

Transiciones de fase de segundo orden y rompimiento espontáneo de simetría

Aplicación de la teoría de Landau-Ginzburg al modelo de Ising en 2D con spines discretos

Leonel Fernando Ardila Peña, Juan Sebastián Florez Jiménez

May 30, 2017

Abstract

En este trabajo se presentan los resultados obtenidos a partir de la simulación del modelo de Ising, producto de ello se observa que existe una transición de fase de segundo orden al alcanzar una temperatura crítica T_c y adicionalmente que las simetrías que identificaban el sistema para temperaturas ($T > T_c$) se rompen cuando el sistema adopta una temperatura menor ($T < T_c$).

1 Introducción

Las transiciones de fase corresponden a cambios en las propiedades macroscópicas de un sistema de partículas. Tales cambios se manifiestan como discontinuidades en las funciones de respuesta del sistema, lo cual implica que están relacionadas con la energía libre del sistema termodinámico, y se caracterizan por la presencia de singularidades en esta. Es por esto que el estudio de las transiciones de fase consiste en encontrar el origen de tales singularidades y caracterizarlas. Ya que la función de partición canónica de un sistema finito de partículas es una función analítica, las transiciones de fase se pueden asociar únicamente a sistemas en el límite termodinámico i.e. **partículas infinitas**.

Las transiciones de fase se clasifican según cuál derivada de la energía libre es discontinua:

- **Primer orden** La primera derivada de la energía es discontinua (el parámetro de orden es discontinuo)
- **Segundo orden** La segunda derivada de la energía es discontinua (parámetro de orden continuo y susceptibilidad discontinua)

De forma similar se pueden definir transiciones de fase de tercer orden, cuarto orden, etc.

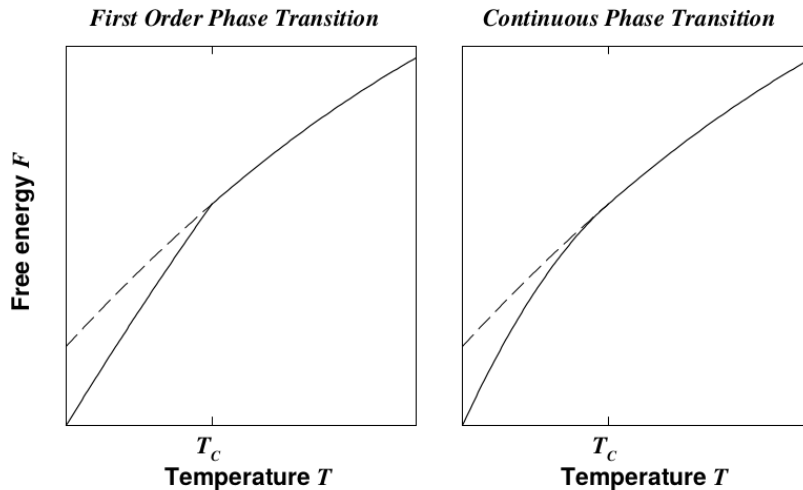


Figure 1: Izquierda: transición de fase de primer orden. Derecha: transición de fase de segundo orden.[Olm00]

Para caracterizar cada fase se utiliza un parámetro de orden, el cual toma diferentes valores en cada fase. Es común que el parámetro de orden sea nulo en la fase con mayor simetría, y no nulo en la fase con menor simetría. Adicionalmente existe un punto crítico donde se produce la transición de fase. Al rededor del punto crítico el comportamiento del sistema está caracterizado por un conjunto de exponentes críticos, los cuales describen la no-analiticidad de varias funciones termodinámicas.

Las funciones termodinámicas siguen una ley de potencias cerca del punto crítico. En el caso de una transición de fase de segundo orden de un ensamble ferromagnetico el parametro de orden es la magnetización promedio por sitio. Sea $t = (T - T_c)/T_c$ un parámetro que mide la distancia al punto crítico, los exponentes críticos toman los siguientes valores en cada fase, i.e. $T > T_c \rightarrow \alpha_+, \beta_+, \gamma_+, T < T_c \rightarrow \alpha_-, \beta_-, \gamma_-$:

$$m(T, H \rightarrow 0) \propto 0, \quad T > T_c, \quad |t|^\beta \quad T < T_c, \quad || \chi_\pm(T, H \rightarrow 0) = \frac{\partial m}{\partial H}|_{H=0+} \propto |t|^{-\gamma_\pm}$$

$$C_\pm = \frac{\partial E}{\partial T} \propto |t|^{\alpha_\pm}$$

Figure 2: Comparación de las exponentes críticos de la teoría de Landau (campo medio) y resultados experimentales en medios tridimensionales[Sim97]

Transition type	Material	α	β	γ	ν
		$C \sim t ^{-\alpha}$	$\langle m \rangle \sim t ^\beta$	$\chi \sim t ^{-\gamma}$	$\xi \sim t ^{-\nu}$
Ferromag. ($n = 3$)	Fe, Ni	-0.1	0.34	1.4	0.7
Superfluid ($n = 2$)	He ⁴	0	0.3	1.3	0.7
Liquid-gas ($n = 1$)	CO ₂ , Xe	0.11	0.32	1.24	0.63
Superconductors		0	1/2	1	1/2
Mean-field		0	1/2	1	1/2

2 Teoria de Landau-Ginzburg

En su teoría Landau hace una serie de aproximaciones para determinar la energía libre de un sistema en una forma tal en la que pueda evidenciarse las discontinuidades que dan cuentas de las transiciones de fase. Para este procedimiento Landau definio cuatro pasos esenciales para el análisis:

1. Definir un parámetro de orden η que da cuenta de la evolución y cambios en el sistema, este parámetro cambiará abruptamente cuando ocurra una transición de fase.
2. Asumir un funcional de energía libre de la forma $\tilde{F} = F_O(T) + F_L(T, \eta)$, donde $F_O(T)$ es una función analítica y $F_L(T, \eta)$ es un funcional que posee la información de las singularidades.
3. Construir $F_L(T, \eta)$ Por lo general se asume como una expansión polinomial que satisfase todas las simetrías asociadas con el parámetro de orden (por lo general invarianzas bajo traslación y rotación).
4. Asumir que al expandir el funcional en los términos de menor orden se encuentran las temperaturas en las que se presentan las discontinuidades.

Al determinar la Función mínima del funcional \tilde{F} con respecto a la temperatura, se puede definir el comportamiento del sistema en la transición de fase.

El abordaje desarrollado por Landau para estudiar las transiciones de fase es fenomenológico, ya que lidia con variables macroscópicas del sistema y no incluye explícitamente las interacciones

microcópicas. Esta es una teoría de campo medio, por que el parámetro de orden se trata como un campo continuo, tal que la variación de este campo es despreciable a escalas comparables con las distancias entre las partículas constituyentes.

2.1 Expansión de la energía libre

El parámetro de orden satisfase que:

- η es cero para fases con alta simetría.
- η no es cero para fases con baja simetría.
- Para una transición de fase de segundo orden η es una función continua de la temperatura.

A partir de la teoría de Landau:

$$F(T, \eta) = F_0 + \alpha\eta + A\eta^2 + C\eta^3 + B\eta^4 + \dots \quad (1)$$

Donde α, A, C, B son coeficientes que dependen de la temperatura.

Para minimizar la función:

$$\frac{\partial F(T, \eta)}{\partial \eta} = \alpha + 2A\eta + 3C\eta^2 + 4B\eta^3 + \dots = 0 \quad (2)$$

- $T > T_c$: $\alpha = 0$, $A \geq 0$, porque $\eta = 0$ debe ser un mínimo
- $T < T_c$: $\alpha = 0$, $A \leq 0$ por razones de simetría.

Por lo tanto, $\alpha = 0$ y $A = 0$

La primera y segunda derivada de la energía libre es entonces:

$$\frac{\partial F(T, \eta)}{\partial \eta} = 3C\eta^2 + 4B\eta^3 \quad \frac{\partial^2 F(T, \eta)}{\partial \eta^2} = 6C\eta + 12B\eta^2 \quad (3)$$

Para que el punto crítico sea estable se debe garantizar que la tercera derivada es cero y la cuarta derivada es positiva.

Como A no presenta singularidades en el punto de transición esta puede ser expandida alrededor de T_c como $A(T) = a(T - T_c)$

Finalmente la energía libre a partir de las consideraciones de Landau está dada por:

$$F(T, \eta) = F_0(T) + \alpha(T - T_c)\eta^2 + B\eta^4 \quad (4)$$

La primera derivada se hace nula cuando:

$$\eta = \pm \sqrt{\frac{a}{2B}(T_c - T)} \quad (5)$$

Cuando el sistema adopta el valor, se genera un rompimiento de la simetría del sistema. La teoría de campo medio de Landau predice que $\beta = 1/2$ independientemente del sistema. La información de las características de las interacciones entre las partículas contituyentes está dentro de las constantes de la expansión de la energía libre.

3 Rompimiento espontaneo de simetría en una transición de fase de segundo orden

Una simetría es una transformación bajo la cual un objeto permanece inalterado. En la naturaleza diversos sistema físico presenta simetrías en su estructura, un claro ejemplo son los cristales, que por lo general presentan simetría traslacional y rotacional. Además de las simetrías espaciales los sistemas también presentan simetrías otras cantidades como en su energía (simetrías en el hamiltoniano).

Cuando la temperatura de un sistema sobrepasa cierto valor (Temperatura crítica) la organización del sistema cambia, las simetrías que lo identifican se modifican al producirse una transición de

fase debido a que las transiciones de fase de segundo orden ocurren cuando un nuevo estado de simetría reducida se genera desde el estado más desordenado (alta temperatura), las transiciones de fase de segundo orden son responsables del rompimiento espontáneo en la simetría del sistema. Esta anomalía del cambio en la simetría se evidencian en particular en sistemas ferromagnéticos, superfluidos y superconductores.

Estos cambios en la simetría se observan al analizar el comportamiento del parámetro de orden que por lo general cambia de forma abrupta cuando sucede la transición de fase.

4 Modelo de Ising

El modelo de Ising es un sistema físico para estudiar materiales ferromagnéticos en el que se puede evidenciar una transición de fase de segundo orden. El modelo consiste de un conglomerado de spines ordenados en una red rectangular cuya energía está definida a partir de las interacciones que cada spin tiene con sus vecinos:

$$H = \sum_{\langle i,j \rangle} -JS_i S_j \quad (6)$$

- $\sum_{\langle i,j \rangle}$ denota la suma de la interacción con las partículas vecinas.
- J denota un factor de escala entre interacción de spines.
- $S_{i,j}$ denota la dirección del spin.

El modelo de que se simula consiste de un arreglo en una malla cuadrada con $N \times N$ spines que solamente pueden adoptar dos direcciones ($S_{i,j} = \pm 1$) y con $J > 0$.

Al analizar el hamiltoniano del modelo de Ising se puede observar que este es invariante bajo la transformación $S_a \rightarrow -S_a$.

A partir de este punto se harán los desarrollos necesarios para mostrar que el cambio de la magnetización promedio a cierta temperatura se debe a una transición de fase de segundo orden.

$$Z = \sum_i e^{-H_i/k_B T} \quad (7)$$

La energía libre de Helmholtz está dada por:

$$F = -K_B T \cdot \ln(Z) \quad (8)$$

Se observa al graficar la magnetización como función de la temperatura una transición de fase al alcanzar la temperatura crítica T_c . Esta transición de fase es una transición de fase de segundo orden.

5 Anexos

5.1 Método de Montecarlo aplicado al modelo de Ising en 2D

Se utilizó un muestreo de Montecarlo para simular el comportamiento de un arreglo de spines dispuesto en un lattice cuadrado, el cual corresponde al modelo de Ising discreto en 2D. La constante de acoplamiento entre celdas vecinas se normalizó, i.e. $J = 1.0$, además, se reescalan las unidades para que $K_B = 1.0 \rightarrow \beta = 1/T$. Se utilizó el algoritmo de Wolff para obtener un muestreo de estados del sistema a temperatura T , y se calcularon las variables termodinámicas a partir de los datos obtenidos. El programa fue creado por el profesor Jose Daniel Muñóz de la Universidad Nacional de Colombia como parte del curso "Mecánica estadística" impartido en el segundo semestre del año 2016.

El generador de números aleatorios fue provisto por el profesor Jose Daniel Muñóz y consiste del siguiente código:

```
#include<iostream>
#include<cmath>
using namespace std;

//Constantes del generador aleatorio
class Crandom{
    unsigned long long u,v,w;
public:
    Crandom(unsigned long long j);
    unsigned long long int64();
    double r() {return 5.42101086242752217E-20 * int64();}
    unsigned int int32(){return (unsigned int) int64();};
    double exponencial(float tau);
    double gauss(float mu, float sigma);
};
Crandom::Crandom(unsigned long long j){
    v=4101842887655102017LL; w=1;
    u = j ^ v; int64();
    v = u; int64();
    w = v; int64();
}
unsigned long long Crandom::int64() {
    u = u * 2862933555777941757LL + 7046029254386353087LL;
    v ^= v >> 17; v ^= v << 31; v ^= v >> 8;
    w = 4294957665U*(w & 0xffffffff) + (w >> 32);
    unsigned long long x = u ^ (u << 21); x ^= x >> 35; x ^= x << 4;
    return (x + v) ^ w;
}

double Crandom::exponencial(float tau){
    return -tau*log(r());
}
double Crandom::gauss(float mu, float sigma){
    return sigma*sqrt(-2*log(r()))*cos(2*M_PI*r())+mu;
}
```

Se implementa el algoritmo de Wolff debido a que disminuye la cantidad de pasos de Montecarlo por sitio necesarios para obtener un sistema en equilibrio termodinámico y la cantidad de pasos de Montecarlo necesarios para tomar muestras independientes del ensamble [Lui06]. Se utilizaron las expresiones encontradas por Martha Guerra y Jose Daniel Muñóz para el escalamiento de los tiempos de equilibrio y tiempos de correlación al aplicar el método de Montecarlo al ensamble canónico del modelo de Ising en 2D [Gue02].

Se utiliza el escalamiento de sistemas finitos para obtener el valor de los exponentes críticos. Con el cumulante de Binder de cuarto orden se obtiene la temperatura crítica del modelo de Ising en el límite termodinámico [Bin81]. Adicionalmente, la derivada del cumulante de Binder evaluada en la temperatura crítica permite obtener el exponente crítico asociado a la distancia de correlación del sistema:

$$U_B = 1 - \frac{1}{3} \frac{\langle M^4 \rangle_T}{\langle M^2 \rangle_T^2}, \quad -\frac{\partial U_B}{\partial T} \Big|_{T_c} \propto L^{1/\nu}, \quad \frac{1}{L^d} \langle M \rangle_{T_c} \propto L^{-\beta/\nu}, \quad T_{\chi_{s_{max}}} \propto T_c + aL^{-1/\nu}, \quad \frac{1}{L^d} \chi_s(T_c) \propto L^{\gamma/\nu}$$

- Se estima el punto de corte de las curvas del cumulante de Binder para diferentes tamaños del lattice, tal corte se produce en la temperatura crítica.
- Se calcula el exponente ν realizando un ajuste de la derivada del cumulante de Binder en la temperatura crítica.
- Se calcula el exponente β ajustando la magnetización por sitio en función del tamaño del sistema.
- Se calcula el exponente γ ajustando la susceptibilidad a T_c en función del tamaño del sistema.

Se pueden escalar la magnetización y la susceptibilidad magnética a partir de los exponentes críticos, lo cual produce un colapsamiento de todas las curvas.

```
//Ising 2D por Wolff
#include <cmath>
#include <iostream>
#include "Random64.h"
using namespace std;
```

```
const int L=8;
const int L2=L*L;
const int k=1, J=1;
```

La variable L da cuenta del tamaño del sistema, tal que L^2 es la cantidad de celdas en el sistema.

```
const int teq=(int)(10*pow(L/8.0,0.58));
const int tcorr=(int)(1*pow(L/8.0,0.58));
const double TC=2/log(1+sqrt(2.0)); //TC=2.269;
```

La variable teq da cuenta de la cantidad de pasos de Metropolis por sitio que se deben realizar para obtener termalizar el sistema. La variable tcorr es la cantidad de pasos de Metropolis por sitio que se deben realizar para obtener medidas independientes. La variable TC es la temperatura crítica teórica.

```
//GRUPO1
const double Tmin=2.2;
const double Tmax=2.4;
const double Tstep=(Tmax-Tmin)/20;
const int Nmuestras=1000000;
```

```
//GRUPO2
//const double Tmin=TC-1.3/pow(L/8.0,1.0);
//const double Tmax=TC+3.0/pow(L/8.0,1.0);
//const double Tstep=(Tmax-Tmin)/500;
//const int Nmuestras=10000;
```

Se realizarón dos grupos de simulaciones, el primer grupo se utilizó para encontrar el valor de los exponentes críticos. El segundo grupo se utilizó para escalar la magnetización y la susceptibilidad magnética.

```
class SpinSystem{
private:
    int s[L][L]; int E,M;
```

```

    int Cluster[L2][2]; int write;
public:
    void InicieArriba(void);
    void Muestre(void);
    void MuestreCluster(void);
    bool NoEstaEnElCluster(int i,int j);
    int WolffCluster(Crandom & ran2,double p);
    double GetE(void){return (double)E;};
    double GetM(void){return fabs((double)M);};
};

```

Se crea un clase llamada SpinSystem que se encarga de guardar el estado del sistema:

- $s[L][L]$ guarda el estado del spin, 1 spin arriba y -1 spin abajo.
- $Cluster[L2][2]$ se utiliza para guardar el estado de cada spin al crear el cluster de Wolff, ya que cada spin se debe tener en cuenta una sola vez al crear el cluster.
- E y M guardan la energía interna y la magnetización del sistema.
- La variable write guarda la cantidad de pasos de Montecarlo que se han realizado al crear el cluster.

```

void SpinSystem::InicieArriba(void){
    for(int i=0;i<L;i++){
        for(int j=0;j<L;j++){
            s[i][j]=1;
        }
    }
    M=L2; E=-2*L2;
    write=0;
}

```

Todos los spines inician con estado +1.

```

void SpinSystem::Muestre(void){
    for(int i=0;i<L;i++){
        for(int j=0;j<L;j++){
            cout<<s[i][j]<<" ";
        }
        cout<<endl;
    }
    cout<<"M="<<M<<" "; E="<<E<<endl<<endl;
}

```

La función Muestre imprime el estado del sistema.

```

bool SpinSystem::NoEstaEnElCluster(int i,int j){
    int search; bool NoEsta=true;
    for(search=0;search<write;search++){
        if(Cluster[search][0]==i && Cluster[search][1]==j){
            NoEsta=false; break;
        }
    }
    return NoEsta;
}

```

La función NoEstaEnElCluster determina si el spin en la posición (i,j) ha sido incluido en el cluster.

```

void SpinSystem::MuestreCluster(void){
    int i,j,search; bool NoEsta;
    for(i=0;i<L;i++){
        for(j=0;j<L;j++){
            if(NoEstaEnElCluster(i,j))
                cout<<"0 ";
            else

```

```

        cout<<"1 ";
    }cout<<endl;
}
cout<<"Tamaño del Cluster="<<write<<endl<<endl;
}

```

La función MuestraCluster permite determinar cuáles spines han sido agregados al cluster.

```

int SpinSystem::WolffCluster(CRandom & ran2, double p){
    int n,i,j,south,north,east,west,dE,sref,read;
    //Escojo el primer espín, y lo añado a la lista de leer;
    write=read=0;
    n=(int) (L2*ran2.r()); i=n/L; j=n%L;
    sref=s[i][j]; Cluster[write][0]=i; Cluster[write][1]=j; write++;
    while(read<write){
        //voltarlo
        i=Cluster[read][0]; j=Cluster[read][1];
        south=(i+1)%L; north=(i+L-1)%L; east=(j+1)%L; west=(j+L-1)%L;
        dE=2*s[i][j]*(s[south][j]+s[north][j]+s[i][east]+s[i][west]);
        s[i][j]*=-1; E+=dE; M+=2*s[i][j];
        read++;
        //introducir nuevos en la lista
        //North
        if(s[north][j]==sref) //Si está en la misma dirección,
            if(ran2.r()<p) //y los dados nos favorecen para agregarlo,
                if(NoEstaEnElCluster(north,j)){//y no está en el cluster
                    Cluster[write][0]=north; Cluster[write][1]=j; write++; //agregarlo.
                }
        //South
        if(s[south][j]==sref) //Si está en la misma dirección,
            if(ran2.r()<p) //y los dados nos favorecen para agregarlo,
                if(NoEstaEnElCluster(south,j)){//y no está en el cluster
                    Cluster[write][0]=south; Cluster[write][1]=j; write++; //agregarlo.
                }
        //East
        if(s[i][east]==sref) //Si está en la misma dirección,
            if(ran2.r()<p) //y los dados nos favorecen para agregarlo,
                if(NoEstaEnElCluster(i,east)){//y no está en el cluster
                    Cluster[write][0]=i; Cluster[write][1]=east; write++; //agregarlo.
                }
        //West
        if(s[i][west]==sref) //Si está en la misma dirección,
            if(ran2.r()<p) //y los dados nos favorecen para agregarlo,
                if(NoEstaEnElCluster(i,west)){//y no está en el cluster
                    Cluster[write][0]=i; Cluster[write][1]=west; write++; //agregarlo al cluster
                }
    }
    return write;
}

```

La función WolffCluster crea el cluster de Wolff:

- Se escoge un spin (i,j) al azar.
- Se escogen los primeros vecinos al spin (i,j) y si tienen la misma alineación que la alineación original del spin (i,j) se añaden al cluster con probabilidad $p = 1 - e^{\frac{-2J}{K_B T}}$.
- Se repite el paso anterior con todos los nuevos spines que pertenecen al cluster, teniendo en cuenta que un spin solo puede ser agregado una vez.
- Se cambia el spin todos los spines agregados al cluster, i.e. $s_{i,j} \rightarrow s_{i,j} * (-1)$. Cada spin que se cambie cuenta como un paso de Metrópolis.

La variable write guarda la información de cuantos spines fueron modificados al crear el cluster de Wolff.

```
int main(){
    SpinSystem Ising;
    Crandom ran2(0);
    int mcs,t,ccorr,c;
    double Mprom,M2prom,M4prom,Eprom,E2prom,E,M;
    double Cv,Xs,Ub;
    double T,p;
```

Se crean las variables del sistema:

- E: energía interna del sistema
- M: magnetización total
- Cv: calor específico
- Xs: susceptibilidad magnética
- Ub: cumulante de Binder
- Mprom : $\langle M \rangle$
- M2prom : $\langle M^2 \rangle$
- M4prom : $\langle M^4 \rangle$
- Eprom : $\langle E \rangle$
- E2prom : $\langle E^2 \rangle$

```
//PARA CALCULAR LOS VALORES MEDIOS EN FUNCION DE LA TEMPERATURA
for (T=Tmin;T<Tmax;T+=Tstep){
    p=1-exp(-(2*J)/(k*T));
    //Inicio el sistema
    Ising.InicieArriba();
    //Equilibrio
    for (t=0;t<teq;t++){
        for (mcs=0;mcs<L2;) //Un MCSS
            mcs+=Ising.WolffCluster(ran2,p);
    }
```

Se termaliza el sistema: partiendo de una configuración de mínima energía (todos los spines con la misma dirección) se realizan teq pasos de Montecarlo por sitio para obtener muestras del ensamble deseado.

```
//Inicio Acumuladores en cero
Mprom=M2prom=M4prom=Eprom=E2prom=0;
//Tomo muestras
for (t=0;t<Nmuestras;t++){
    E=Ising.GetE(); M=Ising.GetM();
    Mprom+=M; M2prom+=M*M; M4prom+=M*M*M*M; Eprom+=E; E2prom+=E*E;
    for (ccorr=0;ccorr<tcorr;ccorr++){
        for (mcs=0;mcs<L2;) //Un MCSS
            mcs+=Ising.WolffCluster(ran2,p);
    }
    //Normalizo los acumuladores
    Mprom/=Nmuestras; M2prom/=Nmuestras; M4prom/=Nmuestras;
    Eprom/=Nmuestras; E2prom/=Nmuestras;
```

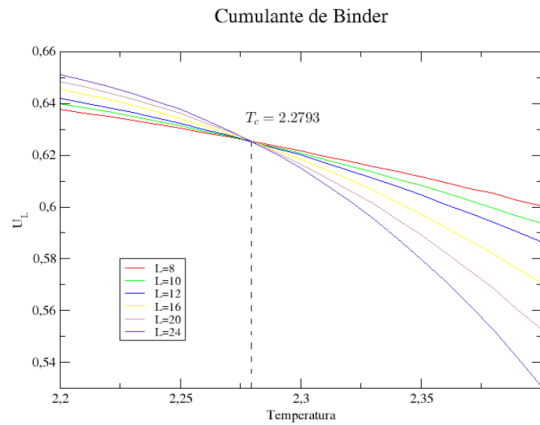
Se registran las variables a partir de Nmuestras independientes pertenecientes al ensamble deseado.

```
//Calculo lo que me interesa
Cv=1/(k*T*T)*(E2prom-Eprom*Eprom);
Xs=1/(k*T)*(M2prom-Mprom*Mprom);
Ub=1.0-1.0/3.0*(M4prom/(M2prom*M2prom));
//Imprimo
cout<<T<<" "<<Eprom<<" "<<Mprom<<" "<<Cv<<" "<<Xs<<" "<<Ub<<" "<<M2prom<<endl;
}
return 0;
}
```

Se calculan la capacidad calorífica, la susceptibilidad magnética y el cumulante de Binder de cuarto orden a partir de los datos registrados. No se calcula el error en las variables calculadas; para ello se deberían guardar los datos de la energía interna y la magnetización del sistema en todos los estados independientes generados.

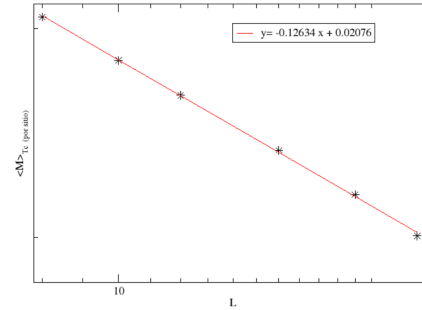
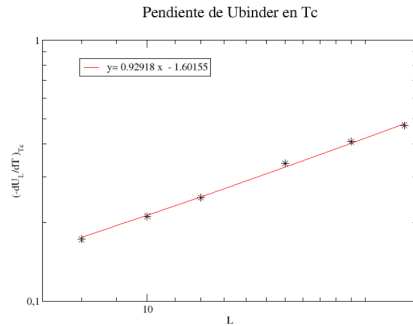
5.2 Reescalamiento aplicado al modelo de Ising

El reescalamiento permite obtener información del sistema en el límite termodinámico a partir de sistemas finitos. Para ello se utiliza el cumulante de Binder, el cual depende poco del tamaño del sistema si se evalúa cerca de la temperatura crítica [Bin81].



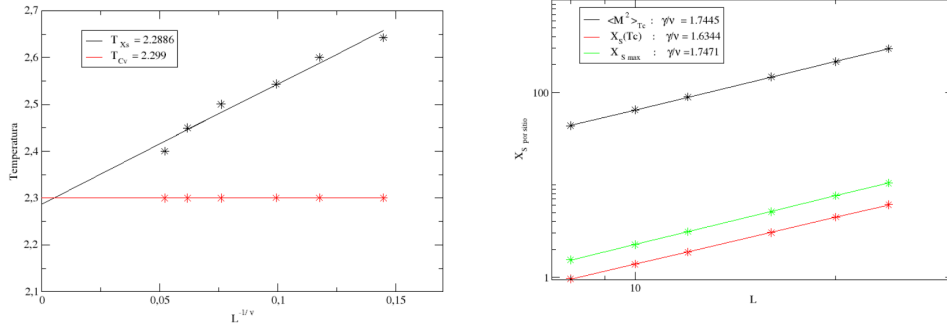
Se obtuvo un valor para la temperatura crítica de $T_c = 2.279(5)$, el cual coincide con el valor teórico de la temperatura crítica dentro de dos desviaciones estándar $T_c \approx 2.269$. Se realizaron 6 simulaciones con arreglos cuadrados de lado $L=8, 10, 12, 16, 20, 24$.

Se procedió a calcular la pendiente del cumulante de Binder en la temperatura crítica, lo cual permitió encontrar el valor de $\nu = 1.076(5)$. Conociendo el valor de ν se puede calcular el valor de β a partir de la magnetización promedio.



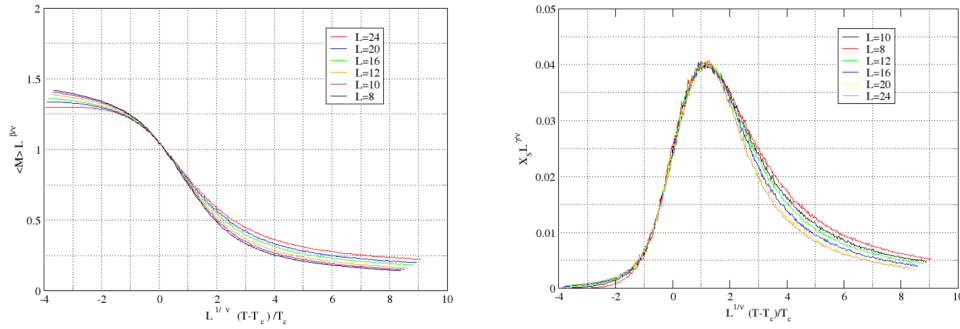
(a) Derivada del cumulante de Binder en T_c en función del tamaño del sistema. (b) Magnetización promedio en función del tamaño del sistema.

El valor calculado para el exponente β es $\beta = 0.1359(4)$. Se puede obtener el valor del exponente crítico γ a partir de la susceptibilidad magnética o del promedio del cuadrado de la magnetización. Se obtiene que $\gamma = 1.878(5)$, con lo cual se tiene los parámetros necesarios para reescalar la magnetización y la susceptibilidad magnética.



(a) Máximos de la capacidad calorífica y la susceptibilidad magnética en función del tamaño del sistema. (b) Máximo de la susceptibilidad magnética, susceptibilidad magnética evaluada en T_c y promedio del cuadrado de la magnetización en función del tamaño del sistema.

Sea $t = (T - T_c)/T_c$ una variable adimensional que indica la distancia entre el sistema y el punto crítico, si se grafica $\langle M \rangle L^{\beta/\nu}$ en función de $tL^{1/\nu}$ todas las gráficas conciden para $t = 0$. Esto se produce porque el sistema cerca del punto crítico sigue una ley de potencias que aparece debido a la divergencia de la distancia de correlación. La divergencia de la distancia de correlación hace que el sistema sea muy sensible a las perturbaciones, y se haga invariante ante un cambio de escala. De forma similar, si se grafica $\xi_s L^{-\gamma/\nu}$ en función de $tL^{1/\nu}$ las curvas de las seis simulaciones convergen para $t = 0$.



(a) Colapso de las curvas de magnetización tras aplicar el reescalamiento. (b) Colapso de las curvas de susceptibilidad tras aplicar el reescalamiento.

En el límite termodinámico se generan discontinuidades en la susceptibilidad magnética debido a la presencia de infinitos grados de libertad. Se puede observar que la susceptibilidad magnética diverge en el punto crítico si se toma el sistema en el límite termodinámico.

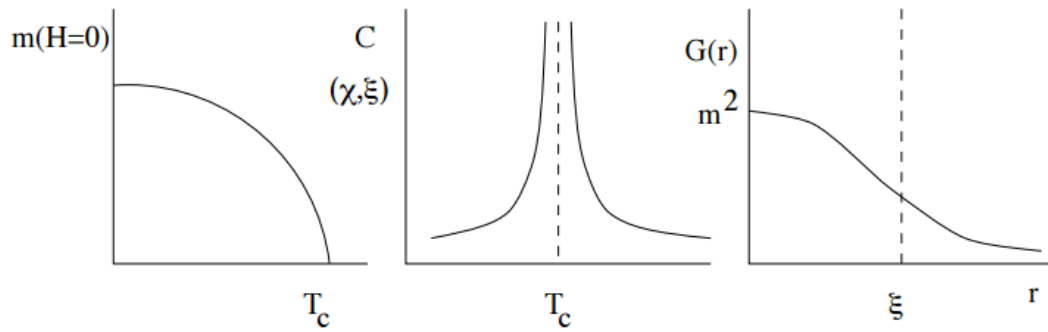


Figure 6: Esquema del comportamiento de las variables macroscópicas del sistema en el límite termodinámico. La susceptibilidad y distancia de correlación divergen en el punto crítico[Sim97].

El hamiltoniano del sistema es invariante al cambiar los spines de sentido i.e. $\sigma_{i,j} \rightarrow -\sigma_{i,j}$. Sin

embargo, cuando $T < T_c$ una inversión en el signo de los spines genera un cambio de signo en la magnetización promedio, lo cual implica que el sistema ya no es simétrico ante una transformación del tipo $\sigma_{i,j} \rightarrow -\sigma_{i,j}$.

References

- [Bin81] Kurt Binder. Finite size scaling analysis of ising model block distribution functions. *Z. Phys. B - Condensed Matter*, 43:119–140, 1981.
- [Gue02] Martha Guerra. Equilibrium times for the multicanonical method. *Departamento de Física, Universidad Nacional de Colombia*, 2002.
- [Lui06] Erik Luijten. Introduction to cluster monte carlo algorithms. *Department of Materials Science and Engineering, Frederick Seitz Materials*, 2006.
- [Olm00] Peter D Olmsted. Lectures on landau theory of phase transitions. *University of Leeds, Department of Physics and Astronomy*, 2000.
- [Sim97] Ben Simons. Phase transitions and collective phenomena. *University of Cambridge, Theory of condensed matter*, 1997.