

IMPLEMENTACIÓN DEL MÉTODO DE METRÓPOLIS PARA EL MODELO DE ISING 2D.

UNIVERSIDAD DE LOS ANDES - FACULTAD DE CIENCIAS

Departamento de Física - Mecánica Estadística

Cristian F. Rodriguez C.

2 de octubre de 2020

Ver versión actualizada de este documento en GitHub: <https://github.com/crodriguez45/StatisticalMechanics.git>

Índice

1. Marco Teórico	1
2. Implementación	1
2.1. Matriz de Espines	1
2.2. Algoritmo Metrópolis	2
2.3. Observables Físicos	2
2.4. Ejecución a temperatura constante	2
2.5. Barrido de Temperaturas	3
3. Discusión y Resultados	3
3.1. Temperatura constante	3
3.2. Barridos de Temperatura	4
3.2.1. Red de mayor tamaño	4

1. Marco Teórico

El hamiltoniano del modelo de Ising en ausencia de campo magnético es

$$\hat{H} = -J' \sum_{\langle A, B \rangle} \hat{s}_A \hat{s}_B,$$

donde la suma sólo corre sobre los vecinos más próximos entre A y B . En un sistema de unidades donde los valores propios de \hat{s}_A sean ± 1 de modo que la energía de una configuración dada es,

$$E = -J \sum_{\langle A, B \rangle} s_A s_B$$

Para este sistema sencillamente la energía media por espín estará dada por

$$\epsilon = \frac{\langle H \rangle}{N} = -\frac{J}{N} \sum \langle A, B \rangle \langle s_A s_B \rangle$$

donde N es el número de espines en el sistema, mientras que la magnetización por espín es directamente

$$\mu = \frac{\langle M \rangle}{N} = \frac{1}{N} \sum_A \langle s_A \rangle$$

Si un espín en el sitio A es espontáneamente invertido, la energía de los cuatro enlaces que recubren a ese espín varía, de modo que la energía del sistema cambia de modo que

$$\Delta_A E = -J \Delta s_A \left(\sum_B s_B \right) \quad (1.1)$$

donde la suma sobre B corre sobre los vecinos de A .

El algoritmo de Metrópolis se basa en generar configuraciones aleatorias del sistema cuyo peso estadístico está en concordancia con la descripción canónica y usar estas para calcular promedios de cantidades físicas.

$$s_{i,j} = s_{i+n,j} \quad s_{i,j} = s_{i,j+m} \quad \forall (i,j) \in \mathbb{Z}^2 \quad (1.2)$$

2. Implementación

2.1. Matriz de Espines

Para la red 2D de espines, se tiene una matriz de dimensión $n \times m$ con condiciones de frontera periódicas, para simplificar la manera en la que estas condiciones de frontera son tomados hacemos una extensión a la matriz agregando una cadena de espines a lo largo de la frontera; de modo que si en la matriz física tenemos dos contadores i, j para caracterizar al espín $A = A(i, j)$, que recorren en el dominio $i \in [1, \dots, n]$ y $j \in [1, \dots, m]$, en el código la matriz de espines s_{ij} tiene entradas entre $i \in [0, \dots, n+1]$ y $j \in [0, \dots, m+1]$ (ver figura (2.1)). Explícitamente las condiciones de frontera fijan los valores de la cadena auxiliar de acuerdo a

$$\begin{cases} s_{0,j} = s_{n,j} \\ s_{n+1,j} = s_{1,j} \end{cases} \quad \begin{cases} s_{i,0} = s_{i,m} \\ s_{i,m+1} = s_{i,1} \end{cases}$$

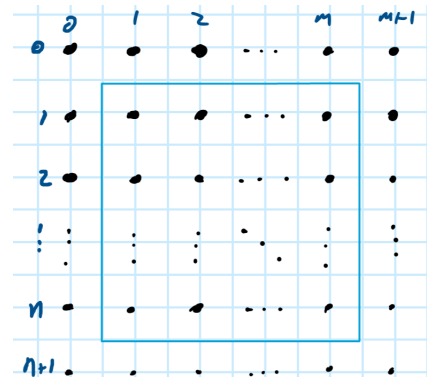


Figura 2.1: Representación esquemática de la matriz de espines, los sitios de red dentro del recuadro hacen parte de la matriz física, mientras que los externos son espines auxiliares que facilitan los cálculos y están determinados por condiciones de frontera periódicas.

En resumen, definimos una matriz de $(n + 2, m + 2)$ entradas, cuyas componentes $i \in [1, \dots, n]$ y $j \in [1, \dots, m]$ son inicializadas al azar entre ± 1 con el algoritmo 1 y las fronteras empleando las condiciones de frontera vía el algoritmo 2.

Algorithm 1: Inicializar

```

1 for  $i \leftarrow 1$  to  $n$  do
2   for  $j \leftarrow 1$  to  $m$  do
3      $s_{i,j} = 2(\text{RandomInteger} \text{ mód } 2) - 1$ ;

```

Algorithm 2: Condiciones de Frontera

```

1 for  $j \leftarrow 1$  to  $m$  do
2    $s_{0,j} \leftarrow s_{n,j}$ 
3    $s_{n+1,j} \leftarrow s_{1,j}$ 
4 for  $i \leftarrow 0$  to  $n + 1$  do
5    $s_{i,0} \leftarrow s_{i,m}$ 
6    $s_{i,m+1} \leftarrow s_{i,1}$ 

```

2.2. Algoritmo Metrópolis

Para la implementación de Metrópolis es necesario conocer el cambio de energía causado por la inversión de un espín en ubicación $s_{i,j}$, para ello calculamos $\Delta E(i, j)$ vía (1.1), los primeros vecinos de $s_{i,j}$ son $s_{i,j-1}$, $s_{i,j+1}$, $s_{i-1,j}$, $s_{i+1,j}$ (ver figura), de modo que explícitamente

$$\Delta E(i, j) = -J\Delta s_{i,j} (s_{i,j-1} + s_{i,j+1} + s_{i-1,j} + s_{i+1,j})$$

por otra parte, la inversión del espín es simplemente un cambio de signo en el mismo, con lo cual $s_{i,j} \mapsto -s_{i,j} \Rightarrow \Delta s_{i,j} = -2s_{i,j}$. Una función simple que calcula esto está dado por el algoritmo (4), lo cual es una pequeña variación más eficiente que calcular la energía final e inicial de todo el sistema.

Algorithm 3: Delta_E(i,j)

```

1  $\Delta s_{i,j} \leftarrow -2s_{i,j}$ 
2 return  $-J\Delta s_{i,j} (s_{i,j-1} + s_{i,j+1} + s_{i-1,j} + s_{i+1,j})$ 

```

Ya para implementar directamente metrópolis se debe seleccionar un espín al azar dentro de la red, claramente se selecciona entre la red física, y al finalizar se verifica que las condiciones de frontera se sigan cumpliendo; para la generación del número aleatorio entre $[0, 1]$, se genera un entero aleatorio y se normaliza al máximo entero que se puede generar.

¹ Para optimizar esto se recomienda ejecutar algún criterio de convergencia.

Algorithm 4: Metropolis

```

1 let  $I \leftarrow 1 + \text{RandomInteger} \text{ mód } n$ 
2 let  $J \leftarrow 1 + \text{RandomInteger} \text{ mód } m$ 
3 if  $\Delta E(I, J) \leq 0$  then
4    $s_{I,J} \leftarrow -1 \times s_{I,J}$ 
5 else
6    $p = \text{RandomInteger} / \text{Random\_Max}$ 
7   if  $p < e^{-\beta \Delta E(I, J)}$  then
8      $s_{I,J} \leftarrow -1 \times s_{I,J}$ 
9 Execute Condiciones de frontera

```

2.3. Observables Físicos

Los principales observables de este sistema son la densidad de energía y la densidad de magnetización, cada una una densidad definida por el promedio de la energía de cada enlace, y la magnetización de cada enlace. Para la magnetización basta con sumar sobre todos los espines de la matriz mientras que con la energía hay que ser cuidadosos de no sumar doblemente la energía de cada enlace. Para ello se propone que por cada sitio en la red se sume solamente la energía del vecino que se encuentra arriba y a la derecha de dicho sitio, al barrer sobre todos los sitios se barre también todos los enlaces sin repetir ninguno. Sin embargo, para la magnetización se tiene una simetría de paridad bajo la inversión de todo el sistema, de modo que para la magnetización tomaremos solamente su magnitud y depreciaremos su dirección, como se muestra al final del algoritmo 6.

Algorithm 5: Densidad de energía

```

1 let  $X = 0$  for  $i \leftarrow 1$  to  $n$  do
2   for  $j \leftarrow 1$  to  $m$  do
3      $X \leftarrow X + s_{i,j} (s_{i-1,j} + s_{i,j+1})$ 
4 return  $-JX / (n \times m)$ 

```

Algorithm 6: Densidad de magnetización

```

1 let  $X = 0$  for  $i \leftarrow 1$  to  $n$  do
2   for  $j \leftarrow 1$  to  $m$  do
3      $X \leftarrow X + s_{i,j}$ 
4 return  $|X| / (n \times m)$ 

```

2.4. Ejecución a temperatura constante

Aquí se recopilan los procedimientos anteriores y se ejecuta el algoritmo metrópolis de manera que el sistema completamente aleatorio termaliza con un reservorio a temperatura T . El número de veces que se ejecuta metrópolis $N_{\text{máx}}$ es determinado heurísticamente partiendo de un valor tentativo y observando el comportamiento del sistema¹; para esto, se registra y grafica el comportamiento del sistema en cada ejecución de metrópolis para una

temperatura dada. Similarmente, ya para obtener los observables al estar en equilibrio térmico heurísticamente se determina el mínimo de pasos que deben efectuarse antes de que termalice el sistema, se promedian los valores en equilibrio para luego ser reportados, ver algoritmo 7.

Algorithm 7: Ejecución a Temperatura constante

```

1 let  $draw = \text{true}$ 
2 let  $NN = 0$ 
3 let  $E = 0$ 
4 let  $M = 0$ 
5 Execute Inicializar
6 Execute Condiciones de Frontera
7 while  $NN < N_{\text{máx}}$  do
8   if  $draw$  then
9     Registrar( $NN$ , Energía, Magnetización)
10  Execute Metrópolis
11  if  $N > N_{\text{mín}}$  then
12     $E \leftarrow E + \text{Energía}$ 
13     $M \leftarrow M + \text{Magnetización}$ 
14 if  $draw$  then
15   Graficar registro( $NN$ , Energía)
16   Graficar registro( $NN$ , Magnetización)
17  $E \leftarrow E / (N_{\text{máx}} - N_{\text{mín}})$ 
18  $M \leftarrow M / (N_{\text{máx}} - N_{\text{mín}})$ 
19 return ( $E$ ,  $M$ )

```

2.5. Barrido de Temperaturas

Para esta sección se escoge un rango de temperaturas partido en n_T subrangos, para cada temperatura en este rango se ejecuta el algoritmo a temperatura constante un total de n_M veces. Para cada temperatura se registran las n_M salidas de energía y magnetización, estas son promediadas y su desviación estándar también es determinada. Se grafican las n_T energías con su desviación contra su respectiva temperatura, y se realiza lo mismo con la magnetización.

Algorithm 8: Ejecución a multiples temperaturas

```

1 for  $T_{\text{mín}}$  to  $T_{\text{máx}}$  do
2    $\beta \leftarrow \frac{1}{K_b T}$ 
3   for  $j \leftarrow 0$  to  $n_M$  do
4     Execute Ejecución a temperatura constante con
        $draw = \text{false}$ 
5     registro1=registrar( $E$ ,  $M$ )
6   registro2=registrar(promedio registro 1)
7 Graficar registro2

```

3. Discusión y Resultados

3.1. Temperatura constante

Ejecutando el algoritmo a temperatura constante, se observaron diferentes comportamientos en función del tamaño de

la red, el número de pasos para la convergencia de la energía y la magnetización crecía con el tamaño de la red, para tener tiempos de cálculo razonables ($\sim 30\text{min}$) se decidió establecer el tamaño de la red en 15×15 , con la que se tuvo una magnetización y una energía en función del tiempo de ejecución mostrado en la figura 3.1 y 3.2. De estas podemos determinar que para este tamaño de red el $N_{\text{mín}} \approx 6 \times 10^4$, mientras que para asegurar la convergencia a tres cifras significativas se establecieron $N_{\text{máx}} \approx 10^6$ iteraciones de MonteCarlo.

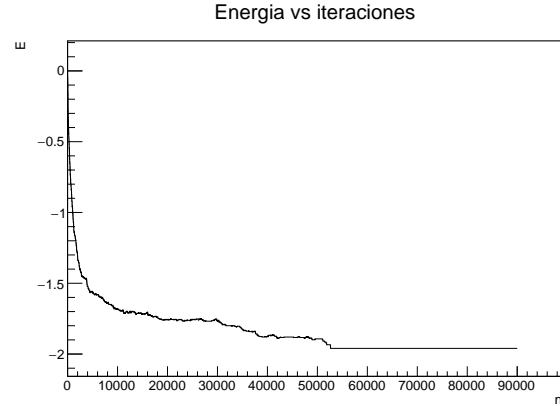


Figura 3.1: Termalización de la energía para una red de 15x15 espines a una temperatura de $0.1 T_c$

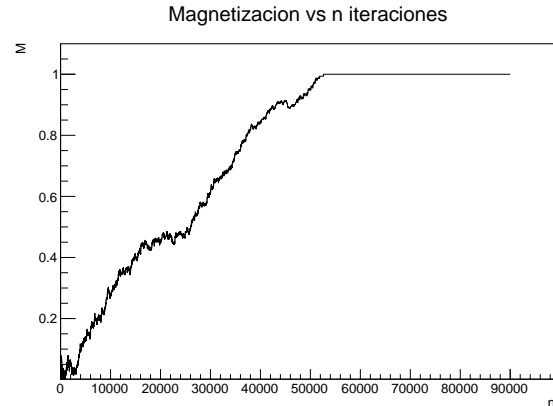


Figura 3.2: Termalización de la Magnetización para una red de 15x15 espines a una temperatura de $0.1 T_c$

Adicionalmente para redes relativamente grandes a bajas temperaturas, aunque la magnetización converge, no parece haber un valor privilegiado sino a un rango de valores, ya que para redes grandes existe la posibilidad de que se formen "islas" de magnetización contraria, como se muestra en la figura 3.3.

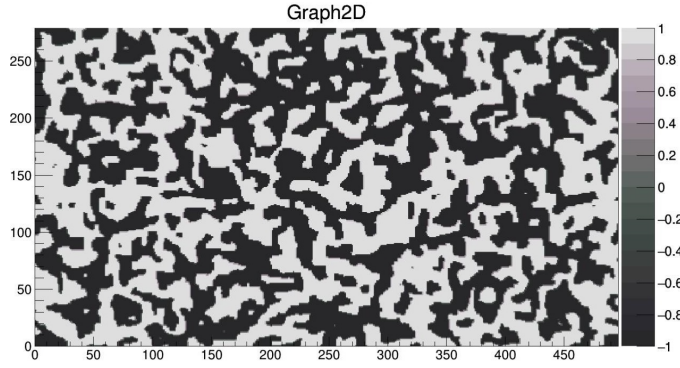


Figura 3.3: Formación de islas para una red de 300x500 a una temperatura de $0.1 T_c$

Este comportamiento extraño tal vez sea depurado con un número suficientemente grande de iteraciones, hace falta más pruebas. Por otra parte no se espera esto, ya que las dos fases son iguales de probables a esa temperatura, entonces puede que coexistan las dos fases, más pruebas podrían revelar más información.

3.2. Barridos de Temperatura

Para una red pequeña de 15×15 espines, la convergencia es inmediata y el barrido de temperaturas revela una transición de fase a aproximadamente $T = 2.6J$ lo cual se esperaba del valor teórico.

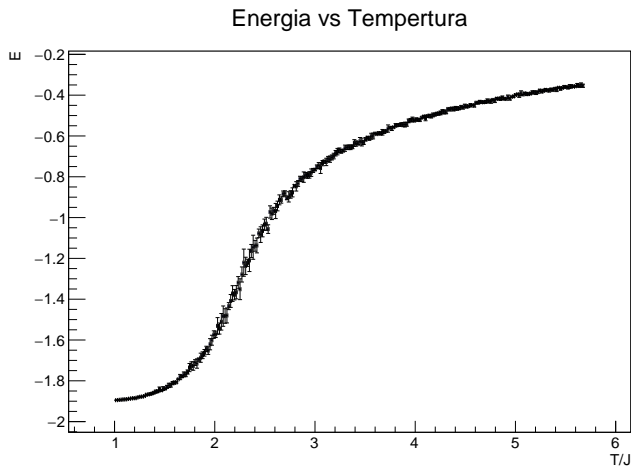


Figura 3.4: Termalización de la Energía para una red de 15×15 del que se registraron $150 \times (10^6 - 6 \times 10^4)$ pasos de montecarlo para cada temperatura entre $[1, 6]J$.

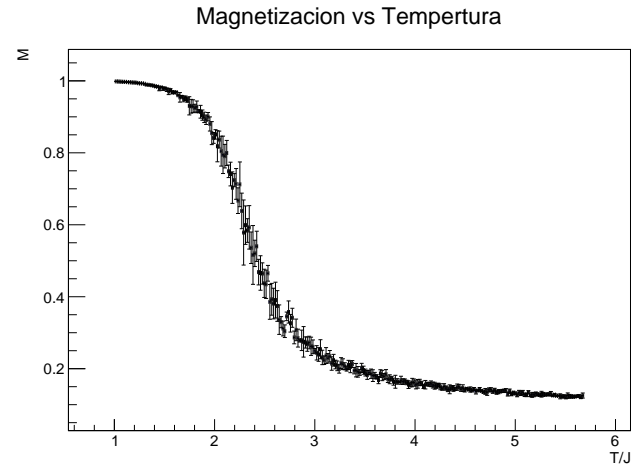


Figura 3.5: Termalización de la Magnetización para una red de 15×15 del que se registraron $150 \times (10^6 - 6 \times 10^4)$ pasos de montecarlo para cada temperatura entre $[1, 6]J$.

3.2.1. Red de mayor tamaño

Para redes de mayor tamaño, como se mencionó, a bajas temperaturas presenta una metaestabilidad donde la magnetización converge a valores no esperados, en la figura 3.6 se ve cómo la magnetización converge a un valor de 0.4, en otras muestras, esta magnetización converge a valores diferentes como se puede evidenciar en las barras de error a bajas temperaturas de la figura 3.7, mientras que la figura 3.8 muestra que estas configuraciones meta estables son de un rango energético similar, por tanto son prácticamente igual de probables que las de magnetización no nula, lo que apoya la hipótesis de que estas configuraciones se mantendrán en el tiempo.

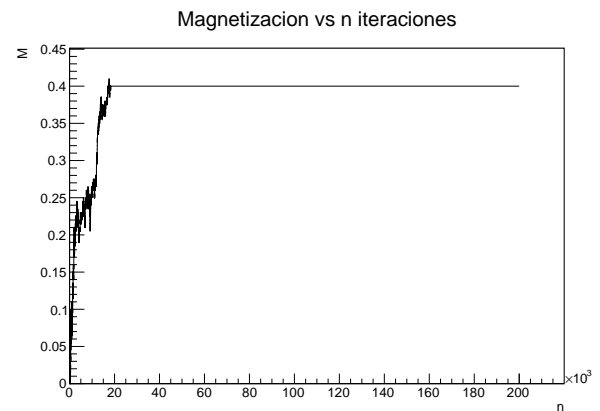


Figura 3.6: Termalización de la Magnetización para una red de 100×100 espines a una temperatura de $0.1 T_c$

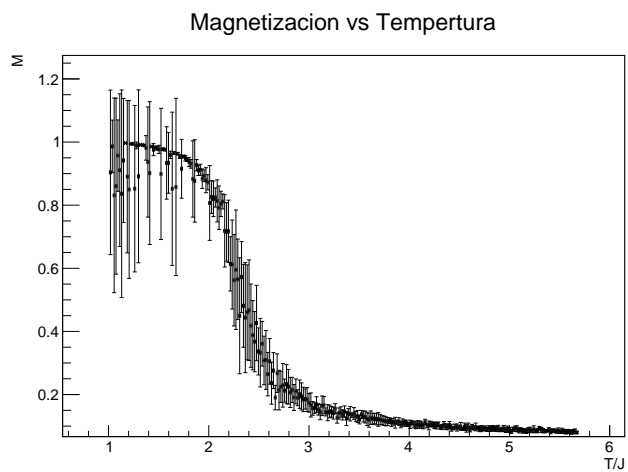


Figura 3.7: Termalización de la Magnetización para una red de 100×100 del que se registraron $150 \times (10^6 - 6 \times 10^4)$ pasos de monte-carlo para cada temperatura entre $[1, 6]J$.

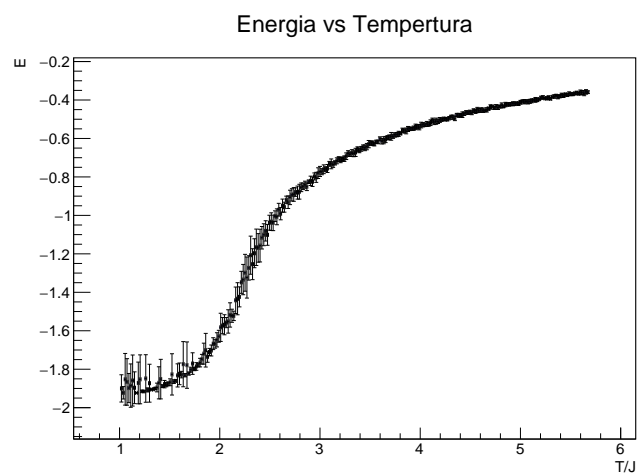


Figura 3.8: Termalización de la Energía para una red de 100×100 del que se registraron $150 \times (10^6 - 6 \times 10^4)$ pasos de monte-carlo para cada temperatura entre $[1, 6]J$.