University of Ottawa

MAT3395 Mathematical Modeling - Final Project

Modelling Cooperation Levels in a Society through Spatial Prisoner's Dilemma

Author:	*Student Number:*
William Clelland	6490111
Mohammed Chamma	6379153

April 24, 2015

CONTENTS

I. OBJECTIVE

Our objective is to study the levels of cooperation in a society where the individuals are in competition for resources. More precisely, we study the levels of cooperation in society by assuming that only two strategies exist, that of cooperation (for example, sharing resources) and defection (keeping resources to yourself or taking them for yourself). We then quantify the advantage defection has over cooperation and track some initial society over time, counting the number of individuals who choose to cooperate and the number of individuals who choose to defect in each time step. We then analyze the local and global behaviour of the society of individuals. In the end, we hope to determine the necessary parameters for long term cooperation and to perhaps understand how cooperation might emerge in society.

II. BACKGROUND

That cooperation should exist in a society of beings that evolved through competition and survival of the fittest seems paradoxical. Cooperation often puts ones own survival in jeopardy, especially if the others don't cooperate with you. For example, if you share your limited food with another, who is either unwilling or unable to share their food with you. Yet cooperation exists in force, in the colonies of bees or ants, in shoals of fish, in business (through sponsorships and investment for example), in politics through alliances and treaties and so on.

Initially introduced in 1950 by Merill Flood and Melvin Dresher, The Prisoner's Dilemma is a game that is often used as a model for the emergence of cooperation[1]. The following is an attempt to explain how the Prisoner's Dilemma game can be used to explain the behaviour of cooperation in terms of self-interest, as opposed to in terms of moral or spiritual reasons. It forms the basis of our model of cooperation, as will be explained later.

In Prisoner's Dilemma, two robbers are caught and placed in separate interrogation rooms. The police have enough evidence to imprison both for 1 year, (say, they found a gun) but not enough evidence to convict for a maximum of 10 years. The police require a confession, so they tell each of the robbers that if they rat out the other (or *Defect*), they will get immunity. Now each robber can choose to either *Defect* to the police and rat out the other or *Cooperate* with their partner and not tell the police anything.

If both Defect, then the police didn't need the confession of one to convict the other, so neither get immunity. The judge would take into account that they confessed and gives both 8 years in jail.

If both Cooperate, then the police have no new evidence and can only convict for the evidence they have. Both get 1 year in jail.

If one Defects while the other Cooperates, then the defector gets immunity (0 years in jail!) while the other gets the maximum sentence of 10 years.

Imagine you're one of the robbers in the cell. You know that if the other Defects, you should Defect too, otherwise you'll get 10 years. You also know that if the other Cooperates then you should, again, defect, so that you get immunity. Regardless of what the other does you conclude that you should defect.

In this example, both defect and both get 8 years in jail each.

Now consider being put in this scenario multiple times. This is called the Iterated Prisoner's Dilemma. Eventually you notice that if you and your partner make the same decision you did the first time you will always end up with 8 years in jail. Collectively, this is 16 years in jail. However, if both choose to cooperate, then each get 1 year in jail. Even though you'd rather go free than be in jail, you'd rather be in jail for a year rather than 8. So, over time (over repeated plays) cooperating can be the more beneficial behaviour for oneself and for others.

III. THE MODEL

In this section we explain the details of how we modelled a society filled with individuals, how we modelled the behaviour of cooperation and defection, and how we modelled the evolution of this society through time. The model described here is based largely on the model described by Nowak and May.[2]

Overview

To model the society of individuals we first observe that the individuals are not necessarily individual people. For example the model explained here can be interpreted as a model of a city where each individual unit represents some collective organization acting in its own self-interest. We are free to think of an individual in our model as a person trying to acquire food for themselves or a company trying to acquire profits for itself. Thus we make the simplifying assumption that the individual units are fixed in space and do not move. We further make the simplifying assumption that individual units do not die, so we don't take into account deaths of individuals. We choose to model the society as a square grid or lattice where each cell is occupied by an individual, with no cells being empty.

To model cooperating and defecting behaviours we can use the Prisoner's Dilemma, for reasons described earlier. To represent interaction between two individuals in the society, we define their interaction as playing a game of Prisoner's Dilemma. The outcome of this interaction depends on whether each individual chose to cooperate or to defect; that is, there are only two possible options for any individual during an interaction: to cooperate or to defect. The outcome of this interaction is a score for each player. The question now is how does the individual decide what action to take? Our goal is to study cooperation in a generic society of individuals so we choose to assume that the individual behaves in the simplest possible way and ignores any past experience and does not construct any elaborate plans for itself. Specifically, each individual either always chooses to cooperate or always chooses to defect. Individuals who always cooperate are labelled *cooperators* and those who always defect are labelled *defectors*. Thus our measurement of the cooperation level in the society is a population count of how many cooperators there are and how many defectors there are.

To model the evolution of the society through time, we suppose each individual interacts with each of its neighbours. The action the individual takes is determined by the individual's label. Each individual then accumulates a total score representing the interactions the individual had with all its neighbours, as determined by the Prisoner's Dilemma game. At the end of this interaction period, an individual looks at its neighbours and identifies the neighbour with the highest score. It then adopts that neighbour's strategy, as if by mimicry. In this way cooperators can become defectors and vice versa.

Parameters

The model describes the evolution of the society through time, and from this, gives a solution of the population of cooperators over time, and a solution of the 'society' over time.

There is a single main parameter to the model. This paramter is the advantage defectors have over cooperators, called b. By making the assumptions that the individuals have no memory and are not clever enough to form elaborate plans, our results will largely be due to this intrinsic advantage defectors might have over cooperators. The b value is therefore a characterization of the environment, not the individuals. For example, if an environment has abundant resources, then the advantage defectors have over cooperators will be small, since a cooperator who has been ripped off by a defector can make up for it from the abundance of the environment.

The initial distribution of individuals, called W, or, the *World,* can be thought of as a parameter, as well as an initial value. The World characterizes what kind of individuals the society is made up of, their quantities in the society, and their placement in the society. For example, one might imagine that early humans were predominantly self-interested and had no conception of the strategy of cooperation. Suppose one day the strategy of cooperation is discovered (or evolved) by a single individual. Then the initial world could be a matrix filled with cells representing defectors with a single cell representing a cooperator. This initial distribution combined with the b value will determine whether or not this strategy will spread through mimicry. The way we choose the initial world can represent many different situations.

Formalism

We now explain how the ideas explained above are made precise and deterministic.

To formalize the interaction of two individuals we use the notation and methods followed by Broom. The possible outcomes of a round of Prisoner's Dilemma are represented by a payoff matrix. That is, the 1 year in jail payoff for both robbers cooperating becomes the Reward $R = -1$, the 8 years for both robbers defecting is the Punishment $P = -8$, the immunity the defector gets is the temptation $T = 0$ and the robber who cooperates while the other defects gets the Sucker's Payoff $S = -10$.

If a robber cooperates, we denote it by C. If a robber defects, we denote it by D. So CC means both cooperate, CD, means one cooperated and the other defected, etc. The payoff matrix is therefore this matrix

$$\begin{bmatrix} CC & CD \\ DC & DD \end{bmatrix} = \begin{bmatrix} R & S \\ T & P \end{bmatrix} = \begin{bmatrix} -1 & -10 \\ 0 & -8 \end{bmatrix}$$

In general you just need $T > R > P > S$. For use in our model, we inject our parameter b as the temptation T and we set the payoff matrix to:

$$\begin{bmatrix} R & S \\ T & P \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ b & \epsilon \end{bmatrix}$$

So we require $b > 1$. The P value is some arbitrarily small number ϵ and in our model we set $\epsilon = 0$ without affecting any results. So in this way, given any two interacting individuals, we can determine what each individual "gets" as a result of their interaction. This result is the score.

To represent the world W we construct a square matrix of dimension M with elements of either 0 or 1. The *defector* will be represented by a 0 and 1 will represent a *cooperator*. This initial world (W_0) can be constructed in different ways either randomly or with a small seed population. To illustrate, a world with 9 individuals with a single defector surrounded by cooperators is represented as

$$W = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

To evolve the World through time (generation by generation) we simulate each individual interacting with its neighbours. The simulation proceeds as follows:

1. Pick a cell. The cell plays the game with each of its 8 neighbours. The payoff the cell gets is determined by the payoff matrix. After playing with each neighbour one by one, the cell has a total payoff. Do this for each cell.

2. Pick a cell. Out of itself and all of its neighbours, find the one that has the best payoff. If the one with the best payoff is a defector, the cell becomes a defector in the next round. If it is a cooperator, the cell becomes a cooperator in the next round.

Specifically, we iterate over each cell of the World matrix and take a 3x3 local matrices representing the individual's neighbours.

$$World_0 = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,m} \end{bmatrix} \xrightarrow{\text{Take an element and its neighbours}} Local = \begin{bmatrix} a_{h-1,j-1} & a_{h-1,j} & a_{h-1,j+1} \\ a_{h,j-1} & a_{h,j} & a_{h,j+1} \\ a_{h+1,j-1} & a_{h+1,j} & a_{h+1,j+1} \end{bmatrix}$$

The element a_{hj} plays Prisoner's Dilemma with each of its eight neighbours. The element a_{hj} is either a cooperator or a defector, and the neighbour it is playing with is either a cooperator or a defector. The interaction score of a_{hj} is determined by the payoff matrix:

$$\begin{bmatrix} CC & CD \\ DC & DD \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ b & 0 \end{bmatrix}$$

Elements on the edge of the world have neighbours missing and play with fewer neighbours. The element a_{hj}'s total score is just the sum of the scores it got from each of its interactions with its neighbours. Therefore for each element we can assign a total score s. Denote the total score of a_{hj} after playing with all of its neighbours s_{hj}. We construct a matrix S containing all the total scores.

$$W_0 = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,m} \end{bmatrix} \to S = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,m} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ s_{m,1} & s_{m,2} & \cdots & s_{m,m} \end{bmatrix}$$

Next, for each a_{hj}, we take the local matrix containing its neighbours and find the indices of the cell containing the highest score.

$$S_{\text{local}} = \begin{bmatrix} s_{h-1,j-1} & s_{h-1,j} & s_{h-1,j+1} \\ s_{h,j-1} & s_{h,j} & s_{h,j+1} \\ s_{h+1,j-1} & s_{h+1,j} & s_{h+1,j+1} \end{bmatrix} \to (m,n)$$

Suppose $s_{h,j-1}$ was the highest score. Then $(m,n) = (h, j-1)$. We then consult the local world at this index and determine what is occupying that cell. If a_{mn} is a cooperator, a_{hj} 'converts' to being a cooperator and we reassign a_{hj}'s label to be a cooperator. If a_{mn} is a defector, then a_{hj} is relabeled a defector. If a_{hj} is already one of these, then the label doesn't change. This process is meant to represent mimicking the most fit individual. Now, for each individual a_{hj} we can assign a new state a^*_{hj}. From these new states we construct a new world:

$$W_0 = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,m} \end{bmatrix} \to S = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,m} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ s_{m,1} & s_{m,2} & \cdots & s_{m,m} \end{bmatrix} \to W_1 = \begin{bmatrix} a^*_{1,1} & a^*_{1,2} & \cdots & a^*_{1,m} \\ a^*_{2,1} & a^*_{2,2} & \cdots & a^*_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a^*_{m,1} & a^*_{m,2} & \cdots & a^*_{m,m} \end{bmatrix}$$

This whole process then repeats using W_1 as the world. Storing each of the worlds after each step, we can count the number of elements representing cooperators and defectors and plot these through time. We can also display the world and see the spatial distribution of both types of individuals. As a movie, these sequences of worlds can be very entrancing to watch.

We implemented this simulation in Matlab. The model was implemented in one file, and another file was used to list initial worlds, world size, the number of generations to simulate and b values. This file was also used to run the simulation.

Notice that individuals on the edges of the world will have fewer neighbours to play with. Though having fewer neighbours means less opportunity to accumulate a high score, it also means less opportunity to be beaten by someone. If one is worried of the effect of this, a 'wraparound' world can be made so that all elements have the same number of neighbours, but this modification produced no major difference to our results. This conclusion is also reached by Nowak and May on the same issue.[2]

IV. ANALYSIS AND METHODS

To analyze our model we designed simulations that generated an initial world, and then simulated that world for different values of b. We varied b between 1.1 and 2.0. The worlds were generated with cooperators and defectors randomly placed throughout using a random number generator. That same initially generated random world was then simulated for different values of b. These worlds were 200x200 square grids and were simulated for about 200 generations. The code for these experiments can be found in the appendix. We will describe how these experiments work and then look at the results.

Random World Generation

For an initial world filled with 50% cooperators and 50% defectors uniformly distributed, the following Matlab command was used:

```
World = randi([DEFECTOR COOPERATOR], 200, 200);
```

For a world randomly filled with 10% cooperators, or 20%, or 1%, or some arbitrary percentage, we used a different procedure. We first generate a world filled with defectors. We then take the desired proportion (example: 0.1 for 10%) p and calculate the number of cooperators we need in the world to meet that proportion. Given the side length of the world s, the number of individuals that can occupy the world is s^2. The number of cooperators required is then calculated as:

$$n = \text{round}(ps^2)$$

Where we use round() to make sure the number calculated is an integer. Then, we initialize a counter $c = 0$ to keep track of how many cooperators we have placed in the world. Then, in a while loop we place cooperators according to the following procedure:

1. While $c < n$:
   - (a) Generate a random x coordinate and a random y coordinate.
   - (b) If the individual in the World at (x, y) is a defector, replace it with a cooperator and increase c by 1.

This procedure is repeated until $c = n$, at which point we know we have placed n cooperators in the world, meeting our desired proportion of p.

Coloring

To view the World we use matlabs image() function, which takes a matrix of data and maps it to colors. For example, all cooperators would be colored blue and all defectors would be colored red.

In order to see at a glance the amount of change that was happening from generation to generation, we colored cells based on if they had changed from the last round. For example cooperators who had been defectors in the last round would be colored orange and defectors who had been cooperators in the last round would be colored yellow. Thus if we look at a world and see very little yellow and orange, we know the world is relatively stable and that not much change is happening from generation to generation. If however we see a world rich with yellow and orange, we know the world is in flux. We can also tell at a glance whether it is the cooperators or the defectors who are dominating a world.

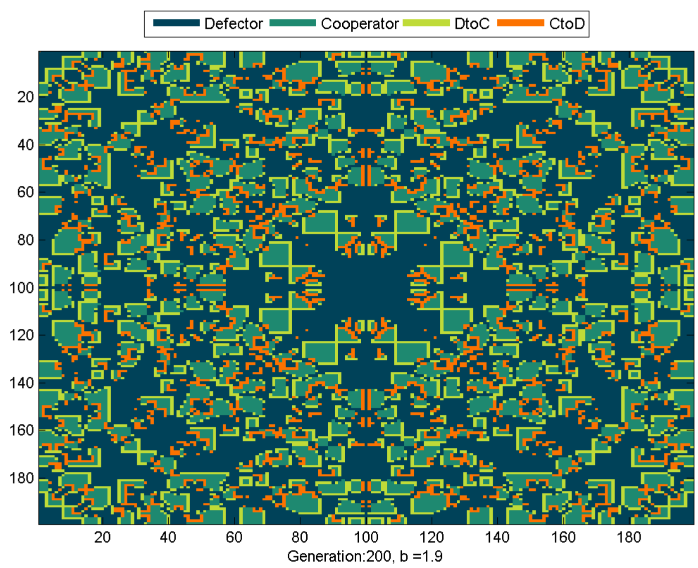


Figure 1: Example of World Coloring

To compute the colors, we calculate a 'colorworld' C_W as a function of an initial world W_i and a final world W_f. We use the entry 2 to denote cooperators who were defectors in the previous round and we use the entry 3 to denote defectors who were cooperators in the previous round. The labelling procedure is as follows:

1. Initialize a matrix C_W and set it equal to the final world. That is, set $C_W = W_f$.
2. Calculate a difference matrix $D_W = W_f - W_i$. That is, subtract component-wise and place the result in D_W. We can do this because we represent defectors by 0 and cooperators by 1.
3. For each cell a_{ij} in C_W, check what the corresponding value x_{ij} is in D_W. We can do this because C_W, D_W, W_f, and W_i are all the same size.
4. If $x_{ij} = 1 - 0 = 1$, then the cell was a defector and became a cooperator. So set $a_{ij} = 2$
5. If $x_{ij} = 0 - 1 = -1$, then the cell was a cooperator and became a defector. So set $a_{ij} = 3$

6. If $x_{ij} = 0$, don't change a_{ij}.

At the end of this procedure, we call image(C_W) which displays the desired result.

Results of the Random World Experiments

Transition from Spatial Stability to Spatial Chaos

Our results of the simulations of the random worlds showed that for values of $b < 1.8$, the world evolves to a stable point where the spatial distributions of population stop changing, and for $b > 1.8$, chaos ensues and the population distributions bubble and roil as the cooperators and defectors continually change strategies.

An example of this is the uniformly distributed initial world (50% cooperators and 50% defectors) for values of $b = 1.799$ and $b = 1.8$. Below are the final states and population graphs over time for both b values and the same initial world.

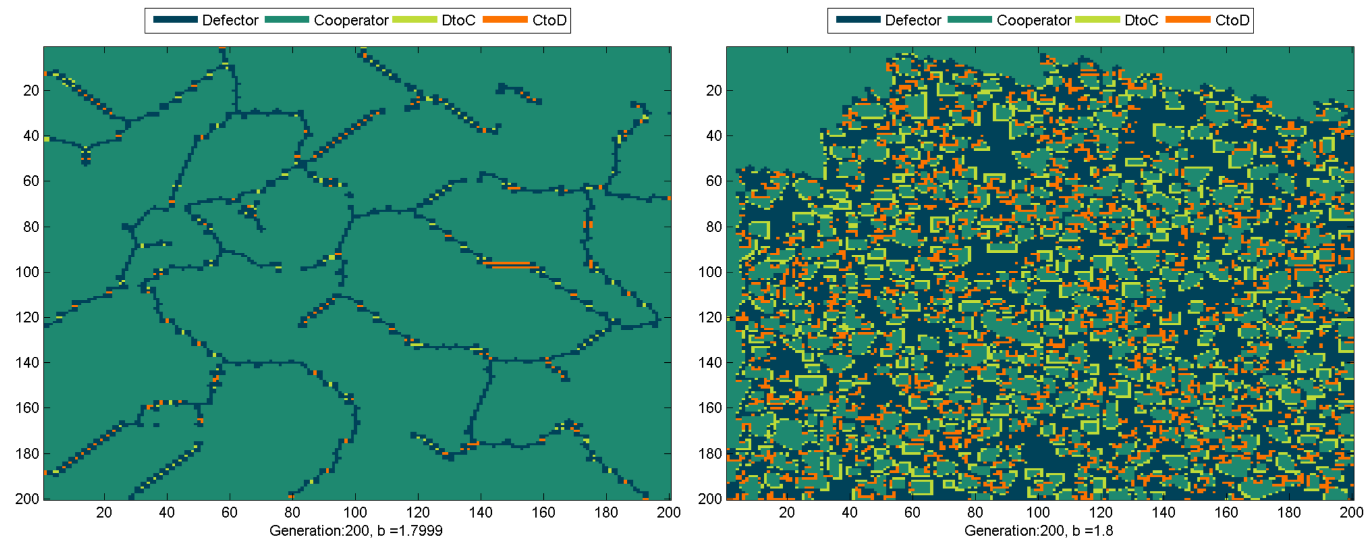


Figure 2: Final States for $b = 1.799$ and $b = 1.8$

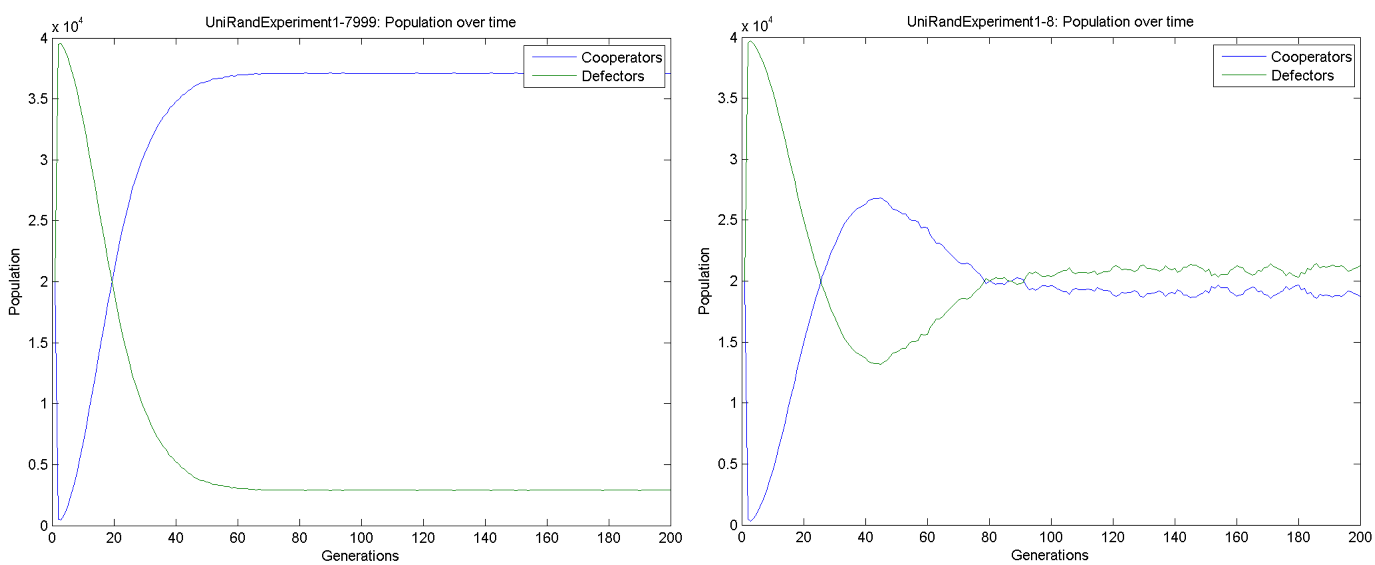


Figure 3: Population Graphs for $b = 1.799$ and $b = 1.8$

Recall that both worlds start with an even split of cooperators and defectors. Notice that in both population graphs, there is a sudden drop in the level of cooperators to almost 0 within the first one or two generations. This is because initially, most cooperators are isolated and surrounded by defectors. These cooperators will be wiped out or converted within a generation. The cooperators that survive are the few that were lucky enough to be close to other cooperators. These tiny clusters then grow and the population of cooperators rises, as shown in both graphs.

In the graph for $b = 1.799$, the level of cooperators steadily grows and reaches a very high and very steady level of around 3.6×10^4 individuals. We know it is steady (other than looking at the graph) because the final state for $b = 1.799$ shows only scattered blips of yellow and orange (representing changing individuals) and we see that the cooperators form large communities seperated by thin strands of defector communities.

In the graph for $b = 1.8$, we see the growth in the level of cooperators peaking at around generation 40 at 2.6×10^4 individuals and then dropping back down to about the 50/50 split. Moreover we see that the level of cooperators oscillates closely but wildly about 2×10^4 individuals. We see these chaotic and noisy fluctuations in the population graph are reflected by an abundance of orange and yellow converters in the final state image. If you were to watch a video of the $b = 1.8$ simulation you would see a constant back and forth roiling of cooperator bubbles becoming defector bubbles and vice versa. In the population graph for $b = 1.8$ we also notice that between generations 100 to 200, there seems to be a very gradual drop in the level of cooperators. We can confirm this by running a longer simulation for $b = 1.8$. Below is the population graph of the same simulation but for 500 generations.

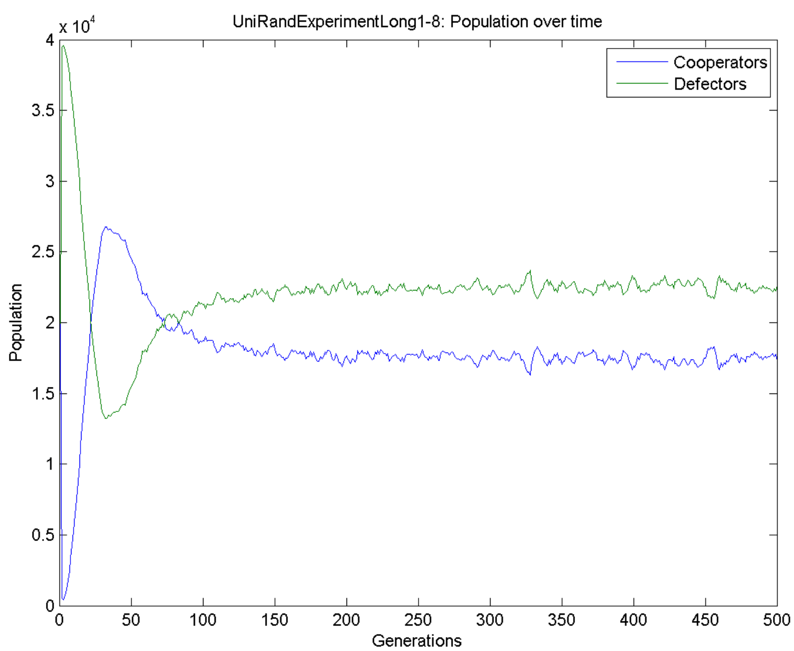


Figure 4: Long Run of $b = 1.8$

We see that this dropping trend in cooperator population does continue, but that it is much less pronounced after around generation 250. We also see that the noise shows no sign of dissipating, and even see some parts where the noise becomes more pronounced, like at about generation 325 and 450.

Regular Oscillations

At $b = 1.7$ of a 50/50 initial random World we observe periodic and stable oscillations in the steady state value of the cooperator population, which is about 3.5×10^4 individuals. Based on the graphs, the period of these oscillations is about 2 generations, and their amplitude is about $0.05 \times 10^4 = 500$ individuals. At $b = 1.7$ we are in the $b < 1.8$ zone, that is we are in the spatially stable zone, and we see large communities of cooperators separated by thin strands of defectors as before. The thin strands are bounded however by long straight lines of communities that rapidly oscillate from defector to cooperator. These lines of converter communities are likely the source of the oscillations in the population graph. The figure below shows the oscillations in simulation for $b = 1.7$

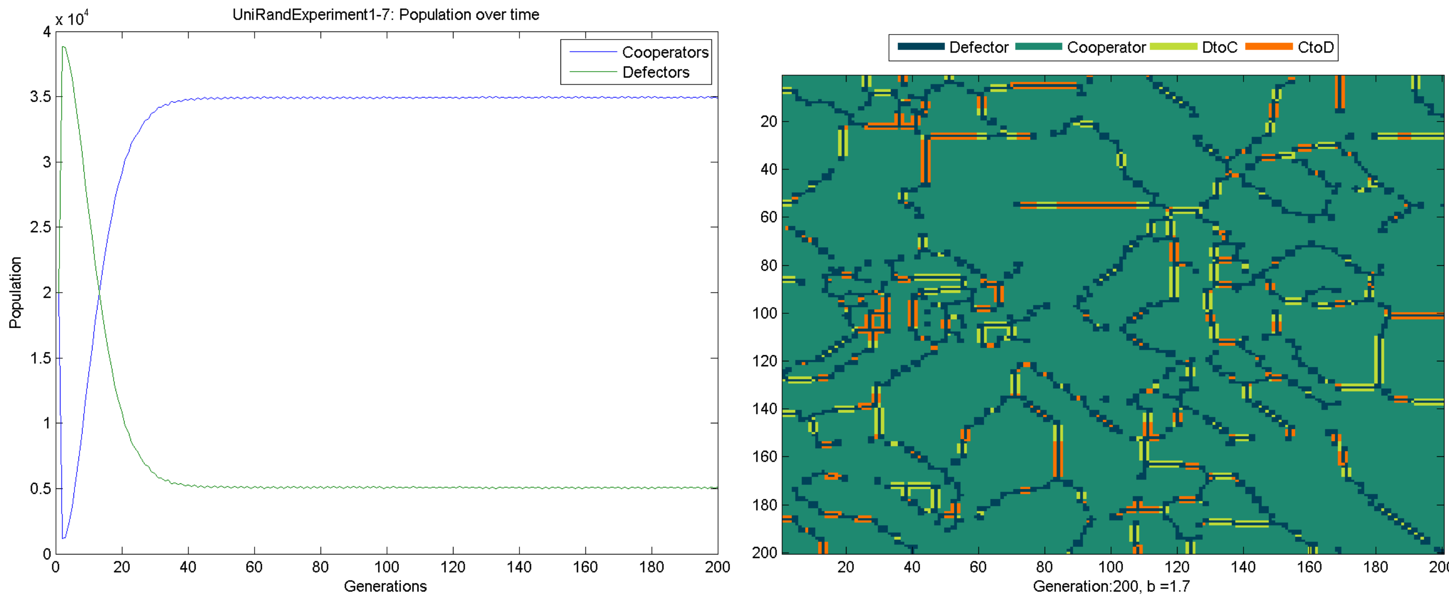


Figure 5: Regular Oscillations at $b = 1.7$

These oscillations can be seen more obviously in the following video of the simulation:

www.youtube.com/watch?v=r-BEvsvViQU

Varying Proportions

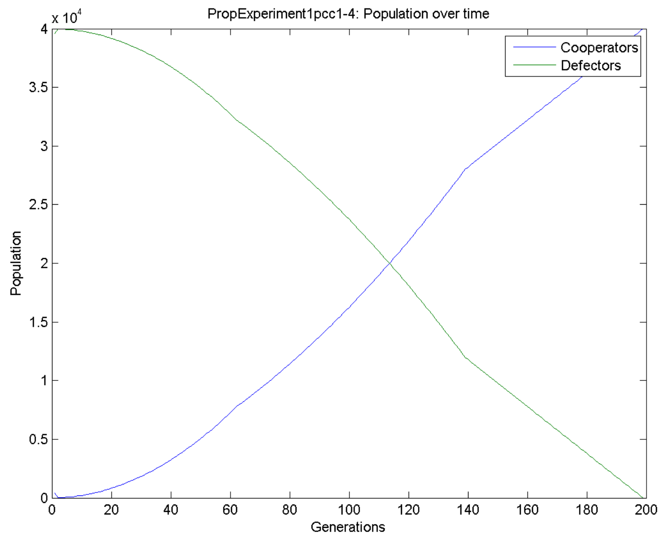
We conducted experiments over varying initial proportions of cooperators and found similar results that we will only summarize here for brevity. In particular we examined initial proportions of 1%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, and 99% for varying values of b.

The different proportions did not affect our observation of spatial stability for $b < 1.8$ and spatial chaos for $b > 1.8$.

Regular, periodic oscillations like described earlier were observed for initial proportions of 30% and higher for $b = 1.7$. No regular, periodic oscillations were seen for an initial proportion of 20%, for any value of b, though flashing converter communities could be seen in the simulation video for $b = 1.7$.

For an initial proportion of 10%, regular periodic oscillations were seen at $b = 1.3$, as well as a combined effect of oscillations and growth for $b = 1.7$. This can be understood from the video: one can see a line of a flashing converter community starting in the middle of the world and growing longer each generation until it reaches the edge.

In regards to the question of the emergence of cooperation assuming a predominantly defector society, we take a closer look at an initial cooperator proportion of 1%. Our results show that a single community of cooperators grows and wipes out all defectors within 200 generations when $b < 1.4$. When $b > 1.4$, the tiny community of cooperators is almost instantly wiped out. This is shown in the following graphs of population:



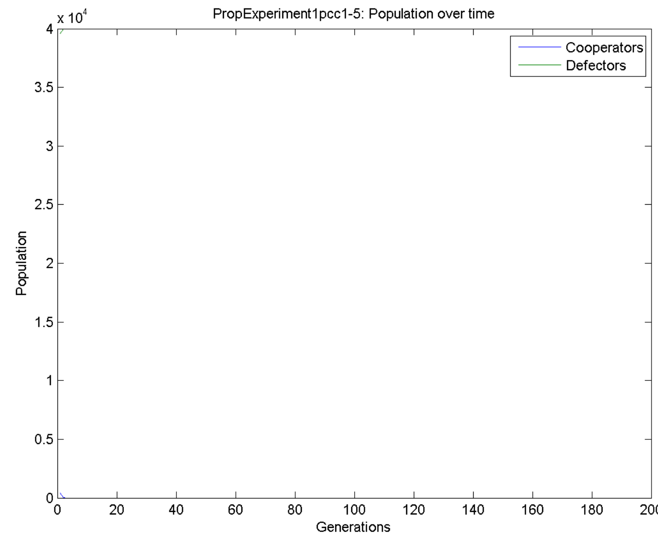


Figure 6: Population over time: initial proportion of 1% for $b = 1.4$ and $b = 1.5$

If the graph for $b = 1.5$ looks empty, it is not. In fact there is a very tiny line indicating an initial small population of cooperators going to 0 within 2 generations. For $b = 1.4$, we see a tiny population of cooperators growing to wipe out all defectors.

As a final observation, we note that in our experiments for $b \geq 2$, the population of defectors always wipes out the population of cooperators when the proportion of cooperators is less than 50%. For proportions larger than 50% at $b = 2$, medium sized and isolated communities of cooperators form in a sea of defectors. There are no small communities of cooperators at these high b values since cooperators that are surrounded by defectors will quickly lose and decide to become a defector.

IV.1. Analysis of Model's Dependance on $b = T$

In order to gain some understanding of why a sharp transition exists at $b = T = 1.8$, we analyze the behaviour of a simple world for various values of b. By keeping the World W the same across different values of T, we can isolate for the influence of T on the World's evolution. The discrete nature of the payoff matrix will result in a discrete transition point between dynamical regimes. To determine interesting values for T, we begin by eliminating certain values analytically by looking at local behaviour. We can show that some values of T will always lead clusters of defectors or cooperators to uncapped growth. It should be

Let A denote an initial 9 block of defectors surrounded by cooperators and B denote an initial 9 block of cooperators sorrounded by defectors.

$$A = \begin{bmatrix} C & C & C & C & C \\ C & D & D & D & C \\ C & D & D & D & C \\ C & D & D & D & C \\ C & C & C & C & C \end{bmatrix} \qquad B = \begin{bmatrix} D & D & D & D & D \\ D & C & C & C & D \\ D & C & C & C & D \\ D & C & C & C & D \\ D & D & D & D & D \end{bmatrix}$$

Suppose in matrix A, $T > 2$. The middle defector (which will have the lowest score) can't change strategy unless one of its neighbours switches to cooperator. The lowest score for a defector on the outside of that center is when that defector plays a maximum number of defectors. Calculating the score of each interaction and adding up the results using the model's payoff matrix:

$$s_D > 3T + 5P = 6$$

The highest score possible that this defector can 'see' and therefore mimic is the score a neighbouring cooperator gets when it plays a minimum number of defectors.

$$s_C = 6C + 2S = 6$$

This worst case scenario is still in favour of the defectors. As it turns out, when $T > 2$ defectors will take over no matter initial conditions.

Now for the same matrix A, we will show that for $T > 1.8$, will grow on the corners. For the worst case of defectors, we have

$$s_D > 3T + 5P = 5.4$$

While its cooperator neighbour with the highest score will have

$$s_C = 6C + 2S = 6$$

and its defector neighbour with the highest score (which occurs at the corner) will have

$$s_D > 5C + 3P = 9$$

Since $s_D > s_C$, the corners will always grow. Since we still neighbour with the largest score a defector, it will continue to grow as a block until we get a 5x5 cluster of D. At that point, the defector in the center of an edge will still only get $s_D > 5.4$ whereas it will no longer have a corner defector of $s_C = 6$ as a neighbour. At this point, our low score defector will switch strategies to cooperator. This brings about chaos since at the corners the defectors continue to grow while in the middle of the edges we get defectors switching to cooperators. This makes further small stable clusters of defectors grow until they either hit a group of defectors or hit the unstable size of 5x5.

V. DISCUSSION

What do our model's results say about cooperation levels in society? What does our model say about the conditions needed for cooperation as a strategy to emerge?

Before interpreting our results, we restate some basic assumptions our model makes about reality, as these assumptions may not be true. In our model, an individual's sole goal is to survive and to increase the amount of resources one has. Our model assumes that individuals forget or choose not to act on their past experiences, that individuals have very short range predictive power (can only think about one generation into the future), and that individuals have no theory of mind. Further, our model assumes that the only reason for cooperation is for the sake of self-growth. This is is significant because our model also assumes that the individual's sole goal is to increase the amount of resources on has. Even if one refuses to consider any moral or spiritual reasons for cooperation, it is important to realize that the model is silent on the posssibility that individuals may choose to cooperate out of some need other than survival and economic growth. For example suppose individuals have a need to contribute to the well-being of others. The existence of this need plausible as a mechanism to explain why parents take care of their children. Or alternately, suppose individuals have a need for safety and to protect themselves. The existence of the need for safety is suggested (for example) by the behaviour of lions or wolves to form packs. The point here is to see that the strategy of cooperation is a strategy that can be chosen to meet the need for safety or the need to contribute to the well-being of one's children. To emphasize: our model assumes cooperation only emerges as a strategy to meet the need for survival and does not consider other self-interested needs that may lead to cooperation. Thus there may be other ways to explain the emergence of cooperation that still focus on self-interest and not on moral or spiritual reasons that our model is silent about.

Our model predicts that for $b < 1.8$, large and stable communities of cooperators will form separated by thin lines of defectors and that for $b > 1.8$, a chaotic society will with constantly shifting communities of defectors and cooperators. This result suggests to use that the stability of the strategy of cooperation is a strategy largely dependent on the environment, and not on the individuals. If, for example we were to look at a small tribe of cooperative individuals and ask ourselves 'what is the cause of this cooperation?' our model suggests that we study the environment these individuals find themselves in and further, our model suggests to us that the individuals in this tribe that we are studying are in an environment that is abundant with resources, or an environment that harshly punishes those who defect. That is, we are inclined to think that something about the environment causes the advantage defection has over cooperation to be very small. This is important because it leads us away from asking, 'what is happening in the individual's mind that is leading to this cooperation?' Thus we believe our model suggests the causes of cooperation are largely environment based.

Further, because of the sudden shift from a stable society to one that is constantly changing, we interpret that as meaning that the balance is very fragile and a change in environment that only slightly affects the advantage of defectors over cooperators, if at the crucial value, can cause very dramatic changes in a society. Suppose we have a society hovering at about $b = 1.79$. We have already seen that this society will have large and stable communities of cooperators. However if the environment changes slightly, say for example there was minor drought that affected the abundance of food, then this small disturbance might be enough to push the b value from 1.79 to 1.8, the crucial value. This tiny disturbance then has the result of completely disturbing the society and the next few generations will be filled with a chaotic reshuffling of communities until the b value drops again. At the same time however, if b were 1.2 for example, then even a large disturbance that pushes b from 1.2 to 1.78 would have very little effect on the society's communities. Thus our model suggests that for a society to be spatially stable and resilient to disturbances, the advantage defectors have over cooperators should be kept far away from the crucial value that the model predicts.

Our model predicts the existence of regular periodic oscillations in the $b < 1.8$ regime, and that these oscillations arise from communities of individuals that continually switch from defector to cooperator. Though we believe the existence of such communities to be unrealistic, we interpret this to mean that a characteristic of $b < 1.8$ societies are individuals who will attempt the strategy of defection before quickly reverting to cooperation. Because we see thin strands of defectors, these 'flippy' individuals are separate from the small but stable groups of individuals that are committed to defecting.

On the matter of the emergence of cooperation we can look at our results for when the initial proportion of cooperators was 1%. We interpret the question of 'how does cooperation emerge in a society' in terms of our model: how can a society mostly composed of defectors allow for the growth of communities of cooperators? Our results show that for $b \leq 1.4$, a 1% initial distribution of cooperators will grow and in fact wipe out the population of defectors. We also see that for $b > 1.4$, the population of cooperators is almost instantly eradicated. Thus we suggest the following answer to the question: Cooperation emerges in a society when a very small percentage of individuals choose to try the strategy of cooperation and when the environment is such that the advantage that the majority defectors have over these minority cooperators is below some sharp threshold value. In our experiments the threshold was 1.4. A few individuals may choose to 'try' cooperating either by accident or evolution or intelligence, the reason can be anything. For this cooperative behaviour to spread to others, we only requre that the environment be such that the advantage defectors have is limited. So even if an advantage for defectors exists, cooperation can emerge and can overtake the behaviour of defecting.

Conclusion

Overall we believe our results shed light on the question by strongly suggesting that when we study a society and its cooperation levels, if we want to understand the causes of the cooperation, we look at the environment the individuals find themselves in and try to characterize or measure the advantage defectors have over cooperators.

Further areas of study that might shed light on the resilience of a society to change would be to simulate a world where the parameter b is a function of time $b(t)$. An example of this would be to simulate a world as b moves from the stable regime to the chaotic regime.

In this report we described methods for modelling a society, cooperation in the society, and a way to evolve the society through time. The primary parameter of this model was the advantage defectors have over cooperators, or the b value. We then implemented this model and conducted experiments to understand how the level of cooperation can change over time and how a cooperation can emerge from a society composed of defectors. We also suggested that cooperation emerges when the environment is such that the advantage defectors have is finite and not too big of an advantage. This report also gives an overview of the results found, and attempts to analyse the reasons for the existence of a sharp transition zone from $b < 1.8$ to $b > 1.8$.

VI. VIDEOS

1. Fractal Experiment: A single defector in the middle of a society of cooperators.
   www.youtube.com/watch?v=qlajhvGfHb4

2. Flashing Lights Experiment: Evidence of regular periodic oscillations
   www.youtube.com/watch?v=r-BEvsvViQU

VII. APPENDIX: MATLAB PROGRAMS

All the code is available online and can be accessed at

github.com/mef51/PrisonersDilemmaWorld

The complete folder with runnable code can be downloaded at

github.com/mef51/PrisonersDilemmaWorld/archive/master.zip

Below is included an implementation of the model described in this report and the scripts used to generate the results summarized.

VII.1. Model

prisonerworld.m

```
% Simulates a PrisonersDilemmaWorld and saves a population graph, a video of the world, and
% the last frame of the video as an image.
% Pass in:
% simName - a string that identifies the experiment you're running
% World - a square matrix filled with DEFECTORs and COOPERATORS. You pick how you want the world.
% generations - a number, the number of generations to simulate
% b - the Temptation. This is the parameter from Nowak & May's paper.
function prisonerworld(simName, World, generations, b)
    % Prisoner's Dilemma Payoffs: T > R > P > S
    T = b;  % Temptation. may and nowak vary this parameter to get their results
    R = 1;  % Reward.
    P = 0;   % Punishment. technically this is some arbitrarily small number > 0
    S = 0;   % Sucker's Payoff.
    DEFECTOR = 0; % a constant, don't change this lel
    COOPERATOR = 1; % don't change this either lel

    worldSize = size(World, 1); % we're assuming World is a square matrix here

    if ~exist('results', 'dir')
        mkdir('results'); % dis is where we dump the results
    end

    % this matrix holds the total score a cell obtains after playing with all its neighbours
    score = updateWorldScores();
    populationCount = zeros(generations, 2); % Keeps track of the cooperator/defectors for each iteration

    hmo = HeatMap(World);
    fig = plot(hmo);

    simVideo(1) = getframe(fig);
    close all hidden; % HeatMaps have hidden handles
    for step = 1:generations
        updatePopCount(step);
        % let the new world begin
        World = updateWorld(World, score);
        score = updateWorldScores();
        hmo = HeatMap(World);
        fig = plot(hmo);
        if step == generations
            print(strcat('results/', simName,'_FinalState'),'-dpng');
        end
        simVideo(step+1) = getframe(fig);
        close all hidden;
    end

    %figure
    movie(simVideo, 1);
    movie2avi(simVideo, strcat('results/', simName,'_World'), 'compression', 'None');
    plot(1:generations, populationCount(:,1), 1:generations, populationCount(:,2));
    xlabel('Generations');
    ylabel('Population');
    title(strcat(simName, ': Population over time'));
    legend('Cooperators', 'Defectors');
    print(strcat('results/', simName,'_Population'),'-dpng');

    close all; % finished

    %grabs data for the number of each population at every step
    function updatePopCount(step)
        numCooperators = sum(World(:) == COOPERATOR); % this is such a cool trick haha
        populationCount(step, 1) = numCooperators;
        populationCount(step, 2) = (worldSize*worldSize) - numCooperators;
    end
```

```
    % for each cell, calculate the score it gains from interacting with its neighbours
    function newScores = updateWorldScores()
        newScores = zeros(worldSize, worldSize);
        for m = 1:worldSize
            for n = 1:worldSize
                newScores(m, n) = getScoreFromInteractions(m, n);
            end
        end
    end

    % update the world for the next generation
    function newWorld = updateWorld(oldWorld, oldScores)
        newWorld = zeros(worldSize, worldSize);
        for m = 1:worldSize
            for n = 1:worldSize
                newWorld(m, n) = findRoleModel(m, n, oldWorld, oldScores);
            end
        end
    end

    % find which neighbour the cell at (m, n) picks as a role model.
    % a cell looks at its neighbours and picks the neighbour with the
    % highest score as its role model.
    function roleModel = findRoleModel(m, n, oldWorld, oldScores)
        maxScore = 0;
        roleModel = oldWorld(m, n);
        for i = -1:1
            for j = -1:1
                if m+i <= worldSize && m+i >= 1 && n+j <= worldSize && n+j >= 1
                    s = oldScores(m+i, n+j);
                    r = oldWorld(m+i, n+j);
                    if maxScore < s
                        maxScore = s;
                        roleModel = r;
                    end
                end
            end
        end
    end

    % take the cell at (m,n) and make it interact with all its neighbours
    % Return the score the cell accumulates after interacting with its neighbours
    % A cell has (if its not on the border) 8 neighbours and can
    % interact with itself. So there are 9 interactions.
    % m is the row #. n is the column #
    function totalScore = getScoreFromInteractions(m, n)
        s = 0;
        for i = -1:1
            for j = -1:1
                % make sure our indices are within the dimensions of the
                % board. Otherwise just ignore that interaction. This means
                % that cells on the border will have fewer neighbours to
                % play with. If we want, we can 'wraparound' our indices to
                % represent a circular pacman-like world.
                if m+i <= worldSize && m+i >= 1 && n+j <= worldSize && n+j >= 1
                    s = s + game(m, n, m+i, n+j);
                end
            end
        end
       totalScore = s;
    end

    % Given two cells, return what the (m,n) cell scores after playing
    % Prisoner's Dilemma with cell (i,j).
    % 0 denotes a Defector
    % 1 denotes a Cooperator
    function result = game(m, n, i, j)
        result = -1;
        if World(m, n) == COOPERATOR
            if World(i, j) == DEFECTOR
                result = S;
            else % World(i, j) == COOPERATOR
                result = R;
            end
        end
        if World(m, n) == DEFECTOR
            if World(i, j) == DEFECTOR
                result = P;
            else % World(i, j) == COOPERATOR
                result = T;
            end
        end
    end
end
```

VII.2. Experiments

ProportionExperiments.m

```matlab
DEFECTOR = 0; % a constant, don't change this lel
COOPERATOR = 1; % don't change this either lel

worldSize = 200;
generations = 200;
b = 1.1:0.1:2.0;
proportionCoops = union(0.1:0.1:0.9, [0.01, 0.99]);

for i = 1:length(proportionCoops)
    World = ones(worldSize, worldSize)*DEFECTOR;
    proportion = proportionCoops(i);
    numCoops = round(proportion*(worldSize*worldSize)); % the # of coops we want in the world
    coopsSoFar = 0;
    while (coopsSoFar < numCoops)
        m = randi(worldSize);
        n = randi(worldSize);
        if World(m,n) == DEFECTOR % make sure we don't replace a cooperator we placed earlier
            World(m,n) = COOPERATOR;
            coopsSoFar = coopsSoFar + 1;
        end
    end

    for j = 1:length(b)
        temptation = b(j);
        suffix = strrep(num2str(temptation), '.', '-');
        percent = num2str(proportion*100);
        simName = strcat('PropExperiment', percent, 'pcc', suffix);
        prisonerworld(simName, World, generations, b(j));
    end
end
```

UniformRandomExperiments.m

```matlab
% Simulations of a random world sweeping b = 1.1 to b = 2.0
% At b = 2.1 The defectors wipe out the cooperators. I assumed it was the
% same case for any b larger than that.
DEFECTOR = 0; % a constant, don't change this lel
COOPERATOR = 1; % don't change this either lel

worldSize = 200;
generations = 200;
% make a random world and use it over and over so all the initial worlds are the same
World = randi([DEFECTOR COOPERATOR], worldSize, worldSize);

% Experiment: Uniform Random distribution of defectors and cooperators
b = 1.1:0.1:2.0;
for i = 1:length(b)
    temptation = b(i);
    suffix = strrep(num2str(temptation), '.', '-');
    simName = strcat('UniRandExperiment', suffix);
    prisonerworld(simName, World, generations, temptation);
end

% I saw that from the UniformRandomExperiments, when b went from 1.7 to
% 1.8, a lot more noise showed up in the later stages of the population
% graph. I also saw that when b went from 1.8 to 1.9, the cooperators no
% longer overtook the defectors, which they had for all b smaller than 1.9.
% So the following experiments look more closely at b values between 1.7 and 1.9
b = 1.70:0.02:1.9;
for i = 1:length(b)
    temptation = b(i);
    suffix = strrep(num2str(temptation), '.', '-');
    simName = strcat('UniRandExperiment', suffix);
    prisonerworld(simName, World, generations, temptation);
end

% Ok the change is still very sudden between 1.78 and 1.8. Let's look
% more closely
b = 1.78:0.0025:1.81;
for i = 1:length(b)
    temptation = b(i);
    suffix = strrep(num2str(temptation), '.', '-');
    simName = strcat('UniRandExperiment', suffix);
    prisonerworld(simName, World, generations, temptation);
end

b = [1.7999, 1.8];
for i = 1:length(b)
    temptation = b(i);
    suffix = strrep(num2str(temptation), '.', '-');
    simName = strcat('UniRandExperiment', suffix);
    prisonerworld(simName, World, generations, temptation);
end

b = 1.8;
suffix = strrep(num2str(b), '.', '-');
simName = strcat('UniRandExperimentLong', suffix);
prisonerworld(simName, World, 500, b);
```

MiscExperiments.m

```
% Run a bunch of experiments
DEFECTOR = 0; % a constant, don't change this lel
COOPERATOR = 1; % don't change this either lel

worldSize = 199;
generations = 200;

% Experiment: 1 Defector in the middle of a World of cooperators
simName = 'DoilyExperiment';
b = 1.9;
World = ones(worldSize,worldSize)*COOPERATOR;
World(round(worldSize/2), round(worldSize/2)) = DEFECTOR;
prisonerworld(simName, World, generations, b);

% Experiment: Try to do the doily experiment with cooperators in the middle of defectors.
worldSize = 59;
generations = 40;
simName = 'CDoilyExperiment';
b = 1.8;
World = ones(worldSize,worldSize)*DEFECTOR;
World(round(worldSize/2), round(worldSize/2)) = COOPERATOR;
World(round(worldSize/2), round(worldSize/2)+1) = COOPERATOR;
World(round(worldSize/2), round(worldSize/2)-1) = COOPERATOR;
World(round(worldSize/2)+1, round(worldSize/2)) = COOPERATOR;
World(round(worldSize/2)-1, round(worldSize/2)) = COOPERATOR;
World(round(worldSize/2)+1, round(worldSize/2)+1) = COOPERATOR;
World(round(worldSize/2)+1, round(worldSize/2)-1) = COOPERATOR;
World(round(worldSize/2)-1, round(worldSize/2)+1) = COOPERATOR;
World(round(worldSize/2)-1, round(worldSize/2)-1) = COOPERATOR;
prisonerworld(simName, World, generations, b);


%Experiment: Alternating 1 and 0 throughout the matrix
simName = 'UniformAlternating';
World = ones(worldSize,worldSize)*COOPERATOR;
for m = 1:worldSize
    for n = 1:worldSize
        if mod(n, 2) == 0
            if mod(m, 2) == 1
                World(m, n) = DEFECTOR;
            end
        else
            if mod(m+1, 2) == 1
                World(m, n) = DEFECTOR;
            end
        end
    end
end
prisonerworld(simName, World, generations, b);
```


[1] Broom, M. and Rychtár, J., *Game-theoretical models in biology* (CRC Press, 2014).

[2] Nowak, M. A. and May, R. M., Nature **359**, 826 (1992).