Lab11/CW6: Entropy and Equilibrium

Objectives

In Chapter 12 of *Matter & Interactions 4e* you have calculated the number of ways to arrange a given amount of energy in a very small nanoparticle modeled as an Einstein solid. In order to deal with a larger number of atoms, we need to use a computer. In this activity we will explore the most probable distributions of 0 to 100 quanta of energy between two solid objects, made of the same material, containing a total of 700 quantum oscillators.

1 The VPython `combin()` function

The `combin()` function in VPython takes two arguments:

```
c = combin(a,b)
```

The definition of this function is this: `combin(a,b)` $= \dfrac{a!}{b!(a-b)!}$

In order to use the `combin()` function you must insert the following line near the beginning of your program:

```
from visual.factorial import *
```

2 Number of Ways of Arranging Energy

Completing the program below (by replacing all question marks with appropriate instructions) should produce a graph identical to Figure 12.16. Make sure your program works correctly before going on to change model parameters.

```
from __future__ import division, print_function
from visual.graph import *
from visual.factorial import *

waygraph = gvbars(delta=0.7, color=color.red) # to make vertical bar graph
Ntotal = 6               ## total number of oscillators
N1 = 3                   ## number of oscillators in object 1
N2 = Ntotal-N1           ## number of oscillators in object 2
qtotal = 4               ## total quanta of energy shared among all the oscillators

q1 = 0                   ## start with no quanta of energy in object 1

while q1 <= qtotal:      ## for each possible value of energy in object 1
    q2 = ?               ## number of quanta of energy in object 2
    ways1 = ?            ## Calculate number of ways to arrange q1 quanta in object 1
    ways2 = ?            ## Calculate number of ways to arrange q2 quanta in object 2:
    waygraph.plot( pos=(q1,??)  )      ## Plot number of ways to arrange energy in both objects
    q1 = q1+1
```

⇒ Complete the program above, then run it. Check to make sure it produces a histogram identical to Figure 12.16.

⇒ Model a system consisting of two solid blocks composed of the same substance. Block 1 contains 400 oscillators and block 2 contains 300 oscillators. Find the possible distribution of 100 quanta of energy between these two blocks, and plot the number of microstates versus number of quanta assigned to block 1.

⇒ Numerically find the most probable energy distribution, and the distribution that is half as probable as the most probable distribution.

⇒ Keep the total number of energy quanta constant at 100. Keep the total number of oscillators fixed at 700, but vary the number of oscillators in each block. For example, consider equal numbers of oscillators, and ratios of 2:1, 5:1, and so on. Describe your observations.

Check your work before continuing.

3 Entropy and Temperatue

⇒ Write a function `microstates(q,N)` that returns the number of microstates Ω for a system with q energy quanta and N oscillators.

⇒ Write a function `entropy(q,N)` that returns the entropy S of a system with q energy quanta and N oscillators.

⇒ For two blocks of 300 oscillators and 400 oscillators, plot $\ln \Omega_1$, $\ln \Omega_2$, and $\ln(\Omega_1 \Omega_2)$ as a function of q_1, for q_1 running from 0 to 100 quanta. In Python the function `log()` gives the natural log (ln).

⇒ Numerically find the maximum value of $\ln(\Omega_1 \Omega_2)$ and the q_1 at which this maximum occurs.

⇒ Write a function `temperature(q,N,w)` that the temperature T of a system with q energy quanta and N oscillators of energy quantum $\Delta E = \hbar w$. To find the temperature, you need to determine the values of ΔE and ΔS that corresponds to the one-quantum change in energy.

⇒ The interatomic constant $k_{s,i}$ of Al is 16 N/m. Find the temperature in Kelvins of an Al block with 35 atoms (105 oscillators) and 150 quanta.

2017-03-14