

UE Projet Scientifique
The Cancer Stem Cells Hypothesis :
Interacting cells dynamics

Jakob Hack, Lucas Lienard

January 2023

Contents

1	Introduction	1
1.1	The cancer stem cell hypothesis	1
1.2	The dynamics of cell to cell interactions	1
1.3	Objectives	1
2	Theoretical background	2
2.1	Mathematical background	2
2.1.1	Process and update formula	2
2.1.2	Random walk's mathematics : Continuous Markov processes	2
2.1.3	The Langevin's equation	2
2.1.4	Ornstein-Uhlenbeck processes	3
2.2	Confined biological systems	3
2.2.1	Relationship with the random walk	3
2.2.2	Interaction of cells in a confined system	4
2.2.3	Contact acceleration map : G-function	4
3	Method	6
3.1	Experiment	6
3.2	Approach-data analysis of τ σ and G	6
3.3	Simulation viewpoint	6
3.3.1	Programming language, code editor and packages used	6
3.3.2	Simulation of a random walk	7
3.3.3	Simulation for a large amount of cells	7
3.3.4	Plotting accelerations against velocities	7
3.3.5	Insertion and propagation of an error in positions	7
3.3.6	Issues while coding interaction map	8
4	Results	9
4.1	Data analysis	9
4.2	Simulations	10
4.2.1	The random walk	10
4.2.2	Accelerations against velocities	10
4.2.3	Large number of cells random walk	11
4.2.4	Propagation of the error	12
5	Discussion	14
6	Conclusion	14
7	Annexes	15

1 Introduction

1.1 The cancer stem cell hypothesis

The cancer stem cell hypothesis is a theory regarding the growth and development of tumors. It states that a small subset of cancer cells is responsible for its growth and initiation as can be seen in section A of figure 1.1. These cells possess the abilities of indefinite self-renewal, slow replication, intrinsic resistance to chemotherapy and radiotherapy and giving rise to differentiated progeny. As pictured in section B of figure 1.1 the cancer stem cells resistance towards chemotherapy and radiotherapy makes it hard to treat the cancer. Some parallels to normal stem cells can be observed: multipotency, that is, the ability to differentiate into different cell types; self-renewal; and the ability to proliferate. However it is important to note that although cancer stem cells share many characteristics with normal stem cells, cancer stem cells do not necessarily derive from normal stem cells.

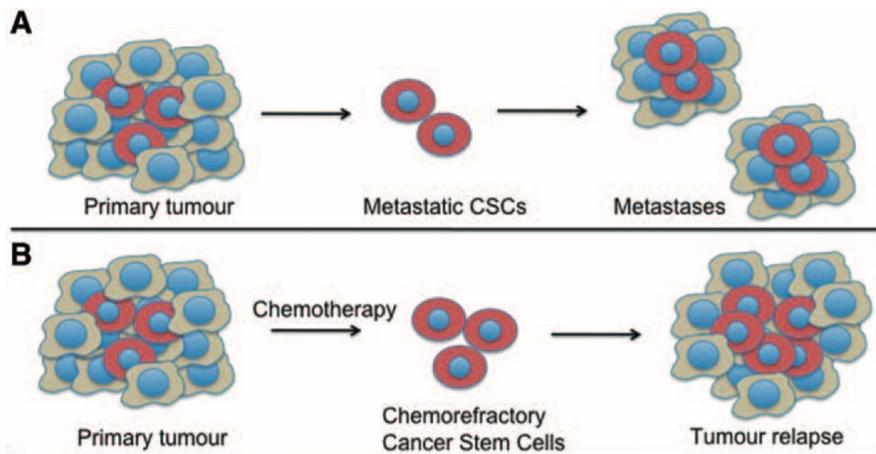


Figure 1: potential implications of cancer stem cell hypothesis[1]

It is interesting to investigate cancer stem cells because new cancer treatment possibilities may arise. Especially can ways be investigated to take action against formation of metastases or recurrently appearing tumors which cancer stem cells are capable to form.[1]

1.2 The dynamics of cell to cell interactions

Contact mediated cell to cell interactions are a cornerstone to migratory dynamics in physiological processes like cancer metastasis. As stated in [2], repulsion and friction interaction can be observed between noncancerous cells. In cancerous cells attraction and antifriction behavior can be observed.

1.3 Objectives

The objective of this project is on the one hand to extract an interaction term between cells from experimental data and on the other hand to generate an interaction term in a mathematical simulation. The objective is to compare the two obtained interaction terms and assess their validity.

2 Theoretical background

2.1 Mathematical background

This section was mainly written by using this article [3] as a support.

2.1.1 Process and update formula

For instance, let V be a physical quantity, we know that this quantity will change over time, following this well-known equation :

$$\frac{dV(t)}{dt} = A(V(t), t) \quad (1)$$

This equation can be rewritten :

$$V(t + dt) = V(t) + A(V(t), t)dt \quad (2)$$

We say that equation (2) is the **update formula** of the **process** V . We must point out the fact that this formula returns a value of the process V at a later time $t + dt$ and this, without requiring the value of V before time t .

We can generalize this equation to some non-deterministic processes that are called **continuous Markov processes**. To such processes are assigned definite probabilities of value at time $t + dt$ knowing the value of the process at time t .

2.1.2 Random walk's mathematics : Continuous Markov processes

A random walk is a mathematical model of a system characterized by a discrete dynamic : a path is created by a succession of random steps. Such systems are described by what we call a random process. In particular, a continuous Markov process is a random process. These processes are nowadays characterized with robust mathematics. This is why we are interested in these.

If the result at time $t + dt$ depends only on t , dt and $V(t)$, then a continuous Markov process V must follow :

$$V(t + dt) - V(t) = A(V(t), t)dt + \sqrt{D(V(t), t)}N(t)\sqrt{dt} \quad (3)$$

Which is basically the update formula with an extra stochastic term constructed from a smooth positive function D . In this equation, N correspond to the unit normal random variable with the particularity of not depending on t , meaning that the random variable N is the same for all time t : $N(t) = N(0, 1)$.

By adding V on both side of the previous equation, we get the **standard form Langevin equation** for the process V :

$$V(t + dt) = V(t) + A(V(t), t)dt + \sqrt{D(V(t), t)}N(t)\sqrt{dt} \quad (4)$$

2.1.3 The Langevin's equation

Some manipulations can be done for our Langevin equation to be more convenient. We can start by dividing both sides of equation (3) by dt , leaving us with :

$$\frac{V(t + dt) - V(t)}{dt} = A(V(t), t) + \sqrt{D(V(t), t)}\frac{N(t)}{\sqrt{dt}} \quad (5)$$

Let's recall the fact that :

$$a + bN(t) = N(a, b^2) \quad (6)$$

We directly get by substitution :

$$\frac{N(t)}{\sqrt{dt}} = \mathbf{N}(0, \frac{1}{dt}) \quad (7)$$

We can now create a stochastic generalization of equation (1) by previously define the **Gaussian white noise** $\Gamma(t)$ as :

$$\Gamma(t) = \lim_{dt \rightarrow 0} \mathbf{N}(0, \frac{1}{dt}) \quad (8)$$

Giving us the **white noise form Langevin equation** :

$$\frac{dV(t)}{dt} = A(V(t), t) + \Gamma(t) \sqrt{D(V(t), t)} \quad (9)$$

The introduction of the Gaussian white noise is useful mainly because of two of its properties :

- (i) the noise has a zero mean : $\langle \Gamma(t) \rangle = 0$
- (ii) the noise is Dirac-correlated : $\langle \Gamma(t)\Gamma(t+dt) \rangle = \delta(dt)$

Finally, we can state the following differential equation for the process V, which will be very useful at some point in the project to compute positions knowing the velocities :

$$\frac{dX(t)}{dt} = V(t) \quad (10)$$

Note that, since equation (10) doesn't have the Langevin form, the integral X of the continuous Markov process V is not a continuous Markov process itself. The form of equation 10 gives us the equivalent update formula :

$$X(t+dt) = X(t) + V(t)dt \quad (11)$$

2.1.4 Ornstein-Uhlenbeck processes

We define an Ornstein-Uhlenbeck (O-U) process as a continuous Markov process such that the drift and diffusion function have the forms:

$$A(V(t), t) = -\frac{1}{\tau}V(t) \text{ and } D(V(t), t) = c \quad (12)$$

Where c is the **diffusion constant** (note that c is not equal to the diffusion coefficient) and τ is the **relaxation time**. We then obtain a Langevin equation under the following form :

$$\frac{dV(t)}{dt} = -\frac{1}{\tau}V(t) + \Gamma(t)\sqrt{c} \quad (13)$$

This equation describes the behavior of what we call a Brownian particle : a particle described by it's velocity V, where V is an Ornstein-Uhlenbeck process.

2.2 Confined biological systems

2.2.1 Relationship with the random walk

Alone, a cell acts and moves following a Brownian motion, meaning we can describe its behavior as being a stochastic process. For systems involving more than one cell, interactions appear : depending on the studied cell, we can see either a repulsion-friction interaction, or an attraction-antifriction (antifriction meaning sliding) interaction. This is leading cells to behave in a more organized way.

A group of researcher made some experiments in which they confined two cells in a way that allows them to study cell-cell interactions and wrote an article [2] on which this section is based.

2.2.2 Interaction of cells in a confined system

We know very little about the stochastic dynamics of interacting cells. According to the article [2], it seems like every cells pairing tend to adopt an exclusive behavior (meaning that the two cells prefer to repel each other rather than collide).

A collision is defined as an event where the two cell nuclei approach each other closer than a threshold length. Even if the cells prefer a mutual exclusion, collisions still occur, giving us data to exploit. What stands out is that we can classify the post-collision states of the system under three main types of behavior depicted in figure 2:

- (i) Reversal event : the cells turn around during the collision
- (ii) Sliding event : the cells slide on each others, interchanging their positions
- (iii) Following event : the cells temporarily move together, but such events seem to be more rare than the two previous ones

To study the dynamics of multi-cells systems, we need to study their accelerations as it is the natural quantity that captures motions.

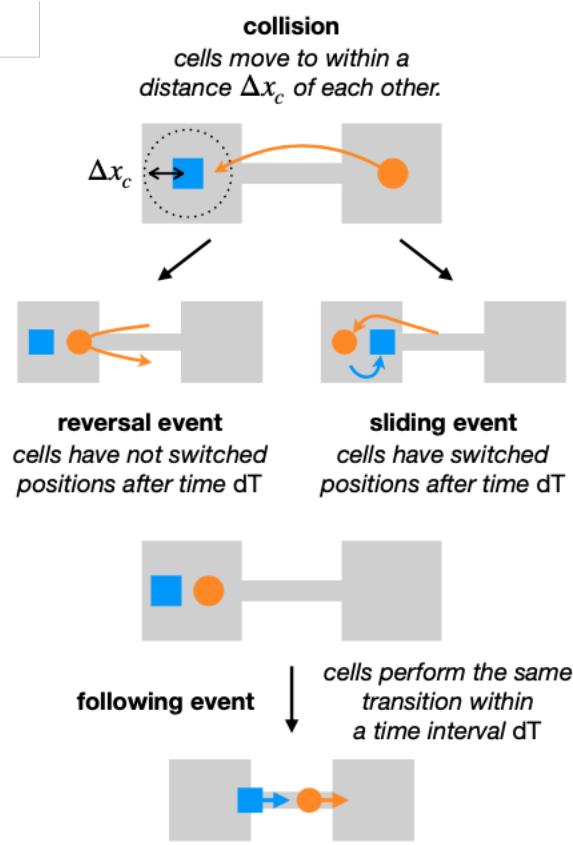


Figure 2: different interactions [2]

2.2.3 Contact acceleration map : G-function

A simple model to model the cell dynamics is given in [2] by

$$\frac{dv}{dt} = F(x, v) + f(|\Delta x|)\Delta x + \gamma(|\Delta x|)\Delta v + \sigma\eta(t). \quad (14)$$

$F(x, v)$ is the one-body term and also describes the interaction between each cell and its confinement. $f(|\Delta x|)\Delta x$ is an interaction term describing attraction and repulsion. $\gamma(|\Delta x|)\Delta v$ is an interaction

term describing friction. $\eta(t)$ is a Gaussian white noise term and accounts for the stochasticity. A contact acceleration map describes the interaction dynamics. This is done by plotting the interaction component of the acceleration $G(\Delta x, \Delta v)$ over Δx and Δv . Following [2] $G(\Delta x, \Delta v)$ is estimated as

$$G(\Delta x, \Delta v) \approx \dot{v}_i - F(x_i, v_i) \quad (15)$$

The different kinds of interactions introduced in 2 can be illustrated as in 3.

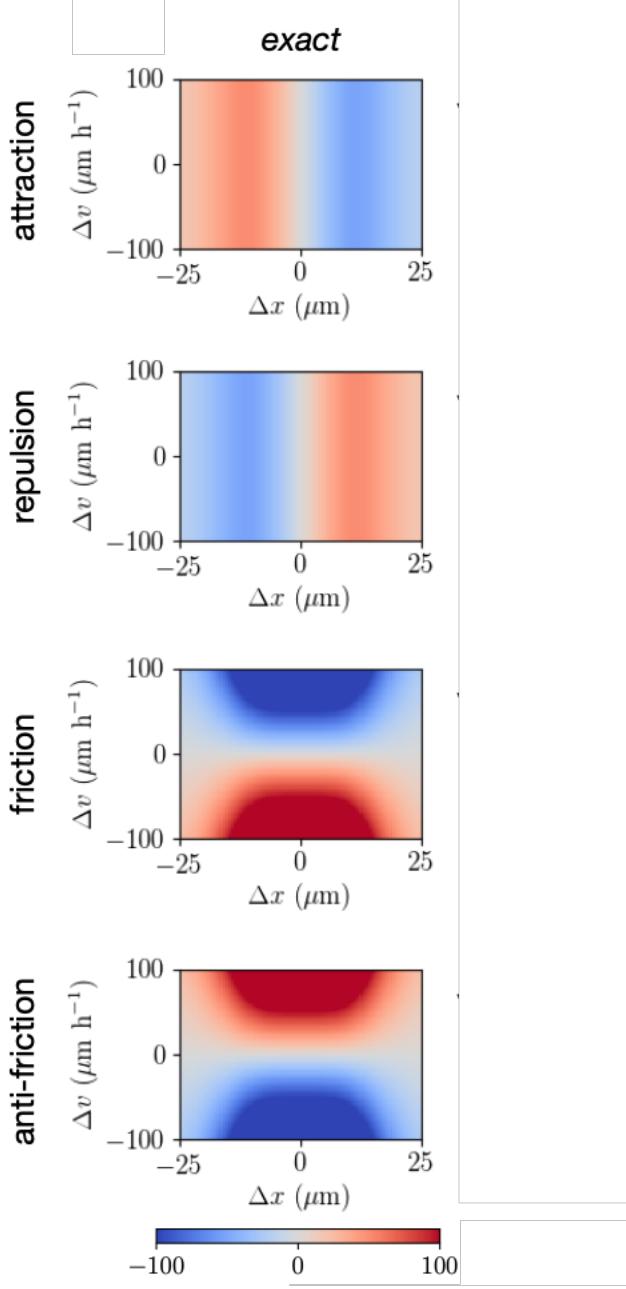


Figure 3: Contact acceleration maps for different interactions [2]

3 Method

3.1 Experiment

For the experiment to generate the data, breast cancer cells were seeded on a petri dish and were imaged for four days using fluorescence microscopy. By segmentation software trajectory data was created for the cells.

3.2 Approach-data analysis of τ σ and G

For the τ -analysis, the data generated in the experiment was analysed numerically. This was done by taking into account only the non-interaction term of equation 14 meaning that the isolated cells (these are the cells with no contact to neighbouring cells) from the cell data were extracted. In this case we set the cut-off length for the isolated cells to $150\mu m$. Cells that are further away from each other than this length are called to be isolated. The non-interaction term of equation 14 looks like this:

$$\dot{v}_i = F(x_i, v_i) + \sigma \eta_i(t) \quad (16)$$

The second term in equation 16 can be neglected because for Gaussian white noise: $\langle \eta(t) \rangle = 0$ and we average over a large amount of timesteps. Because the cells in the experiment underlie no confinement the first term consists only of the drift function introduced in equation 12. Hence τ is given by the slope of $a_i(v_i)$

$$\frac{dv_i}{dt} = -\frac{v_i}{\tau}. \quad (17)$$

v being the velocity of a cell. σ is given by the standard deviation of τ . To extract the isolated cells from the data we calculated distance matrices for every timestep in the recording. The entries of this matrix are the distance between the cell in line m and the cell in line n . We did this because the extraction of the isolated cells is repeatedly occurring in the program and thus it is time saving to calculate distance matrices in advance and accessing them repeatedly. The calculations were still very demanding for a usual laptop and the distance matrices use a lot of memory so the calculation were conducted on a remote server provided by the university of Lille.

To dive more deeply into the calculation of the distance matrices we have to explain the structure of the data that was generated in an experiment: at this point the relevant parameters for the data analysis are the x- and y-position which were given in μm , the cellID (an ID that gets assigned to every cell) and the image ID (during the 4 days of the experiment every 0.71h a picture was taken so images with imageID 1 and 2 are 0.71h apart).

To get the velocity (in x direction) of a cell at a given timestep we used the following formula:

$$v_x(t_i) = \frac{x(t_{i+1}) - x(t_i)}{\Delta t} \quad (18)$$

To get the acceleration we used the formula:

$$a_x(t_i) = \frac{v_x(t_{i+1}) - v_x(t_i)}{\Delta t} \quad (19)$$

When we plot the acceleration against the velocity we have to take care that we plot the acceleration against the velocity of the same cell. A lot of errors can occur here.

G function analysis:

Having obtained τ the cells with one neighbouring cell are extracted to get the interaction term.

3.3 Simulation viewpoint

3.3.1 Programming language, code editor and packages used

All the programs were written in **python** (programming language). The code editor used was **Spyder IDE**, on a **Linux OS** computer.

For scientific calculus, we mainly used the package **numpy**. For plotting the results, **matplotlib** was the one used. And the last one was the package **os**, to create folders directly from the programs. For

everything else, python built-in functions or self-defined functions were sufficient.

All the programs are on a github, the link is in the annexes.

3.3.2 Simulation of a random walk

In the simulation approach, our first goal was to generate a random walk. To do so, we used equation 4 in which we replace dt by Δt that plays the role of a timestep for the update formula and also replace $N(t)$ by n , a sample value of $\mathbf{N}(0, 1)$:

$$V(t + \Delta t) = V(t) + A(V(t), t)\Delta t + n\sqrt{D(V(t), t)}\sqrt{\Delta t} \quad (20)$$

We then substitute drift and diffusion functions by their expression for an O-U process to obtain :

$$V(t + \Delta t) = V(t) - \frac{1}{\tau}V(t)\Delta t + n\sqrt{c}\sqrt{\Delta t} \quad (21)$$

This is the form we'll use for coding the simulations. Note that, in the simulation, \sqrt{c} is evaluated in the data approach with a Gaussian fit, which is not exactly right. That being said, the general behaviour of a an isolated cell system will still be captured in the simulation.

Now that we have the velocity (accordingly to the Langevin equation), we need to extract the position. As said in the theoretical background part, we can compute the positions following equation (11):

$$X(t + dt) = X(t) + V(t)dt \quad (22)$$

In the case of the simulations, we use a timestep Δt , for this equation to still be valid toward equation (21), we need to take a timestep such that : $\Delta t \ll \tau$.

We will start by running simulations for a 1D random walk, and continue by making the program able to generate 2D random walks. We won't run simulations in 3 dimensions because measurements were extracted from the experiment in 2 dimensions.

3.3.3 Simulation for a large amount of cells

To generate data for a large number of cells, we stock velocities and positions in different files that are stocked in a folder with the cell's name, and this is done for N cells, giving us N folders with two files for positions (x and y) and two files for velocities (Vx and Vy) for each.

Next, we establish a program that read all the positions (or velocities, or both) of a given cell, or for as much cells as we want. The program then loops over all the selected directories and plots the results at the end.

3.3.4 Plotting accelerations against velocities

One of the main result for the data analysis approach was to plot accelerations of isolated cells against their velocities to directly extract a value for the relaxation time. Since this plot was necessary to make comparisons, it was the first interesting one.

As the Langevin equation gives us the evolution of the velocities, we need to manually compute the acceleration. In order to do so, we said that the acceleration between two consecutive timesteps corresponds to the change of velocity during the timestep:

$$a(t) = \frac{V(t + \Delta t) - V(t)}{\Delta t} \quad (23)$$

3.3.5 Insertion and propagation of an error in positions

During the real experiment, positions of the cells were taken at their nucleus, but it is difficult to be sure where the nucleus actually is. So, we have errors in the position of the cells.

The idea between the insertion of the error for the simulations is to see how these errors will spread

from the position to the velocities and from the velocities to the acceleration. The fact is that, maybe, the interaction term measured actually arises from errors in the positions measurement.

To introduce an error in the data was actually quite easy, we open the positions file we want, read all of the positions and stock them in an array. During the experiment, at each timestep, we had an error. We thus introduce an error in the whole array (at each timestep).

The error was generated by sampling a uniform distribution going from $-2 \mu m$ and $2 \mu m$.

3.3.6 Issues while coding interaction map

For establishing the code leading to an interaction map, we needed to create an algorithm that search for the neighbor of a selected cell, and study, locally, the rate at which the two cells approach or move away. Unfortunately, we didn't manage to create this program, on both the simulation and the data approach. This would have made a very good conclusion to our project. Still, we have some results we want to show up in the next section.

4 Results

4.1 Data analysis

In figure 4 we plotted the acceleration for every cell at every timestep in x direction over it's velocity. In figure 5 we did the same thing in y direction. By fitting a linear function and following equation 17 one gets the value of τ as the slope of the function. When the acceleration in y direction is plotted over the velocity in x direction or the other way around no more linear behavior but a random behavior is observed figure 6 and figure 7.

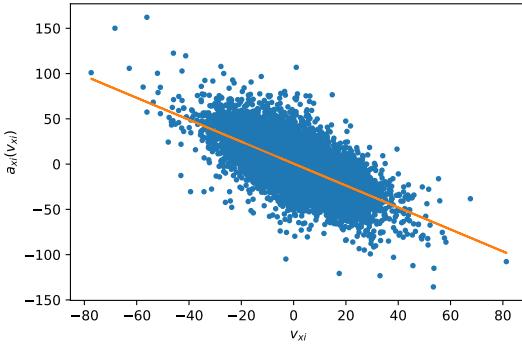


Figure 4: $a_{xi}(v_{xi}) \tau = 1.2$

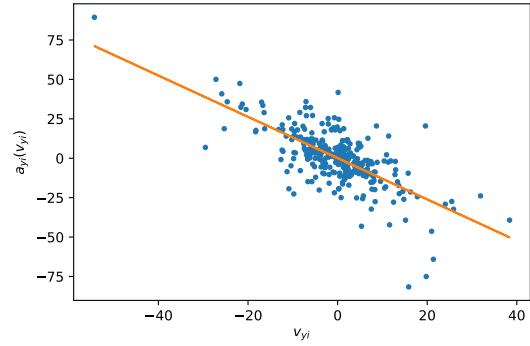


Figure 5: $a_{yi}(v_{yi}) \tau = 1.3$

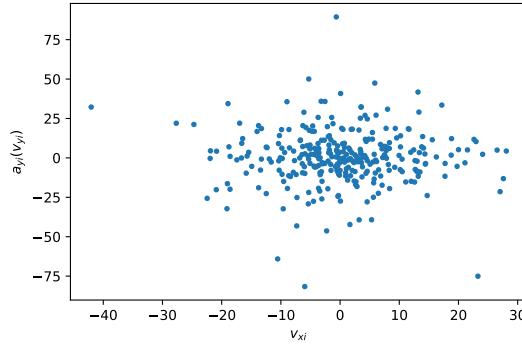


Figure 6: $a_{yi}(v_{xi})$

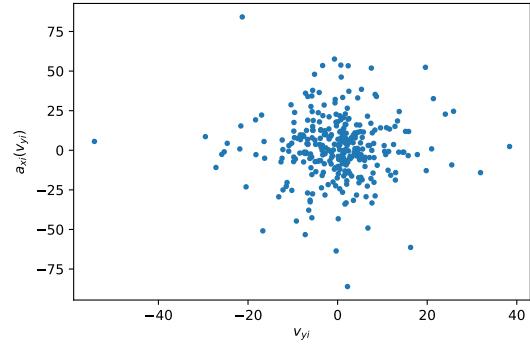


Figure 7: $a_{xi}(v_{yi})$

In figure 8 we plotted a PDF of $a_x + v_x$ and fitted a gaussian. The fit-parameter σ is 12.36

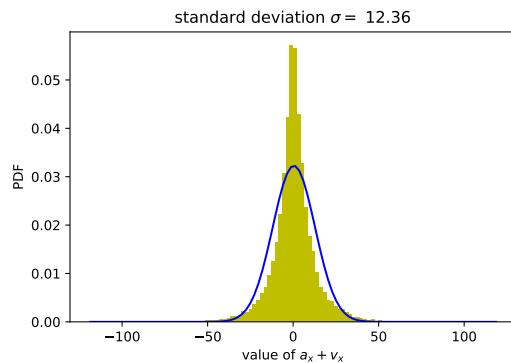


Figure 8: PDF

4.2 Simulations

4.2.1 The random walk

The first result for the simulation part consisted of a program returning a random walk in the sense of the Langevin equation. We can start by showing a random walk that doesn't satisfy the condition $\Delta t \ll \tau$:

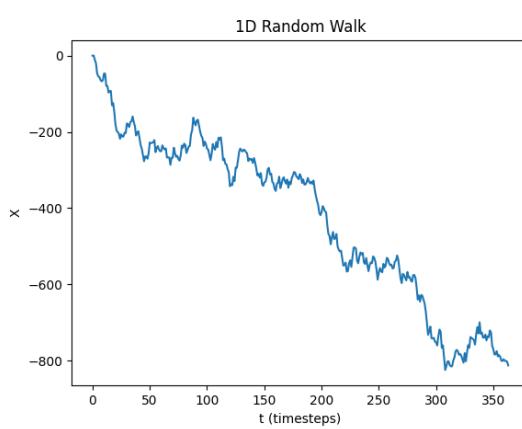


Figure 9: 1D random walk

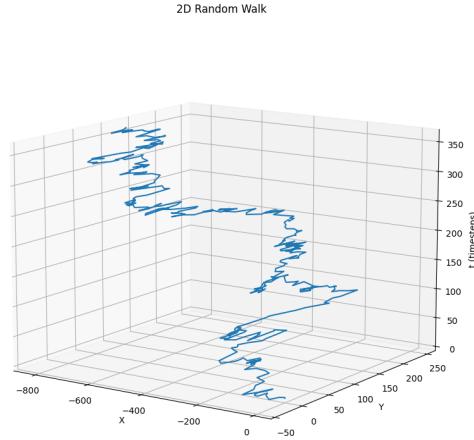


Figure 10: 2D random walk

Both figure 9 and figure 10 don't satisfy this condition because we took $\Delta t = \tau$. Now let's plot a random walk such that $\Delta t = \frac{\tau}{25}$ and see the difference : figures 11 and 12.

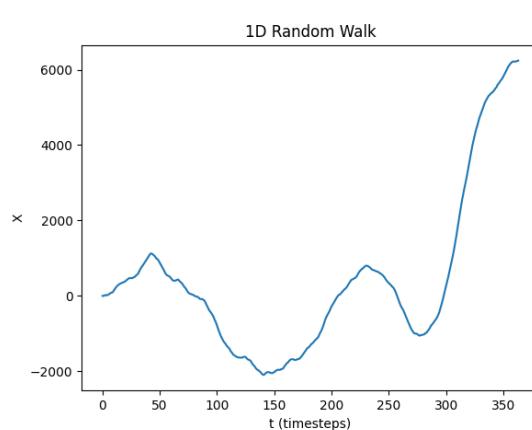


Figure 11: 1D random walk

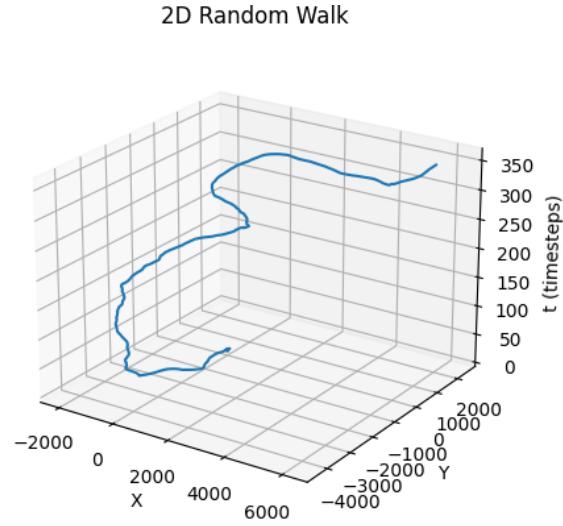


Figure 12: 2D random walk

We can observe the fact that, when the condition is respected, we have a smoother random walk in the sense that, at each timestep, we have a variation in position that is way smaller. Also, the "rate" at which the curve makes turnovers is far smaller as well.

4.2.2 Accelerations against velocities

Now that we have our random walk, we shall plot the acceleration of the cell against it's velocity. As explained before, this is useful because, with such a result, we can make our first comparison with the experimental side of the project.

To properly display the behavior of the cell, we will also plot the acceleration distribution by making an histogram of the accelerations in terms of the velocities : see figure 13.

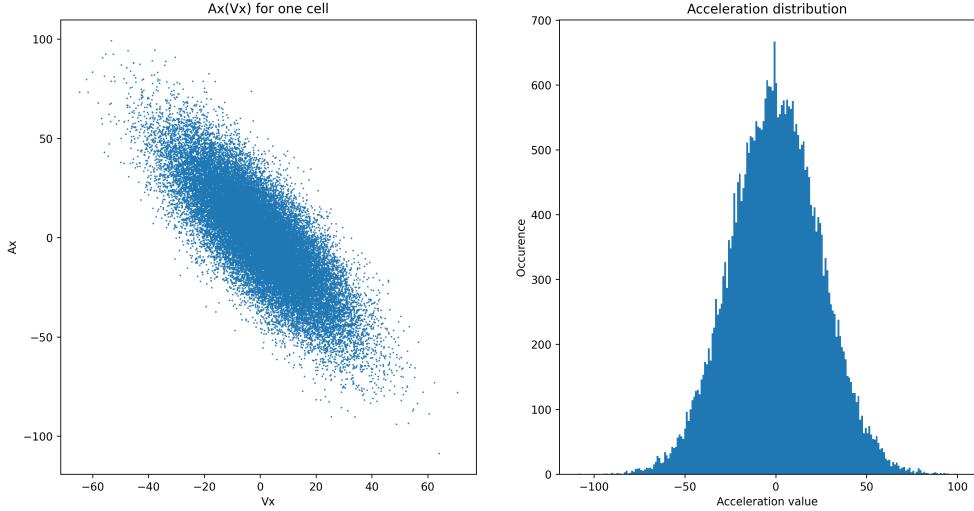


Figure 13: Accelerations and distribution for the 2D random walk of 1 cell

So, we're left with a normal distribution of the acceleration and more precisely a Gaussian one : this is not surprising because the cell moves randomly following a random normal distribution, invoking the central limit theorem, we must converge toward a Gaussian.

For this simulation, we took parameters that represent the real experiment. That is to say $\tau = 1.2$, $\sqrt{c} = 12.3$, $\Delta t = 0.71$ and ended the simulation at a time $t = stop$ such that : $t_{stop} = 96h = 135.2$ timesteps. This was done for an equivalent total of 27278 cells. To be precise, the simulation was launched for one cell, but with a timestep 27278 higher. It gives the same result because the cell walks randomly for a number sufficiently high of steps (27278 times more, so over 3.6 millions of timesteps). The reason we ran the simulation for a bigger time with only one cell, rather than with 27278 cells, is that our machine isn't sufficiently powerful to plot or even just save a .pdf figure with so many different cells.

4.2.3 Large number of cells random walk

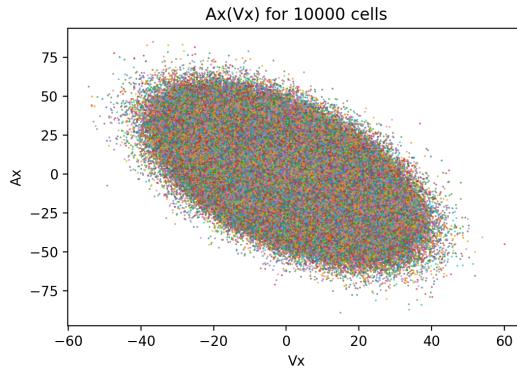


Figure 14: 2D random walk of 10000 cells

Following what we discussed before, even if we can't simulate a random walk for 27278 cells, we still managed to run one simulation for 10000 cells, which still represent a fair amount of cells: see figure 14. The parameters here are the same as the ones of the real experiment.

4.2.4 Propagation of the error

We begin with a quick reminder: we generate an error by sampling the random uniform variable going from $-2 \mu\text{m}$ to $2 \mu\text{m}$: see figure 15 where we sampled the random uniform variable 100000 times and plotted an histogram of the samples through 128 intervals.

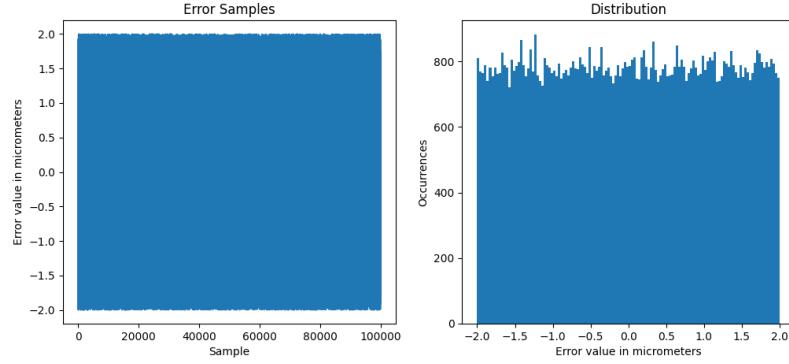


Figure 15: Representation of the distribution used

We can now introduce an error of this type in the positions. To start, we may want to get a better understanding of what's happening by plotting the error for a one dimension random walk. We won't plot the results in 2 dimensions since plotting the accelerations against the velocities is done in one dimension. To be exhaustive, we must say that it there is no significant reason to plots the system in two dimensions since, because of the nature of the random walk, there is no particular reason for the -isolated- cells to prefer moving in the y-direction rather than moving in the x-direction.

1D Random walk with error: first analysis

To simulate our unidimensional random walk, we'll take a cell similar to one of the experiment (same timesteps and parameters), introduce an error in the position list, and see how the error propagates in the velocities and the accelerations. The results are displayed in figure 16.

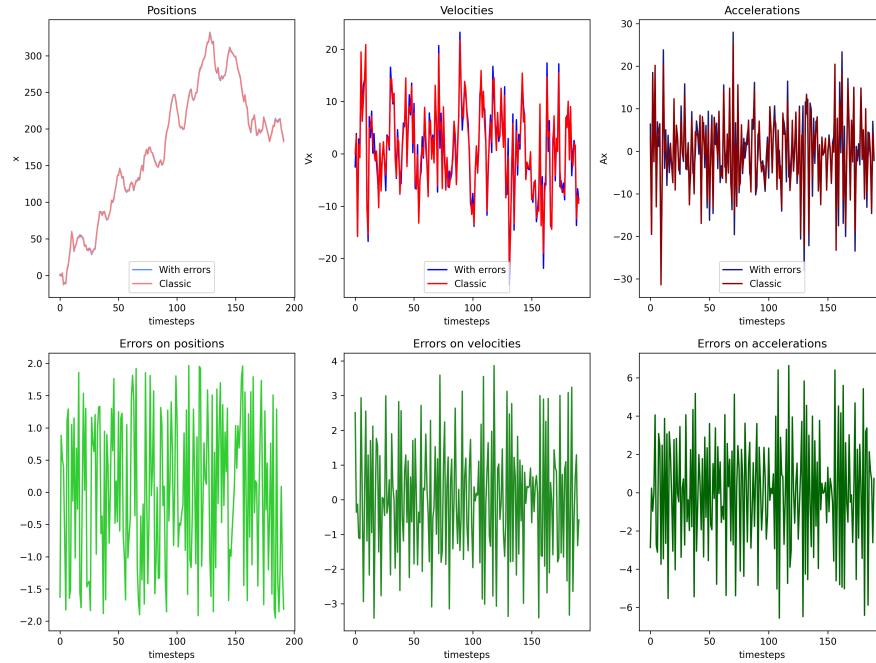


Figure 16: Error propagation for a 1D random walk

In the figure 16, the three graphs in green represent the difference between each physical quantity ($\Delta x = x - x_\epsilon$, $\Delta v = v_x - v_\epsilon$, $\Delta a = a_x - a_\epsilon$). We can observe in these results that, due to the propagation of the error, this error can increase up to three times the initial value. This is a serious deviation facing the fact that, in this simulation, the maximum value the acceleration takes is $a_{max} = 25.255 \mu m/s^2$. So, for this example, an error of $6 \mu m$ represent around 24% of the maximum acceleration value!

1D Random walk with error: accelerations against velocities

To properly illustrate the effect of this deviation from the initial acceleration, we can plot the accelerations against the velocities with errors and compare: see figure 17.

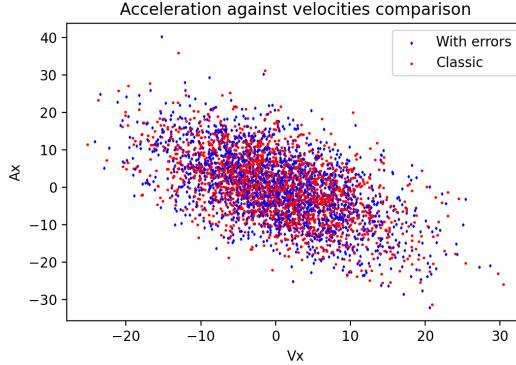


Figure 17: Accelerations errors

We observe no significant change in the distribution, and it is normal: even if the acceleration can reach up to $6 \mu m$, it may be at a timestep where the acceleration doesn't change that much without the error. Basically, the error just makes the behavior of the cell more chaotic, or less "smooth". Most of the time, the error remains small, thus not affecting the system that much.

Even if this analysis may be sufficient, we can still ask ourselves: What would happen in the case of a longer experiment or, equivalently, an experiment involving a huge amount of cells? We wanted to introduce an error in the positions to analyse the amplitude of this error as we end up making some plots of the accelerations. If the error is large, then maybe there's no reason to talk about functions of interaction (G function). If not, then it is very plausible that there actually is a notable interaction between the cells. This is the last thing we'll discuss in this section.

1D Random walk with error : an extension to very long simulations

For this last part, we will be running a simulation for 1 cell, for over a million of timesteps (to be precise: 1408452 timesteps). This should provide us with robust enough statistics to assess the amplitude of the error propagation. The results of the simulation are represented in the figure 18 and figure 19.

As we can see in figure 19, the distribution with and without error is almost the same, meaning it is still a Gaussian. So, basically, if we make some mistakes in the measures of the positions of the cells during an experiment, we're actually left with data that describe the same physics. Then, obtaining a Lorentzian instead of a Gaussian while doing a real-data analysis means that the physics of our system isn't completely captured by the Langevin equation : there must be an extra term that explains the behaviour of cells in interaction.

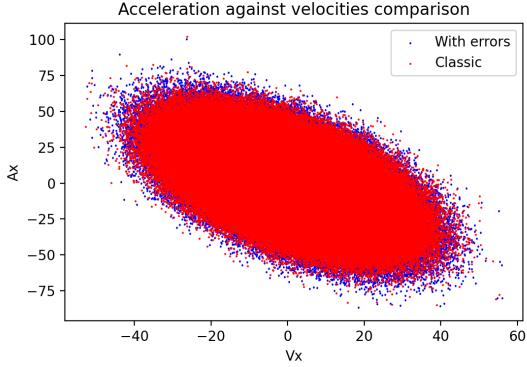


Figure 18: Error versus Classic : accelerations

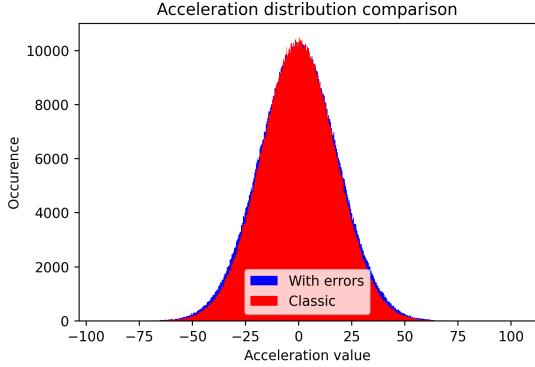


Figure 19: Error versus Classic : distribution

5 Discussion

Cells confined systems consist in a lot of cells constantly interacting with each others in a closed space. One of the big question at the moment is not to characterize the signals themselves, which are more or less well known for most, but rather to understand what it implies in terms of cells-cells dynamics. Now, as a matter of fact, we know that the dynamics of such cells can't be clearly pictured with a classical Langevin equation because of the cells-cells interactions. These interactions lead cells to walk "not so randomly", meaning the general pattern is still a random walk, but the signals emitted between cells change their behaviour. Thus, the Langevin equation alone is not sufficient in its classical form to properly describe the dynamics of the system. Still, such systems can be partially described by a relaxation time and a diffusion coefficient the same way we describe an Ornstein-Uhlenbeck process, but we need to add an extra term. This term take the form of the G-function, according to article [2]. Where an Ornstein-Uhlenbeck process returns a Gaussian distribution of the accelerations (against the velocities) with zero mean, our system returned more of a Lorentzian-looking distribution. Mathematically, a Lorentzian distribution is the ratio of two independent normal random variables with mean zero. Physically, it would mean that we have a competition between two phenomena, two interactions, two behaviors. But it would be too much of a guessing to try to more deeply interpret the physical meaning of the Lorentzian distribution returned by our system. For instance, the probability density function of a Lorentzian distribution doesn't have a mean. Indeed, the conclusion of the law of large numbers fails for a Lorentzian distribution. More precisely, such distribution are characterized by having no moment generating functions, meaning we can't use moments to analyse the distribution : we must work directly on the probability density function. This is not convenient in the case of the Lorentzian distribution because both its expected value and its variance are undefined. So, the over way around to analyse the cells-cells dynamics is by studying the G-function, or more generally, the dynamical consequences of the cells-cells interactions. To sum up shortly, both G-function and the probability density function of the Lorentzian are a way to further investigate cells-cells dynamics.

6 Conclusion

In this project we managed on the data analysis side to extract the relaxation time τ and it's standard deviation σ which is the square of the diffusion constant from the experimental data. This has been done considering a Gaussian fit of the distribution which is only approximately true. Knowing this, these parameters have been used in the simulation side of our project to generate theoretical data. It turns out that these data returned graphs that are almost the same as the one of the real experiment: the difference remains in the distributions: we get a Gaussian distribution in the simulation while the real data return a Lorentzian distribution. This particularity in the distribution is due to the cells-cells interactions. These interactions must be taken into account to picture the dynamics of the system: and they seem to take the form of the G-function. This means that we have to add an extra term to the Langevin equation. Moreover, thanks to the simulation approach, we confirmed the relevance of the G-function by testing if the difference of distribution was not simply due to measure errors on the

positions of the cells during the experiment.

On the data analysis side of the project, the next step would have been to extract the interaction strength from the data, that is to say, plotting a G-mapping. The same would have been done on the simulation side in order to compare both the approaches and maybe make some interesting conclusions. Unfortunately, we did not manage to achieve this objective.

7 Annexes

All the codes for the simulation part are available on the following github : [cliquez ici](#).

References

- [1] Dean Fennell Derek Richard John Reynolds John O'Leary Kenneth O'Byrne John D O'Flaherty, Martin Barr. The cancer stem-cell hypothesis: its emerging role in lung cancer biology and its relevance for future therapy. *J Thorac Oncol.* 2012 Dec;7(12):1880-1890, 2012.
- [2] Alexandra Fink Pierre Ronceray Joachim O. Rädler David B. Brückner, Nicolas Arlt and Chase P. Broedersz. Learning the dynamics of cell-cell interactions in confined cell migration. *PNAS* 2021 Vol. 118 No. 7, 2021.
- [3] Daniel T. Gillespie. The mathematics of brownian motion and johnson noise. *American Journal of Physics* 64, 225 (1996), 1996.