

蒙特卡洛方法求解 Ising 模型 (二)

何翼成 *

May 23, 2022

Project 1

一 题目分析



二维 ($L \times L$) Ising模型的临界动力学 (5月23日交)

$$\mathcal{H} = -J \sum_{\langle ij \rangle}^N s_i s_j \quad \text{The spin } s_i \text{ for } i = 1, 2, \dots, N \text{ can take values of either } 1 \text{ or } -1$$

制备初态为一个高温的热力学平衡态, 突然将温度变到相变温度上, 研究不同尺寸下, 平均能量 $E(t) - E_0$ 在长时间下的弛豫行为, 并得到热力学极限下 (L 无穷大) 的结果

注: 1. $J=1$, E_0 为相变温度处的平均能量

2. 时间 t 的单位定义为: 在二维格点上从第一个格点起逐一翻转每个自旋并计算更新是否接受, 所有格点全部翻转一遍后, 这一过程的时间步长计为单位1

3. 取足够多的马尔可夫链做系综平均

Figure 1: 题目总览

二 代码展示

```
1 %创建方格矩阵
2 Ls=[50]; Tspan=linspace(100,100,2000); T0=2.269; num=800; Num=3e4;%计算步数
3 A=zeros(length(Ls),num);%储存不同L下的能量-时间曲线
4 for in=1:length(Ls)
```

*学号:520072910043;

邮箱地址: heyicheng@sjtu.edu.cn

```

5      L=Ls(in);
6      c=cputime;
7      %设定初态的温度
8      for tn=1:length(Tspan)
9          T=Tspan(tn);
10         Es=zeros(length(Tspan),num);
11         %得到高温的热力学平衡态
12         %确认尺寸，生成随机矩阵
13         M=randi([0,1],L,L)*2-1;
14         %逐行扫描元素，完全扫描整个矩阵为一个时间单位
15         %演化num步数
16         for tt=1:num
17             for jj=1:L %y
18                 for ii=1:L %x
19                     M_1=M;%保留磁矩矩阵的原信息
20                     M_2=M;M_2(ii,jj)=-M_1(ii,jj);%翻转后的磁矩矩阵
21                     pudM=pud(M);
22                     deltaE=2*M(ii,jj)*(pudM(ii+2,jj+1)+pudM(ii,jj+1)+pudM(ii+1,jj+2)+pudM(ii+1,jj));
23                     %判断状态是否保留
24                     flag=Metro(deltaE,T0);
25                     if flag==1
26                         M=M_2;
27                     else
28                         M=M_1;
29                     end
30                 end
31             end
32             Es(tn,tt)=H(M);%计算一次能量并且储存
33             cc=cputime;
34             disp("计算长度为"+L+",温度进度为"+tn/length(Tspan)*100+"%, 已完成进度"+tt/num*100+"%,已耗时"+num2str((c-cc)/60));
35             end
36         end
37         Eavg=sum(Es)/length(Tspan);
38         A(in,:)=Eavg;
39     end
40
41     %计算T0时的平均能量
42     E0=AverageH(L,T,Num,10);
43     plot(1:num,A-E0)
44     legend("L="+L)
45     xlabel("LogTime")
46     ylabel("Log(E-E_{0})")
47
48     %%
49     %Metropolis算法函数的定义
50     function flag=Metro(deltaE,T)
51         beta=1/T;
52         if deltaE<=0
53             p=1;

```

```

54         else
55             p=exp(-beta*deltaE);
56         end
57         z=rand();
58         if z<p
59             flag=1;
60         else
61             flag=0;
62         end
63     end
64     %%
65     %Pud, 辅助计算. 扩展为(L+2)**2的矩阵
66     function pudM=pud(M)
67     L=size(M,1);
68     pudM=zeros(L+2,L+2);%分配储存空间
69     %pudding, 采用周期性边界条件
70     pudM(2:(L+1),2:(L+1))=M;
71     %行的移动
72     pudM(1,2:(L+1))=M(L,:);
73     pudM(L+2,2:(L+1))=M(1,:);
74     %列的移动
75     pudM(2:(L+1),1)=M(:,L);
76     pudM(2:(L+1),L+2)=M(:,1);
77     end
78
79     %能量计算
80     function E=H(M)
81     L=size(M,1);
82     pudM=pud(M);
83     H=0;
84     for i=2:L
85         for j=2:L
86             H=H-pudM(i,j)*(pudM(i,j-1)+pudM(i,j+1)+pudM(i-1,j)+pudM(i+1,j));
87         end
88     end
89     E=H/2;
90     end
91
92     %%
93     %平均能量计算. 输入尺寸, 温度, 循环次数, 并行链条数, 计算该温度下的矩阵的平均能量
94     function avgH=AverageH(L,T,Num,s)
95     %创建随机矩阵
96     Matrix=randi([0,1],L,L)*2-1;
97     %创建空数组记录能量值
98     Hels=zeros(s,Num);
99     for ss=1:s
100         for mm=1:Num
101             locs=randi([1,L],1,2);%创建随机坐标
102             x=locs(1);y=locs(2);

```

```

103     Matrix_1=Matrix;%保留磁矩矩阵的原信息
104     Matrix_2=Matrix;Matrix_2(x,y)=-Matrix_1(x,y);%翻转后的磁矩矩阵
105     pudMatrix=pud(Matrix);
106     deltaE=2*Matrix(x,y)*(pudMatrix(x+2,y+1)+pudMatrix(x,y+1)+pudMatrix(x+1,y+2)+pudMatrix(x+1,y));
107     %判断状态是否保留
108     flag=Metro(deltaE,T);
109     if flag==1
110         Matrix=Matrix_2;
111     else
112         Matrix=Matrix_1;
113     end
114     Hels(ss,Num)=H(Matrix);
115 end
116 end
117 avgH=sum(Hels,"all")/(Num*s);
118 end

```

三 结果分析与结论

3.1 原图像

3.2 取对数轴进行逼近

由于笔记本 CPU 性能问题, 在计算过程中即使采取了多种近似和精简方法, 在 L 增大至 50 左右时, 所需要的时间就开始急速增长. 所以本文中分析过程也会局限于 $L < 50$ 的范围.

根据图像分析可知, 随着 L 取值的增大, 其图线的意义逐渐趋向于热力学极限. 从原始图像来看, 就是原本高温的平衡态逐渐平衡在了 $T_0 = 2.269K$. 为了方便分析, 我们对代表时间意义的 Time 轴和平均能量均取对数, 从而得到了一个弯曲的曲线. 我们可以合理推论: 当 L 趋向于无穷大时, 其图线将会趋于一条直线.

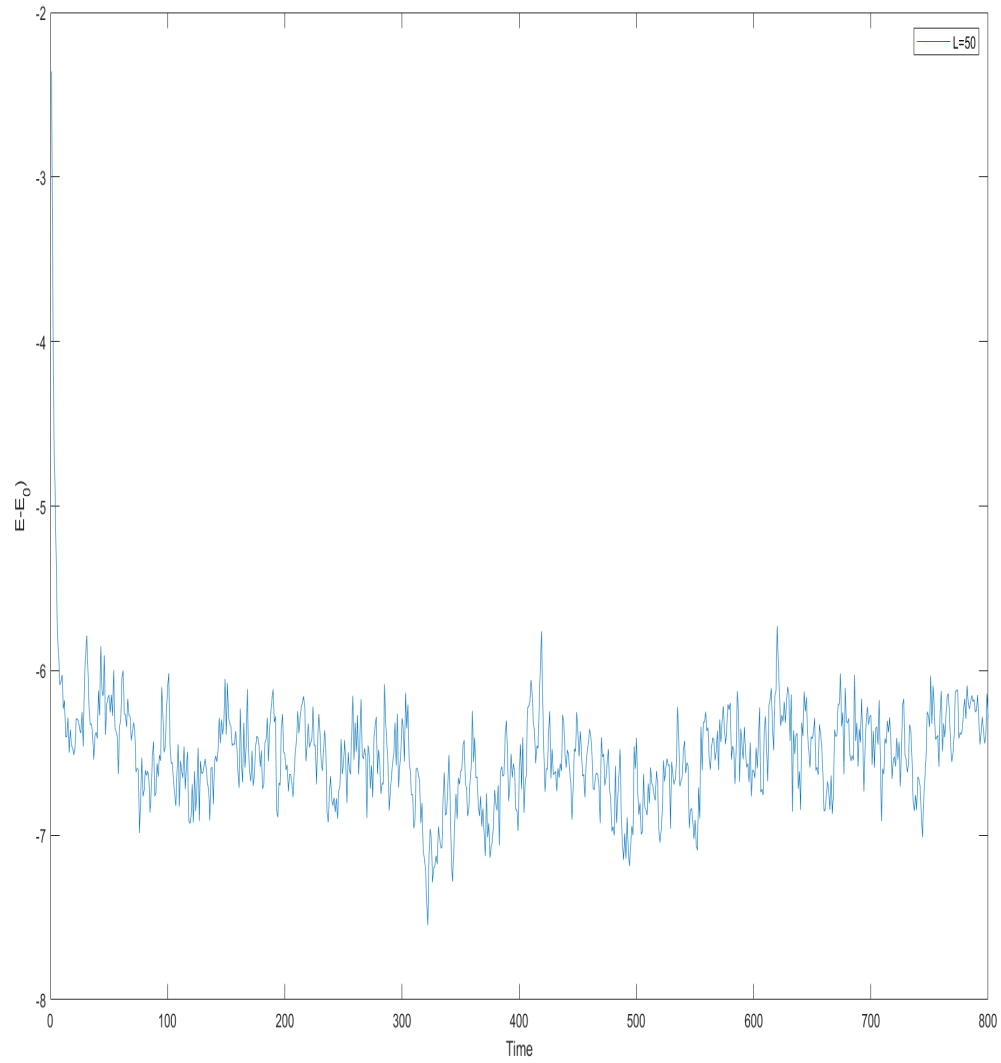


Figure 2: $\langle H \rangle$ -Time with $L=50$

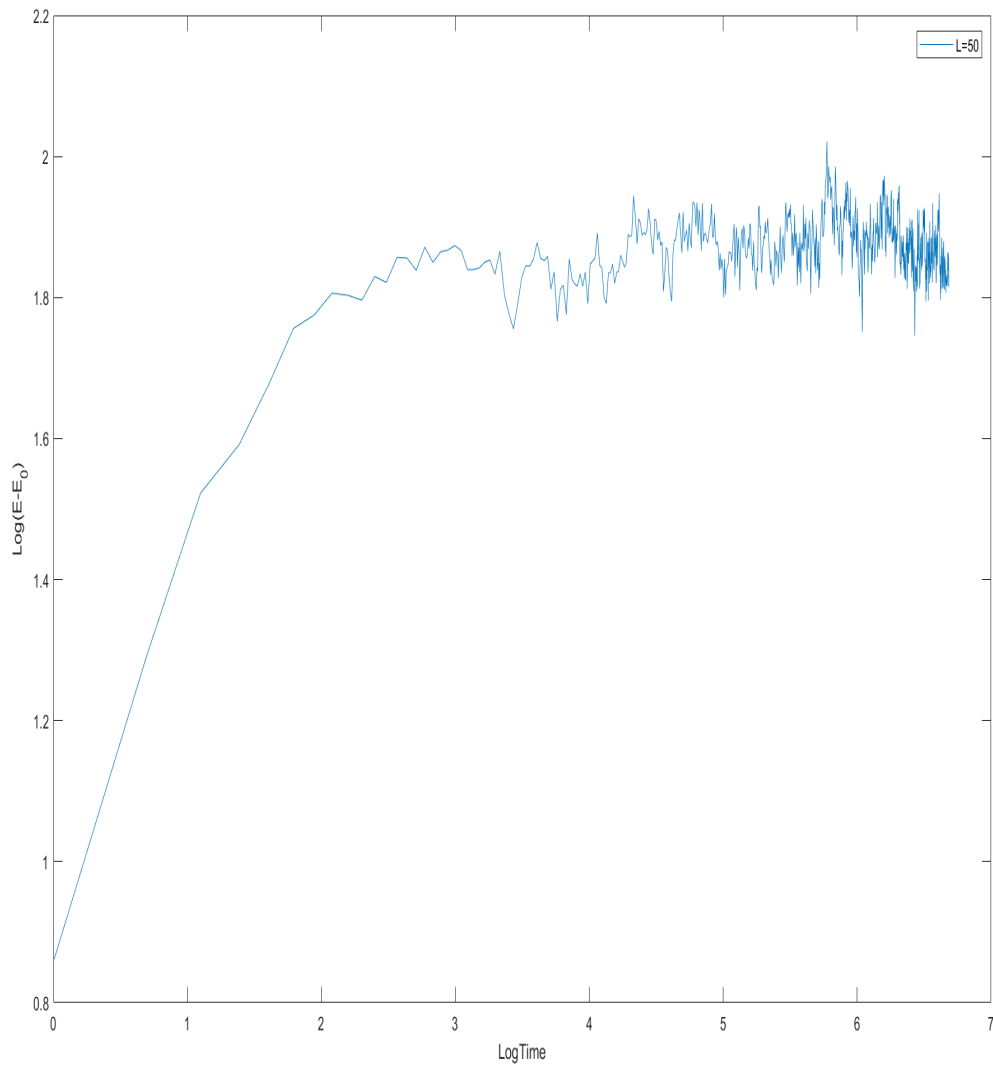


Figure 3: $\log\langle H \rangle - \log\text{Time}$