Noah Mifsud
Modeling Natural Phenomena
Bacteria Multi-Agent Model

Introduction

In this week's lab we were tasked with implement and experimenting on a multi agent model for bacteria in a petri dish. Our goal was to write a code which would move 1μm wide bacteria at a speed of 20 μm/s through the dish with cyclic boundary conditions over 0.2s timesteps. In this dish we were to simulate two different versions of a food-density field. The bacteria, being agents with some knowledge of the world around them, would measure the change in the food density at each timestep and decide whether to rotate or maintain a straight path.

The first food density field is a gradient with the highest density at the center and lowest at the edges calculated using this formula:

$$p_a = \frac{1}{1 + \mathrm{norm}(c\text{-}x)}$$

Where c is the center and x the position of the bacteria. Thus P_a=1/2 when the bacteria is at the center and approaches zero when the bacteria is far away.

The second is far more rudimentary, it states: 'if *norm(c-x)<=15* the value of P_b is one, otherwise it is zero'.

Using one of these methods for calculating the value of P at each timestep the bacteria would decide to do one of two possible things. If *P(t)>=P(t-1),* meaning the bacteria was getting closer to the food source, it would have a 90% chance of continuing its path and a 10% chance randomly picking a new direction. If *P(t)<=P(t-1)* the bacteria would have only a 50% chance of continuing. In this way each bacterium was modeled as an agent trying to get as close as possible to the food source with the ability to swim forward or stop and rotate.

This lab report will first outline my implementation of the multi-agent bacteria model. It will then analyze the behavior of the bacteria for both food density field functions and examine how varying the bacteria's speed affects their behavior. This report will describe the implementation and behavior of another bacteria model, featuring bacteria who measure the center of mass of those around them and move towards it. Finally, this report will sketch possible methods for parallelizing the previous models.

My implementation

To implement the above outlined model, I used a continuous space.[1] Each bacterium was represented by a body endowed with position coordinates and a velocity vector. To model their motion, I ensured the velocity vector always has a length of ‘v’ for some chosen input velocity by normalizing the vector and multiplying by v each time it was changed. The bacteria moved in discrete time, with position and velocity vectors updated over 0.2 second intervals. This formulation bore many similarities to the galaxy model from the previous lab, only this time bodies lacked inherent mass.

Calculating the food density fields in this model was very straightforward as norms are easily found so the equations presented in the introduction could be represented exactly in the system.

To give the bodies agency I allowed them to calculate their previous position in space. By subtracting the timestep multiplied by their velocity vector from their current position they could determine their location at the previous timestep and thus determine the past the value of P, the food density. They would then compare this to their current P and the program would determine whether each body was going to continue in the same direction or rotate using the corresponding probability. The probability calculation was done using a weighted random choice between two options, with the weights determined by the relationship between the past and present food density. Their rotation was represented as a random choice of new velocity vector which could point in any direction[2].

The agent’s positions and velocities were updated in a non-simultaneous fashion, however, since they do not interact with each other this did not affect the results.

The petri dish was modeled with cyclic boundary conditions. At every timestep, after the agent’s velocity had been determined but before their position was updated, the system calculated whether their new velocity would move them outside the dish in the current timestep. If so, it would adjust their position accordingly to place them in the dish on the opposite side, then update their position.

On the first timestep all agents were placed inside a small circle in the bottom left of the petri dish and each given a random velocity direction.

Results

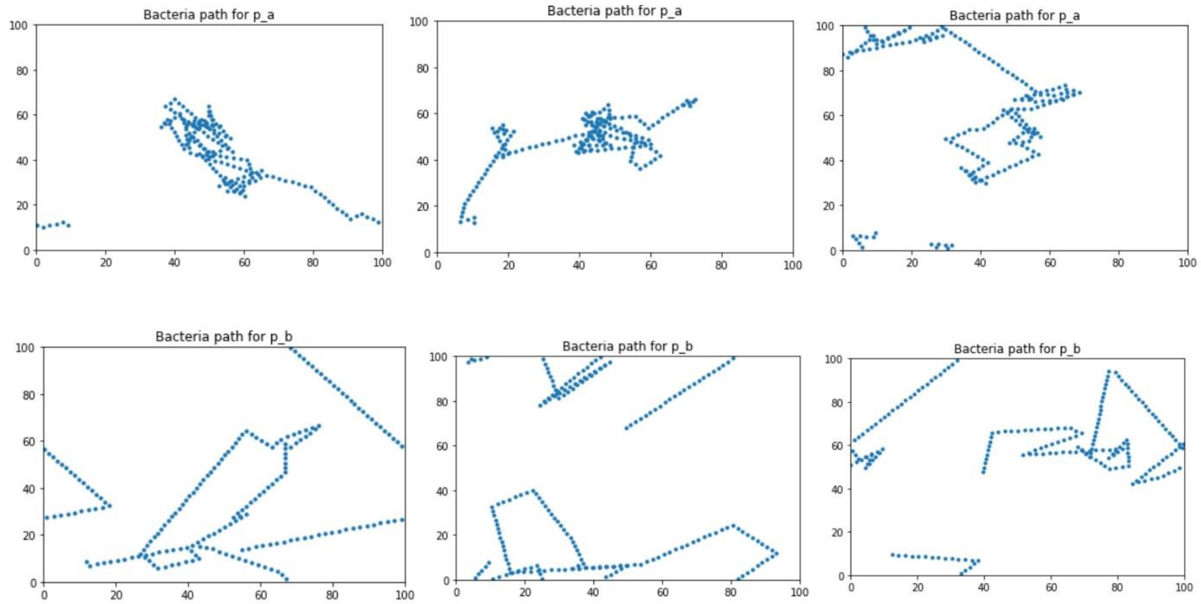
Before modeling the large scale behavior of the bacteria I tested the program on single agents for both the P_a and P_b methods of calculating the food density gradient. I used the starting

[1] It was not until I had completed 95% of the programing that I realized perhaps this was meant to me modeled using rectangular cells with the bacteria moving only in the cardinal directions. Alas, I now have an overcomplicated method

[2] This was achieved by calculating a random a and random b between -100 and 100, then normalizing the resulting (a,b) vector, and multiplying it by the input velocity v.

condition of v=20 μm/s, with the timestep at 0.2s. Figure 1 shows three example paths for the bacteria in each case.

Figure 1



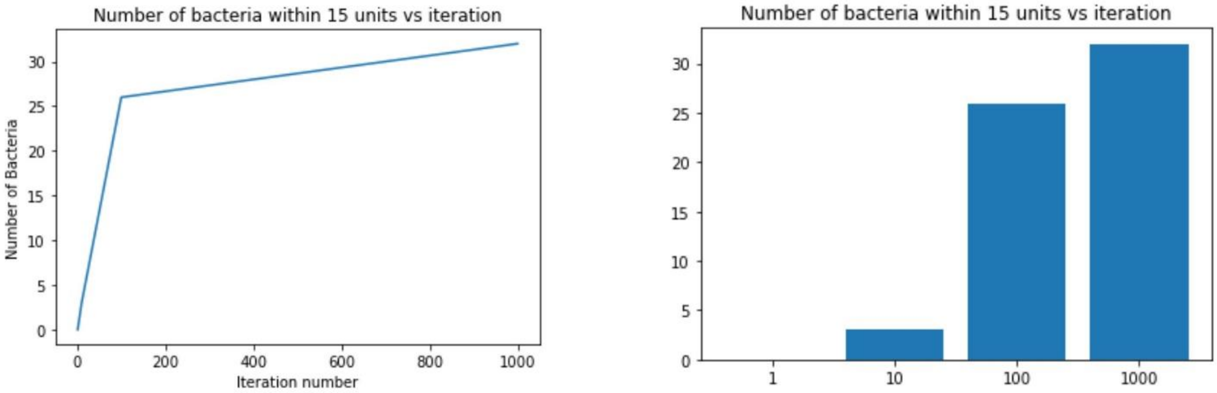
In each case the bacteria starts on the bottom left hand side and moves from there.

We observe that in all three P_a cases the bacteria are drawn towards, and eventually find, the center, despite starting with three completely different velocities. We also observe that in this formulation, the bacteria have no way of stopping or slowing down once they have reached the center of the food gradient. Coupled with the randomness of the velocity direction each time they rotate, this results in an orbit-like behavior where they wander around the center, sometimes straying quite far away before returning. In the middle and right most cases the bacteria achieve very close proximity to the food source before wandering 30 or more μm away. This behavior will become important later.

The P_b cases are not as successful. Because of the on/off nature of the food gradient the bacteria cannot effectively navigate. Outside the central circle of radius 15 μm they are completely lost, walking and rotating totally randomly. In the middle case of figure 1 the bacteria never even reaches the center. However, even when they do stumble across the center, they have a constant 90% probability of walking straight until they leave again, as happens to both the left and right cases. This creates a sort of paradox where, outside the food they change direction often, making it take longer to move long distances, while inside the food circle they rarely change direction, making them walk out of it as fast as possible! The P_b case effectively keeps the bacteria out of the food source.

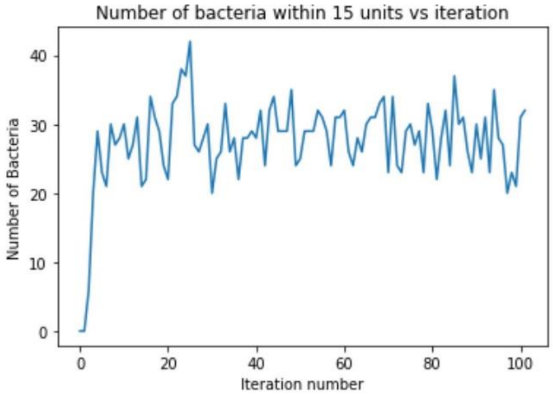
With this understanding of the behavior, we can now model a system with 100 bacteria to observe large-scale effects. To see how effective the bacteria are at reaching the center for each food source gradient, I checked at the 1,10,100 and 1000th timestep how many bacteria were within 15μm of the center. Figure 2 shows an example of the results for the P_a method.

Figure 2



Looking just as the graphs in figure 2, we see that only a few bacteria make it to the food source within the first ten timesteps. After 100 timesteps more than 25 have made it, and by 1000 timesteps over 30, or 1/3 of the bacteria have made it. However, this does not tell the whole story. As we determined above, a bacterium can come very close to the center before wandering away again. This means the bacteria who made it in the first 10 timesteps, could have left and come many times, and may not have been there at the 100 or 1000th timestep. Thus, the 100 and 1000th timesteps, which are very similar in number of bacteria, represent more of an average number of bacteria within 15 µm for these parameters over long durations. To further illustrate this, figure 3 shows a graph the number of bacteria within 15 µm every tenth timestep for 100 timesteps.

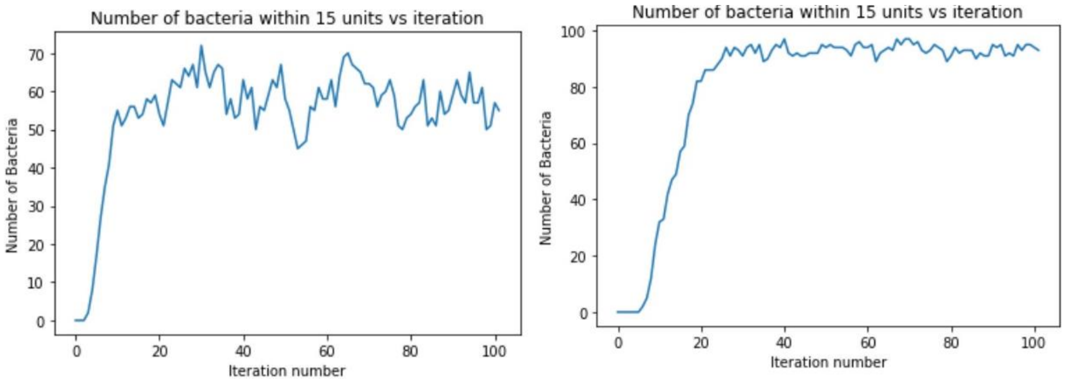
Figure 3



Here we see that after the bacteria have left their starting place and found the center (the initial sharp increase) , they all begin to wander/orbit around it, resulting in a number of bacteria within 15 µm of the center which behaves like a random number generator oscillating around a central average of about 27. Considering the semi random nature of their behavior, this result is logical. Thus, for our starting parameters, we see an emergent behavior which results in an average of 27 bacteria within 15 µm of the food source. It is then interesting to ask if this can be improved.

For more timesteps we will get the same result. Since the bacteria don't interact, adding more won't affect the outcome either.[3] To encourage them to stay closer to the center, I reduced their speed by half to 10μm/s, then by half again. The results are shown in figure 4:

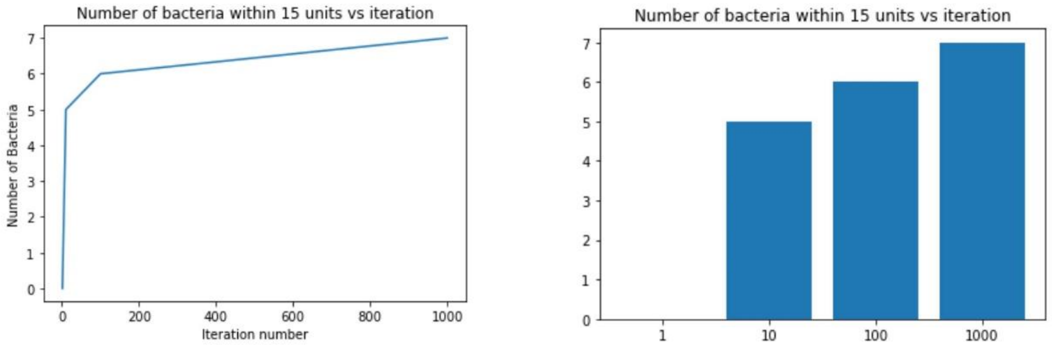
Figure 4



The left side bacteria had velocity ½ of the original group, while the right had ¼. As we can see, this reduction in velocity drastically increases the bacteria's efficiency at staying close to the center. The reduction by half of the speed doubles the average number of bacteria to almost 60, and the reducing in half again brings the average to almost 90. Based on our observations of the bacteria's behavior this follows, since is becomes less likely that the bacteria will wander far from the center once they have reached it as their speed decreases. Also logical is the fact that we see it taking longer of the bacteria to reach the center, on the right side it takes them twice as much time as the left.

We can also observe the behavior of the bacteria in the P_b gradient. Figure 5 shows the same data as was collected for figure 2:

Figure 5



For the P_b food gradient we see that after the first ten timesteps, some bacteria have managed to stumble into the center, but the number of bacteria in the center does not increase singificantly over the rest of the timesteps. In fact, it is interesting to note that a circle of radius 15μm makes up around 7.1% of the area of the 100μm*100μm petri dish, meaning if you spread the bacteria

[3] I tested this using 300 bacteria and the result matched exactly the proportions of 100.

out randomly over the whole dish, there would, on average be more of them in the center circle than there where using this food gradient method after 100 and 1000 timesteps. With this food gradient our bacteria are no more effective at finding the center than a random distribution.

Having analyzed the behavior of the bacteria for these to methods of motion, we now turn to different model for bacteria movement.

Center of Mass experiment

We now turn our attention to a different model. In this case, each bacterium calculates the 'center of mass' of the surrounding bacteria which are within 10μm of its position. It then creates a normalized vector *d* which points towards this center of mass and updates its own velocity using this equation

$$v(t+1) = v(t) + v*a*t*d$$

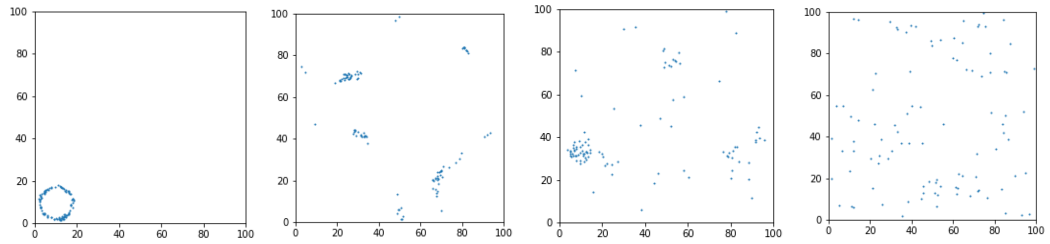
where a is a constant (in this case equal to 0.1) v is the input velocity and t is the timestep. In this way the bacteria will tend towards the center of mass of its neighbors.

To implement this, I used two for-loops, one to loop over all the bacteria, the second to loop over all the bacteria within 10μm of the target from the first loop. I calculated the normal vector to the center of mass of the resulting bodies and calculated a correction using the above equation.

Since all the bodies are interacting with each other, the velocities and positions in this model needed to be updated simultaneously. To achieve this, I saved all the initial velocities of the particles as temporary arrays, then calculated the corresponding correction, and saved the sum of those two in an ordered list. Thus, once the calculation was complete, I could update the particle velocities with the pre-calculated list and update their positions accordingly. Cyclic boundary conditions were achieved the same as before, and all the starting conditions were maintained.

With the model functioning I could now model the behavior of 100 particles. Figure 6 shows snapshots of the resulting particle positions at the timesteps 1,10,100 and 200.

Figure 6



As we can see the bacteria begin by spreading outwards in clumps, with the first thee evolutions showing the formation and motion of larger structures. However, the fourth frame shows a dissolution of those structures in favor of an even distribution. This unexpected behavior is perhaps the result of a combination of the bacteria growing too far apart to interact, or of clump formation resulting in them passing through the groups and ‘shooting’ out of the sides in their attempt to reach the strong center of mass, like gravitation slingshoting.

What more interesting is, when we invert the sign of alpha, we find the same behavior over larger timescales. As the bacteria are repelled from one another they spread into an diffuse even distribution, and by the 200th timestep are indistinguishable in behavior from the bacteria which are attracted to one another.

Parallelization

If we wished to parallelize the calculations for the above programs, we would want to instruct python to perform simultaneous calculations for the velocity and positions updates of the agents. We would want to calculate their decisions at the same time, rather than in order.

Conclusion

In this lab we first outlined my implementation of the multi-agent bacteria model and analyzed the behavior of the bacteria for the P_a and P_b food density field functions. We then examined how varying the bacteria’s speed affects their behavior. This report also described the implementation and behavior of another bacteria model, featuring bacteria who measure the center of mass of those around them and move towards it. Finally, this report sketched possible methods for parallelizing the previous models.