BIRD FLOCKING AND SWARM INTELLIGENCE

Numerical Physics Project (PHY571)

December 2021

Antoine COMBRET–RENOUARD - Armelle CELARIER
with support from Michel FERRERO and Arnaud COUAIRON

ABSTRACT

Our project consists of modeling numerically a flock of birds. Why is it interesting as a *physical* phenomenon ? When birds flock, they seem to be well organized. It seems that they all move and behave as a single body - which is surprising given the large sizes of the flock. Indeed, it is their decentralized decisions, based on knowledge of their local environment, that drive their individual movements. But once aggregated, these interactions create a global order. This phenomenon is an example of what happens in many biological or physical systems, is usually referred to as "swarm intelligence". Simulations on such many-body systems can be very useful in numerous applications, such as optimizing the location of transmission infrastructure for wireless communication networks. In this paper we analyse how the order appear in such system and exhibit the relevant parameter to characterize it. The simulations from [2] are reproduced with a decent level of fidelity, and some algorithmic improvements are considered. In particular, a version of the code using a "Particle In Cell" method from plasma physics is tested.

CONTENTS

1
A BASIC IMPLEMENTATION : VICSEK MODEL (1995)

The system we have considered is a population of agents (the birds) that move randomly in a periodic 2D space and that behave in reaction to their close environment. We want to characterize the effect of simple rules of interaction between birds on the global movement. Doing so, we have exhibited the emergence of self-ordered motion and analysed it.

1.1 Modelling

In order to describe the collective flight of birds, we first used a simplistic model where birds have a constant absolute speed and a varying orientation (angle). This model was first described by Tamas Vicsek [2] and is named after him.

Modelisation of space and birds
The space is a square (2 dimensions) with periodic boundary conditions and is considered continuous. Time is discretised. The birds are assumed to have the same absolute speed v. At each time step, the orientation of their velocity can change, and their position is updated accordingly to their new velocity.

Interaction between birds and decision model
We now have to model how the birds are deciding where they go. They only have to choose the direction in which they want to move (as their speed is fixed). The model predicts an evolution following two independent factors. First, birds orient themselves according to the flight of the other birds they can perceive. This is modeled by averaging the velocity orientations of birds that are within an interaction radius R around the bird. On the other hand, birds orient themselves according to their own willing, which is modeled here by adding a random orientation to the local average orientation. This random angle is taken in an interval $[-\eta/2, +\eta/2]$ where η is the "noise".

Review of the time and length parameters

- The unit of length is the interaction radius, and therefore we take R=1.
- The size of the box is thus the relevant length paraneter, and is used to characterise the scope of the interaction between the birds. It is typically between 1 and 50.
- The time is discretised. This discretisation defines the unit of time.
- Space is continuous.

1.2 Architecture of the algorithm

The algorithm proceed in two steps, repeated at each time step :

- The birds "interact" with their neighbors; we calculate for each bird the average orientation of the birds around him.
- The birds update their orientation, by adding a random angle to the local average angle they "see".

- The birds update their position, according to their orientation and speed.

In order to be consistent with the historical implementation of Vicsek, instead of taking into account the average angle, we considered :

$$\langle\theta\rangle = \text{atan}\left(\frac{\langle\cos(\theta)\rangle}{\langle\sin(\theta)\rangle}\right)$$

1.3 Discussing the model

The algorithm that we have implemented is simple, but allows to highlight a very interesting phenomenon with multiple physical applications, as mentioned in the introduction. However, in order to correspond with more acuity to the physical reality and to take into account more phenomena, some adaptations can be considered. We will illustrate this with the case of the flight of birds, which interested us in the first place and on which researchers have already worked, and in particular [1].

There are some things that our model does not take into account :

- Birds cannot interpenetrate. Moreover, they will generally intent to stay at a minimum distance from each others. This applies to other physical entities (short-range interactions).
- Likewise, the birds are confronted to obstacles, which they absolutely want to avoid. This has been the object of development in [1].
- Instead of avoiding obstacles or other birds, birds might also want to go in a particular direction. This can affect all birds - identically or not - as special kind of aligning "field". We can also imagine that some "leading" birds go more or less in a precise direction, drawing behind them the others. This could be the case for a migration process.

Some other points that our model decide can be discussed :

Vision of the birds In our model, the bird can "see" all the birds around him in all directions, and has perfect information about their orientation. In reality, this is not possible. For example, the vision will be mostly valid for the birds in front of him, and the depth of his field of vision will be greater in front of him than on the side. Moreover, if one of his neighbors hides another, he will have only imperfect information. The modeling is complex, and our particularly simplistic model can surely be improved.

Self-interaction In the model, a bird considers himself as a normal "neighbor". This is useful, because when he has no other neighbor, he continues to follow his path with only a random noise. This means also that when he is surrounded by lot's of neighbors, he tends to neglect what was his previous movement. On the contrary, when the bird is nearly alone he will keep inertia (which costs less energy), the random orientation's change being apart. This seems "natural" to us, but may be a subject of discussion.

Timing of the decision In the model we used, all the birds take their decision at the same time and then move in a rectilinear way. This is linked to the fact that the time is discretised, which is not the case in reality. First, the movement is continuous and second, the birds don't take their decisions all together. We can adapt the algorithm to solve this questions. First, we need to have smaller time steps (as it is the unit of time, we thus need to adapt the speed). Second, we can compare different ways to prioritize decisions and movements. The way to do that was not described in the article from Vicsek. Therefore, we decided to use two methods (decisions taken one after the other, or decisions taken simultaneously), whose results being in fact similar.

2 RESULTS - CONFIRMATION OF THE CONCLUSION OF VICSEK

2.1 Qualitative results

Our strategy for assessing our algorithm was to try to re-obtain the results shown in [2]. The first step was to reproduce the first figure in which the physical system was displayed at different stages of evolution and with different parameters.
Our own results are displayed in figure 1.

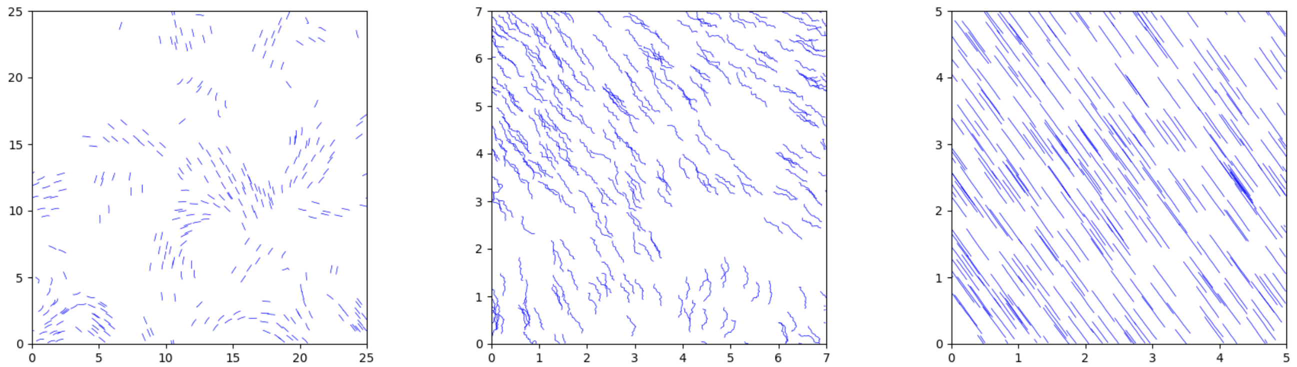

Figure 1: In this figure, we reproduced the test run by Vicsek [2]. Thus, the velocities of the particles are displayed for varying values of the density and the noise. The number of particles is N = 300 and the velocity is 0.03 in each case. (left) L = 25, noise : $\eta = 0.1$ (center) Same situation as in a., after a longer time. (right) L = 7, noise : $\eta = 2.0$ d. L = 5, noise : $\eta = 0.1$

In the first situation (left), multiple and independent flocks are forming, with a global movement that could be described as viscous : the particle move along very smooth curves in a coherent manner. In the second situation their path is more harsh but also less flocked (higher density). In the last situation the density is high enough to form a single flock.

We have not displayed arrows to indicate each particle's direction, and this lead to underestimate the density of particle 'flock' in our simulation. Nevertheless we believe that the behavior shown are quite the same, and that further comparison is possible.

2.2 Quantitative results

A more substantial way to assess our code was to compare quantitative results. Since we had made the choice to reproduce the exact system that was described in [2], we expected to obtain the same values for any synthetic quantity.

An interesting quantity to study in our system is the average velocity of the particles, that can be written as :

$$v_a = \frac{1}{Nv}\left|\sum_{\text{part.}} \vec{v}_i\right| = \frac{1}{N}\left[\left(\sum_{\text{part.}}\cos\theta\right)^2 + \left(\sum_{\text{part.}}\sin\theta\right)^2\right]$$

In fact, this step of our project has helped us to solve a number of issues that went undetected during our first trials. They were mostly differences between our code and what was specified in the article : wrong limit conditions, updating particles' orientations too early... etc. These problems were easily fixed and improved fitness of our results in comparisons to the ones in the article.
Our attempt to reproduce the quantitative results of [2] is shown in figure 2.

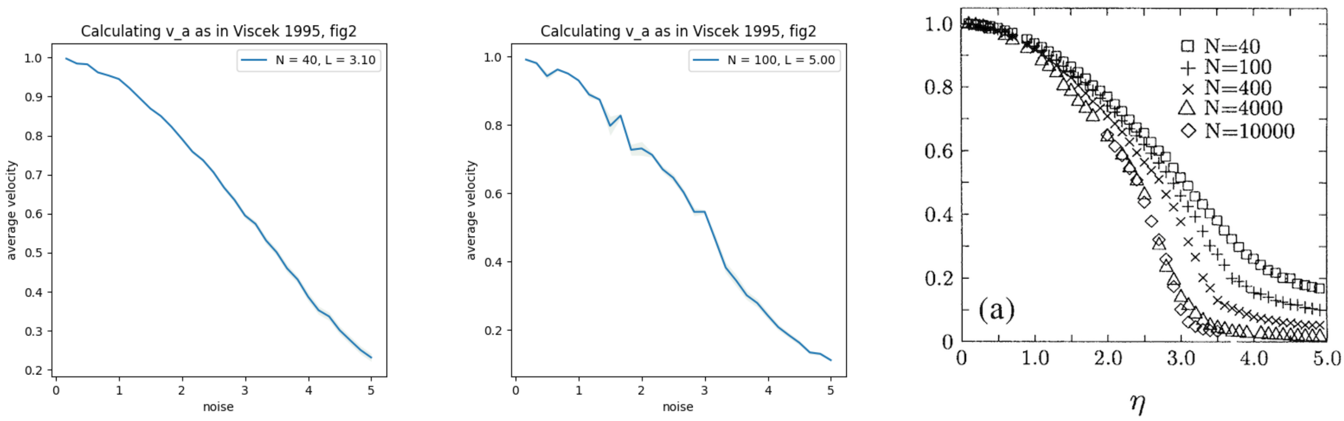

Figure 2: (left & center) Our curves obtained for two different configuration (see legend). (right) The original curves in [2]. The end of the curve do not flatten exactly in the same way. Also, the numerical values are not the same (see fig. 4).

There are still differences between our results and the ones in the article, and a few problems have necessitated us more time to investigate. In particular, the right way to handle fluctuations and initial conditions was very unclear at first.

Fluctuations We are investigating a dynamical system in situations where hopefully a phase transition must happen. Consequently we expect to observe a lot of fluctuations in its dynamics. These fluctuations must be considered when calculating the average velocity (an **order parameter**) : if we average on a too short period of time, the value might be irrelevant.
That is why we attempted to display the time-evolution of the average speed for our system. We can see on figure 3 the results for several configurations at $N = 100$.

It is clear that the fluctuations near the transition $(1,5 < \eta < 2,5)$ are not negligible. This means that for a given initial state the system might not reach all parts of the phase space, and the average speed might be biased for these values.

Averaging on initial conditions A possible way to investigate this bias is to test the system on many initial conditions and to study how the average speed behave. If the initial state is chosen randomly, it should sample a significant part of the phase space and allow a better estimate of the average speed.
Below are the results for two situations (at $N = 40$ and $N = 100$) for 30 trials (fig. 4).

We observe that the time-averaged values obtained in figure 2 are in fact valid, and that adding a initial-condition average does change the curve.

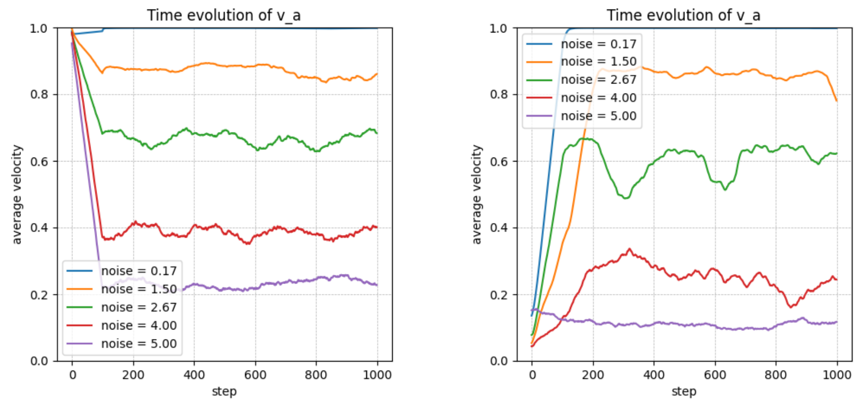

Figure 3: Time fluctuations (flattened for readability) for several situations. Fluctuations are maximal near the transition described by [2]

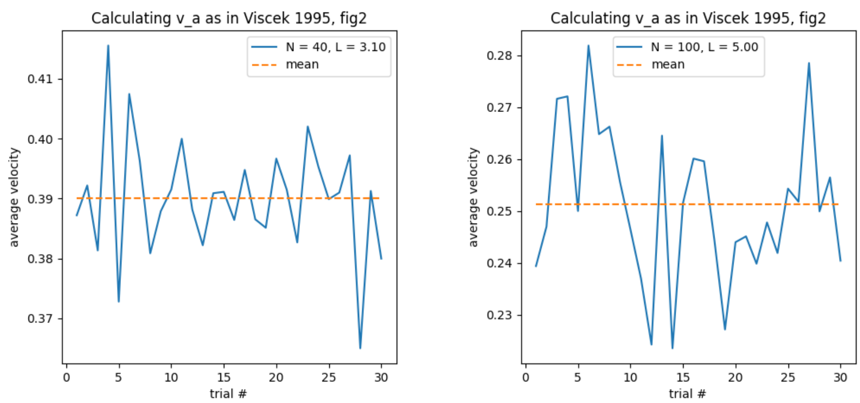

Figure 4: In [2], for $\eta = 4$, we have $v_a = 0,25$ for $N = 40$ and $v_a = 0,17$ for $N = 100$. Here, with different initial conditions we measure on average $v_a = 0,39$ and $v_a = 0,25$ respectively, with and typical deviation of $0,02$.

This last curve (fig. 4 confirms that our results for the velocity as a function of noise are different than the ones in [2]. As of today we cannot explain properly these differences.

3
ACCELERATION OF THE ALGORITHM

3.1 Search for an quicker method

Our method was efficient but not fast. Indeed the complexity of the algorithm is in $O(N^2)$, N being the number of particles in the box. The loop that leads to this complexity is that of searching for the neighbors. Indeed, for each particle, we look if each of the other particles is within the interaction distance or not. That's why we try and search ways to avoid the use of this loop.
A first idea is to search the neighbors of a particle only in the "close neighborhood". This comes from the fact that a very distant particle (several interaction radii) will not be a neighbor for several time steps. This works only because the speed is controlled (fixed in our simulation) and we therefore know the maximum distance that a particle can travel in a given time.

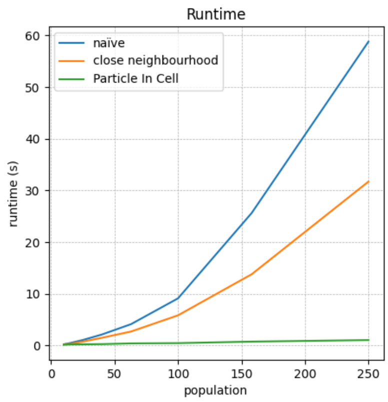

Figure 5: Runtime for the three methods used for a run with 100 particles and 100 timesteps

We considered a second method to speed up the algorithm. The idea is to model the interaction between the particles through a field. For numerical modeling, this field is carried by the nodes of a grid. This is why the method is called "PIC" : Particle In Cell. It is this method that we have chosen to implement, because unlike the others the interaction between particles is not done via taking into account particle-to-particle interactions, but rather particle to grid interactions. The particles act differently according to their proximity to the different nodes of the network. We will have to choose a mesh and a method of field/particle interaction which are consistent with the interaction we wish to model; which is not obvious at first sight !

3.2 Implementation of a PIC (Particle In Cell) method

We will first present the different steps of the algorithm. At each time step, the following steps are performed :

- The particles, via the orientations of their speeds, create a field. Thus, for each particle, we modify the parameters of the nodes of the network that surrounds it (at the four corners of the cell in which the particle is located).
- Then, we evaluate for each particle the influence that the field has on it: we calculate the average orientation of the velocities around the particle from the parameters contained in the four nodes at the corners of the box in which the particle is located.
- A random component is added to the orientation of the velocities of the particles obtained.
- The position of each particle is then updated according to the velocity we have just calculated (absolute position and cell in which it is located).

We have described the main steps of the algorithm. We now need to define how the particles interact with the field. This interaction is done in two steps: the particles create the field, then we calculate the orientation of the average velocity around each particle according to the field. For this, we will use the figure 6.

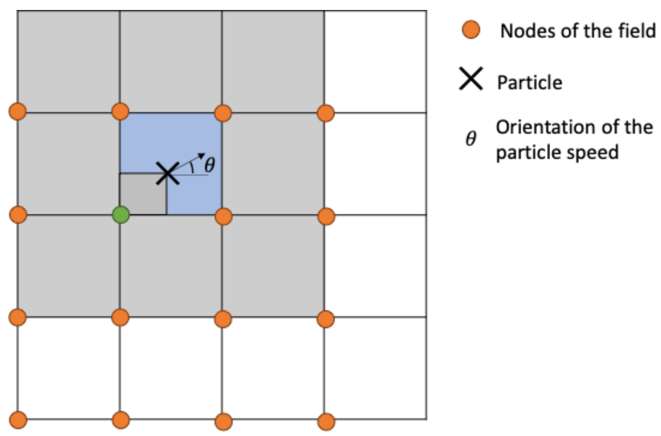

Figure 6: Principle of the Particle In Cell (PIC) algorithm

The particle (indicated by a cross) is located in the cell marked by the green node at the bottom left. It influences this node in proportion to the blue area. Thus, the closer the particle is to the node, the more it influences the parameters of the field carried by this node. The particle also influences the three other nodes that frame the cell in which it is located, in proportion to the corresponding areas.
Let us note theta, the angle that the particle's velocity makes with the horizontal. The parameters contained in each node are the average of the $cos(\theta)$ and $sin(\theta)$ of each particle, weighted by the proximity factor of the particle with the node (corresponding to the blue area).

We decided to use this method for several reasons.

Size of the grid The main objective of this method is to speed up the algorithm, so by making the grid large enough, the interactions between the particles and the field will require a minimum of computation. Here, the particle only interacts with four nodes of the grid.

Influence of a particle As we have seen, the fact that the grid is so large has the effect that the particle interacts with the particle in the 9 boxes around it. We wanted the particle to feel more clearly the effect of a particle close to it than that of a particle further away. However, this is not feasible only through the field, as we designed it. This is the reason why we have weighted by an area.

Form of the cell The shape of the cells that make up our field is a square. This impacts the interactions between particles. Indeed, as illustrated in the figure 7, a closer bird may have less influence than a more distant bird. Thus, in the figure, the swallow (center) perceives the influence of the gull (top left) via the node of the network between them, indicated in green. On the other hand, the robin (right) is at the limit of the interaction zone and being on the left edge of the cell, the area that weights its influence on the right nodes of its cell is null. Thus, it does not influence the swallow, although it is closer to it than the gull.

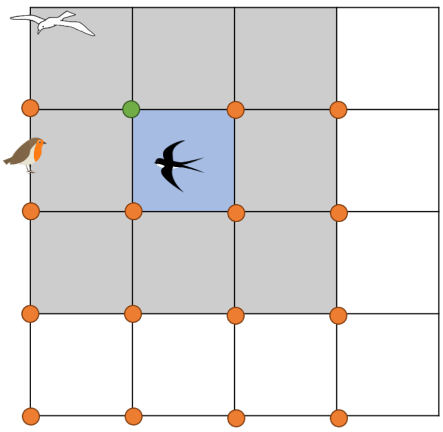

Figure 7: Influence of the shape of the grid on the extent of interactions between birds.

Note that the particles interacting with the one indicated by a cross are therefore those in the same cell or in the neighboring ones, i.e. in the grey area (figure 6).
Concerning the periodic boundary conditions, some more details can be given :

- The nodes completely up, respectively right are the same as the ones completely at the bottom, respectively left (that's why they are not indicated in the figure 6) ;
- A particle can be found at the border of a cell of the grid supporting the field. In this case, it is considered to belong to the right-hand cell. In the case where it is at the right edge of the complete grid, it belongs to the first cell completely to the left. The same applies to up/down.

3.3 Results of the PIC method

Runtime gain

The results in terms of runtime are shown in figure 5.

The PIC method that we have used leads to different results from the first method. This is to be expected since it does not only accelerate the algorithm but also modifies the model, at the level of the interaction of the particles with each other.

Qualitative observation of the transition phase

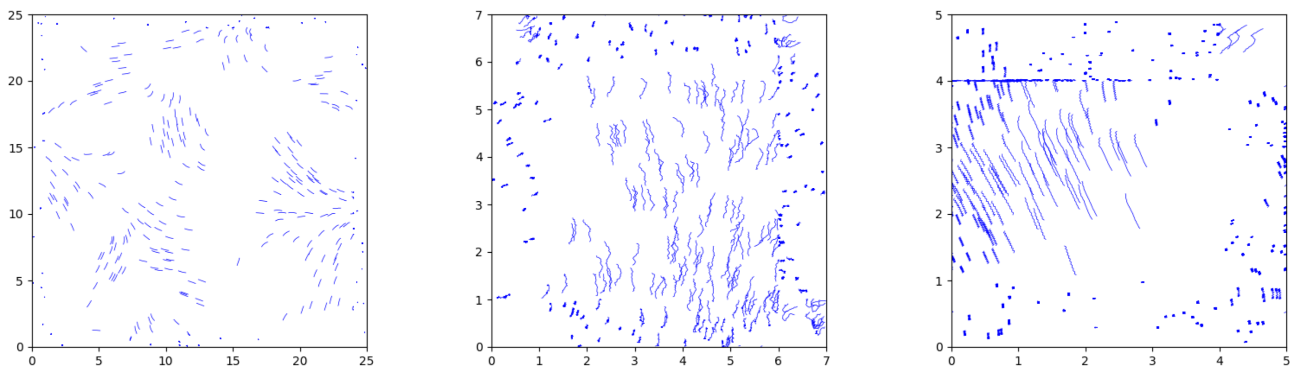

Figure 8: In this figure, we reproduced the test run by Vicsek [2]. Thus, the velocities of the particles are displayed for varying values of the density and the noise. The number of particles is N = 300 and the velocity is 0.03 in each case. (left) L = 25, noise : $\eta = 0.1$ (center) Same situation as in a., after a longer time. (right) L = 7, noise : $\eta = 2.0$ d. L = 5, noise : $\eta = 0.1$

As long as the densities are not too high, we observe results that are more or less identical to those of the first method: the birds gather in groups that evolve in a highly correlated manner.
However, there are some anomalies: some birds seem to stay in fixed positions. As the density is increased, this situation worsens. The proportion of almost fixed birds increases, as can be seen in the figure 8 where the density is increasing. We found in our tests that the birds were not in fact fixed, but oscillated around their position. This is illustrated in figure 9. We notice in particular that these oscillating equilibrium are observed for particles in interaction with others, as for isolated particles (the one circled in red).

Quantitative observation of the phase transition

Figure 10 illustrates the overall consistency of the birds. It can be seen that if the tendency to decrease with increasing noise is well observed, the shape of the curve as well the numerical values are different from our previous results. This is probably due to the numerous "oscillating" birds, a behavior that was not observed before and that influences the average speed. The average velocity is lower than before, from which we can deduce that the "oscillating" birds have a global effect of motion decoherence. These movements do not correspond to any physical

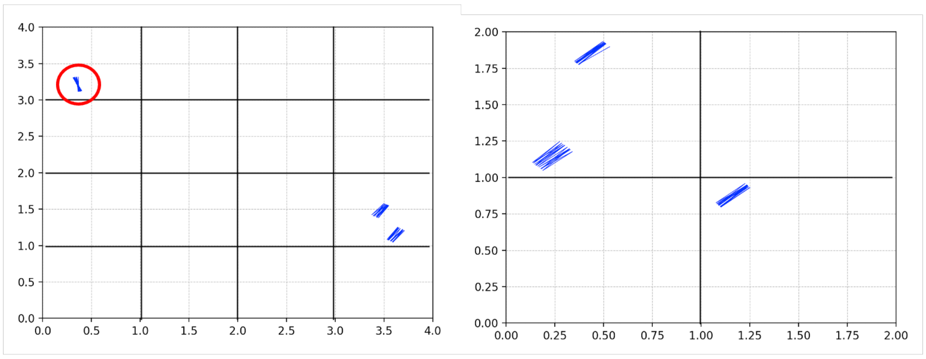

Figure 9: We run the PIC simulation for 3 particles, with speed 0.2, noise $\eta = 0.2$ and in boxes of size a.L=4 and b.L=2. We let the system evolve for some time and took the picture when it was definitively stabilized in this oscillating configuration (figure taken after around 300 time steps).

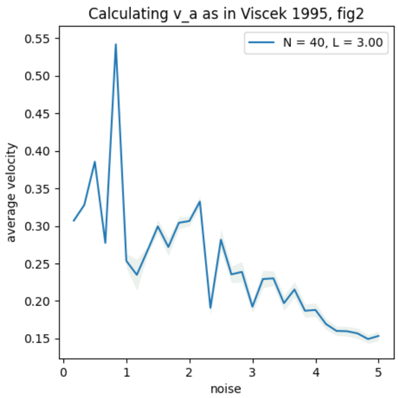

Figure 10: As in Vicsek [2], this figure plot the absolute value of the average velocity versus the noise η in cells of various sizes for a fixed density : L=3, N=40 birds.

reality, and it is therefore difficult to interpret this curve further.

CONCLUSION AND PERSPECTIVES

We have implemented the PIC method with a mesh parameter of the order of the interaction radius. However, we could make a fine grid by simply adding the $cos(\theta)$ (without distance/area factor) in the nodes contained in the disk of radius 1 (interaction radius) around the particle. This is illustrated on the figure 11. The central swallow interacts with all the green nodes, at the corners of the shaded boxes. This would be much closer to the results of the first implementation of the problem, since the particles taken into account and the way to do it are the same, except for the errors due to the fact that the disk in which the neighbors are taken is "pixelated".

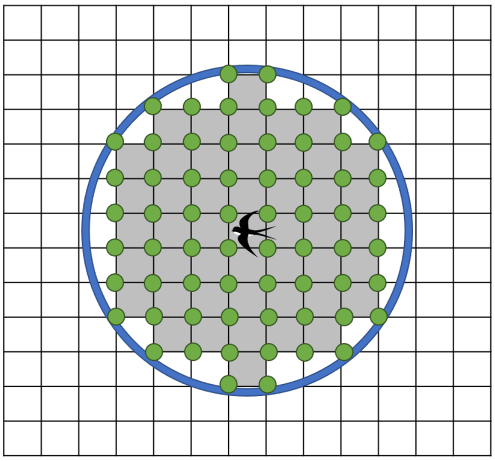

Figure 11: Schematic of the PIC method, modeling an interaction in a disk of given radius. In grey, the nodes on which the swallow has an influence.

In the PIC method, it is possible to adjust the weighting in the field/particle interaction terms, in order to better fit a given physical reality. For example, one could imagine that each node of the network is given the same weight in the node/bird interaction when calculating the average perceived angle. Physically, this would mean that a cohort of birds on the right would have as much impact as a single bird on the left for the central bird. More subtly, one could define a parameter that would allow to weight the influence of a group and a single individual. This would allow to find a balance between perceiving individuals and perceiving areas of space.

We could also evaluate the impact of changing the metric or the system of taking into account the distances. For example, instead of weighting by area, one could weight by the root of the area (distance measure).

Finally, it seems to us very useful to have a field able to influence the birds locally. It could be used to simulate a wind, or obstacles, a particular intention, etc.; provided that the field/bird interaction formulas are well chosen.

REFERENCES

[1] Craig W. Reynolds. Flocks, herds, and schools : A distributed behavioral model.

[2] Eshel Ben-Jacob Inon Cohen Tamas Vicsek, Andras Czirok and Ofer Shochet. Novel type of phase transition in a system of self-driven particles.