

## Modelo de Ising 2D – Enumeração Exata

Nesta tarefa daremos início ao estudo do modelo de Ising numa rede bidimensional quadrada através de simulações computacionais. Nossa primeira abordagem será através da enumeração exata de todos os estados. Devido ao alto custo computacional deste tipo de abordagem, veremos que apenas redes muito pequenas poderão ser estudadas, mesmo assim, num tempo computacional considerável, o que nos levará a pensar em maneiras mais eficientes de obter as propriedades do sistema. No que segue, vocês serão guiados na solução do modelo através desse método.

### Introdução:

Nosso objetivo será obter propriedades termodinâmicas do modelo de Ising partindo de sua função energia. Para tanto, precisamos calcular a função de partição e, a partir dela, as propriedades termodinâmicas poderão ser obtidas por diferenciação ou cálculo numérico direto. A função de partição é dada por

$$Z = \sum_{\{\sigma\}} e^{-\beta E(\sigma)},$$

onde a soma é realizada sobre todas as configurações acessíveis ao sistema e  $E(\sigma)$  é a energia da configuração  $\sigma$ . Assim, a abordagem que será tratada aqui corresponde a gerar através de um código computacional todas as configurações acessíveis ao sistema, calculando a energia de cada uma delas e realizando a soma. Uma forma mais conveniente de escrever a função de partição e que permitirá um ganho de tempo e esforço computacional é

$$Z = \sum_E g(E) e^{-\beta E},$$

onde  $g(E)$  é o número de estados com energia  $E$ . A maior vantagem de se escrever a função de partição dessa forma está no fato de termos trocado uma soma sobre os estados ( $2^N$  para o modelo de Ising com  $N$  spins) por uma soma sobre os valores de energia ( $N + 1$  valores distintos). O preço a se pagar nessa simplificação está na dificuldade em se determinar  $g(E)$ . Escrevendo desta forma, o valor esperado de qualquer grandeza física,  $A$ , que dependa apenas da energia pode ser escrito como

$$\langle A(E) \rangle = \frac{\sum_E A(E) g(E) e^{-\beta E}}{Z}.$$

A energia média, por exemplo, será dada por

$$\langle E \rangle = \frac{\sum_E E g(E) e^{-\beta E}}{Z}.$$

Perceba que toda dependência na temperatura está na variável  $\beta$  e que  $g(E)$  independe da temperatura ou outras condições às quais o sistema está sujeito. Caso a grandeza de interesse dependa também da magnetização,  $M$ , temos

$$\langle A(E, M) \rangle = \frac{\sum_{E, M} A(E, M) g(E, M) e^{-\beta E}}{Z},$$

onde  $g(E, M)$  representa o número de configurações do sistema com energia  $E$  e magnetização  $M$ .

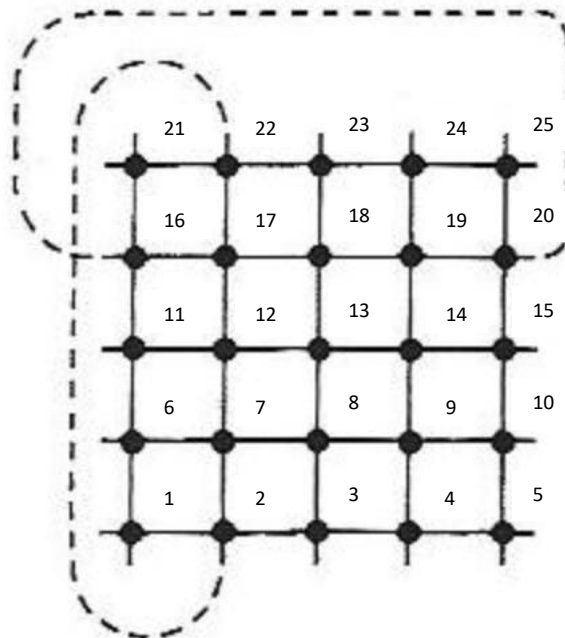
A forma mais direta que podemos pensar em obter  $g(E)$  (ou equivalentemente  $g(E, M)$ ) é gerar todas as configurações do sistema, calcular a energia de cada uma delas e contar quantas configurações com cada valor de energia temos. O que iremos fazer é praticamente isso, porém de

uma forma um pouco mais eficiente. Antes de passar ao cálculo em si, convém fazermos considerações de como representaremos o modelo de Ising num programa computacional.

### Implementação computacional do modelo de Ising.

Visando uma implementação um pouco mais otimizada do modelo de Ising do que um programador sem experiência na área faria, abaixo você será guiado em formas mais eficientes e gerais de representar modelos em redes.

Iremos considerar o modelo de Ising numa rede quadrada bidimensional com  $L \times L$  sítios. A posição geométrica de cada sítio, ou seja, suas coordenadas  $x$  e  $y$ , pelo menos em nossa abordagem inicial, não será necessária. Assim, é mais conveniente e eficiente que cada sítio seja indexado por apenas um número inteiro. A convenção a ser adotada aqui é a representada na figura abaixo para uma rede  $5 \times 5$ .



De posse dessa convenção, o estado do sistema pode ser representado por um array unidimensional com  $N = L^2$  elementos. Cada elemento recebe o valor  $\pm 1$ , indicando o valor da variável de spin do sítio correspondente. Uma sugestão é nomear esse array de  $S$ .

Uma vez definido o estado do sistema, devemos ser capazes de calcular a energia deste estado. Por definição, a energia é dada por

$$E = - \sum_{(i,j)} s_i s_j - B \sum_i s_i.$$

Por simplicidade vamos considerar apenas o caso  $B = 0$ . Nesta definição,  $(i,j)$  representa os primeiros vizinhos na rede quadrada. Assim, por exemplo, o sítio 13 tem como primeiros vizinhos os sítios 14, 18, 12 e 8. Uma pergunta que cabe neste ponto é como trataremos os spins nas bordas. Quais seriam os vizinhos do sítio 5, por exemplo. Uma vez que estamos interessados, em geral, no comportamento do sistema no limite termodinâmico, ou seja, de um sistema muito grande, suas bordas devem ser irrelevantes. Assim, para minimizarmos os efeitos das fronteiras (bordas) da rede

sobre as propriedades do sistema, é comum usar condições de contorno periódicas. Neste tipo de tratamento das fronteiras, consideraremos que o sítio 5 tem como vizinho à direita o sítio 1 e como vizinho abaixo, o sítio 25. É como se transformássemos essa rede quadrada em um toro, unindo suas extremidades. Uma forma de guardarmos de forma adequada essas informações é utilizar uma tabela de vizinhos. Há diversas formas de fazermos isso. Aqui, iremos usar um array bidimensional onde o primeiro elemento representará o sítio da rede enquanto o segundo representará qual dos vizinhos estamos considerando. Utilizaremos o índice 1 para o vizinho à direita, o 2 para o vizinho acima, o 3 para o vizinho à esquerda e o 4 para o vizinho abaixo. Os sítios da linha inferior têm índice,  $i$ , menor que  $L + 1$ . Já os da lateral esquerda são aqueles para os quais o resto da divisão inteira de  $i - 1$  por  $L$  é zero. Os da lateral direita são aqueles para os quais o resto da divisão inteira de  $i$  por  $L$  é zero e os da linha superior são aquelas para os quais  $i > N - l$ . Podemos, então, usar o seguinte pseudocódigo:

```

Para  $i = 1, \dots, N$  faça
     $viz(k, 1) \leftarrow k + 1$ 
    se  $(k \bmod l = 0)$   $viz(k, 1) \leftarrow k + 1 - l$ 
     $viz(k, 2) \leftarrow k + l$ 
    se  $(k > n - l)$   $viz(k, 2) \leftarrow k + l - n$ 
     $viz(k, 3) \leftarrow k - 1$ 
    se  $(k - 1 \bmod l = 0)$   $viz(k, 3) \leftarrow k + l - 1$ 
     $viz(k, 4) \leftarrow k - l$ 
    se  $(k < l + 1)$   $viz(k, 4) \leftarrow k + n - l$ 

```

Uma vez que definimos a rede, os vizinhos e as variáveis de spin, podemos calcular a energia de qualquer configuração. Perceba, no entanto, que para contar cada par de spins apenas uma vez podemos considerar apenas os vizinhos à direita (1) e acima (2) de cada sítio. Então, um pseudocódigo do cálculo da energia fica:

```

 $E \leftarrow 0$ 
Para  $i = 1, \dots, N$  faça
     $h \leftarrow s(viz(i, 1)) + s(viz(i, 2))$ 
     $E \leftarrow E - s(i) * h$ 

```

Neste ponto você deve estar se perguntando: “se precisamos de apenas de 2 dos 4 vizinhos, por que nos preocupamos em definir os 4 vizinhos anteriormente?” A definição dos 4 vizinhos se justifica pelo fato de o cálculo da energia total envolver um loop de tamanho  $N$ . Mas, ao flipar (mudar a direção) apenas um spin podemos calcular a diferença de energia considerando apenas os 4 vizinhos. Assim, se fliparmos o spin do sítio  $i$ , ou seja, se fizermos  $s'_i = -s_i$ , a energia após o flip,  $E_f$ , pode ser obtida da energia antes de fliparmos o spin,  $E_i$ , fazendo

$$E_f = E_i + 2 * s_i * \left( \sum_j s_j \right) = E_i - 2 * s'_i * \left( \sum_j s_j \right).$$

Essa última expressão mostra que as energias permitidas para o modelo de Ising estão no intervalo de  $-2N$  a  $2N$  em passos de 4, sendo que os estados de energia  $-2N + 4$  e  $2N - 4$  não são permitidos. Verifique essas afirmações.

## Enumeração exata

Pelo exposto até agora estamos em condições de fazer algumas considerações sobre a enumeração exata do modelo de Ising para determinar  $g(E)$ . Em primeiro lugar devemos notar que um sistema com  $N$  spins possui  $2^N$  estados. Isso implica que uma rede “minúscula” de apenas 6x6 spins possui um número de estados maior que aquele que conseguimos representar numa variável inteira de 32 bits convencional. Se pensarmos num sistema com 100 spins, uma rede 10x10, possui da ordem de  $10^{30}$  configurações. Se o tempo gasto para gerar e calcular a energia de cada um dos estados for de  $10^{-9}$  s, um nano-segundo, o tempo total para se determinar  $g(E)$  seria da ordem de  $10^{21}$  segundos, muito maior que a idade estimada do universo, 14 bilhões de anos  $\sim 10^{17}$  segundos! O tempo de computação cresce exponencialmente, tornando completamente inviável a enumeração de redes “grandes”. De qualquer forma, esse é um exercício instrutivo. Deve-se ressaltar que há métodos que permitem a obtenção do  $g(E)$  exato para redes muito maiores que conseguimos enumerar. É possível, num intervalo de cerca de 1 semana de computação num desktop convencional, obter  $g(E)$  para redes de tamanho 150x150. Aqui, conseguiremos de forma razoável ir até redes 6x6.

Com já foi argumentado, não iremos calcular a energia de todas as configurações. Apenas as diferenças de energia serão consideradas, numa tentativa de diminuir o tempo computacional, permitindo obter resultados para sistemas maiores. Uma ideia comum nesse ponto é utilizarmos a representação binária dos números inteiros para codificar o estado do sistema. Um problema nessa abordagem é que do número inteiro 1 para o 2, por exemplo, alteramos dois bits de uma vez na representação binária. De fato, numa palavra de 4 bits o número 1 é representado por 0001 e o número 2 por 0010. Assim, se representarmos cada estado do sistema através da representação binária dos números deveríamos calcular a energia da rede inteira ou dois spin flips a cada nova configuração (número) gerado. O custo computacional disso é elevado devido ao número excessivo de operações envolvidas no cálculo da energia. Por esse motivo, proponho usar um processo chamado de gray code. Em gray code, dois valores consecutivos diferem por apenas um bit. Assim como tudo o que foi exposto até aqui, há diferentes formas de se fazer e implementar essa ideia. Proponho usarmos a seguinte rotina, que indica a sequência de sítios da rede que deve ser flipada para conseguirmos gerar todas as configurações através de flips únicos.

```
Rotina gray_flip  
Entrada  $\{t_0, t_1, \dots, t_N\}$   
 $k \leftarrow t_0$   
Se  $(k > N)$  Saia  
 $t_{k-1} \leftarrow t_k$   
 $t_k \leftarrow k + 1$   
Se  $(k \neq 1)$   $t_0 \leftarrow 1$   
Saída  $k, \{t_0, t_1, \dots, t_N\}$ 
```

Inicialmente devemos ter  $\{t_0, t_1, \dots, t_N\} = \{1, \dots, N + 1\}$ .

A ideia deste algoritmo pode ser sumarizada pela figura abaixo, que deve ser lida da seguinte forma: Imagine que você dobre a tabela na linha horizontal entre as configurações  $i = 8$  e  $i = 9$ . Neste caso as configurações dos spins  $s_1, s_2$  e  $s_3$  permanecem inalteradas entre as configurações  $i = 8$  e  $i = 9$  e o único spin flipado é o  $s_4$  que muda de  $-$  para  $+$ . O mesmo acontece para as configurações  $i = 7$  e  $i = 10$ ,  $i = 6$  e  $i = 11$  e assim por diante. Ao incluirmos mais um spin, a ideia seria refletir a tabela

abaixo da linha 16, chegando até a configuração 32, incluir mais uma coluna ao final da tabela preenchida até 16 com – e preencher a coluna adicionada com + até a configuração 32.

$i$	$\{\sigma_1, \dots, \sigma_4\}$			
1	–	–	–	–
2	+	–	–	–
3	+	+	–	–
4	–	+	–	–
5	–	+	+	–
6	+	+	+	–
7	+	–	+	–
8	–	–	+	–
9	–	–	+	+
10	+	–	+	+
11	+	+	+	+
12	–	+	+	+
13	–	+	–	+
14	+	+	–	+
15	+	–	–	+
16	–	–	–	+

Podemos economizar um pouco mais de tempo computacional observando uma simetria presente no modelo de Ising a campo nulo,  $B = 0$ . Nesse caso, ao multiplicar todos os spins por  $-1$ , a energia do sistema permanece inalterada. Com isso, precisamos considerar apenas metade das configurações no cálculo de  $g(E)$ . Estamos, então, em condições de escrever um algoritmo capaz de obter  $g(E)$  para o modelo de Ising.

**Rotina** *enumera\_ising*

$\{g(-2N), \dots, g(2N)\} \leftarrow \{0, \dots, 0\}$

$\{s_1, \dots, s_N\} \leftarrow \{-1, \dots, -1\}$

$\{t_0, \dots, t_N\} \leftarrow \{1, \dots, N + 1\}$

$E \leftarrow -2N$

$g(E) \leftarrow 2$

**Para**  $i = 1, \dots, 2^{N-1} - 1$  **faça**

$k \leftarrow \text{gray\_flip}(\{t_0, \dots, t_N\})$

$h \leftarrow \sum_{(k,j)} s_j$ . (essa soma é sobre os vizinhos de  $k$ )

$E \leftarrow E + 2 \cdot s_k \cdot h$

$g(E) \leftarrow g(E) + 2$

$s_k \leftarrow -s_k$

**Saída**  $\{g(E) > 0\}$

## Obtendo propriedades termodinâmicas

As principais grandezas termodinâmicas que iremos obter são a energia média e o calor específico (ou capacidade térmica). Você deve se lembrar da capacidade térmica do ensino médio, que relaciona a quantidade de calor (energia) fornecida ao sistema e o aumento de temperatura correspondente. A diferença entre calor específico e capacidade térmica é que a capacidade térmica

depende da massa (ou volume) do sistema em consideração, enquanto o calor específico é a capacidade térmica por unidade de massa (volume), sendo, portanto, uma propriedade da substância. Em muitos casos, numa transição de fase, essa grandeza apresenta uma divergência no limite termodinâmico, indicando que uma quantidade enorme de energia deve ser fornecida para o sistema para provocar uma mudança relativamente pequena em sua temperatura. Em outras palavras, para que a transição entre fases ocorra, uma grande quantidade de energia deve ser fornecida. Em alguns casos, como da ebulição ou fusão da água, temos um calor latente, onde energia é fornecida sem que haja mudança na temperatura. Isso se reflete como uma descontinuidade na capacidade térmica. Em outros casos, não há tal descontinuidade e toda energia fornecida se reflete num aumento da temperatura, no entanto, a quantidade de energia que deve ser fornecida para um dado aumento de temperatura será maior próximo à transição de fase.

Pode-se mostrar que o calor específico pode ser calculado como:

$$c_v = \frac{\beta^2}{N} (\langle E^2 \rangle - \langle E \rangle^2)$$

De posse de  $g(E)$  a energia média por spin,  $\langle e \rangle$ , e o calor específico,  $c_v$ , podem ser facilmente obtidos. Para tanto, basta seguir o código apresentado abaixo.

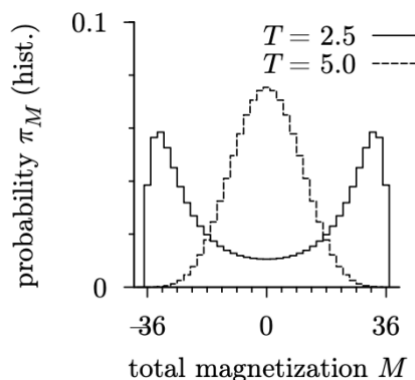
```

Rotina media_termodinamica
Entrada { $g(E_{min}), \dots, g(E_{max})$ }
 $Z \leftarrow 0$ 
 $\langle E' \rangle \leftarrow 0$ 
 $\langle E'^2 \rangle \leftarrow 0$ 
Para  $E = E_{min}, \dots, E_{max}$  faça
     $E' \leftarrow E - E_{min}$ 
     $Z \leftarrow Z + g(E)e^{-\beta E'}$ 
     $\langle E' \rangle \leftarrow \langle E' \rangle + E' g(E)e^{-\beta E'}$ 
     $\langle E'^2 \rangle \leftarrow \langle E'^2 \rangle + E'^2 g(E)e^{-\beta E'}$ 
 $\langle E' \rangle \leftarrow \langle E' \rangle / Z$ 
 $\langle E'^2 \rangle \leftarrow \langle E'^2 \rangle / Z$ 
 $Z \leftarrow Z e^{-\beta E_{min}}$ 
 $c_v \leftarrow \beta^2 (\langle E'^2 \rangle - \langle E' \rangle^2) / N$ 
 $\langle e \rangle \leftarrow \frac{(\langle E' \rangle + E_{min})}{N}$ 
Saída { $Z, \langle e \rangle, c_v$ }

```

Os algoritmos apresentados podem ser modificados para calcular grandezas que dependam da magnetização do sistema. Apenas a título de ilustração, apresento abaixo um gráfico da probabilidade de se obter diferentes valores de magnetização em duas temperaturas distintas, 2.5 e 5.0 para uma rede 6x6 obtidos com o procedimento delineado acima. Perceba que a altas temperaturas a distribuição de probabilidades apresenta apenas um máximo próximo da magnetização nula. A baixas temperaturas, há uma modificação considerável no comportamento e a distribuição de probabilidades passa a apresentar dois máximos, um próximo a -36 e outro próximo a 36. Essa mudança de comportamento que pode ser verificada até mesmo numa rede tão pequena é uma característica da transição de fase que ocorre no sistema. A baixas temperaturas o sistema “escolherá” uma destas duas possibilidades, magnetização positiva ou negativa, indicando

o ordenamento do sistema. A altas temperaturas o sistema estará sem ordenamento, resultando numa magnetização próxima de zero.



## TAREFA

Nesta tarefa vocês devem implementar o método discutido neste documento e realizar cálculos para redes de tamanhos 2x2, 4x4 e 6x6. Vocês deverão fazer gráficos das seguintes grandezas para temperaturas entre 1 e 5 com os três tamanhos estudados:

- i)  $\ln(g(E))$  como função de  $E/N$ . (Perceba que isto independe da temperatura)
- ii) Energia por spin,  $\langle e \rangle$ , como função da temperatura
- iii) Calor específico,  $c_v$ , como função da temperatura
- iv) Energia livre por spin,  $f(T) = -\frac{1}{N\beta} \ln Z$
- v) Entropia por spin,  $s(T) = \frac{\langle e \rangle - f(T)}{T}$

Vocês devem entregar um arquivo pdf (pode ser a impressão de um notebook, por exemplo) contendo o código usado, os gráficos e uma análise dos resultados obtidos.

Fiquem atentos, principalmente ao lidar com a rede maior, 6x6, uma vez que os valores de  $g(E)$ , assim como o número de estados a serem enumerados superam o limite de números inteiros de 32 bits usados como padrão em várias linguagens de programação. Deixo, ainda, duas dicas úteis para programação deste tipo de problema:

- 1) Antes de começar a programar, pense! Você realmente entendeu o que deve fazer? Entendeu o modelo e suas peculiaridades? Pensou em como implementar tudo que pretende fazer?
- 2) No começo, pense pequeno! O ideal é sempre começar pelo menor tamanho de rede, onde é possível fazer tudo “na mão” para conferir. Aumente o tamanho aos poucos tentando perceber o que está acontecendo. Não se assuste com o aumento no tempo computacional. Lembre-se que esse tempo aumenta exponencialmente! Ao simular a rede 6x6, tome um café enquanto o programa calcula  $g(E)$ . Não deixe de salvar o resultado em um arquivo separado para não ter necessidade de rodar novamente, uma vez que esse é um cálculo que demanda certo tempo computacional.

## REFERÊNCIA:

- W. Krauth, Statistical Mechanics Algorithms and Computations, Oxford 2006