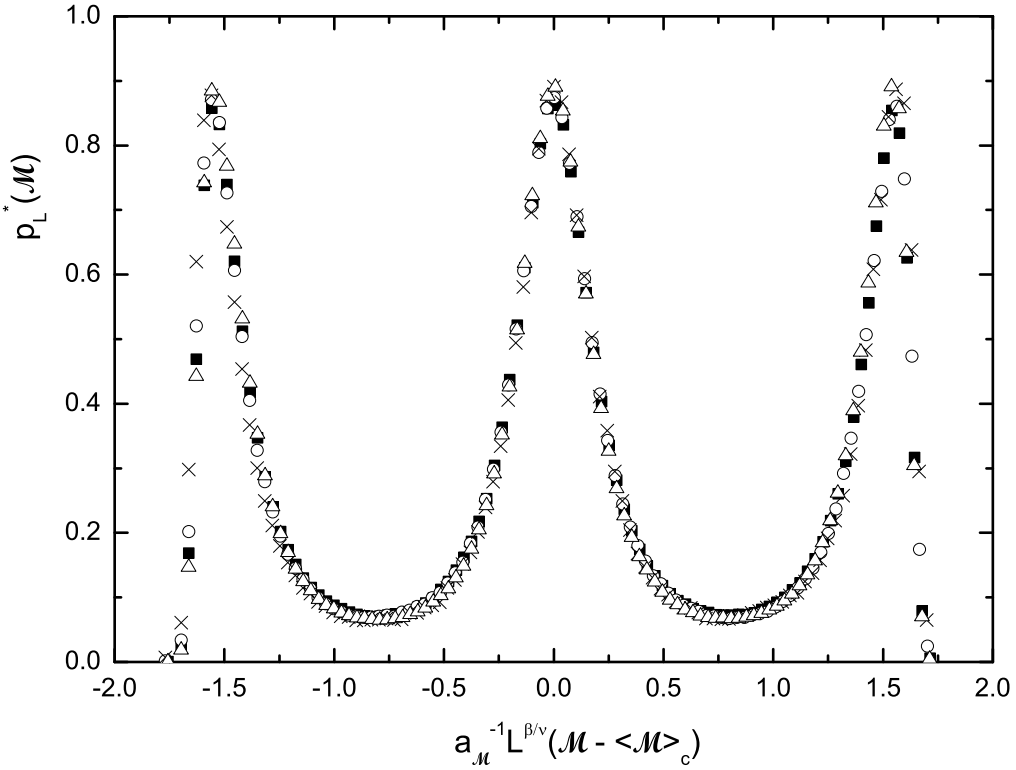

Figure 4.13: Tricritical density distribution of the order parameter for 2D Ising universality. $R = 1/4, R_t = -1/4$: $L = 128$ $k_BT/|J_{nn}| = 0.5915, H/|J_{nn}| = 2.98077$, ■; $L = 256$ $k_BT/|J_{nn}| = 0.5915, H/|J_{nn}| = 2.980734$, ○; $L = 512$ $k_BT/|J_{nn}| = 0.59152, H/|J_{nn}| = 2.980720$, ×. $R = -1/2, R_t = 0$: $L = 128$ $k_BT/|J_{nn}| = 0.8300, H/|J_{nn}| = 3.965638$, △.

our current simulations provide so far the most accurate evaluation of the universal distribution.

4.6 Summary

Monte Carlo simulations have been used to extract phase diagrams for simple models on a square lattice with three-body interactions which are larger in magnitude than those which have been previously studied. We find qualitatively different behavior than that which had been suggested in Ref. [41]. If third-nearest neighbor two-body interactions are added, the ground state degeneracy for the (2×2) state will be removed; but of course there is no guarantee that the finite temperature behavior will not show remnants of this effect. These results further demonstrate the complexity which may be found in relatively simple models with competing interactions. This problem is of interest to statistical mechanics in its own right and we believe that further Monte Carlo studies of such models will continue to display many of the features observed in experimental studies of adsorbed monolayers.

Chapter 5

Structural and thermodynamic properties of water clusters

5.1 Introduction

Simulations of cluster systems can be challenging, since the free energy landscapes for such systems are often rugged, i.e., there exists large numbers of local minima, sampling across which requires considerable computer time depending on the sizes of energy barriers. In the case of water clusters, there are numerous semi-empirical and ab initio quantum mechanical studies [49]. For simulations of empirical models, Monte Carlo with Minimization (MCM, also known as basin hopping) [50, 51, 52, 53] or genetic algorithms [54, 55] have been used if the ground state properties are of interest, and simulated annealing or parallel tempering [56] have been applied if finite temperature behavior is the focus.

Recently, Wang-Landau sampling [3] has attracted interest for the simulation of continuous systems [4] with complex behavior. Since the sampling is essentially a random walk in energy space and the temperature does not explicitly enter the process, one can obtain the density of states of the system which can, in turn, be used to determine the properties of the system over the entire temperature range through one simulation.

Water clusters have been an interesting and challenging subject for both theoretical and experimental studies [57]. Although most of the water models were devised primarily for the simulation of liquid water, studies of clusters can still be very informative. We perform Monte Carlo simulations using Wang-Landau sampling to investigate three well-studied nonpolarizable potential models: Simple Point Charge/Extended (SPC/E), Transferable Intermolecular Potential 3 Point (TIP3P) and 4 Point (TIP4P), and another model with polarizable point dipole: Gaussian Charge Polarizable (GCP). The SPC/E and TIP3P are three interaction

site models, hence computationally more economical than the last; while the TIP4P potential is known to be better than TIP3P in reproducing properties of liquid water [58]; For the GCP model, although it's computationally expensive, it provided the best overall results compared with quantum mechanical calculations [59]. From our simulations, we can obtain not only the ground states properties but also the melting "phase transition" behaviors for $(H_2O)_N$ (N up to 50) clusters.

In the next section, we will review the water models we used, and some details of our Wang-Landau simulations. The results about the structural and thermodynamic properties will be presented in Sec. 5.3 and 5.4, respectively. We shall provide some suggestion on the water model selection in Sec. 5.5. From Sec. 5.6 to Sec. 5.10, we will show the implementation and the performance of a multi-GPU massively parallel Wang-Landau sampling approach.

5.2 Water Models and Wang-Landau Method

For the non-polarizable models, SPC/E, TIP3P and TIP4P, the potential energy ϵ_{mn} for interaction sites m and n of water molecule i and j, respectively, is given by [58]

$$\epsilon_{mn} = \sum_{i}^{m}\sum_{j}^{n} \frac{q_i q_j e^2}{r_{ij}} + \frac{A}{r_{OO}^{12}} - \frac{C}{r_{OO}^{6}}$$

where r_{ij} is the distance between two interaction sites, and q, A and C are model specified parameters which are listed in Table. 5.1. For the polarizable model GCP [60], the partial charges of a molecule are represented by spherical Gaussian charge distributions; and hence, in addition to the charge-charge electrostatic energy and the dispersion energy, there is an extra contribution in term of the polarization energy. In this case, the configurational energy ϵ_{mn}^{p} is given by

$$\begin{aligned}
\epsilon_{mn}^{p} &= \sum_{i}^{m}\sum_{j}^{n} \frac{q_i q_j e^2}{r_{ij}} erf\left(\frac{r_{ij}}{\sqrt{2(\sigma_m^2+\sigma_n^2)}}\right) \\
&+ \frac{\epsilon}{1-6/\gamma}\left(\frac{6}{\gamma}\exp\left(\gamma\left[1-\frac{r_{OO}}{\sigma}\right]\right) - \left(\frac{\sigma}{r_{OO}}\right)^6\right) \\
&- \frac{1}{2}\left(\mathbf{p}_i\cdot\mathbf{E}_i^q + \mathbf{p}_j\cdot\mathbf{E}_j^q\right)
\end{aligned}$$

where the Gaussian width parameter σ_m, m could be the interaction site M or H, and the Buckingham exp-6 potential parameter σ, ϵ, γ are also listed in Table. 5.1. The induced dipole $\mathbf{p}_i$, the charge electric field $\mathbf{E}_i^q$ and the induced electric field $\mathbf{E}_i^p$ are given by

$$\begin{aligned}
\mathbf{p}_i &= \alpha_{pol}\left(\mathbf{E}_i^q + \mathbf{E}_i^p\right), \\
\mathbf{E}_i^q &= \sum_{j\neq i}\sum_{j}^{n} \frac{q_j(\mathbf{r}_i - \mathbf{r}_j)}{r_{ij}^3}\left[erf\left(\frac{r_{ij}}{\sqrt{2(\sigma_M^2+\sigma_n^2)}}\right)\right. \\
&\quad \left. - \frac{\sqrt{2}r_{ij}}{\sqrt{\pi(\sigma_M^2+\sigma_n^2)}}\exp\left(\frac{-r_{ij}^2}{2(\sigma_M^2+\sigma_n^2)}\right)\right], \\
\mathbf{E}_i^p &= \sum_{j\neq i}\mathbf{T}_{ij}\cdot\mathbf{p}_j.
\end{aligned}$$

$\mathbf{T}_{ij}$ is a 3×3 matrix written as

$$\mathbf{T}_{ij} = \frac{1}{r_{ij}^3}\left(f\frac{3\mathbf{r}_{ij}\mathbf{r}_{ij}}{r_{ij}^2} - g\mathbf{I}\right)$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{r}_{ij}\mathbf{r}_{ij}$ are vector direct product of $\mathbf{r}_{ij}$. The scalars f and g are given by

$$\begin{aligned}
f &= erf\left(\frac{r_{ij}}{2\sigma_M}\right) - \left[\frac{r_{ij}}{\sqrt{\pi}\sigma_M} + \frac{r_{ij}^3}{6\sqrt{\pi}\sigma_M^3}\right]\exp\left(\frac{-r_{ij}^2}{4\sigma_M^2}\right) \\
g &= erf\left(\frac{r_{ij}}{2\sigma_M}\right) - \frac{r_{ij}}{\sqrt{\pi}\sigma_M}\exp\left(\frac{-r_{ij}^2}{4\sigma_M^2}\right)
\end{aligned}$$

In practice, the induced dipole and the induced electric field are solved for iteratively.

In order to obtain correct thermodynamics, the evaporation event of a water molecule needs to be prevented. In the literature [56], a constraining sphere with the origin at the center of the mass of the system is usually used for this purpose. While this approach works for small systems, with a carefully chosen radius, the situation become complicated for relatively large clusters($N > 10$), since it is difficult to choose a radius such that the constraining sphere is neither so small that it distorts the structures nor so big that evaporation is still problematic.

In this study, we turn to the use of "pressure" to prevent evaporation. Similar to the uniform sampling of volume V in the NPT ensemble[61], the biased distribution for the uniform sampling of energy E would be

$$p_{bias} = \frac{\exp\left(-\beta PV\right)}{\Omega(E)}\frac{Q(NVT)}{Q(NPT)}$$

Table 5.1: Parameters for potential models: SPC/E, TIP3P, TIP4P and GCP.

Property	SPC/E	TIP3P	TIP4P	GCP
r_{OH}(Å)	1.0	0.9572	0.9572	0.9572
$\angle HOH$(deg)	109.47	104.52	104.52	104.52
$A \times 10^{-3}$(kcal Å^{12}/mol)	629.4	582.0	600.0	
C(kcal Å^6/mol)	625.5	595.0	610.0	
exp-6 potential: σ(Å), $\epsilon/k_B(K), \gamma$				3.69, 110, 12.75
q_O	-0.8476	-0.834		
q_H	0.4238	0.417	0.52	0.6113
q_M			-1.04	-1.2226
Gaussian widths: σ_H, σ_M				0.610, 0.455
r_{OM}(Å)			0.15	0.27
polarizability α_{pol}(Å^3)				1.444

where Q is the partition function for the corresponding ensemble, and Ω is the density of states. We defined V^* as V/σ^{*3} and P^* as $\beta P\sigma^{*3}$ ($\sigma^* = 3.16$Å, which is about $(A/C)^{\frac{1}{6}}$ for non-polarizable models), and let T approach infinity (every state of the system has the same probability). Then, the ratio of the two partition functions is proportional to V^N. Therefore, we have the transition probability acc(b→a) from state "b" to state "a"

$$acc(b \to a) = min\left\{1, \frac{\Omega(E_b)}{\Omega(E_a)}\frac{V_a^{*N}}{V_b^{*N}}\exp\left[-P^*(V_a^* - V_b^*)\right]\right\}$$

Another intuitive way to reach the transition probability follows Ref. [62]. Note that, in the NPT ensemble, the probability of the system assuming a certain configuration Γ is proportional to

$$\exp\left[-\beta\left(E(\Gamma) + PV\right) + N\mathbf{ln}(V)\right]$$

and by replacing βE with the entropy S and defining V^* and P^* as above, we have a "Hamiltonian" for the uniform sampling

$$\mathcal{H} = -S - P^*V^* + N\mathbf{ln}(V^*)$$

, The transition probability follows $min[1, \exp(-\delta\mathcal{H})]$, which is the same as acc(b→a).

In addition to the translational and rotational moves of a single molecule, a volume change move is also introduced to maintain constant P^*. For each type of move, the trial step length is generated from the uniformly distributed random number in the range of (-1,1) multiplying the maximum step length which is chosen properly such that the average acceptance rates are about $20 \sim 40\%$. From the analysis of data obtained, the dimension of the containing box is the same within error bars for different temperatures, and as expected, it depends only on the value of P^*. In fact, as shown in Fig. 5.1, given the length of the box, simulations using the constraining box can reproduce the results from simulations using "pressure" as a constraint, which indicates somewhat of an equivalence of these two types of conditions. But the latter has the advantage of simplicity, since the same P^* constraint is applied to all system sizes. For small clusters, there are tested values of constraining radius reported [56], and by comparing average box sizes for different values of P^*, we find 0.1[1] $|\sigma^{*3}|$ is a proper value for this system. Note that the constraining volume is consist of "hard wall", and it may influence the molecular structure at high temperatures.

During Wang-Landau sampling computations, the modification factor is reduced to $\mathbf{ln}(f) \rightarrow 10^{-7}$ for most cases, although we find 10^{-6} seems to be sufficient for our system, and for each iteration we adopt a flatness criterion that requires the minimum energy histogram entry to be no less than $50 \sim 60\%$ of the histogram mean. Since the ground state energy is generally unknown, we extend entries for both the density of states and energy histogram when a lower energy state is found. Then, we reset the histogram to zero and restart this iteration again. Each Monte Carlo step consists of translational and rotational moves for every molecule and a volume change move. After each iteration, a global update is performed to the density of states to make the system stay in the lower energy state more often.

[1] 0.1 is in units of atm / k_B / K, where k_B is the Boltzmann constant and 1 atm / k_B / K = $\frac{1}{136.26}\mathring{A}^{-3}$

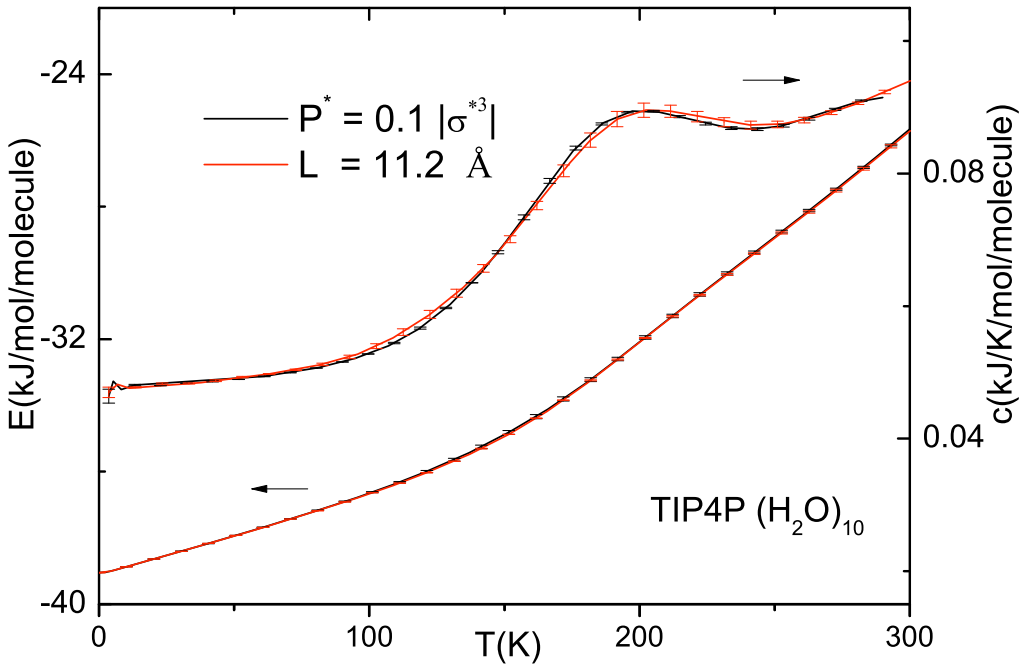

Figure 5.1: Comparison of the constant "pressure" constraint (black line) and the constraining box constraint (red line) for $(H_2O)_{10}$ using the TIP4P potential model.

The simulation is divided into two stages: in the first stage, we aim to estimate the density of states, and no measurement is performed. Then, in a production run, other than the energy histogram, histograms for the radial distribution function, pair energy distribution, and hydrogen bond distribution are also collected. With these distributions, we can calculate thermodynamic properties for the above quantities at any temperature. On the other hand, configurations with the 100 lowest energies are saved and used as initial states for the optimization procedures [63]: direction set and conjugate gradient method, to find the local minima. Then, the overall minima would be our estimate of the ground state.

To speed up the simulations, we implement a parallel scheme for the Wang-Landau method. Essentially, Wang-Landau sampling is a random walk in energy space, and each simulation unit is viewed as a random walker. In our simulations, we can have several walkers

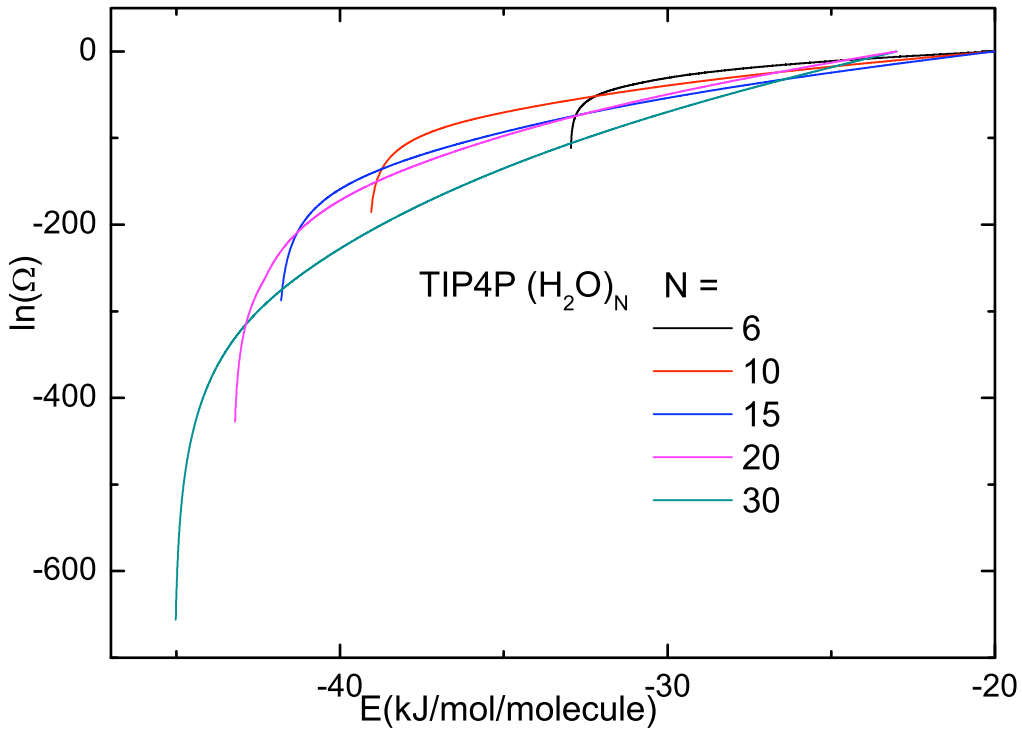

Figure 5.2: Densities of states Ω (in natural logarithm scale) for $(H_2O)_N$ $N = 6, 10, 15, 20, 30$ with TIP4P model. For $N = 6, 10, 15$ the maximum energy in simulations is -20 kJ/mol/molecule, and -23 kJ/mol/molecule for $N = 20, 30$. Data are normalized for the maximum energy, respectively.

with the same initial states but different random number sequences, and they share the same density of states and energy histograms. The synchronization is made after every single trial move and the communication between walkers is by Message Passing Interface (MPI). Of the four models, SPC/E and TIP3P are the least computationally demanding, and given overheads in our code, simulations with TIP4P run almost as fast. For the GCP model, however, the code is about 20 times slower than that of TIP4P, mainly due to the iterative calculation of the polarization energy involving all the molecules. (This is not a performance issue in Molecular Dynamics simulations, since at each time step an update is applied to all the molecules anyway regardless which potential model is used.)

Water cluster systems can be very complicated, and regular Metropolis sampling does not work efficiently or at all. There is a parallel tempering study recently reported [56], but it can not approach the ground state of the model since in the low energy region, large numbers of replicas are needed to ensure enough overlaps of energy distributions so that replica exchange occurs. On the other hand, to study ground state structures, MCM[50, 51, 52, 53] is usually employed, however, such studies can not provide any information about thermodynamics and distributions of structures at finite temperatures since the physical ensemble can not be restored from such samplings. The Wang-Landau method can effectively and efficiently sample a broad energy range of interest. It does not suffer from the cost of overcoming energy barriers as a Metropolis-type algorithm does, since a flat histogram of energy is designed, and it collects the density of states in the process, therefore thermodynamic quantities can be calculated. In Fig. 5.2, densities of states Ω for water cluster $(H_2O)_N$ $N = 6, 10, 15, 20, 30$ are plotted. Note that Ω is in natural logarithm scale, and for $N = 30$, it crosses over 600 orders of magnitude.

Monte Carlo simulations have been performed on $(H_2O)_N$, $N = 6 \sim 30$ with ground states searching. For each case, 5 to 8 independent runs are taken to calculate standard statistical error bars, and in all the plots of data and analysis shown in following sections, if error bars are not shown they are always smaller than the size of the symbols.

5.3 Structural properties

In Fig. 5.3 and Fig. 5.4, we list the ground states of $(H_2O)_N$, $N = 6, 8, 10, 12, 15, 20, 25, 30$, found via Wang-Landau sampling for potential models SPC/E, TIP3P, TIP4P, and GCP. SPC/E ground state structures are generally qualitatively closer to those obtained for TIP4P and GCP than were those obtained for TIP3P. There are extensive results, using MCM or genetic algorithms, reported on TIP4P up to $N = 37$ [50, 55, 51, 52, 53], TIP3P up to $N = 30$ [52], and SPC/E up to $N = 25$ [55]. By comparison, our results of SPC/E and TIP4P are consistent with Ref. [55] and Ref. [50], respectively, and we obtain generally better structures

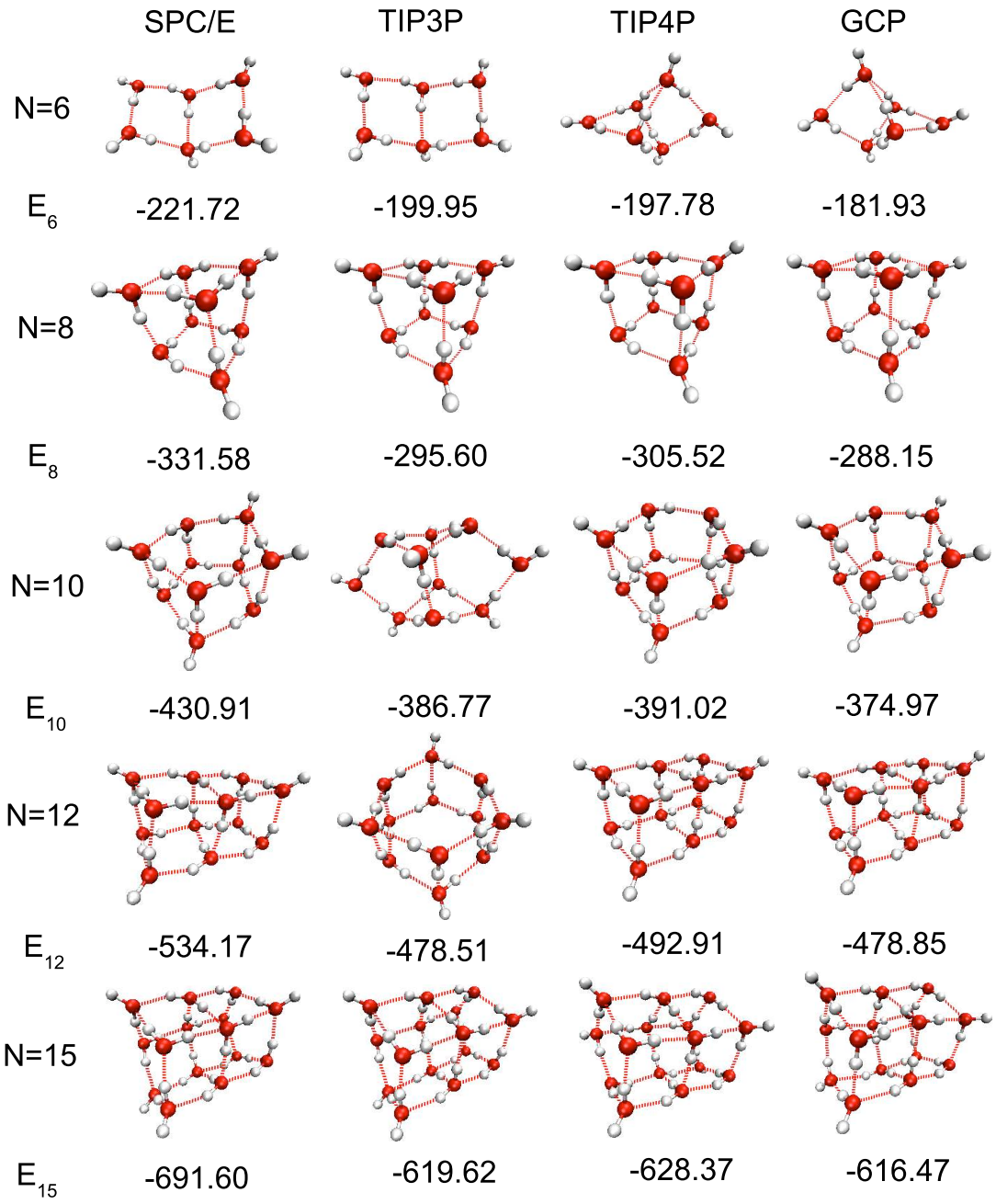

Figure 5.3: The ground states of $(H_2O)_N$, $N = 6, 8, 10, 12, 15$, for potential model SPC/E, TIP3P, TIP4P, and GCP. Energy are given in kJ/mol.

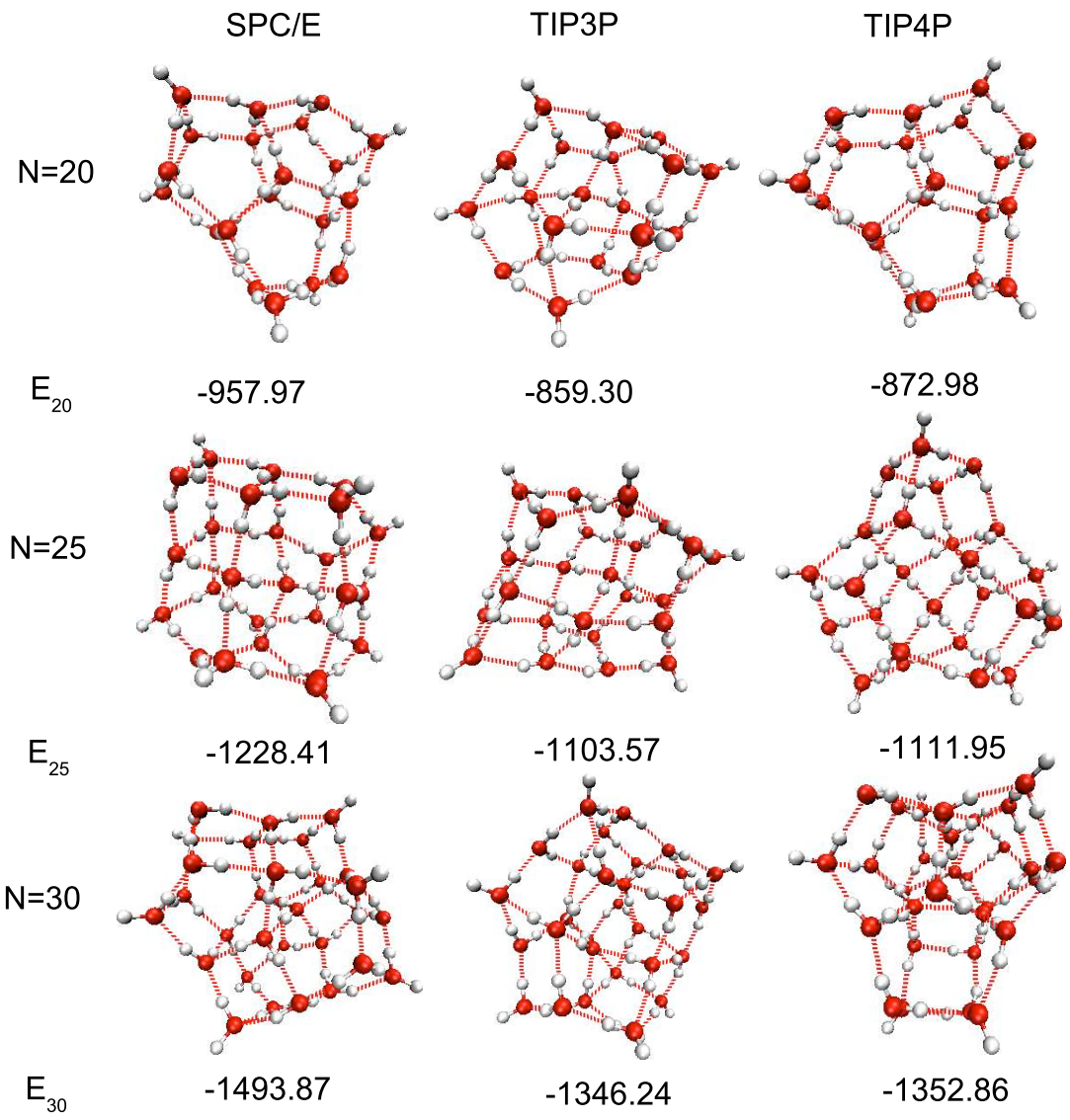

Figure 5.4: The ground states of $(H_2O)_N$, $N = 20, 25, 30$, for potential model SPC/E, TIP3P, and TIP4P. Energy are given in kJ/mol/mol.

Table 5.2: Comparisons of the ground state energy (kJ/mol) of $(H_2O)_N$ for TIP4P and TIP4P/2005.

Model	$N = 6$	$N = 8$	$N = 10$	$N = 12$	$N = 15$	$N = 20$
TIP4P	-197.78	-305.52	-391.02	-492.91	-628.37	-872.98
TIP4P/2005	-218.01	-336.87	-431.06	-543.72	-692.98	-962.88

with TIP3P than that reported in Ref. [55] and consistent with Ref. [52]. For the GCP model, we approach the same ground state of hexamer as in Ref. [59].

Ground state structures are also evaluated for the model TIP4P/2005[64], which is a reparameterization of TIP4P. The differences are the charge values $q_H = 0.5564$ (q_M=$2q_H$), the Lennard-Jones parameters A = 731.3×10^3(kcal Å^{12}/mol), C = 736(kcal Å^{12}/mol), and the distance of the negative charge $r_{OM} = 0.1546$Å. In comparison with TIP4P, it provides similar structures for $N = 6, 8, 10, 12, 15, 20$ but the energies are lower (See Table. 5.2). Our estimation of the ground state energy of hexamer for TIP4P/2005 is consistent with Ref. [59].

Although the radial distribution function $g(r)$ is usually used to describe liquid structures, for the case of clusters it can also provide valuable information. With the histograms of such distributions as functions of energy, we can calculate $g(r)$ at any temperatures. In Fig. 5.5, 5.6 and 5.7, we plot $g(r)$ for O-O, O-H and H-H pairs for $(H_2O)_{12}$ at temperatures $T = 50, 150$, and 300 K. As expected, at room temperature all structural distributions are quite similar for these four potentials, since they were devised to target properties of liquid water. However, the low temperature structures are significantly different. Fortunately, it is not an issue for most applications where simulations are performed near room temperature. The average distances of the first and second neighbor O-O pair at different temperatures are listed in Table. 5.3. It appears the three models have similar d^1_{OO}, and it does not significantly depend on the temperatures or the cluster size; while the d^2_{OO} shows notable changes with

$(H_2O)_N$	Model	$T = 50$ K		$T = 150$ K		$T = 300$ K	
		d^1_{OO}	d^2_{OO}	d^1_{OO}	d^2_{OO}	d^1_{OO}	d^2_{OO}
$N = 8$	SPC/E	2.75(2)	3.52(3)	2.74(3)	3.53(1)	2.75(2)	4.17(3)
	TIP3P	2.76(2)	3.60(2)	2.74(3)	4.17(4)	2.76(2)	4.25(3)
	TIP4P	2.74(2)	3.77(2)	2.77(2)	3.88(2)	2.77(3)	3.96(2)
	GCP	2.83(2)	3.88(2)	2.82(2)	3.88(2)	2.83(3)	4.05(3)
$N = 10$	SPC/E	2.74(2)	3.47(3)	2.74(2)	3.58(5)	2.75(2)	4.17(2)
	TIP3P	2.74(3)	3.27(2)	2.74(2)	4.20(3)	2.76(2)	4.20(6)
	TIP4P	2.74(2)	3.72(2)	2.75(1)	4.07(3)	2.77(1)	4.00(5)
	GCP	2.82(2)	3.73(2)	2.80(3)	4.32(2)	2.82(2)	4.12(5)
$N = 12$	SPC/E	2.76(2)	3.56(2)	2.74(2)	3.62(3)	2.75(3)	4.21(4)
	TIP3P	2.74(3)	3.57(4)	2.74(2)	4.21(3)	2.77(2)	4.22(5)
	TIP4P	2.75(2)	3.90(4)	2.76(2)	3.93(3)	2.77(3)	4.10(6)
	GCP	2.82(2)	3.92(2)	2.81(3)	3.93(2)	2.82(2)	4.12(4)
$N = 15$	SPC/E	2.74(3)	3.51(4)	2.74(2)	3.62(6)	2.75(2)	4.22(5)
	TIP3P	2.76(2)	3.63(2)	2.75(2)	4.17(4)	2.77(2)	4.18(3)
	TIP4P	2.74(3)	3.73(7)	2.75(2)	4.07(3)	2.77(1)	4.11(5)
	GCP	2.80(3)	3.67(3)	2.80(3)	4.32(2)	2.82(2)	4.23(4)
$N = 30$	SPC/E	2.73(4)	4.22(8)	2.72(3)	4.27(6)	2.75(2)	4.30(5)
	TIP3P	2.74(3)	4.17(4)	2.76(3)	4.18(4)	2.77(2)	4.24(4)
	TIP4P	2.74(3)	4.04(5)	2.74(3)	4.18(4)	2.77(2)	4.21(4)

Table 5.3: The average distances of the first and second neighbor O-O pair at $T = 50, 150$ and 300 K for $(H_2O)_N$ $N = 8, 10, 12, 15, 30$.

temperature, and behaves differently for different system sizes since they generally have different transition temperatures.

Another quantity of interest is the distribution of hydrogen bonds, which is the key to water's abnormal properties. Actually, there exist many ways to define the hydrogen bond, either from a distance or an energy point of view. In this research, if the binding energy of two water molecules exceeds a certain threshold, then we consider there is a hydrogen bond connecting these two molecules. The threshold is then determined from the pair energy distribution, as shown in Fig. 5.8(a). Of the four models, GCP has the narrowest low energy band, hence the most well defined hydrogen bond energy. The spike near 0 kJ/mol/molecule

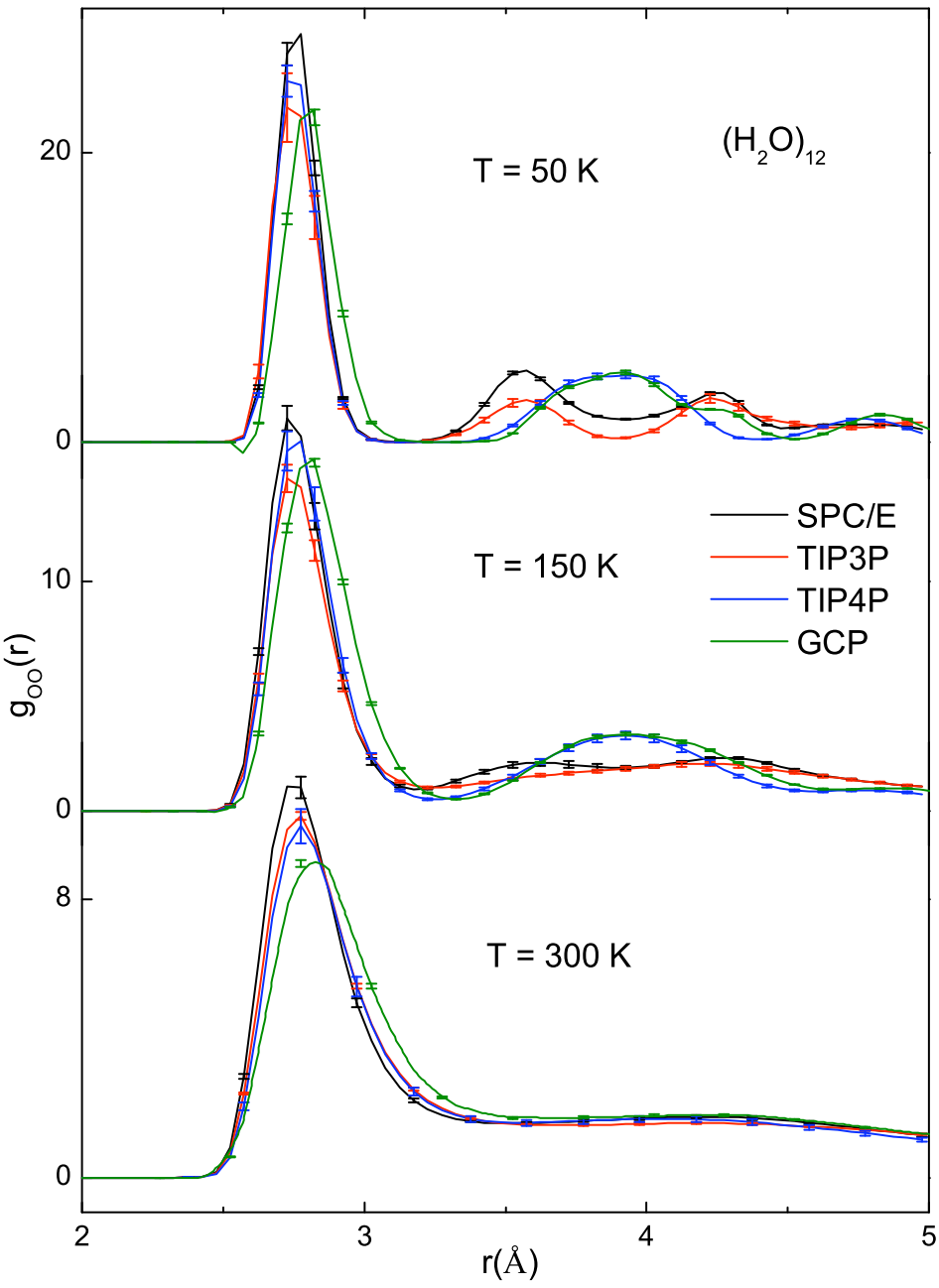

Figure 5.5: Radial distribution functions $g_{OO}(r)$ of $(H_2O)_{12}$ for the O-O pair at $T = 50, 150$ and 300 K. Note that the vertical scale are different for different temperatures.

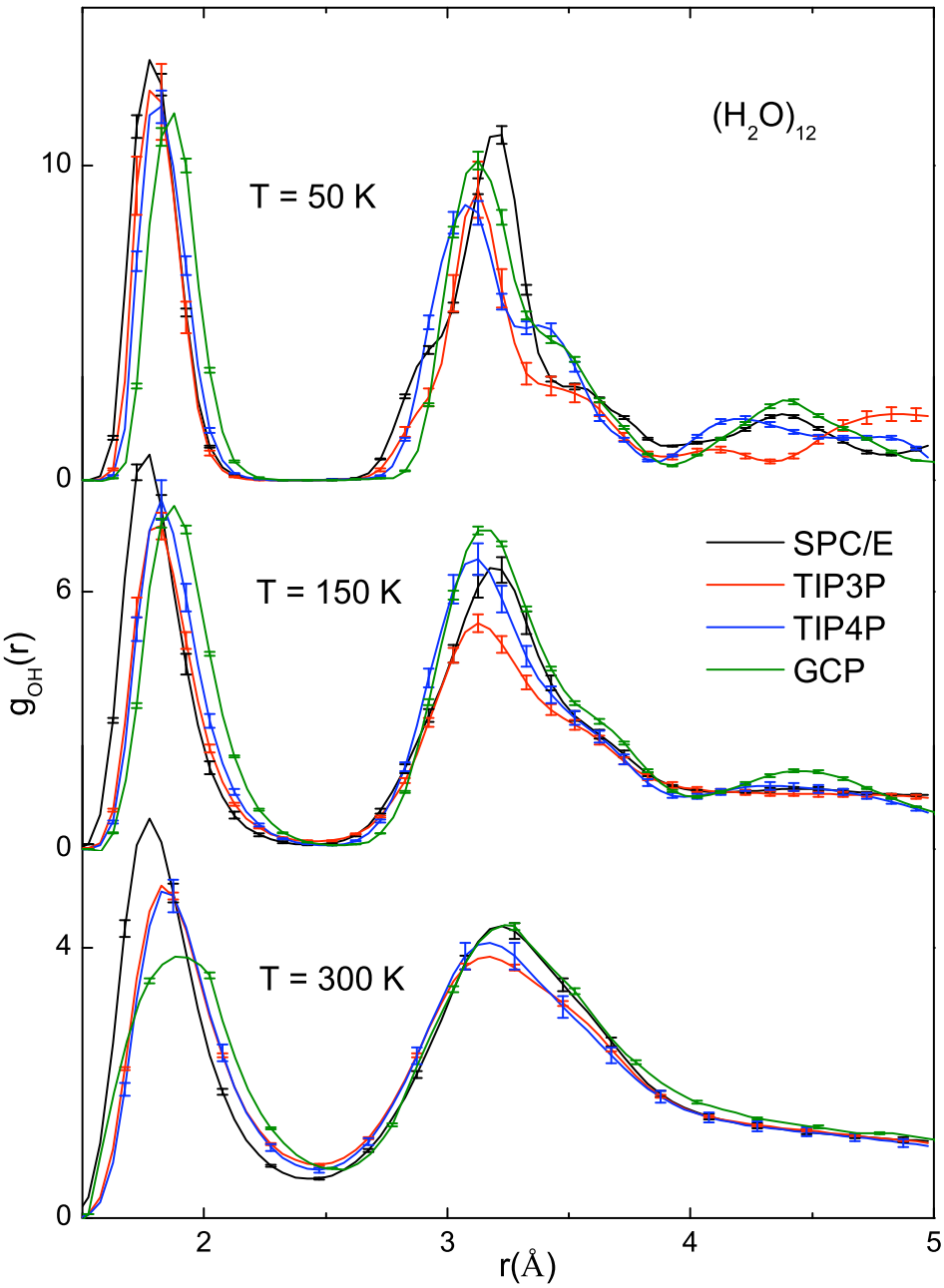

Figure 5.6: Radial distribution functions $g_{OH}(r)$ of $(H_2O)_{12}$ for the O-H pair at $T = 50, 150$ and 300 K. Note that the vertical scale are different for different temperatures.

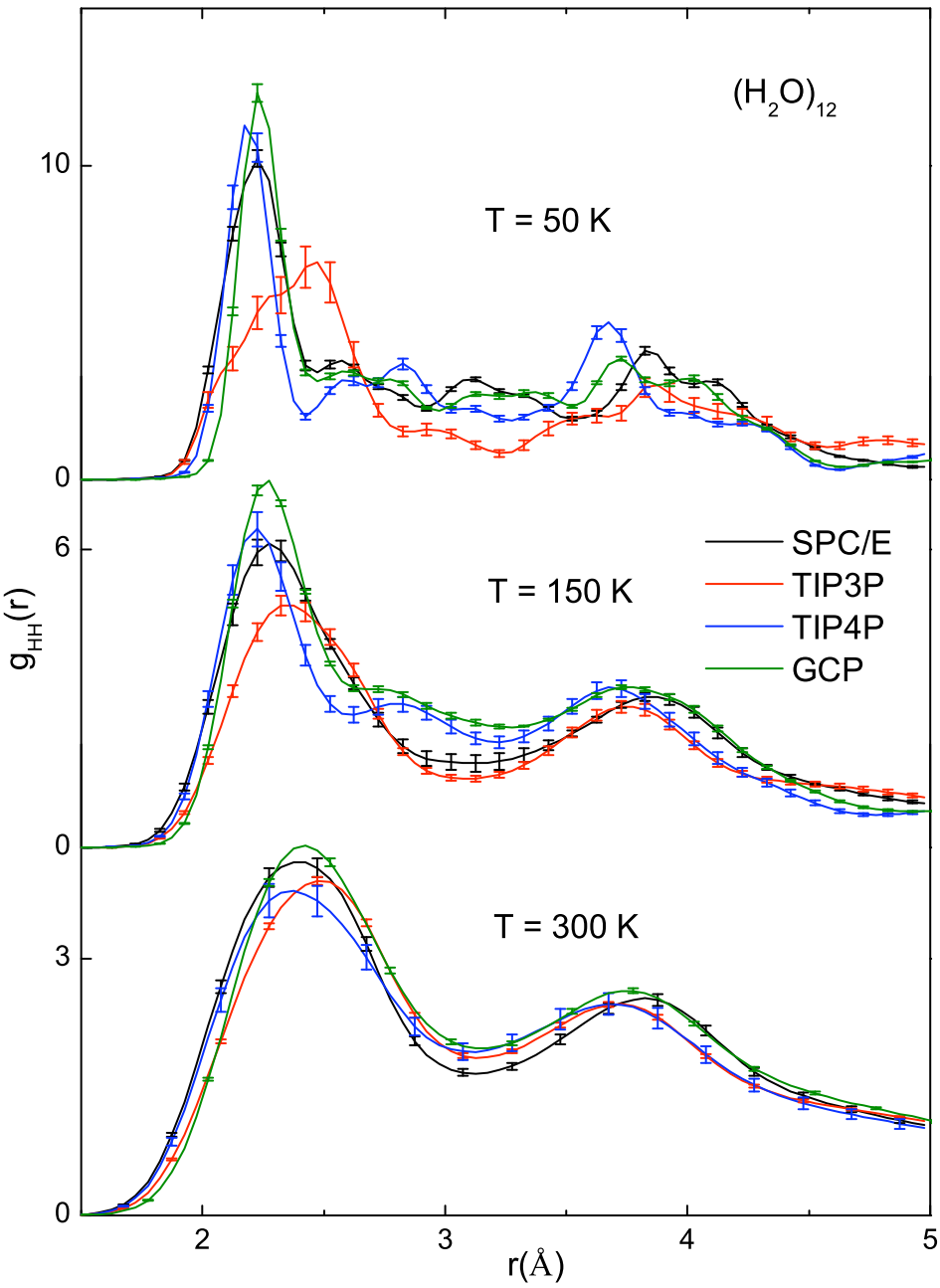

Figure 5.7: Radial distribution functions $g_{HH}(r)$ of $(H_2O)_{12}$ for the H-H pair at $T = 50, 150$ and 300 K. Note that the vertical scale are different for different temperatures.

is due to many distant molecules in the cluster, and the low energy band signals pairs connected with the hydrogen bond. A minima (near E = -9 kJ/mol/molecule for GCP and -10 for others) separating these two parts is used as the threshold energy to identify the hydrogen bond. Relative to TIP3P, the hydrogen bond distributions for SPC/E, TIP4P and GCP are very close to each other (See Fig. 5.8(b)).

5.4 Thermodynamic properties

The appearance of a pronounced peak in the specific heat curve usually signals a phase change of the system. Since the Wang-Landau method can efficiently sample a broad range of energy (from near the ground state energy to any high energy of interest) and provide an estimation of the density of states, we can calculate any thermodynamic quantity, such as internal energy, specific heat, etc, as continuous functions of temperature. There are some common features of the specific heat curves for all four models: at higher temperatures (close to $T = 300$ K), clusters are essentially melted and similar to liquid water, hence for large clusters (about $N > 20$) values of specific heat are roughly the same; when approaching $T = 0$ K, clusters, independent of sizes, seem to have the same limiting value (close to 6R, where R is the gas constant). Note that to accurately study the behavior at low temperatures, quantum effects must be taken into account [65].

The melting behavior of small size water clusters($N < 10$) has been extensively studied [56] for various potential models, since it provides valuable insight both for the empirical models and the cluster systems. Model dependent peak positions were observed in specific heat curves, and bimodal structures were found in potential energy distributions for certain cases.

In Fig. 5.9, our specific heat curves are plotted for $N = 6$ and 8, and the peak positions are consistent with Ref. [56]. For $(H_2O)_6$, specific heat curves are smooth, while $(H_2O)_8$ shows pronounced peaks for all four models, which could indicate sharp melting "transitions". With the density of states and energy histogram, we can easily obtain the energy distribution at

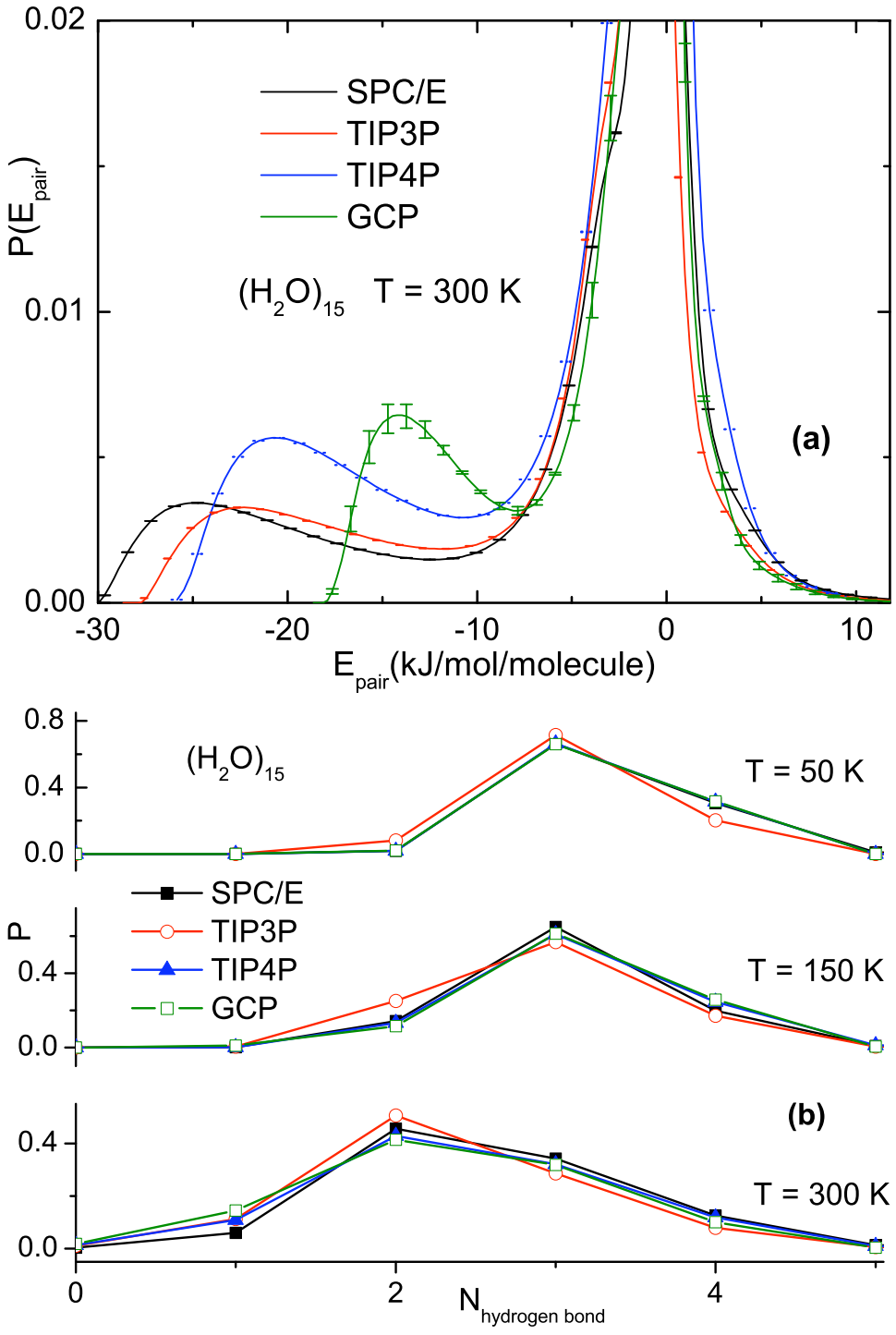

Figure 5.8: $(H_2O)_{15}$: (a) The pair energy distribution at T = 300 K. (b) The hydrogen bond distribution at T = 50, 150 and 300 K.

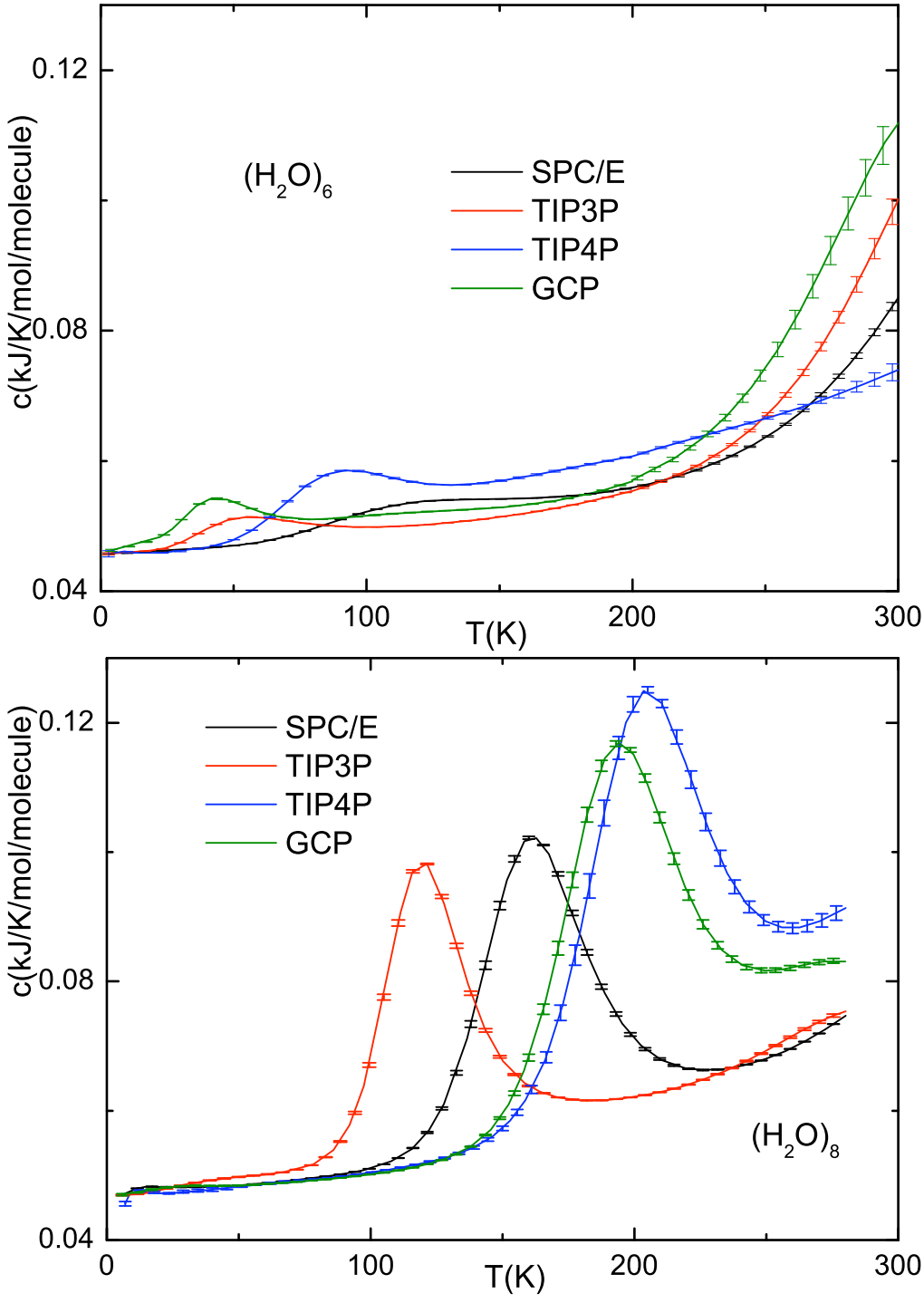

Figure 5.9: Specific heat curves of $(H_2O)_6$ and $(H_2O)_8$.

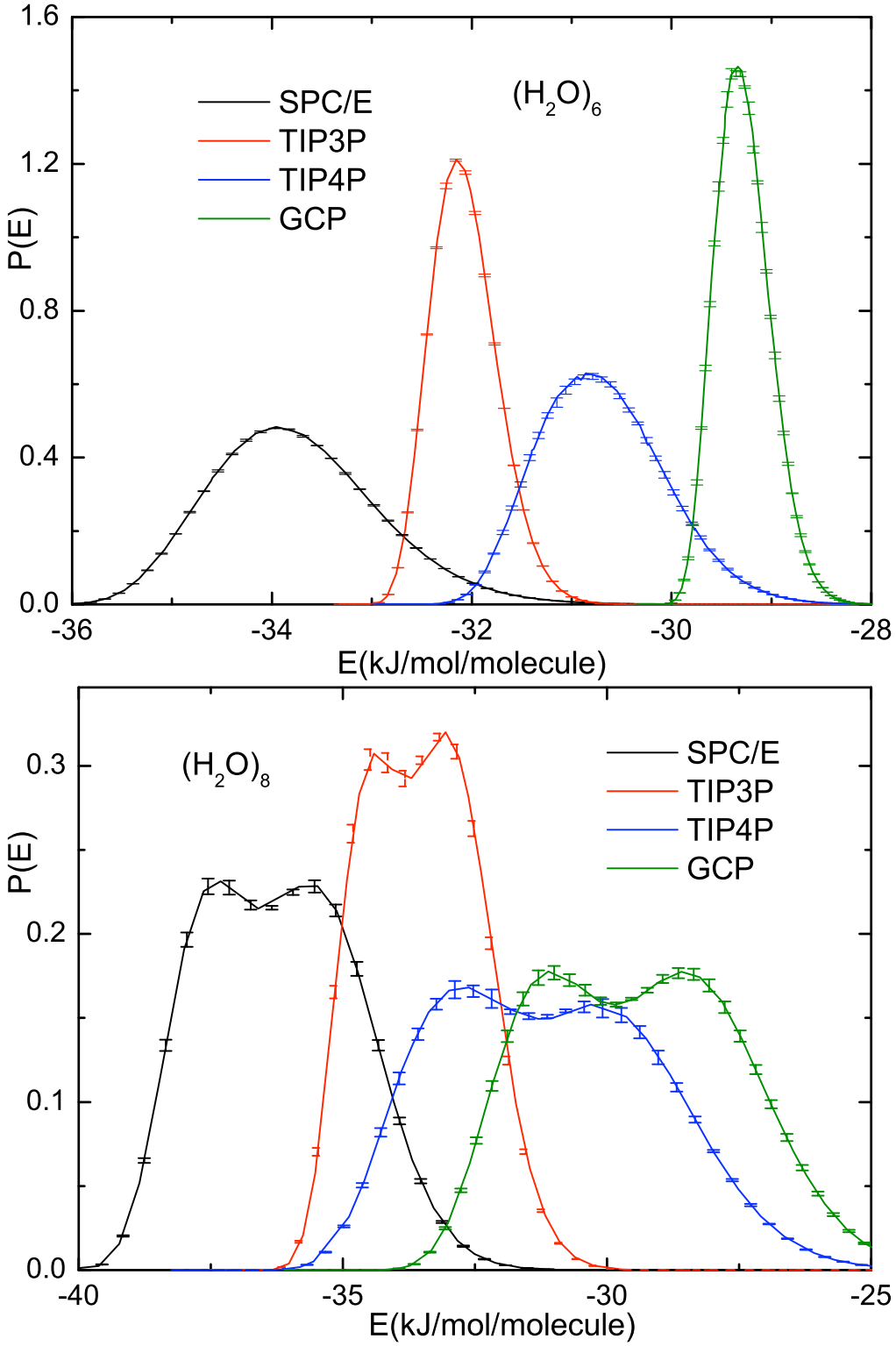

Figure 5.10: The energy distributions of $(H_2O)_6$ and $(H_2O)_8$ near temperatures with peak specific heat.

any temperature, and bimodal distributions are found near peak temperatures for $(H_2O)_8$, as shown in Fig. 5.10.

For the cases of $N = 10$, see Fig. 5.11, the TIP4P and GCP peaks are less pronounced than those for TIP3P and SPC/E, while for $N = 12$, the pattern is reversed. In fact, it depends on the inherent structures of different energy that are accessible at the transition temperatures. For instance, the transition for $(H_2O)_8$ is considered to be from a cubic to non-cubic structures [56], and the sampled distribution of these two have clearly different peak positions, i.e. a bimodal potential energy distribution, therefore, a pronounced transition is observed. While for $(H_2O)_6$ the energy distributions (Fig. 5.10) near temperatures of specific heat peaks show single peaked structures for all models, hence smooth specific heat curves are observed.

To show the effect of TIP4P/2005 reparameterization over TIP4P on the thermodynamics of clusters, we plot specific heat curves for the two models for $N = 8, 10, 12, 15$, as shown in Fig. 5.12. For these clusters, curves are of similar shapes respectively in all cases, but there is about a 20 K shift towards higher temperature after reparameterization.

Actually, in all cases, peak positions of SPC/E are closer to TIP4P than those of TIP3P, and except for the smallest $N = 6$, curves of TIP4P and GCP are similar to each other. Given that GCP provides the best overall results comparing with quantum mechanical calculations and TIP4P is generally recognized to be better than TIP3P, such results seem to favor SPC/E over TIP3P. For larger cluster sizes, the trend becomes more clear, as shown in Fig. 5.13(a) and (b) for $N = 15$ and $N = 20$, respectively. In the case of $N = 30$, the specific curves of SPC/E and TIP4P almost coincide at temperatures below the peak temperature, and as the size of the cluster increases, the transition becomes smooth, see Fig. 5.13(d) for $N = 100$ with potential TIP4P.

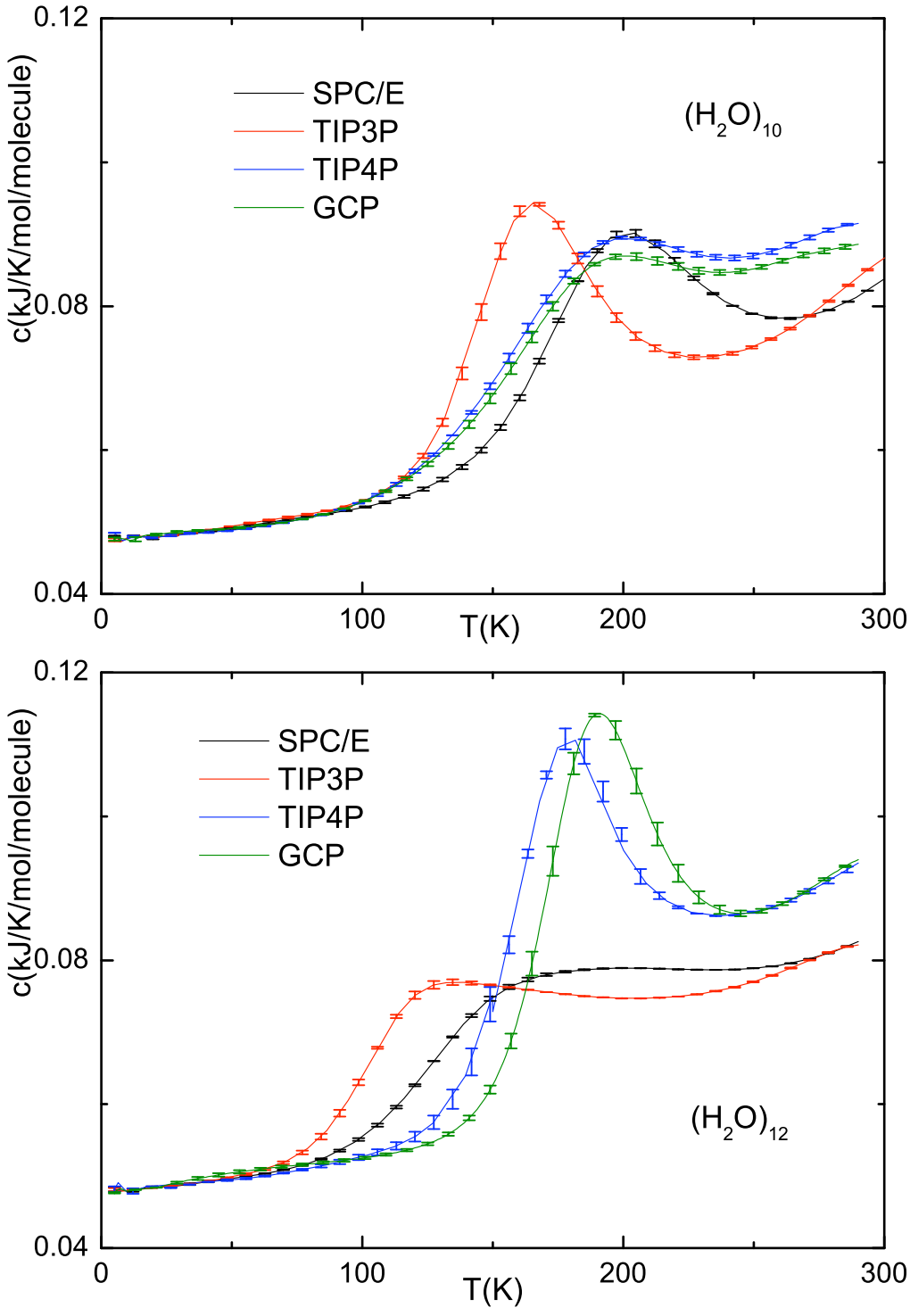

Figure 5.11: Specific heat curves of $(H_2O)_{10}$ and $(H_2O)_{12}$.

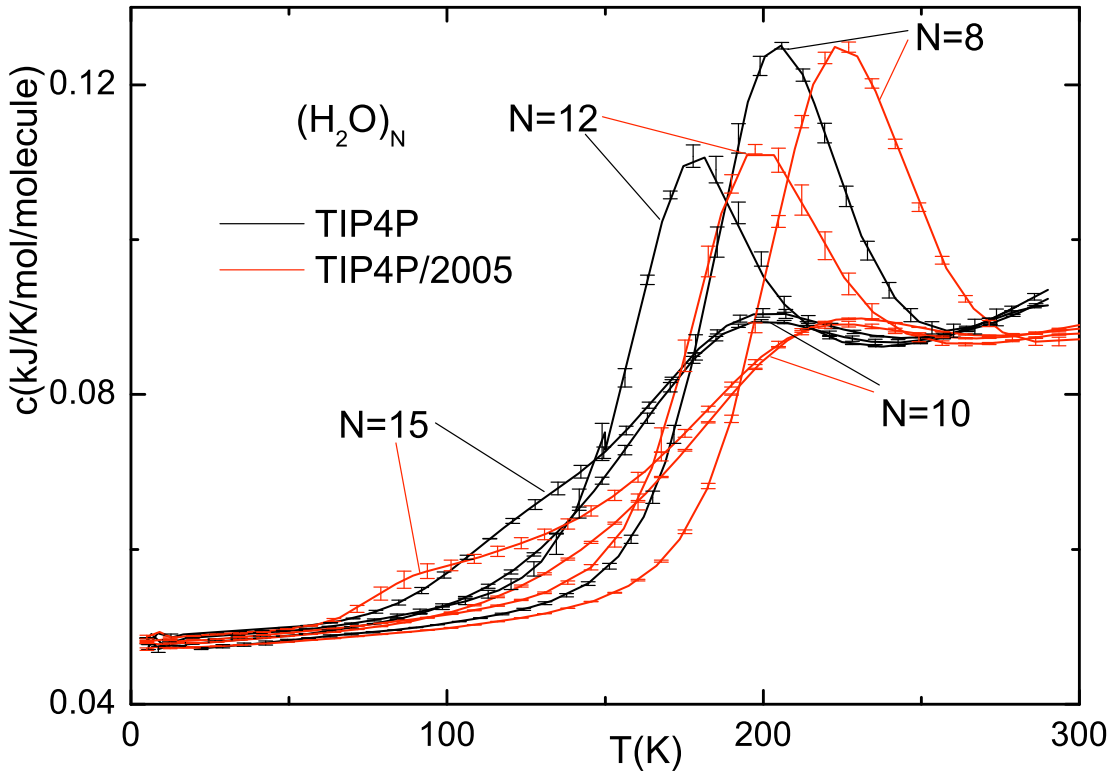

Figure 5.12: Specific heat curves for TIP4P and TIP4P/2005 of $(H_2O)_8$, $(H_2O)_{10}$, $(H_2O)_{12}$ and $(H_2O)_{15}$.

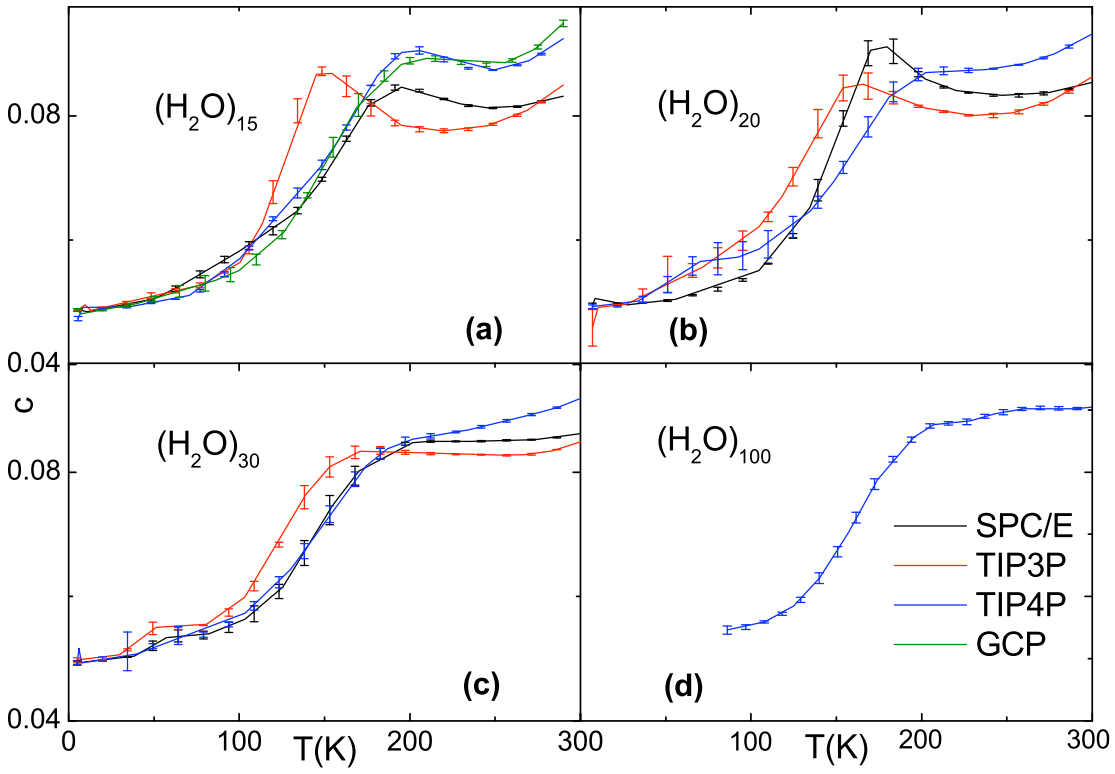

Figure 5.13: Specific heat c (in unit of kJ/K/mol/molecule) for $(H_2O)_{15}$, $(H_2O)_{20}$, $(H_2O)_{30}$ and $(H_2O)_{100}$.

5.5 Suggestion on water model selection

We performed extensive Wang-Landau Monte Carlo simulations of water clusters of sizes up to 30 with four different potential models: SPC/E, TIP3P, TIP4P, and GCP. For structural properties, we estimated ground states, radial distribution functions, and pair energy and hydrogen bond distributions. Since these models were devised to target properties of liquid water, as expected, the behavior of $g(r)$ are similar at high temperatures, and we observed that within error bars the first peak in $g_{OO}(r)$ seems to be independent of both temperature and cluster size ($N > 8$). Although TIP4P and GCP have similar ground state structures in all cases we considered, the GCP model provides the most well defined hydrogen bond energy among the four, and the distance for the nearest-neighbor O-O pair are significantly larger than that of others. Given results [49] from quantum mechanical calculations (about 2.85 and 2.74 Å for liquid water and regular ice, respectively), the GCP model agrees better in the case of liquid water. As for thermodynamic properties, except for $N = 6$, melting transition temperatures of TIP4P and GCP are very close to each other, and results of SPC/E are closer to TIP4P than TIP3P in all cases. Using results of GCP as a criterion, in the perspective of clusters, the quality of the other three widely used water potentials follows: TIP4P $>$ SPC/E $>$ TIP3P. As the cluster size increases, all four models give similar specific heat curves since they are parameterized to reproduce the same bulk quantities. Given an order of magnitude on computation advantage in performance of TIP4P over GCP, and relatively high quality results, we recommend the TIP4P model for Monte Carlo simulations of water clusters and/or inclusions.

5.6 Massively parallel Wang-Landau Sampling on GPUs

For simulations in the previous sections, we used a parallel Wang-Landau scheme with eight random walkers, and by comparing results obtained from the parallel tempering method, we were confident it worked in our case. There are similar applications reported in the literature

[66], in which several, or over hundreds of random walkers, are sampling simultaneously with a common density of states. However, the convergence of such a parallel scheme was not carefully tested before. We should stress that the parallel Wang-Landau method with many random walkers is neither a simple statistical average of many independent simulations, nor a direct extension of single Wang-Landau sampling.

In fact, as shown in the following sections, we find that the originally proposed parameters for single Wang-Landau simulation would not ensure convergence to the correct density of states for the corresponding parallel one, especially in the case of a massively parallel simulation, and we propose a remedy for this potential pitfall.

The first issue to be attended to for a Monte Carlo simulation is the choice of a proper random number generator, and on a GPU, the implementation itself is not trivial. One natural candidate is the linear congruential generator, since it is simple to realize and takes only one or two integers to save the intermediate state on the device memory. For instance, the multiplicative, congruential random number generator RANECU (two integers for the state) has good properties and is suitable for small to medium applications [6]. However, in our simulations, thousands of replicas of the system run simultaneously, so the desired case would require each replica to have an independent random number generator rather than a portion of the same sequence from a single random number generator. The Mersenne Twister (MT) [32, 68] with dynamically generated parameters is an ideal match for such an application, and sample implementation can be found in the CUDA software development kit. MT requires 21 integers to store the intermediate state for each thread; so it runs slightly slower than RANECU. But the MT code [69] we use provides a random number sequence with period 2^{607}, and can accommodate up to 4096 independent random walkers with different MT parameters (essentially different random number generators).

Fig. 5.14 illustrates the scheme for a four GPU implementation. On the GPU level, each card possesses thousands of threads, with each thread representing a random walker. In the plot, these walkers share the same histogram $H(E)$ and the density of states $g(E)$, which

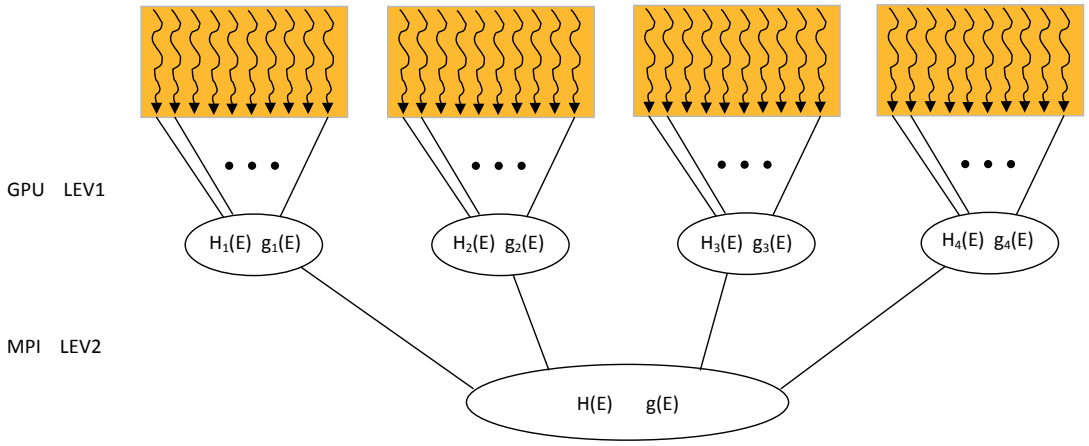

Figure 5.14: Schematic plot of a four GPU implementation of the parallel Wang-Landau algorithm, where $H(E)$ is the energy histogram and $g(E)$ is the density of states. Each rectangle stands for a GPU card which contains thousands of threads.

can result in substantial reduction in the performance due to the memory access conflict. So, in practice, a "gather" pattern is employed where each thread has an independent memory space for $H(E)$ and $g(E)$. Every LEV1 Monte Carlo steps, a reduction procedure is invoked to synchronize the information across all threads. Hence the net effect would be all the walkers share the same $H(E)$ and $g(E)$. On the MPI level, each GPU works as a big random walker, and their accumulated $H(E)$ and $g(E)$ also need to be synchronized every LEV2 Monte Carlo step. Note that CUDA and MPI are two orthogonal components, which means GPUs can not communicate with each other directly through MPI and the synchronization is actually across CPUs by MPI. The choice of LEV1 and LEV2 depends on the tradeoff between the overhead in each level of synchronizing and the rate of convergence to the true density of states.

There are several other issues that need to be considered for applications on a GPU. For instance, since on the current generation GPU, a single floating point operation is much faster than that in double precision (on a Fermi architecture card, it is about two times

faster), a mixed precision calculation is often used to improve the performance. This should be done with great caution. At least, the compensated summation [70] needs to be employed in the calculation of the total energy of the continuous system to minimize the truncation error. Also, for large applications, issues like register overspilling, memory coalescing, etc., could greatly affect the performance.

5.7 Convergence issues

In fact, parallel Wang-Landau sampling has already been used in several applications [66]. It seems that the originally proposed Wang-Landau parameters work fine in those specific cases. However, we find this is not generally true, especially for simulations with large numbers of parallel random walkers. In Fig. 5.15(a), we plot the time series of the energy for several walkers in the simulation of $(H_2O)_{12}$ clusters with 32 walkers in total. It shows that some walkers seem to get "stuck" at low energy, which signals the underestimation of the density of states in the corresponding energy region, as shown in Fig. 5.15(b). The explanation is that although in Wang-Landau sampling every energy state is given the same probability, given a certain energy the ability for a walker to visit the neighboring energy state is not the same [71]. Usually, the lower energy region (in our case, the "solid" phase) is much more difficult to approach, as a result of which the density of states tends to be underestimated in the first few iterations. Therefore, once a walker gets in the region, it will stay there to rectify the underestimation. The problem is the regular Wang-Landau parameters cannot ensure convergence to the true density of states in parallel Wang-Landau simulations. Of course one could make the flatness criterion more stringent, e.g., requiring that every walker visits the entire energy range. It may rectify the problem, but it also reduces the efficiency brought by parallelism in the first place.

One remedy follows the idea of frontier sampling (also known as the global update) [4]. At the beginning of each iteration, an artificial distribution could be added to the density of states to force more walkers to sample towards the lower energy region. As shown in

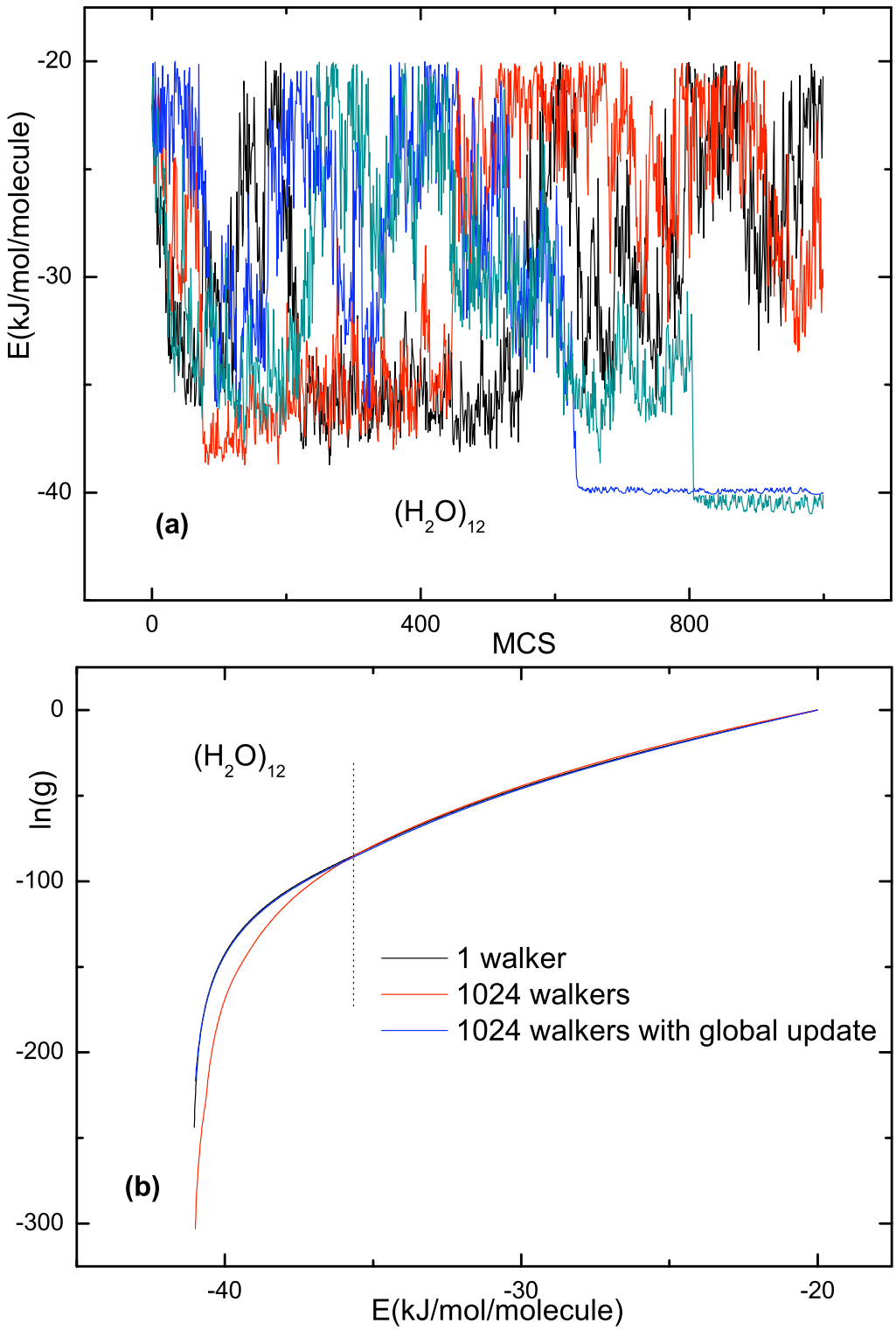

Figure 5.15: (a) Energy time series of 4 typical random walkers (different color) in a simulation of totally 32 random walkers. (b) Density of states from simulations with the same Wang-Landau parameters (modification factor, flatness criterion and number of iterations) but different number of random walkers with or without global update.

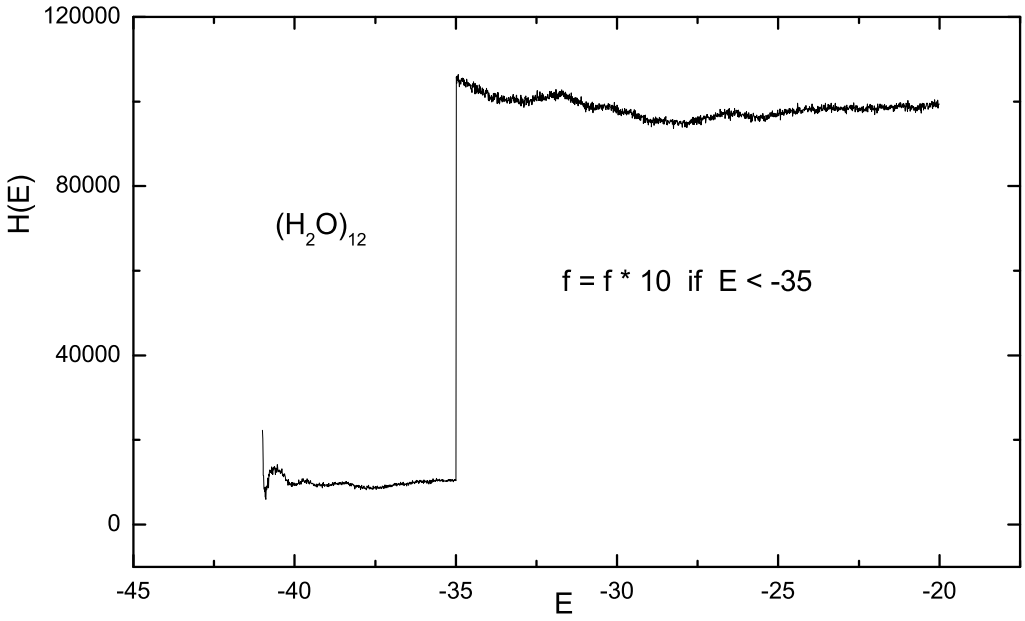

Figure 5.16: Energy histogram for simulations with a non-uniform modification factor.

Fig. 5.15(b), it appears to fix the problem. But the choice of the additive distribution seems to be arbitrary and the system is very sensitive to it. If it is too small, then the lower energy region is still underestimated; If it is too big, then the density of states is distorted too much and longer computation times are needed to converge.

5.8 Non-uniform modification factor

We propose to use a non-uniform modification factor f for parallel Wang-Landau sampling. In the energy region where a walker is harder to approach, a larger modification factor is added to the density of states for each visit, hence a faster update rate in the region. One natural choice is to use a step function for the modification factor, and we find it already works well for the water cluster system. In simulations, $\log(f)$ and $\log(g)$ are usually used and our iteration factor would be $(\kappa\Theta(E_0 - E) + 1)\log(f)$, where Θ is the Heaviside step

function. Basically, $\log(g)$ is updated κ times faster where $E < E_0$. The determination of E_0 depends on prior knowledge, such as the average energy for the interested transition temperature, or one can perform some preliminary analysis of the energy time series. Also, the value of κ needs to be tested. A large value won't overshoot the density of states, just takes longer to converge. From the updating scheme $\log(g) \leftarrow \log(g) + \log(f)$, the final value of the density of states needs to be corrected following $\log(g) \leftarrow \log(g) + \log(\kappa)\Theta(E_0 - E)$, and correspondingly, $H \leftarrow H/(1 + \kappa\Theta(E_0 - E))$ for the histogram. Of course, now the histogram is no longer flat, see Fig. 5.16, just like the case in optimal ensemble sampling [71]. The advantage of parallel Wang-Landau sampling is that with so many random walkers, a simple step-function-like energy histogram already works well.

For $(H_2O)_{12}$ clusters, we adopted E_0 = -35 kJ/mol/molecule and $\kappa = 50$, and obtained the correct convergence efficiently. In Fig. 5.17, we plot the corrected energy time series, and the specific heat curve. Note that one side effect of the large iteration factor is that the statistical error in the lower energy region is also magnified. If the low energy is of interest, extra iterations may be necessary. Luckily, the computational time will not be excessive for massively parallel Wang-Landau simulations. We also test the reverse scheme where instead of increasing f in the lower energy region, we decrease it in the higher energy region. For $(H_2O)_{12}$, we use $\kappa = 0.2$ for $E >$ -35 kJ/mol/molecule, which also works well, and it seems the reverse scheme is even more robust.

5.9 Performance comparisons

In CUDA thread Hierarchy [16], threads are grouped into blocks, and blocks are further grouped into grids. Usually thousands of threads are needed for a GPU application to run efficiently, and deciding how to divide the threads into (grid, block) configuration actually affects the performance. In Fig. 5.18(a), we plot the computing time per MC step for different setups on a Tesla C2050 card. From the results, generally, using a large block is not bad, but for large applications there are often not enough resources to accommodate so many threads

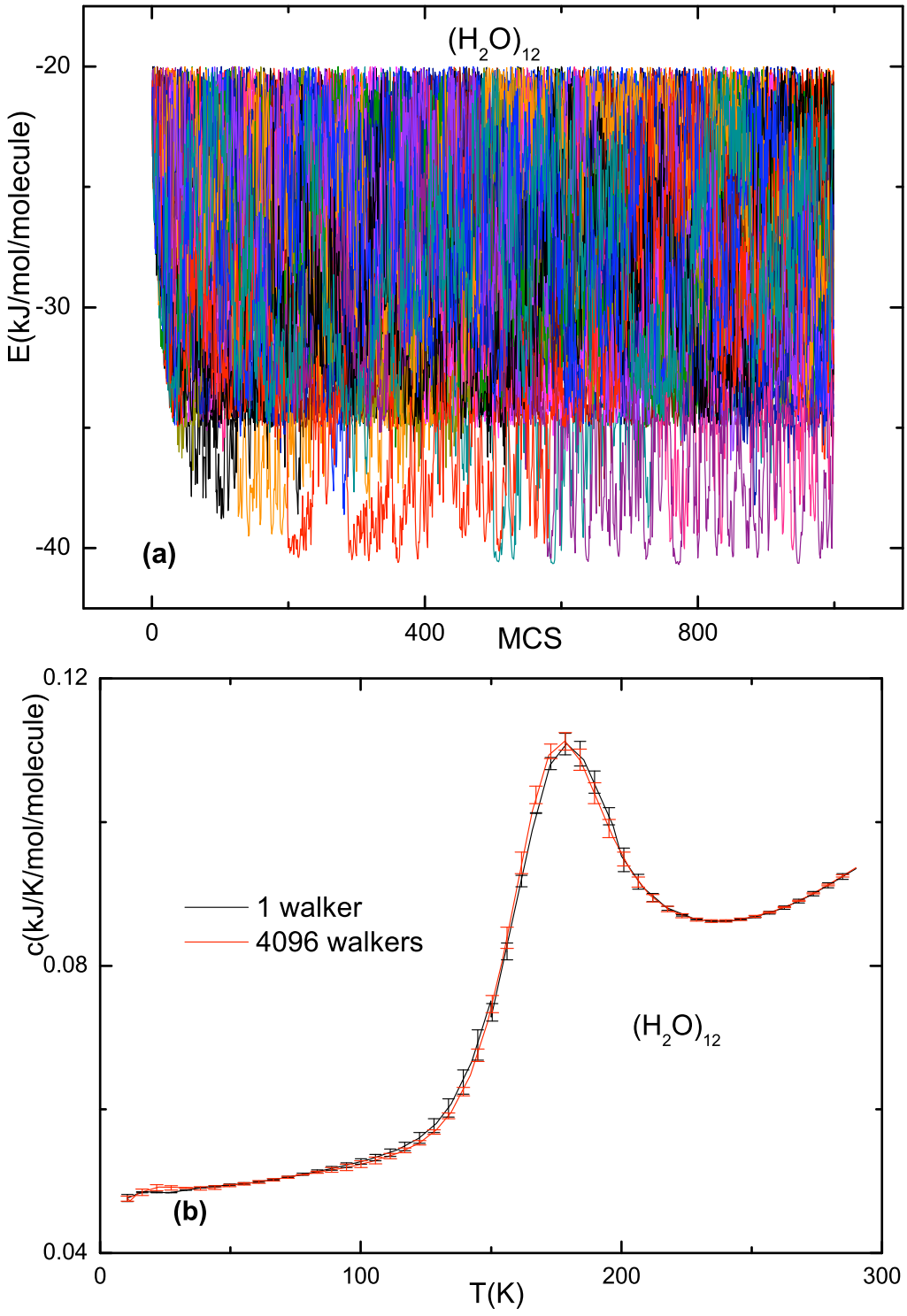

Figure 5.17: $(H_2O)_{12}$: (a) Energy time series for all 32 random walkers (different color). (b) Specific heat curves for single Wang-Landau and parallel Wang-Landau with 4096 random walkers.

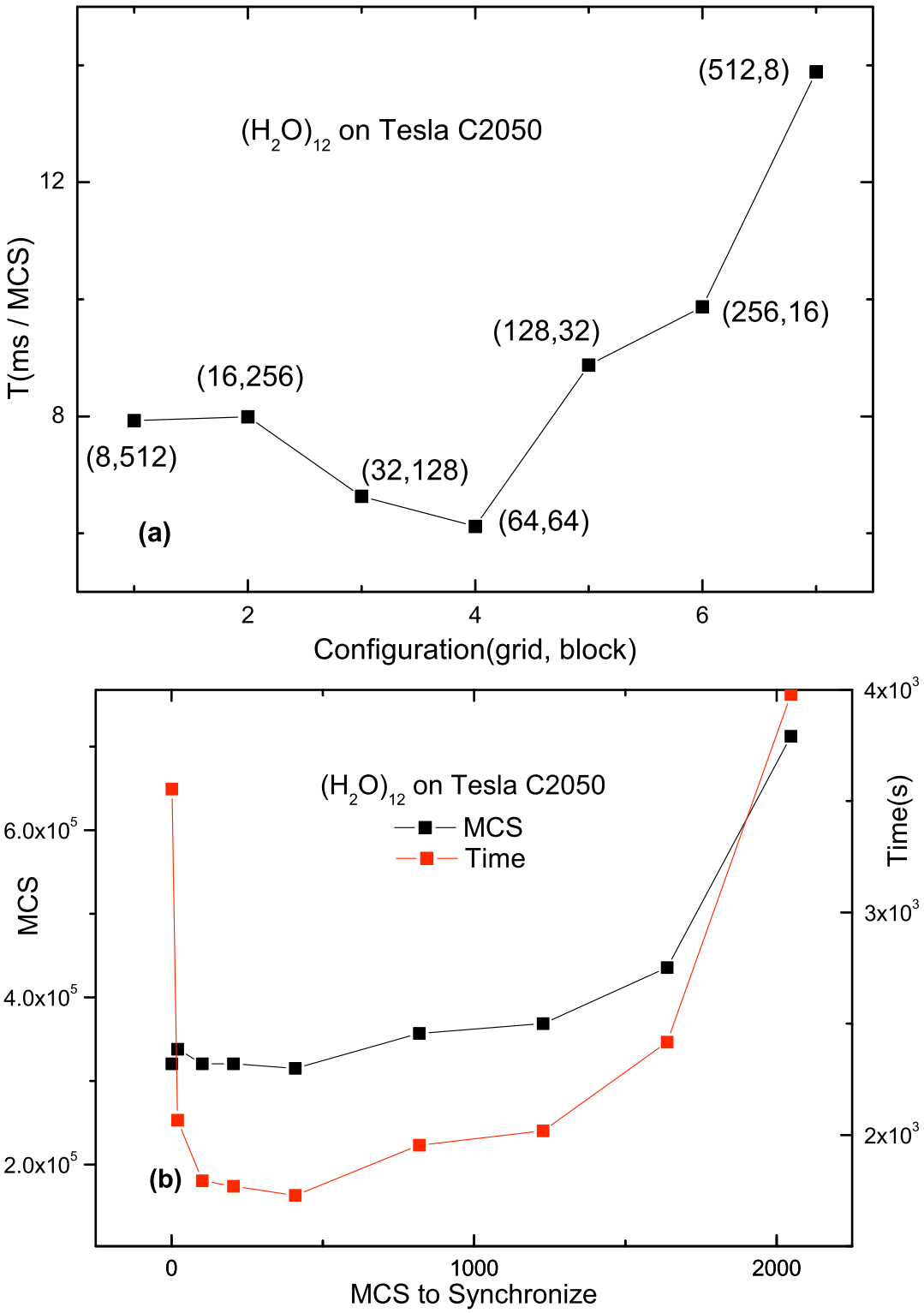

Figure 5.18: (a) Computing time per MCS for different (grid, block) configurations on Tesla C2050. (b) Total number of MCS and computing time versus number of MCS between two consecutive synchronization of the energy histogram and the density of states.

within a block, hence a comparable grid to block size gives the best performance for our code. For parallel Wang-Landau applications, the frequency of two consecutive synchronizations of the information, such as the density of states, the energy histogram, etc, will also affect the efficiency. Basically, it is a tradeoff between the overhead of the reduction and the convergence rate, as shown in Fig. 5.18b. Synchronizing too often or too few will reduce the performance, and every several hundreds of MC steps seems to be the best frequency for our application.

In Fig. 5.19(a), we compare the performance of the single thread CPU (i7-930) code with the single GPU implementation on three different cards. GTX285 and Tesla S1070 have comparable performance, which is about 30 times faster than i7-930, and Tesla C2050 (Fermi architecture) outperforms the other two GPU cards by almost a factor of two. The main reason is that the Fermi card offers twice the amount of streaming processor cores and provides an order of magnitude faster peak performance in double precision (we use mixed precision). To show the scalability of our implementation, we plot the number of MC steps per second versus the number of GPU cards used, as shown in Fig. 5.19(b). Each card runs with 1024 threads, and the overall performance scales linearly with the number of cards.

To further illustrate the efficiency of the GPU implementation, we study a much bigger cluster $(H_2O)_{50}$. The specific heat and two snapshots for corresponding temperatures and another snapshot for the ground state structure (after quenching to the local minima by direction set and conjugate gradient methods), are plotted in Fig. 5.20. The melting "transition" seems to be very smooth, which is related to distributions of inherent structures near the transition temperatures [56, 67]. The ground state energy we find is -46.05kJ/mol/molecule.

5.10 Remarks on the multi-GPU Wang-Landau sampling

We presented a multiple GPU implementation of parallel Wang-Landau sampling. A potential pitfall for such applications is pointed out in that single Wang-Landau parameters would

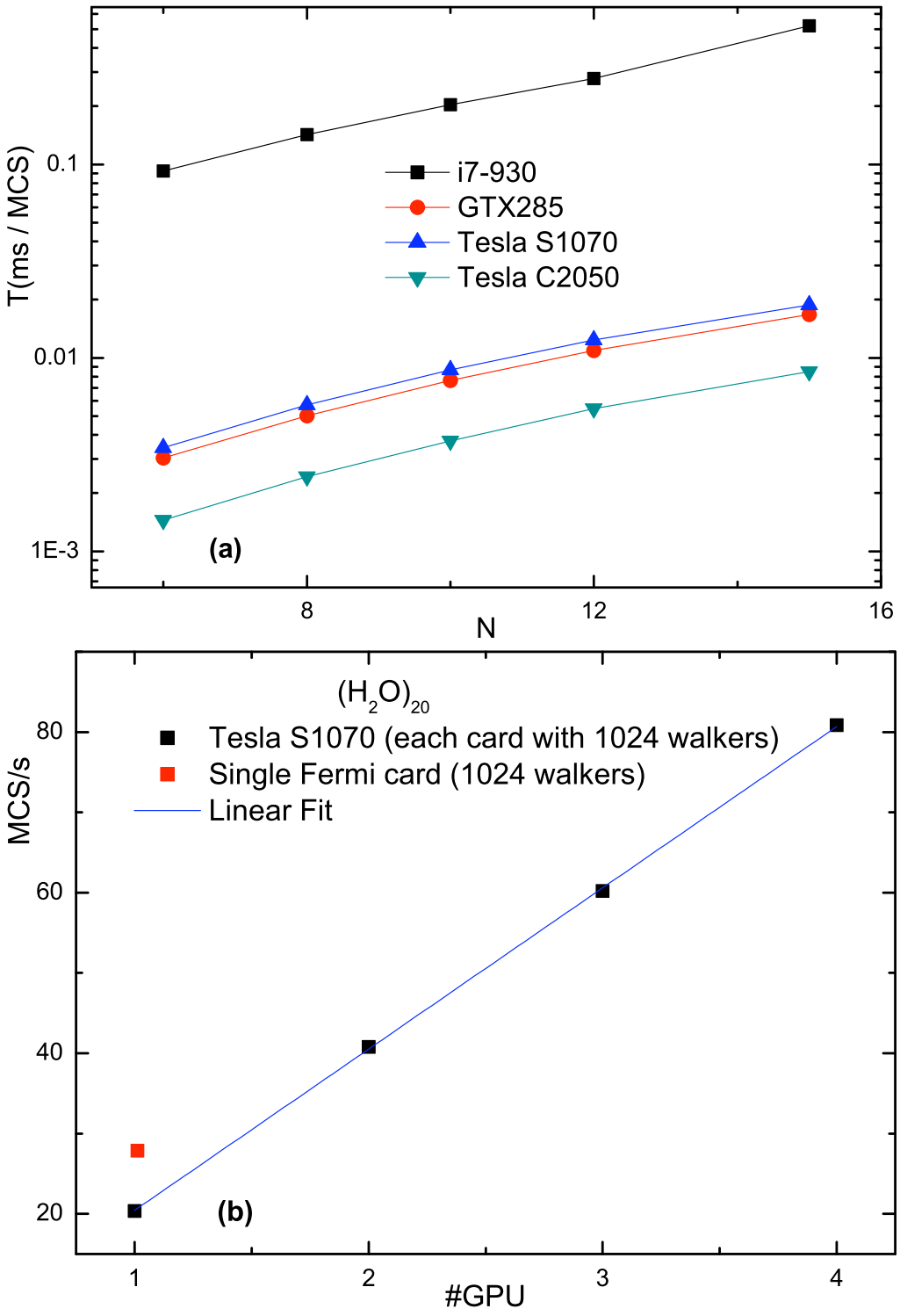

Figure 5.19: (a) Computing time per MCS for i7-930 CPU versus three types of GPU. (b) MCS per seconds versus the number of Tesla S1070 GPU cards.

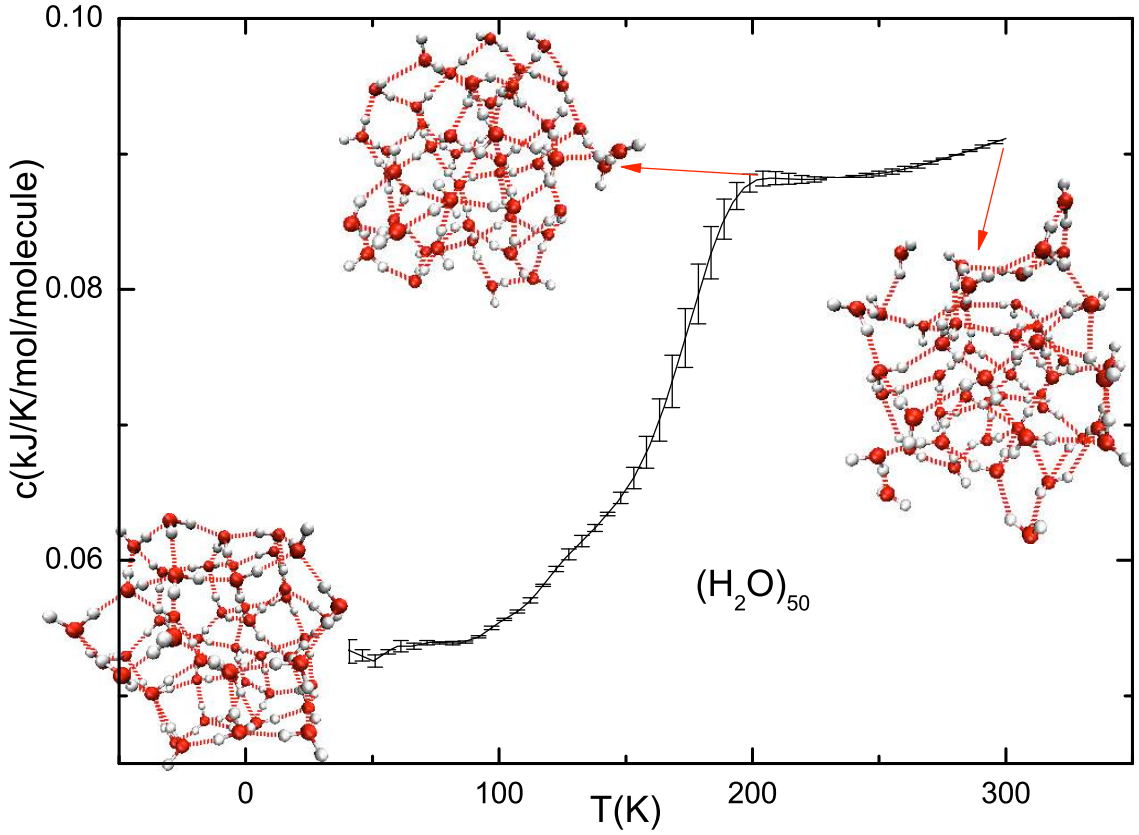

Figure 5.20: Specific heat curve for $(H_2O)_{50}$. Two snapshots (with pointing arrow) are taken at the average energy for corresponding temperatures, respectively, and another snapshot (leftmost) is for the ground state structure after quenching. The red ball stands for the Oxygen atom, the white ball for the Hydrogen atom, and the dash line for hydrogen bonding.

not ensure the convergence in a parallel Wang-Landau simulation. We propose using a non-uniform modification factor to accelerate the convergence. By choosing an appropriate form of the modification factor, the convergence is restored and we obtain an average of over 50 times speedup for a given workload compared to a single thread code on an intel i7-930. We were able to obtain both the ground state structure and the thermodynamics of $(H_2O)_{50}$, which was not evaluated before. Tesla C2050 with the new generation Fermi Architecture outperforms the old GT200 Architecture (GTX285, Tesla S1070) by about a factor of two. According to our test on Tesla S1070 four GPU clusters, our implementation scales linearly with the number of GPU.

Bibliography

[1] F.A. Kassan-Ogly and B.N. Filippov, Solid State Phen. **168** 427 (2011).

[2] A. O'Hare, F.V. Kusmartsev and K.I. Kugel, Int. J. Mod. Phys. B **23** 3951 (2009).

[3] F. G. Wang and D. P. Landau, Phys. Rev. Lett. **86** 2050 (2001); Phys. Rev. E**64** 056101 (2001); D.P. Landau, S.-H. Tsai, and M. Exler, Amer. J. Phys. **72** 1294, (2004).

[4] C. G. Zhou, T. C. Schulthess, S. Torbrugge and D. P. Landau, Phys. Rev. Lett. **96** 120201 (2006); C. G. Zhou, T. C. Schulthess and D. P. Landau, J. Appl. Phys **99** 08H906 (2006); T. Wüst and D. P. Landau, Phys. Rev. Lett **102** 178101 (2009); C. Desgranges and J. Delhommelle, J. Chem. Phys. **120** 244109 (2009); D. T. Seaton, T. Wüest and D. P. Landau, Phys. Rev. E**81** 011802 (2010); F. Lou and P. Clote, Bioinformatics **26** i278 (2010);

[5] T. Preis, P. Virnau, W. Paul, and J.J. Schneider, J. Comp. Phys. **228** 4468 (2009); B. Block, P. Virnau and T. Preis, Comp. Phys. Comm. **181** 1549 (2010);

[6] J. Yin and D. P. Landau, Phys. Rev. E**80** 051117 (2009); Phys. Rev. E**81** 031121 (2010).

[7] M.E. Fisher and M.N. Barber, Phys. Rev. Lett. **28** 1516 (1972).

[8] Alan M. Ferrenberg and D.P. Landau, Phys. Rev. B **44** 5081 (1991).

[9] G.S. Rushbrooke, J. Chem. Phys. **39** 842 (1963).

[10] J. J. Rehr and N. D. Mermin, Phys. Rev. A **8** 472 (1973).

[11] A. D. Bruce and N. B. Wilding, Phys. Rev. Lett. **68** 193 (1992); J. Phys.: Condens. Matter **4** 3087 (1992).

[12] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, J. Chem. Phys. **21** 1087 (1953).

[13] R.H. Swendsen and J.S. Wang, Phys. Rev. Lett **57** 2607 (1986).

[14] K. Hukushima and K. Nemoto, J. Phys. Soc. Japan **65** 1604 (1996).

[15] S. Kirkpatrick, C. D. Gelatt, Jr. and M. P. Vecchi, Science **220** 671 (1983).

[16] NVIDIA Corporation, NVIDIA CUDA – Programming Guide, version 3.1.1 (2010).

[17] D.P. Landau, J. Appl. Phys. **42** 1284 (1971).

[18] D.P. Landau, Phys. Rev. B **21** 1285 (1980).

[19] K. Binder and D.P. Landau, Phys. Rev. B **21** 1941 (1980).

[20] E. Domany, M. Schick, J.S. Walker, and R.B. Griffiths, Phys. Rev. B **18** 2209 (1978).

[21] D.P. Landau, Phys. Rev. B **27** 5604 (1983).

[22] P.A. Slotte, J. Phys. C **16** 2935 (1983).

[23] K. Kaski, W. Kinzel, and J.D. Gunton, Phys. Rev. B **27** 6777 (1983).

[24] J. Amar, K. Kaski, and J.D. Gunton, Phys. Rev. B **29** 1462 (1984).

[25] J.L. Moran-Lopez, F. Aguilera-Granja, J.M. Sanchez, Phys. Rev. B **48** 3519 (1993).

[26] J.L. Moran-Lopez, F. Aguilera-Granja, J.M. Sanchez, J. Phys. C **6** 9759 (1994).

[27] M. Suzuki, Prog. Theor. Phys. **51** 1992 (1974).

[28] A. Malakis, P. Kalozoumis, and N. Tyraskis, Eur. Phys. J. B **50** 63 (2006).

[29] D.A. Kofke, J. Chem. Phys. **120** 10852 (2004).

[30] F. James, Comp. Phys. Comm. **60** 329 (1990).

[31] P. L'Ecuyer, Commun. ACM. **31** 742 (1988).

[32] M. Matsumoto and T. Nishimura, ACM Trans. on Modeling and Computer Simulation **8** 3 (1998).

[33] J. Yang, Y. Wang, and Y. Chen, J. Comp. Phys. **221** 799 (2007).

[34] J.A. Anderson, C.D. Lorenz, and A. Travesset, J. Comp. Phys. **227** 5342 (2008).

[35] G.Y. Hu and S.C. Ying, Physica **140A** 585 (1987).

[36] For reviews see J. G. Dash, Phys. Rep. **38C** 177 (1978); E. Bauer, *Structure and Dynamics of Surfaces II*, edited by W. Schommers and P. von Blanckenhagen (Springer, Berlin, 1987); See also, Adv. Chem. Phys. Molecule Surface interactions, edited by K. P. Lawley (Wiley, London, 1989).

[37] K. Binder and D. P. Landau, Adv. Chem. Phys., Molecule Surface Interactions, edited by K. P. Lawley, 91 (Wiley, London, 1989).

[38] K. Binder and D. P. Landau, Phys. Rev. B **21** 1941 (1980).

[39] K. Binder, W. Kinzel, and D. P. Landau, Surf. Sci. **117** 232 (1982); D. P. Landau, Phys. Rev. B **27** 5604 (1983).

[40] T. L. Einstein and J. R. Schrieffer, Phys. Rev. B **7** 3679 (1973).

[41] K. Binder and D. P. Landau, Surf. Sci. **108** 503 (1981).

[42] A. Milchev, M. Paunov, Surf. Sci. **108** 25 (1981); A. Milchev, Electrochem. Acta **28** 941 (1983).

[43] O. A. Pinto, A. J. Ramirez-Pastor and F. Nieto, Surf. Sci. **602** 1763 (2008).

[44] See, for example, *Finite Size Scaling and Numerical Simulation of Statistical Systems*, edited by V. Privman (World Scientific, Singapore, 1990); D. P. Landau and K. Binder,

A Guide to Monte Carlo Simulation in Statistical Physics (Cambridge University Press, Cambridge, U.K., 2000).

[45] J. M. Kosterlitz and D. J. Thouless, J. Phys. C **6** 1181 (1973).

[46] M. P. M. den Nijs, J. Phys. A **12** 1857 (1979); B. Nienhuis, A. N. Berker, E. K. Riedel, and M. Schick, Phys. Rev. Lett. **43** 737 (1979); R. B. Pearson, Phys. Rev. B **22** 2579 (1980); D. P. Landau and R. H. Swendsen, Phys. Rev. Lett **46** 1437 (1981); Phys. Rev. B **33** 7700 (1986).

[47] J.L. Moran-Lopez, F. Aguilera-Granja, J.M. Sanchez, Phys. Rev. B **48** 3519 (1993); J. Phys. C **6** 9759 (1994); C. Buzano, M. Pretti, Phys. Rev. B **56** 636(1997).

[48] A. Malakis, P. Kalozoumis, and N. Tyraskis, Eur. Phys. J. B **50** 63 (2006).

[49] R. Ludwig, Angew. Chem. Int. Ed. **40** 1808 (2001); G. S. Fanourgakis, E. Aprà and S. S. Xantheas, J. Chem. Phys. **121** 2655 (2004); A. Lenz and L. Ojamäe, Phys. Chem. Chem. Phys. **7** 1905 (2005); S. Bulusu, S. Yoo, E. Aprà, S. Xantheas and X. C. Zeng, J. Phys. Chem. A **110** 11781 (2006); J. Cui, H. Liu and K. D. Jordan, J. Phys. Chem. B **110** 18872 (2006); R. Bukowshi, K. Szalewicz, G. C. Groenenboom and A. van der Avoird, Science **315** 1249 (2007).

[50] D. J. Wales and M. P. Hodges, Chem. Phys. Lett. **286** 65 (1998).

[51] H. Kabrede, Chem. Phys. Lett **430** 336 (2006).

[52] H. Takeuchi, J. Chem. Inf. Model. **48** 2226 (2008).

[53] S. Kazachenko and A. J. Thakkar, AIP Conf. Proc. **1108** 90 (2009); Chem. Phys. Lett. **476** 120 (2009).

[54] J. A. Niesse, H. R. Mayne, J. Comp. Chem. **18** 1233 (1997).

[55] H. Kabrede and R. Hentschke, J. Phys. Chem. B **107** 3914 (2003).

[56] C. J. Tsai and K. D. Jordan, J. Chem. Phys. bf 95 3850 (1991); J. M. Pedulla and K. D. Jordan, Chem. Phys **239** 593 (1998); A. N. Tharrington and K. D. Jordan, J. of Phys. Chem. A **107** 7380 (2003).

[57] K. Liu, M. G. Brown, C. Carter, R. J. Saykally, J. K. Gregory and D. C. Clary, Nature **381** 501 (1996); K. Liu, J. D. Cruzan and R. J. Saykally, Science **271** 929 (1996); C. J. Gruenloh, J. R. Carney, C. A. Arrington, T. S. Zwier, S. Y. Fredericks and K. D. Jordan, Science, **276** 1678 (1997); C. J. Gruenloh, J. R. Carney, F. C. Hagemeister, C. A. Arrington, T. S. Zwier, S. Y. Fredericks, J. T. Wood and K. D. Jordan, J. Chem. Phys. **109** 6601 (1998); M. G. Brown, F. N. Keutsch and R. J. Saykally, J. Chem. Phys. **109** 9645 (1998); C. J. Gruenloh, J. R. Carney, F. C. Hagemeister, T. S. Zwier, T. J. Wood and K. D. Jordan, J. Chem. Phys. **113** 2290 (2000).

[58] W. L. Jorgensen, J. Am. Chem. Soc. **103** 335 (1981); J. Chem. Phys. **77** 4156 (1982).

[59] P. T. Kiss and A. Baranyai, J. Chem. Phys. **131** 204310 (2009).

[60] P. Paricaud, M. Psedota, A. A. Chialvo and P. T. Cummings, J. Chem. Phys. **122** 244511 (2005).

[61] G. Ganzenmüller and P. J. Camp, J. Chem. Phys. **127** 154504 (2007).

[62] U. H. E. Hansmann, Y. Okamoto and F. Eisenmenger, Chem. Phys. Lett. **259** 321 (1996).

[63] W. Press, S. Teukolsky, W. Vetterling and B. Flannery, *Numerical Recipes in C* (Cambridge University Press, NY, USA, 1992)

[64] J. L. F. Abascal and C. Vega, J. Chem. Phys. **123** 234505 (2005).

[65] C. McBride, C. Vega, E. G. Noya, R. Ramírez, and L. M. Sesé, J. Chem. Phys. **131**, 024506 (2009); E. G. Noya, C. Vega, L. M. Sesé, and R. Ramírez, J. Chem. Phys. **131**

124518 (2009); C. Vega, M. M. Conde, C. McBride, J. L. F. Abascal, E. G. Noya, R. Ramírez, and L. M. Sesé, J. Chem. Phys. **132** 046101 (2010).

[66] L. Zhan, Comp. Phys. Comm. **179** 339 (2008); M. Eisenbach, C.-G. Zhou, D. M. Nicholson, G. Brown, J. Larkin and T. C. Schulthess, SC, Portland, Oregon, USA, November 14-20. ACM, New York (2009).

[67] J. Yin and D. P. Landau, J. Chem. Phys. **134** 074501 (2011).

[68] M. Matsumoto and T. Nishimura, Monte Carlo and Quasi-Monte Carlo Methods, 56 (2000).

[69] V. Podlozhnyuk, Parallel Mersenne Twister, *Nvidia CUDA SDK 1.0* (2007).

[70] W. Kahan, Comm. ACM **8** 40 (1965).

[71] S. Trebst, D. A. Huse and M. Troyer, Phys. Rev. E **70** 046701 (2004).

Appendix A

parallel tempering with checkerboard scheme

```
//main kernel function for 2d square lattice Ising model
//with two and three body interactions.
//SUPER_BLOCK_SIZE = # of replicas
//BLOCK_SIZE = THREAD_SIZE = half lattice size(L/2)
//therefore, each thread is in charge of a (2x2) lattice unit
__global__ void device_function_main(int* S,int* R,int sub) {
// S = Spin array; R = Random number array; sub = sublattice;
int dH1=0;
int dH2=0;
int dH3=0;
float ae;
int I1,I2,IZ;
//Allocate shared memory
__shared__ int r[2*THREAD_SIZE];
r[threadIdx.x] = R[threadIdx.x+THREAD_SIZE*blockIdx.y+THREAD_SIZE*BLOCK_SIZE*blockIdx.x];
r[threadIdx.x+THREAD_SIZE] = R[threadIdx.x+THREAD_SIZE*blockIdx.y
                               +THREAD_SIZE*BLOCK_SIZE*blockIdx.x
                               +THREAD_SIZE*BLOCK_SIZE*SUPER_BLOCK_SIZE];
__syncthreads();
int site;

switch(sub) {
 case(1):
//Create new random numbers; RANECU
 I1= r[threadIdx.x]/53668;
 r[threadIdx.x]=40014*( r[threadIdx.x]-I1*53668)-I1*12211;
 if( r[threadIdx.x]<0)
    r[threadIdx.x]= r[threadIdx.x]+2147483563;

 I2= r[threadIdx.x+THREAD_SIZE]/52774;
 r[threadIdx.x+THREAD_SIZE]=40692*( r[threadIdx.x+THREAD_SIZE]-I2*52774)-I2*3791;
 if( r[threadIdx.x+THREAD_SIZE]<0)
    r[threadIdx.x+THREAD_SIZE]= r[threadIdx.x+THREAD_SIZE]+2147483399;

 IZ= r[threadIdx.x]- r[threadIdx.x+THREAD_SIZE];
 if(IZ<1) IZ=IZ+2147483562;
```

```
 site=blockIdx.x*L+2*threadIdx.x+2*L*SUPER_BLOCK_SIZE*blockIdx.y;
//Spin update top right
 if(blockIdx.y==0) { //Top
    if(threadIdx.x==0) { //Left
   dH1=S[site]*( S[site-1+L]+S[site+1]+S[site+L*SUPER_BLOCK_SIZE]
                    +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]);
   dH2=S[site]*( S[site+L*SUPER_BLOCK_SIZE-1+L]
                    +S[site+L*SUPER_BLOCK_SIZE+1]
                    +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1+L]
                    +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]);
       dH3=S[site]*( S[site+L*SUPER_BLOCK_SIZE]*S[site+1]
                    +S[site+1]*S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
                    +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]*S[site-1+L]
                    +S[site-1+L]*S[site+L*SUPER_BLOCK_SIZE]
            +S[site+L*SUPER_BLOCK_SIZE-1+L]*S[site-1+L]
                    +S[site-1+L]*S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1+L]
                    +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1+L]
                    *S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
                    +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
                    *S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]
                    +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]
                    *S[site+1]+S[site+1]*S[site+L*SUPER_BLOCK_SIZE+1]
                    +S[site+L*SUPER_BLOCK_SIZE+1]*S[site+L*SUPER_BLOCK_SIZE]
                    +S[site+L*SUPER_BLOCK_SIZE]*S[site+L*SUPER_BLOCK_SIZE-1+L]);
    }
    else {
   dH1=S[site]*( S[site+1]+S[site-1]+S[site+L*SUPER_BLOCK_SIZE]
          +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]);
   dH2=S[site]*( S[site+L*SUPER_BLOCK_SIZE+1]+S[site+L*SUPER_BLOCK_SIZE-1]
          +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]
          +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1]);
       dH3=S[site]*( S[site+L*SUPER_BLOCK_SIZE]*S[site+1]
          +S[site+1]*S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
          +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
          *S[site-1]+S[site-1]*S[site+L*SUPER_BLOCK_SIZE]
  +S[site+L*SUPER_BLOCK_SIZE-1]*S[site-1]+S[site-1]
          *S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1]
  +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE-1]
          *S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
          +S[site-L*SUPER_BLOCK_SIZE+L*L*SUPER_BLOCK_SIZE]
          *S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]
  +S[site+L*L*SUPER_BLOCK_SIZE-L*SUPER_BLOCK_SIZE+1]
          *S[site+1]+S[site+1]*S[site+L*SUPER_BLOCK_SIZE+1]
  +S[site+L*SUPER_BLOCK_SIZE+1]*S[site+L*SUPER_BLOCK_SIZE]
          +S[site+L*SUPER_BLOCK_SIZE]*S[site+L*SUPER_BLOCK_SIZE-1]);
  }
```

```
}
    else {
   if(threadIdx.x==0) { //Left
       dH1=S[site]*( S[site-1+L]+S[site+1]
                        +S[site+L*SUPER_BLOCK_SIZE]+S[site-L*SUPER_BLOCK_SIZE]);
   dH2=S[site]*( S[site+L*SUPER_BLOCK_SIZE+1]+S[site+L*SUPER_BLOCK_SIZE-1+L]
                        +S[site-L*SUPER_BLOCK_SIZE+1]+S[site-L*SUPER_BLOCK_SIZE-1+L]);
   dH3=S[site]*( S[site+L*SUPER_BLOCK_SIZE]*S[site+1]
                        +S[site+1]*S[site-L*SUPER_BLOCK_SIZE]
                        +S[site-L*SUPER_BLOCK_SIZE]*S[site-1+L]
                        +S[site-1+L]*S[site+L*SUPER_BLOCK_SIZE]
        +S[site+L*SUPER_BLOCK_SIZE-1+L]
                        *S[site-1+L]+S[site-1+L]*S[site-L*SUPER_BLOCK_SIZE-1+L]
        +S[site-L*SUPER_BLOCK_SIZE-1+L]*S[site-L*SUPER_BLOCK_SIZE]
                        +S[site-L*SUPER_BLOCK_SIZE]*S[site-L*SUPER_BLOCK_SIZE+1]
        +S[site-L*SUPER_BLOCK_SIZE+1]*S[site+1]
                        +S[site+1]*S[site+L*SUPER_BLOCK_SIZE+1]
        +S[site+L*SUPER_BLOCK_SIZE+1]*S[site+L*SUPER_BLOCK_SIZE]
                        +S[site+L*SUPER_BLOCK_SIZE]*S[site+L*SUPER_BLOCK_SIZE-1+L]);

    }
    else {
  dH1=S[site]*( S[site+1]+S[site-1]
            +S[site+L*SUPER_BLOCK_SIZE]+S[site-L*SUPER_BLOCK_SIZE]);
  dH2=S[site]*( S[site+L*SUPER_BLOCK_SIZE+1]+S[site+L*SUPER_BLOCK_SIZE-1]
            +S[site-L*SUPER_BLOCK_SIZE+1]+S[site-L*SUPER_BLOCK_SIZE-1]);
  dH3=S[site]*( S[site+L*SUPER_BLOCK_SIZE]*S[site+1]
                       +S[site+1]*S[site-L*SUPER_BLOCK_SIZE]
                       +S[site-L*SUPER_BLOCK_SIZE]*S[site-1]
                       +S[site-1]*S[site+L*SUPER_BLOCK_SIZE]
      +S[site+L*SUPER_BLOCK_SIZE-1]*S[site-1]+S[site-1]
                       *S[site-L*SUPER_BLOCK_SIZE-1]
      +S[site-L*SUPER_BLOCK_SIZE-1]*S[site-L*SUPER_BLOCK_SIZE]
                       +S[site-L*SUPER_BLOCK_SIZE]*S[site-L*SUPER_BLOCK_SIZE+1]
      +S[site-L*SUPER_BLOCK_SIZE+1]*S[site+1]+S[site+1]
                       *S[site+L*SUPER_BLOCK_SIZE+1]
      +S[site+L*SUPER_BLOCK_SIZE+1]*S[site+L*SUPER_BLOCK_SIZE]
                       +S[site+L*SUPER_BLOCK_SIZE]*S[site+L*SUPER_BLOCK_SIZE-1]);
  }
}

ae=K1*dH1+K2*dH2+K3*dH3+B*S[site];
if(ae>0){
S[site]*=-1;
//deltaE=-2*ae;
}
else{
```

```
if(IZ*4.656612e-10< d_pt[(dH1/2+2)*5*7*2*SUPER_BLOCK_SIZE
                                    +(dH2/2+2)*7*2*SUPER_BLOCK_SIZE
                                    +(dH3/4+3)*2*SUPER_BLOCK_SIZE
                                    +(S[site]+1)/2*SUPER_BLOCK_SIZE+blockIdx.x]){
S[site]*=-1;
//deltaE=-2*ae;
}
}

//__syncthreads();

break;

case(2): case(3): case(4):// similar to case(1);

default:          //error handling;
break;
  }

  //Transfer random data back to global memory
  R[threadIdx.x+THREAD_SIZE*blockIdx.y+THREAD_SIZE*BLOCK_SIZE*blockIdx.x]= r[threadIdx.x];
  R[threadIdx.x+THREAD_SIZE*blockIdx.y
   +THREAD_SIZE*BLOCK_SIZE*blockIdx.x
   +THREAD_SIZE*BLOCK_SIZE*SUPER_BLOCK_SIZE]= r[threadIdx.x+THREAD_SIZE];

}
```

Appendix B

parallel Wang-Landau method

```
// device function for trial move kernel function
// define PROFE to use non-uniform modification factor
// where (PROF_Eu, PROF_El) specify step function energy range
// and PROF_factor specify the magnitude;
// d_reduce takes in account that initially all random walkers
// take the similar path, so initial update to density of state
// should reduce by number of threads;
__device__ bool d_WangLandau(float Ei, float Ef, unsigned* d_wlH, double* d_wllng,
                             double* d_wllngc, double* d_wllngca, struct Water* Wmol,
                             struct Water* d_conf, int *d_Iconf, float rng)
{
// Ei = initial energy; Ef = final energy
// d_wlH = energy histogram; d_wllng = density of state of each thread
// d_wllngc = base density of state of each gpu
// d_wllngca = overall base density of state
// Wmol = water clusters array;
// d_conf = output configuration array of water clusters
// d_Iconf = specify which energy level to output configuration
// rng = random number
int iti;//indices of initial config
int fti;//indices of final config
float lngi,lngf1, lngf2;  //values for the density of states, intial and final
float R;
#ifdef PROFE
float profei, profef;
#endif
int site = threadIdx.x+blockIdx.x*(blockDim.x);
int disg = THREAD_SIZE*BLOCK_SIZE;

R = Ef;
if(R != R) return false; // in case Ef = NAN;

//Primary direction index
iti=(int) ((Ei/d_N-d_WLD1min)*d_invdWLD1);
fti=(int) ((Ef/d_N-d_WLD1min)*d_invdWLD1);
#ifdef PROFE
profei = (Ei/d_N < PROF_Eu && Ei/d_N > PROF_El)?PROF_factor:1.0;
```

```
profef = (Ef/d_N < PROF_Eu && Ef/d_N > PROF_El)?PROF_factor:1.0;
#endif

/* if(iti <0 || iti >=d_D1BINS){
d_flat = iti;
return false;
}*/

if((fti<0)||(fti>=d_D1BINS))
    {
//reject, outside of bounds
d_wlH[site+iti*disg]+=1;
#ifdef PROFE
d_wllng[site+iti*disg]+=  d_lnwlf*d_reduce/profei;
#else
d_wllng[site+iti*disg]+=  d_lnwlf*d_reduce;
#endif

return false;

    }
else
    {
lngi = d_wllng[site+iti*disg] - d_wllng[site+fti*disg];
lngf1 = d_wllngc[iti] - d_wllngc[fti];
lngf2 = d_wllngca[iti] - d_wllngca[fti];

R=__expf(lngi+lngf1+lngf2);

if(rng<R)
{
//accept

  d_wlH[site+fti*disg]+=1;
#ifdef PROFE
  d_wllng[site+fti*disg]+= d_lnwlf*d_reduce/profef;
#else
  d_wllng[site+fti*disg]+= d_lnwlf*d_reduce;
#endif

if(fti < NCONF && d_Iconf[fti] == 0){
if(atomicExch(&d_Iconf[fti], 1) == 0){   //data lock;
for(int i=0; i<d_N;i++){
d_conf[fti*d_N+i] = Wmol[site*d_N+i];
}
}
}
```

```
return true;
}
else
{
//reject
  d_wlH[site+iti*disg]+=1;
#ifdef PROFE
  d_wllng[site+iti*disg]+= d_lnwlf*d_reduce/profei;
#else
  d_wllng[site+iti*disg]+= d_lnwlf*d_reduce;
#endif
  return false;
};
    };

}
```