Università degli studi di Trieste
Corso di laurea magistrale in Fisica della materia

Laboratorio di Fisica Computazionale

Unit X
Lattice gas model

Davide Bernocco
Matricola s285296

Programming language used:
Python

Contents

1 Self-diffusion coefficient in a lattice gas model

One of the easiest way to simulate diffusion phenomena, as we saw in unit IV, is through *Random Walks (RWs)*. However, if we are interested into studying a gas of interacting particles a more sophisticated approach is needed.

First of all, let us restrict to the *discrete* case and consider a finite square lattice of size $L * L = N$ filled with a certain number of particles N_p such that $N_p \leq N$. The particles can then move on the lattice by jumps to one of the nearest sites, as long as it is not already occupied, according to the Metropolis MC method with a *Boltzmann-like* trial function. This is a typical implementation of a restricted RW in normal behaviour.

The quantity we are mostly interested in is the *instantaneous diffusion* $D(t) = \frac{1}{2td}\langle \Delta R(t)^2 \rangle$ where $\langle \Delta R(t)^2 \rangle$ is the *mean square distance* (the average made on the particles!) and d the system dimension ($d = 2$ here). At very large t the instantaneous diffusion should asymptotically converge to a fixed value, the *autodiffusion coefficient* $D = \lim_{t\to\infty} D(t)$.

Now, contrary to what has been done in unit IV, to study this kind of system we need to simulate the motion of all the particles at the same time, not taking the average over N_p independent single particle motion. The logical passages implemented within the algorithm are the following ones:

- Once L and N_p (and thus the concentration ρ) are chosen, the particles are randomly allocated into available empty sites.
- Loop over a predetermined number of MC steps: during one of these units of times each particle attempts (in a random order) one jumps with a simplified Metropolis approach:

$$\Delta E = \begin{cases} 0 & \text{if no overlap} \\ +\infty & \text{if overlap} \end{cases} \quad \Longrightarrow \quad e^{-\beta \Delta E} = \begin{cases} 1 & \text{new configuration ACCEPTED} \\ 0 & \text{new configuration NOT ACCEPTED} \end{cases} \tag{1}$$

- At the end of each MC step the particles positions are updated and we evaluate $\langle \Delta R(t)^2 \rangle$ and $D(t) = \frac{1}{4N_i^{MC}}\langle \Delta R(t)^2 \rangle$ (where the time is parameterized through the N_i^{MC})

The autodiffusion coefficient is then estimated performing a block average on the obtained data of $D(t)$, as well as its relative fluctuation.

We conclude this brief introductory section with a note: since we are dealing with motion in a finite systems, it is important to specify what happens at the border of our lattice. We decided to apply PBC in the algorithm, aware of the fact that this choice will have consequences on the diffusion behaviour on the long term. In fact, reliable results can be obtained only for $\langle \Delta R(t)^2 \rangle < (L/2)^2$ and going beyond this value, the obtained data can be affected by the imposed periodicity: this, in turn, set a limit on the number of MC steps. In §1.2 we will study this peculiar effect looking at diffusion fluctuation.

1.1 Mean square distance

Since we are dealing with multiple *Random Walks*, the first thing we can do is studying the trend of $\langle \Delta R(t)^2 \rangle$ as a function of t and comparing it to what we know about independent RWs. In particular, the mean square distance of a single RW in normal behaviour is characterized by a linear law in t.
Fig. 1.1 shows the gathered points for a numerical simulation on a lattice of fixed size $L = 100$ for different values of density: the logarithmic scale helps us to recognize a linear trend:

$$\langle \Delta R(t)^2 \rangle \sim t \tag{2}$$

This tells us that, over the time, the particles progressively step away from their initial positions, giving rise to the diffusion phenomenon we originally wanted to simulate. The distinct values of y-intercept, however, provide us a more detailed interpretation: it is true that in all the analyzed cases the points lie on a line, but the slopes are not the same. In other words, as the density grows, the interacting particles on average have at disposal fewer and fewer free sites to which move. As a result, just for a small number of particle the behaviour resembles the one of independent RWs, while for $N_P \to N$ particles hardly manage to move away from the initial positions resulting in smaller men square positions.

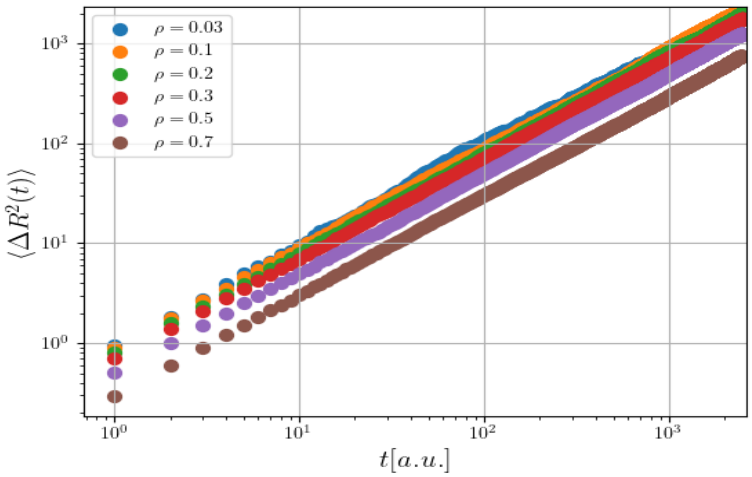

Fig 1.1 Mean square distance as function of t, different ρ (log-log scale).

1.2 First remarks: equilibration and diffusion fluctuation

Just to be more clear and provide a concrete example, let us start with $L = 20$ and $\rho = 0.2$ (i.e. $N_p = 80$) on a run with $N_{MC} = 10^5$.

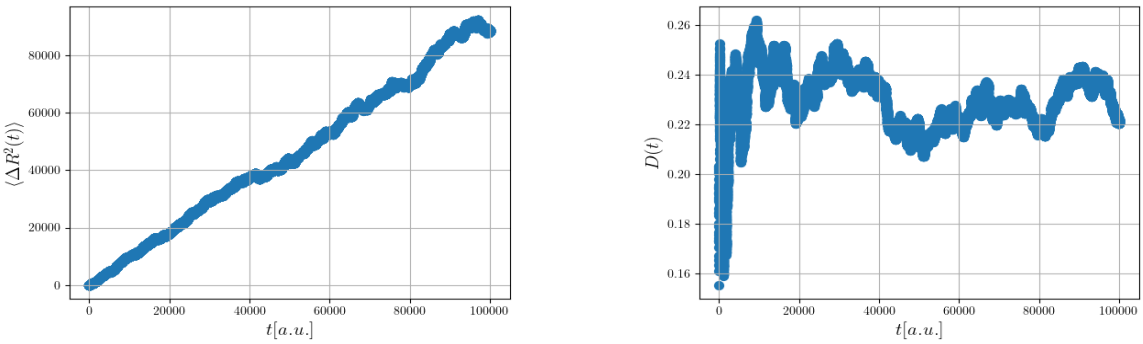

Fig. 1.2 Numerical simulation for a) the instantaneous mean square position $\langle \Delta R(t)^2 \rangle$ and b) instantaneous diffusion $D(t)$.

Fig. 1.2 shows the collected data for $\langle \Delta R(t)^2 \rangle$ and $D(t)$[1]. Looking at the instantaneous diffusion, two things stand out: firstly we recognize an initial equilibration sequence typical of a Metropolis MC - like algorithm and secondly $D(t)$ keep oscillating even at high t.
To begin with, we would like to remove the points generated within the first equilibration steps in order to perform more accurate averages on the dataset later on. A piece of extra code has thus been added at the beginning of the algorithm so that only after a proper time the quantities of interest are saved. It is based on block averages: just when the absolute difference between two consecutive means are less than the threshold value of 0.001 the code actually start collecting data. In fig 1.3 are shown the points from the same previous seed, after the equilibration procedure has been applied[2].

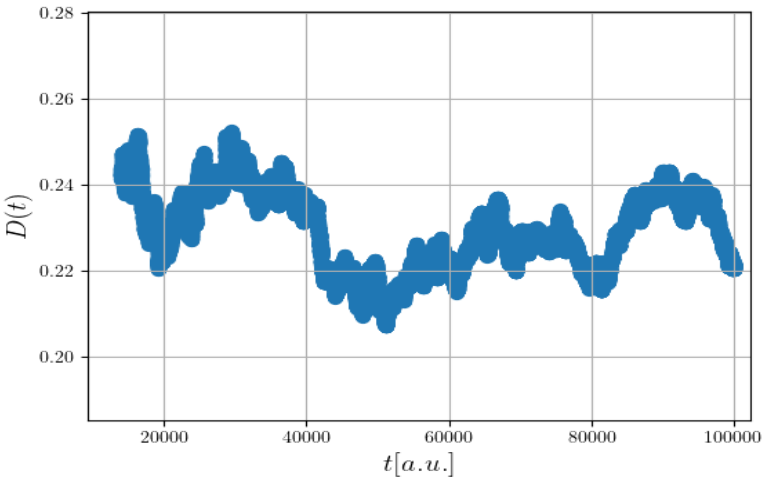

Fig 1.3 Instantaneous diffusion after the equilibration phase.

As for the fluctuations, it would be interesting to understand whether their amplitude change with time and if so, how much. To do so we followed a very simple approach, still reasoning by blocks: in fig. 1.4 are plotted the standard deviations σ_s performed on consecutive groups of 100 $D(t)$ points. After a

[1] Here t denotes the number of MC steps
[2] $1.4 * 10^4$ points removed here

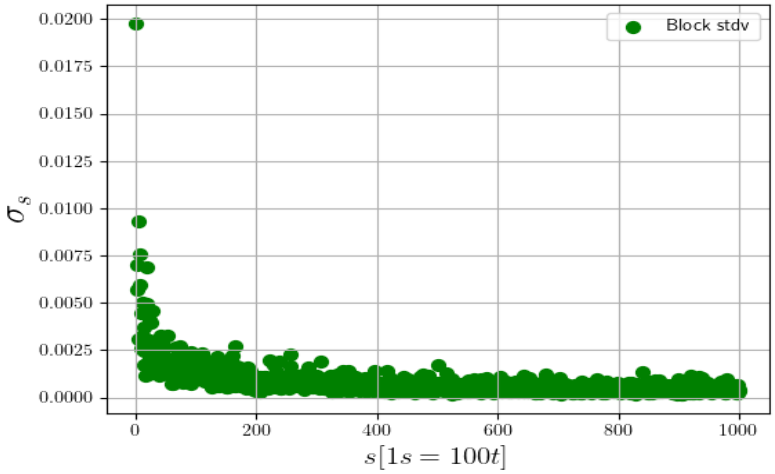

Fig 1.4 Fluctuations amplitude as function of t.

first decreasing trend corresponding to the equilibration sequence, the fluctuations seems to settle around a constant value which is very low but not naught. This fact confirms that the instantaneous diffusion oscillates and keep doing it even for long values of t, with an almost constant non-vanishing amplitude.

The oscillatory behaviour just described can be seen as a direct consequence of imposing PBC on the system. Because of this peculiar condition at the border, particles spontaneously tend to alternate phases in which are closer to each other - and thus characterized by a lower instantaneous diffusion - with phases in which are relatively farther - bringing to a larger instantaneous diffusion.

A further validation of this hypothesis comes from the comparison between the instantaneous diffusion simulated for different densities. This time we will not evaluate the fluctuation for each single dataset, but we are content with the qualitative estimation that can be done looking at fig. 1.5. What we focus our attention on here is not the spacing between the plots[3], we are rather interested in fluctuations amplitude: in fact, at larger density they become smaller and smaller. This perfectly fit with our interpretation on the effect of PBC on the diffusion: with $N_p \to N$, there is less room for density fluctuation both in the bulk and - most importantly - at the border.

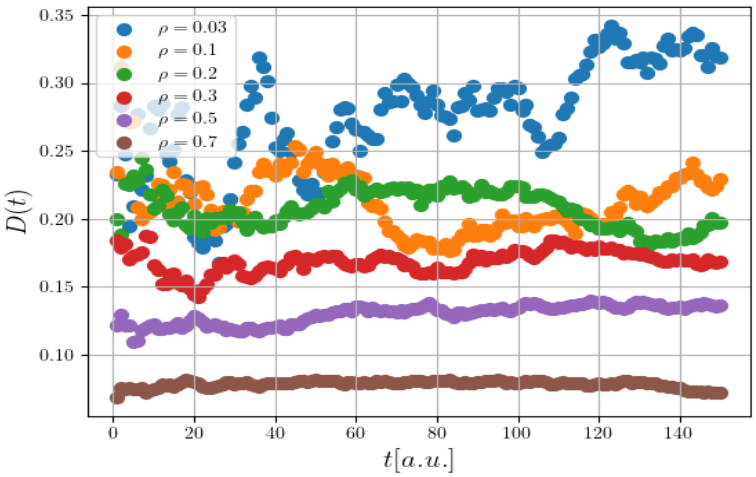

Fig 1.5 Intantaneous diffusion as function of t, different ρ (L=20).

[3]This issue will be addressed in the final part of the report

1.3 Numerical estimation of the autodiffusion coefficient D

Let us focus now on finding a numerical estimation of the autodiffusion coefficient D (alongside its corresponding error), that is the primary goal when implementing this lattice gas model. After removing the equilibration sequence, a sample of $(L/2)^2$ is considered from the $D(t)$ dataset so that - on average - the PBC effect are minimal. Then, a block average is taken on these points and this will provide an average value of D with its uncertainity σ_D.

At this stage, a more accurate estimation can be obtained implementing two upgrades: we can make an average over many runs keeping L and N_p fixed, but also consider larger lattices with the same density. So, if we select $\rho = 0.03$, instead of picking the usual $L = 20$ we can choose $L = 100$ and in addition perform an average on 10 runs. What we obtain is

$$D_{\rho=0.03} = 0.24 \pm 0.01$$

This is a satisfying result since we know that at low density the interacting gas tends to behave like a collection of independent RWs, whose analytical solution for D is $1/4$.

1.4 Behaviour of σ_D as function of N_p

Since we first talked about MC simulations, the issue of how to provide a reasonable estimation of uncertainty over numerical results has been of fundamental importance, especially when dealing with correlated data. In the previous section we operated *block averages* and now we would like to check whether the σ_D dependence on N_p follows the predicted $1/\sqrt{N_p}$ behaviour. In fig. 1.6 are reported the block-averaged fluctuations σ_D coming from *single* runs of increasing L (at fixed density, obviously!).

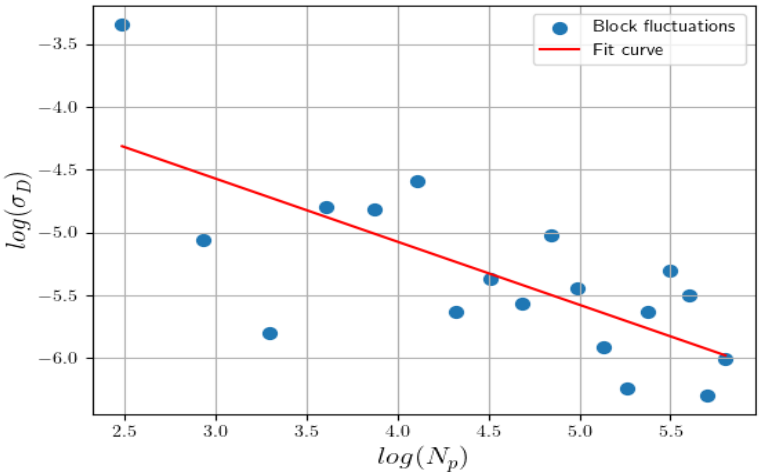

Fig 1.6 Numerical error on D as function of the number of particles: different L, $\rho = cost$ (log-log scale).

The slope of the linear fit on the log of these data is -0.5 ± 0.1 : although in a rough way, it confirms our expectations. Clearly, if we took σ_D as the deviation of a set of coefficients D_i from their mean value D obtained in a a multiple-run experiment, and studied how it goes with N_p, we would get a more precise estimation of the same fluctuation trend.

1.5 Dependence of D on the concentration

We conclude investigating a phenomenon we already mentioned when analyzing fig. 1.5, that is the apparent dependence of the autodiffusion coefficient on the concentration. On a lattice of size $L = 50$ 30 independent runs have been carried out in order to estimate D for different values of ρ. The gathered points with their corresponding uncertainties[4] are plotted in fig. 1.7 alongside a linear interpolation curve $y = a + bx$. The obtained coefficients are:

$$a = 0.250 \pm 0.001$$
$$b = -0.250 \pm 0.001$$

In this way we quantitatively validated the initial guess on the dependence on ρ, finding out the trend is linear monotonically decreasing. A possible formal explanation can be traced back to the mean square position: since it grows slower as the concentration is increased and because of how the instantaneous diffusion is calculated out of $\langle \Delta R(t)^2 \rangle$ (basically dividing it by $4t$), the monotonically decreasing $D(\rho)$ behaviour comes out quite straightforward as a consequence.

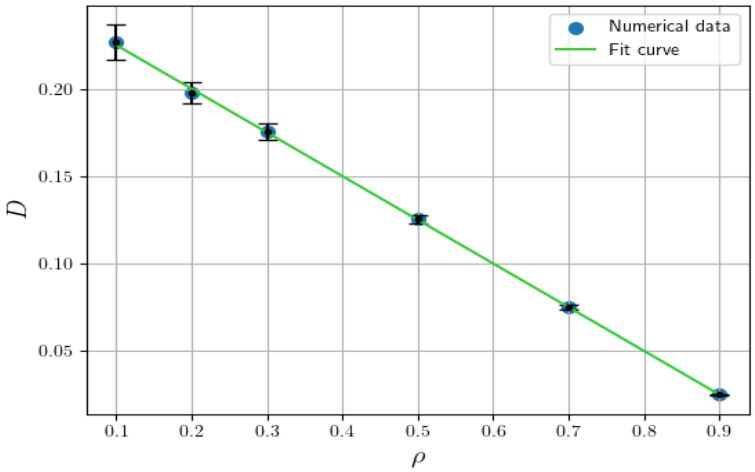

Fig 1.7 Autodiffusion coefficient as function of particles concentration ($L = 50$).

[4] From error propagation $\sigma_{D,TOT} = \sqrt{\sigma_{D,1}^2 + \sigma_{D,2}^2 + ... + \sigma_{D,30}^2}$