Ising Model – Numerical Demonstrations

Yikai Yin, Seunghwa Ryu and Wei Cai

Stanford University

(Last modified: 2016/9/12)

1. Download and Compile MD++

Direct your browser to http://micro.stanford.edu/MDpp , click **Download** for MD++ source file. Download the latest MD++ source file, e.g. md++-2016-06-23.tar.gz, to your home directory. We recommend that you make a sub-directory **Codes** in your home directory, then unzip the source file here.

On your Unix/Linux workstation (we use the Terminal on a Macbook here), do

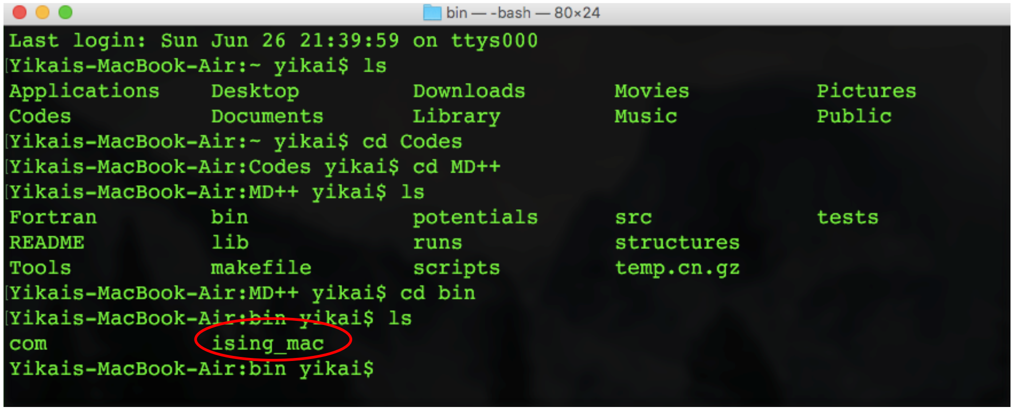
```
$ cd ~
$ mkdir Codes
$ cd Codes
$ tar zxvf ~/ md++-2016-06-23.tar.gz
```

MD++ will be unzipped in the **~/Codes/MD++** directory.

To compile the code, do

```
$ cd MD++
$ make ising build=R SYS=mac
```

This will create the executable **ising_mac** in the **~/Codes/MD++/bin** directory as shown below:



```
Last login: Sun Jun 26 21:39:59 on ttys000
Yikais-MacBook-Air:~ yikai$ ls
Applications    Desktop         Downloads       Movies          Pictures
Codes           Documents       Library         Music           Public
Yikais-MacBook-Air:~ yikai$ cd Codes
Yikais-MacBook-Air:Codes yikai$ cd MD++
Yikais-MacBook-Air:MD++ yikai$ ls
Fortran         bin             potentials      src             tests
README          lib             runs            structures
Tools           makefile        scripts         temp.cn.gz
Yikais-MacBook-Air:MD++ yikai$ cd bin
Yikais-MacBook-Air:bin yikai$ ls
com             ising_mac
Yikais-MacBook-Air:bin yikai$
```

If you are using other systems, set **SYS = gpp** or **SYS = intel** on the previous command.

More information about how to use MD++ can be found at:
http://micro.stanford.edu/wiki/MD%2B%2B_Manuals

In order to visualize the results of MD++, you will need the X-windows application. For OS X on Mac, you can download XQuartz from https://www.xquartz.org/ , for details on how to apply X-windows, please refer to https://uit.stanford.edu/service/sharedcomputing/moreX .

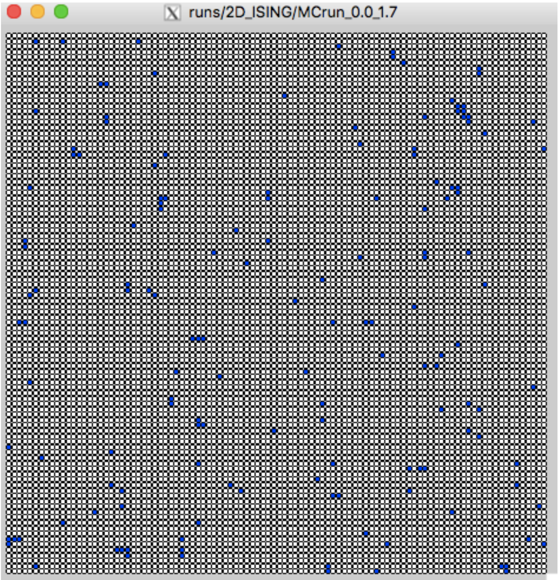
2. **Monte Carlo (MC) Simulation of the Ising Model**

2.1. **Run MC simulation**

Execute the following commands.

```
$ cd ~/Codes/MD++
$ bin/ising_mac scripts/LLNL2010/Day1_ising_Mcrun.tcl 0.0 1.7 10000
```

The first argument (`0.0`) specifies **h**, the magnetic field. The second argument (`1.7`) specifies **kBT**, the temperature. The third argument (`10000`) specifies **totalsteps**, the simulation time in unit of Monte Carlo steps per site (MCSS).



The simulation will open a window (see above) displaying the flipping of the spins during the MC simulation. White squares are -1 spins and blue squares are +1 spins. In this example, the Ising model is a 100 x 100 square lattice with J = 1. At each MC step, the algorithm choose a spin randomly and attempts to flip it.

2.2. **Process MC simulation results**

At the end of this simulation, the following lines will be printed on the terminal.

```
Magnetization from Monte Carlo Simulation =-0.9702 +- 0.00339263909074
Magnetization from Analytic Onsager Solution=0.9700778116023373
Average Largest Cluster Size = 7
```

The first line is the average magnetization (i.e. average value of the spins). We can see that most of the spins are pointing downward (-1) in this simulation. The second line prints the analytic solution predicted by Onsager. The third line prints the average value of the largest cluster (islands of +1 spins) during the simulation.

More data are saved in the **`~/Codes/MD++/runs/2D_Ising/MCrun_0.0_1.7`** directory. The **`prop.out`** file contains statistical information taken periodically during the simulation. To see the content of this file, do

```
$ tail -3 runs/2D_ISING/MCrun_0.00_1.7/prop.out
9997 -9718 141 10000 4 1
9998 -9722 139 10000 4 1
9999 -9720 140 10000 5 1
```

There are six columns in the **prop.out** file. Their meaning is explained below.
Column 1 : time in MCSS
Column 2 : **`Stot`** = **`Nup`** – **`Ndown`** (Magnetization is defined as Stot/Ntot)
Column 3 : **`Nup,`** number of up (+1) spins
Column 4 : **`Ntot`** = **`Nup`** + **`Ndown,`** total number of spins
Column 5 : **`Nc`**, size of largest cluster
Column 6 : for debug purposes. You can ignore this.

3. **Prepare the initial conditions for US simulations**

First, execute the following commands.

```
$ cd ~/Codes/MD++
$ bin/ising_mac scripts/LLNL2010/Day1_ising_FFS1cpu.tcl 0 4 0.06 1.7
```

Next, execute the following command.

```
$ bin/ising_mac scripts/LLNL2010/Day1_ising_FFS1cpu.tcl 5 100 0.06 1.7
```

The spin configurations saved at each step can be found in directory
`~/Codes/MD++/runs/2D_Ising/FFS_0.06_1.7/run_[n]` .

4. Umbrella Sampling (US) Simulation of the Ising Model

4.1. Run US simulation

The US simulations can be started only after preparing the initial conditions for US simulations. The syntax for US simulation command is the following.

```
$ bin/ising_mac scripts/LLNL2010/Day1_ising_US.tcl N n_c h_w h kBT
```

where
`N` : the index number of the umbrella sampling window

`n_c` : the center of the bias potential
`h_w` : the half width of the histogram
`h` : magnetic field
`kBT` : temperature

As an example, execute the following commands.

```
$ bin/ising_mac scripts/LLNL2010/Day1_ising_US.tcl 0 6 8 0.06 1.7
```

After the simulation finishes, folder **`~/Codes/MD++/runs/2D_Ising/FFS_0.06_1.7/UMB_0`** contains the following three data files.

`Narray.txt` : the n-axis of the histogram, where n is the size of largest nucleus
`Freq.txt` : raw data of histogram, number of times a given nucleus size is sampled
`Prob.txt` : raw histogram multiplied by exp(W/kBT) where W is the bias function

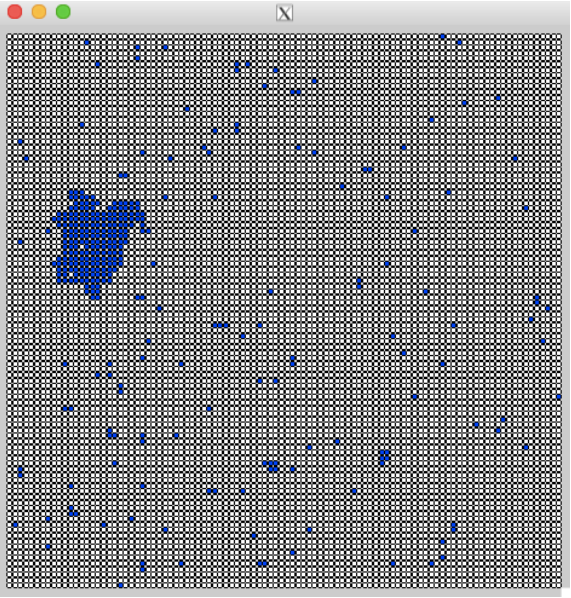
Note that the bias function is only apply when **`n_c < 0.7 λ0`**. When the bias function is not applied, **`Freq.txt`** should be identical to **`Prob.txt`**.

5. Visualization of Saved Spin Configurations

The spin configurations saved from MC and US simulations can be visualized using the **`Day1_ising_show.tcl`** script file, followed by the file name you wish to view. For example, execute the following command.

```
$ bin/ising_mac scripts/LLNL2010/Day1_ising_show.tcl \
                runs/2D_ISING/FFS_0.06_1.7/run_6/FFS_6_0001.cn.gz
```

You should see something like the following, in which an island (nucleus) is clearly visible.



References:

[1] S. Ryu and W. Cai, "The Validity of Classical Nucleation Theory for Ising Models", *Phys. Rev. E*, 81, 030601 (R) (2010).

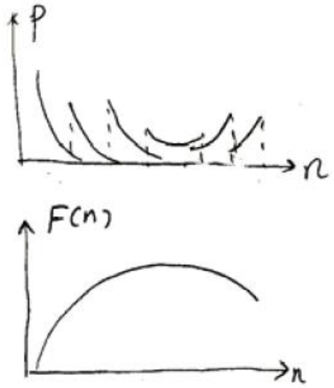
[2] S. Ryu and W. Cai, "Numerical Tests of Nucleation Theories for the Ising Models", *Physical Review E*, 82, 011603 (2010).

Appendix A. Numerical Homework

1. Run MC simulations of 2D (100x100) Ising model at h = 0 and J = 1 at several temperatures. Plot averaged magnetization as a function of temperature on top of Onsager's analytic solution $M = [\ 1 - (\sinh 2\beta)^{-4}\ ]^{1/8}$, where $\beta = 1/kBT$.

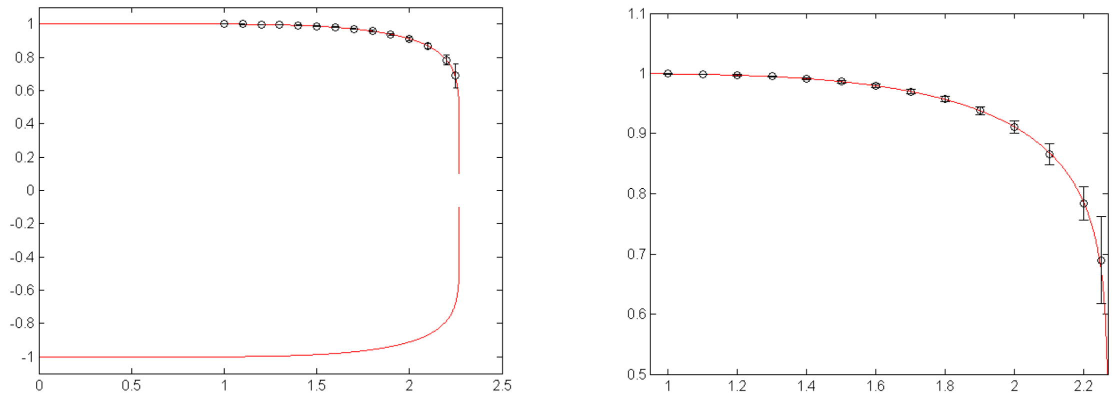
2. Run US simulations of 2D (100x100) Ising model at h = 0.06, J = 1 and kBT = 1.7, using 20 windows of half width `h_w` = 8, centered at `n_c` = 6 + 12 *i*, where *i* = 0, 1, 2, …, 19. Convert the probability data (in `Prob.txt` files) into a single continuous curve for the free energy F(n).

You will need several Matlab scripts to plot the curves, the scripts are available to download on Wei's website.



Appendix B. Homework Solutions
1. MC simulations

```
%% Matlab code for plotting the result
clear;
%% Onsager’s analytic formula
kBT1=[0.0:0.001:2.268 2.269:0.00000001:2/log(1+sqrt(2))];
M1=(1-(sinh(2./kBT1)).^(-4)).^(1/8);
%% Numerical results
kBT=[ 1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7 ...
1.8 1.9 2.0 2.1 2.2 2.25];
M =[ 0.9993 0.9985 0.9971 0.9949 0.9914 0.9866 0.9797 0.9699 ...
0.9570 0.9380 0.9111 0.8656 0.7837 0.6893];
dM =[ 0.0004 0.00058 0.00082 0.0011 0.0016 0.0020 0.0026 0.0034 ...
0.0044 0.0062 0.0099 0.0171 0.0273 0.0721];
%% Plot data
figure(1)
plot(kBT1,M1,'r-',kBT1,-M1,'r-')
hold on
errorbar(kBT,M,dM,'ko')
ylim([-1.1 +1.1])
figure(2)
plot(kBT1,M1,'r-',kBT1,-M1,'r-')
hold on
errorbar(kBT,M,dM,'ko')
xlim([0.95 max(kBT1)])
ylim([0.5 1.1])
```



Red curve: analytic formula. Black symbols: numerical data. Right: a close-up view.

2. US simulations

The overlapping histogram method (Ref. Frenkel and Smit, Understanding Molecular Simulations, Academic Press, 2002) is used to piece together the hisogram from different sampling windows.

First of all, run the simulation by using 20 windows of half width h_w = 8, centered at n_c = 6 + 12 i, where i = 0, 1, 2, …, 19. After finishing the simulation, you will get 20 folders named from UMB_0 to UMB_19 in `~/Codes/MD++/runs/2D_Ising/FFS_0.06_1.7` directory.

Suppose you copy the data folders to `Ising/UMB/FFS_0.06_1.7` in order to run Matlab, then copy the three script files which are provided beforehand

```
errorbarlogy.m
sclabs.m
UMB_draw3_for2D_single.m
```

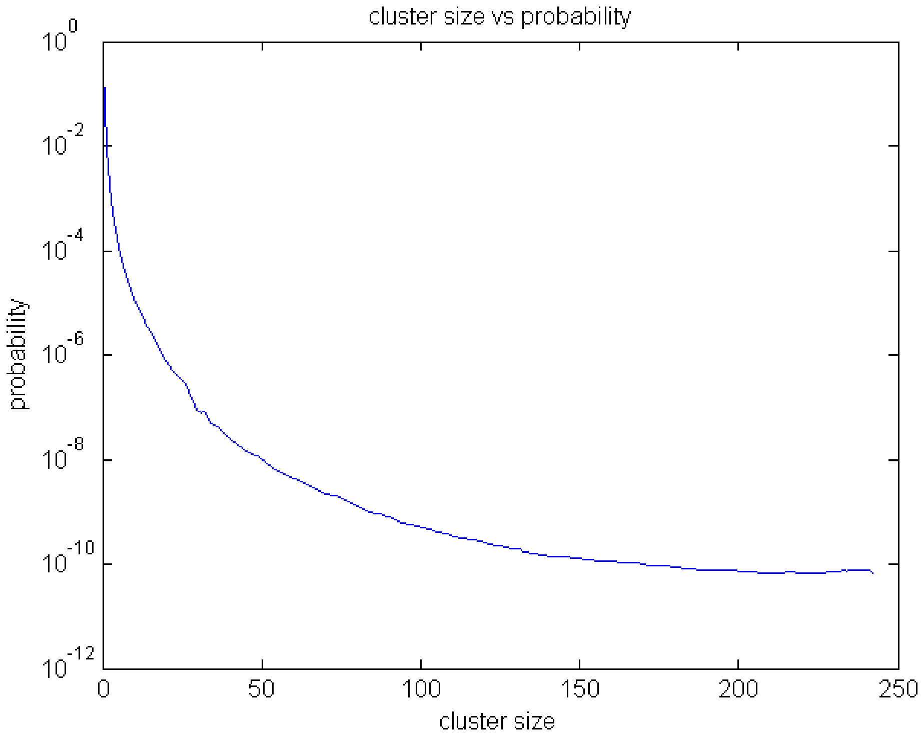
under `Ising` folder. Open `UMB_draw3_for2D_single.m` and change the first two parameters:

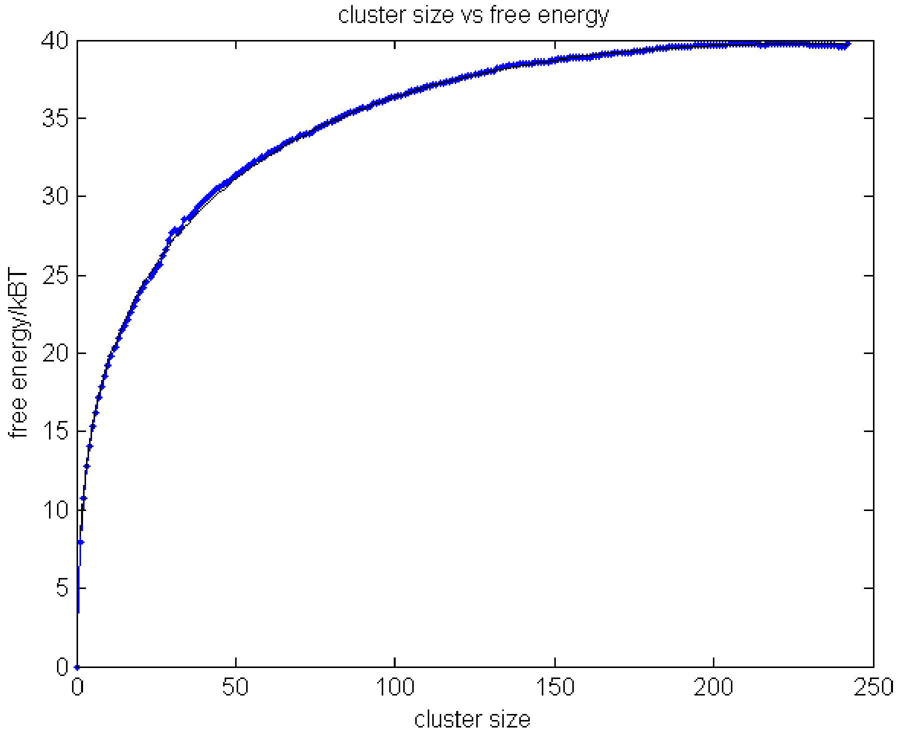
```
demoh
demokBT
```

to be the ones that correspond to the FFS_ folders. Then run the script

```
 UMB_draw3_for2D_single.m
```

it will generate the free energy curves.





Blue dotted line: US data. Black solid line: analytic predictions. The analytic expressions can be found in Ryu and Cai 2010 (see References).