DOING PHYSICS WITH PYTHON

QUANTUM STATISTICS

PROBABILITY DISTRIBUTIONS

MAXWELL-BOLTZMANN DISTRIBUTION

Ian Cooper

matlabvisualphysics@gmail.com

DOWNLOAD DIRECTORY FOR PYTHON SCRIPTS

qmSM01.py

The Maxwell-Boltzmann distribution for 6 particles with the constraint that the total energy of the system is 8E

qmSM02.py

The Maxwell-Boltzmann distribution at different temperatures

qmSM03.py

Relative populations of the excited states of hydrogen atoms at different temperatures

GitHub

Google Drive

INTRODUCTION

Quantum statistical mechanics is the study of the occupation of quantum levels using probability theory. It plays an essential role in quantum semiconductor theory where two important quantities are the **density of states** and the **probability distribution** which are essential in quantum transport. These concepts lead to how semiconductor devices conduct an electric current (flow of charge in the form of electrons and holes).

There are a number of probability distributions that will be discussed on my web articles for statistical mechanics:

- **Maxwell-Boltzmann distribution** concerns the distribution of a given amount of energy between identical but distinguishable particles. The **Maxwell velocity distribution** is a special case is the Maxwell-Boltzmann distribution and forms the basis of the kinetic theory of gases and defines the distribution of speeds for a gas at a certain temperature. From this distribution function, the most probable speed, the average speed, and the root-mean-square speed can be derived. The particles are disguisable, and there is no limit on the number of particles in a given energy state. Classical distribution
- **Boltzmann distribution** gives the probability that a system will be in a certain state as a function of that state's energy and the temperature of the system and applies to designable particles. Classical distribution

- **Fermi-Dirac distribution** describes the possible ways in which a system of indistinguishable (fermions: odd or half integral spin) particles can be distributed among a set of energy states occupied by only one particle. These particles are indistinguishable obeying the Pauli exclusion principle, and there can be no more than one particle per quantum state. Quantum distribution
- **Bose-Einstein distribution** describes the statistical behavior of integer spin particles (bosons) on the ways in which the particles may occupy a set of available discrete energy states. These particles are indistinguishable and do not obey the Pauli exclusion principle, and there is no limit on the number of particles per quantum state. Quantum distribution

But first, we will consider **Maxwell–Boltzmann distribution** (MBD) for a system of particles. The basic assumptions for the M-B are:

- The particles are identical in terms of physical properties but distinguishable in terms of position, path, or trajectory (the particle size is small compared with the average distance between particles).
- The equilibrium distribution is the most probable way of distributing the particles among various allowed energy states subject to the constraints of a fixed number of particles and fixed total energy.

To illustrate the main concepts of the MBD, we will consider a system of six particles and that the total energy of the six particles is 8*E* where *E* is an indivisible unit of energy. So, any one particle can have an energy:

$$0E, 1E, 2E, 3E, 4E, 5E, 6E, 7E, 8E$$

Thus, there are 9 energy levels and there are consequently 20 possible ways of sharing an energy of 8*E* among six indistinguishable particles. Each of these 20 arrangements are called **macrostates**, However, we are really interested in distinguishable particles. So, each of the 20 arrangements can be decomposed into many distinguishable **microstates**.

Macrostate 1: one particle has energy 8E, therefore the other 5 particles have zero energy. But any one of the 6 particles could have energy 8E, therefore there are 6 microstates for macrostate 1.

Macrostate 2: one particle has energy 7*E*, so another particle must have energy 1*E*. Therefore, the number of microstates is (6)(5) = 30 since the any of the 6 particles can have energy 7*E* and any of the 5 remaining particles can have energy 1*E*.

We can continue this process, and you will find that there are 20 macrostates and 1287 microstates. The number of microstates for the each of the 20 macrostates is given by microstates $N_{MB} = \dfrac{N!}{n_0!\ n_2!\ldots\ n_8!}$ $N = 6 \quad 0! = 1$

where N is the total number of particles and n_i is the number of particles with energy E_i.

Macrostate 1: $n_0 = 5$ and $n_8 = 1$ $N_{MB} = 6!/[(5!)(1!)] = 6$

Macrostate 2: $n_0 = 4$ and $n_7 = 1$ $N_{MB} = 6!/[(4!)(1!)] = (6)(5) = 30$

The Python Code **qmSM01.py** is used for simulations of the system of 6 particles with fixed total energy $8E$.

```
N = 6   # Total number of particles
nE = 9  # Number of energy levels: energy levels 0 1 2 3 4 5 6 7 8
nMacro = 20  # Number of macroStates

# Populations of macrostates: number of particles for each energy
level
macroS = zeros([nMacro,nE])
macroS[0,:]  = np.array([5,0,0,0,0,0,0,0,1])
macroS[1,:]  = np.array([4,1,0,0,0,0,0,1,0])
macroS[2,:]  = np.array([4,0,1,0,0,0,1,0,0])
macroS[3,:]  = np.array([3,2,0,0,0,0,1,0,0])
macroS[4,:]  = np.array([4,0,0,1,0,1,0,0,0])
macroS[5,:]  = np.array([3,1,1,0,0,1,0,0,0])
macroS[6,:]  = np.array([2,3,0,0,0,1,0,0,0])
macroS[7,:]  = np.array([4,0,0,0,0,2,0,0,0])
macroS[8,:]  = np.array([3,1,0,1,1,0,0,0,0])
macroS[9,:]  = np.array([3,0,2,0,1,0,0,0,0])
macroS[10,:] = np.array([2,2,1,0,1,0,0,0,0])
macroS[11,:] = np.array([1,4,1,0,1,0,0,0,0])
macroS[12,:] = np.array([3,0,1,2,0,0,0,0,0])
```

```
macroS[13,:] = np.array([2,2,0,2,0,0,0,0,0])
macroS[14,:] = np.array([2,1,2,1,0,0,0,0,0])
macroS[15,:] = np.array([1,3,1,1,0,0,0,0,0])
macroS[16,:] = np.array([0,5,0,1,0,0,0,0,0])
macroS[17,:] = np.array([2,0,4,0,0,0,0,0,0])
macroS[18,:] = np.array([1,2,3,0,0,0,0,0,0])
macroS[19,:] = np.array([0,4,2,0,0,0,0,0,0])
```

We can compute the number of microstates for each of the 20 macrostates:

```
# Number of microstates for each macrostate
microS = zeros(nMacro)
Nf = factorial(N)
nf = zeros(9)
for c1 in range(nMacro):
   q = 1
   for c2 in range(nE):
      q = q*factorial(macroS[c1,c2])
   microS[c1] = Nf / q
# total number of microstates
microN = sum(microS)
```

Console output

```
Number of microstates for each energy level (macrostate)
[  6.  30.  30.  60.  30. 120.  60.  15. 120.  60. 180.  30.  60.  90.
 180. 120.   6.  15.  60.  15.]
Total number of microstates
1287.0
```

We can find the average number of particles n_{i_avg} with an energy E_i

probability of observing a given macrostate =
number of microstates in macrostate / total number of microstates

```
# Probability of observing of a given macrostate
p = zeros(nMacro)
nAvg = zeros(nE)
for c in range(nMacro):
   p[c] = microS[c]/microN
```

Console output

```
array([0.004662  , 0.02331002, 0.02331002, 0.04662005,
0.02331002, 0.09324009, 0.04662005, 0.01165501, 0.09324009,
0.04662005, 0.13986014, 0.02331002, 0.04662005, 0.06993007,
0.13986014, 0.09324009, 0.004662  , 0.01165501, 0.04662005,
0.01165501])
```

```
# Average number of particles in a given macrostate
for c in range(nE):
   nAvg[c] = sum(macroS[:,c]*p)

# Probability of finding a particle with a given energy
probE = nAvg/6
```

Console output

```
Probability of finding a particle with a given energy
[0.38461538  0.25641026  0.16705517  0.0979021  0.05050505
0.03108003  0.01165501  0.003885   0.000777  ]
```

The graph of the probability of finding a particle with a given energy against the energy levels gives the distribution function of 6 distinguishable particles with a total energy 8*E* is shown in figure 1. There is a rapidly decrease in probability with increasing energy and the decrease in probability can be modelled as an exponential function. More particles are found in a lower energy level than the next higher energy level.

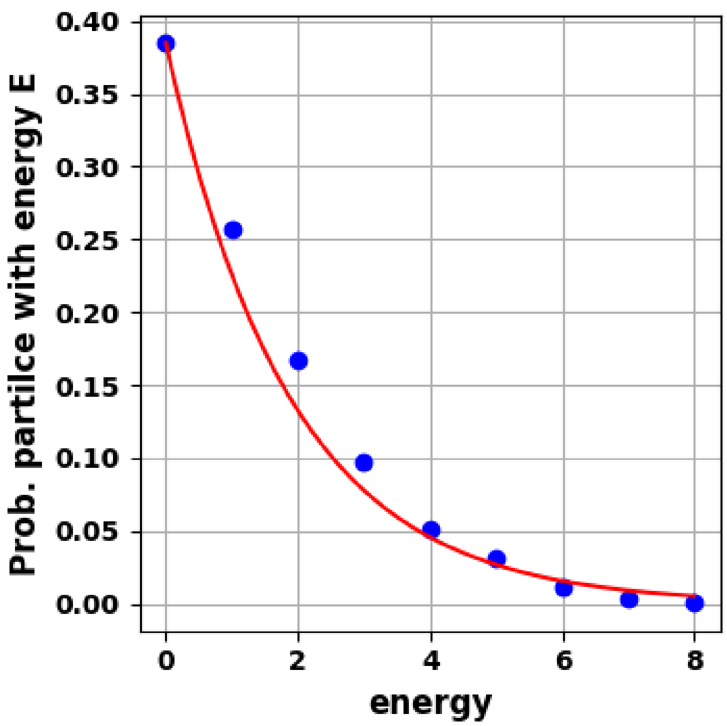


Fig. 1. Distribution function of the 6 particles with a total energy of 8*E*. **qmSM01.py**

MAXWELL – BOLTZMANN DISTRIBUTION

We can now consider a system containing a very large number of particles and how the energy of the particles are distributed. For the **Maxwell–Boltzmann distribution** the constraints are:

- The total number of particles is constant at any temperature.
- The total energy of the system is fixed at a given temperature.
- There maybe different states which have the same energy and this is referred to as degeneracy g ($g(E)$) (statistical weight) and is the density of states.

The exponential form of the Maxwell–Boltzmann distribution is

$$f_{MB} = A\exp(-E / k_B T)$$

where k_B is the Boltzmann constant, T is the absolute temperature and A is a normalization constant.

The number of particles $n_E\, dE$ is the number of particles per unit volume with energies between E and dE. For N particles in a volume V, the number density is $n = N / V$.

$$n_E\, dE = g\; f_{MB}\, dE$$

$$n = \frac{N}{V} = \int_0^\infty g\; f_{MB}\, dE$$

SIMULATION

qmSM02.py

We can model the Maxwell-Boltzmann distribution for different temperatures T = 300, 500, 700, 900 K. The system considered has a fixed number of particles $N = 10^8$ in a unit volume V = 1 and the sum of the energy of all the particles is a constant E_{max} = 0.40 eV. The energy of each particle can have a value ranging continuously from E_{min} = 0 to E_{max} = 0.4 eV. The degeneracy for the system is g = 1.

```
# Total energy of system of particles [J]
        E = e*linspace(Emin,Emax,num)
```

The Maxwell-Boltzmann distribution

$$f_{MB} = A\exp(-E / k_B T)$$

is calculated with the Python function

```
# Function: Maxwell-Boltzmann distribution: fixed number of
particles N
def fn(T):
    k = -1/(kB*T)
    f = exp(k*E)
    I = simps(f,E)
    A = N/I
```

```
f = A*exp(k*E)
return f
```

where A = 1 is the start value for the normalization constant. The value of A is adjusted for each temperature T so that the number of particles is $N = 10^8$ by using the relationship

$$n = \frac{N}{V} = \int_0^{\infty} g\ f_{MB}\, dE$$

The results are displayed as shown in figure 2 for n against E.

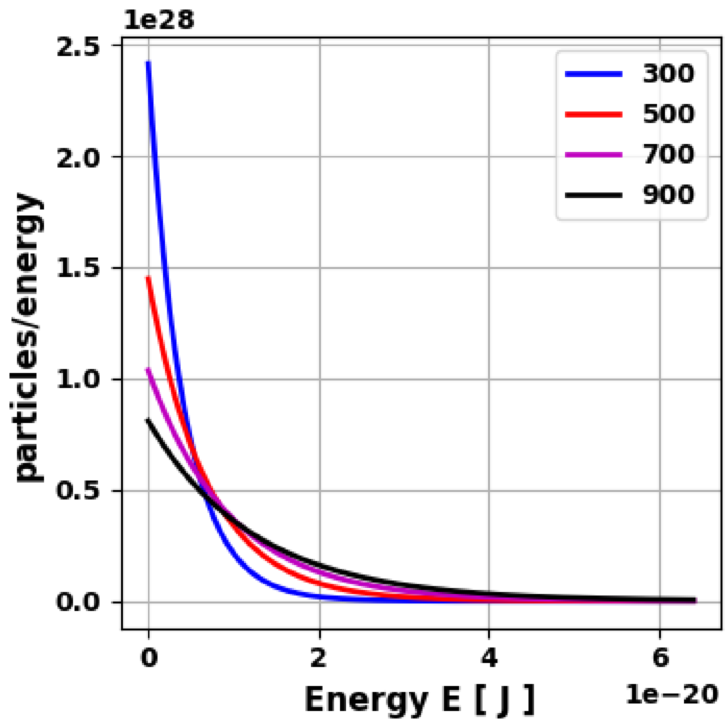
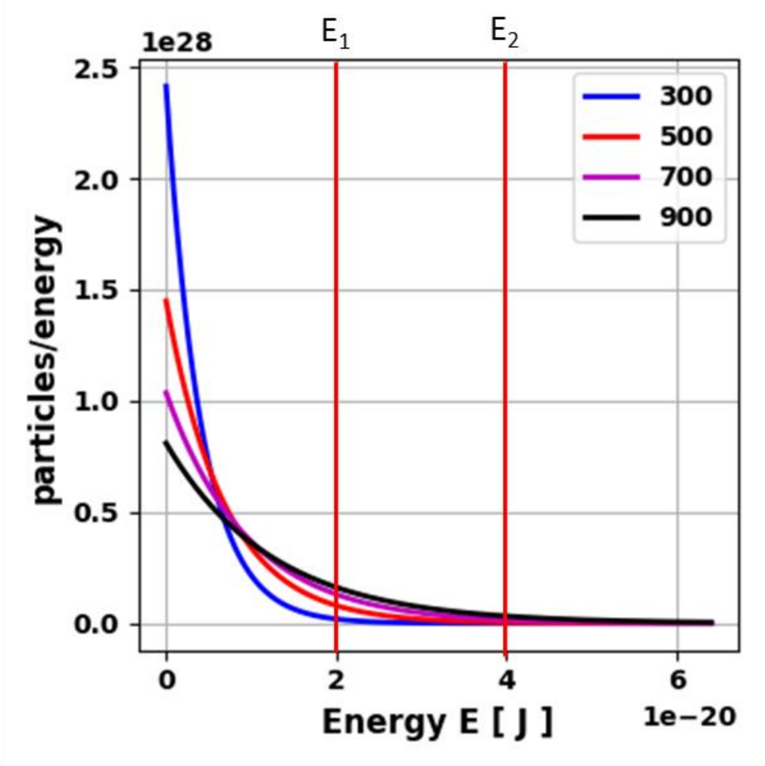


Fig. 1A. The Maxwell-Boltzmann distribution for the system of 10^8 particles with total energy 0.4 eV at different temperatures.

qmSM02.py



Temperature T [K]	300	500	700	900
Percentage particles 0 to E1	99.2	94.4	87.4	80.3
Percentage particles E1 to E2	0.8	5.2	11.1	16.1
Percentage particles E2 to E_{max}	0.0	0.3	1.5	3.4

Fig. 2B. The Maxwell-Boltzmann distribution for the system of 10^8 particles with total energy 0.4 eV at different temperatures.

qmSM02.py

Figure 2B shows the percentage number of particles in the three energy bands for the four temperatures. At the lowest temperature nearly all the particles are restricted to the lowest energy band. As the temperature increases, more and more individual particles have higher energy.

Example Emission lines of hydrogen atoms

Find the populations of the first six states of atomic hydrogen relative to the ground state population of 100 for different temperatures.

Solution Python Code qmSM03.py

The first six energy levels for the hydrogen atom are given by

$$E_m = -13.6 / m^2 \quad m = 1,\ 2,\ 3,\ 4,\ 5,\ 6$$

where m is the principal quantum number.

The ground state energy is E_1 = -13.6 eV. To use the Maxwell-Boltzmann distribution, the ground state is set to zeros and the excited states are measured w.r.t. the ground state energy E_{R1} = 0.

$$E_{Rm} = E_m - E_1$$

The Maxwell-Boltzmann distribution yields the number of particles per unit volume

$$n = g\,A \exp\left(-\frac{E}{k_B\,T}\right)$$

where g is the density of states or the number of energy states per unit volume. For the hydrogen atom, the density of states g depends upon the principal m where

$$g = 2\,m^2$$ (n is the particle number density)

The Python Code (**qmSM03.py**) to calculate the relative populations;

```
def fn(c,ER,T):
   g = 2*c**2
   k = -e/(kB*T)
   f = g*exp(k*ER)
   return f

E = zeros(6); ER = zeros(6); nR = zeros(6)
for c in range(6):
   E[c] = -13.6/(c+1)**2
   ER[c] = E[c] - E[0]
   nR[c] = fn(c+1,ER[c],T)
nR = 100*nR/nR[0]
```

The computations are displayed in the Console Window.

The energy level of the hydrogen atom:

```
Energy levels  [eV]
State   En     ER
  1   -13.60   0.00
  2   -3.40   10.20
  3   -1.51   12.09
  4   -0.85   12.75
  5   -0.54   13.06
  6   -0.38   13.22
```

The relative populations at different temperatures:

Temperature T = 300 K

Relative populations: nR = 100 for state 1

State	nR
1	100.0000
2	0.0000
3	0.0000
4	0.0000
5	0.0000
6	0.0000

At room temperature T =300 K, all the hydrogen atoms are in the ground state.

Temperature T = 20000 K

Relative populations: nR = 100 for state 1

State	nR
1	100.0000
2	1.0782
3	0.8111
4	0.9827
5	1.2858
6	1.6814

Temperature T = 50000 K

Relative populations: nR = 100 for state 1

State	nR
1	100.0000
2	37.5249
3	54.4729
4	83.0701
5	120.9018
6	167.5123

At the high temperature of 20 000 K (flame, electric arc)), some of the atoms will be in an excited state. However, at the temperature of 50 000 K the highest energy levels are even more populated than the ground state.

The strength of an emission or absorption line is proportional to the number of atomic transitions per unit time. For particles obeying Maxwell-Boltzmann statistics, the number of transitions per unit time from some initial state *i* to some final state *f* equals the product of the population of the initial state and the probability for the transition from the initial state to the final state $(i \to f)$. The transition rate for particles obeying Maxwell-Boltzmann statistics depends only on the initial population, since there are no restrictions on the number of particles in the final state.

Emission strength $(i \to f)$

= population(*i*) x probability of transition $(i \to f)$

Hence, we can conclude that the emission line will be stronger the more highly populated is the excited state.