Complex System 530 Lab 1

Shirlyn Wang

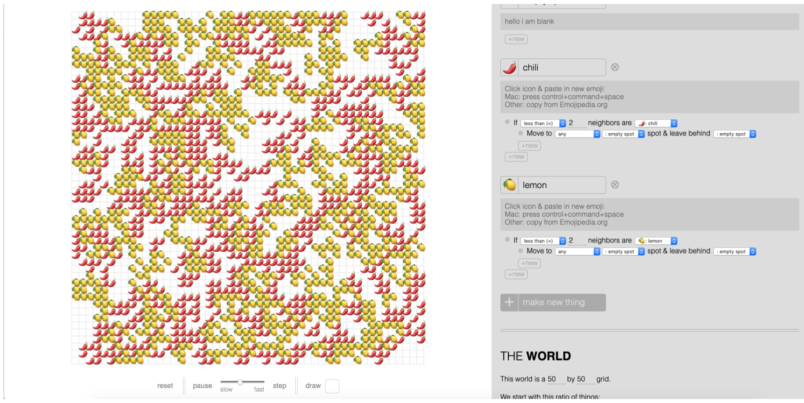
Feb 8, 2021

All codes, figures and vidoes are found in my Github repository: `https://github.com/ShirlynWY/CMPLXSYS_LABS.git`

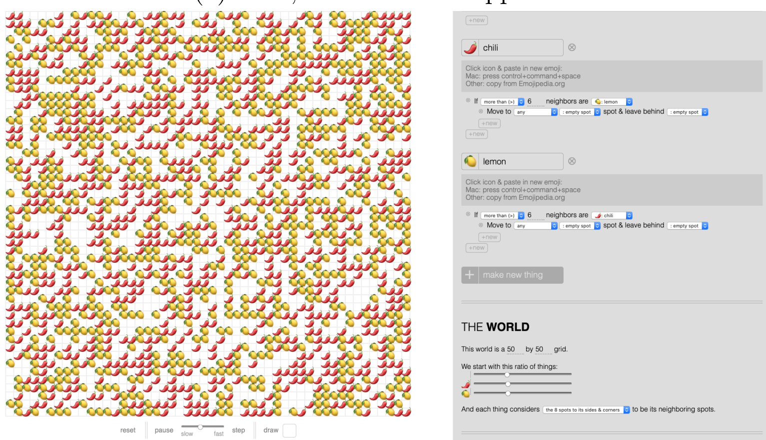
Part I: Emoji-ABMs

1. The Schelling Model
   My model uses lemon and chili.
   Rule 1: if less than X of its neighbors are like itself, it will move to a random location anywhere on the board.
   Rule 2: if more than Y of its neighbors are NOT like itself, it will move to a random location anywhere on the board.
   In the explanations below, when I talk about X, I mean Rule 1 is applied. When I talk about Y, I mean Rule 2 is applied. Each subfigure contrasts the different results of rule1 and rule2 even though they sound similar.
   I started the simulation with about equal numbers of empty spaces, lemons and chilis unless otherwise noted.

   (a) I started the simulation with about equal numbers of empty spaces, lemons and chilis. Figure 1 shows the results with the 2 slightly different rules. If each plant requires at least 2 of it neighbors to be the same, small clusters appear. If each plant requires no more than 6 of it neighbors to be the other kind, even smaller clusters appear.



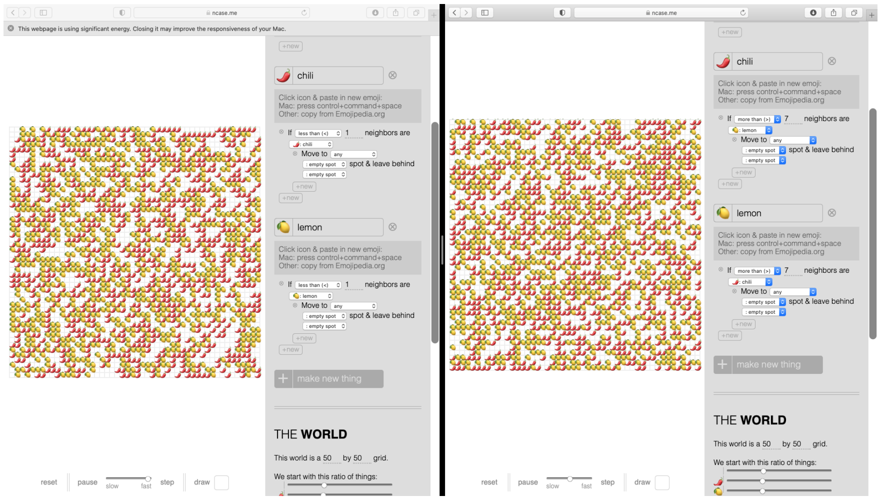
(a) X=2, small clusters appear



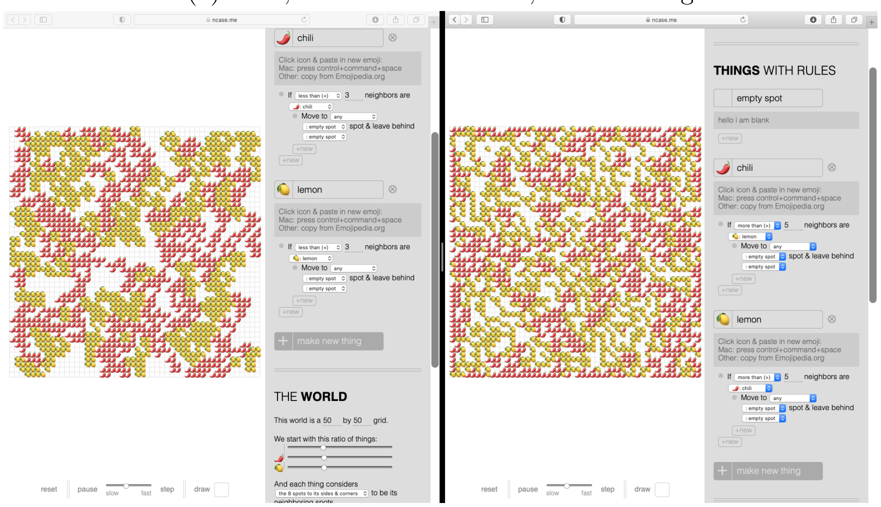
(b) Y=6, smaller clusters compared to (a)

Figure 1

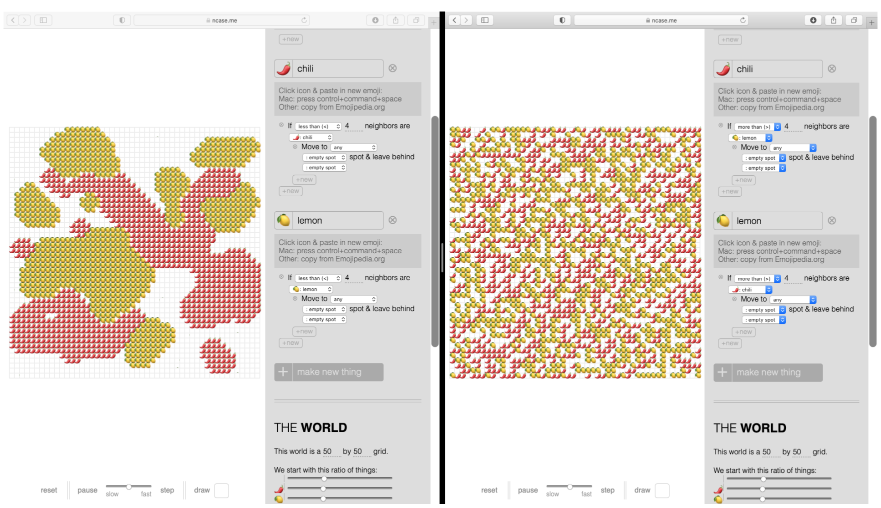
(b) Figure 2 shows the results of varied number of neighbors for both rules. In each figure, the window on the left shows the result of Rule 1 ($X < a$). The window on the right shows the result of Rule 2 ($Y > b$) where $a + b = 8$.



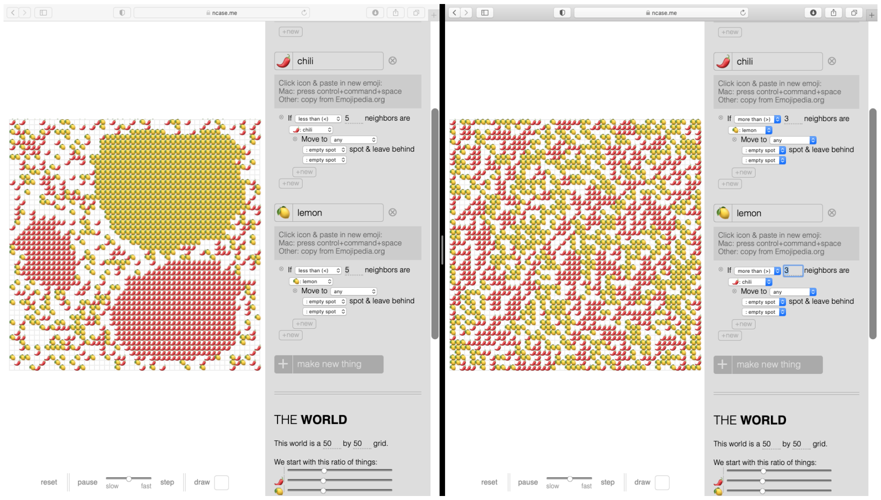
(a) X=1, Y=7: no one moves, no clustering



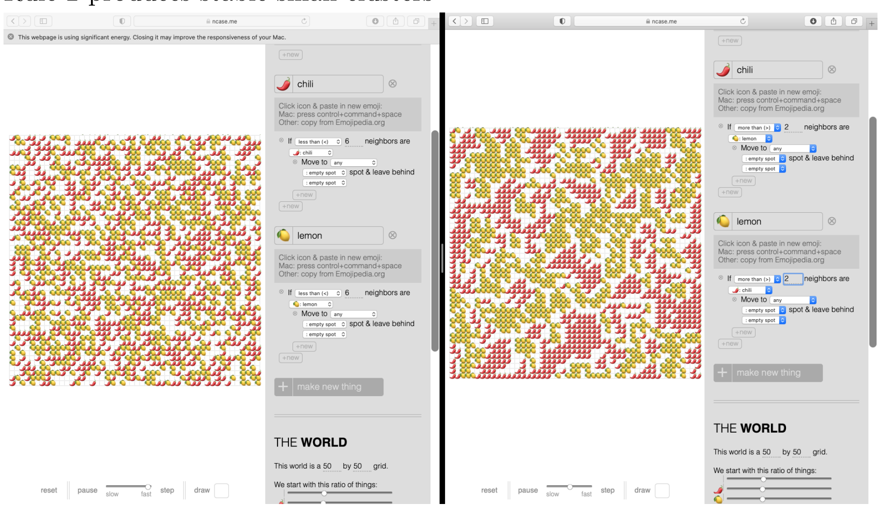
(b) X=3, Y=5: Rule 1 produces bigger clusters with no gaps between clusters.



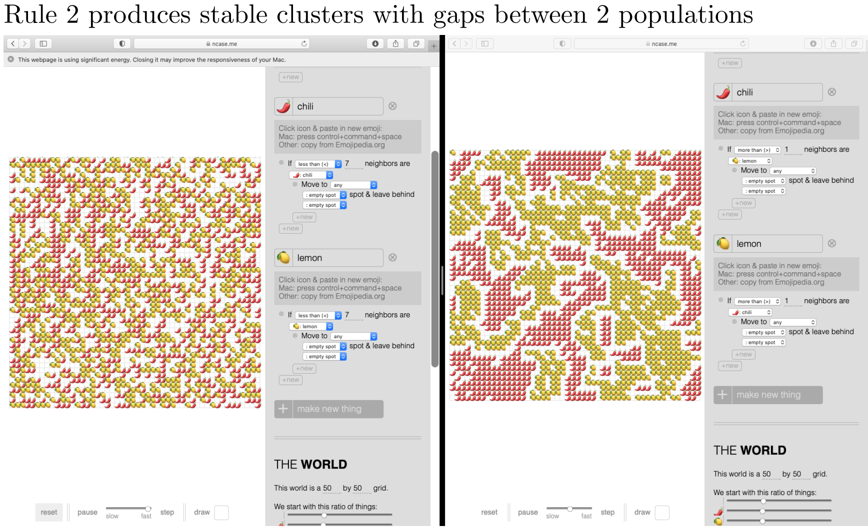
(c) X=4, Y=4: Rule 1 produces big clusters with no gaps between clusters. Rule 2 doesn't produce obvious clusters



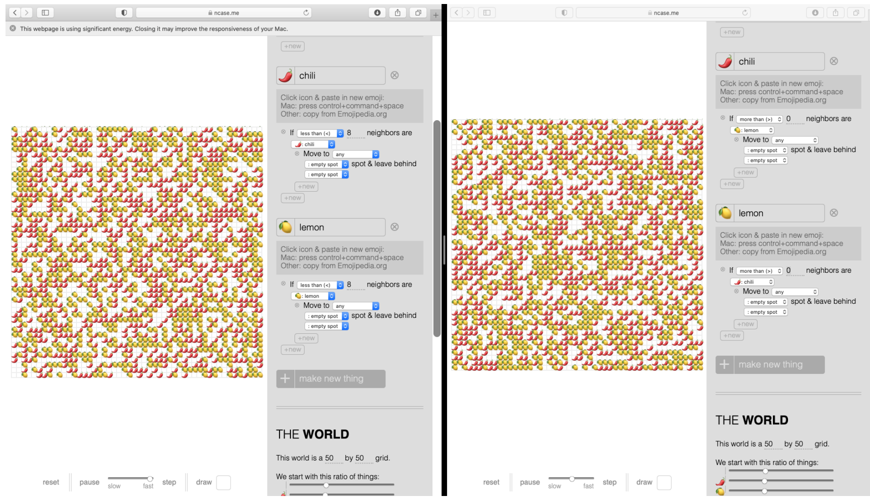
(d) X=5, Y=3: Rule 1 produces big clusters but boundaries keep fluttering. Rule 2 produces stable small clusters



(e) X=6, Y=2: Rule 1 plants keep moving and no clustering is observed. Rule 2 produces stable clusters with gaps between 2 populations



(f) X=7, Y=1: Rule 1 plants keep moving and no clustering is observed. Rule 2 produces stable bigger clusters with gaps between clusters

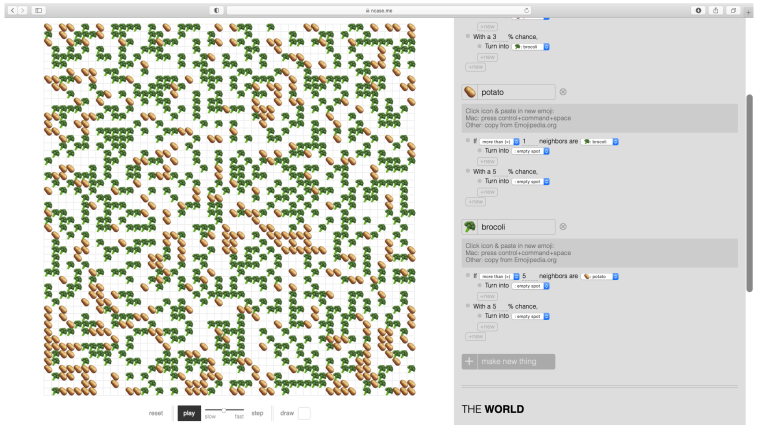


(a) X=8, Y=0: plants keep moving and no cluster is observed with either rule

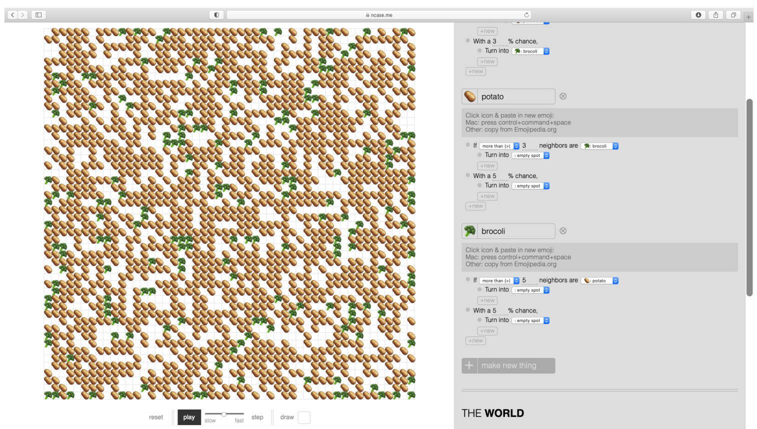
Figure 3

(c) Rule 1 produces clustering with possible fluttering on the boundaries regardless of the density of plants. There is no separation in all cases of rule 1. Rule 2 only produces clustering if the density is big enough. When $Y \geq 4$, there is clustering but no separation. When $Y = 2, 3$, there is clustering and separation.

2. My Cancer-Immune Dynamics model (Figure 4)
Let broccolis represent immune cells, and potatoes represent cancer cells. The system starts with more cancer cells than immune cells. Assume cancer cells grow faster than immune cells. To model this in the Emoji simulator, I set the empty space to have a 10% chance to turn into a potato and 3% chance to turn into a broccoli. Assume both types of cells die at the same rate. I set both potato and broccoli to have a 5% chance of turning into an empty space. Immune cells attack cancer cells. On the other hand, immune cells die from interacting with cancer cells. Potatoes turn into an empty space if more than X neighbors are broccolis. Broccolis turn into an empty space if more than Y neighbors are potatoes. Varying X and Y produces different result. Small X and large Y model strong immune cells with good killing abilities. Only when X is low and Y is high can there be fewer potatoes than broccolis on the map in the long run.



(a) X = 1, Y = 5: There is more broccolis on the grid in the long term



(b) X = 3, Y = 5: There is more potatoes on the grid in the long term

Figure 4

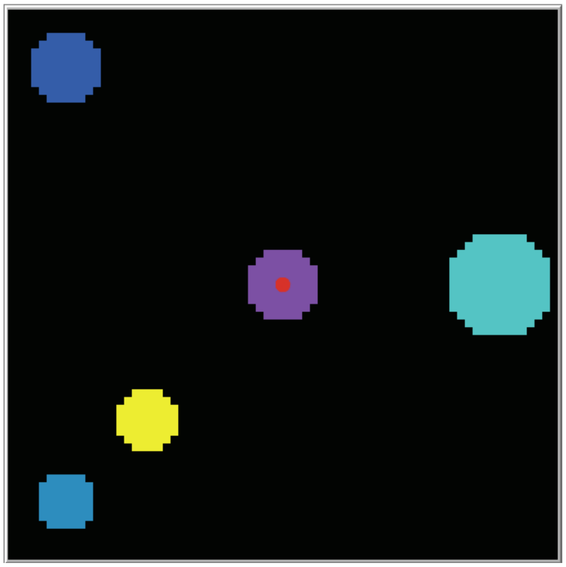
Part II: NetLogo Models

1. Ants Model

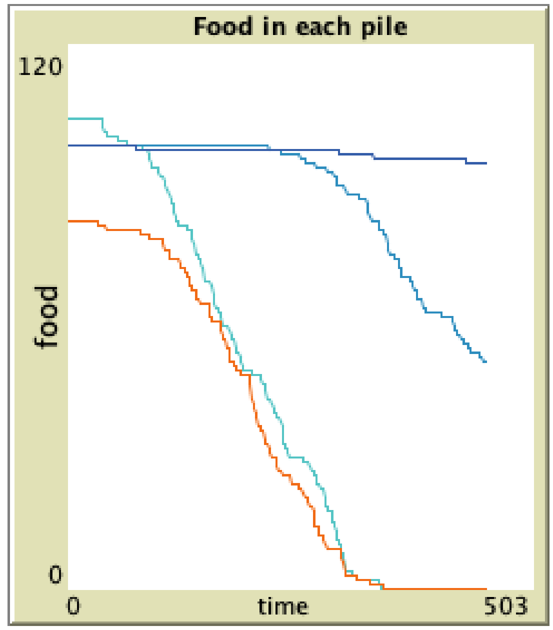
   (a) When diffusion rate = 99, evaporation rate = 90, all 3 piles of food are consumed simultaneously.
   When diffusion rate = anything, evaporation rate = 99, the ants are really inefficient.
   When diffusion rate = 20 and evaporation rate = 4, the ants are very efficient.

   (b) (Shown in figure 5) Adjust the two of the food sources to

   - make them farther away from the nest
   - make one bigger and another one smaller
   - Add a new food source between the nest and another food source



(a)



(b)

Figure 5

The ants takes longer to consume all the food on the map because the new food source (yellow) distract most of the ants from finding the blue food source at the lower left corner. It is impossible for ants to consume all piles of food at the same time in this case.

(c) Adjust the original Ants model code to "Poison" one of the food sources so that ants who eat from it have

- a 50% chance of dying immediately and then
- have a 50% chance of dying within 5 ticks of eating from it

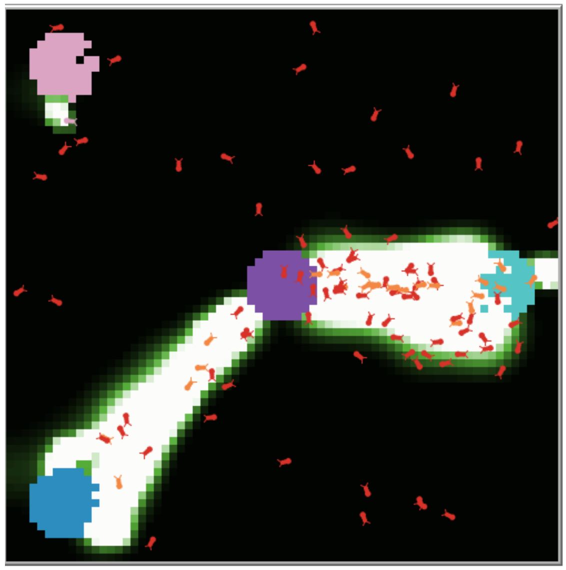


Figure 6: Caption

The pink pile is the poison. The pink pile attracts ants and kills them. Most ants die from eating in the pink pile. Those that don't die might wander into the poison again and get killed.

(d) The nest-scent is a new kind of pheromone that is released by ants for some limited time period after they have left the nest.

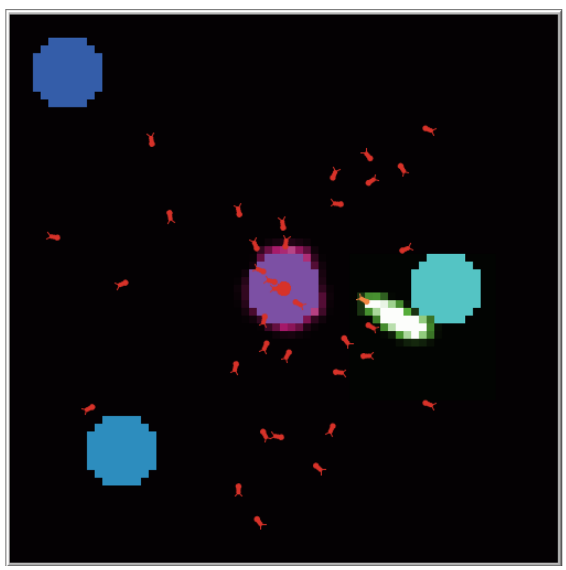


Figure 7: Caption

I don't think this change affect on nest-scent impact the results much. If there are enough ants on the map, there will always be some ants leaving the nest, so the nest scent will be strong enough so that the ants carrying food will be able to return to the nest accurately.

2. Build a simple model
   Color patches appear as the figure below. (Code is found in Github file named "PART2_simple.nlogo".

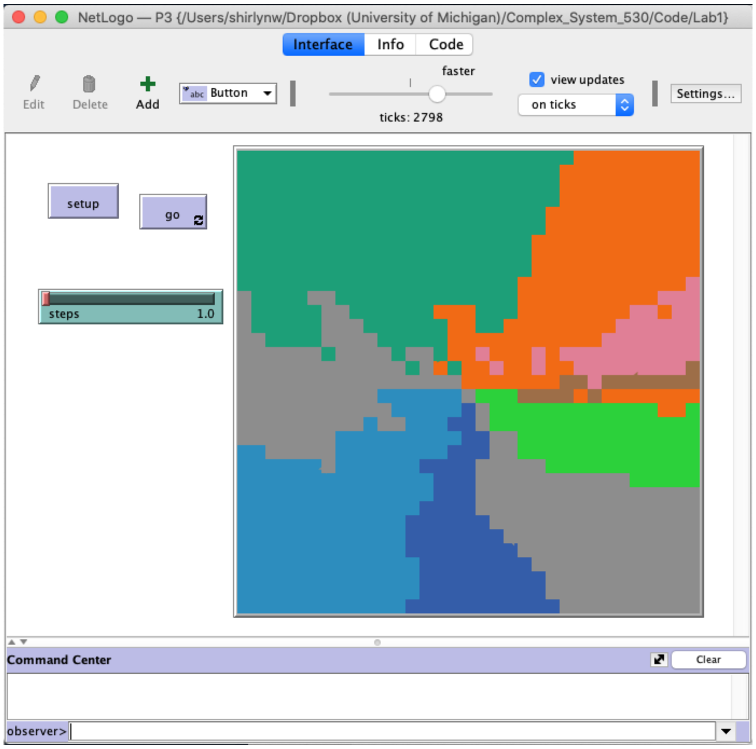


Figure 8: Simple model for color updates

Part III: Python

1. Code in Github: "chaosgame_animate.py". I also generated a video for the original chaos game.

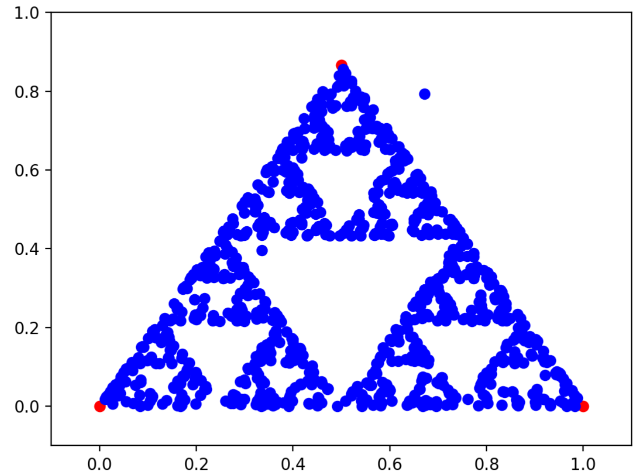
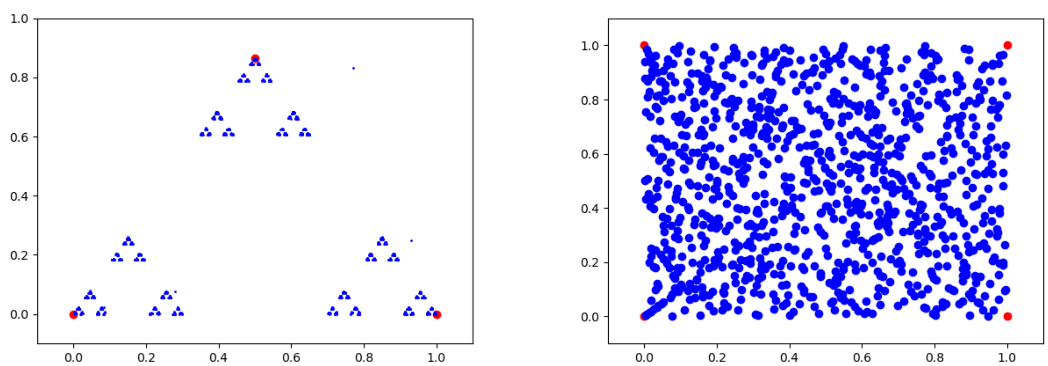


Figure 9: Chaos game

2. Optional Problem 2
   Regular pattern is still observed when I changed the midpoint function to the point that divide the distance between 2 points 6:4 (figure 10a) . But when we start with 4 points on the vertices of a square, there's no pattern observed (figure 10b.



(a) (b)

Figure 10: Modified chaos games

Part IV: Project Ideas

I'm thinking about building an ABM model for cancer-immune dynamics or some problems in ecology.