Chapter 1

Slime Molds and Swarms

1.1 How do Molds aggregate?

The self-organizing behavior of slime mold is another complex phenomenon. When environmental situations worsen, the slime mold amoebas spontaneously change from moving around randomly, to aggregating together. How is this happening? In their paper *The Initiator cell for Slime Mold Aggregation* written in 1958, Herbert Ennis and Maurice Sussman proposed the naive explanation: that there must be some organizer. They “conclude[d] that the formation of an aggregate requires the presence of a specially endowed cell, termed the “initiator,” which induces its neighbors to begin aggregation”.

It was not until the seventies when progress was made toward the modern explanation of the slime mold aggregation. In 1972, Edmund Goidal and co., hypothesizing that mold aggregation occurred because of the molds’ release and sense of pheromone, conducted an experiment that consisted of releasing antibodies into the mold environment which broke down the pheromone. The results confirmed the hypothesis: the mold failed to aggregate.

With the discovery of pheromone, it becomes apparent that an agent based model is appropriate for simulating the slime mold: the behavior can be reduced to as being made up of simple agents operating under simple rules, interacting locally and without any central control. In *Turtles, Termites and Traffic Jams*, Mitchel Resnick presents such a model. The mold world is made up of a 2-D grid with hard edges. Each square on the grid represents a patch. With every time step, the mold agent:

- Drops pheromone on the patch it is currently resting
- Senses for pheromone in the surrounding patches
- Moves to the one with the highest density
The density level on the patches diffuses and evaporates.

**Exercise 1** - The different rules described by Resnick can be easily implemented. Thus in order to better understand the model and the real mold’s behavior, fill in the empty methods in the provided framework: [http://greenteapress.com/complexity/slimemold/exercise1.py](http://greenteapress.com/complexity/slimemold/exercise1.py). If you prefer to look at the final code directly, jump to exercise 2.

1. In the class Patch, modify the method `receive()` so that it takes a patch object and increases the density of that patch by one.

2. In the class Mold, modify the method `emit()` so that the mold increases the density of its current patch by one.

3. In Patch, modify the method `evaporate()` so that the given patch’s density is decreased by 5%.

4. In Patch, modify the method `get_neighbors()` so that it returns a list of the given patch’s 8 neighbors.

5. In Patch, modify the method `diffuse()` so that it looks at the given patch’s neighbors, finds the one with the maximum density and assigns 95% of its density to the current patch.
When some randomness, an appropriate balance between diffusion and evaporation, and a large enough population is added to the system, the amoebas begin to form clusters. The clusters grow and merge until, with enough time, all of the amoebas have aggregated to a single cluster.

The agent based model exhibits behavior similar to that of the real life slime mold, so the real-life aggregation of the mold could technically be an emergent property of a decentralized—not centralized—system. The fact, that slime-mold amoebas can cluster given that they are provided to the same, simple rules proves that mold clusters formed in this way can exist. This is an example of an existence proof, see wikipedia.org/wiki/Constructive_proof.

1.2 Parameters, deciding factors

Resnick’s slime mold simulation is constrained by a handful of parameters. In addition to the number of mold, key methods like \texttt{emit()}, \texttt{sniff()}, \texttt{evaporate()}, and \texttt{diffuse()} depend on parameters: how much pheromone to emit? where to sniff? how much to evaporate? how much and to whom to diffuse? These, in combination, are responsible for making or breaking the desired behavior. The ideal conditions allow for the eventual formation of one cluster. Less than ideal conditions lead to the formation of either no clusters or small clusters that have little hope for further aggregation. Clusters fail to form when their is too little or too much pheromone.

Figure 1.2 shows the relationship between the number of slime mold, the range of the molds’ sniff and the behavior of the system. With few amoebas and they have no chance of coming close enough to one another to form a pheromone cloud to attract others. Too many amoebas and the grid is overwhelmed: the mold cannot move to a patch that is currently occupied by another mold.

A wider sniff range, on the other hand, means that the amoebas are able to make smarter decisions about the location of the pheromone gradient: they will travel less and be more boring. They will form clusters faster; the clusters will be smaller, mostly stationary, and unlikely to merge with other clusters. More limited sniffing forces the mold to travel and, in some cases, even break away from a cluster. Interestingly, breakaways frequently facilitate merges of clusters.

Exercise 2 - In this exercise, download http://greenteapress.com/complexity/slimemold/SlimeMold.py and explore the effects of \texttt{evaporate()} and \texttt{diffuse()}, by moving the sliders. What happens when they are both at their max? min? switched? Can you find conditions that allow for the faster formation of a single cluster?
CHAPTER 1. SLIME MOLDS AND SWARMS

1.3 What work?

Positive feedback

Positive feedback is common in many systems; that it is present in a system modeling decentralized swarm behavior is more unusual. In *Emergence* (2001), Steven Johnson comments on Resnick’s model and suggests that the slime mold model can be used to talk about an idea revolution. Each mold is an individual, and the pheromone, his idea. As an individual randomly encounters another their ideas coalesce (maybe one’s idea overtakes the other). When there are two or more of them, they will be more likely to attract still others to their cause (positive feedback) until, after many time steps and under the right conditions, all of the individuals form one cluster. All of the individuals unite under one idea. Further, the type of ideas in question could be anything from fashion to politics; the model does work in the direction of answering under what conditions all of the ideas would coalesce.

Figure 1.2: Phase diagram between the two parameters, number of agents and sniff range. Paired together, the two decide the size and fate of clusters.
1.3. WHAT WORK?

Slime mold in other models

In an experiment published in January 2010, researchers from Japan and England used slime mold and oat flakes to reproduce the Japanese rail system: a large transport network, famed for its efficiency; the product of years-worth of toil by many talented engineers.\(^1\) The source of the motivation for the experiment is that the particular species of slime mold, *Physarum polycephalum* exhibits the swarm intelligence to find the shortest path between food wells. The experiment produced a network that was staggeringly similar to the actual railway network.

Unlike the computational model discussed in this chapter, the experiment was composed of real slime mold placed in a world that resembles Tokyo bay. Tokyo city is the molds nest and the surrounding major cities, oat flakes. The whole setup lies on Plexiglas with hard edges around the cost so that the mold do not escape into the ‘sea’. At first the slime mold spread out evenly from the nest to every spot on the map, looking for food; however, as time progresses paths begin to emerge between the oat-flake cities and the mold fade from the empty spots on the map.

Digital mold are still faster and easier to implement. The creation of a highly-efficient transit network could be as easy as laying out the desired map and running a simulation, assuming that the model accurately reproduced the behavior of the real slime mold. It is important to note, that this model is different from the one discussed in this chapter as it describes the behavior of a different species of slime mold. The aggregation model could potentially be extended to account for the existence of food. It would be necessary to implement a new "food" object with few new rules concerning this element, draw a specific world that is relevant and run the application.