

Simulación del modelo de Ising con el método de Monte Carlo

Angélica L. Gelover-Santiago

$$H = -J \sum S_i S_j - h \sum S_i$$

Temas de Física



Simulación del modelo de Ising con el método de Monte Carlo

ANGÉLICA L. GELOVER-SANTIAGO

FACULTAD DE CIENCIAS, UNAM



Simulación del modelo de Ising con el método de Monte Carlo

1a. edición, 1990

2a. edición, 2005

- © Coordinación de Servicios Editoriales,
Facultad de Ciencias, UNAM
Diseño de portada: Laura Uribe
Edición: Pablo Rosell y Héctor Miguel Cejudo Camacho

ISBN: 970-32-2869-0

Impreso y hecho en México

AGRADECIMIENTOS

Deseo expresar mi agradecimiento a las siguientes personas, quienes de una u otra manera tuvieron alguna participación en el presente trabajo. A los Dres. Raúl M. Rechtman Scherenzel, Gustavo C. Martínez Mekler y Jorge S. Lomnitz Adler[†] por su valioso apoyo en la elaboración de la primera versión de este trabajo. Al Dr. Gerardo Carmona Ruíz por haberme animado a trabajar en esta segunda edición. Al Dr. Maximino Aldana González por su sugerencia de utilizar la subrutina para generar números pseudoaleatorios propuesta por Press et. al. A Ma. del Consuelo Escobedo Lozano por sus observaciones sobre la primera edición. Al Dr. Manuel G. Corona Galindo por las fructíferas discusiones que mantuvimos a lo largo de los meses que duró la preparación de esta segunda edición así como por la lectura de las diversas versiones preliminares del manuscrito. Al Dr. Dieter Ebner de la Universidad de Konstanz-Alemania por su ayuda con algunos aspectos técnicos de L^AT_EX. A los árbitros, cuyos acertados comentarios contribuyeron a mejorar el contenido. Finalmente, estos agradecimientos no estarían completos si no mencionara la amable hospitalidad, tanto del Instituto de Química de la UNAM, como del Departamento de Física de la Universidad de Konstanz-Alemania, especialmente del Prof. Dr. Heinz Dehnen, en donde se realizaron algunas etapas de este trabajo. Así mismo, agradezco al CONACYT y al DAAD (Deutscher Akademischer Austauschdienst) el apoyo económico otorgado para realizar la estancia de investigación en la Universidad de Konstanz.

CONTENIDO

| | | |
|----------|--|-----------|
| 1 | Introducción | 1 |
| 2 | Transiciones de fase y fenómenos críticos | 5 |
| 3 | Magnetismo y el modelo de Ising | 15 |
| 4 | Método de Monte Carlo | 23 |
| 5 | Descripción del programa de computadora | 31 |
| 6 | Resultados | 39 |
| | Bibliografía | 49 |
| | Apéndices | 54 |
| A | Programa Ising.c | 55 |
| B | Subrutina alea1.h | 85 |

INTRODUCCIÓN

La mecánica estadística tiene por objeto el estudio de sistemas formados por muchas partículas, tales como sólidos, líquidos, gases, etc. La tarea de comprender el comportamiento de estos sistemas dista mucho de ser trivial debido a su complejidad, sobre todo cuando existen interacciones apreciables entre las partículas que los forman. Por ejemplo, un hecho difícil de comprender es el que los átomos que forman un gas puedan condensarse bruscamente para formar un líquido con propiedades muy diferentes. La razón de la brusquedad de este tipo de transiciones es que las interacciones entre las partículas pueden dar lugar a comportamientos cooperativos. Estos comportamientos cooperativos producen correlaciones entre todos los elementos que constituyen al sistema, haciendo que éste se vuelva difícil de estudiar teóricamente.

En mecánica estadística se considera que un problema está resuelto cuando se ha evaluado su función de partición exacta, Z , ya que ésta contiene toda la información referente al sistema, incluyendo la presencia de transiciones de fase. El objetivo es por lo tanto, evaluar la función de partición ya que entonces pueden calcularse los valores esperados de las cantidades observables del sistema como energía, temperatura, magnetización o calor específico. Sin embargo, el problema teórico está precisamente en la dificultad para evaluar la función de partición correspondiente a sistemas de muchas partículas en donde las interacciones entre ellas son apreciables. Por sistemas con interacciones apreciables debemos entender sistemas en los que cada partícula no pueda considerarse independiente de las

demás. Cuando no es posible evaluar la función de partición en forma analítica, es necesario utilizar algún método numérico entre los que el Monte Carlo es uno de los más usados.

El objetivo del presente trabajo es proporcionar una presentación de carácter didáctico de cómo se puede estudiar el comportamiento crítico de un sistema utilizando el método de Monte Carlo. El sistema que consideraremos es el modelo de Ising ya que ha sido ampliamente estudiado y se conocen resultados exactos para los casos unidimensional, con y sin campo magnético externo, y bidimensional en ausencia de campo magnético externo. Sin embargo, es importante señalar que, hasta la fecha, el modelo en tres dimensiones no ha sido resuelto ni aún en ausencia de campo magnético externo.

Para lograr nuestro objetivo, se implementó un programa de cómputo con el cual se simula el modelo de Ising utilizando el método de Monte Carlo y se calculan algunas de sus propiedades termodinámicas. El programa corre en una computadora personal compatible con el sistema IBM-PC y está basado en el trabajo de S. E. Koonin [Koonin, S. E., 1985]. En la primera edición del presente trabajo, el programa original escrito en BASIC y diseñado para simular el modelo de Ising en los casos ferro y antiferromagnético, se tradujo al lenguaje PASCAL [Gelover-Santiago, A. L., 1990]. En esta segunda edición, el programa está escrito en lenguaje C, utilizando el paquete de software Turbo C++ versión 1.00 de Borland International Inc., y gracias al desarrollo de computadoras cada vez más potentes, permite simular una red de 150x150 espines en lugar de la de 30x30 que se manejó en la primera edición. Para visualizar la evolución del sistema es recomendable que la computadora cuente con monitor a color.

El trabajo está estructurado de la siguiente manera: en la Sección 2 se hace una presentación breve sobre transiciones de fase y fenómenos críticos así como del lenguaje que se usa para describirlos. La Sección 3 está dedicada a los fenómenos magnéticos y su origen, así como a la presentación del modelo de Ising. En la Sección 4 se introduce el método de Monte Carlo

INTRODUCCIÓN

y se discute su aplicación dentro de la mecánica estadística para calcular promedios de las observables de un sistema. La Sección **5** contiene una descripción del programa que se implementó. En la Sección **6** se exponen los resultados obtenidos en cada uno de los casos estudiados: ferromagnetismo con y sin campo magnético externo; antiferromagnetismo sin campo magnético externo; ferromagnetismo anisotrópico sin campo magnético externo y, por último, un sistema en el que se observa el fenómeno de frustración. Finalmente, en el Apéndice **A** se incluye el programa *Ising.c* y en el Apéndice **B** la subrutina *alea1.h*, utilizada para generar números pseudoaleatorios.

Es importante mencionar que aparte de los resultados numéricos que se obtienen con el programa, es posible también entender a través de éste algunos aspectos de la fenomenología tanto de las transiciones de fase como de los fenómenos críticos y por lo tanto puede ser de gran utilidad en la enseñanza de la mecánica estadística, tanto a nivel licenciatura como a nivel posgrado.

TRANSICIONES DE FASE Y FENÓMENOS CRÍTICOS

Las sustancias pueden hallarse en una gran variedad de estados, cada uno de los cuales está caracterizado por los valores de sus variables termodinámicas. Cada estado presenta ciertas propiedades macroscópicas que se conservan al evolucionar las variables termodinámicas, pero existen intervalos muy estrechos de estas variables en los que el sistema cambia notablemente sus propiedades, se dice entonces que el sistema experimenta una transición de fase. Los ejemplos más familiares de transiciones de fase los constituyen las transiciones líquido-gas, sólido-gas y sólido-líquido de un fluido clásico pero hay una gran variedad de éstas, aunque no tan familiares, como son las ferromagnéticas, ferroeléctricas, fluido-superfluido y conductor-superconductor, entre otras.

Las primeras teorías que surgieron en el estudio de las transiciones de fase son las llamadas clásicas o de campo medio, como la teoría de van der Waals de las transiciones de fase líquido-gas (1873), la teoría de Landau de las transiciones de fase que él denomina continuas (1937) y la teoría de Curie-Weiss de las transiciones de fase magnéticas (1948) [Chourhury, D., et. al., 2000], [Huang, K., 1987], [Landau, L. D., 1985], [Pathria, R. K., 1993], [Reichl, L. E., 1980], [Stanley, H. E., 1971]. El nombre de teorías de campo medio se debe a que pueden derivarse suponiendo que cada partícula interactúa con el campo producido por todas las demás en lugar de considerar interacciones individuales entre las partículas^{*}. Desde el surgimiento de es-

^{*} En inglés, el término que se utiliza para referirse al campo con el

tas teorías, se ha dado un desarrollo sistemático en el área y se han consolidado teorías muy completas que han permitido el entendimiento de estos fenómenos con relativo éxito. Expondremos aquí las principales características de las transiciones de fase y el lenguaje que se utiliza para su estudio, prestando especial atención a la región crítica de éstas.

Consideremos el ejemplo de un fluido, cuyo diagrama de fases se muestra en la Figura 2.1. Cuando el líquido se evapora, a lo largo de la curva de coexistencia líquido-gas se observan dos comportamientos termodinámicos distintos. En todos los puntos de la curva, excepto en el punto terminal que se conoce como *punto crítico*, se observan simultáneamente un calor latente así como una discontinuidad en la densidad. En el punto crítico, por el contrario, se pasa continuamente de la fase líquida a la fase gaseosa y no se observa un calor latente ni una discontinuidad en la densidad. En el caso de la transición de fase sólido-líquido la curva de coexistencia no exhibe un punto terminal y la transición siempre se presenta con un calor latente. Entonces, desde el punto de vista termodinámico, es posible distinguir dos tipos de transiciones de fase: aquellas que se presentan con un calor latente y aquellas en las que éste no se observa [Papon, P., et. al., 2002].

La primera clasificación de las transiciones de fase fue propuesta por P. Ehrenfest en 1933, la cual está basada en los potenciales termodinámicos [Ehrenfest, P., 1933], [Papon, P., et. al., 2002], [Pathria, R. K., 1993], [Reichl, L. E., 1980]. De acuerdo con esta clasificación, son transiciones de fase de primer orden aquellas que están acompañadas por discontinuidades de cantidades termodinámicas correspondientes a derivadas de primer orden de algún potencial termodinámico. Ejemplos de derivadas de primer orden del potencial de Gibbs, G , son:

cual interactúa cada partícula es *mean field*, es decir campo promedio. En español es común referirse a éste como *campo medio*.

TRANSICIONES DE FASE Y FENÓMENOS CRÍTICOS

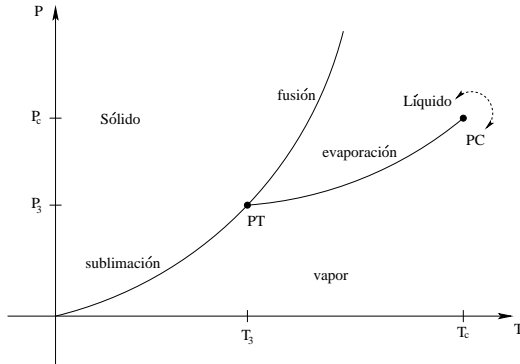


Figura 2.1 Diagrama de fases correspondiente a un fluido. Además de las curvas de coexistencia entre las fases, se muestra el punto triple (PT), que es el único punto en donde coexisten las tres fases, y el punto crítico (PC), que es el punto terminal de la curva de coexistencia líquido-gas. (T_3, P_3) son los valores de la Temperatura y la Presión en el punto triple y (T_c, P_c) los valores de estas mismas variables correspondientes al punto crítico

$$S = -\left(\frac{\partial G}{\partial T}\right)_P \quad V = \left(\frac{\partial G}{\partial P}\right)_T \quad H = -\left(\frac{\partial(G/T)}{\partial(1/T)}\right)_P,$$

en donde S es la entropía, V el volumen, T la temperatura, P la presión y H la entalpía. En la transición de fase estas cantidades, correspondientes a derivadas de primer orden del potencial de Gibbs, son discontinuas. En particular, el calor latente está asociado con una discontinuidad en la entropía. Por otra parte, son transiciones de fase de segundo orden aquellas para las cuales, los potenciales termodinámicos así como sus derivadas de primer orden son continuos, mientras que alguna de sus derivadas de segundo orden es discontinua en el punto de transición. Ejemplos de derivadas de segundo orden son:

$$\frac{C_P}{T} = -\left(\frac{\partial^2 G}{\partial T^2}\right)_P = \left(\frac{\partial S}{\partial T}\right)_P \quad \kappa_T V = -\left(\frac{\partial^2 G}{\partial P^2}\right)_T = -\left(\frac{\partial V}{\partial P}\right)_T,$$

en donde C_P y κ_T representan el calor específico a presión constante y la compresibilidad a temperatura constante, respectivamente. Para estas transiciones, el paso de una fase a la otra es continuo, por lo que no es posible hablar de una coexistencia de las dos fases; la fase líquida no puede distinguirse de la fase gaseosa debido a que sus densidades son estrictamente iguales. Las situaciones correspondientes a cada una de estas dos categorías de transiciones están representadas de manera simple en la Figura 2.2.

Con base en el esquema de Ehrenfest, se habla de transiciones de fase de orden superior al segundo, en cuyo caso el orden de la transición sería igual al orden de la derivada que es discontinua [Papon, P., et. al., 2002], [Reichl, L. E., 1980] sin embargo, es común aplicar el calificativo de *segundo orden* a todas las transiciones que no son de primer orden [Chourhury, D., et. al., 2000].

Es importante señalar que, si bien la clasificación de Ehrenfest de las transiciones de fase tiene el gran mérito de revelar las similitudes existentes entre fenómenos tan diferentes como el magnetismo, la ferroelectricidad, la superconductividad y la transición líquido-vapor en el punto crítico, ésta no es correcta. Por ejemplo, de acuerdo con esta clasificación, en la transición de fase ferromagnética debería observarse una discontinuidad finita en la capacidad calorífica. En un ferromagneto real, la capacidad calorífica diverge hacia infinito (Figura 2.3).

En 1937, L. D. Landau observó que una transición de fase sin calor latente está acompañada de un cambio de simetría del sistema. La transición líquido-vapor en el punto crítico, es una excepción a este comportamiento. Para entender mejor el concepto de simetría, consideremos el caso de un material magnético, el cual a temperaturas altas y en ausencia de campo magnético externo no exhibe una magnetización permanente mientras que por abajo de la llamada *temperatura de Curie* sí la exhibe, ver Figura 2.4. En este caso, la transición tiene lugar precisamente en la temperatura de Curie y está acompañada por una reducción de simetría del sistema, pues por debajo de

TRANSICIONES DE FASE Y FENÓMENOS CRÍTICOS

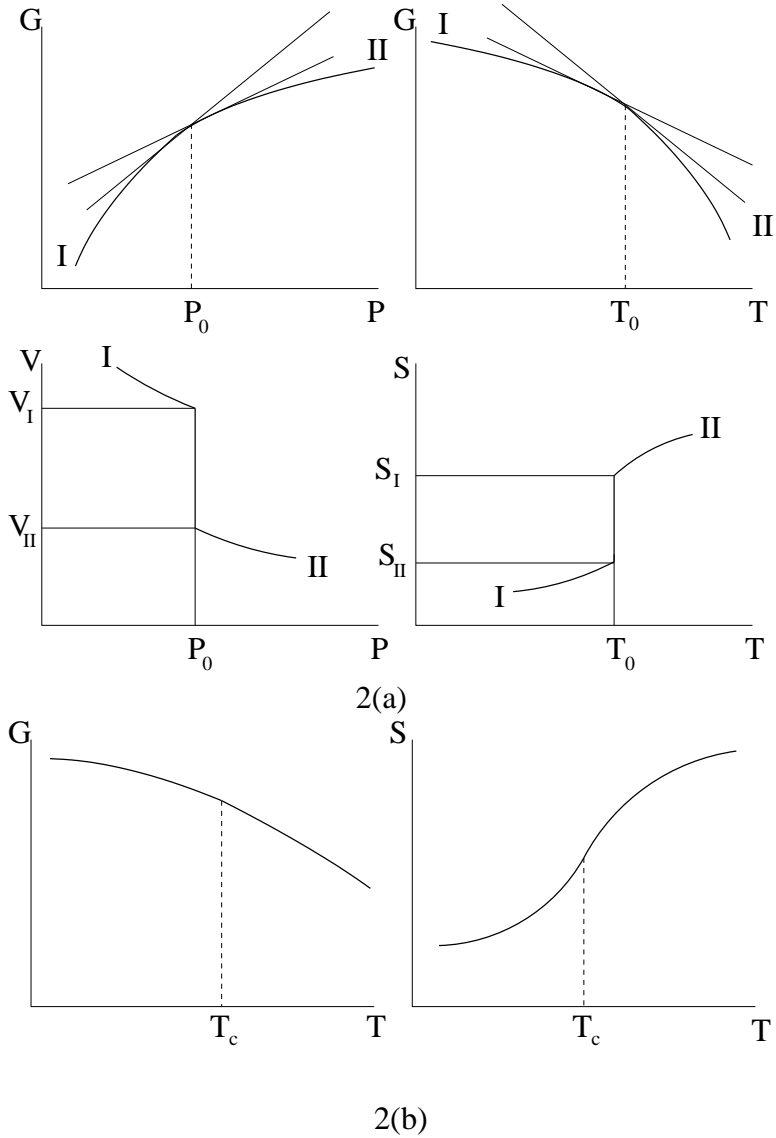


Figura 2.2 (a) Para una transición de fase de primer orden, el potencial de Gibbs, G , es continuo pero las primeras derivadas, tales como V y S son discontinuas. (b) Para una transición de fase de segundo orden, tanto el potencial de Gibbs como sus primeras derivadas son continuas.

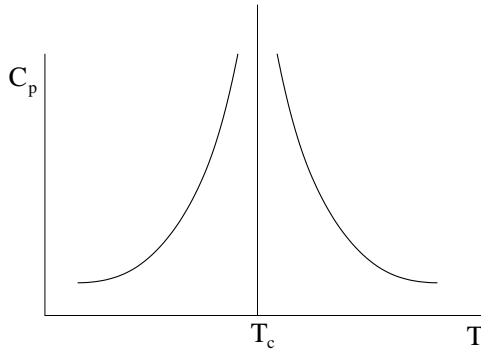


Figura 2.3 Para la transición de fase ferromagnética las primeras derivadas de potencial de Gibbs, G , son continuas pero algunas derivadas de segundo orden, por ejemplo el calor específico a campo magnético constante, C_H , divergen.

esta temperatura, el sistema es invariante sólo ante rotaciones alrededor de un eje orientado en la dirección de la magnetización. Por lo tanto, la simetría es un aspecto importante a considerar en el estudio de las transiciones de fase.

Landau asoció la idea de *parámetro de orden* con los cambios o rompimientos de simetría que acompañan a las transiciones de fase. El parámetro de orden es, en general, una cantidad física de carácter intensivo que es cero en la fase más simétrica, o más desordenada, y diferente de cero en la fase menos simétrica, o más ordenada [Papon, P., et. al., 2002].

La noción de parámetro de orden tiene un significado cualitativo bastante claro: cuando la temperatura decrece, el orden del sistema se incrementa. Por ejemplo, cuando un líquido se enfría por debajo de su temperatura de solidificación, el sólido resultante está más ordenado que el líquido. Por otra parte, cuando un sistema magnético se enfría por debajo de su temperatura de Curie, el orden magnético del sistema se incrementa apareciendo una magnetización macroscópica, que indica la existencia de un orden magnético. En el caso del ferromagnetismo,

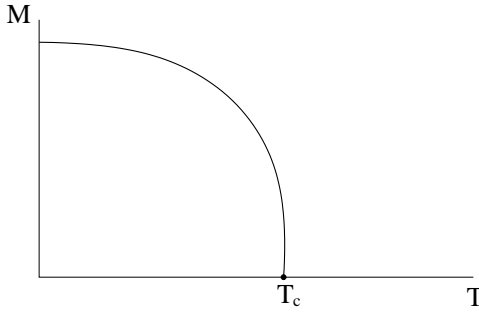


Figura 2.4 Magnetización como función de la temperatura, en ausencia de campo magnético externo. Por arriba de la temperatura de Curie, T_c , la magnetización es cero.

el parámetro de orden es la magnetización, mientras que para un sistema ferroeléctrico es la polarización eléctrica. Sin embargo, la elección del parámetro de orden no siempre es evidente. Por ejemplo en el caso de la superfluidez, el parámetro de orden resulta ser la función de onda de la fase superfluida.

En términos del parámetro de orden, se pueden distinguir dos tipos de transiciones de fase. Por una parte, transiciones para las que el parámetro de orden es discontinuo en el punto de transición (Figura 2.5(a)) y por otra, transiciones para las que el parámetro de orden tiende a cero continuamente cuando la temperatura aumenta hacia la temperatura de transición (Figura 2.5(b)). Las transiciones de fase sólido-líquido y líquido-vapor (excepto en el punto crítico) de un fluido, son ejemplos de transiciones dentro del primer grupo, las cuales, en una nomenclatura más moderna se conocen como transiciones de fase discontinuas. En el segundo grupo se encuentran transiciones de fase tales como la fluido-superfluido, la ferromagnética y la transición líquido-vapor de un fluido, en el punto crítico. Estas transiciones son generalmente conocidas como transiciones de fase críticas o continuas por lo que los correspondientes puntos de transición se conocen como puntos críticos. Un punto crítico

de orden p es aquel en donde p fases se vuelven indistinguibles. De acuerdo con la regla de las fases de Gibbs, para que un sistema pueda exhibir un punto crítico de orden p , debe tener un mínimo de $2p - 3$ componentes [Papon, P., et. al., 2002].

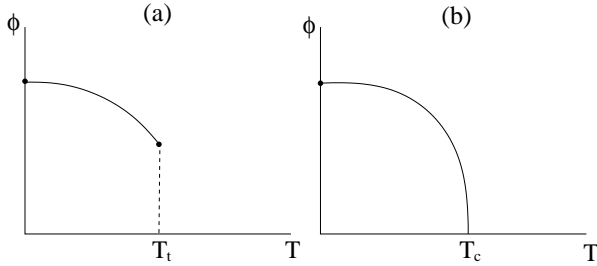


Figura 2.5 Comportamiento del parámetro de orden, ϕ , como función de la temperatura, para una transición de fase (a) discontinua, donde T_t es la temperatura de transición (b) continua, donde T_c es la temperatura crítica.

El punto crítico juega un papel fundamental en la teoría de las transiciones de fase, por lo que es muy importante estudiar detenidamente las características del sistema en ese punto. A continuación se expondrá brevemente lo que se considera una de las mejores caracterizaciones del comportamiento de un sistema en el entorno del punto crítico, a saber, los exponentes críticos.

En las proximidades del punto crítico las funciones termodinámicas se pueden desarrollar en la forma:

$$F(\varepsilon) = A\varepsilon^\lambda(1 + B\varepsilon^\nu + \dots),$$

con $\nu > 0$, donde $\varepsilon = (T - T_c)/T_c$, T es la temperatura y T_c el valor de la temperatura en el punto crítico. El exponente crítico correspondiente al punto crítico $\varepsilon = 0$ de la función $F(\varepsilon)$ se define entonces como:

$$\lambda = \lim_{\varepsilon \rightarrow 0} \frac{\ln F(\varepsilon)}{\ln \varepsilon},$$

si el límite existe.

| Exp. | Definición | Condiciones | | | Cantidad que caracteriza |
|----------|-----------------------------------|---------------|-------|-------|--|
| | | ε | H | M | |
| α | $C \sim (\varepsilon)^{-\alpha}$ | > 0 | $= 0$ | $= 0$ | Calor específico para $H = 0$ |
| β | $M \sim (-\varepsilon)^\beta$ | < 0 | $= 0$ | $= 0$ | Magnetización para $H = 0$ |
| γ | $\chi \sim (\varepsilon)^\gamma$ | > 0 | $= 0$ | $= 0$ | Susceptibilidad magnética para $H = 0$ |
| δ | $H \sim M ^\delta \text{sgn}(M)$ | $= 0$ | $= 0$ | $= 0$ | Grado de la isoterma crítica |

C denota el calor específico, M la magnetización, χ la susceptibilidad magnética y H el campo magnético.

Tabla 2.1 Definición de los exponentes críticos para un ferromagneto.

El comportamiento de la función $F(\varepsilon)$ en la vecindad de $\varepsilon = 0$ depende fundamentalmente del valor de λ . Por ejemplo, si $\lambda < 0$, $F(\varepsilon)$ diverge en el punto crítico; si $\lambda > 0$, la función se hace cero y, si $\lambda = 0$, puede tenerse una divergencia logarítmica, una discontinuidad o bien una divergencia de la forma:

$$F(\varepsilon) = A + B|\varepsilon|^{\frac{1}{2}}.$$

En esta discusión el parámetro de expansión ε se ha elegido como dependiente de la temperatura pero pueden introducirse otros parámetros que dependan, por ejemplo, de la presión, el volumen, el campo magnético externo, etc., y definir en forma análoga los respectivos exponentes críticos. En la Tabla 2.1 se definen algunos de los exponentes críticos más usados, relativos a ε , para un sistema magnético.

En muchos de los casos que se han estudiado, las teorías clásicas predicen correctamente el comportamiento cualitativo

del sistema pero no proporcionan los valores correctos de los exponentes críticos ni de la temperatura crítica, debido a que no toman en cuenta las correlaciones de corto alcance que son especialmente importantes en el punto crítico.

Otros métodos de solución para este tipo de problemas son series de alta y baja temperatura, el grupo de renormalización, el método de Monte Carlo, además de algunas soluciones exactas. En años recientes, el método de Monte Carlo ha recibido mucha atención debido a que por medio de él pueden simularse numéricamente las propiedades termodinámicas de un sistema. Los resultados que se obtienen son precisos y los errores estadísticos involucrados pueden hacerse tan pequeños como se quiera si se invierte suficiente tiempo de máquina [Binder, K., 1979], [Binder, K., et. al., 2002]. De las soluciones exactas que se mencionan, la más conocida es la de L. Onsager al modelo de Ising bidimensional en ausencia de campo magnético externo [Onsager, L., 1994]. Dicho modelo es uno de los más estudiados debido a que es sencillo y permite estudiar las transiciones de fase, en particular el comportamiento crítico del sistema.

El modelo de Ising, originalmente propuesto para explicar algunos aspectos del magnetismo, tiene aplicaciones en diversos problemas de la física estadística tales como mezclas binarias, aleaciones, gases en redes y emulsiones.

MAGNETISMO Y EL MODELO DE ISING

Los diferentes tipos de magnetismo que se han observado son alrededor de catorce [Hurd, C. M., 1982], que pueden dividirse en dos grandes grupos, los de tipo no cooperativo y los de tipo cooperativo. En los primeros no hay interacción entre los espines magnéticos mientras que en los de tipo cooperativo las interacciones entre los espines son intrínsecamente importantes. Los fenómenos ferro y antiferromagnético pertenecen al segundo grupo, pues se deben a las interacciones entre los momentos magnéticos atómicos. La interacción predominante entre los momentos magnéticos es la llamada interacción de intercambio. En 1928, W. K. Heisenberg demostró que el Hamiltoniano correspondiente a dicha interacción entre los átomos i y j está dado por

$$H = -2J\mathbf{S}_i \cdot \mathbf{S}_j, \quad (3.1)$$

donde \mathbf{S}_i es el operador de espín del i -ésimo átomo y J es un parámetro que mide la intensidad de la interacción de intercambio [Gopal, E. S. R., 1974]. Si $J > 0$, la energía de interacción es más pequeña cuando los momentos magnéticos son paralelos que cuando son antiparalelos. Cuando $J < 0$ se tiene la situación inversa. Por lo tanto, si $J > 0$ ($J < 0$) el estado de mínima energía será el que favorezca la orientación paralela (antiparalela) de los momentos magnéticos, esto es, aquel que tienda a producir ferromagnetismo (antiferromagnetismo). Además, como la interacción de intercambio depende del grado de traslape que puedan alcanzar las funciones de onda de los electrones de los dos átomos, J cae rápidamente con el incremento en la separa-

ción entre los átomos, por lo que la interacción de intercambio es apreciable sólo si los átomos están lo suficientemente próximos unos de otros. La mecánica estadística de partículas con interacciones entre vecinos cercanos de la forma $-2J\mathbf{S}_i \cdot \mathbf{S}_j$ es muy complicada, debido al comportamiento cooperativo que presentan. Por este motivo han surgido aproximaciones al modelo de Heisenberg tales como el modelo de Weiss o el modelo de Ising.

El número de modelos que se han estudiado con el fin de obtener información acerca de la naturaleza de las transiciones de fase es muy grande, en los últimos años se ha logrado avanzar considerablemente trabajando con el modelo de Ising. En 1920, el físico alemán W. Lenz propuso un modelo para explicar algunos aspectos del fenómeno del magnetismo que se conocían experimentalmente [Brush, S.G., 1967]. Años después, por sugerencia y bajo la supervisión del propio Lenz, E. Ising desarrolló su trabajo de tesis doctoral estudiando dicho modelo en su versión unidimensional. Con su trabajo Ising mostró que el modelo unidimensional no exhibe una transición de fase para $T > 0$, y extendió este resultado a los casos bi y tridimensional. Como veremos, años más tarde, Onsager demostró que esta afirmación es falsa en el caso bidimensional. A partir del trabajo de R. Peirls titulado ‘On Ising’s Model of Ferromagnetism’ [Peirls, R., 1936], el modelo recibe el nombre de *modelo de Ising*.

El modelo de Ising pertenece a una clase de modelos en donde los sistemas se modelan como una red con una base de microsistemas en los sitios de la red, dichos microsistemas pueden tener un número discreto o continuo de estados y habrá una cierta interacción definida entre los sitios. En esta clase de modelos se encuentran también el modelo de Potts [Wu, F. Y., 1982], el modelo Gaussiano [Berlin, T. H. y Kac, M., 1952], el modelo XY [Kosterlitz, J. M. y Thouless, D. J., 1973], los modelos $O(n)$ [Stanley, H. E., 1968], Z_n [Baxter, R. J., 1972], etc. [Domb, C y Green, M. S.], [Domb, C. y Lebowitz, J. L.]. El modelo de Ising ha sido estudiado en diversos tipos de redes, en su versión original se asocian dos estados a cada uno de los microsistemas y se establece una interacción a primeros veci-

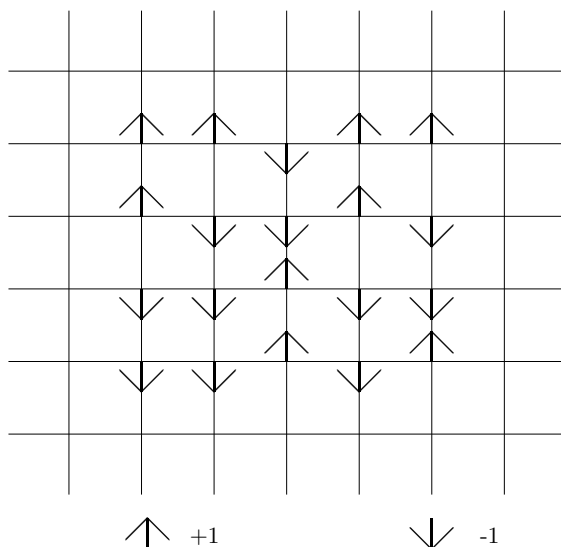


Figura 3.1 Modelo de Ising en red cuadrada.

nos. Los microsistemas son en este caso átomos cuyos espines sólo pueden apuntar hacia arriba o hacia abajo, tomando los valores ± 1 , respectivamente. Es decir, en la ecuación (3.1) sólo se conserva el término $-2JS_iS_j$ como término dominante de la interacción de intercambio y los espines están normalizados de tal manera que el Hamiltoniano (3.1) se reduce a

$$H = -2JS_iS_j,$$

donde S_i representa la proyección del operador de espín \mathbf{S}_i sobre el eje z . En red cuadrada el modelo se ilustra en la Figura 3.1.

Cuando se trabaja con el modelo bidimensional, es conveniente identificar a los espines S_i a través de dos índices que indiquen su localización en la red. Si en ésta hay un total de $N = N_x N_y$ sitios, es decir N_x en la dirección horizontal y N_y

en la dirección vertical, entonces el modelo tiene 2^N posibles configuraciones, cada una de las cuales está determinada por el conjunto:

$$\{S_i\} = \{S_{1,1}, S_{1,2}, \dots, S_{1,N_x}, \dots, S_{N_y,1}, S_{N_y,2}, \dots, S_{N_y,N_x}\},$$

donde $S_{i,j} = \pm 1$.

El Hamiltoniano del modelo en red cuadrada, en presencia de un campo magnético externo, está dado entonces por

$$H = -J \left(\sum_i \sum_j S_{i,j} S_{i+1,j} + \sum_i \sum_j S_{i,j} S_{i,j+1} \right) - h \sum_i \sum_j S_{i,j}, \quad (3.2)$$

donde J es la constante de interacción entre dos espines, h el campo magnético externo, que se supone en la misma dirección de los espines, y el término $-h \sum_i \sum_j S_{i,j}$ denota lo que se conoce como energía de Zeeman del sistema.

El estudio sistemático del modelo de Ising se inició como consecuencia del desarrollo de la física de aleaciones más que por un interés directo. La evaluación exacta de la función de partición en el caso bidimensional se obtuvo en 1944 con el trabajo de L. Onsager, quien resolvió el modelo en ausencia de campo magnético externo [Onsager, L., 1994]. Dicha solución muestra que en ausencia de campo magnético externo, el modelo de Ising bidimensional exhibe una transición de fase de segundo orden a una temperatura finita T_c dada por:

$$K_b T_c / J = 2.26918\dots$$

donde K_b es la constante de Boltzmann. Algunos de los exponentes críticos obtenidos con esta solución son (c.f. Tabla 2.1)

$$\alpha = \alpha' = 0_{\log}, \quad \beta = 1/8, \quad \gamma = \gamma' = 7/4, \quad \delta = 15$$

La solución de Onsager constituyó la primera demostración de que la función de partición de un sistema contiene toda la información referente a éste incluyendo la existencia de transiciones de fase. A raíz de esta solución, el modelo de Ising y

en general los modelos en redes han recibido mucha atención debido a que una gran variedad de sistemas tales como ferromagnetos, polímeros, emulsiones, aleaciones y mezclas de fluidos se vuelven tratables bajo este esquema.

Nos referiremos ahora a algunos modelos que están estrechamente relacionados con el modelo de Ising, entre los que se encuentran aquellos en los que:

1.- Las interacciones no son uniformes, es decir la constante de interacción depende del sitio (i, j) , por lo que el Hamiltoniano, sin incluir la energía de Zeeman es

$$H = \sum_{\langle i,j \rangle} J_{ij} S_i S_j,$$

en donde al igual que antes, los espines sólo interactúan con sus vecinos más cercanos. El símbolo $\langle i, j \rangle$ indica que la suma se realiza únicamente sobre los pares de espines que son vecinos cercanos. Bajo estas condiciones se dice que el sistema es anisotrópico en el espacio físico. En el caso bidimensional, un ejemplo sencillo lo constituye el denominado modelo rectangular de Ising, en el que hay una constante de interacción J_v entre los espines a lo largo de las columnas y otra J_h a lo largo de los renglones, de tal manera que

$$H = -J_v \sum_{\langle i,j \rangle} S_{i,j} S_{i+1,j} - J_h \sum_{\langle i,j \rangle} S_{i,j} S_{i,j+1}. \quad (3.3)$$

2.- Las interacciones se extienden a más vecinos, haciendo más realista el modelo. En este caso el Hamiltoniano es

$$H = \sum_{i,j} J_{ij} S_i S_j,$$

3.- Las interacciones son anisotrópicas en el espacio interno de los espines y a vecinos cercanos, es decir el Hamiltoniano del sistema se convierte en

$$H = \sum_{\langle i,j \rangle} \sum_{n=1}^D J_n S_{in} S_{jn},$$

donde D es la dimensión del espacio en el que se encuentran los espines. Claramente este caso se reduce al caso isotrópico cuando $J_n = J$ para $n = 1, 2, \dots, D$.

Otros modelos en redes que es interesante estudiar y que guardan cierta semejanza con el modelo de Ising son aquellos que presentan frustración y/o desorden. Un modelo que presenta frustración es aquel que, no siendo capaz de alcanzar un estado de mínima energía que satisfaga enteramente sus constricciones microscópicas, posee varios estados con la misma energía, es decir un modelo frustrado no tiene una configuración única para su estado base y como resultado de ello muestra metaestabilidad, efectos de histéresis, etc. El modelo más sencillo que presenta frustración es un antiferromagneto en red triangular (Figura 3.2). Consideremos un triángulo elemental o plaqueta de la red, mediante una rápida inspección es fácil convencerse de que dada cualquier configuración, cada plaqueta tendrá al menos un par de espines que no están en su estado de mínima energía, lo cual según lo expuesto anteriormente significa que el modelo presenta frustración.

Existen otros modelos que presentan frustración en los que también aparece el desorden y existen varias clasificaciones de los diferentes tipos de desorden, que dependen del área de trabajo que se considere. Dentro del contexto de este trabajo es conveniente dividir a los sistemas desordenados en varios grupos dependiendo de si el desorden se debe a la presencia de campos aleatorios, constantes de interacción aleatorias o anisotropías aleatorias [Hertz, J., 1985]. Es posible además hacer clasificaciones más finas al especificar si el desorden es débil o fuerte y cual es la función de distribución de la variable que provoca el desorden. En un modelo ligeramente desordenado, el desorden no da lugar a la aparición de nuevos puntos críticos o nuevas fases, mientras que en uno fuertemente desordenado si aparecen nuevas fases. Una situación de desorden débil producido por constantes de interacción aleatorias puede obtenerse a través de la aleación de dos sustancias con temperaturas críticas diferentes o por la introducción de impurezas no

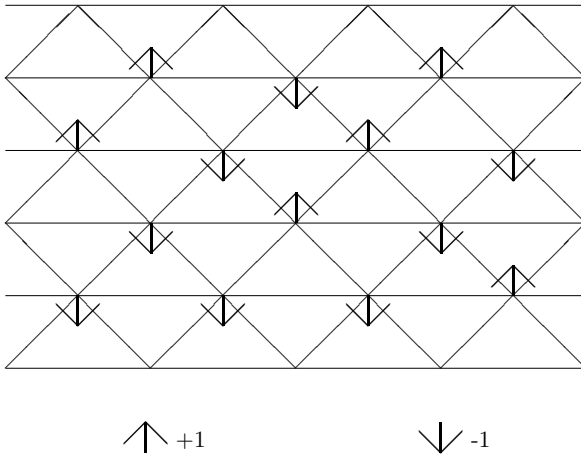


Figura 3.2 Antiferromagneto en red triangular.

magnéticas en un material magnético. Un vidrio de espín es un ejemplo de un ferromagneto fuertemente desordenado, en donde el desorden está provocado por interacciones aleatorias, aunque estrictamente hablando son los signos de estas constantes de interacción los que tienen un carácter aleatorio más que las interacciones mismas. Típicamente, en un vidrio de espín 50% de las interacciones son ferromagnéticas y el otro 50% son antiferromagnéticas [Parisi, G., 1986]. Esto se manifiesta en que el modelo no presenta una condensación ya sea ferromagnética o antiferromagnética a bajas temperaturas, sino que pasa por una serie de estados de manera recurrente sin alcanzar un estado de equilibrio. Un vidrio de espín está caracterizado precisamente por esta fase de baja temperatura, en la que el tiempo de relajación hacia el equilibrio es muy largo, como se observa en los vidrios reales. Esta característica puede relacionarse con la imposibilidad práctica de encontrar numéricamente el estado base del vidrio de espín. El número de operaciones que es necesario realizar a través de cualquier algoritmo para llegar al estado de

equilibrio crece exponencialmente con el volumen [Parisi, G., 1986]. En lo que resta, sólo nos ocuparemos del caso de una distribución de interacciones particular que provoca que todas las plaquetas de la red estén frustradas.

MÉTODO DE MONTE CARLO

Los sistemas que se estudian en física estadística son, en general, sistemas con un gran número de grados de libertad. La descripción de estos sistemas frecuentemente involucra la evaluación de integrales en espacios de dimensión muy grande. Consideremos, por ejemplo, un gas formado por N partículas de masa m que se encuentra a una temperatura T y cuyo Hamiltoniano es

$$H(\mathbf{x}) = \sum \mathbf{p}_i^2/2m + \sum u(r_{ij}).$$

En esta ecuación $\mathbf{x} = (\mathbf{r}, \mathbf{p})$, $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$, $\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N)$, $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$, siendo \mathbf{r}_i el vector de posición de la i -ésima partícula, $\mathbf{p}_i = m\dot{\mathbf{r}}_i$ su momento, y $u(r_{ij})$ el potencial entre las partículas separadas por una distancia r_{ij} . La función de partición canónica para este sistema está dada por

$$Z(N, V, T) = \int \exp[-\beta H(\mathbf{x})] \mathbf{dr} \mathbf{dp}, \quad (4.1)$$

donde $\beta = \frac{1}{K_b T}$ y el valor esperado de cualquier observable \mathbf{A} del sistema se obtiene al evaluar

$$\langle \mathbf{A} \rangle = (1/Z) \int \mathbf{A}(\mathbf{x}) \exp[-\beta H(\mathbf{x})] \mathbf{dr} \mathbf{dp}. \quad (4.2)$$

Cuando las fuerzas entre las partículas no dependen de los momentos, la integral de la ecuación (4.1), definida en un espacio $6N$ -dimensional se reduce al producto de dos integrales sobre espacios $3N$ -dimensionales. Lo mismo sucede con la integral de la

ecuación (4.2) en el caso en que $\mathbf{A}(\mathbf{x}) = \mathbf{A}_1(\mathbf{r})\mathbf{A}_2(\mathbf{p})$. Para describir las propiedades termodinámicas de este sistema es necesario evaluar dichas integrales, pero su evaluación analítica la mayoría de las veces es irrealizable, por lo que se lleva a cabo numéricamente. Sin embargo, esta evaluación tampoco es directa, ya que no es posible utilizar métodos numéricos convencionales excepto para valores muy pequeños de N , en casos como este se emplea el llamado método de Monte Carlo. Una breve historia sobre este método puede consultarse en [Newman, M. E. J., et. al., 1994].

En general, el método de Monte Carlo permite evaluar numéricamente cierto tipo de integrales mediante una discretización aleatoria del espacio en cuestión. En mecánica estadística el método de Monte Carlo se emplea para estudiar modelos de sistemas termodinámicos por medio de una simulación estocástica en una computadora. La descripción del sistema que se desea estudiar se hace en términos de un modelo y el Hamiltoniano asociado a éste. Como ya mencionamos, la evaluación del tipo de integrales en que estamos interesados requiere una discretización del espacio de integración. La realización más sencilla de dicha discretización es a través de un muestreo aleatorio simple, sin embargo, en la evaluación de integrales como las que aparecen en las ecuaciones (4.1) y (4.2) esto no es posible debido a que el término $\exp[-\beta H(\mathbf{x})]$ tiene variaciones de varios ordenes de magnitud. Por lo tanto es necesario implementar un algoritmo que permita muestrear eficientemente el espacio de integración. El algoritmo que permite aplicar eficientemente el método de Monte Carlo en la evaluación del tipo de integrales en que estamos interesados, fue propuesto por N. Metropolis, A.W. Rosenbluth, N. Rosenbluth, A. Teller y E. Teller en 1953 [Metropolis, N., et. al., 1953]. El enfoque de Metropolis et. al. se basa en la idea de muestrear las regiones del espacio de integración donde se encuentran los estados importantes del sistema. De acuerdo con el algoritmo de Metropolis et. al., los puntos adecuados para la evaluación de una integral se eligen conforme a una probabilidad $P(\mathbf{x}_\nu)$, $\nu = 1, 2, \dots, n$, de tal manera que sea

más probable considerar puntos en donde la función tenga valores más significativos que en otras regiones. La probabilidad adecuada es en este caso

$$P(\mathbf{x}_\nu) = \frac{1}{Z} \exp[-\beta H(\mathbf{x}_\nu)], \quad (4.3)$$

donde Z es la función de partición definida por la ecuación (4.1). La probabilidad dada por la ecuación (4.3) es la adecuada, ya que al elegir los puntos con esta probabilidad, el valor esperado de la observable \mathbf{A} definido por la ecuación (4.2) puede aproximarse por

$$\langle \mathbf{A} \rangle \approx \tilde{\mathbf{A}} = \frac{1}{n} \sum_{\nu=1}^n \mathbf{A}(\mathbf{x}_\nu), \quad (4.4)$$

donde \mathbf{x}_ν son los puntos que se tomarán en cuenta para la evaluación numérica. La implementación del algoritmo no es directa porque no conocemos explícitamente la probabilidad $P(\mathbf{x}_\nu)$, sin embargo, los puntos necesarios para la evaluación pueden generarse a través de una cadena de Markov de longitud M . La cadena de Markov estará caracterizada por una probabilidad de transición $W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ de un punto a otro, la cual deberá construirse de manera que en el límite $M \rightarrow \infty$ la probabilidad de elegir alguna configuración \mathbf{x}_ν sea precisamente la probabilidad $P(\mathbf{x}_\nu)$ dada por la ecuación (4.3). Para que la probabilidad de transición $W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ tenga la propiedad deseada, son condiciones necesarias y suficientes las siguientes:

- (i) Condición de normalización:

$$\sum_{\nu'} W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = 1, \text{ para toda } \nu'.$$

- (ii) Condición de ergodicidad:

$$\text{Si } P(\mathbf{x}_\nu) > 0 \text{ y } P(\mathbf{x}_{\nu'}) > 0, \text{ entonces } W(\mathbf{x}_\nu, \mathbf{x}_{\nu'}) > 0.$$

- (iii) Condición de estado estacionario:

$$\sum_{\nu} P(\mathbf{x}_\nu) W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = P(\mathbf{x}_{\nu'}), \text{ para toda } \nu'.$$

Una forma sencilla de garantizar que se satisface la condición (iii) es elegir

$$P(\mathbf{x}_\nu)W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = P(\mathbf{x}_{\nu'})W(\mathbf{x}_{\nu'} \rightarrow \mathbf{x}_\nu), \quad (4.5)$$

que es la condición de reversibilidad microscópica.

La ecuación (4.5) escrita en la forma

$$\frac{P(\mathbf{x}_\nu)}{P(\mathbf{x}_{\nu'})} = \frac{W(\mathbf{x}_{\nu'} \rightarrow \mathbf{x}_\nu)}{W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})},$$

indica que la razón de las probabilidades de transición sólo depende del cambio en la energía, ya que

$$\frac{P(\mathbf{x}_{\nu'})}{P(\mathbf{x}_\nu)} = \exp[-\beta(H(\mathbf{x}_{\nu'}) - H(\mathbf{x}_\nu))].$$

El siguiente paso es construir la probabilidad de transición $W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ de acuerdo a las condiciones (i), (ii) y (iii) que no la determinan unívocamente, pues existe cierto grado de arbitrariedad en su elección. Una definición que se encuentra adecuada es

$$W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = \begin{cases} \exp[-\beta\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})] & \text{si } \delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) > 0 \\ 1 & \text{en otro caso} \end{cases} \quad (4.6)$$

donde $\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = H(\mathbf{x}_{\nu'}) - H(\mathbf{x}_\nu)$. Nótese que esta probabilidad no está normalizada, sin embargo esto no es necesario para la simulación.

Ahora bien, de acuerdo con la ecuación (4.6) dado un punto \mathbf{x}_n en el espacio fase, para generar \mathbf{x}_{n+1} se da un paso tentativo al punto \mathbf{x}_t , el cual se elige de acuerdo a una probabilidad uniforme dentro de un cubo de lado a alrededor de \mathbf{x}_n y se calcula el cambio $\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t)$ en la energía del sistema asociado con el paso $\mathbf{x}_n \rightarrow \mathbf{x}_t$. Si $\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t) < 0$, es decir, si el nuevo estado del sistema es un estado de menor energía, el cambio se acepta y $\mathbf{x}_{n+1} = \mathbf{x}_t$. Por otro lado, si $\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t) > 0$, entonces el cambio se hace con probabilidad $\exp[-\beta\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t)]$. En la práctica esto se lleva a cabo comparando la cantidad

$\exp[-\beta\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t)]$ con un número aleatorio η distribuido uniformemente en el intervalo $[0, 1)$ [Newman, M. E. J., et. al., 1994]. Si $\eta < \exp[-\beta\delta H(\mathbf{x}_n \rightarrow \mathbf{x}_t)]$ el cambio se lleva a cabo y $\mathbf{x}_{n+1} = \mathbf{x}_t$, en caso contrario $\mathbf{x}_{n+1} = \mathbf{x}_n$. El proceso se repite hasta formar el conjunto de puntos necesario. El parámetro a que fija los estados elegibles se determina empíricamente teniendo en cuenta que si a es muy grande es poco probable que se acepte el estado de prueba y si a es muy pequeña, el espacio fase se recorrerá muy lentamente. Las características más importantes del algoritmo de Metropolis et. al. son: a) proporcionar el promedio de una función tomando en cuenta los valores de ésta en puntos del espacio cuyas probabilidades son significativas y b) asegurar la convergencia del promedio calculado sobre la cadena de Markov, ecuación (4.4), hacia el promedio definido por la ecuación (4.2) cuando $M \rightarrow \infty$.

El algoritmo hasta aquí descrito es un procedimiento que permite generar una configuración a partir de otra, pero que no proporciona un criterio para elegir la configuración inicial, lo cual conduce a que dicha configuración pueda construirse con cierta arbitrariedad y que no corresponda a un estado de equilibrio del sistema en consideración. Así pues, para llegar al equilibrio es necesario avanzar un cierto número de configuraciones, las cuales deberán ignorarse al calcular las observables para que éstas no dependan de la condición inicial. Debido a que puede existir una alta correlación entre los puntos adyacentes de la cadena de Markov conviene calcular las observables del sistema en configuraciones lo suficientemente separadas para disminuir dicha correlación, por ejemplo, considerando configuraciones separadas por algunos puntos dentro de la cadena, de tal manera que

$$\tilde{\mathbf{A}} = \frac{1}{n} \sum_{i=1}^n \mathbf{A}(\mathbf{x}_{\nu_i}), \quad (4.7)$$

donde $\nu_i = \nu_0 + i\mu_0$, $i = 1, 2, \dots, n$ y $n = (M - \nu_0)/\mu_0$. En la ecuación (4.7) ν_0 es el número de pasos necesarios para llevar al sistema a un estado de equilibrio y μ_0 es el número de pa-

sos que separan a las configuraciones en donde se calculan las observables. Una estimación de la incertidumbre asociada a \mathbf{A} está dada por:

$$(\sigma_{\mathbf{A}})^2 = \frac{1}{n} \left[\frac{1}{n} \sum_{i=1}^n \mathbf{A}_i^2 - \left(\frac{1}{n} \sum_{i=1}^n \mathbf{A}_i \right)^2 \right] \quad (4.8)$$

en donde se ha considerado a $\mathbf{A}_i = \mathbf{A}(\mathbf{x}_{\nu_i})$ como una variable aleatoria y se ha utilizado el teorema del límite central para n grande [Koonin, S. E., 1985]. Esta expresión indica, en general, que el error en el cálculo de \mathbf{A} disminuye al aumentar el número n de puntos que se tomen en cuenta. Debido a esta última observación, dado un número de puntos en la cadena, sería conveniente tomar mediciones con mucha frecuencia, es decir elegir μ_0 pequeña. Sin embargo, esto puede resultar costoso en tiempo de máquina y no ser lo más conveniente para lograr un muestreo eficiente. Es necesario entonces tener un criterio que tome en cuenta estos dos aspectos, el cual dependerá del problema particular que se estudie. Algunos factores que pueden afectar la confiabilidad de los cálculos y que por lo tanto hay que tomar en cuenta son los siguientes: (1) En la simulación se están utilizando números que son en realidad pseudoaleatorios, ya que se generan a través de un procedimiento determinista. Para determinar la influencia de estos números sobre los promedios se han hecho estudios con sistemas cuya solución se conoce, encontrándose que cuando se realizan del orden de $10^3 - 10^4$ pasos de Monte Carlo por partícula:

$$\frac{(\tilde{\mathbf{A}} - \langle \mathbf{A} \rangle)}{\langle \mathbf{A} \rangle} \sim 10^{-2},$$

donde $\tilde{\mathbf{A}}$ es el promedio calculado con los números pseudoaleatorios [Binder, K., 1979]. En el método de Monte Carlo la discretización no es un problema de principio pues se ha demostrado su convergencia estocástica, queda sin embargo por comprobar si en la práctica ésta es alcanzable en un número razonable de pasos. Esto ha sido comprobado favorablemente

en muchos casos. (2) El inconveniente de simular sistemas macroscópicos formados por alrededor de 10^{23} partículas a través de sistemas de aproximadamente 10^3 partículas que se puede reducir mediante el uso de condiciones a la frontera periódicas. Además de los aspectos anteriores, hay comportamientos intrínsecos de carácter físico de cada sistema que dificultan alcanzar el equilibrio termodinámico en puntos cercanos a un punto crítico.

A continuación nos ocuparemos de la descripción del programa que se implementó a fin de facilitar su uso.

DESCRIPCIÓN DEL PROGRAMA DE COMPUTADORA

En esta sección describiremos el programa que se implementó para simular el modelo de Ising bidimensional en red cuadrada cuyo Hamiltoniano está dado por la ecuación (3.2). La simulación se lleva a cabo por medio del método de Monte Carlo y haciendo uso del algoritmo de Metropolis et. al. En la simulación se imponen condiciones a la frontera periódicas, es decir, el modelo se encuentra sobre la superficie de un toro. A través de este programa es posible también simular otros modelos, como los mencionados en la sección 3. El programa está implementado en lenguaje C y está diseñado para correrse en una computadora personal compatible con el sistema IBM-PC.

El programa realiza una simulación de Monte Carlo del modelo de Ising bidimensional en red cuadrada, empleando el algoritmo de Metropolis et. al.. Para implementar el algoritmo de Metropolis et. al., el paso de una configuración \mathbf{x}_ν a otra $\mathbf{x}_{\nu'}$, puede consistir por ejemplo en cambiar aleatoriamente todos los espines, sin embargo esto puede producir una configuración $\mathbf{x}_{\nu'}$, muy diferente de \mathbf{x}_ν y por lo tanto habrá una probabilidad muy grande de rechazarla. Por esta razón y otras que veremos más adelante, tomaremos como configuración tentativa una que difiera de la anterior en tan solo un espín. Recuérdese que, de acuerdo con el algoritmo de Metropolis et. al., el cambio de configuración se acepta si

$$W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = \exp[-\beta\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})] > \eta,$$

donde $\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ es el cambio en la energía asociado al paso

$\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}$ y η es un número aleatorio distribuido uniformemente en el intervalo $[0, 1)$. Debido a que sólo se considera el cambio del espín $S_{i,j}$, entonces

$$\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'}) = 2S_{i,j}(Jf + h),$$

donde $f = S_{i+1,j} + S_{i-1,j} + S_{i,j+1} + S_{i,j-1}$, es la suma de los espines que son vecinos cercanos del espín $S_{i,j}$, J es la constante de interacción entre los espines y h es el campo magnético externo. Dado que $S_{i,j} = \pm 1$, f sólo puede tomar los valores $0, \pm 2$ y ± 4 por lo que $W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ puede tomar únicamente 10 valores, los cuales se calculan una sola vez y se almacenan en la memoria a fin de evitar su cálculo repetidamente. Nótese que si la configuración $\mathbf{x}_{\nu'}$ difiere en más de un espín de la anterior, $\delta H(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ y por lo tanto, $W(\mathbf{x}_\nu \rightarrow \mathbf{x}_{\nu'})$ tomará una forma más complicada.

Para iniciar una simulación, en primer lugar se debe indicar qué modelo se va a simular, a continuación se deben proporcionar los valores de las variables que determinan el tamaño del sistema y el estado termodinámico en el que se encuentra. El tamaño está determinado por nx y ny , que son el número de columnas y renglones en la red y el estado termodinámico lo determinan la constante de interacción J y el campo magnético externo h , ambos en unidades de $K_b T$. En lugar de la constante de interacción J se puede manejar el inverso de ésta, que identificamos con la temperatura T , aunque en realidad es una cantidad adimensional al igual que J . Finalmente, es necesario proporcionar la semilla para generar los números pseudoaleatorios así como los parámetros que intervienen en el cálculo de las observables. Una vez proporcionada esta información, la simulación se inicia partiendo de una configuración \mathbf{x}_0 y se procede a realizar los pasos de Monte Carlo que constituyen la termalización del sistema. Llamamos aquí pasos de termalización a los ν_0 pasos que, según lo expuesto en la sección 4, es necesario realizar a fin de que los promedios de las observables no dependan de la condición inicial, en el programa $\nu_0 = nterm$. Cada paso de Monte Carlo consiste en recorrer sistemáticamente todos y

cada uno de los espines de la red aplicando el algoritmo a la configuración que se obtiene al invertir dicho espín. La red y el número de cambios aceptados, *accept*, se muestran en la pantalla de la computadora después de un determinado número de pasos de Monte Carlo. La red se dibuja en la pantalla asignando un pixel de color azul a los espines positivos y un pixel de color gris a los espines negativos. Una vez que se ha llevado a cabo la termalización del sistema se procede a calcular los promedios de las observables.

Las observables básicas, energía y magnetización, se calculan cada μ_0 pasos de Monte Carlo para evitar que exista correlación entre los diferentes valores empleados en el cálculo de los promedios, en el programa $\mu_0=freq$. Para mayor claridad véase la Figura 5.1, que muestra un diagrama del funcionamiento del programa.

Con el fin de contar con una estimación de los errores estadísticos asociados tanto al calor específico como a la susceptibilidad magnética, para calcular los promedios de las observables se forman *n* grupos de *tam* datos cada uno. Por otra parte, este esquema permite obtener dos estimaciones de los errores estadísticos asociados a las observables básicas, por lo que es posible monitorear la correlación entre valores sucesivos de dichas observables, a través de la comparación de ambas estimaciones [Koonin, S. E., 1985]. Cabe mencionar que cuando el programa implementado se corre en una computadora con una velocidad de 750 MHZ, éste invierte alrededor de 3.5 minutos en proporcionar resultados para cada temperatura cuando se trabaja con una red de 150x150 espines y la estadística se realiza, después de un período de 1000 pasos de termalización, sobre una muestra de 10 grupos de 10 datos tomados cada 10 pasos de Monte Carlo. Los valores de las observables se muestran en la pantalla después de que se ha formado cada grupo. Al terminar la simulación se muestran las cantidades finales. Dentro de una misma corrida, el programa permite formar hasta 15 grupos de hasta 15 datos cada uno. Estos valores, así como el tamaño del sistema, pueden incrementarse dependiendo del equipo con que

DESCRIPCIÓN DEL PROGRAMA DE COMPUTADORA

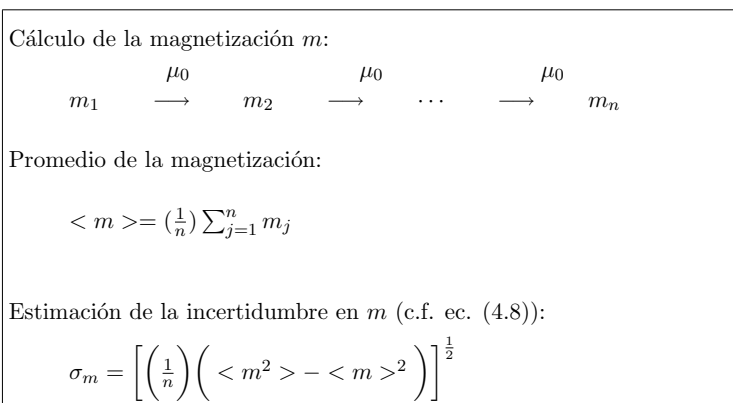
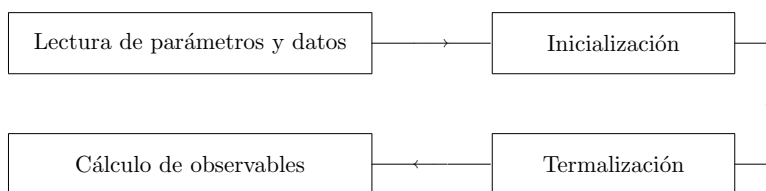


Figura 5.1 Diagrama del funcionamiento del programa *Ising.c*, se desglosa el cálculo de la magnetización m a partir de n valores.

se cuenta.

Como se indicó en la sección 4, el algoritmo de Metropolis *et al.* no proporciona un criterio para elegir la configuración inicial \mathbf{x}_0 . Las dos configuraciones iniciales más comunmente usadas son la ordenada, correspondiente a $T = 0$, y la aleatoria, correspondiente a $T = \infty$. Para temperaturas por abajo de la temperatura crítica, elegir una configuración inicial aleatoria implica que, antes de iniciar el cálculo de las observables, se debe remover del sistema una gran cantidad de energía, lo cual se traduce en un aumento en el tiempo de máquina requerido para la

simulación [Binder, K., et. al., 2002]. Por esta razón, en todos los casos considerados en este trabajo se partió de la configuración correspondiente a $T = 0$, excepto en el caso anisotrópico en el que ésta no es única. En este caso se recomienda partir de una configuración aleatoria, es decir correspondiente a $T = \infty$.

En el apéndice **A** se encuentra el listado del programa *Ising.c* el cual lleva a cabo la simulación del modelo de Ising en los casos ferromagnético, antiferromagnético, anisotrópico y frustrado. El programa consta de nueve subrutinas básicas: *inicializa*, *termaliza*, *montecarlo*, *despliega*, *observables*, *calculos*, *calculos1*, *grupos* y *ordena*. En la primera se lleva a cabo la lectura de los datos que determinan el tamaño y el estado del sistema, se inicializa la red de espines y se realiza la lectura de los parámetros que intervienen en el cálculo de los promedios; en la segunda se llevan a cabo los pasos de termalización con ayuda de las subrutinas *montecarlo* y *despliega* donde esta última se encarga de dibujar la red en la pantalla de la computadora. Los promedios de las observables se calculan a través de las cinco subrutinas restantes. *Ordena* es una subrutina que se utiliza para ordenar los datos antes de calcular los promedios a fin de minimizar los errores numéricos involucrados. El apéndice **B** contiene la subrutina *alea1.h* que se utilizó para generar los números pseudoaleatorios, la cual está tomada de Press et. al. [Press, W. H., et al., 1988].

Dentro del programa *Ising.c*, las subrutinas *montecarlo* y *observables* toman en cuenta los detalles particulares relativos a cada uno de los casos considerados: ferromagnetismo, antiferromagnetismo, anisotrópico y frustrado. En todos los casos es posible incluir un campo magnético externo.

En el caso de un antiferromagneto es necesario distinguir dos subredes, que al superponerse generan al sistema total, y calcular la magnetización en cada una de estas subredes, ya que la magnetización total deja de ser una observable adecuada pues en ausencia de campo magnético externo el promedio de ésta es cero para cualquier temperatura. La observable adecuada es ahora la magnetización en cada una de las subredes, la cual de-

berá exhibir un comportamiento similar al de la magnetización total de un ferromagneto. Otros modelos que pueden simularse introduciendo algunas modificaciones son modelos anisotrópicos y/o desordenados. Para simular un ferromagneto anisotrópico, como el descrito por la ecuación (3.3), es necesario introducir dos constantes de acoplamiento entre los espines, una a lo largo de los renglones (J_h) y otra a lo largo de las columnas (J_v). Un aspecto que es interesante estudiar en este caso es el comportamiento de la temperatura crítica cuando una de las constantes de interacción (en este caso J_v) tiende a cero, es decir, cuando pasamos del modelo bidimensional al unidimensional. Para simular un modelo desordenado que presente frustración, es necesario introducir dos constantes de interacción, las cuales toman valores simétricos $+J$ y $-J$. En este caso lo que se pretende mostrar es la existencia de varios estados metaestables que poseen la misma energía y que están igualmente frustrados. En la Figura 5.2 se muestra la distribución de interacciones que se empleó.

Finalmente, en la próxima sección se discuten algunos de los resultados obtenidos con el programa así como el tipo de trabajo que aún puede llevarse a cabo.

DESCRIPCIÓN DEL PROGRAMA DE COMPUTADORA

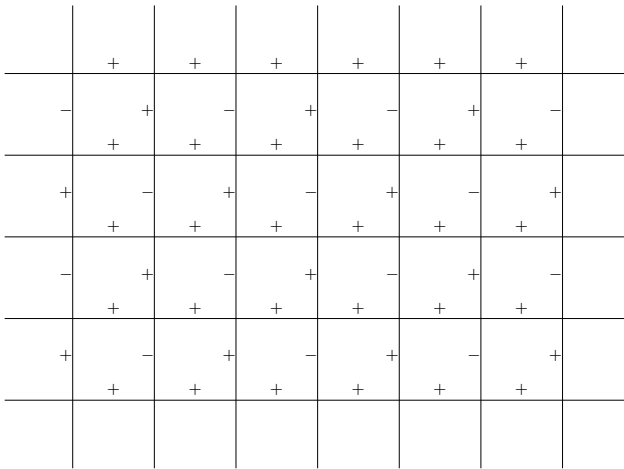


Figura 5.2 Sistema con todas las plaquetas frustradas.

RESULTADOS

En esta sección se exponen los resultados obtenidos en los casos analizados en este trabajo: ferromagnetismo con y sin campo magnético externo, antiferromagnetismo sin campo magnético externo, ferromagnetismo anisotrópico (en el espacio real) sin campo magnético externo y finalmente un sistema que presenta frustración. En todos los casos, la red considerada consta de 150×150 espines y los promedios de las observables básicas se calculan utilizando una muestra de 100 valores, organizados en 10 grupos de 10 valores cada uno, calculados cada 10 pasos de Monte Carlo. El cálculo de las observables se inicia después de un período de termalización de 1000 pasos de Monte Carlo. La susceptibilidad magnética y el calor específico se calculan como la desviación estándar en la magnetización y la energía, respectivamente.

En los casos ferromagnético y anisotrópico en ausencia de campo magnético, al realizar el cálculo de la magnetización, es necesario considerar el valor absoluto de ésta, ya que para temperaturas por abajo de la temperatura crítica la distribución de probabilidad de la magnetización, $P(m)$, tiene dos máximos uno en $+|m_{sp}|$ y otro en $-|m_{sp}|$, donde m_{sp} es la magnetización espontánea, característica de este intervalo de temperaturas, y entre estos dos valores $P(m)$ es diferente de cero, aún para $m = 0$, [Binder, K., et. al., 2002]. La misma consideración se aplica al caso antiferromagnético, en ausencia de campo magnético externo, al calcular la magnetización en cada una de las subredes.

Las observables que el programa calcula son: energía, e ,

magnetización, m , calor específico, ce y susceptibilidad magnética, sm .

En las Figuras 6.1 a 6.9, las líneas únicamente se han trazado para unir los puntos y no tienen mayor significado.

a.- Ferromagnetismo sin campo magnético externo.

En este caso, los resultados se muestran en las Figuras 6.1 a 6.4 respectivamente. El comportamiento cualitativo de las observables concuerda bastante bien con los resultados de la solución de Onsager [Onsager, L., 1994]. Las diferencias que se observan se deben a que el sistema que se está tratando es finito. Las diferencias más notables son el valor de la temperatura crítica, que es $T_0=2.4$ (y no $T_c=2.26918\dots$), la caída más o menos suave de la magnetización en $T = T_0$ así como la no divergencia de la susceptibilidad magnética y el calor específico. Se observa sin embargo, un pico bien marcado tanto en la gráfica del calor específico como en la de la susceptibilidad magnética en $T = T_0$ (Figuras 6.3 y 6.4). El valor de T_0 se determinó a partir de la gráfica de la Figura 6.2, tomando como T_0 el valor de T para el cual por primera vez $m < 0.100000$, viniendo de temperaturas bajas. Los incrementos en la temperatura al construir las gráficas fueron de 0.1 unidades, por lo que el valor de T_0 no puede darse con más cifras. Por abajo de T_0 , el sistema exhibe una fase ordenada en la que la mayoría de los espines están orientados en la misma dirección, dando lugar a una magnetización espontánea. En la vecindad de T_0 se observa la formación de dominios característicos de la transición de fase. La diversidad de tamaños de estos dominios aumenta a medida que nos aproximamos a T_0 , siendo esto una manifestación de la existencia de correlaciones a todas las escalas. Por arriba de T_0 se observa una fase paramagnética, es decir, desordenada, que está caracterizada por la orientación aleatoria de los espines y la desaparición de la magnetización espontánea.

RESULTADOS

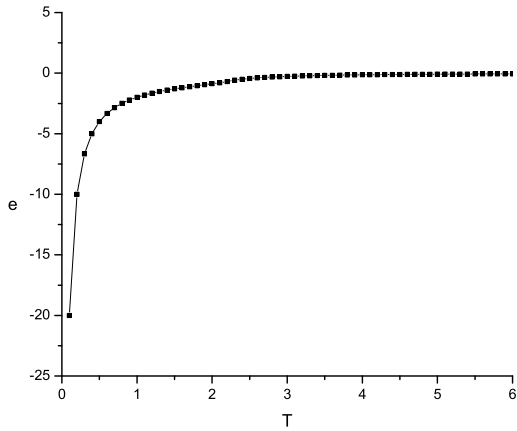


Figura 6.1 Energía como función de la temperatura (ferromagnetismo, $h = 0$).

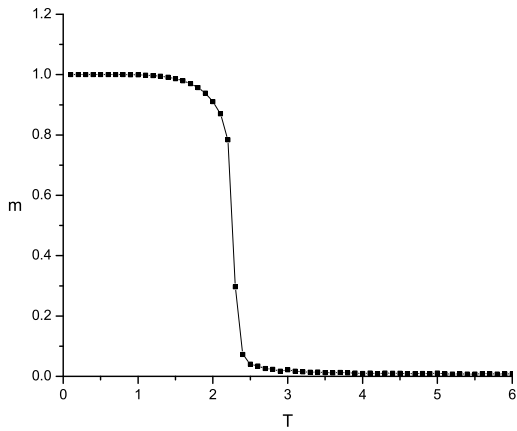


Figura 6.2 Magnetización como función de la temperatura (ferromagnetismo, $h = 0$).

RESULTADOS

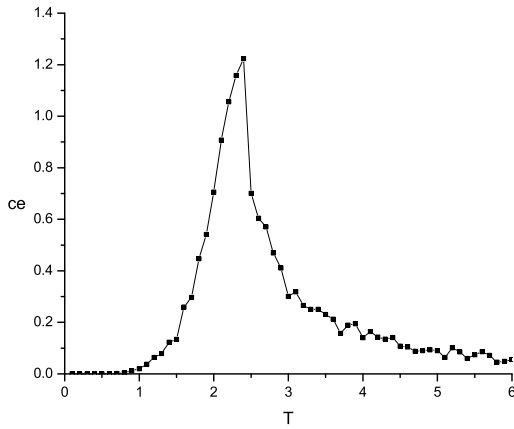


Figura 6.3 Calor específico como función de la temperatura (ferromagnetismo, $h = 0$).

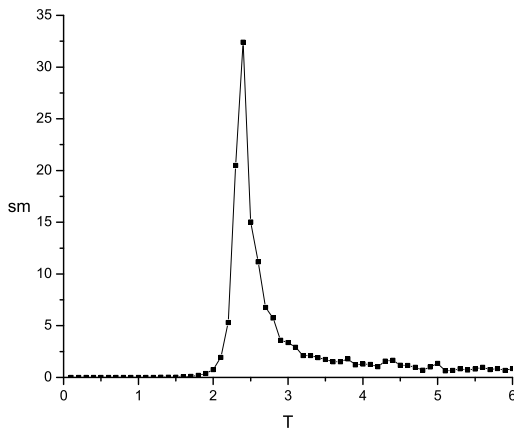


Figura 6.4 Susceptibilidad magnética como función de la temperatura (ferromagnetismo, $h = 0$).

b.- Ferromagnetismo con campo magnético externo.

Cuando se aplica un campo magnético externo a un ferromagneto se observa que la magnetización tiene una caída mucho más suave que en el caso anterior, esto se muestra en la Figura 6.5. Es decir, el sistema comienza a ordenarse a una temperatura superior a T_0 . Otro comportamiento que puede estudiarse al aplicar un campo magnético externo a un ferromagneto es el de la magnetización por arriba y por abajo de T_0 en función de éste. Como se muestra en las Figuras 6.6 y 6.7, se obtienen comportamientos diferentes para $T < T_0$ y $T > T_0$ respectivamente y se observa que los resultados concuerdan cualitativamente con los resultados experimentales.

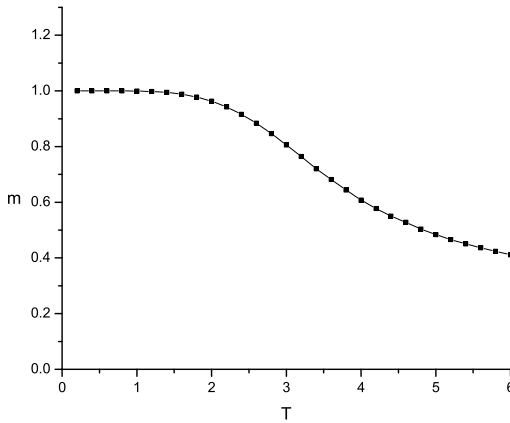


Figura 6.5 Magnetización como función de la temperatura (ferromagnetismo, $h=0.20$).

c.- Antiferromagnetismo sin campo magnético externo.

En este caso las observables calculadas son la magnetización en cada una de las subredes y el calor específico. Al igual que en el

RESULTADOS

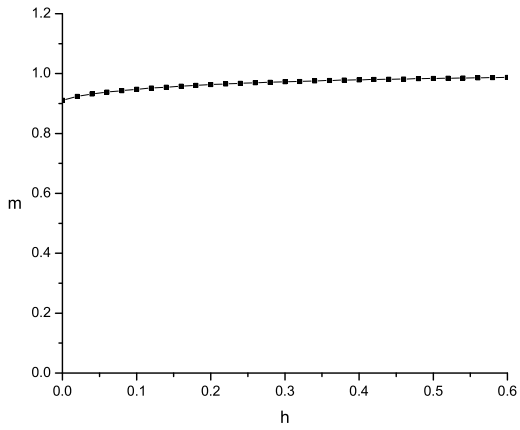


Figura 6.6 Magnetización como función del campo magnético externo (ferromagnetismo, $T < T_0$).

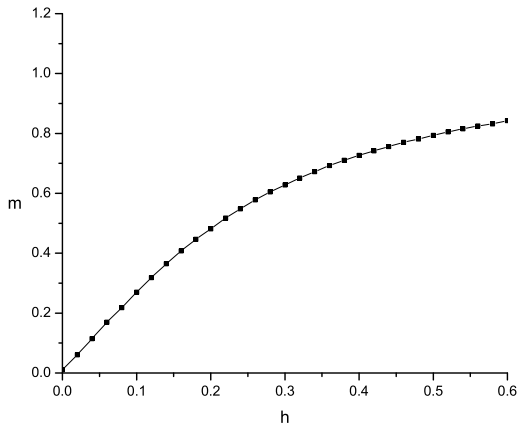


Figura 6.7 Magnetización como función del campo magnético externo (ferromagnetismo, $T > T_0$).

caso ferromagnético, la magnetización en cada una de las subredes cae a cero en $T = T_0$ (Figura 6.8). El calor específico sigue teniendo un máximo, (Figura 6.9), lo cual indica que las correlaciones de corto alcance aún están presentes. En este caso el sistema muestra un ordenamiento a bajas temperaturas, mientras que a temperaturas altas exhibe una fase paramagnética, al igual que el caso ferromagnético.

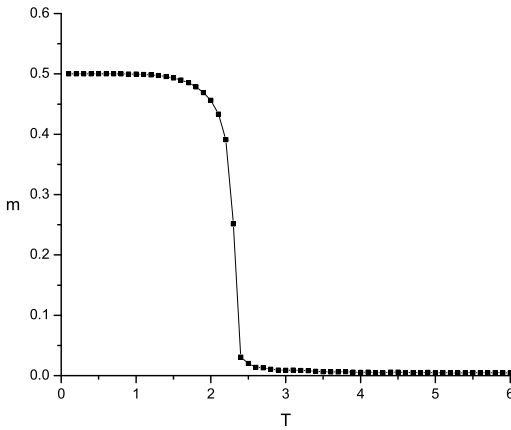


Figura 6.8 Magnetización como función de la temperatura (antiferromagnetismo, $h = 0$).

d.- Ferromagnetismo anisotrópico (red rectangular de Ising).

En este caso, más que estudiar el comportamiento cualitativo del modelo rectangular de Ising, nos gustaría analizar el comportamiento de la temperatura crítica cuando una de las interacciones entre los espines tiende a cero, es decir cuando se pasa del sistema bidimensional al unidimensional. De acuerdo con el trabajo de C. H. Chang [Chang, C. H., 1952], en ausencia de campo magnético externo y por abajo de la temperatura crítica,

RESULTADOS

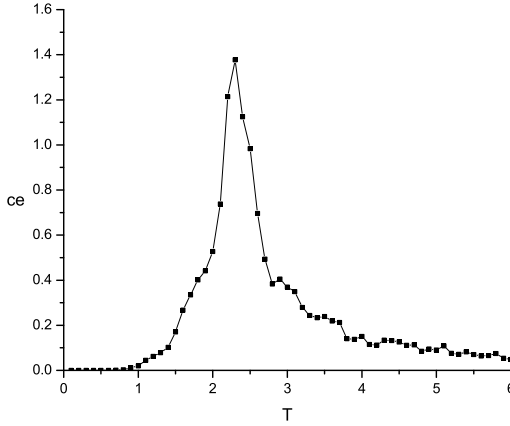


Figura 6.9 Calor específico como función de la temperatura (antiferromagnetismo, $h = 0$).

se satisface la relación:

$$(x_v + 1)(x_h + 1) < 2 \quad (6.1)$$

donde $x_v = \exp(-2J_v)$, $x_h = \exp(-2J_h)$ y tanto J_v como J_h están expresadas en unidades de $K_b T$. Para obtener la temperatura crítica en términos de las constantes de interacción tendríamos que resolver una ecuación trascendental lo cual no es posible, pero lo que si podemos hacer es analizar la relación que existe entre J_h y J_v en el punto crítico. Para ello, el programa se corrió para diferentes valores de una de las interacciones, J_h , mientras que la otra, J_v , permaneció constante. Para cada valor de J_v se identificó el valor de J_h para el cual por primera vez $m < 0.100000$, viniendo de valores pequeños de J_h . Los resultados obtenidos se muestran en la Figura 6.10. Los puntos unidos por una línea punteada corresponden a los resultados de la simulación mientras que la línea continua se obtuvo a partir del resultado de Chang, ecuación (6.1), [Chang, C. H., 1952]. Los

resultados de la simulación concuerdan bastante bien con los reportados por Chang por lo que, al igual que en ese caso, la temperatura crítica debe tender a cero cuando la constante de interacción J_v tiende a cero, obteniéndose así el resultado de Ising para el caso unidimensional.

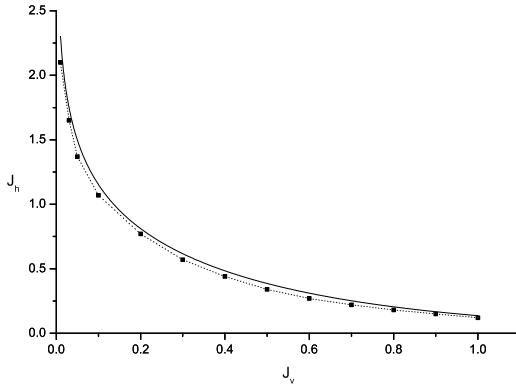


Figura 6.10 Constante de interacción horizontal como función de la constante de interacción vertical en el punto crítico. Los puntos unidos por la línea punteada corresponden a los resultados de la simulación mientras que la línea continua se obtuvo a partir del resultado de Chang, ecuación (6.1).

e.- Sistema con frustración.

El sistema estudiado se muestra en la Figura 5.2, en la cual se observa que todas las plaquetas están frustradas. El comportamiento encontrado concuerda con lo esperado en el sentido de que, por abajo de una cierta temperatura, el sistema pasa por una serie de estados recurrentemente, lo cual indica que es incapaz de alcanzar su estado base y queda atrapado en este conjunto de estados. El número de estados crece con el tamaño del sistema, por lo que para apreciar adecuadamente

este efecto, no es recomendable considerar redes muy grandes. En el intervalo de temperaturas altas el sistema muestra una fase paramagnética, es decir, existe una cierta temperatura que es capaz de desordenar al sistema aún cuando las plaquetas estén frustradas. Es interesante estudiar este sistema en el caso en que no todas las plaquetas estén frustradas, lo cual puede hacerse sustituyendo enlaces positivos por negativos o bien aumentando el tamaño del sistema en proporción diferente a aquella en que aumentan los enlaces negativos.

Para terminar, mencionaremos algunos sistemas que pueden estudiarse utilizando este programa después de implementar algunas modificaciones. En el caso ferromagnético puede investigarse el efecto de un campo magnético externo variable, ya sea periódico o aleatorio, mientras que en el caso antiferromagnético puede introducirse un campo externo que puede ser constante o variable. Otros modelos de interés son el modelo de Ising en tres dimensiones, modelos en donde aparecen constantes de interacción aleatorias, como en los vidrios de espín, o modelos que involucren cambios en el número de estados internos de los espines. También es posible modelar otros sistemas como emulsiones, micelas, fluidos multicomponentes y aleaciones así como realizar estudios cualitativos de propiedades de escalamiento de tamaño finito. Para todos estos sistemas es posible además, analizar correlaciones entre los valores de las observables así como determinar los valores más adecuados de los parámetros que intervienen tanto en la termalización como en el cálculo de observables.

Es importante mencionar que, además de los resultados numéricos arriba presentados, el programa implementado permite visualizar algunos aspectos cualitativos del comportamiento cooperativo cerca y lejos del punto crítico tales como fluctuaciones, formación de dominios, distancias de correlación, efectos de un campo magnético externo en el caso ferromagnético, la aparición de una transición de fase de primer orden, efectos de frustración así como una reducción dimensional causada por anisotropías.

BIBLIOGRAFÍA

- [1] BAXTER, R. J. (1972). "Partition function of the eight-vertex lattice model". *Ann. Phys. (N. Y.)* **70** pp. 193:228.
- [2] BERLIN, T. H. & M. Kac. (1952). "The spherical model of a ferromagnet". *Phys. Rev.* **86** pp. 821:835
- [3] BINDER, K. (1979). *Introduction, Theory and technical aspects of Monte Carlo simulations*. En: Monte Carlo methods in statistical physics. Berlín, Heidelberg, Nueva York: Springer. pp. 1:45
- [4] BINDER, K. & D. W. Heermann. (2002). *Monte Carlo simulation in statistical physics. An introduction*. Berlín, Heidelberg: Springer.
- [5] BRUSH, S. G. (1967). "History of the Lenz-Ising model". *Rev. Mod. Phys.* **39** pp. 883:893
- [6] CHANG, C. H. (1952). "The spontaneous magnetization of a two-dimensional rectangular Ising model". *Phys. Rev.* **88** pp. 1422:1423
- [7] CHOWDHURY, D. & D. Stauffer. (2000). *Principles of equilibrium statistical mechanics*. Alemania: Wiley-VCH. pp. 538
- [8] EHRENFEST, P. (1959). "Phasenumwandlungen im ueblichen und erweiterten Sinn, classifiziert nach den entsprechenden Singularitaeten des thermodynamischen

BIBLIOGRAFÍA

- Potenciales". En EHRENFEST, P. *Colected Scientific Papers*. Ed. Klein, M. J., Interscience Publishers, North-Holland. pp. 628:632
- [9] FISCHER, K. H. y J. A. Hertz. (1991). *Spin glasses*. Cambridge University Press, Cambridge. pp. 375
- [10] GELOVER-SANTIAGO, A. L. (1990). *Simulación del modelo de Ising con el método de Monte Carlo: una presentación pedagógica*. Coordinación de servicios editoriales, Facultad de Ciencias-UNAM, México.
- [11] GOPAL, E. S. R. (1974). *Statistical mechanics and properties of matter. Theory and applications*. John Wiley and Sons, Nueva York.
- [12] HERTZ, J. (1985). "Disordered systems". *Phys. Scr.* **T10**
- [13] HUANG, K. (1987). *Statistical mechanics*. John Wiley and Sons, Nueva York. pp. 493
- [14] HURD, C. M. (1982). "Varieties of magnetic order in solids". *Contemp. Phys.* **23** pp. 469:493.
- [15] KOONIN, S. E. (1986). *Computational physics*. Addison-Wesley, Red Wood City, CA. pp. 409
- [16] KOSTERLITZ, J. M. & D. J. THOULESS. (1973). "Ordering metastability and phase transitions in two dimensional systems". *J. Phys. C* **6** pp. 1181:1203
- [17] LANDAU, L. D. (1985). "On the theory of phase transitions". En Ter Haar, D. (ed.) *Collected Papers of Landau L. D.*, Pergamon, Oxford. pp. 193:216
- [18] METROPOLIS, N., A. ROSENBLUTH, M. ROSENBLUTH, A. TELLER. & E. TELLER. (1953). "Equation of state calculations by fast computing machines". *J. Chem. Phys.* **21** pp. 1087:1092

BIBLIOGRAFÍA

- [19] NEWMAN, M. E. J. y G. T. Barkema. (1999). *Monte Carlo methods in statistical physics*. Clarendon Press, Oxford. pp. 475
- [20] ONSAGER, L. (1944). “Crystal Statistics I. A two dimensional model with an order-disorder transition”. *Phys. Rev.* **65** pp. 117:149
- [21] PAPON, P., J. LEBOND & P. H. E. MEIJER. (2002). *The physics of phase transitions. Concepts and applications*. Springer, Berlin, Heidelberg. pp. 397
- [22] PARISI, G. (1986). “Spin glasses theory”. En: *Statistical physics. Invited lectures from Statphys 16, the 16th International Conference of Thermodynamics and Statistical Mechanics, Boston University August 11-15, 1986*. Ed. Stanley, H. E. North Holland Physics Publishing, Amsterdam. pp. 312:318
- [23] PATHRIA, R. K. (1993). *Statistical mechanics*. Pergamon Press, Oxford. pp. 529
- [24] PEIRLA, R. (1936). “On Ising’s model of ferromagnetism”. *Proc. Cambridge Phil. Soc.* **32** pp. 477.
- [25] PRESS, W.H., S.A. TEUKOLSKY, W. T. VETTERLING & B. P. FLANNERY. (1988). *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge. pp. 735
- [26] REICHL, L. E. (1980). *A modern course in statistical physics*. University of Texas Press, Austin, TX. pp. 709
- [27] STANLEY, H. E. (1968). “Dependence of critical properties on dimensionality of spins”. *Phys. Rev. Lett* **20** pp. 589:592
- [28] STANLEY, H. E. (1971). *Introduction to phase transitions and critical phenomena*. Clarendon Press, Oxford. pp. 308

BIBLIOGRAFÍA

- [29] WU, F. Y. (1982). “The Pootts model”. *Rev. Mod. Phys.*
54 pp. 235:268

APÉNDICES

A

PROGRAMA ISING.C

/*

El programa ising.c lleva a cabo una simulacion del modelo de Ising bidimensional en los casos ferromagnetico y antiferromagnetico, anisotropico y frustrado, usando el metodo de Monte Carlo segun el algoritmo de Metropolis et. al. [1], y esta basado en el trabajo de S. E. Koonin [2].

El programa utiliza una subrutina para generar numeros pseudoaleatorios, tomada de: "Numerical recipes in C. The art of scientific computing" [3].

La configuracion inicial puede ser uniforme (correspondiente a $T=0$) o al azar (correspondiente a $T=\infty$).

[1] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. y Teller, E., "Equation of state calculations by fast computing machines", J. Chem. Phys. 21 (1953) 1807.

[2] Koonin, S.E., "Computational Physics", (1985), Addison-Wesley, Red Wood City, CA, USA.

[3] Press, W. H., Teukolsky, S. A., Vetterling, W. T. y Flannery, B. P. "Numerical recipes in C. The art of scientific computing", (1988), Cambridge University Press, EU.

*/

```
#include <math.h>
#include <graphics.h>
#include <c:\tc\alea1.h>
```

```
#define Nmax 150
```

```
#define M 15
```

```
/*
```

PROGRAMA ISING.C

```

Nmax:maximo numero de columnas y renglones en la red
M:maximo numero de grupos y maximo tamaño de los grupos
*/

long    s;
/*
s:semilla para generar los numeros pseudoaleatorios
*/

int     modelo, nx, ny, n_espines, nterm, n_grupos, n1_grupos=0,
        tam, frec, grupo, mas_grupos=0, acept,
        espines[Nmax+1][Nmax+1], gdriver = DETECT,
        gmode, errorcode;
/*
modelo:modelo que se va a simular
nx:longitud horizontal de la red (numero de columnas)
ny:longitud vertical de la red (numero de renglones)
n_espines: numero total de espines en la red
nterm:numero de pasos de termalizacion
n_grupos:numero de grupos que se forman para realizar la estadistica
n1_grupos:variable auxiliar para n_grupos
tam:tamaño de los grupos
frec:frecuencia de muestreo
grupo:contador para los grupos
mas_grupos:grupos adicionales que se forman durante una misma corrida
acept:numero de cambios aceptados en un paso de Monte Carlo
espines:matriz que contiene las variables de espin
gdriver, gmode, errorcode:variables que inicializan el modo grafico
*/

double  h, J, Jv, Jh, T, energ, mag, sig_e, sig_m, e_grupo,
        m_grupo, e_grupo2, m_grupo2, r[7][4], e, m, ce, sm,
        energ1[M], energ2[M], magnet1[M], magnet2[M], ce1[M],
        sm1[M],sige[M], sigm[M], e2_max, m2_max, ce_max,
        sm_max, sige_max, sigm_max;
/*
h:campo magnetico
J:constante de interaccion entre los espines
Jv:constante de interaccion entre los espines en la direccion vertical
Jh:constante de interaccion entre los espines en la direccion horizontal
T:temperatura del sistema (T=1/J)
energ:energia correspondiente a una configuracion de la red
mag:magnetizacion correspondiente a una configuracion de la red
sig_e:error asociado al calculo de la energia dentro de un grupo

```

PROGRAMA ISING.C

```
sig_m:error asociado al calculo de la magnetizacion dentro de un grupo
e_grupo:promedio de la energia dentro de un grupo
m_grupo:promedio de la magnetizacion dentro de un grupo
e_grupo2:promedio del cuadrado de la energia dentro de un grupo
m_grupo2:promedio del cuadrado de la magnetizacion dentro de un grupo
r:matriz que contiene los posibles valores del cambio en el Hamiltoniano
e:valor promedio de la energia durante la simulacion
m:valor promedio de la magnetizacion durante la simulacion
ce:valor promedio del calor especifico durante la simulacion
sm:valor promedio de la susceptibilidad magnetica durante la simulacion
energ1:vector en donde se almacenan los valores normalizados de energ
energ2:vector en donde se almacenan los valores normalizados de prom_e
magnet1:vector en donde se almacenan los valores normalizados de mag
magnet2:vector en donde se almacenan los valores normalizados de prom_m
ce1:vector que contiene los valores normalizados de ce
sm1:vector que contiene los valores normalizados de sm
sige:vector en donde se almacenan los valores de sig_e
sigm:vector en donde se almacenan los valores de sig_m
e2_max:valor maximo de energ2
m2_max:valor maximo de magnet2
ce_max:valor maximo de ce1
sm_max:valor maximo de sm1
sige_max:valor maximo de sige
sigm_max:valor maximo de sigm
*/

void inicializa(void);
void termaliza(void);
void montecarlo_1_2(void);
void montecarlo_3(void);
void montecarlo_4(void);
void despliega(void);
void observables_1(void);
void observables_2(void);
void observables_3(void);
void observables_4(void);
void calculos(void);
void grupos(void);
void calculos1(void);
void ordena(double vector[M], int kk);

/***** EMPIEZA EL PROGRAMA PRINCIPAL *****/
/*
```


PROGRAMA ISING.C

```
printf("1.-Ferromagnetismo\n");
printf("2.-Antiferromagnetismo\n");
printf("3.-Sistema anisotropico\n");
printf("4.-Sistema frustrado: ");
scanf("%d", &modelo);
}
while (modelo<=0 || modelo>4);
do
{
printf("\nIntroduzca la semilla de los numeros aleatorios\n");
printf("(entero negativo de hasta nueve digitos):          ");
scanf("%ld", &s);
}
while (s>=0 || s<-999999999);

inicializa();

registerbgidriver(EGAVGA_driver);
initgraph(&gdriver, &gmode, "..\\bgi");
errorcode = graphresult();

if (errorcode != grOk) /* SI OCURRIO UN ERROR */
{
printf("Graphics error: %s\n", grapherrormsg(errorcode));
printf("Press any key to halt:");
getch();
}

switch (modelo)
{
case 1:
gotoxy(28,1);
printf("FERROMAGNETISMO");
break;
case 2:
gotoxy(26,1);
printf("ANTIFERROMAGNETISMO");
break;
case 3:
gotoxy(26,1);
printf("SISTEMA ANISOTROPICO");
break;
default:
gotoxy(27,1);
```

PROGRAMA ISING.C

```
        printf("SISTEMA FRUSTRADO ");
    break;
}
gotoxy(1,4);
printf("nx=%d", nx);
gotoxy(1,5);
printf("ny=%d", ny);
switch (modelo)
{
    case 1:
        gotoxy(1,6);
        printf("J=%lf", J);
        gotoxy(1,7);
        printf("T=%lf", T);
    break;
    case 2:
        gotoxy(1,6);
        printf("J=%lf", J);
        gotoxy(1,7);
        printf("T=%lf", T);
    break;
    case 3:
        gotoxy(1,6);
        printf("Jv=%lf", Jv);
        gotoxy(1,7);
        printf("Jh=%lf", Jh);
    break;
    default:
        gotoxy(1,6);
        printf("J=%lf", J);
        gotoxy(1,7);
        printf("T=%lf", T);
    break;
}
gotoxy(1,8);
printf("h=%lf", h);
gotoxy(1, 10);
printf("grupos=%d", n_grupos);
gotoxy(1, 11);
printf("tamano=%d", tam);
gotoxy(1, 12);
printf("frecuencia=%d", frec);
gotoxy(25,2);
printf("Configuracion inicial");
```

PROGRAMA ISING.C

```

despliega();
delay(1500);
gotoxy(25,2);
printf("                ");
termaliza();
if (kbhit())
{
    gotoxy(1,25);
    printf("SIMULACION INTERRUMPIDA");
    delay(1000);
    exit(0);
}
gotoxy(20,2);
printf("                ");
gotoxy(25,3);
printf("Fin de la termalizacion");
delay(1500);
gotoxy(25,3);
printf("                ");
gotoxy(25,2);
printf("Calculando observables");
grupos();
if (kbhit())
{
    gotoxy(1,25);
    printf("SIMULACION INTERRUMPIDA");
    delay(1000);
    exit(0);
}
calculos1();
gotoxy(25,2);
printf("                ");
n1_grupos=n_grupos;
uno: gotoxy(1,25);
printf("*** PARA CONTINUAR OPRIMA UNA TECLA ***");
getch();
if (n_grupos <15)
{
    gotoxy(1,25);
    printf("Cuantos grupos mas quiere formar?");
    printf("<=0 para concluir la simulacion): ");
    scanf("%d", &mas_grupos);
    gotoxy(1,25);
    printf("                ");

```

PROGRAMA ISING.C

```

if (mas_grupos>0 && (n_grupos+mas_grupos<15))
{
    if (e2_max!=0)
        for (k=0; k<n_grupos; k++)
            energ2[k]*=e2_max;
    if (m2_max!=0)
        for (k=0; k<n_grupos; k++)
            magnet2[k]*=m2_max;
    if (ce_max!=0)
        for (k=0; k<n_grupos; k++)
            ce1[k]*=ce_max;
    if (sm_max!=0)
        for (k=0; k<n_grupos; k++)
            sm1[k]*=sm_max;
    if (sige_max!=0)
        for (k=0; k<n_grupos; k++)
            sige[k]*=sige_max;
    if (sigm_max!=0)
        for (k=0; k<n_grupos; k++)
            sigm[k]*=sigm_max;
    n_grupos+=mas_grupos;
    gotoxy(1,10);
    printf("grupos=%d", n_grupos);
    gotoxy(1,22);
    printf("                ");
    gotoxy(1,23);
    printf("                                ");
    gotoxy(1,24);
    printf("                                ");
    gotoxy(25,2);
    printf("Calculando observables");
    grupos();
    if (kbhit())
    {
        gotoxy(1,25);
        printf("SIMULACION INTERRUMPIDA");
        delay(1000);
        exit(0);
    }
    calculos1();
    gotoxy(25,2);
    printf("                                ");
    n1_grupos=n_grupos;
    if (n_grupos<15) goto uno;
}

```


PROGRAMA ISING.C

```
    }
    else
    {
        gotoxy(1,25);
        printf("FIN DE LA SIMULACION");
        delay(1000);
    }
}
else
{
    gotoxy(1,25);
    printf("
");
    gotoxy(1,25);
    printf("FIN DE LA SIMULACION");
    delay(1000);
}
}
/***** FIN DEL PROGRAMA PRINCIPAL *****/

/** EMPIEZA EL PROCEDIMIENTO INICIALIZA **/
/*
En el PROCEDIMIENTO INICIALIZA se lleva a cabo la lectura de los datos
que determinan el tamaño y el estado del sistema, se inicializa la
red de espines y se leen los parámetros que intervienen en el cálculo
de los promedios.
*/
void inicializa(void)
{
    int    i, j, k, k1, ci;
    /*
i:contador para el número de renglones de la red
j:contador para el número de columnas de la red
k,k1:contadores
ci:indica la condición inicial para la red (al azar u ordenada)
*/

    do
    {
        printf("\nLongitud horizontal de la red (nx<=%d):  ", Nmax);
        scanf("%d", &nx);
    }
    while (nx<=0 || nx>Nmax);
```

PROGRAMA ISING.C

```

do
{
    printf("Longitud vertical de la red (ny<=%d):      ", Nmax);
    scanf("%d", &ny);
}
while (ny<=0 || ny>Nmax);
switch (modelo)
{
    case 1:
        do
        {
            printf("\nTemperatura (T):                ");
            scanf("%lf", &T);
            J=1/T;
        }
        while (T<=0);
        break;
    case 2:
        do
        {
            printf("\nTemperatura (T):                ");
            scanf("%lf", &T);
            J=-1/T;
        }
        while (T<=0);
        break;
    case 3:
        do
        {
            printf("\nConstante de interaccion vertical (Jv): ");
            scanf("%lf", &Jv);
            printf("Constante de interaccion horizontal (Jh): ");
            scanf("%lf", &Jh);
        }
        while (Jv==0 && Jh==0);
        break;
    default:
        do
        {
            printf("\nConstante de interaccion (J):                ");
            scanf("%lf", &J);
            T=1/fabs(J);
        }
        while (J==0);
}

```

PROGRAMA ISING.C

```

    break;
}
printf("Campo magnetico externo (h):           ");
scanf("%lf", &h);
printf("\nNumero de pasos de termalizacion (nterm):   ");
scanf("%d", &nterm);
do
{
    printf("Numero de grupos(n_grupos<=15):           ");
    scanf("%d", &n_grupos);
}
while (n_grupos<=0 || n_grupos>15);
do
{
    printf("Tamano de los grupos(tam<=15):           ");
    scanf("%d", &tam);
}
while (tam<=0 || tam>15);
do
{
    printf("Frecuencia de muestreo (freq):           ");
    scanf("%d", &frec);
}
while (frec<=0);

n_espines=nx*ny;
do
{
    printf("\nConfiguracion inicial: (1)ordenada (T=0) (2)al azar
           (T=infinito)? ");
    scanf("%d", &ci);
}
while (ci!=1 && ci!=2);
if (ci==1)
{
    if (modelo==1 || modelo==3)
        for (i=1; i<=ny; i++)
            for (j=1; j<=nx; j++)
                espines[i][j]=1;
    else
        if (modelo==2)
            for (i=1; i<=ny; i++)
                for (j=1; j<=nx; j++)
                    if ((i%2==0 && j%2==0)||((i+1)%2==0 && (j+1)%2==0))

```

PROGRAMA ISING.C

```

        espines[i][j]=1;
    else
        espines[i][j]=-1;
else
{
    printf("La configuracion correspondiente a T=0 no es ");
    printf("unica, se utilizara \nuna configuracion al azar");
    delay(2000);
    for (i=1; i<=ny; i++)
        for (j=1; j<=nx; j++)
            if (alea(&s) < 0.5) espines[i][j]=1;
            else espines[i][j]=-1;
}
}
else
    for (i=1; i<=ny; i++)
        for (j=1; j<=nx; j++)
            if (alea(&s) < 0.5) espines[i][j]=1;
            else espines[i][j]=-1;

if (modelo==1 || modelo==2 || modelo==4)
    for (k=1; k<=5; k++)
    {
        r[k][2]=exp(-2*(J*(2*k-6)+h));
        r[k][1]=1/r[k][2];
    }
else
    for (k=1; k<=3; k++)
        for (k1=1; k1<=3; k1++)
        {
            r[k][k1]=exp(-2*(Jv*(2*k-4)+Jh*(2*k1-4)+h));
            r[k+3][k1]=1/r[k][k1];
        }
    mas_grupos=n_grupos;
}
/** FIN DEL PROCEDIMIENTO INICIALIZA **/

/** EMPIEZA EL PROCEDIMIENTO TERMALIZA **/
/*
El PROCEDIMIENTO TERMALIZA lleva a cabo la termalizacion del sistema,
para evitar que los valores de las observables dependan de la condicion
inicial.

```

PROGRAMA ISING.C

```
*/
void termaliza(void)
{

    int    paso;

/*
paso:indica el paso de Monte Carlo en que se encuentra la simulacion
*/

    if (nterm>0)
    {
        paso=1;
        while(paso<nterm+1 && !kbhit())
        {
            if (modelo==1 || modelo==2)
                montecarlo_1_2();
            else
                if (modelo==3)
                    montecarlo_3();
                else
                {
                    montecarlo_4();
                    delay(500);
                }
            if (paso%5==0)
                despliega();
            gotoxy(20,2);
            printf("Termalización: paso %4d de %d", paso, nterm);
            gotoxy(31,3);
            printf(" acept=%5d",accept);
            paso+=1;
        }
    }
}

/** FIN DEL PROCEDIMIENTO TERMALIZA **/

/** EMPIEZA EL PROCEDIMIENTO PASOS DE MONTECARLO **/
/** CASOS FERROMAGNETICO Y ANTIFERROMAGNETICO **/
/*
El PROCEDIMIENTO MONTECARLO_1_2 lleva a cabo los pasos de Montecarlo en
los casos ferromagnetico y antiferromagnetico
*/
void montecarlo_1_2(void)
```

PROGRAMA ISING.C

```

{

    int      i, j, f, im, jm, ip, jp, u, v;
/*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
f:suma de los cuatro espines vecinos de un sitio dado
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
ip:indice del renglon vecino hacia abajo
jp:indice de la columna vecina hacia la derecha
u, v:indices que determinan el valor del cambio en el Hamiltoniano
asociado con la inversion de un espin
*/

    accept=0;
    for (i=1; i<=ny; i++)
    {
        if (i<ny) ip=i+1;
        else ip=1;
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j<nx) jp=j+1;
            else jp=1;
            if (j>1) jm=j-1;
            else jm=nx;
            f=espines[ip][j]+espines[im][j]+
            espines[i][jp]+espines[i][jm];
            u=floor(3+f/2);
            v=(3+espines[i][j])/2;
            if (alea(&s)<r[u][v])
            {
                espines[i][j]*=-1;
                accept+=1;
            }
        }
    }
}

/** FIN DEL PROCEDIMIENTO PASOS DE MONTECARLO ***/
/** CASOS FERROMAGNETICO Y ANTIFERROMAGNETICO ***/

```

PROGRAMA ISING.C

```

/** EMPIEZA EL PROCEDIMIENTO PASOS DE MONTECARLO ***/
/** CASO ANISOTROPICO ***/
/*
El PROCEDIMIENTO MONTECARLO_3 lleva a cabo los pasos de Montecarlo en el
caso anisotropico
*/
void montecarlo_3(void)
{

    int      i, j, f1, f2, im, jm, ip, jp, u, v;
/*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
f1:suma de los dos espines vecinos de un sitio dado, en la misma columna
f2:suma de los dos espines vecinos de un sitio dado, en el mismo renglon
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
ip:indice del renglon vecino hacia abajo
jp:indice de la columna vecina hacia la derecha
u, v:indices que determinan el valor del cambio en el Hamiltoniano
asociado con la inversion de un espin
*/

    accept=0;
    for (i=1; i<=ny; i++)
    {
        if (i<ny) ip=i+1;
        else ip=1;
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j<nx) jp=j+1;
            else jp=1;
            if (j>1) jm=j-1;
            else jm=nx;
            f1=espines[ip][j]+espines[im][j];
            f2=espines[i][jp]+espines[i][jm];
            u=floor(2+f1/2+3*(1-espines[i][j])/2);
            v=floor(2+f2/2);
            if (alea(&s)<r[u][v])
            {

```


PROGRAMA ISING.C

```

else jp=1;
if (j>1) jm=j-1;
else jm=nx;
if ((i%2==0 && j%2==0) || (i%2==1 && j%2==1))
    f=-espines[ip][j]+espines[im][j]+
    espines[i][jp]+espines[i][jm];
else
    f=espines[ip][j]+espines[im][j]+
    espines[i][jp]+espines[i][jm];
u=floor(3+f/2);
v=(3+espines[i][j])/2;
if (alea(&s)<r[u][v])
{
    espines[i][j]*=-1;
    acept+=1;
}
}
}

/** FIN DEL PROCEDIMIENTO PASOS DE MONTECARLO ***/
/** CASO FRUSTRADO ***/

/** EMPIEZA EL PROCEDIMIENTO DESPLIEGA ***/
/*
El PROCEDIMIENTO DESPLIEGA se encarga de dibujar la red de espines en la
pantalla de la computadora, asignando un pixel de color azul a los
espines positivos y un pixel de color gris a los espines negativos.
*/
void despliega(void)
{
    int    i, j;
/*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
*/
    for (i=1; i<=ny; i++)
        for (j=1; j<=nx; j++)
            if (espines[i][j]==1) putpixel(2*j+(280-nx),2*i+(198-ny),9);
            else putpixel(2*j+(280-nx),2*i+(198-ny),15);
}

```

PROGRAMA ISING.C

```

/** FIN DEL PROCEDIMIENTO DESPLIEGA **/

/** EMPIEZA EL PROCEDIMIENTO OBSERVABLES **/
/** CASO FERROMAGNETICO **/
/*
El PROCEDIMIENTO OBSERVABLES_1 lleva a cabo el calculo de las
observables basicas del sistema, energia y magnetizacion, en
el caso ferromagnetico.
*/
void observables_1(void)
{
    int    i, j, im, jm;
/*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
*/

    double  sums;
/*
sums:suma de productos de espines (espines[i][j]*
(espines[im][j]+espines[i][jm]))
*/

    mag=0;
    sums=0;
    for (i=1; i<=ny; i++)
    {
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j>1) jm=j-1;
            else jm=nx;
            sums+=espines[i][j]*(espines[im][j]+espines[i][jm]);
            mag+=espines[i][j];
        }
    }
    if (h==0) mag=fabs(mag);
    energ=(-1)*(J*sums+h*mag);
}

```

PROGRAMA ISING.C

```

/** FIN DEL PROCEDIMIENTO OBSERVABLES **/
/** CASO FERROMAGNETICO **/

/** EMPIEZA EL PROCEDIMIENTO OBSERVABLES **/
/** CASO ANTIFERROMAGNETICO **/
/*
El PROCEDIMIENTO OBSERVABLES_2 lleva a cabo el calculo de las
observables basicas del sistema, energia y magnetizacion, en
el caso antiferromagnetico.
*/
void observables_2(void)
{
    int    i, j, im, jm;
    /*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
*/

    double  sums, smag, smag1, smag2;
    /*
sums:suma de productos de espines (espines[i][j]*(espines[im][j]+
espines[i][jm]))
smag1, smag2:suma de espines en cada una de las dos subredes del
antiferromagneto
smag:promedio de smag1 y smag2
*/

    mag=0;
    sums=0;
    for (i=1; i<=ny; i++)
    {
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j>1) jm=j-1;
            else jm=nx;
            sums+=espines[i][j]*(espines[im][j]+espines[i][jm]);
        }
    }
    energ=(-1)*(J*sums+h*mag);

```

PROGRAMA ISING.C

```

smag1=0;
smag2=0;
for (i=1; i<=ny; i++)
{
    if (i%2==0)
    {
        j=1;
        while (j<=nx)
        {
            smag2+=espines[i][j];
            j+=2;
        }
        j=1;
        while (j<nx)
        {
            smag1+=espines[i][j+1];
            j+=2;
        }
    }
    else
    {
        j=1;
        while (j<=nx)
        {
            smag1+=espines[i][j];
            j+=2;
        }
        j=1;
        while (j<nx)
        {
            smag2+=espines[i][j+1];
            j+=2;
        }
    }
}
if (h==0)
{
    smag1=fabs(smag1);
    smag2=-fabs(smag2);
}
smag=(smag1-smag2)/2;
mag=smag;
}
/*** FIN DEL PROCEDIMIENTO OBSERVABLES ***/

```

PROGRAMA ISING.C

```

/**/ CASO ANTIFERROMAGNETICO ***/

/**/ EMPIEZA EL PROCEDIMIENTO OBSERVABLES ***/
/**/ CASO ANISOTROPICO ***/
/*
El PROCEDIMIENTO OBSERVABLES_3 lleva a cabo el calculo de las
observables basicas del sistema, energia y magnetizacion, en
el caso anisotropico.
*/
void observables_3(void)
{
    int      i, j, im, jm;
/*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
*/

    double   sums1, sums2;
/*
sums1:suma de productos de espines en la misma columna
(espines[i][j]*espines[im][j])
sums2:suma de productos de espines en el mismo renglon
(espines[i][j]*espines[i][jm])
*/

    mag=0;
    sums1=0;
    sums2=0;
    for (i=1; i<=ny; i++)
    {
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j>1) jm=j-1;
            else jm=nx;
            sums1+=espines[i][j]*espines[im][j];
            sums2+=espines[i][j]*espines[i][jm];
            mag+=espines[i][j];
        }
    }
}

```

PROGRAMA ISING.C

```

    }
    if (h==0) mag=fabs(mag);
    energ=(-1)*(Jv*sums1+Jh*sums2+h*mag);
}
/** FIN DEL PROCEDIMIENTO OBSERVABLES ***/
/** CASO ANISOTROPICO ***/

/** EMPIEZA EL PROCEDIMIENTO OBSERVABLES ***/
/** CASO FRUSTRADO ***/
/*
El PROCEDIMIENTO OBSERVABLES_4 lleva a cabo el calculo de las
observables basicas del sistema, energia y magnetizacion, en
el caso frustrado.
*/
void observables_4(void)
{
    int      i, j, im, jm;
    /*
i:contador para el numero de renglones de la red
j:contador para el numero de columnas de la red
im:indice del renglon vecino hacia arriba
jm:indice de la columna vecina a la izquierda
*/

    double  sums;
    /*
sums:suma de productos de espines
(espines[i][j]*(espines[im][j]+espines[i][jm]))
*/

    mag=0;
    sums=0;
    for (i=1; i<=ny; i++)
    {
        if (i>1) im=i-1;
        else im=ny;
        for (j=1; j<=nx; j++)
        {
            if (j>1) jm=j-1;
            else jm=nx;
            if ((i%2==0 && j%2==0) || (i%2==1 && j%2==1))
                sums+=espines[i][j]*(-espines[im][j]+espines[i][jm]);
        }
    }
}

```

PROGRAMA ISING.C

```

        else
            sums+=espines[i][j]*(espines[im][j]+espines[i][jm]);
            mag+=espines[i][j];
        }
    }
    if (h==0) mag=fabs(mag);
    energ=(-1)*(J*sums+h*mag);
}
/** FIN DEL PROCEDIMIENTO OBSERVABLES ***/
/** CASO FRUSTRADO ***/

/** EMPIEZA EL PROCEDIMIENTO CALCULOS ***/
/*
El PROCEDIMIENTO CALCULOS lleva a cabo el calculo de los promedios
de las observables dentro de los grupos.
*/
void calculos(void)
{
    int    k1;
/*
k1:contador para los elementos de los grupos
*/

    double  prom_e, prom_e2, prom_m, prom_m2, e_max, m_max;
/*
prom_e:promedio de energ dentro de un grupo
prom_e2:promedio del cuadrado de energ dentro de un grupo
prom_m:promedio de magnet dentro de un grupo
prom_m2:promedio del cuadrado de magnet dentro de un grupo
e_max:valor maximo de energ1 dentro de un grupo
m_max:valor maximo de magnet1 dentro de un grupo
*/

    e_grupo=0;
    e_grupo2=0;
    m_grupo=0;
    m_grupo2=0;
    ordena(energ1, tam);
    ordena(magnet1, tam);
    e_max=(fabs(energ1[0])>fabs(energ1[tam-1]))?
    fabs(energ1[0]):fabs(energ1[tam-1]);

```

PROGRAMA ISING.C

```

m_max=(fabs(magnet1[0])>fabs(magnet1[tam-1]))?
fabs(magnet1[0]):fabs(magnet1[tam-1]);
if (e_max!=0)
    for (k1=0; k1<tam; k1++)
        energ1[k1]/=e_max;
if (m_max!=0)
    for (k1=0; k1<tam; k1++)
        magnet1[k1]/=m_max;
for (k1=0; k1<tam; k1++)
{
    e_grupo+=energ1[k1];
    e_grupo2+=pow(energ1[k1], 2);
    m_grupo+=magnet1[k1];
    m_grupo2+=pow(magnet1[k1], 2);
}
prom_e=e_grupo/tam;
prom_e2=e_grupo2/tam;
prom_m=m_grupo/tam;
prom_m2=m_grupo2/tam;
ce=prom_e2-pow(prom_e, 2);
if (e_max!=0)
    ce*=pow(e_max,2);
sm=prom_m2-pow(prom_m, 2);
if (m_max!=0)
    sm*=pow(m_max,2);
sig_e=sqrt(ce/tam);
sig_m=sqrt(sm/tam);
if (e_max!=0)
    energ2[grupo-1]=prom_e*e_max;
if (m_max!=0)
    magnet2[grupo-1]=prom_m*m_max;
ce1[grupo-1]=ce;
sm1[grupo-1]=sm;
sige[grupo-1]=sig_e;
sigm[grupo-1]=sig_m;
gotoxy(57,4);
printf("grupo %d" ,grupo);
gotoxy(57,5);
printf("                                ");
gotoxy(57,6);
printf("                                ");
gotoxy(57,7);
printf("                                ");
gotoxy(57,8);

```


PROGRAMA ISING.C

```

printf("                ");
gotoxy(57,5);
printf("e=%lf+-(%lf)", energ2[grupo-1]/n_espines,
      sig_e/n_espines);
gotoxy(57,6);
printf("m=%lf+-(%lf)", magnet2[grupo-1]/n_espines,
      sig_m/n_espines);
gotoxy(57,7);
printf("ce=%lf", ce1[grupo-1]/n_espines);
gotoxy(57,8);
printf("sm=%lf", sm1[grupo-1]/n_espines);
}
/** FIN DEL PROCEDIMIENTO CALCULOS ***/

```

```

/** EMPIEZA EL PROCEDIMIENTO CALCULOS1 ***/

```

```

/*

```

El PROCEDIMIENTO CALCULOS1 lleva a cabo el calculo de los promedios de las observables sobre los grupos, para obtener los valores finales de estas.

```

*/

```

```

void calculos1(void)

```

```

{

```

```

    int    k1;

```

```

/*

```

k1:contador para el numero de grupos

```

*/

```

```

    double  sum_e, sum_e2, sum_m, sum_m2, sumsig_e,
            sumsig_m, sum_ce, sum_ce2, sum_sm,
            sum_sm2, sig_e1, sig_m1, sig_ce, sig_sm;

```

```

/*

```

sum_e:suma de los valores de prom_e

sum_e2:suma de los cuadrados de prom_e

sum_m:suma de los valores de prom_m

sum_m2:suma de los cuadrados de prom_m

sumsig_e:suma de los cuadrados de sig_e

sumsig_m:suma de los cuadrados de sig_m

sum_ce:suma de los valores de ce1

sum_ce2:suma de los cuadrados de ce1

sum_sm:suma de los valores de sm1

PROGRAMA ISING.C

```

sum_sm2:suma de los cuadrados de sm1
sig_e1:promedio de los valores de sumsig_e
sig_m1:promedio de los valores de sumsig_m
sig_ce:promedio de los valores de sig_ce
sig_sm:promedio de los valores de sig_sm
*/

sum_e=0;
sum_e2=0;
sumsig_e=0;
sum_m=0;
sum_m2=0;
sumsig_m=0;
sum_ce=0;
sum_ce2=0;
sum_sm=0;
sum_sm2=0;
ordena(energ2, n_grupos);
ordena(magnet2, n_grupos);
ordena(ce1, n_grupos);
ordena(sm1, n_grupos);
ordena(sige, n_grupos);
ordena(sigm, n_grupos);
e2_max=(fabs(energ2[0])>fabs(energ2[n_grupos-1]))?
fabs(energ2[0]):fabs(energ2[n_grupos-1]);
m2_max=(fabs(magnet2[0])>fabs(magnet2[n_grupos-1]))?
fabs(magnet2[0]):fabs(magnet2[n_grupos-1]);
ce_max=ce1[n_grupos-1];
sm_max=sm1[n_grupos-1];
sige_max=sige[n_grupos-1];
sigm_max=sigm[n_grupos-1];
if (e2_max!=0)
    for (k1=0; k1<n_grupos; k1++)
        energ2[k1]/=e2_max;
if (m2_max!=0)
    for (k1=0; k1<n_grupos; k1++)
        magnet2[k1]/=m2_max;
if (ce_max!=0)
    for (k1=0; k1<n_grupos; k1++)
        ce1[k1]/=ce_max;
if (sm_max!=0)
    for (k1=0; k1<n_grupos; k1++)
        sm1[k1]/=sm_max;
if (sige_max!=0)

```

PROGRAMA ISING.C

```

    for (k1=0; k1<n_grupos; k1++)
        sige[k1]/=sige_max;
if (sige_max!=0)
    for (k1=0; k1<n_grupos; k1++)
        sigm[k1]/=sigm_max;
for (k1=0; k1<n_grupos; k1++)
{
    sum_e+=energ2[k1];
    sum_e2+=pow(energ2[k1],2);
    sum_m+=magnet2[k1];
    sum_m2+=pow(magnet2[k1],2);
    sumsig_e+=pow(sige[k1],2);
    sum_ce+=ce1[k1];
    sum_ce2+=pow(ce1[k1],2);
    sumsig_m+=pow(sigm[k1],2);
    sum_sm+=sm1[k1];
    sum_sm2+=pow(sm1[k1],2);
}
e=sum_e/n_grupos;
sig_e=sum_e2/n_grupos-pow(e,2);
sig_e=sqrt(sig_e/n_grupos);
sig_e1=sqrt(sumsig_e)/n_grupos;
ce=sum_ce/n_grupos;
sig_ce=sum_ce2/n_grupos-pow(ce,2);
sig_ce=sqrt(sig_ce/n_grupos);

m=sum_m/n_grupos;
sig_m=sum_m2/n_grupos-pow(m,2);
sig_m=sqrt(sig_m/n_grupos);
sig_m1=sqrt(sumsig_m)/n_grupos;
sm=sum_sm/n_grupos;
sig_sm=sum_sm2/n_grupos-pow(sm,2);
sig_sm=sqrt(sig_sm/n_grupos);

e/=n_espines;
sig_e/=n_espines;
if (e2_max!=0)
{
    e*=e2_max;
    sig_e*=e2_max;
}
sig_e1/=n_espines;
if (sige_max!=0)
    sig_e1*=sige_max;

```

PROGRAMA ISING.C

```

m/=n_espines;
sig_m/=n_espines;
if (m2_max!=0)
{
    m*=m2_max;
    sig_m*=m2_max;
}
sig_m1/=n_espines;
if (sigm_max!=0)
    sig_m1*=sigm_max;
ce/=n_espines;
sig_ce/=n_espines;
if (ce_max!=0)
{
    ce*=ce_max;
    sig_ce*=ce_max;
}
sm/=n_espines;
sig_sm/=n_espines;
if (sm_max!=0)
{
    sm*=sm_max;
    sig_sm*=sm_max;
}
gotoxy(1,22);
printf("Totales");
gotoxy(1,23);
printf("e=%lf+-(%lf,%lf), ", e, sig_e, sig_e1);
printf("m=%lf+-(%lf,%lf)", m, sig_m, sig_m1);
gotoxy(1,24);
printf("ce=%lf+-(%lf), ", ce, sig_ce);
printf("sm=%lf+-(%lf), ", sm, sig_sm);
}
/**/ FIN DEL PROCEDIMIENTO CALCULOS1 ***/

/**/ EMPIEZA EL PROCEDIMIENTO GRUPOS ***/
/*
En el PROCEDIMIENTO GRUPOS se forman los grupos para llevar a cabo
el calculo de los promedios de las observables.
*/
void grupos(void)
{

```

PROGRAMA ISING.C

```
int      paso, k1;
/*
k1:contador para los elementos de los grupos
*/

grupo=n1_grupos+1;
while(grupo<=n_grupos && !kbhit())
{
    k1=0;
    paso=1;
    while(paso<frec*tam+1 && !kbhit())
    {
        if (modelo==1 || modelo==2)
            montecarlo_1_2();
        else
            if (modelo==3)
                montecarlo_3();
            else
            {
                montecarlo_4();
                delay(500);
            }
        gotoxy(31,3);
        printf("accept=%5d",accept);
        despliega();
        if (paso%frec==0)
        {
            switch (modelo)
            {
                case 1:
                    observables_1();
                    break;
                case 2:
                    observables_2();
                    break;
                case 3:
                    observables_3();
                    break;
                default:
                    observables_4();
                    break;
            }
            energ1[k1]=energ;
            magnet1[k1]=mag;
        }
    }
}
```

PROGRAMA ISING.C

```

        k1+=1;
    }
    paso+=1;
}
gotoxy(31,3);
printf("                ");
calculos();
grupo+=1;
}
}
/** FIN DEL PROCEDIMIENTO GRUPOS **/

/** EMPIEZA EL PROCEDIMIENTO ORDENA **/
/*
El PROCEDIMIENTO ORDENA se utiliza para ordenar los datos antes de
calcular los promedios a fin de minimizar los errores numericos
asociados.
*/
void ordena(double vect[M], int k2)
{
    int    i, k, q;
    double  x0;

    for (k=0; k<k2-1; k++)
    {
        x0=vect[k];
        q=-1;
        for (i=k+1; i<k2; i++)
            if (vect[i]<x0)
            {
                x0=vect[i];
                q=i;
            }
        if (q>0)
        {
            for (i=q; i>k; i--)
                vect[i]=vect[i-1];
            vect[k]=x0;
        }
    }
}
/** FIN DEL PROCEDIMIENTO ORDENA **/

```

B

SUBROUTINA ALEA1.H

```
/*
  La subrutina alea1.h genera numeros aleatorios en el intervalo
  (0,1) con un algoritmo tomado de "Numerical recipes in C. The art
  of scientific computing" [1].

  [1] Press, W. H., Teukolsky, S. A., Vetterling, W. T. y
      Flannery, B. P., "Numerical recipes in C. The art of
      scientific computing", (1988), Cambridge University Press, EU.
*/

#define IA 16807
#define IM 2147483647
#define AM (1.0/IM)
#define IQ 127773
#define IR 2836
#define NTAB 32
#define NDIV (1+(IM-1)/NTAB)
#define EPS 1.2e-7
#define RNMX (1.0-EPS)

double alea(long *idum)
{
    int j;
    long k;
    static long iy=0;
    static long iv[NTAB];
    double na,temp;

    if(*idum <= 0 || !iy)
    {
```

SUBROUTINA ALEA1.H

```

if(-(*idum) < 1)
  *idum = 1;
else
  *idum = -(*idum);

for(j = NTAB+7; j>=0; j--)
  {
    k = (*idum)/IQ;
    *idum = IA*((*idum)-k*IQ)-IR*k;

    if(*idum < 0) *idum += IM;
    if(j < NTAB) iv[j] = *idum;
  }
  iy = iv[0];
}
k = (*idum)/IQ;
*idum = IA*((*idum)-k*IQ)-IR*k;
if(*idum < 0) *idum += IM;
j = iy/NDIV;
iy = iv[j];
iv[j] = *idum;
temp = AM*iy;
if(temp > RNMX) na = RNMX;
else na = temp;
return na;
}
/* Fin de la subrutina alea1.h */

```