SPP tutorial for BIOL357 Spring 2023

All the m-files you need to do this tutorial is in the folder SPP that you should now have on your desktop and from the previous Matlab/Netlogo tutorial you know how to operate Matlab and run such m-files.

ATTRACTION ONLY (The **L**ocal **A**ttraction **M**odel.)

Here you will investigate the attraction only model without a blind angle. Review the lecture slides so you know what the model parameters represents and how the model works. You will only vary the model parameter ***c*** here. All other model parameters have been fixed to d=1, R=4, δ=0.5. You also have to specify the number of time steps ***t*** you want the simulation to run for.

1. To complete a) and b) below you will use the command `LAM(c,t)` to generate all three groups known to be produced by this model. Hint: One figure on slide 30 in Lecture 6 may be helpful.

a) Start by running the command with ***c=0.15*** and ***t=2000***. Keep watching the animation until the simulation stops, save the figure, and close the figure window. You **will** get one of the three groups you are looking for here, be **patient** and observe, it will eventually happen.

b) Now set ***t=50*** and run new simulations by slowly increasing ***c*** from ***0.35*** (in increments of ***0.2***). If the group you are looking for hasn't appeared when the simulation stops, increase ***c*** a bit and try again. Once you got one of the groups write down the ***c*** value used to generate it and save the figure. Continue until you have ***c*** values and figures for all three groups.

ATTRACTION AND ORIENTATION

Here you will explore how adding orientation affects group formation in the LAM.

2. Use the command `LAMO(o,c)` to investigate how increasing the relative strength of orientation ***o*** affects the formation and persistence of the groups you found in exercise 1. That is, use the same ***c*** value you used to obtain each particular group in 1 (mill, polarized, swarm) and run a sequence of simulations with increasing ***o*** from ***o=0.05*** (in increments of ***0.05***) with this same **c** and observe how the resulting group is affected by increasing **o**.

How much orientation does it take to significantly affect the formation and/or behavior of each type of group? That is, when ***o*** increases above some value another type of group forms, or the same group forms but much faster/slower. Write down an estimate of this ***o*** value for each type of group. Save figures showing what the 'significantly affected' groups look like.

ATTRACTION ONLY WITH A BLIND ZONE

Here we will study the attraction only model with a blind zone and will only vary the parameter ***d***. All other parameters are fixed to c=1, R=4, δ=0.5, and the angle specifying the blind zone is 2 radians ≈ 115 degrees.

3. Use the command `LAMB(d,t)` to generate a String, an Oriented Mill, and a Figure of Eight. Use one of the figures on lecture slide 32 to guide your choice of ***d*** values to try. Once you got one of the groups, write down the ***d*** value used. Also use the **ZOOM tool** in the Matlab figure window to zoom in on the group and then save it as a **png** figure.

String(s): Use **t=300**. If no string has formed by then change the ***d*** value and try again.

Oriented Mill: Use **t=500**. If you get non-oriented mills try again. (To save time, if a non-oriented mills has formed you may stop the simulation by closing the figure window.)

Figure of Eight: Use ***t=1500***. Do not mistake something else for a Figure of Eight! The Figure of Eight moves across the figure window in a fairly straight line. It is never stationary or rotate around a fixed point, that would be something else. Also, you may have to run several simulations (even with a good ***d*** value) before a figure of eight emerges.

ATTRACTION & REPULSION IN A WALLED-OFF REGION

Here you will work with an attraction-repulsion model where the particles move inside a walled-off rectangular area rather than on a doughnut shaped (torus) environment as before.

4. In a)-c) below you will use the command `LARWM(β,N)` to explore the potential existence of transitional behavior between milling and polarized groups by varying the size of the blind zone via the blind angle parameter β.

a) If $\beta=2.5$ we get milling groups, and if $\beta=3.35$ we get polarized groups. Verify this by running a simulations for each of these β values with $N=100$.

b) This suggests that something interesting happens for some β values between ***2.5*** and ***3.35***. Two possible scenarios A and B are

A. As β is increased from ***2.5*** to ***3.35*** there is a value of β, a threshold of sorts, at which the model entirely stop producing mills and from then on only produce polarized groups.

or

B. For some range of β values between ***2.5*** and ***3.35*** the model produces groups that continuously transition between milling an polarized motion.

Try to figure out which scenario most accurately describe what is going on by varying β in some appropriate way. Again using $N=100$. Describe how you decided to vary β and motivate which of the scenarios A and B you think is most likely. If neither A or B are any good, what actually happened?

c) Pick the β value that generated the most interesting behavior in b) and try a few different N values (e.g. ***30***, ***70***, 1***50***). Save the most interesting figure you obtained.

THE SHEPHERDING MODEL

5. Here you will use the command `SHEEP(N,n,t)` to study a shepherding algorithm. ***N*** is the total number of sheep and ***n*** is the number of neighbors a sheep has (topological interactions). If ***n≈N-1*** all sheep are attracted to all other sheep. If ***n=0*** a sheep is not attracted to other sheep at all and will run off on its own if the dog approaches. ***t*** is the maximum simulation time and the simulation will stop before then if the sheep has been delivered to the target (green asterisk).

Pick an ***N*** between ***30*** and ***60*** and run ***two*** simulations with ***n=N-1*** and ***t=2000***. Does the dog manage to herd the sheep to the target? If yes, decrease ***n*** by an appropriate amount and run ***two*** simulations (keeping ***N*** and ***t*** the same). Keep decreasing ***n*** until the dog fails in both simulations for a particular ***n***. Write down the *largest* ***n*** at which failure occurred, the ***N*** used, and save the figure. Note that it can happen that the dog never fails, even at ***n***=0. In this case write down ***n=0*** and save the figure.

END OF SPP-MODEL TUTORIAL

How to make an mp4 movie of a simulation

1. **Immediately** after the simulation has stopped type (or copy paste) **exactly** `MakeMovie(ans)` on the command line i.e.

```
>> MakeMovie(ans)
```

and press enter.

2. A file called filename.mp4 should now have appeared in your SPP folder. **Manually change the filename** to something else, e.g. oddthing.mp4.

Important

- If you run another command in between 'simulation stop' and 'movie making' **no movie** will be created. (`ans` is the output of the last command executed in Matlab).

- If you forget to rename the file so that it is still called filename.mp4 the next time you make a movie the old one will be overwritten and **lost forever.**