Instituto Superior Técnico

Complex Network Science

Fairness in the Ultimatum Game

Group 27:

- André Ribeiro, nº 86384
- Diogo Ramalho, nº 86407
- Jorge Pacheco, nº 86457

Introduction

The Ultimatum Game is an exercise in Experimental Economics that has gained popularity since it's debut.

The premise is simple, involving two players: one plays the role of proposer and the other the role of responder. Given a certain amount, the proposer will suggest a split ratio between the two players, and the responder will have to either accept or refuse. If the proposal is refused, no player will receive anything. Thus, it becomes a dilemma of proposing a ratio high enough to be accepted and having an expectation low enough for the ratio to be proposed.

A Game with such a structure allows us to study many social behaviours, since there are numerous properties that factor into decision making, such as reputation or preference, to name a few.

This project will focus on how distributions of offer and demand evolve over time, across multiple scenarios and different types of players.

Structure

The code used for the development of the project was written in Python 3, and each type of scenario and element was split into different files, in order to promove code modularity and better development.

For each simulation, the two scenarios that were considered were described in the paper *"The Spatial Ultimatum Game"* by Page, Nowak and Sigmund. The first scenario represents a one dimensional ring, where each player has k neighbours on each of its sides. The second one represents a two dimensional grid, where each player is a neighbour to each of its adjacent players (up, down, left and right).

Each player has a default offer and acceptance values, which are randomized at the start of the algorithm. After each play cycle, the payoff is calculated considering each player's offer (p) and acceptance (q) values, more specifically, it is the sum of each slice obtained in accepted splits with another player S2:

$$Payoff = (1 - P(S1)) * a1 + P(S2) * a2$$

where a1 and a2 represent boolean values (0 or 1) that dictate whether or not each proposal was accepted. That is, a1 is true if q(S2)<=p(S1) and a2 is true if q(S1)<=p(S2).

After each iteration of plays, a player will imitate a random neighbour if the latter has a higher amount of fitness (represented by the sum of payoffs obtained with playing with its neighbours), otherwise, it will remain unchanged, this imitation process also involves a degree of error (ε), which prevents the fitness from a given player to stagnate the entire board.

There is also a subtype of player, the Robot, that possesses most of the properties of a standard player, except for the fact that it cannot imitate other players. What these elements introduce to the Game is the idea of constant flow of change of strategy since in every iteration they change their strategy randomly.

Hypothesis

All players, in every scenario described, will implement a random strategy at the beginning of the simulation. Such an approach is merely to avoid an already converged simulation, where the values of offering and expectancy (p and q) are constant across the entire population and thus will not change any further. We expect that, in most scenarios the p and q values to eventually converge.

Should both of these values converge to an amount close to 0.5, we are then dealing with a "fair" scenario, where both players receive the same amount. However, besides simulating and evaluating fairness occurrence, we also want to deduce which parameters may deviate the values of convergence, where players will settle with an unfair split.

Considering every scenario, we expect the population to reach a state of fairness on the standard ring and grid setups, since they hold the same essential social mechanics. Despite the fact that all players start with a random strategy, through fitness comparison, the most optimized strategies will eventually predominate over the social structure. Considering other situations, where, for example, robots are included, we may be dealing with a scenario where there will never be a consensus throughout the population.

Grid Population

As described previously, in a grid, the players are organized in a two dimensional array, this means that each player will have, at most, 4 neighbours with which one can play the Ultimatum Game.

Each iteration consists of a player negotiating with a random neighbour, being both the proposer and the responder, this would mean that the possible payoff fits in the value space between 0 and 1. Considering that, we can deduce that in order to reach social consensus, there will need to be a large amount of iterations, representing thousands and thousands of proposals.

The simulations that were developed will plot the average ps and qs along the iterations, however, for a more thorough analysis, a subroutine was implemented that plots each 400th iteration as a heatmap, where we can more intuitively visualyze the evolution of the Game.

The Distributions were also converted into GIF format and were included in the project folder. These display each frame taken between every 400 iterations, which represents the variation of ps and qs across the population in a more fluid fashion.

It should be noted that 150 frames were used, which consists of a total of 60000 iterations, however, the frames represented aren't equidistant from each other, these were chosen simply to give a good idea of the strategy evolution (as represented in Figure 1).

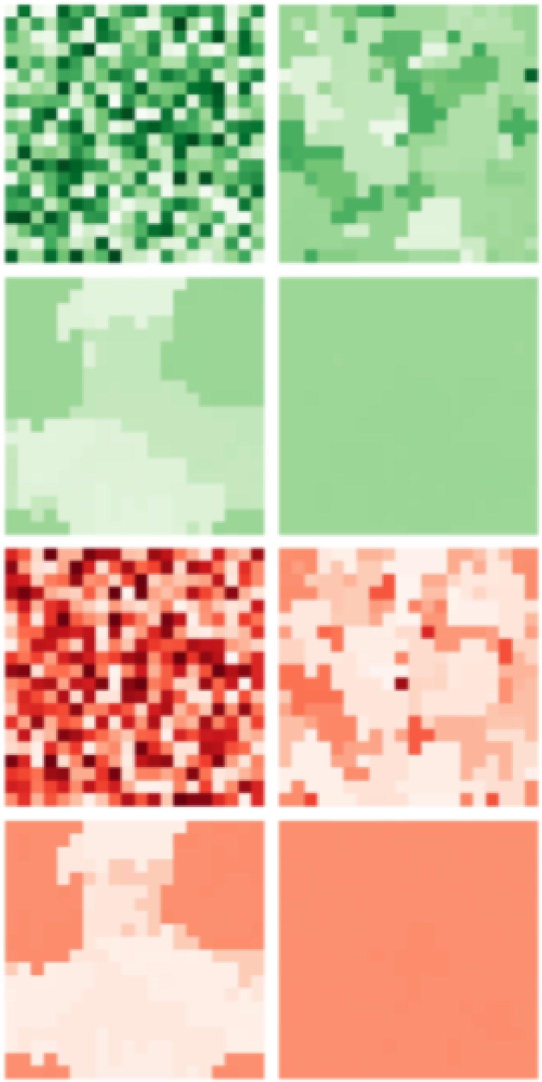

Figure 1. P (green) and Q (red) distributions over time (1st, 2800th, 22000th and 37600th iteration). Lighter colors represent lower values, and darker colors represent higher values.

In fact, looking at the gif files, one can deduce that the convergence into a state of fairness can be represented by a logarithmic function, in the sense that higher changes in strategy are observed in earlier iterations rather that later ones.

After running the simulation for a considerable amount of iterations, the consensus was 0.33 for p and 0.32 for q. To put this in perspective, players are accepting 0.33% of a share, under the premise that they are allowed to only give 33% of their own share. Although these values do not represent the ideal fairness of a 50-50 split, the players have in fact developed their own idea of fairness, since their payoff is still maximum ($q < p$ and $p + (1 - p) = 1$).

Ring Population

Unlike a grid, players in a ring are spread out in a one-dimensional line, where the extremes are connected to each other.

For this particular set of tests, the amount of neighbours each player can interact with is 4 (2 on each side), however, the effect of number of neighbors on the presence of fairness will also be discussed further.

Running the simulation, it was observed a similar type of evolution of strategies, even though only restricted to one dimension.

We will once again resort to the heatmap representation for easier viewing, however, it should be noted that frames were saved between fewer iterations (200 instead of 400), from which we can conclude that this scenario converged to fairness more quickly than it's two-dimensional counterpart. This time, the entire population adopted a "fair" strategy in roughly 16400 iterations, 21200 less than the last test.

It is theorized that the quicker spread of fairness was due to the smaller size of the population, however such a hypothesis will require more testing with a varied number of players for each simulation.

Despite the quicker conversion, the populations version of fairness was roughly the same, with p and q values presenting values of 0.3. From this we can conclude that both scenarios present the same social mechanics, as represented by their spread and type of fairness

Figure 2. P (green) and Q (red) distributions over time (1st, 1400th, 8800th and 16400th iteration).

Further Variations

There are further variables which we can use to manipulate the population in order to verify their impact on fairness.

<u>Robots</u>

Robots, as introduced before, are a type of player that are constantly changing their strategy and will not mimic other players.

Given these properties, their presence on populations are expected to simply alter the state of fairness, possibly even impeding the process.

Inserting robots into the previous one dimensional population, there was a stratification of fairness on players between robots. In other words, there was no absolute convergence, only among players separated by these randomly distributed robots.

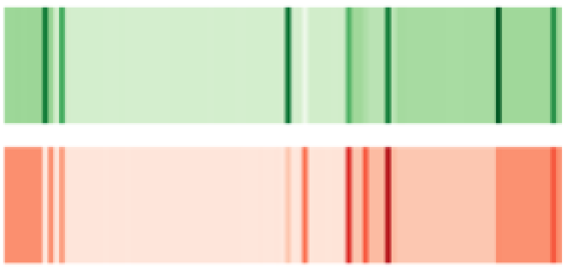

Figure 3. Population's P and Q distribution after 30000 iterations,throughout the population (x axis). Different colored columns represent the robots, whose strategies change after each iteration they participate in.

Despite the presence of robots, the properties of the players do not differ from the original population, that means that the amount of players (robot and non-robot), as well as the number of neighbours, remains the same.

One final observation is that the value of each partitions P and Q strategy seems to be higher the smaller the size of said partition, however that value also seems to depend on the partitions that neighbour each other. This is due to the fact that players placed near robots can still interact with players of the other partition, as they have 4 neighbours on each side.

Despite this stratified society, results showed that the converged strategy presented higher values for P.

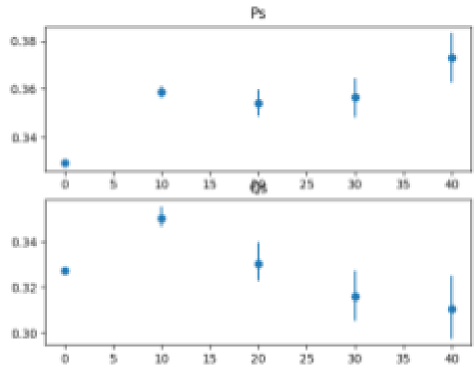

Figure 4. P and Q values (y axis) in relation to amount of robots (x axis)

Epsilon

Epsilon is the name of the variable that dictates how closely a player will imitate players with higher fitness, the lower the epsilon, the closer the imitation. This means that there may be a correlation between such value and the amount of iterations a population needs to converge.

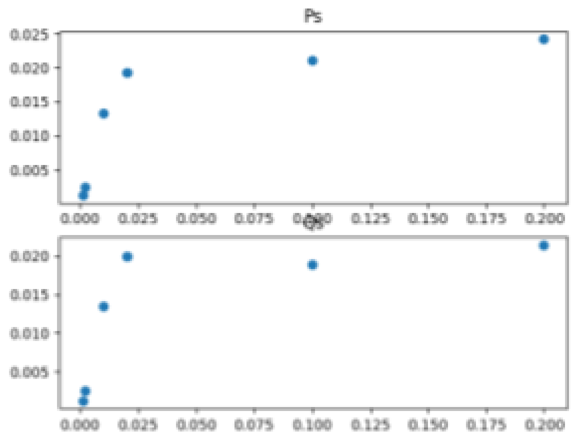

Figure 5. Variation of average P and Q values (y axis) in relation to Epsilon (x axis).

From figure 4, we can conclude that not only does the deviation of p and q values increase with ε, but also the value of both averages decrease.

Population Size

We can also correlate fairness with population size. All previous simulations presented a group of 101 players.

Running the grid simulation with different population sizes presented the following results:

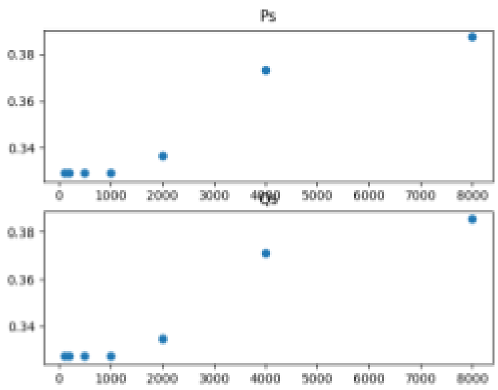

Figure 6. Variation of p and q convergence values (y axis) in relation to population size (x axis)

The above simulations demonstrate that higher populations lead to more fair splits, in order words, more populated simulations will provide more rational players.

Number of Neighbours

As hinted before by the different amount of neighbours between the ring and grid simulations, populations were also tested with different amounts of neighbours for each player.

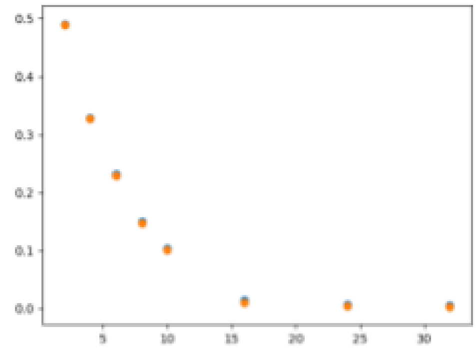

Figure 7. P and Q variance (y axis) in relation to amount of neighbours (x axis).

Running similar simulations with different amounts of neighbours for each player effectively demonstrated that a more connected population will also agree on less fair splits. Similarly to the population size, the more a community is connected, the more selfish a player will be.

Mutant players

A Mutant player is a subtype of player that has, for each play cycle (iteration where it is selected), a small percentage of randomly changing its strategy. It was implemented with the hypothesis of changing a converged population.

After running a grid simulation and interpreting the results, occasional appearances of new strategies were observed, however these were eventually surpassed by the dominant strategy. What these mutant players are doing is causing disturbances in a converged population in an attempt to set the new dominant strategy.

Conclusion

Several simulations with different properties demonstrated that the occurrence of fairness happens in very specific scenarios. In fact the scenario with the most fair split ended up being the simplest one: a ring population where each player had only two neighbours.

The lack of scenarios where fairness was the dominant strategy can be pointed to the current version of payoff. Has explained in the Grid Population section, the players have established their own idea of fairness, since fitness was calculated not using instances of games, but pairs of instances of games, in other words, two players played both the proposer and the responder. This leads to the idea that as long as all players have the same offer and acceptance values, it doesn't really matter what value is proposed, since the player will always get a total sum of 1. In fact, any value of p could be used as long as it was larger than q and it was globally used.

One possible way of changing this scenario would be to remove the proposer-responder cycle, in other words, the selected player would only play the proposer and the random neighbour the responder. This would lead to a bigger conflict of interest and the split would theoretically be pushed to a more fair split.