

# Covid simulation

## 1 Introduction

### 1.1 The project

The idea behind this project is to simulate the propagation of an epidemic (for example the covid-19) inside a population. Results should show that this propagation is exponential, as it is in real life (figure 1). Besides, we want to illustrate methods that could slow the propagation of the epidemic : social distancing or border closure.

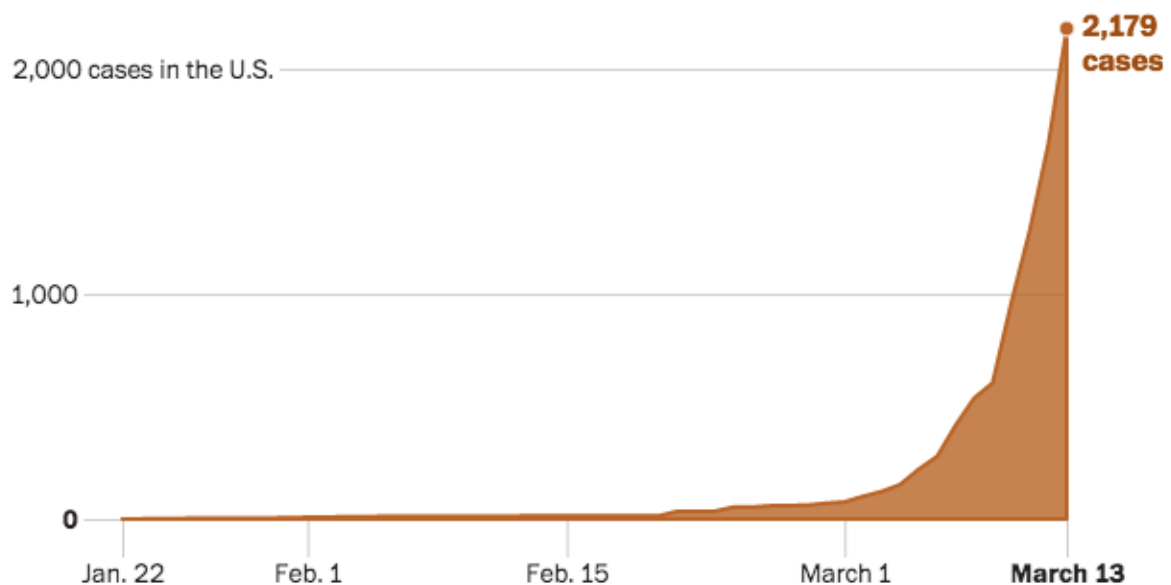


Figure 1: Covid propagation in the USA

### 1.2 Hypothesis

In our model, every person is materialised by a sphere moving at a certain pace (following a normal distribution) in a two-dimensional space. There are three possible state of health: healthy, sick or recovered.

All the experiments are run with 200 people with 5 of them already sick.

The epidemic propagate when sick people encounter healthy ones with a probability of one. A sick person stay in that state for a time  $\tau$  which follow a normal distribution. When a person has recovered, they can't be sick again.

### 1.3 Limitations

Our hypothesis do not permit to simulate all the complexity of an epidemic like the covid. For instance, nobody die from it and recovered people can't be sick again which is not the case with the covid.

Furthermore, the simulation is too small to represent a country (only 200 people) and the movements of people inside it are very simple.

If these hypothesis are clearly not the best to fit the real situation, they could help understand how an epidemic propagate and what methods could slow it down.

## 2 Programming

### 2.1 Monomer model

Monomer are moving around in a box (figure 2). They can neither get out of the box nor be in the same place at the same time. In order to simulate their movements, we need to take into account collisions between one monomer and limits of the box (in this case, the collision is elastic: the monomer rebound in the other direction and conserve his energy) as well as the collisions between two monomers (where the total amount of energy is conserved)

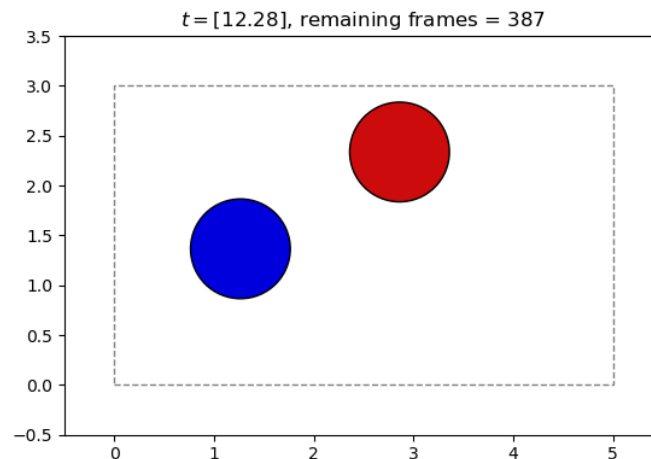


Figure 2: Two monomers in the box

### 2.2 Covid Simulation

To get to the covid simulation from the monomers' one, some changes need to be made. For one, it's needed to take into account in which state of health each monomer is (it will be materialised via coloration: red for sick people, blue for healthy ones and green for the recovered).

A new function is implemented in the code to update the health state of two monomers colliding. The algorithm check if one of them is sick and the other healthy. If this is the case, the healthy one catch the disease with a probability of one.

As people could recovered from the disease, the algorithm check at each event if  $t > \tau$  where  $\tau$  is the time a person needs to recover.

### 2.3 Event-driven loop

The algorithm could be initialized by a file or if it doesn't exist by the data given in the script file. Parameters like the number of frame or the time step can be set.

The algorithm is driven by a loop that check what is the next event and when it will happened. While the time before the next event is shorter that the one before the next frame, the event take place and the time move forward. When the condition is no longer met, we change frames and move the time to the good moment. There are two types of event: collisions between a wall and a sphere and collision between two spheres. This method can only work because the probability of two event taking place at the exact same time is null.

## 3 Running

### 3.1 Propagation

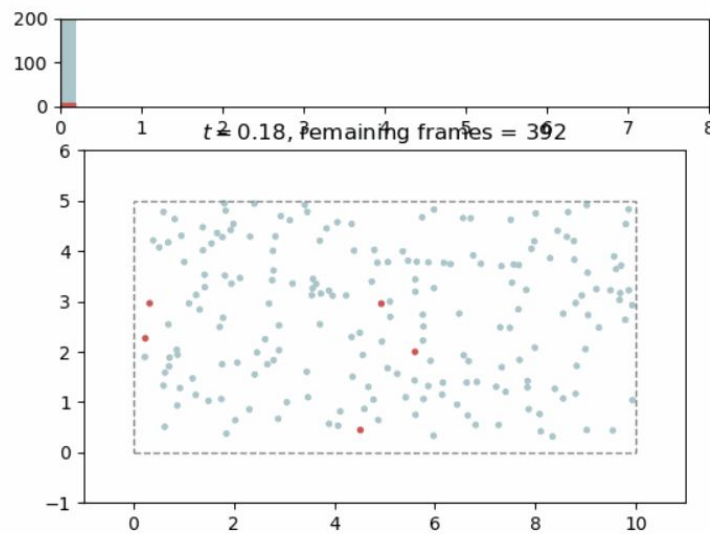


Figure 3: Initial situation

In this first section, let's look at how the epidemic propagate in our box when nothing is done to stop it. The figure 3 shows the initial state.

As show the figure 4, the propagation is exponential.

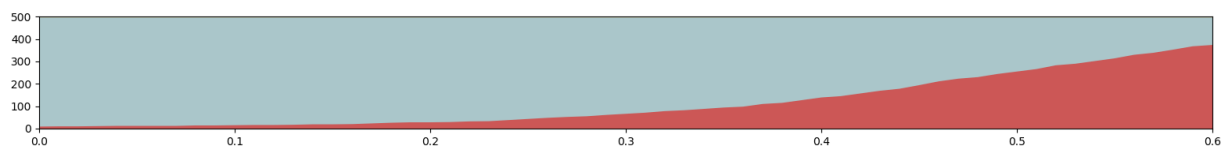


Figure 4: propagation of the disease

Seeing as the box contain a very small amount of sphere, the probability of an encounter between a sick person and an healthy one is quickly diminishing which stop the exponential (figure 5a). It's the herd immunity phenomenon which is quickly obtain with our simulation of only two hundred people.

In this first simulation, our population is going through life quite slowly and some people resist to the disease.

When people move faster (when the mobility is enhanced in the country), results show that more people catch the disease in a shorter amount of time (figure 5b).

In the second simulation, the distribution of velocities of people was centered around a value ten times more important than in the first simulation. Results show that the pic of sickness (which is in our simulation the moment when everyone has caught the disease) was three time before (around the frame 150 instead of 400).

With these figure, it's quite easy to understand how limiting people mobility is crucial to slow the epidemic.

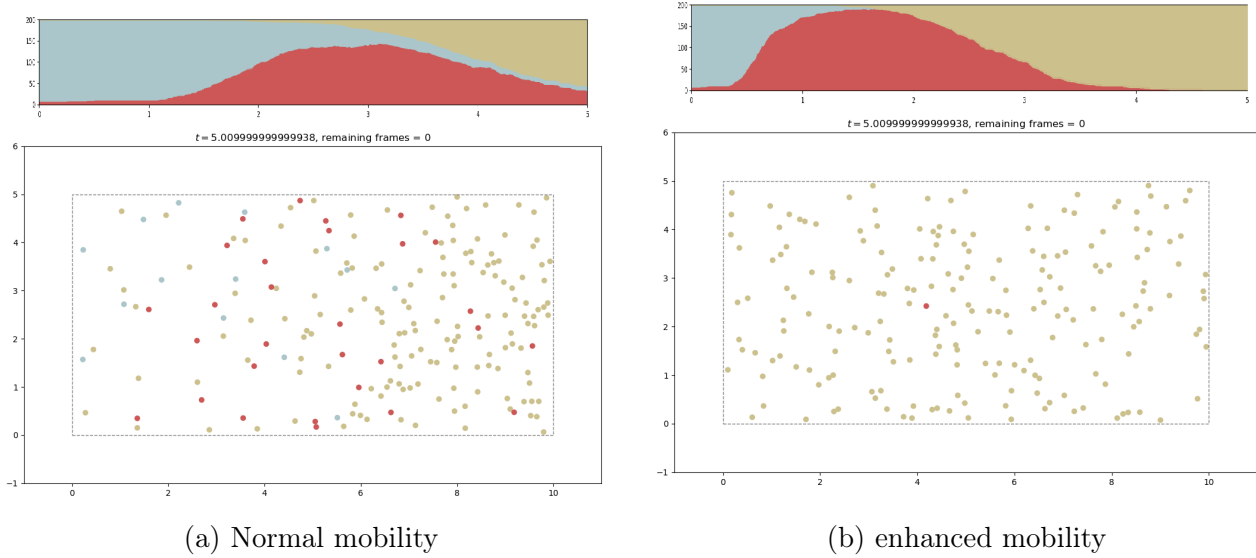


Figure 5: Propagation when no measure are taken

### 3.2 Closing border

The first modification we made to the simulation is to put a wall to illustrate the closed border of a country. Every sick person is at the left of the wall (which could represent China at the beginning of the epidemic for instance) where the epidemic could propagate. After a certain amount of time some people are able to cross the border -as reality show, closing one's border completely is impossible; so our wall became a pierced wall and let some people cross to the other side of the box.

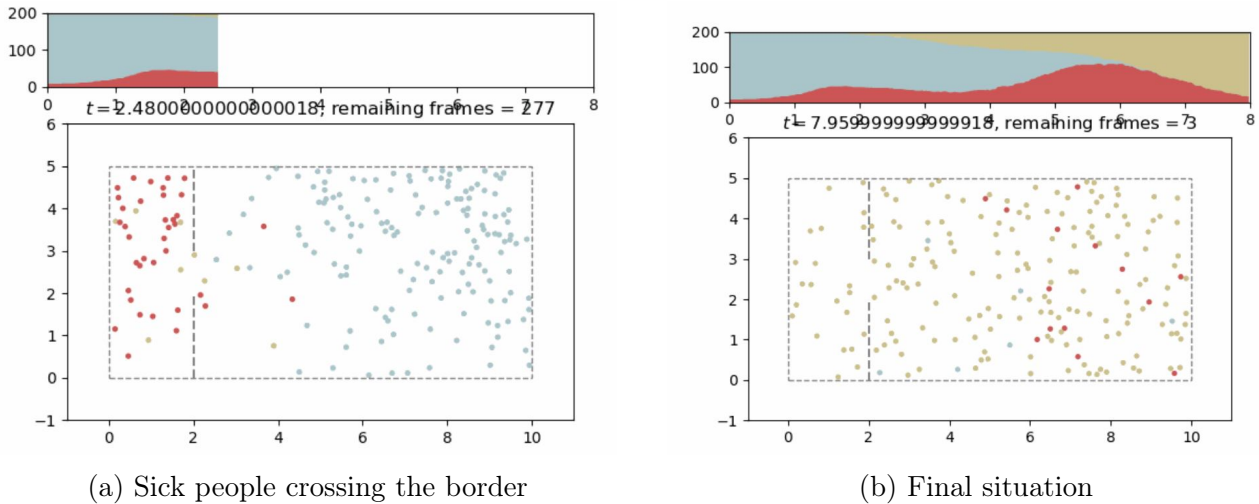


Figure 6: Propagation when borders are closed

Results show that if closing one's border slow the propagation of the disease, it's not as efficient as we would like. Just one sick person crossing into the other part of the box is enough to infect all of the other. It's because our disease is very contagious, one sick person will infect every other she encounter.

### 3.3 Social distancing

One way to try and stop the propagation is to enforce a lockdown. For the same reasons the wall kept an hole in it, it seem impossible to stop every people from going to work or to buy

groceries. So some spheres kept moving in the simulation but 150 out of 200 were locked in their initial position.

With this configuration, the propagation of the disease is tremendously slowed and the number of sick people reach a plateau and stabilise thus containing the epidemic. (figure7)

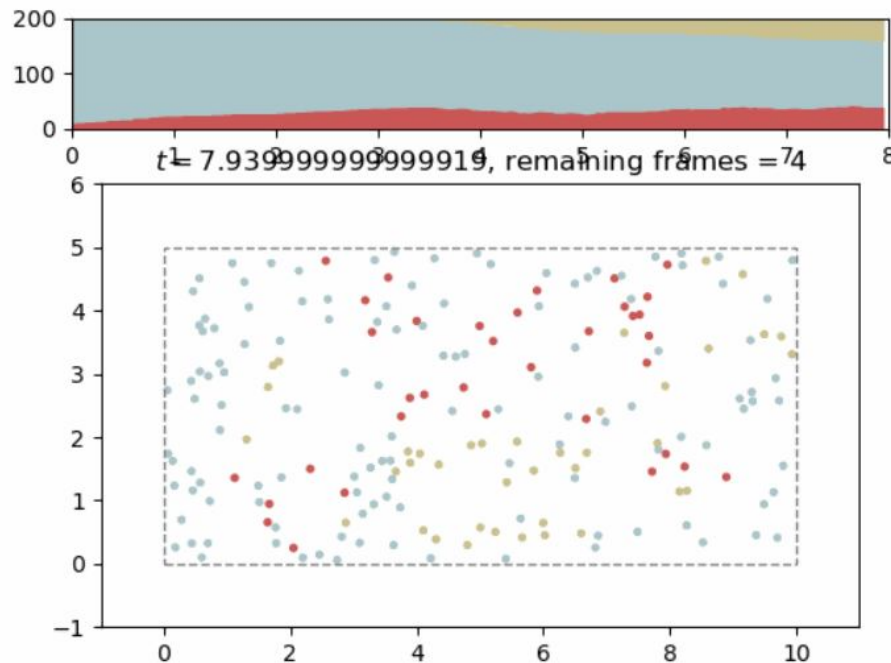


Figure 7: Propagation when lockdown is enforce

## 4 Conclusion

### 4.1 What we learn from this simulation

This simulation shows how disease like covid spread exponentially throughout a population. It also shows that containing people mobility is a good way to slow the propagation down. When people move more slowly in the simulation, some of them don't catch it. One way to really contain the epidemic is to stop all movement from as many people as possible. This simulation thus permit to easily understand why lockdown is a good way to stop the disease from spreading into the entire population. In fact, results show that it's the most efficient way to contain the epidemic.

What is also showing is that trying to close all frontiers is not really efficient: it does slow the propagation but only a little because it take only one sick person crossing the border to infect the other side of the box. If country were able to keep their frontiers entirely closed for a longer amount of time they could have protect themselves better. What our simulation show is that as long as the border are closed, people are safe but after the arrival of one sick person, the epidemic propagate quickly through the country. One must keep in mind that it is only a simulation, in real life this one sick person could be isolated from other as soon a she arrived.

## 4.2 Limitations

As it was said, this simulation is only good to show some consequences of measures taken. But our box could not possibly be as complex and whole as the world and some parameters (like the time to heal, the people direction or their velocities...) do not really take into account the variability of our world.

Also, herd immunity was observed very quickly because of the size of our population. In real life, it's much more difficult to obtain this phenomenon especially since people can catch the covid multiple times.

Thus, this simulation has for only goal to show some massive effects and to help understand why some measures seem reasonable to contain epidemics -like the covid.

## 4.3 Where to find the simulation

Our code is on GitHub: [https://github.com/louisld/covid\\_simulation](https://github.com/louisld/covid_simulation)

To run the simulation, just run the python script *event\_code\_classMonomer.py*. Parameters can be adjusted within the script. Animated animation can be seen in the README file of the repository.