

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/265051069>

# Estudo do Modelo de Ising Bidimensional Utilizando o Algoritmo de Metropolis

Technical Report · July 2012

DOI: 10.13140/RG.2.1.2354.1840

CITATION

1

READS

552

1 author:



Felipe Leite Antunes

Universidade Federal do Rio Grande do Sul

7 PUBLICATIONS 4 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Data Science Applications in the Government Sector [View project](#)



UNIVERSIDADE DO RIO GRANDE DO SUL

TRABALHO FINAL DE  
MÉTODOS COMPUTACIONAIS DA FÍSICA B

---

# Estudo do Modelo de Ising Bidimensional Utilizando o Algoritmo de Metropolis

---

*Autor:*  
Felipe ANTUNES

*Professor:*  
Dr. Sebastián GONÇALVES

2 de Julho de 2012

# Conteúdo

<b>1</b>	<b>Motivação e Procedimento Computacional</b>	<b>3</b>
1.1	O modelo de Ising . . . . .	3
1.2	Procedimento computacional . . . . .	3
1.3	Médias . . . . .	4
1.4	Algoritmo de Metropolis . . . . .	4
1.5	Observáveis . . . . .	5
<b>2</b>	<b>Transições de Fase de Segunda Ordem e Resultados</b>	<b>7</b>
2.1	Magnetização . . . . .	7
2.2	Energia . . . . .	8
2.3	Calor Específico . . . . .	9
2.4	Conclusão . . . . .	9
<b>A</b>	<b>Programa</b>	<b>10</b>
<b>B</b>	<b>Susceptibilidade Magnética</b>	<b>14</b>

# Lista de Figuras

1.1	Algoritmo Metropolis [4] (retirada de [2]. . . . .	6
2.1	Comportamento da magnetização absoluta por spin com a temperatura devido ao aumento da rede. . . . .	7
2.2	Comportamento da energia por spin com a temperatura devido ao aumento da rede. Extrapolando, percebe-se que a inclinação próximo a $T = 2.27$ diverge. . . . .	8
2.3	Comportamento do calor específico com a temperatura devido ao aumento da rede. Extrapolando, percebe-se a divergência em $T = 2.27$ . . . . .	9
B.1	Comportamento da susceptibilidade magnética com a temperatura devido ao aumento da rede. Extrapolando, percebe-se a divergência em $T = 2.27$ . . . . .	14

# Capítulo 1

## Motivação e Procedimento Computacional

Nossa própria experiência revela que materiais usuais, tais como o ferro, não apresentam magnetização naturalmente nas temperaturas cotidianas. A mecânica quântica nos trouxe o spin e sua relação com o momento magnético dos átomos. Podemos entender o primeiro fato tendo em vista o segundo, i.e a distribuição aleatória dos momento de dipolo dos átomos constituintes do material, resulta em uma magnetização total nula. No entanto, e.g. ferro, sob condições especiais, e.g baixas temperaturas, podem exibir uma magnetização total diferente de zero, que atribuímos à uma direção preferencial dos seus *spins*.

Dois grandes princípios estão por trás deste fenômeno: o da minimização da energia e da maximização da entropia. Estão princípios competem entre si e é a temperatura que determina qual será o dominante. A relação entre estas três quantidades é dada pela chamada distribuição de Boltzmann

$$P(\alpha) = e^{\frac{-E(\alpha)}{kT}} \quad (1.1)$$

onde  $k$  é a constante de Boltzmann, tomada igual à 1 a partir de agora.

### 1.1 O modelo de Ising

O Modelo de Ising consiste numa rede de *spins*, na qual cada um pode apontar para cima ou para baixo. Nenhuma outra orientação é permitida. Portanto o  $i$ -ésimo spin no sistema pode ter apenas dois possíveis valores, que por conveniência tomamos como  $s_i = \pm 1$ . Estes *spins* interagem com os outros na rede; num ferromagneto esta interação favorece o alinhamento paralelo dos pares de spin. Numa situação real, a interação é mais forte entre os *spins* vizinhos, caindo rapidamente. Com esta motivação o Modelo de Ising mais simples assume uma interação somente entre os vizinhos mais próximos, de tal forma que o hamiltoniano do sistema é dado por:

$$H = -J \sum_{\langle ij \rangle} s_i s_j \quad (1.2)$$

onde a soma é feita sobre todos os vizinhos próximos, e  $J$  é a constante de acoplamento, que é positiva para materiais ferromagnéticos (alinhamento paralelo dos *spins*) e negativa para materiais antiferromagnéticos (favorecimento de um alinhamento anti-paralelo[1]. Neste trabalho, assumo  $J=+1$  e os valores de *spin*  $+1$  para *spin up* e  $-1$  para *spin down*.

O tratamento dados aos *spins* dos cantos da rede, é tal que eles devem interagir com os *spins* geometricamente opostos. Portanto utilizo condições periódicas de contorno, o que é equivalente topologicamente à superfície de um toro  $d+1$  onde  $d$  é a dimensão da rede, que neste trabalho será tomada como 2, pois é a rede mais simples onde fenômenos importantes surgem<sup>1</sup>[2].

### 1.2 Procedimento computacional

Vamos tentar entender como o modelo de Ising pode ser usado computacionalmente para obter observáveis. Se a energia de cada estado possível do sistema é especificada, então da equação 1.1, obtemos a probabilidade para o sistema estar em cada possível estado (numa dada temperatura), então as quantidades macroscópicas podem

---

<sup>1</sup>Condições periódicas de contorno minimizam o efeito do sistema ser finito, no entanto não eliminam tais efeitos, a melhor maneira de calcular as propriedades de um sistema muito grande, é fazendo simulações com diferentes tamanhos de rede e extrapolar estes resultados para o caso de uma rede infinita[1].

ser obtidas. Por exemplo para magnetização, que é dada pelo excesso do numero de *spins* apontando para cima ou para baixo,

$$M(\alpha) = N_{up}(\alpha) - N_{down}(\alpha) \quad (1.3)$$

e para a energia, dada pelo hamiltoniano 1.2, os valores esperados são, respectivamente,

$$\langle M \rangle = \sum_{\alpha} M(\alpha) P(\alpha), \quad (1.4)$$

e,

$$\langle E \rangle = \sum_{\alpha} E(\alpha) P(\alpha). \quad (1.5)$$

De um ponto de vista prático isto é um problema, pois com duas configurações de spin, temos  $2^N$  configurações possíveis. Com  $N$  se tornando grande, é evidente que a computação pode tornar-se extremamente longa. Uma alternativa numérica é usar uma simulação para gerar dados sobre "estados representativos", cujas frequências devem representar a situação real. Mas como atingir este objetivo?

### 1.3 Médias

No ensemble canônico a média de um observável  $A(x)$  é definida como

$$\langle A(x) \rangle = \frac{\int e^{-\beta H(x)} A(x) dx}{Z} \quad (1.6)$$

onde  $x$  é um vetor no espaço de fase e  $\beta = 1/kT$ . A função de partição,  $Z$ , é dada por

$$Z = \int e^{-\beta H(x)} dx \quad (1.7)$$

enquanto o fator de Boltzmann normalizado é

$$P(x) = \frac{e^{-\beta H(x)}}{Z} \quad (1.8)$$

Esta probabilidade dá o peso estatístico com o qual a configuração  $x$  ocorre no equilíbrio térmico. Para uma porção finita do espaço de fase temos

$$\langle A(x) \rangle = \frac{\sum_{l=1}^M e^{-\beta H(x_l)} A(x_l)}{\sum_{l=1}^M e^{-\beta H(x_l)}} \quad (1.9)$$

O problema de proceder desta forma, coletando uma parte finita do espaço de fase, é que não temos garantias de que o pico da distribuição de probabilidade levou em conta a região com os dados cruciais. Precisamos de um processo de escolha inteligente para esta região, de forma que ela contenha a informação importante. Mais precisamente, precisamos de uma distribuição de probabilidade  $P(x)$  para todo  $x$ , e selecionar as posições onde  $P(x)$  é grande. O problema agora é encontrar esta distribuição.

### 1.4 Algoritmo de Metropolis

O algoritmo de Metropolis é baseado em duas ideias. Primeiramente, dado um estado inicial a busca pelos estados mais prováveis é feita utilizando uma cadeia de Markov, i.e uma série de estados muito próximos construídos a partir de seu antecessor e a segunda é que deve haver uma função de transição  $W(x \rightarrow x')$  que satisfaça a equação de balanço detalhado

$$P(x)W(x \rightarrow x') = P(x')W(x' \rightarrow x) \quad (1.10)$$

Esta equação expressa a ideia de equilíbrio, e garante uma transição reversível  $x \leftrightarrow x'$  e é suficiente, mas não necessária, para que a cadeia de Markov seja convergente para a distribuição desejada. Pegando a razão entre as duas probabilidades de transição torna-se evidente que o movimento  $x \rightarrow x'$  e o movimento inverso  $x' \rightarrow x$  são dependentes apenas da variação da energia  $\delta H = H(x') - H(x)$

$$\frac{W(x \rightarrow x')}{W(x' \rightarrow x)} = e^{-\delta H \beta} \quad (1.11)$$

Todavia sabemos que as transições sempre devem ocorrer se o estado final é mais energeticamente favorável, portanto introduzimos:

$$W(x \rightarrow x') = \begin{cases} e^{-\delta H \beta} & \text{se } \delta H < 0, \\ 1 & \text{outro,} \end{cases} \quad (1.12)$$

Consideramos o sistema sendo ergódico<sup>2</sup>

$$\langle A(t) \rangle = \frac{1}{t_M} \int A(t) dt \quad (1.13)$$

Que se reduz a

$$\langle A(t) \rangle = \frac{1}{M} \sum_{t=1}^M A(x(t)), \quad (1.14)$$

se considerarmos o número de passos Monte Carlo como passos de tempo. Portanto a estratégia de Metropolis pode ser interpretada como tomar a média temporal ao longo de uma trajetória estocástica no espaço de fase, controlada pela equação Chapman-Kolmogorov.

Passamos a descrever o algoritmo implementado no programa desenvolvidoA, chamado Algoritmo de Metropolis [4]:

1. No primeiro passo a rede é **inicializada**, dado uma temperatura T e campo B. Depois povoa-se a rede com *spins*, com *spins* de maneira homogênea ou randômica;
2. O segundo passo é o processo de **seleção** de uma posição na rede, utilizando o gerador de números aleatórios;
3. Dada a posição, o processo de **decisão** é feito através da pergunta  $\delta H \leq 0$  ?
  - Se sim, o spin nesta localização é mudado e **armazenado** para o cálculo dos observáveis
  - Se não, é preciso **decidir** se o spin é *flipado* ou não. Um número aleatório entre 0 e 1 é gerado, se este número é menor do que  $e^{-\delta H}$ , então o spin muda, caso contrário permanece o mesmo;
4. As etapas acima, exeto a inicialização, são repetidas N vezes até o equilíbrio térmico ser atingido;
5. Após o equilíbrio, inicia-se um processo para adicionar todas essas mudanças a rede juntamente com a configuração inicial, de forma a produzir uma nova configuração.
6. Todas essas etapas estão contidas num *loop* de Monte Carlo, e uma decisão de continuidade precisa ser tomada.
7. Uma vez completo o *loop* de Monte Carlo, o programa está com a soma de todas as configurações geradas, e um processo é chamado para obter a mudança acumulada nos observáveis.
8. Por fim o algoritmo devolve os dados na forma de um arquivo.

A figura 1.1 ilustra o algoritmo. É importante notar que este algoritmo produz os valores dos observáveis para uma dada temperatura, portanto para observar fenômenos como transições de fase, é necessário um *loop* externo que poderá feito com um *script shell* ou implementado diretamente no programa; optei por esta última alternativa.

## 1.5 Observáveis

Para resultados quantitativos, medi a energia e magnetização em cada passo da rotina, tomando o cuidado de não inicializar o sistema enquanto o mesmo está fora do equilíbrio (meu programa entra em equilíbrio após o n° **transiente** de passos). Obtém-se a magnetização somando todos os *spins* da rede e a energia somando a energia de cada spin e dividindo por 2, por contagens duplas. Ainda, o calor específico pode ser obtido da variância na energia:

$$C_V = \frac{1}{kT^2} [\langle E^2 \rangle - \langle E \rangle^2] \quad (1.15)$$

e similarmente a susceptibilidade,  $\chi$ , pode ser escrita em termos da variância da magnetização[3]:

$$\chi = \frac{1}{kT} [\langle M^2 \rangle - \langle M \rangle^2] \quad (1.16)$$

---

<sup>2</sup>através da equação de balanço detalhado1.10, temos  $\frac{dP(x,t)}{dt} = 0$ , dado que o processo se dá pela equação de Chapman-Kolmogorov  $dP(x,t)/dt = -\sum_{x'} W(x_l \leftarrow x_{l'}) + \sum_{x'} W(x_{l'} \leftarrow x_l)$  (ou equação mestra markoviana), e o processo envolve somente estados de equilíbrio.

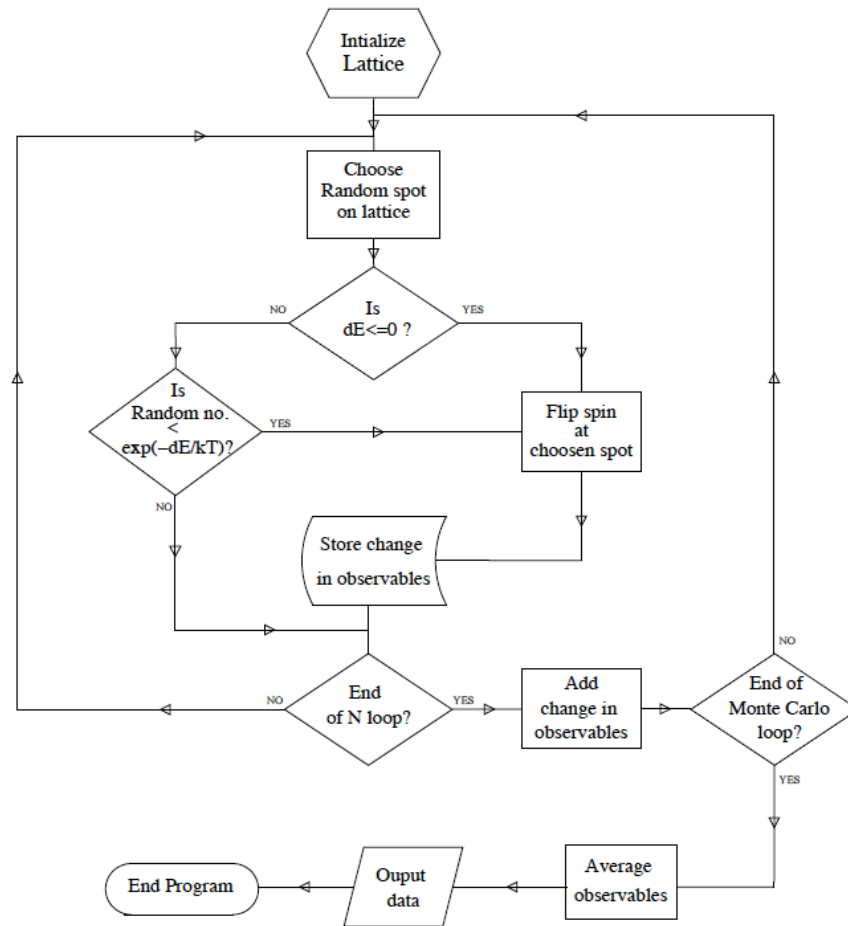


Figura 1.1: Algoritmo Metropolis [4] (retirada de [2]).



## Capítulo 2

# Transições de Fase de Segunda Ordem e Resultados

Nas figuras a seguir mostro os resultados obtidos através do programa construído. Todos os resultados foram obtidos com 10000 passos de Monte Carlo, via Algoritmo de Metropolis, para diferentes redes bi-dimensionais. O incremento de temperatura utilizado foi 0.1 e a termalização acontece após 1000 passos de Monte Carlo (**transiente**). Além disso, optei por uma configuração inicial aleatória e campo magnético,  $B$ , nulo, no entanto isto é arbitrário (ver por exemplo referência [1]).

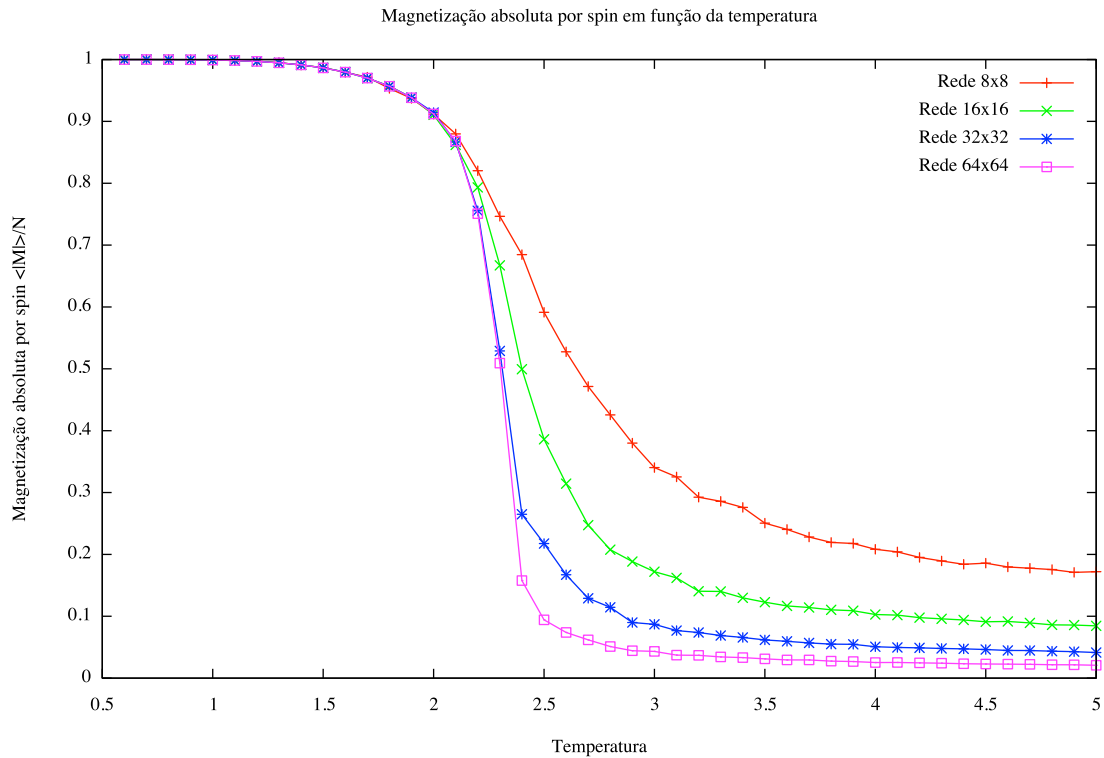


Figura 2.1: Comportamento da magnetização absoluta por spin com a temperatura devido ao aumento da rede.

### 2.1 Magnetização

A partir da configuração inicial, a simulação utiliza o procedimento de Monte Carlo para calcular a magnetização como função do tempo Monte Carlo; portanto cada spin tem a uma oportunidade de inverter-se, por passo de tempo.

Para baixas temperaturas, todos os *spins* estão alinhados paralelamente, com pequenas flutuações durante os passos de Monte Carlo. Com a temperatura aumentada para  $T=2$ , o valor médio de  $M$  diminui a valor correspondente a aproximadamente 90% do total (figura 2.1). O sistema ainda permanece ferromagnético, mas

menos ordenado do que antes. Mais importante que isso, a magnitude das flutuações aumenta significativamente. O aumento das flutuações é um sinal de que estamos nos aproximando de uma transição de fase de segunda ordem, também conhecido como ponto crítico [1].

Um sistema próximo a este ponto crítico torna-se muito sensível as perturbações, assim como suas propriedades mudam rapidamente com o aumento da temperatura. Estas flutuações estão intimamente ligadas a singularidades em  $T = T_c$ , chamada temperatura de Curie. Próximo a  $T=2.25$ , o momento magnético muda completamente de direção, i.e a probabilidade de ocorrer inversões no spin devido a energia fica muito grande. A solução exata para este problema leva a uma temperatura de aproximadamente  $2.27^1$ , muito próxima ao ponto crítico da simulação. Para temperaturas ainda mais elevadas a magnetização se anula e obtemos uma fase paramagnética<sup>2</sup>.

## 2.2 Energia

Na figura 2.2 apresento o a variação da energia com a temperatura, obtida com a mesma simulação utilizada para a magnetização. Para baixas temperaturas, com os *spins* totalmente alinhados, cada spin possui uma energia de interação  $-J$  com cada uma dos seus 4 vizinhos. Para  $T=0$  a energia é, portanto,  $-4NJ/2$ , onde o fator 2 é inserido por contamos 2 vezes cada par. A energia por spin para  $T \rightarrow 0$  é, então, -2 (lembrando que tomamos  $J=1$ ); o resultado da simulação está em perfeito acordo com esta previsão.

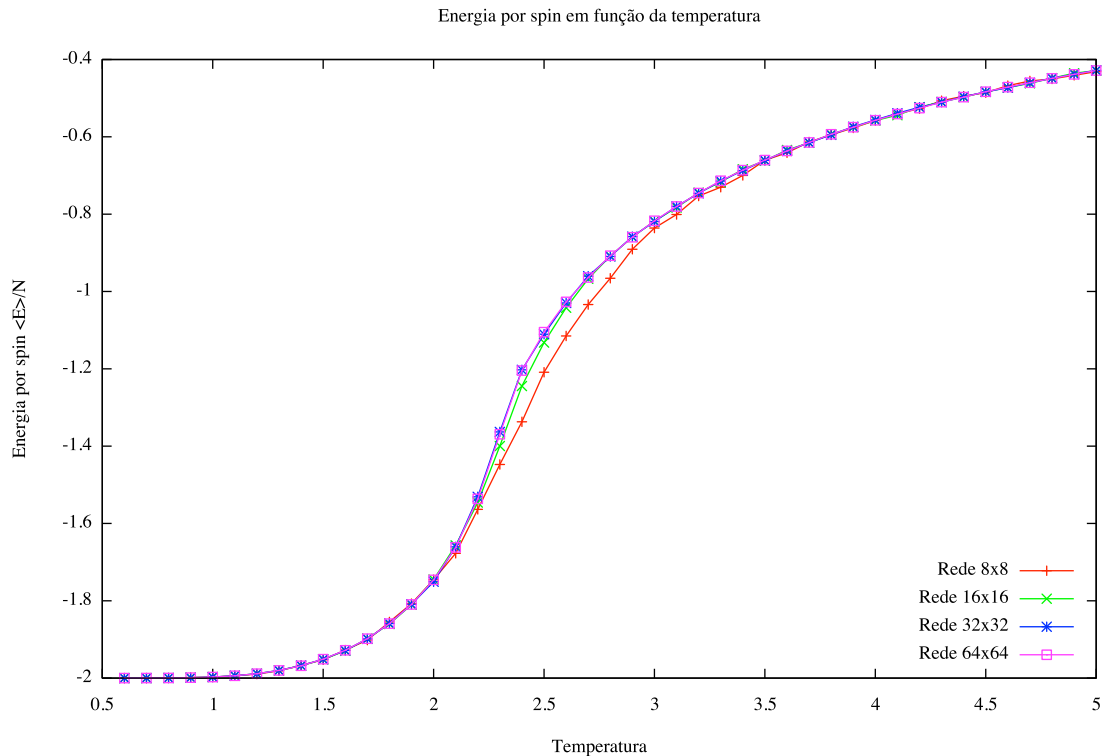


Figura 2.2: Comportamento da energia por spin com a temperatura devido ao aumento da rede. Extrapolando, percebe-se que a inclinação próximo a  $T = 2.27$  diverge.

No outro limite, esperamos *spins* alinhados de maneira aleatória, de forma que  $\langle E \rangle = 0$ , no entanto, para  $T=5$ , bem acima da temperatura de Curie, o resultado observado está substancialmente abaixo do esperado. Isto sugere que as orientações dos *spins* vizinhos são correlacionadas<sup>3</sup>.

Estes argumentos nos permitem entender o comportamento para duas situações limite, mas o comportamento próximo a  $T_c$  não é tão simples. Vemos da figura 2.2 que a  $\langle E \rangle$  exibe um ponto de inflexão, com uma grande inclinação próximo a  $T_c$ . Ora, da termodinâmica, sabemos que esta inclinação está relacionada ao calor específico, significando que o mesmo diverge para  $T = T_c$ <sup>4</sup>.

<sup>1</sup>Em 1942, Onsager desenvolveu a solução exata num verdadeiro *tour de force* e obteve  $T_c = 2/\ln(1 + \sqrt{2})$  [5].

<sup>2</sup>A variação de  $M$  próximo a  $T = T_c$  é determinada pelo expoente crítico  $\beta$  de acordo com  $M \sim (T_c - T)^\beta$ , o resultado analítico fornece  $\beta = 1/8$  [6]

<sup>3</sup>Esta correlação é responsável pelas largas flutuações da magnetização e do calor específico em  $T_c$ . Este é um importante aspecto da teoria que não será explorado além desta nota de rodapé. Para detalhes ver [1]

<sup>4</sup>Esta é outra singularidade associada ao ponto crítico, assim como a magnetização a singularidade é descrita, em geral, por uma

## 2.3 Calor Específico

A equação 1.15 nos mostra que o calor específico está intimamente ligado com as flutuações na energia, portanto a singularidade em  $T = T_c$ , está diretamente relacionada com as grandes flutuações que ocorrem neste ponto. Essa equação pode ser usada para calcular  $C$  usando os resultados obtidos para  $\langle E \rangle$  durante a simulação.

O comportamento do calor específico obtido desta forma é mostrado na figura B.1. Como esperado,  $C$  apresenta um pico na vizinhança de  $T = T_c$ , não divergindo devido ao tamanho finito da rede. Como dito anteriormente, pode-se extrapolar este resultado estudando o comportamento do observável com o aumento da rede, o que revela um aumento do pico com o aumento de  $N$ .

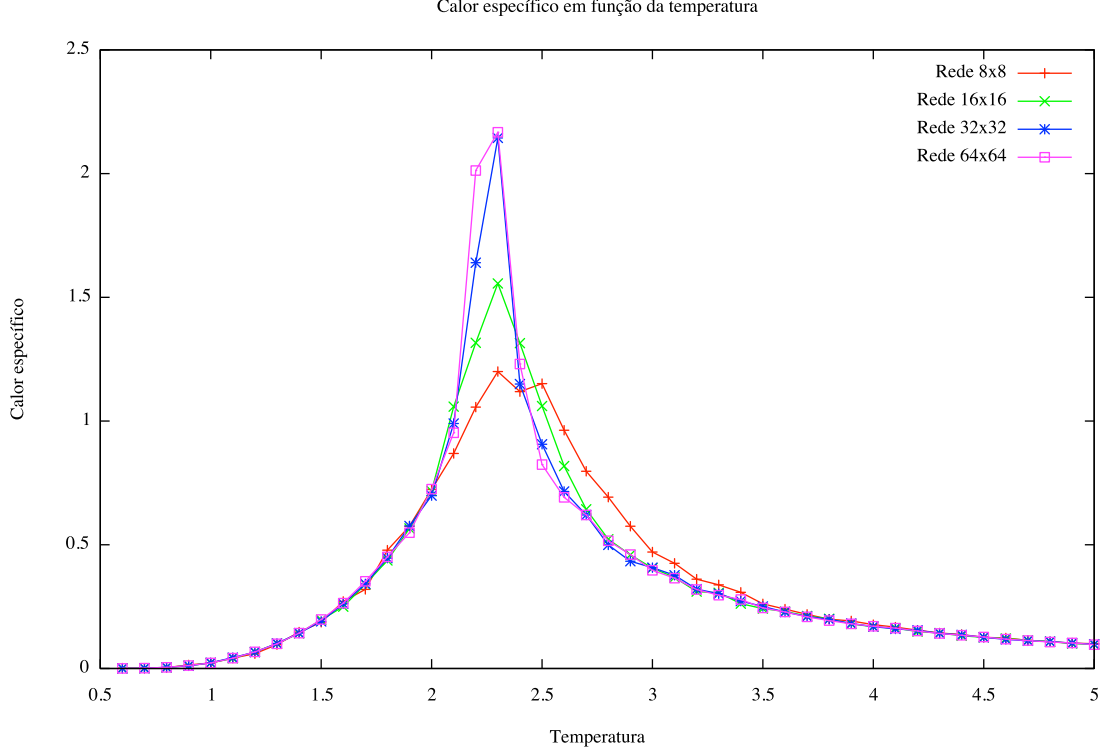


Figura 2.3: Comportamento do calor específico com a temperatura devido ao aumento da rede. Extrapolando, percebe-se a divergência em  $T = 2.27$ .

## 2.4 Conclusão

Neste trabalho estudo uma solução computacional para o modelo de Ising bidimensional, considerando um processo estocástico em combinação com o Algoritmo de Metropolis. Além do estudo, durante o trabalho, desenvolvo um programa (apêndice B) no qual utilizei os conhecimentos obtidos durante a disciplina de Métodos Computacionais da Física B, tais como utilização de números aleatórios e Monte Carlo.

Os resultados obtidos são favoráveis aos resultados teóricos, tanto qualitativamente, e. g. na forma das curvas, quanto quantitativamente, e.g. valor da temperatura de Curie. Estes resultados possuem precisão satisfatória já para uma rede pequena 8x8, assim como melhoram a medida que a aumentamos, e.g. na rede 64x64. A extrapolação nos leva diretamente para a solução exata de Onsager.

O programa desenvolvido permite o estudo de outros observáveis tais como a susceptibilidade magnética, no entanto a análise deste resultado está além do objetivo deste trabalho introdutório, pois envolve o procedimento de *finite size scaling* [2]. A figura produzida está no apêndice B.

---

lei de potência  $C \sim \frac{1}{|T - T_c|^\alpha}$ , onde  $\alpha$  é outro expoente crítico [6].

# Apêndice A

## Programa

O programa foi desenvolvido baseado nos programas apresentados em [3] e [2].

```
#include <iostream>
#include <math.h>
#include <stdlib.h>
#include <fstream>
#define L 8
#include <gsl/gsl_rng.h> /*Compilar com: g++ ising_V0.2.c -lgsl -lgslcblas -lm*/

using namespace std;

ofstream DATA("DATA.1.dat",ios::out);
//Estrutura para a Matriz 2D
struct matrix
{
    int x;
    int y;
};

const int n=L*L; // # de spins
double T=5; //Temperatura Final
const float Tmin=0.5; //temperatura inicial
const float dT=0.1; //passo para a temperatura no loop de temperatura
int S[L][L]; //rede bidimensional
long unsigned int MS=10000; //numero de steps de MC
int transiente=1000; //numero de passos descartados (para evitar a parte fora do equilíbrio.
//int seed=1243; //semente para o gerador
const gsl_rng *r;
//Inicializar a rede
void inicia_rede()
{
    for(int y=0;y<L;y++) //varredura em y
    {
        for(int x=0;x<L;x++)
        {
            if(gsl_rng_uniform(r)>0.5)
                S[x][y]=1;
            else
                S[x][y]=-1;
            //printf("S[%d][%d]=%d \n",x,y,S[x][y]);
        }
    }
}
```

```

//Escolha aleatória do sítio
int escolha_spin(matrix &pos)
{
pos.x=(gsl_rng_uniform(r)*(L)); // (0,L-1)
pos.y=(gsl_rng_uniform(r)*(L));
//printf("[%d] [%d] \n",pos.x,pos.y);
}

//Calculo da energia associada a posição
int energy_pos(matrix &pos)
{
//condições periódicas de contorno
int up, down, left, right,e;
right=(pos.x+1)%L;
left=(pos.x+L-1)%L;
up=(pos.y+1)%L;
down=(pos.y+L-1)%L;

//energia da posição
e=-1*S[pos.x][pos.y]*(S[left][pos.y]+S[right][pos.y]+S[pos.x][up]+S[pos.x][down]); //J=1
//printf("S[x=%d][y=%d], S[L=%d][y=%d], S[R=%d][y=%d], S[x=%d][U=%d], S[x=%d][D=%d] \n", pos.x,pos.y,
return e;
}

//função decisão

bool teste_flip(matrix pos, int &de)
{
de=-2*energy_pos(pos);
if(de<0)
return true; //flip pela energia
else
{
if(gsl_rng_uniform(r)<exp(-de/T))
return true; //flip pelo banho térmico
else
return false; //não flip
}
}

//função de flip

int flip(matrix pos)
{
S[pos.x][pos.y]=-S[pos.x][pos.y];
}

//função que remove o transiente

void transiente_final()
{
matrix pos;
int de=0;
for(int a=1;a<=transiente;a++)
{
for(int b=1;b<=n;b++)
{
escolha_spin(pos);
if(teste_flip(pos,de))
{

```

```

    flip(pos);
  }
}

//Magnetização total da rede (soma dos spins)

int magnetization()
{
    int m=0;
    for(int y=L-1;y>=0;y--)
    {
        for(int x=0;x<L;x++)
        {
            //printf("S[%d] [%d]=%d \n",x,y,S[x][y]);
            m+=S[x][y];
        }
    }
    return m;
}

//Energia total da rede

int energia_total()
{
    matrix pos;
    int e=0;
    for (int y=L-1;y>=0;y--)
    {
        pos.y=y;
        for(int x=0;x<L;x++)
        {
            pos.x=x;
            e+=energy_pos(pos);
        }
    }
    return e;
}

//Programa

int main()
{
    //variáveis para o cálculo dos observáveis
    r = gsl_rng_alloc(gsl_rng_mt19937);
    gsl_rng_set(r, 158235);
    double E=0,Esq=0,Esq_avg=0,E_avg=0,etotal=0, etotalsq=0;

    double M=0, Msq=0, Msq_avg=0,M_avg=0, mtotal=0, mtotalsq=0;
    double Mabs=0,Mabs_avg=0,Mq_avg=0, mabstotal=0, mqttotal=0;
    int de=0;
    matrix pos;

    //inicia a rede de maneira aleatória
    inicia_rede();

    //loop de temperatura
    for(;T>=Tmin;T-=dT)
    {

```

```

//transiente
transiente_final();

//observáveis com valores no equilíbrio (após o transiente)
M=magnetization();
// Mabs=fabs(magnetization());
E=energia_total();

//reseta as variáveis para cada loop
etotal=0;
etotalsq=0;
mtotal=0;
mtotalsq=0;
mabstotal=0;
mqttotal=0;
//loop do Monte Carlo (MS=tempo)
for(int a=1;a<=MS;a++)
{
    //Metropolis
    for(int b=1;b<=n;b++)
    {
        escolha_spin(pos);
        if(teste_flip(pos,de))
        {
            flip(pos);
            //ajusta os observáveis
            E+=2*de;
            M+=2*S[pos.x][pos.y];
            //Mabs+=abs(S[pos.x][pos.y]);
        }
    }
    //soma dos observáveis
    etotal+=E/2.0;
    etotalsq+=(E/2.0)*(E/2.0);
    mtotal+=M;
    mtotalsq+=M*M;
    mqttotal+=M*M*M*M;
    mabstotal+=sqrt(M*M);
}

//Médias:
E_avg=etotal/(MS*n);
Esq_avg=etotalsq/(MS*n);
M_avg=mtotal/(MS*n);
Mabs_avg=mabstotal/(MS*n);
Msq_avg=mtotalsq/(MS*n);
Mq_avg=mqttotal/(MS*n);
//printf("%lf %lf %lf \n",T,Mabs_avg,M_avg);
//saída para arquivo
DATA<<T<< //temperatura
"\t"<<M_avg<<"\t"<<Mabs_avg<<"\t"<<Msq_avg<< //<M>;<|M|>;<M^2> por spin
"\t"<<(Msq_avg-(M_avg*M_avg*n))/(T)<< //suscetibilidade por spin (X)
"\t"<<(Msq_avg-(Mabs_avg*Mabs_avg*n))/(T)<< //suscetibilidade por spin (X')
"\t"<<E_avg<<"\t"<<Esq_avg<< //<E>;<E^2> por spin
"\t"<<(Esq_avg-(E_avg*E_avg*n))/(T*T)<< // capacidade termica por spin
"\t"<<1-((Mq_avg)/(3*Msq_avg))<<endl; //cumulante(U_L)
}
return 0;
}

```

## Apêndice B

# Susceptibilidade Magnética

Embora a transição de fase esteja localizada aproximadamente na temperatura de Curie, a magnitude do pico exige uma análise mais detalhada, feito em [2].

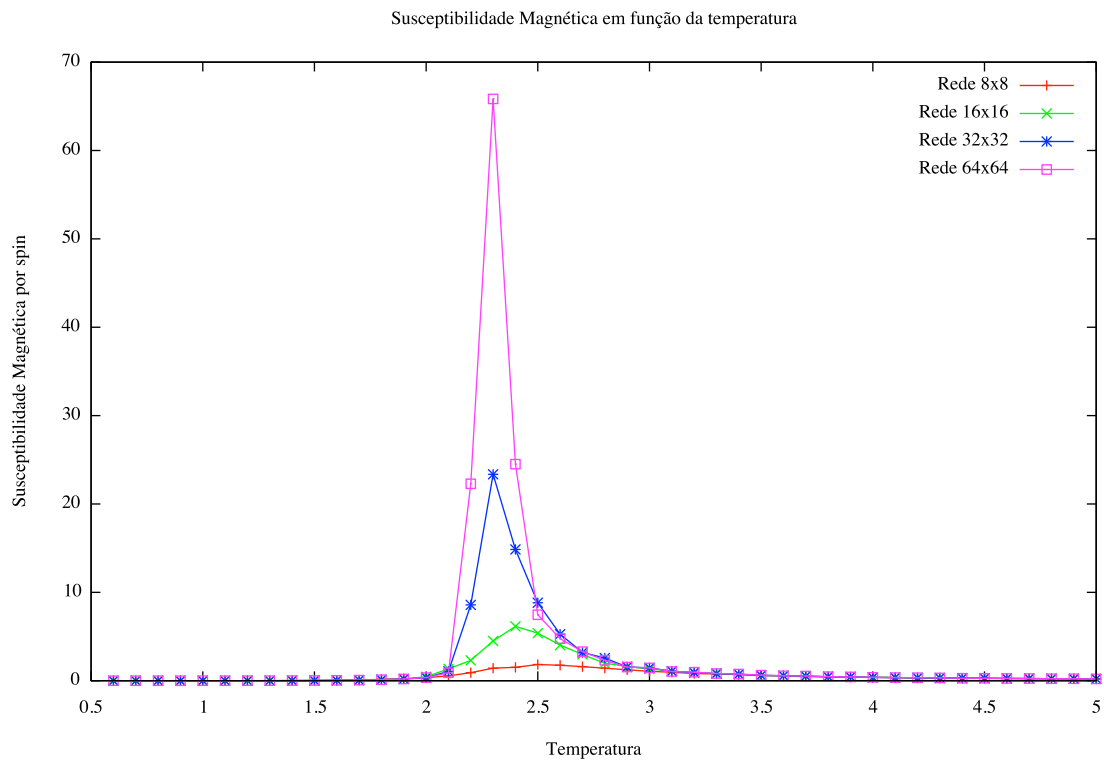


Figura B.1: Comportamento da susceptibilidade magnética com a temperatura devido ao aumento da rede. Extrapolando, percebe-se a divergência em  $T = 2.27$ .



# Bibliografia

- [1] Nicholas J. Giordano. *Computational Physics*, chapter 8. Prentice Hall, 1997.
- [2] J. Kotze. Introduction to Monte Carlo methods for an Ising Model of a Ferromagnet. *ArXiv e-prints*, March 2008.
- [3] Lisa Larrimore. Monte carlo simulation of the 2d ising model, 2011.
- [4] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.
- [5] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944.
- [6] H.E. Stanley. *Introduction to Phase Transitions and Critical Phenomena*. Claredon Press, 1971.