3rd Report - *Montecarlo Methods*

Simulació de Sistemes Nanomètrics - *Nanociència i Nanotecnologia - 22/23*

Daniel Bedmar Romero 1565494

1 Description of the problem

I have chosen to modify the ising method code so it automatically stops if the Energy and the Magnetization of the system converge or if the code has been running for too long (the user can decide the maximum amounts of Monte Carlo steps the code will try).

For doing this I've decided that a good convergence would be achived if the energy of the systems does not fluctuate more than $2K_BT$ and the magnetization more than 0.075.

2 The code

I made the code so the last 200 hundreds values of energy and magnetization are put in a list and the difference between the maximum and the minimum are compared with the values set to achieve convergence. The code can be seen in the Figure bellow.

```
#Perform the MC iterations
Econv=False
Mconv=False
while(((Econv == False) or (Mconv == False)) and (len(E) <= msrmnt)): #or (len(E) <= msrmnt)):
        #call MC calculation
        mcmove(config, N, 1.0/temp)
        #update variables
        t=t+1                          # update MC step
        Ene = calcEnergy(config)/(N*N)     # calculate average energy
        Mag = calcMag(config)/(N*N)        # calculate average magnetisation
        #Update
        step.append(t)
        E.append(Ene)
        M.append(Mag)
        #plot certain configurations
        if t%10 == 0:
                print('\nMC step=',t,' Energy=',Ene,' M=',Mag)
                print(config)
                configPlot(f, config, t, N)
                print("Econv:",Econv,"Mconv:",Mconv, "Len(E)", len(E))
        if (len(E)>=200):
                LastE = E[-200:]                 # take the last 100 energy values
                LastM = M[-200:]                 # take the last 100 magnetization values
                DifE = max(LastE)-min(LastE)
                #print(DifE,"< que",3/temp)
                DifM = max(LastM)-min(LastM)
                #print(DifM,"< que", 0.1)
                if (DifE < 2*temp):
                        Econv = True
                if (DifM < 0.075):
                        Mconv = True
#Print end
print('\nSimulation finished after',t, 'MC steps')
print("The simulation converged before the maxium", msrmnt, "steps you set")
```

Figure 1