Departamento de Física
UNIVERSIDADE DE AVEIRO

Modelação em Física Estatística

2019.07.2

Recurso

Primeira Parte

1. Pretende-se fazer uma simulação de um gás ideal bidimensional de Fermiões, em que as partículas estão confinadas a um quadrado de lado L, usando o algoritmo *de Metropolis* no ensemble Canónico (no qual o sistema se encontra a temperatura T com número de partículas fixo). Os estados de um gás de Fermiões podem ser especificados pelo número de Fermiões, $n_{\vec{k}}$ que têm um dado vetor de onda, $k_{x,y}=\frac{\pi}{L}n_{x,y}$ e $n_{x,y}=1,2,\ldots,\infty$ com a restrição de que não pode existir mais que uma partícula com um dado vetor de onda (principio de exclusão de Pauli). A energia de um Fermião com vetor de onda $\vec{k}$ é $E=\frac{\hbar^2 k^2}{2m}$ onde m é a massa das partículas e $\hbar=\frac{h}{2\pi}$ é a constante de Planck dividida por 2π. Use unidade de energia, $u_E=\frac{\hbar^2\pi^2}{2mL^2}$ e uma unidade de temperatura, $u_T=u_E/k_B$.

a) Construa uma função
`function [Emedio,E2medio, nkmed]=metropolisFermioes(T,nequi,nmedidas,N,nmax)` que simula o sistema usando o algoritmo de Metropolis.

b) Considerando `nequi=5000`, `nmedidas=20000` e `N=100` partículas obtenha resultados de simulação para 30 temperaturas entre 3 e 300. Compare os resultados obtidos para `Emedio-2N` e para a fugacidade z (calculada a partir de `nkmedio(1)` usando a relação $\langle n_1\rangle=\frac{1}{z^{-1}+1}$) com os valores esperados das mesmas quantidades para um gás de Fermiões bidimensional (nas unidades definidas) : $\langle E\rangle=-\frac{\pi}{4}T^2 Li_2(1+z)$ e $1+z=\exp(\frac{4N}{\pi T})$ onde $Li_2(x)$ é a função di-logaritmica cujos valores podem ser obtidos em matlab fazendo `dilog(x)` . Faça gráficos que comparem os valores obtidos com os valores esperados. Calcule também a partir das simulações a capacidade térmica, C_V , do sistema em função da temperatura. Faça um gráfico em função da temperatura desta quantidade. Se a função desenvolvida em b) não estiver a funcionar corretamente assuma esta função e faça um script onde faz os cálculos pedidos nesta alínea.

Segunda Parte

2.

a) A função `listv_rede_triangular(L)` cria uma rede triangular em que cada sítio tem 6 vizinhos, com condições fronteira periódicas:

```
function [lv,nv]=listv_rede_triangular(L)
N=L*L;
lv=zeros(N,6); nv=ones(N,1)*6;

for ix=1:L
   for iy=1:L
      i=ix+(iy-1)*L;
      i1=mod(ix,L)+1+(iy-1)*L; lv(i,1)=i1;
      i2=ix+mod(iy,L)*L; lv(i,2)=i2;
      i3=mod(ix-2,L)+1+mod(iy,L)*L;lv(i,3)=i3;
      i4=mod(ix-2,L)+1+(iy-1)*L;lv(i,4)=i4;
      i5=ix+mod(iy-2,L)*L;lv(i,5)=i5;
      i6=mod(ix,L)+1+mod(iy-2,L)*L;lv(i,6)=i6;
   end

end
```

Explique porque as coordenadas (x,y) dos vértices da rede são dadas por `x=ix+(iy-1)*cos(pi/3); y=(iy-1)*sin(pi/3)+1;` e faça um gráfico dos pontos da rede.

b) Admitindo que os vértices da rede estão presentes com probabilidade *p* mostre numericamente, considerando sistemas de tamanho variável $L=8,16,32$ *e* 64 , diferentes valores de $p=0.4:0.01:0.8$ e 1000 amostras, que os resultados para o parâmetro de ordem N_∞/N *onde* N_∞ é o tamanho do maior agregado e N é o número total de vértices *da rede* são compatíveis com a existência de uma transição de percolação num sistema infinito para um valor de $p_c=0.5$.

c) Faça um gráfico do número de vértices que pertencem ao maior agregado presente no sistema, N_∞ para $p_c=0.5$ em função do tamanho do lado do sistema, *L*, em escala log-log e determine deste modo a dimensão fractal dos agregados, d_f . Represente o ajuste no mesmo gráfico.

d) Explique justificando porque se espera que, para sistemas muito grandes, $d_f=91/48<2$