CO20: The Boltzmann Distribution

F. I. Giasemis
Keble College, University of Oxford

November 2016

Abstract

The aim of this project is to get an insight into the statistical nature of the *Boltzmann distribution*. Firstly, we used a specific process to illustrate that the distribution of energies is such that the number of microstates Ω is maximized. After that, we visualized how the process modifies the distribution of energies after each iteration and finally we created a GIF animation of it.

1 Introduction

1.1 The Boltzmann distribution

In statistical mechanics, a Boltzmann distribution is a probability distribution of particles in a system over various possible states. The distribution is expressed in the form

$$\boxed{F(state) \propto e^{-\frac{E}{kT}}} \tag{1}$$

where E is the state energy, k is Boltzmann's constant and T is the thermodynamic temperature. It 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.[3] It is given by :

$$P_i = \frac{e^{-\frac{\epsilon_i}{kT}}}{\sum\limits_{j=1}^{N} e^{-\frac{\epsilon_j}{kT}}} \tag{2}$$

where P_i is the probability of state i, ϵ_i the energy of state i, k the Boltzmann constant, T the temperature of the system and N is the number of states accessible to the system.[3]

Because of this, the Boltzmann distribution can be used to solve a very wide variety of problems. The distribution shows that states with lower energy will always have a higher probability of being occupied than the states with higher energy.[2]

1.2 Useful definitions

- Macrostate: The macrostate of a system refers to its macroscopic properties, such as its temperature, pressure, volume and density.
- Microstate: A microstate is a specific microscopic configuration of a thermodynamic system that the system may occupy with a certain probability in the course of its thermal fluctuations.[4]
- Often, the number of different microstates a system can be in, is denoted by Ω.

2 A first approach

To illustrate the statistical nature of the Boltzmann distribution, we will play a game in which quanta of energy are distributed in a lattice. We choose a lattice of 400 sites, arranged for convenience on a 20×20 grid.[1] The initial arrangement is shown in Fig. 1.

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 1: In the initial state, one quantum is placed on each site.[1]

We now choose a site at random and remove the quantum from that site and place it on a second, randomly chosen site. This redistribution process is repeated many times. The Matlab code for this process is the following:

```
%Boltzmann dist
%Fotios Ioannis Giasemis

N=input('N='); %dimension of square matrix; here we set N=20
M=N; %square matrix
Z=input('Iterations='); %number of iterations of the process

A=ones(N,M); %matrix: initial configuration

for i=1:Z;

    X=randi(N); Y=randi(M); %pick two random integers -> random cell

    while A(X,Y)<=0 %make sure the energy E is not zero or less
    X=randi(N); Y=randi(M);
    end

    A(X,Y)=A(X,Y)-1; %extract one quantum

    X=randi(N); Y=randi(M); %pick a second random cell

    A(X,Y)=A(X,Y)+1; %add one quantum

end

histogram(A)

```

```
clc; clear;
```

The result is shown in Fig. 2. The histogram describing this looks very much like a Boltzmann distribution.

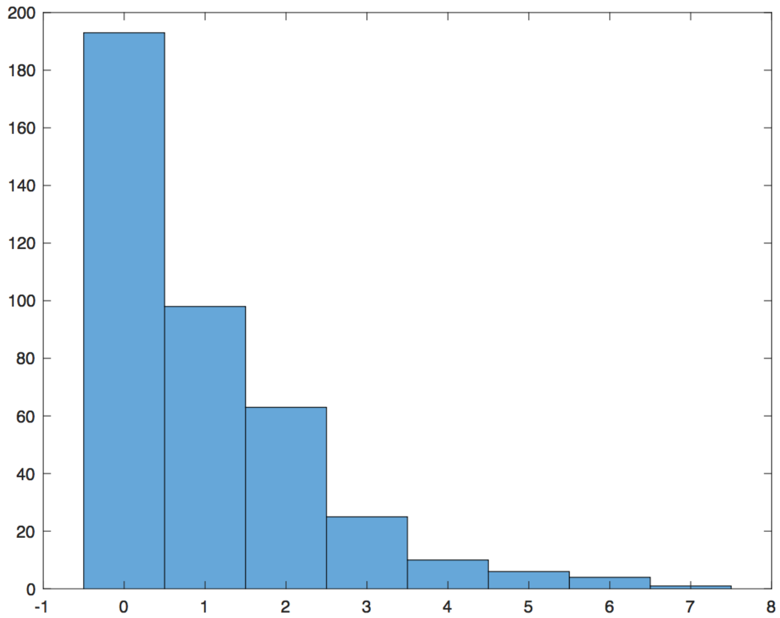

Figure 2: $histogram(A)$ for N=20 and 1000 iterations (i.e. $Z = 1000$)

The initial distribution shown in Fig. 1 gives an equal amount of energy to every site. It is however very statistically unlikely because it is associated with only a single microstate, i.e. $\Omega = 1$. There are many more microstates associated with other macrostates. For example, the state obtained after a single iteration, is much more likely, since there are 400 ways to choose the site from which a quantum has been removed, and then 399 ways to choose the site to which a quantum is added; hence $\Omega = 400 \times 399 = 19600$, and that is for only one iteration. For 1000 iterations the value of Ω becomes enormous. The Boltzmann distribution is simply a matter of probability.

3 Visualizing the process

In this section, we will modify our Matlab code in order to be able to visualize the process. We are interested to see how the $histogram(A)$ evolves over the number of iterations.

```
%Boltzmann dist
%Visualization
%Fotios Ioannis Giasemis

N=input('N=');
M=N;
Z=input('Iterations=');

A=ones(N,M);

for i=1:Z;

    X=randi(N); Y=randi(M);

    while A(X,Y)<=0
```

```
    X=randi(N); Y=randi(M);
    end

    A(X,Y)=A(X,Y)-1;

    X=randi(N); Y=randi(M);
    A(X,Y)=A(X,Y)+1;

    histogram(A)
    xlim([-.5 10]) %fix the limits of the axes
    ylim([0 N*M]) %we know that y cannot be larger than NM
    pause(.0001) %stop the process for a small interval of time
end

histogram(A)
xlim([-.5 10])
ylim([0 N*M])
clc; clear
```

4 Creating an animated GIF

Finally, we will we modify our code again: we want our program to save the figure of *histogram*(A) in the "current folder" for each iteration of the process.

```
%Boltzmann dist
%Visualization
%GIF
%Fotios Ioannis Giasemis

N=input('N=');
M=N;
Z=input('Iterations)=');

A=ones(N,M);

for i=1:Z;

    X=randi(N); Y=randi(M);

    while A(X,Y)<=0
    X=randi(N); Y=randi(M);
    end

    A(X,Y)=A(X,Y)-1;

    X=randi(N); Y=randi(M);
    A(X,Y)=A(X,Y)+1;

    fig=figure %name the figure
    histogram(A)
    xlim([-.5 10])
    ylim([0 N*M])

    %in order to save space, repeat every 10 iterations
    if mod(i,10)==0 | i==1
```

```
        saveas(fig,sprintf('FIG%d.png',i)) %save "fig"
    end

    close
end

histogram(A)
xlim([-.5 10])
ylim([0 N*M])
clc; clear
```

After running our program, PicGIF Lite[5] was used to create the animated GIF with FPS=20 and size 640 × 400 pixel.

The result can be found here: `https://1drv.ms/i/s!AlgNQmhpzzNPh6dfqURZChQL01aGFA`.

5 Conclusion

In the case that we have studied, we set $N = 20$ (i.e. 20×20 square matrix), which is a pretty small number compared to thermodynamical scales. Nevertheless, the result we got in Fig. 2, was obviously resembling a Boltzmann distribution. The result is even better for $N = 100$ and $Z = 10^6$ (iterations).

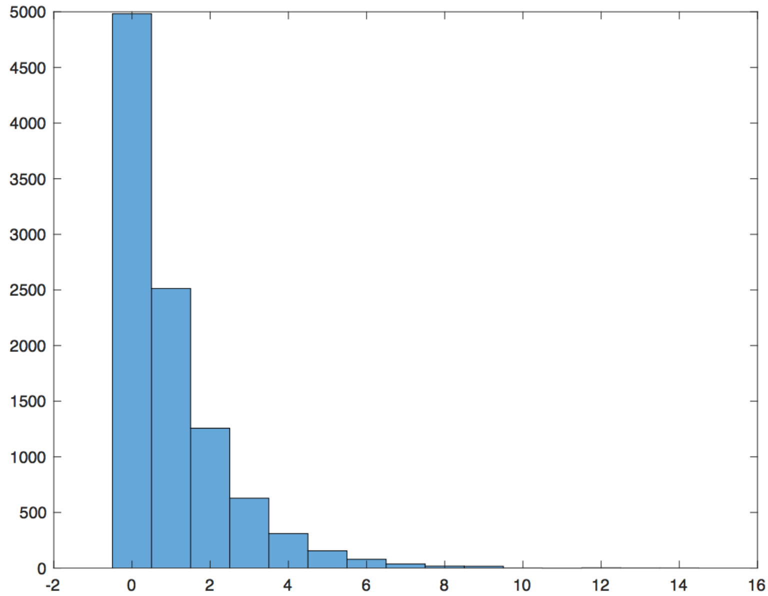

Figure 3: $histogram(A)$ for $N = 20$ and $Z = 10^6$

In thermodynamical scales, the Boltzmann distribution proves to be a major rule describing how the *world* works.

6 References

[1] Stephen J. Blundell and Katherine M. Blundell, Concepts in Thermal Physics, 2nd edition, Oxford University Press.

[2] Atkins, P. W. (2010) Quanta, W. H. Freeman and Company, New York.

[3] `https://en.wikipedia.org/wiki/Boltzmann_distribution`

[4] `https://en.wikipedia.org/wiki/Microstate_(statistical_mechanics)`

[5] `https://itunes.apple.com/gb/app/picgif-lite/id844918735?mt=12`