Orientational ordering of long rods on a lattice

D. Dhar[1] and A. Ghosh[2]

[1]Department of Theoretical Physics, Tata Institute of Fundamental Research - Homi Bhabha Road, Mumbai 400 005, India
[2]Department of Theoretical Physics, Tata Institute of Fundamental Research - Homi Bhabha Road, Mumbai 400 005, India

April 2023

Report Submitted by:
Aayushman(20004) Department of Data Science & Engineering,
S John Sharon Sandeep(20021) Department of Physics

1 Introduction

:

The study of the orientational ordering of long rods on a lattice is a fascinating subject in statistical physics. Long rods are idealized models of linear molecules like polymers or biological filaments, and their ordering can have significant consequences for the properties of materials in various contexts, including liquid crystals, gels, and biomaterials. The paper "On the orientational ordering of long rods on a lattice" by A. Ghosh and D. Dhar, published in the Journal of Physics A: Mathematical and Theoretical, explores the orientational ordering of long rods on a lattice in two dimensions.

2 Theory

The system we are considering can be modeled as a two-dimensional square "space" of size $L \times L$ sq. units which can hold a variable number of identical rods of length k units(initially let's ignore the lattice aspects and the discrete orientations of the rod). A general potential energy for the system when it has say N rods would be of the form:

$$V(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}, \theta_\mathbf{1}, \theta_\mathbf{2}, \ldots, \theta_\mathbf{N}) = \sum_{i=1}^{N} f_1(\mathbf{x_i}, \theta_\mathbf{i}) + \frac{1}{2} \sum_{i,j=1}^{N} f_2(|\mathbf{x_i} - \mathbf{x_i}|, |\theta_\mathbf{i} - \theta_\mathbf{j}|) \tag{1}$$

Where f_1 and f_2 are some functions, while $\mathbf{x_i}$ & $\theta_\mathbf{i}$ are the position (of the center) and angular coordinates of the i^{th} rod.

We ignore motion in the rods and consider energy to be solely due to V.

Let's consider this system to be immersed in a large reservoir with which it can exchange both energy and particles. For a given time t, define N & E to be the number of rods in the system and the energy of the system respectively. Let's assume that for any $l \in \mathbb{W}$, the states corresponding to $N = l$ are countable (i.e. the whole set of states can be labeled). Then define E_{l_m} to be the energy eigenvalue for the m^{th} state for $N = l$.
Note: $\{E_{l_m}\}_l$ accounts for degeneracy.
The probability to find the system to be in the state $N = n, E = E_{n_s}$ for any valid $n, s \in \mathbb{W}$, would be:

$$P_{n,s} = \frac{exp(-\alpha n - \beta E_{n_s})}{\sum_{l,m} exp(-\alpha l - \beta E_{l_m})} \tag{2}$$

where $\beta = 1/(k_B T)$ and $\alpha = -\mu\beta$ for some T and μ called the temperature and chemical potential of the system respectively. This is a standard result from statistical mechanics for systems in a grand canonical ensemble.

(2) can be rewritten as:

$$P_{n,s} = \frac{z^n exp(\beta E_{n_s})}{\sum_{l,m} z^l exp(\beta E_{l_m})} \tag{3}$$

Where $z = exp(-\alpha)$, called the fugacity of the system.

Now let's define a set of *lattice points* $\{p_i\}_i$ by dividing the said square region into $L \times L$ squares and labeling the endpoints with p_i.

We now fix V such that: (4)

- f_1 in (1) diverges to ∞ when $\mathbf{x_i} \neq p_j \ \forall \ p_j$ for any x_i and when $\theta_\mathbf{i} \notin \{0, \pi/2\}$ for any x_i.
  This ensures that only positions in the *lattice points* are allowed, and the only possible orientations are horizontal and vertical for each rod.
- f_2 in (1) diverges to ∞ when any two rods overlap at one or more points.
  Such an interaction is termed as *hardcore interaction*.
- f_1 & f_2 are 0 for any other point in the position space.
  i.e. V is zero for any valid configuration.

Applying (4) to (3), the probability to find the system in the α^{th} distinct configuration of $N = n$ is:

$$P_{n,\alpha} = \frac{z^n}{\sum\limits^{''}_{n_v,n_h} z^{(n_v+n_h)} \mathcal{N}(n_v, n_h)} \tag{5}$$

Where $''$ over $\sum$ indicates summation over those Whole numbers n_v, n_h such that $n_v + n_h \leq L^2/k$ (the maximum number of rods possible), and $\mathcal{N}(n_v, n_h)$ is the (finite) number of configurations that have n_v vertical rods and n_h horizontal rods. $\mathcal{N}(n_v, n_h)$ is present inside the summation term as we accounted for degeneracy in (2).

Now for a physical quantity $A(n_h, n_v)$, define the expectation value of that quantity as:

$$\langle A \rangle = \sum_{n}^{''} \sum_{\alpha} P_{n,\alpha} A(n_h(n,\alpha), n_v(n,\alpha)) \tag{6}$$

where the first summation is over all possible values of N viz. $n \in [0, L^2/k], n \in \mathbb{W}$. While the second summation is over all possible labeled configurations α for $N = n$.
$n_h(n, \alpha)$ returns the number of horizontal rods for the α^{th} configuration of $N = n$, and similarly we define n_v.

In this work, we are particularly interested in the quantity:

$$Q = \frac{\langle n_v - n_h \rangle}{\langle n_v + n_h \rangle} \tag{7}$$

Called the nematic order parameter of the system, and is a function of the fugacity z defined under (3).

3 Methodology

We want to calculate the quantity Q in (7) using (5) & (6) and plot it as a function of the fugacity z.

To do this we use the montecarlo technique called Markov Chain as follows.

Firstly it's possible to label/identify all the possible states with natural numbers from 1 to M, for some finite M (it's easy to show that the numbers of possible states are finite).

Suppose we have an operation O that acts on a state A and probabilistically gives a new state B with a fixed probability independent of time. Define an $M \times M$ matrix $\mathrm{T}(B, A) \equiv \mathrm{T}(A \to B)$ that gives the probability for $O(A)$ to be equal to B.

By taking any initial state A, we can associate a vector ρ_i to the state $O^i(A)$ as follows:

$$\rho_i = \mathrm{T}^i \rho_0, \quad \text{where } \rho_0 \text{ is s.t. } \rho_0(A) = 1 \ \& \ \rho_0(B) = 0 \ \forall \ B \neq A \tag{8}$$

Here, $\rho_i(B)$ gives the probability that $O^i(A)$ equals B, i.e. the probability that we would get the state B after applying O exactly i times on the state A.

If we find such an operator O (i.e. an algorithm for the same) such that for large N, if O^N (acted on any state) mimics the probability distribution in (5), we could exploit it to find the mean of any quantity as defined in (6).

The following theorem (stated without proof) guarantees that O would do this if it satisfies three conditions stated in the theorem:

Theorem. *A system with a finite number of states can be guaranteed to converge to an equilibrium distribution P if the computer algorithm/ operation O:*

1. *is Markovian/ has no memory*
   That is, the operation's output is only dependent on the previous state.

2. *is ergodic*
   That is, it can reach everywhere and is acyclic.

3. *satisfies detailed balance w.r.t. P*
   That is, $\mathrm{T}(A, B)P_B = \mathrm{T}(B, A)P_A \ \forall$ states A, B (where $\mathrm{T}(A, B)$ is as previously defined).

The authors have used the following algorithm for creating one such O:

Algorithm 1 Deposition-Evaporation Algorithm

```
Initialize a lattice of size N × N with all sites empty
p ← probability of deposition
k ← length of rod
n ← number of steps to simulate
for i ← 1 to n do
    if random number ≤ p then
        randomly select horizontal or vertical orientation
        select a random site on the lattice
        if the chosen direction is unoccupied for the next k − 1 sites then
            occupy the k sites in the chosen direction
        end if
    else
        randomly select a site on the lattice
        Remove the rod occupying that site if any
    end if
end for
```

Here we roughly sketch a proof that the above algorithm satisfies the stated theorem.

Proof. 1. It is Markovian as O takes only one input state (the previous state)

2. - It is irreducible
   As it is possible to create a sequence of moves that takes us from any state A to any state B. A rigorous proof of this would be out of the scope for this report.
   - Has aperiodicity
   We found the proof of this difficult to comprehend in the stipulated time, therefore this shall be skipped too.

3. It satisfies detailed balance w.r.t. $P_{n,\alpha}$ in (5), i.e. $\mathrm{T}(A,B)P_B = \mathrm{T}(B,A)P_A \ \forall$ states A, B
   For a particular value of z, we would take $p = 2kz/(1+2kz)$. Let A and B be any two states.
   For the case when A and B cannot be related by a single move, $\mathrm{T}(A,B) = \mathrm{T}(A,B) = 0$, and hence the equation of detailed balance trivially holds.
   For the case when A and B can be related by a single move, without loss of generality take A and B of identical configuration except that B has one extra rod of orientation 'd' at position p.
   Then $\mathrm{T}(B,A) = P(\text{choose deposition}|\text{choose p}|\text{choose d}) = p\frac{1}{L^2}\frac{1}{2}$
   And $\mathrm{T}(A,B) = P(\text{choose evaporation}|\text{choose one of p,p+1,..,p+k-1}) = (1-p)\frac{k}{L^2}$
   Then $\frac{P_A}{P_B}\mathrm{T}(B,A) = \frac{1}{z}\frac{p}{2L^2} = \frac{1}{z}\frac{2kz}{(1+2kz)2L^2} = (1-p)\frac{k}{L^2}$
   In above $\frac{P_A}{P_B} = 1/z$ as B has one rod more than A and in lieu of (**??**) has one more power of z.
   □

4 Algorithm

:

Algorithm 2 valid_position

```
1: function VALID_POSITION(pos, direction)
2:     for i ← 0 to k − 1 do
3:         if lattice[(pos[0] + direction[0] ∗ i)%L, (pos[1] + direction[1] ∗ i)%L] ≠ 0 then
4:             return False
5:         end if
6:     end for
7:     return True
8: end function
```

Algorithm 3 find_rod

```
1: function FIND_ROD(pos)
2:     tip ← lattice[pos[0], pos[1]]
3:     d ← (1, 0)
4:     if lattice[(pos[0] − 1)%L, pos[1]] == tip or lattice[(pos[0] + 1)%L, pos[1]] == tip then
5:         k ← 1
6:         while lattice[(pos[0] − k)%L, pos[1]] == tip do
7:             k ← k + 1
8:         end while
9:         p ← [(pos[0] − k + 1)%L, pos[1]]
10:    else
11:        d ← (0, 1)
12:        k ← 1
13:        while lattice[pos[0], (pos[1] − k)%L] == tip do
14:            k ← k + 1
15:        end while
16:        p ← [pos[0], (pos[1] − k + 1)%L]
17:    end if
18:    return p, d
19: end function
```

Algorithm 4 evaporation_decompostion

```
function DEPOSITION_EVAPORATION(L, k, p, num_steps)
    lattice ← 2D array of zeros of size L × L
    n_h, n_v ← 0, 0
    results ← empty list
    for step ← 1 to num_steps do
        pos ← random position in the lattice
        if random number < p then
            if random number < 0.5 then
                direction ← (1, 0)                          ▷ Horizontal deposition attempt
            else
                direction ← (0, 1)                          ▷ Vertical deposition attempt
            end if
            if VALID_POSITION(pos, direction) then
                tip ← next available rod number
                for i ← 0 to k − 1 do
                    lattice[(pos[0] + i × direction[0]) mod L][(pos[1] + i × direction[1]) mod L] ← tip
                end for
                if direction == (1, 0) then
                    n_h ← n_h + 1
                else
                    n_v ← n_v + 1
                end if
            end if
        else
            tip ← lattice[pos[0], pos[1]]
            if tip ∉ list of available rod numbers and tip ≠ 0 then
                add tip to the list of available rod numbers
                head, d ← FIND_ROD(pos)
                for i ← 0 to k − 1 do
                    lattice[(head[0] + i × d[0]) mod L][(head[1] + i × d[1]) mod L] ← 0
                end for
                if d == (1, 0) then
                    n_h ← n_h − 1
                else
                    n_v ← n_v − 1
                end if
            end if
        end if
        rho ← k × (n_h + n_v)/(L × L)
        Q ← |n_h − n_v|/(n_h + n_v + 10^−2)
        norm1 ← n_v − n_h
        norm2 ← n_v + n_h
        append [step, norm1, rho, Q, norm2] to results
    end for
    return results
end function
function VALID_POSITION(pos, direction)
    for i ← 0 to k − 1 do
        if lattice[(pos[0] + i × direction[0]) mod L][(pos[1] + i × direction[1]) mod L] ≠ 0 then
            return False
        end if
    end for
    return True
end function
```

5 Analysis and Results

:

We implemented the above algorithm to plot the density ρ as a function of the deposition probability parameter p and then obtain a function of the same, which was possible as density was always approximately a smooth function of p with very few fluctuations. We then plot the Order parameter Q as a function of p and then as a function of density.

Denoting by N the number of iterations, m the number of initial values dropped for calculating ensemble average or mean, and by l the interval between successive values for calculating mean, we present our results as follows.
we took the case of k=10, L=90. On implementing the program for N=5,00,000, m=10000, and l=400, the following is a plot of order parameter Q as a function of p.

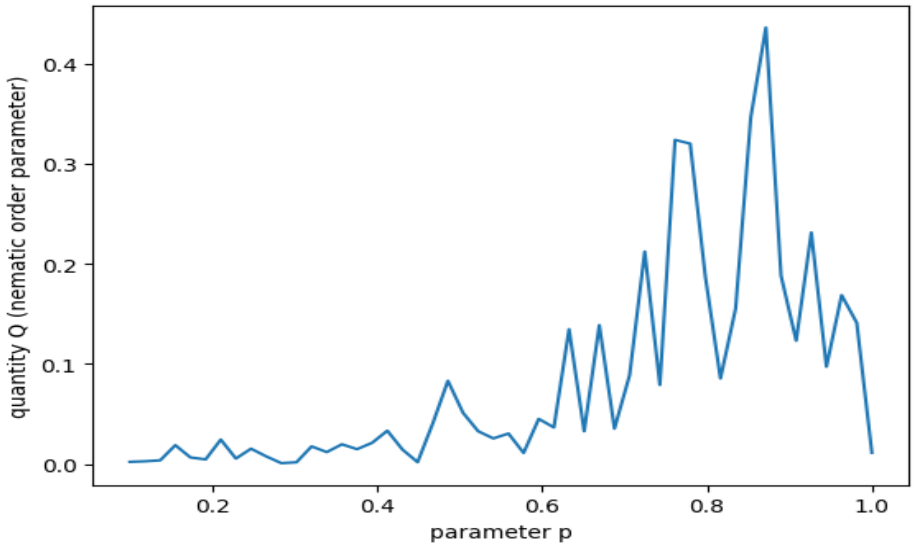

Figure 1: Q vs p for k=10, L=90

The density ρ as a function p is as follows.

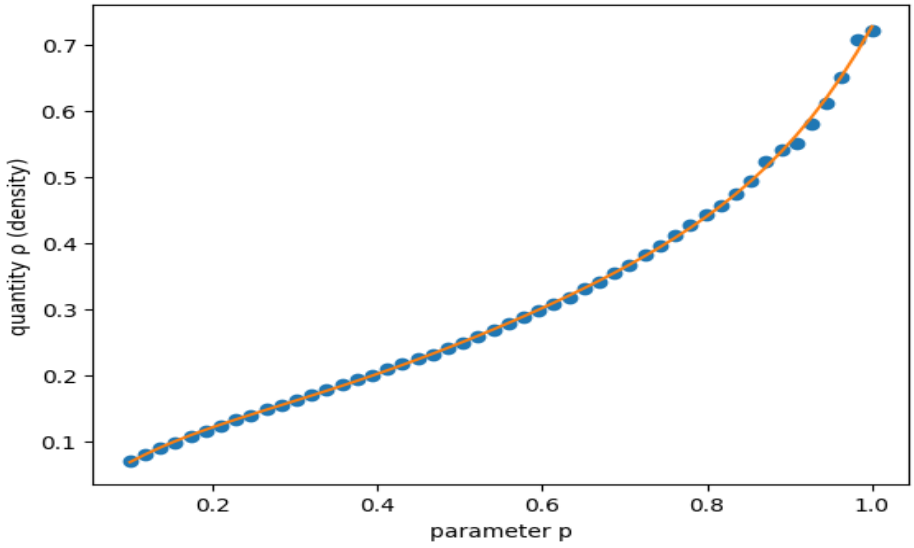

Figure 2: density ρ vs p for k=10, L=90

As already said, density is a nice function of p, which can be stored. Having obtained this, we plot Q as a function of ρ

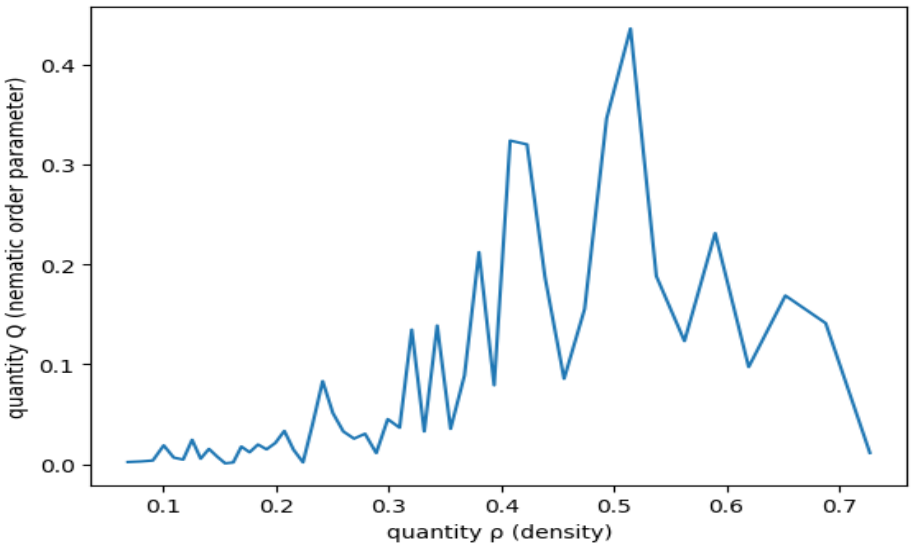

Figure 3: Q vs ρ for k=10, L=90

Meanwhile, the following are the plots of order parameter Q as a function of ρ for various k and L as given by the paper by D.Dhar and A.Ghosh.

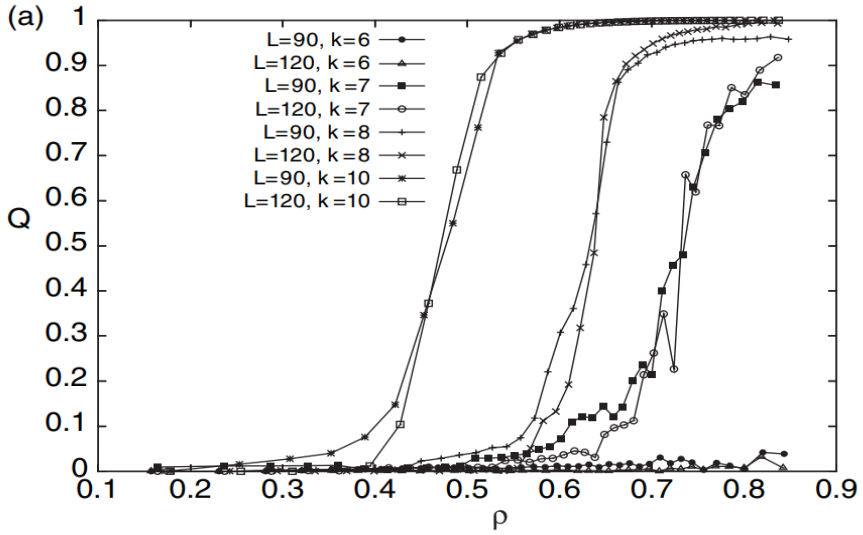

Figure 4: The order parameter Q as a function of densities ρ is shown for different k and L

While in the first glance, it is seen that Figure 3 resembles very little of the case of k=10, L=90 in Figure 4. But at the same time they have a similarity that after a certain value of p, Q has peaks that rise in a similar manner to the graph of Figure 4. On the other hand, apart from the numerous crests in our result, the maximum value for Q is also much less which is near 0.4, while in the paper it reaches as much as 9.8.
We would like to mention here that simulations we ran for N= **1x**10^6 **and 2x**10^6 gave results that had no resemblance to the graph in Figure 4, that we had to depend on a more time-intensive N as 5x10^6.

Since the graphs for N=5x10^6 were an improvement over the previous choices of N, we next tried N=10x10^6 with m=10000, and l=500 which gave the following results:

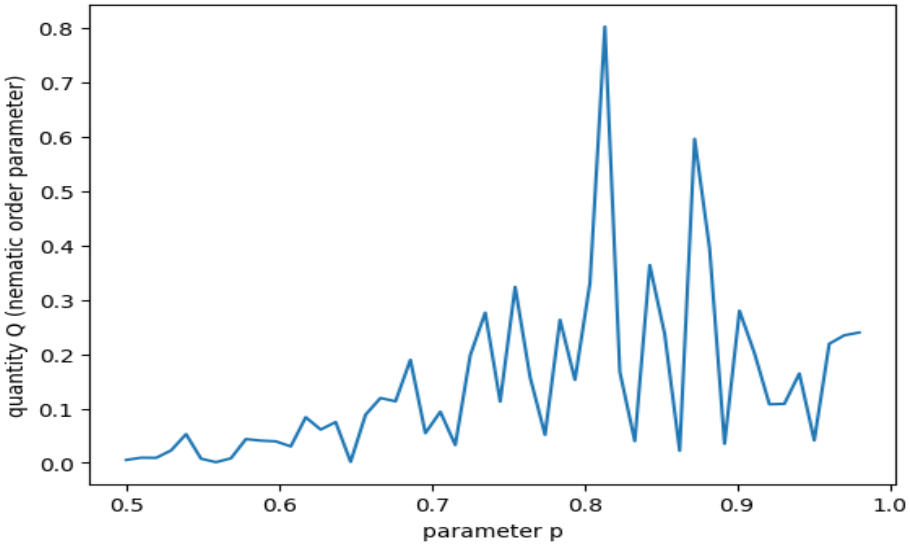

Figure 5: Q vs p for k=10, L=90

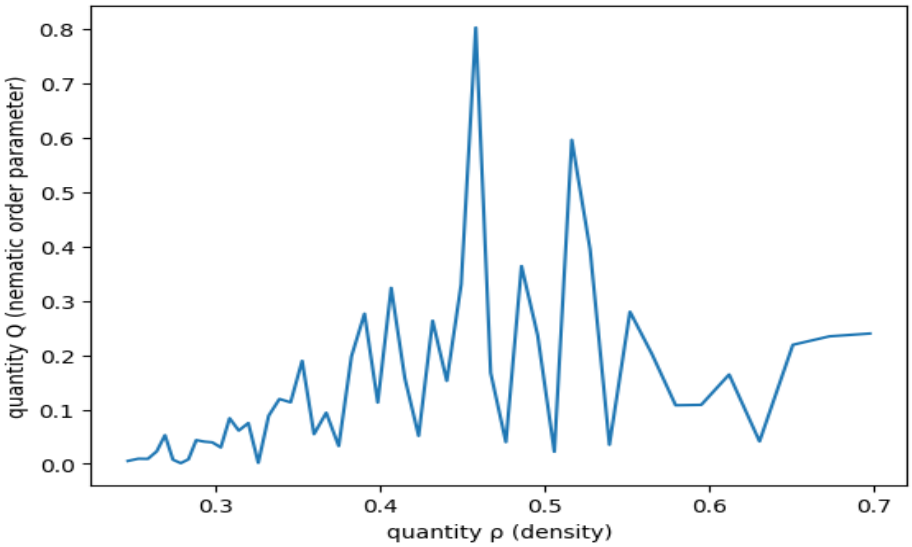

Figure 6: Q vs ρ for k=10, L=90

The problem of crests/fluctations did not go but the maximum value went to 0.8 which is an improvement over 0.4 from the previous trial.

6 Conclusion And Improvements

:

The paper "On the orientational ordering of long rods on a lattice" by A. Ghosh and D. Dhar provides a detailed study of the orientational ordering of long rods on a two-dimensional lattice. The authors use Monte Carlo simulations to investigate the behavior of the order parameter as a function of density for various values k and L.

7 Bibliography

:

- Orientational ordering of long rods on a lattice -D.Dhar, A.Ghosh
- Statistical mechanics, entropy, order parameters and complexity - James.P.Sethna
- An Introduction to computational physics - Tao Pang