Finite-Size Lattice Effects in Potts Model

Sasha Soane and Conner Galloway*
MIT Department of Physics

(Dated: May 15, 2008)

We explore finite-size effects in the context of a 2-dimensional Potts model on a square lattice. We begin with a complete description of a Matlab code that effectively simulates a Potts model and discuss how to implement a finite-size experiment. Using data collected by our algorithm, we probe the effects of varying the lattice size. Based on theory presented in Ferrenberg and Landau [1], finite-size effects are used to extract the phase transition value K_c for a $q = 3$ Potts model.

1. THEORY AND MOTIVATION

Magnetization on a lattice has been the traditional sandbox for statistical physics. The conditions imposed by having a lattice structure work well with the heuristic understanding of how magnetization behaves in a solid. Furthermore, the lattice framework has the mathematical advantage of being able to graft physical theories to a workable (computationally feasible) environment. The Potts model is one example of a theory that describes how magnetization arises from spin interactions.

The principle element that distinguishes models from one another is the formulation of the Hamiltonian. In the case of the Potts model, the Hamiltonian takes the form:

$$\beta H = -K \sum_{\langle ij \rangle} \delta_{\sigma_i, \sigma_j} \tag{1}$$

where the symbol $\langle ij \rangle$ is the usual notation for a nearest-neighbor sum. The delta function insures that only when neighboring spins, defined by σ, are equal will the energy be lowered. The sign of the K coefficient determines ferromagnetism versus anti-ferromagnetism (at least in the reduced Ising case).

In general, spins in a Potts model can take any of q different values, running from 1 through q. $q = 2$ is mapped to Ising. For higher values of q we begin to notice interesting effects. For a 2-dimensional lattice, theoretical analysis shows that when $q < 4$ we have a second-order phase transition, which means that the second derivative of the free energy with respect to temperature, $\frac{d^2 f}{dT^2}$, is discontinuous when the model passes the critical temperature T_c. In contrast, values of $q > 4$ exhibit a first-order transition, in which case it is $\frac{df}{dT}$ that is discontinuous at T_c.

In principle, it is uncomplicated to detect the difference between a first-order and a second-order transition by plotting the free energy as a function of temperature. For a truly infinite size lattice and infinite time simulation, the numerical results should agree with theory. In practice, however, the distinguishing features of each transition become "blurred". In the Potts $q < 4$ case, this blurriness is a consequence of the fact that simulations involve *finite* lattices and *finite* time simulation runs. In contrast, Lee and Kosterlitz claim that for a five-state Potts model, the values of L needed to see the finite-size effects are prohibitively large [5].

In our work, we follow the steps outlined by Ferrenberg and Landau [1] to examine the effects of finite-size scaling in a Potts model for $q = 3$. Although a finite-sized lattice is inconsistent with the infinite-lattice theory, we can still use the lattice size as a new variable L and observe how the system changes with L. We take Ferrenberg and Landau's scaling ansatz as our starting point:

$$F(L, T, h) = L^{-(2-\alpha)/\nu} \mathcal{F}(tL^{1/\nu}, hL^{(\gamma+\beta)/\nu}), \tag{2}$$

where $t = (T - T_c)/T_c$. Since we include no external field in our model, the only relevant scaling parameter is $x_t = tL^{1/\nu}$. As usual, thermodynamic derivatives have a peak where the scaling function is at a maximum. The location of this peak defines the effective transition temperature $T_c(L)$. In the infinite-lattice case, $T_c(L) \to T_c$ as $L \to \infty$. However, we no longer have this condition. The effective transition temperature may be expanded as a series with higher order corrections. Thus, to leading order our effective temperature becomes

$$T_c(L) = T_c + \lambda_T L^{-1/\nu} \tag{3}$$

and is equivalently

$$K_c(L) = K_c + \lambda_K L^{-1/\nu}. \tag{4}$$

Ferrenberg and Landau pursuit further this idea by including a correction term $L^{-\omega}$ as

$$K_c(L) = K_c + \lambda_K L^{-1/\nu}(1 + b_K L^{-\omega}), \tag{5}$$

but we will not adopt this higher-order correction in our analysis.

In general, thermodynamic derivatives on a finite-sized lattice share the same basic scaling as Eqs. (3) and (4); however, it is necessary to include additional fitting parameters such as proportionality constants in these equations. Including additional fitting parameters becomes a

*Electronic address: `sasha08@mit.edu,cg1729@mit.edu` problem when running Monte Carlo simulations, as often the time necessary to wait for appropriately good data is long. Landau and Binder [4] present a unique thermodynamic quantity, the fourth-order magnetization cumulant, that conveniently circumvents this parameter-fitting problem by the fact that the *maximum* value of its derivative with respect to K scales with $\sim aL^{1/\nu}$. A scaling relation $\sim aL^{1/\nu}$ is useful because it provides a means of directly measuring ν. For an observable $\mathcal{O}$ with such a scaling relation, we can take the natural logarithm:

$$\mathcal{O} = aL^{1/\nu} \tag{6}$$
$$\log(\mathcal{O}) = \log(a) + \frac{1}{\nu}\log(L).$$

It is clear that a plot of $\log(\mathcal{O})$ as a function of $\log(L)$ will allow us to extract ν as the slope. Furthermore, Ferrenberg and Landau show that the maximum value of the logarithmic derivative of any quantity such as $\langle m^n \rangle$ scales as in Eq. (6). In our analysis, we use $\langle m \rangle$ and $\langle m^2 \rangle$.

Once ν has been measured, we may use its value in Eq. (4). $K_c(L)$ is linear as a function of $L^{-1/\nu}$, which we have now determined. In the $L \to \infty$ limit, $L^{-1/\nu} \to 0$. Thus, regardless of specific slope constants λ, we recover the exact value for K_c when $L^{-1/\nu} \to \infty$. In essence, we have used the scaling ansatz to determine a value for the infinite-lattice phase transition point.

2. MATLAB SIMULATION

2.1. Key concepts

If we allow ourselves the restriction of a 2-dimensional square lattice, the problem of simulation becomes tractable. Matlab is an ideal candidate for a simulating program because a square lattice has a natural representation as a matrix. In effect, our world is reduced to values on a $L \times L$ *spin matrix.* Given the spin matrix, we can construct a recipe for implementing the Potts model:

1. Initialize state

2. Determine spin flipping

3. Calculate relevant physical parameters

4. Update spin matrix

The first step involves creating an initial state (either random or specified) on a $L \times L$ matrix representing the system. The system is changed by a Monte Carlo process according to some algorithm that respects the condition of *detailed balance* [2, 3]. Detailed balance dictates that the ratio of the probability of any two sampled states is equal to the ratio of their Boltzmann weights. Physical parameters are calculated from the new lattice state, and the process is repeated for a set number of times.

2.2. Spin flip determination

The second item on the list comprises the most challenging aspect of the simulation. How do we efficiently compute the "flipping matrix" that evolves the state towards equilibrium? The Metropolis algorithm is the standard technique used to evaluate this process. Using the Hamiltonian in Eq. (1), we know the Boltzmann weight of every spin. For a q state Potts model on a square lattice, a spin can have anywhere from 0 to 4 similarly-valued nearest neighbors. The simplest implementation is to consider one spin. The change in energy of the system if the spin is randomly flipped to another q value is first calculated. The spin is then flipped if

$$r < e^{-\frac{K\Delta E}{T}}, \tag{7}$$

where r is a random number $0 < r < 1$. [4]. This process, once applied to every spin in the lattice, is considered one whole lattice sweep. In our language, we refer to a whole lattice sweep as a "time step". Although this incremental method clearly works, it is time consuming due to inefficiency. Of several possible alternatives, such as the Cluster Flipping Method and the Swendson-Wang algorithm, we chose to use a sublattice matrix approach that is simple yet couples neighboring spins.

2.3. Checkerboard algorithm

Our first matrix method was to utilize the **circshift** Matlab function. This function takes an input matrix and reproduces a duplicate matrix but with indices shifted by a chosen amount. By combining four **circshift** commands, each of a single index shift, we could look at the four neighboring spins (periodic boundary conditions) of every spin simultaneously. Adding logic "delta functions", we constructed a $L \times L$ energy matrix that gave an energy value to each spin based on the Hamiltonian in Eq. (1). Flipping, determined by the Metropolis algorithm, occured simultaneously for all spins, which concluded a time step. This method, although computationally much more efficient than a spin-by-spin flipping, nonetheless suffered from a tendancy to oscillate. If we consider a restricted system of only two spins, oscillations become clear. Assume that the spins are anti-aligned. Each spin will want to flip in order to minimize energy. Since our algorithm performs spin flips simultaneously, the result of a time step is that the spins are still anti-aligned (but are now opposite to before).

Oscillations, which are metastable states, are physically unrealistic. In terms of collecting data, such oscillatory states ruin values for magnetization (although the energy is actually the same) because the values fluctuate wildly. What is needed is a simulation method that does not succumb to oscillations. Clearly, simultaneous flipping causes rather than fixes the problem. In order to address this issue, we decided to pursue a "checkerboard" strategy, as discussed in Landau and Binder [4]. Motivation for the checkerboard method was provided by the fact that nearest neighbor interactions only take place along the x and y axis of the square lattice. Diagonal interactions are not present. Thus, it is possible to decompose the square lattice into two sublattices, which schematically look like the black and white tiles of a checkerboard. After the energy matrix is computed, the checkerboard method only flips *one* of the two sublattices before recomputing the spin flipping decisions for the unchanged sublattice. Thus, a true "time step" is accomplished only after two matrix manipulations (one for each sublattice).

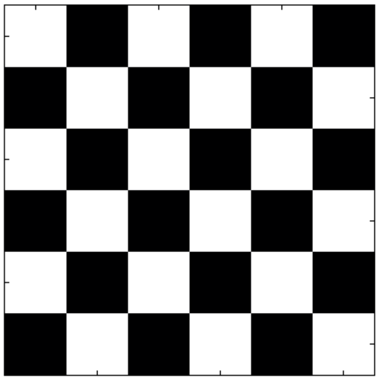

FIG. 1: Black and White tiles represent the two sublattices.

3. RAW DATA

3.1. Algorithm considerations

An important aspect of any Monte Carlo simulation for lattice systems is the correlation of states between time steps. In the previous section, we mentioned the importance of correlation between nearest neighbor spins. While it is true that nearest neighbor spins are correlated, there is a distinction between this fact and the correlation of *system states* between time steps. For an initialized random state there is a waiting period before a system can safely be called thermalized.

System correlation is a factor when taking actual data. For a truly thermalized system, an observable A should be independent of which particular time step t is chosen as a "snapshot" of the system. Because our algorithm evolves the system towards equilibrium one step at a time, the variable t is still relevant. Therefore, it takes a certain number of time steps between measuring obervables to obtain uncorrelated data. This is especially true near the critical temperature T_c. Additionally, the system takes many more time steps to evolve to equilibrium when the temperature is nearly critical than it normally would. This effect is known as *critical relation* and is discussed in Kardar [2]. Mathematically, Landau and Binder [4] summarize this as the "autocorrelation function", given by their Eq. (4.41):

$$\Phi_A(t) = \frac{[\langle A(0)A(t)\rangle - \langle A\rangle^2]}{[\langle A^2\rangle - \langle A\rangle^2]} \tag{8}$$

where we are implicitly considering a temperature T. Difficulty in implementing the autocorrelation function to find an efficient time interval led us to use a time interval of twenty cycles (whole lattice sweeps) between data acquisitions. This is a safe number but is not the most expedient possible.

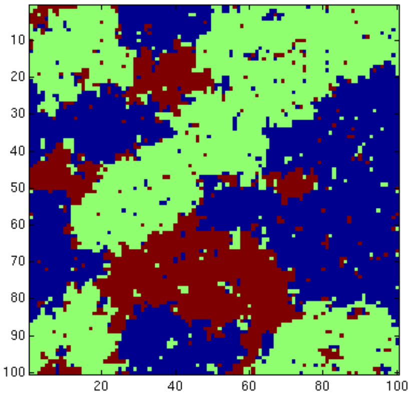

FIG. 2: Simulation at $T = 0.995$, the infinite-lattice critical temperature for $L = 100$ and $q = 3$.

3.2. Data collection

We chose to use four seperate values of L: 20, 50, 74, and 100 in our simulation runs with the coupling in Eq. (1) normalized such that $K = 1/T$. These runs were spaced through fifty one temperatures T, ranging from 0.95 to 1.05 in increments of 0.002. The Matlab code recorded several different quantities of interest, including the logarithmic derivative of $\langle m\rangle$ and $\langle m^2\rangle$, energy, heat capacity, magnetization, and susceptibility. The acquisition process broke each temperature into bins, each bin having a number of different snapshot samplings of the system (seperated by twenty lattice sweeps to keep successive measurements uncorrelated). Per temperature T, we produced two different quantity values, "grand" and "sum". The "grand" averaged every single measurement, effectively ignoring the binning process. The "sum" averaged the bin values, which were themselves averages of the stored values. In the thermodynamic limit, the "grand" and "sum" values should be equal.

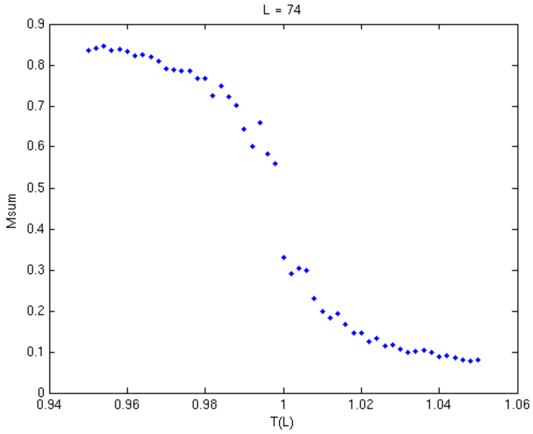

FIG. 3: Magnetization for a lattice of $L = 74$. Infinite-lattice theory states that $T_c = \log(1 + \sqrt{q})$, which for $q = 3$ is 0.9950.

3.3. Analysis

Measured quantities do not exhibit a simple mathematical behavior around the critical point. Despite the complicated behavior, we are only interested in the peak value at the effective critical point. Consequently, we decided to exclude points far away from T_c and fit the remainder to gaussians. Although the true behavior is not gaussian around T_c, reduced $\chi^2 \sim 1$ motivated adopting this approach for the general analysis.

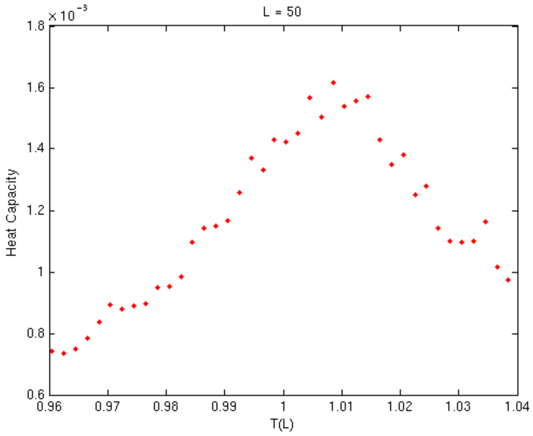

FIG. 4: Heat capacity at $L = 50$ as a function of temperature. Distribution of points may be modeled by a gaussian fit at the peak.

For $L = 50, 74, 100$ we fit the logarithmic derivatives of $\langle m \rangle$ and $\langle m^2 \rangle$ according to Eq. (6) to calculate ν.

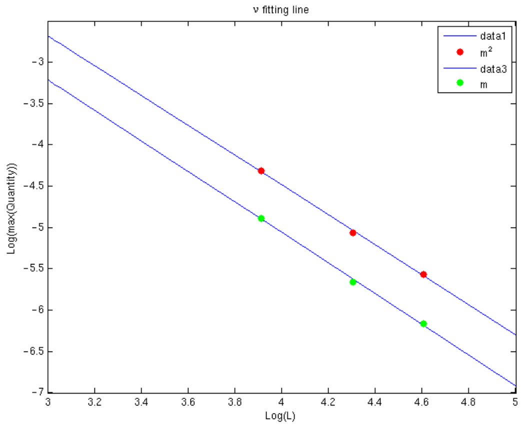

FIG. 5: Log-log plot of relevant $\langle m \rangle$ and $\langle m^2 \rangle$ quantities. Linear fit confirmed the theoretical scaling ansatz.

Based on the available data, we may tabulate the measured values for ν.

	ν
$\langle m \rangle$	1.853
$\langle m^2 \rangle$	1.807

It should be noted that the slope evident in Fig. 5 is negative. This is a descrepancy between our analysis and that of Ferrenberg and Landau [1] that we cannot explain. It is possibly due to the fact that we have a three-state Potts model instead of an Ising model.

Using an averaged value for $\nu = 1.83$, we looked at the behavior of T_c as a function of $L^{-1/\nu}$. The analysis yielded pleasantly good results for an estimated T_c.

T_c	0.995 ± 0.003

4. CONCLUSION

We successfully programmed an original simulation of the Potts model on a 2-dimensional square lattice. Guided by the work of Ferrenberg and Landau [1], we exploited finite-size scaling to extract the theoretical, infinite-lattice T_c. Although our results are within a standard deviation of the known value 0.995, there are several areas in which we could improve our analysis. First, the checkerboard approach to spin flipping contains interesting metastable states in the Ising limit. Around the phase transition, critical slowing down severely affects the efficiency of the checkerboard model. We believe that a more useful approach would be a Cluster flipping algorithm as described by Landau and Binder [4].

The simulation runs we undertook required a considerable amount of time to thermalize and produce reliable

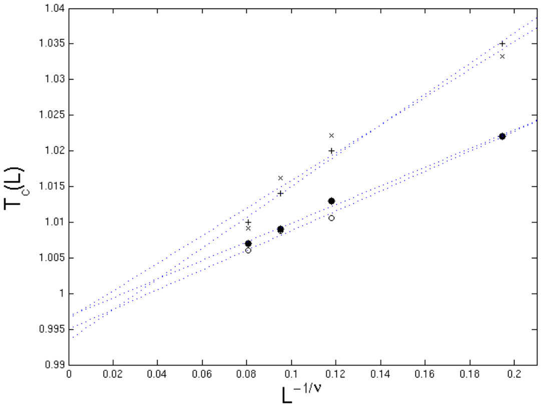

FIG. 6: Linear behavior of T_c as a function of $L^{-1/\nu}$. Plotted data include $\langle m \rangle$, $\langle m^2 \rangle$, heat capacity, and susceptibility. Extrapolated T_c at $L \to \infty$ is close to 0.995, the theoretical value.

data. Thus, we were limited in the number of total runs performed. Data analysis suffered due to a lack of data sets. Likewise, as Ferrenberg and Landau point out [1], the theoretical scaling in Eqs. (3) and (4) is only a first order approximation. With additional data sets, a more detailed analysis of Eq. (5) may be performed.

Ultimately, we demonstrated that the scaling ansatz in Eq. (2) is a viable postulate. Additionally, we showed that finite-size scaling, although visibly present in Potts models of $q < 4$ (in 2-dimensions), may be harnessed to provide interesting results.

[1] Alan M. Ferrenberg and D. P. Landau. Critical behavior of the three-dimensional ising model: A high-resolution monte carlo study. *Phys. Rev. B*, 44(10):5081–5091, Sep 1991.

[2] Mehran Kardar. *Statistical Physics of Fields*. Cambridge University Press, Cambridge, first edition, 2007.

[3] Steven Koonin and Dawn Meredith. *Computational Physics*. Westview Press, New York, 1990.

[4] David Landau and Kurt Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, Cambridge, second edition, 2000.

[5] Jooyoung Lee and J. M. Kosterlitz. Finite-size scaling and monte carlo simulations of first-order phase transitions. *Phys. Rev. B*, 43(4):3265–3277, Feb 1991.