**PROGRESS REPORT MAY 20**

**Response to Proposal Feedback:**

 While considering possible forms of implementation, I came to the conclusion that writing a model for a variable number of players with a variable number of strategies could result in an unbelievable amount of data. Therefore, I decided to simplify it down to merely a 2 player game with each player having 2 strategies. However, I am going to allow the user to decide the payoffs for each strategy choice. This allows one model to look at a very large amount of games and strategy choices.

 Furthermore, the ultimate goal is to determine how often each player chooses each strategy. The driving principle behind this is the fact that each player wants to maximize their utility at any given point in time. As a result, the player’s “learn” over time as they react to their opponent’s decision. To do this, each player starts with randomly chosen probabilities for choosing each strategy (NOTE: the sum of choosing strategy 1 and of choosing strategy 2 is equal to 1 for each player). Then, at each tick, the player will look at its current utility value and compare it to what it could’ve been had it chosen the other strategy with the opponent keeping its same strategy. If the player could’ve done better having switched, then it assigns more probability to choosing that strategy and takes some away from the one it just chose.

 In regards to analyzing the equilibrium, a plot is used to show how each of the probabilities change over time.

**Agent Behavior:**

 Currently, the only agents are 2 turtles. These turtles choose what strategies they want to play based on their probability of choosing said strategy. As I mentioned previously, the turtles “learn” over time about how often they should play each strategy. The process for this is described above. Additionally, each turtle displays its probability of choosing strategy 1 (simply for visualization, though the plot is effective too).

**System Behavior:**

 The system is merely based on the turtle’s decisions and learning capabilities. Outside of this, there is currently no further system.

**Rationale for Agent Rules:**

 There’s really only one driving principle behind the agent’s choices. That is that they wish to maximize their utility at any given time step.

**Model Output**:

 The output produces expected results for some setups, while producing unexpected results for other setups. In other words, a number of results match up with the theoretical solution (prisoner’s dilemma works, zero-sum games work), but others are hitting equilibrium at unexpected points. This would mean that either the economic theory is wrong (not likely…) or that there is an issue in the driving principle behind the turtle’s decision making (more likely). However, I don’t quite know what this issue would be. Because of this, I’m going to attempt to meet with, or the very least email, one of my economic professors to gain a better understanding of the problem at hand.

**Questions:**

* + Ideas for extending the current model?
	+ Visualization tips?
	+ I somewhat took the “brute-force” approach to coding this, specifically the utility assignment, so I was wondering if you have any tips on maybe cleaning things up or shortening the code?
	+ Thoughts on the turtles’ learning procedure? / Any ideas why it doesn’t produce consistent results with the economic theory?

**Next Steps:**

1. Figure out the bug and come up with a solution to the issue.
2. Brainstorm extensions and additions.
3. Implement them.