Worm Blob Movements Across the Ground

Zachary Ryan, Adam Salyers

Dated: 6 May 2020

The congregation and clumping of worms in a cohesive manner to survive for periods of time in both water and open air conditions, is currently being studied. We have built on worm observations from Georgia Tech, as well as herding simulations from the University of Uppsala Sweden. In particular we have simulated the creation and movement of worm blobs in an open air system to better understand the interactions between individuals worms, as well as their properties from both liquids and solids. By modeling the creation and movement of the individuals and collective blob worms we presented a viable base for future simulations in the field.

I. Introduction

The occurrence of group coordination between individuals, most often simple, as a mechanism for survival occurs across a variety of areas of biological study. Group coordination occurs most often in smaller species, specifically insects. Ant's will create towers that act in similarity to a reverse flowing fountain[1], bees will vibrate in cohesion as a defense mechanism against hornets[2]. Or in the case we modeled, was group worm behavior as an open-air survival mechanism. Worms of the genus tubifex, species california blackworm, and other species, have been observed to form masses, or blobs, similar in properties to both a liquid and a solid[3]. The worms have adapted to form these masses as a survival mechanism. In water the blobs more easily float and form bridges[3-1], and in open air conditions allows the worms to survive for longer. In open air conditions a lone worm will usually die within one hour. Whereas a thousand worm clump was observed to have twenty percent of its members survive after eighteen hours[3].

Modelling of group coordination requires observation and experimentation in order to obtain realistic models. Although often it is impossible to make minor changes to already existing models, we can sometimes take elements and incorporate them into our own model. As a starting point it was important to understand how a single blob moved and acted. A paper out of Uppsala University in Sweden outlined a model created to simulate the shepherding of sheep with dogs, and shepherds[4]. Within the paper they outline repulsion and attraction between shepard, sheep, and the sheeps neighbors. These influence how the sheep's velocity changes, and the results are a relatively effective model for shepherding sheep. The concept of worm blob modelling and shepherding share many similarities. The goal of the sheep agents is to remain away from the shepherd, but more importantly to move towards the center of the flock. The model presented in the paper showed the flock moving as a group with individuals members having somewhat random movement. For worm blobs the characteristics are similar. Unlike the flock and shepherd though, we require an extremely low repulsion force so as to allow the worms to come together. So we decided to use a similar idea to model the group movement, however instead of individual agents and nearest neighbors we used a center of mass of worms and of worm clusters. When observing worm clusters we noticed that the blobs moved far more like blobs than as individuals. So logically it would make sense for the worms to follow what the blob was doing instead of individual neighbors.

II. Related Work

The study of worms is a newly emerging area of study for biologists, especially in relation to simulating how these animals move individually and collectively. Our primary sources of information are all from recent studies and observations conducted on worm blobs in controlled and uncontrolled settings.

1. Worm blobs act like a fluid and a solid

Milius 2019[3] describes the nature of a worm blob to have properties of both a liquid and a solid. Each individual worm is a solid object that takes on no liquid like properties, but when a large mass of worms congregate together the mass starts to take on other physical properties not seen in solid objects. This mechanism causing the worm to congregate is thought to be a survival tactic. In the event of a worm blob becoming too hot, worms will start to disperse from the mass to cool off. The opposite is seen when worms get too cold, they form a mass to keep the temperature up. Another reason for the massing is that a single worm can only survive a few hours without water. When the worms mass however, the amount of time worms can survive is three or four fold what they could survive on their own. A video[3-1] shows the movements of worms on both water and land.

2. Creatures from the Sewer

M 2009[5] shows a video of what initially starts out as the inside of a sewer pipe. After a few seconds we are given a clear view of what appears to be a growth of some kind and would occasionally contract like you would see in a human muscle. This growth would eventually be identified as a mass of worms within the sewer line. As for the contractions Dr. Timothy Wood can be quoted as saying “the contractions you see are the result of a single worm contracting and then stimulating all the others to do the same almost simultaneously, so it looks like a single big muscle contracting.”

3. The Internet Is Freaking Out Over This Perfectly Normal Pulsating Mass of Sewer Worms

Star 2018[6] refers to a video taken outside Houston where a mass of worms can be seen in a few inches of water. This mass of worms behaves slightly differently than the one seen in M 2009[5]. Here the individual worms are being much more active, they are starting to branch out from the mass, and some are leaving the mass altogether.

III. Problem and Direction

The goal of this project was to successfully model the movements of a worm mass when on solid ground and to show the scattering of worms from the mass. The rules and assumptions used for creating the model largely come from the videos talked about previously.

Rules:

- All worms tend to move towards the center of mass of the group of worms
- Once a worm has left the mass it will continue to move away from it
- There are two navigation methods
  - worms in the mass navigate through combining multiple interacting forces to determine its next move
  - Worms branching out move in the opposite direction of the center of mass of the worms

Assumptions:

- Worms only move one direction, i.e. the tail must always follow the head and can not move backwards, head following tail
- Radius of the initializing circle is 10 units and 0° needed between points.
- Each worm is 5 points long
- Distance between all points of a worm are not equal.
- Total number of worms to a blob, we have found 400 to be sufficient
- Max number of branches is 20% of the total number of worms
- A worm has the option to branch if it's in the outer 5% of what the radius is found to be for the mass of worms
- Worms are not allowed to branch for the first 10 steps to allow the for the entangling of the mass of worms
- Even though the worms are touching in the mass they still have a small repulsion force to one another
- Coefficients for the velocity vectors were found by running the model until getting the desired effect

With these rules and assumptions in mind it is possible to create a model that successfully mimics the movements of worms on the ground.

IV. Running the Simulation

A. Setup

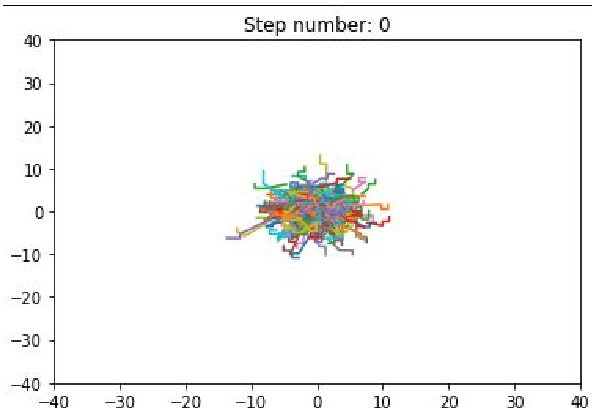
Initially we were unsure if something like this had already been done or if other programs could solve what we needed to get done. When the project was accepted however, we were given MatLab code by Chantal Nguyen, which was a program designed for protein folding that was changed to depict the movements of very small mass of worms. When trying to work with this though it proved to be too computational time consuming for the number of worms that we needed.

This setback prompted us to look at the flocking and herding code from homework 3 to see if it could be modified to suit our needs. The structure of this code proved to be very useful and thus was the basis from which we began modifying to suit our needs.

Something else of importance to make note of is the type of canvas we used has a wrap around edge. While this is not realistic of what happens in real life, we felt it was better than a worm going out of frame.

B. Initialization

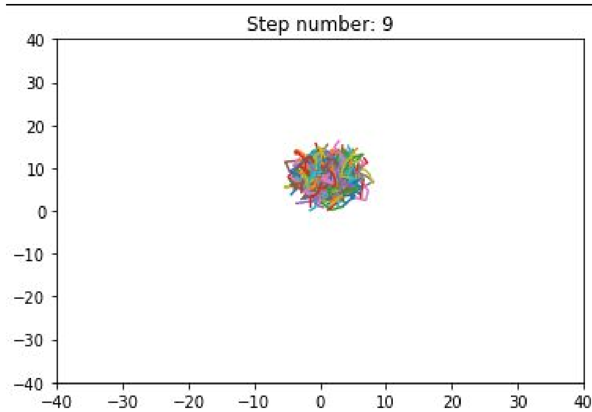
For the worm to actually become part of the model it must first be created and given some starting parameters. It may or may not come as a surprise but the best starting shape for all worms was a random distribution of worms that forms a circle. To do this an array is created with the number of elements specified by the user for the number of worms. Each element consists of an 'x' and 'y' coordinate from the Cartesian plane and this point is considered to be the head of each worm. Each head position was found through a python function that returns a random distribution of points in the specified circle size. Now we need to initialize the rest of the worm's body. This was done by appending four more points to the head of the worm where each point was a distance of one unit away from the previous and the direction of each new point is random. Each worm is also given a small random velocity to begin, each velocity has a max of 1 unit here. An example of this can be seen in the figure below:



Something to note about this figure is it may appear as though we are already seeing branching occuring. This is not the case, due to the randomness of how a worm's body is formed it's possible for the body to appear as though it's branching away from the mass at step 0.

C. Stepping through the Program

After initialization of the worms happens, each concurrent step must calculate the new position and velocity of each worm. For the first ten steps all worm velocities are found through the non branching method. This is done because we want to simulate the movements of a mass of worms before beginning to branch. After the tenth step branching is allowed to occur and if a worm fits all the criteria to begin branching its velocity is then found through the branching method. Below is a picture of what could be expected after the tenth step:



After step ten we must start determining if worms should branch or not. To do this we only need a few calculations. The first being the center of mass of the individual worm, the second being the center of mass of the blob of worms, and the third being the radius of the blob of worms. The radius is found by taking the worms farthest apart in the vertical and horizontal directions, then take the average of the two distances and divide by two. We then find the distance between the center of mass of the worm and center of mass of the blob. We then check if the distance found is greater than what 95% of the blob radius would be. If so the worm would then be considered as branching and will get its velocities from the branching method described below. If not it will continue to follow the non-branching method described below.

1. **Non-branching**

The purpose of the non-branching method is to find velocities for all worms so that the mass of worms will stay together while the worms move throughout it. This also has the added benefit of simulating the movement of the entire blob as a whole around the frame. This was done using a combination of four different forces:

Attraction

The attraction force between a pair of worms is the vector from worm A to worm B. We then multiply this value by the attraction coefficient of 0.001 to get the attraction velocity or v^1.

Repulsion

The repulsion force between a pair of worms is calculated by dividing the vector from worm A to worm B and then dividing by the square of the vector's magnitude. This is then all multiplied by the repulsion coefficient of 0.01 to get the repulsion velocity or v^2.

Average

The average force for all worms is simply the average of all previous velocities. This is multiplied by a coefficient of 1 to get the average velocity or v^3.

Random

The random force is used to better replicate the unexpected macro movements of the worm. Our random velocity or v^4 is simply a decimal number between 1 and 2 which is then multiplied by a coefficient of 0.90.

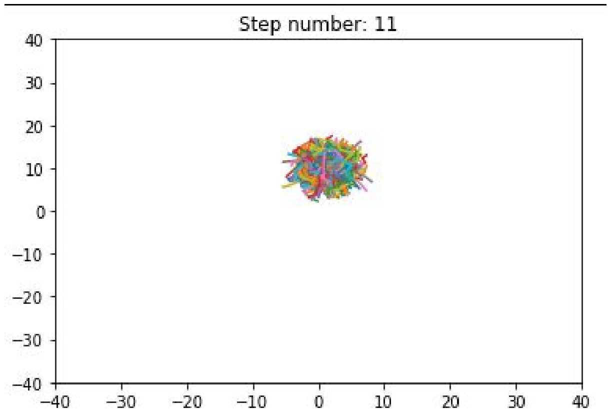
When summing each of the above velocities together it provides the velocity at which our worm is moving. This is done for each worm individually and is recalculated every step. If a worm is never chosen to branch it will continue to use this method to obtain its pathing for the remainder of the simulation.

2. **Branching**

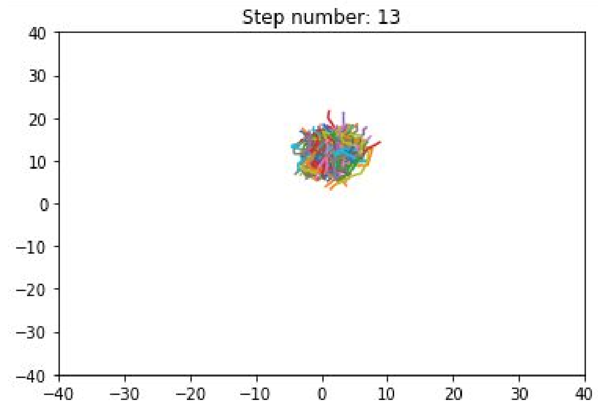
The purpose of the branching path finding is to allow worms that have been chosen for branching to move away from the mass of worms. To do this we simply wanted the branching worm to move away from the center of mass of worms. To do this though was not as simple as just giving the worm a direction, the correct direction had to be found and for each worm this is different. By taking the rise and run between the wom and the center of mass of worms we could get a slope that we would want our worm to roughly follow. This slope could then be used to calculate the vertical and horizontal velocities of worms branching out. If a worm is considered to be branching this method for calculating the velocity is always used.

A check was needed to be put in place however after calculating the branching worms velocity. It was possible from this method that a velocity would be so large that it was not physically possible for a worm to move that much distance in one step. If this was the case then the velocities are changed back to something that would be expected but with some randomness so as not to become too uniform.

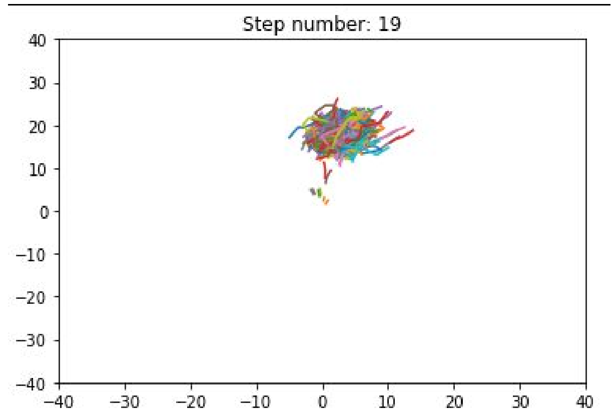
After the first step of allowing branching it can be hard to see if any branches are even forming. In the example below 18 worms have started to branch:



Moving just two steps forward in time it becomes much clearer that worms are starting to branch:



Then finally by step 20 you can see clear separation between branching worms and the worm mass:



V. Limitations, Conclusions, and Future Work

A. Limitations

As seen by the figures and models generated by our program we only modelled in 2D. This allowed us to save on computing and simulation time. However it would be far more realistic to compute a 3D simulation. With such high density on individual agents we can't realistically prohibit two agents from occupying the same space. Obviously in reality this is impossible.

In addition the program is limited in that it cannot accurately simulate followers of the branching worms. We tried creating a solution to this but could not find one that also would not destroy the mass of worms. This is necessary in modeling the worms behavior in water.

B. Conclusions

Modeling worm blob movement could lead to many important insights regarding not only biological group cohesion, but also the movement and behavior of liquid/solid objects. Our model presents an effective and realistic simulation of these worm blobs in an open air environment. By building on observations from shepherding simulations as well as the worm blobs themselves we could produce models that could simulate a blob moving and surviving for an extended time period, as individual worms move and leave independently.

This presents a basis for further worm movement. Any further models or simulations can use the same ideas and implementations of individual worm clusters to better inform their own.

C. Future Work

Our program modeled the open air conditions with a single cluster of worms. Since we found very little observations of open air worm clumping with multiple clumps our model is not designed to model multiple clumps. Initializing multiple clumps will not break the simulation, but checks would need to be made for what group a worm belongs to accurately describe the movements. Would also need to look at the interaction between a worm branching from one group and another groups blob of worms.

We also were unable to model the worm clumps behavior in water. In water the worm clump observations have a few key differences. Firstly, the worms branch out in specific places, worms don't often leave the clump like seen in open air observations. Secondly the only accurate observation we have from water submerged clumps was with multiple clumps.

Both of these would be very good directions for future work. In addition temperature gradient, and the introduction of food could provide some more insight into the individual behavior of the worms. This insight in turn could provide some insight into how individuals moving as a collective can act as both a liquid and a solid.

VI. Contributions

Both group members contributed to the coding and writing of the paper. Oftentimes we would be working together through a zoom call or a normal phone call.

[1]Foster, P. C., Mlot, N. J., Lin, A. and Hu, D. L. (2014). Fire ants actively control spacing and orientation within self-assemblages. *J. Exp. Biol.* 217, 2089-2100.

[2]Yamaguchi, Y., Ugajin, A., Utagawa, S. *et al.* Double-edged heat: honeybee participation in a hot defensive bee ball reduces life

expectancy with an increased likelihood of engaging in future defense. *Behav Ecol Sociobiol* **72,** 123 (2018). https://doi.org/10.1007/s00265-018-2545-z

[3]Milius S. Worm blobs act like a fluid and a solid [Internet]. Science News - February 2, 2019 Worm blobs act like a fluid and a solid. Science News; 2019 [cited 2020Apr27]. Available from: https://www.sciencenewsdigital.org/sciencenews/february_2__2019/MobilePagedArticle.action?articleId=1458580#articleId1458580

[3-1]https://www.youtube.com/watch?v=LmCZNJW8kdE

[3-2]https://www.youtube.com/watch?v=uOkPBGwH0lM&feature=youtu.be

[4]Stro ̈mbom D, Mann RP,Wilson AM, Hailes S, Morton AJ, Sumpter DJT,King AJ. 2014 Solving the shepherdingproblem: heuristics for herding autonomous,interacting agents.J. R. Soc. Interface11:20140719.http://dx.doi.org/10.1098/rsif.2014.071

[5]M. Creatures from the Sewer [Internet]. Deep Sea News. Deep Sea News; 2009 [cited 2020Apr27]. Available from: http://www.deepseanews.com/2009/06/creatures-from-the-sewer/

[6]Starr M. The Internet Is Freaking Out Over This Perfectly Normal Pulsating Mass of Sewer Worms [Internet]. ScienceAlert. 2018 [cited 2020Apr27]. Available from: https://www.sciencealert.com/internet-freaking-out-over-this-perfectly-normal-pulsating-mass-of-sewer-worms

Github link to code:

https://github.com/AdamSalyers/Dynamic-Models-Final