Complex Networks Project Report - Group 23
Evolution of Fairness in Different Networks

João Antunes nº 87668
Guilherme Serpa nº 82078

Abstract

The ultimatum game is a mathematical framework that models the conflicting of rationality versus fairness. In this report we analyse the evolution of fairness throughout different typology's types. We also explore the idea of applying penalties to low offer values to see how they affect the behavior of each player. We conclude that the model does not depend on how many players a player exchanges with but rather on the update of the players p and q strategy values. We also conclude that systems with penalties are more rational and less fair while systems without penalties end up being more empathetic and more fair.

1 Introduction

In this project we intend to evaluate the results of playing the ultimatum game in different network settings, analyse the evolution of fairness and see how applying penalties to low offers may affect the overall fairness in the model. The ultimatum game is an experimental game in which a player has a certain amount of money. It is this player's job then to split this amount of money with another player. The player on the receiving end decides whether or not if the offer p will be accepted. If the receiver rejects the first player's offer, regarding its acceptance threshold q, then neither of the players receives anything; if the receiver accepts the first player's offer then the money will be split between the players as per the first player's proposal. What we consider to be a fair play is for a player to offer a value of p where $0.4 \leq p \leq 0.5$ [1]. If the p value is greater than or equal to the q value of another player, the offer will be accepted, and both players split the offer, otherwise it will be rejected and both players get nothing. We considered 4 settings: The baseline (every player interchanges plays with every other player) [1]; a scale-free network (Albert-Barabasi); a random network; and a small-world network (Watts-Strogatz). We replicate the multi-agent-system model of *The spatial ultimatum game* [1] and *The Ultimatum Game in complex networks* [2]. We chose to work on this topic because we wished to study the behavior of players in a game when presented with either a fair or unfair choice and verify if it replicates what happens in a real world environment. We investigate whether or not the players will be accepting unfair but rational solutions or if they will gravitate towards fair and empathetic solutions.

2 Methods

We implemented our project using the Python3 coding language with the following libraries and packages: NetworkX [3] to generate the graphs, NumPy [4] to model the graph and player's features and Altair [5] to generate the presented charts.

2.1 Model's input variables

Our model receives as input 7 variables:

- `N`: Number of players.
- `m`: Number of initial links of a player (only relevant to Albert-Barabasi and Watts-Strogatz networks).
- `generations`: Determines how many times the model is executed.
- `prob`: Probability of creating links between players (only relevant to Random and Watts-Strogatz networks).
- `network_type`: Determines which type of network the model will have.
- `mutation_error`: Decimal value $\mathcal{E}$ (between 0 and 1) that adds more random values to a player's p and q values.
- `low_offers_penalty_flag`: Flag that determines if the model will apply penalties, it can be either (it is either 0 or 1).

2.2 Graph and Node's features

The graph has the following features:

- `Offer_Matrix`: Each cell represents the difference $1-p$ of a player's offer.
- `Reward_Matrix`: Each cell represents the Reward value received by a proposal of a player of value p.
- `Payoff_Matrix`: Each cell is the sum of the respective offers and reward values and represents the overall payoff of a player.

Each player has the following features:

- `p_Value`: Value of a player's offer.
- `q_Value`: Player's acceptance value.
- `Play_Flag`: Indicates whether a player is eligible to be proposed an offer.

At the beginning of every simulation all players have random offer and acceptance values. [1]

2.3 Natural Selection and Random Strategies

At the end-step of each generation every player updates its p and q strategy according to the payoff proportion of the fittest players of the previous generation. That being said, this update will also take into account a given mutation error $\mathcal{E}$ in order to add more random strategies.

2.4 Low Offer Penalties

Our model has the possibility of applying penalties to players that propose low offer values, this penalty consists of marking the player as not being suitable to perform a proposal with, thus affecting its payoff. Once a player is marked as not fit to play with, other players will refuse to play with this marked player and no proposal will be made. An equal amount of tests were run with both the penalty flag turned off and on to insure the best comparisons.

2.5 Model Execution

In each generation, each player will play either with every other player (in the baseline model) or play with all its neighbours (in a network environment), as both the proposer and the receiver.
Every time an offer is accepted:

- `Offer_Matrix[proposer][receiver]` = $1 - p$.
- `Reward_Matrix[receiver][proposer]` = p.

After all players have exchanged with each other, the payoff of each player is calculated. This payoff is obtained by adding the Offer Matrix with the Reward Matrix [2]. At the end of each generation the strategies (the p and q values of each player) will be propagated through the next generation proportionally to their payoff levels. The update of these strategies will also take into account the mutation error that will choose the p and q values randomly within the given error interval centered around the previous generation's p and q values. [1]

In the next section we compare and evaluate the baseline model approach with the network model approach by comparing their respective average p and q values, the percentage of executed fair plays, how altering the m and *prob* parameters will affect these results and by analyzing the effects of applying penalties to low offers.

3 Results and Discussion

All the simulations were executed with 10^4 generations, due to time constraints, 0.1 mutation error $\mathcal{E}$ (except for the values produced in the table below where it were executed 10^5 generations with several mutation errors $\mathcal{E}$) and with a population of N = 100 players. The time convergence scale is logarithmic ($\log_{10} t$) where t is the timestamp of each generation) to illustrate the albeit noisy convergence in a long term temporal setting. The fairness percentage is calculated for each generation. There is a counter that resets with every new generation, and every time a player makes an offer that is considered fair ($0.4 \leq p \leq 0.5$), the counter is incremented. At the end of every generation, the counter for fair plays is divided by the total amount of plays that occurred, leading to the fairness percentage value.

3.1 No Penalty simulations

The following table shows the average offer and acceptance values for different mutation errors in the baseline model. These values were obtained only considering values after the point of convergence of value $\log_{10} t = 2$ of the noisy convergence scale. These values replicate the values of the *Table 1* of the work done in *The spatial ultimatum game.* [1]

$\mathcal{E}$	p	q
0.001	0.0638	0.0424
0.01	0.1164	0.0529
0.1	0.3245	0.1376
0.2	0.3384	0.1269

In Figure 1 (Baseline), Figure 2 (Albert-Barabasi), Figure 3 (Random Network) and Figure 4 (Watts-Strogatz) we compare the baseline with different kinds of networks, evaluating the evolution of the p and q values. What we conclude from these charts is that they all show the same kind of behaviour and that the values of p and q tend to decrease throughout the generations until their convergence point.

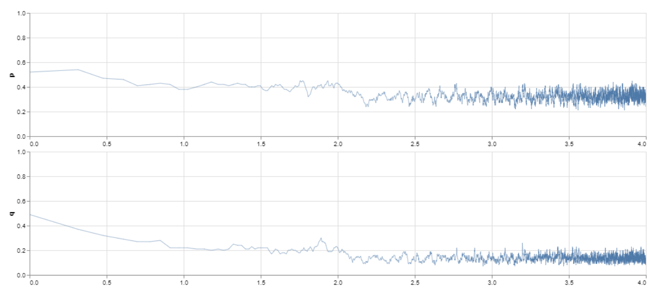

Figure 1. Evolution of p and q values in the Baseline model. Every player plays with every other player as both a proposer and a receiver. These values replicate the values of the *Figure 1* of the work done in *The spatial ultimatum game.* [1]

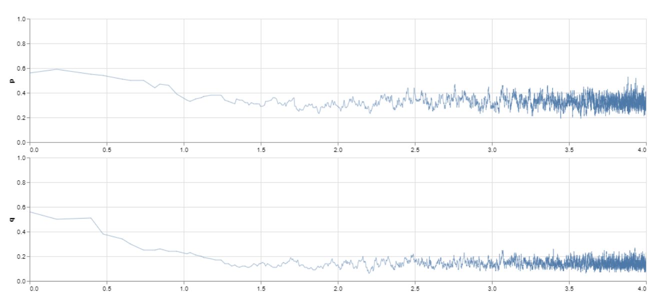

Figure 2. Evolution of p and q values in a Albert-Barabasi network for $m = 5$. Every player plays with its neighbors as both a proposer and a receiver.

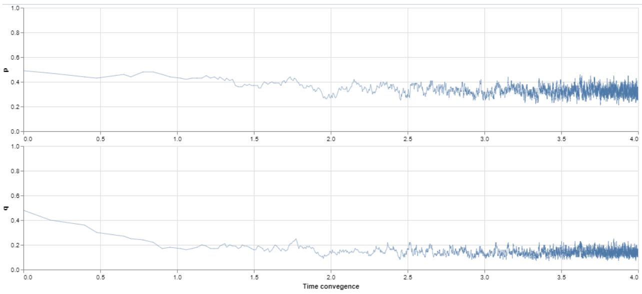

Figure 3. Evolution of p and q values in a Random network for $prob$ = 0.1. Once again every player plays with its neighbors as both a proposer and a receiver.

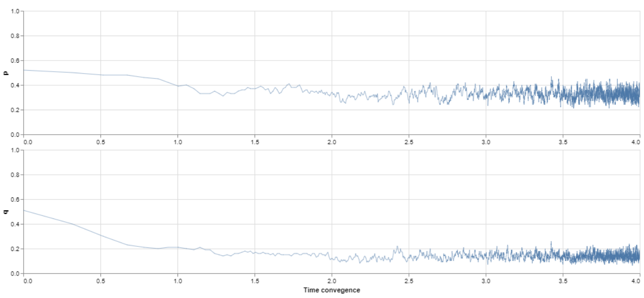

Figure 4. Evolution of p and q values in a Watts-Strogatz network with m = 5 and $prob$ = 0.5. Once again every player plays with its neighbors as both a proposer and a receiver.

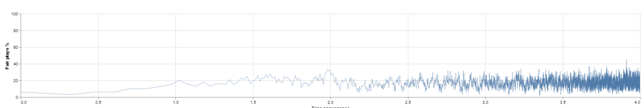

Figure 5. Evolution of the fairness percentage values in in the Baseline model.

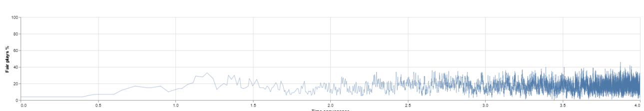

Figure 6. Evolution of the fairness percentage values in a Albert-Barabasi network for m = 5.

In Figure 5 (Baseline), Figure 6 (Albert-Barabasi), Figure 7 (Random Network) and Figure 8 (Watts-Strogatz) we compare the baseline with different kinds of networks, but this time we evaluate the evolution of fairness throughout all generations. We once again observe that all charts show the same type evolution of the fairness values. Overall the levels of fairness increase until they reach their convergence point around $\log_{10} t = 2$ of the noisy convergence scale.

In Figure 9 (Albert-Barabasi), Figure 10 (Random Network) and Figure 11 (Watts-Strogatz) we compare the the average value of fairness with different parameters of m and $prob$.

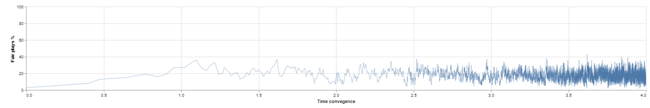

Figure 7. Evolution of the fairness percentage values in a Random network for $prob$ = 0.1.

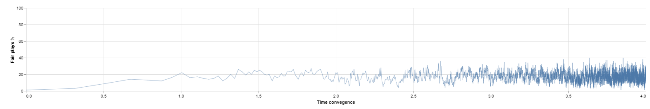

Figure 8. Evolution of fairness percentage values in a Watts-Strogatz network for m = 5 $prob$ = 0.5.

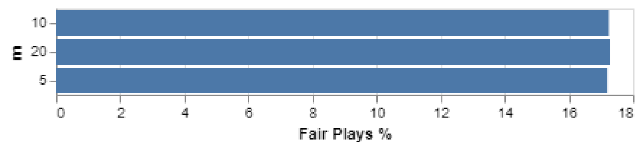

Figure 9. Evolution of the fairness percentage values in a Albert-Barabasi network for different values of m.

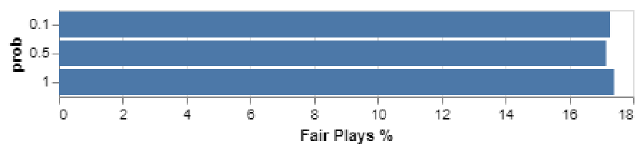

Figure 10. Evolution of the fairness percentage values in a Random network for different values of $prob$.

These average values are only considering the values after the convergence point (in this case we assume the convergence point point around $\log_{10} t = 2$ of the noisy convergence scale). We observe that overall the levels of fairness do not change with different parameters of m and $prob$.

What we were able to conclude from the these results is that despite there being subtle differences in the charts, the evolution of fairness does not depend on how many players each player interchanges with but rather on how the strategies are updated. Since the player's strategies take into account all players of the network it makes sense that they show equal variation, despite this, one difference between the baseline and network setting is that each player now has lower payoff values (when compared to the baseline model) since that in a network graph each player does the ultimatum game with fewer players each round. It is also possible to observe that the players showed a more fair and empathetic behaviour since the fairness percentage increased until its convergence point.

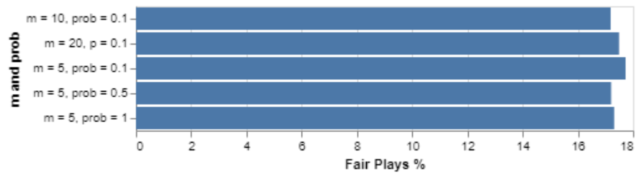

Figure 11. Evolution of the fairness percentage values in a Watts-Strogatz network for different values of m and $prob$.

3.2 Penalty simulations

What we consider to be a low offer is any offer with a value p such that $0.1 \leq p \leq 0.2$.

In Figure 12 (Baseline), Figure 13 (Albert-Barabasi), Figure 14 (Random Network) and Figure 15 (Watts-Strogatz) we once again compare the baseline with different kinds of networks, evaluating the evolution of the p and q values while applying penalties. This time, it is possible to observe that applying penalties had severe changes the the evolution of p and q values of the baseline model, and in a network setting the values of p and q started to have more scattered values when the convergence point is reached. This differs from when the penalties were not being applied and the p and q values tended to more constant values. One aspect worth mentioning is that the the results obtained from the network simulations now differ a lot from the baseline simulation. This happens because since each player now exchanges with fewer players than the baseline model each round, there are less players to become unfit to perform offers to and thus, less players to offer low values of p.

In Figure 16 (Baseline), Figure 17 (Albert-Barabasi), Figure 18 (Random Network) and Figure 19 (Watts-Strogatz) we compare the baseline with different kinds of networks, but once again we evaluate the evolution of fairness throughout all iterations. All charts present very different results than

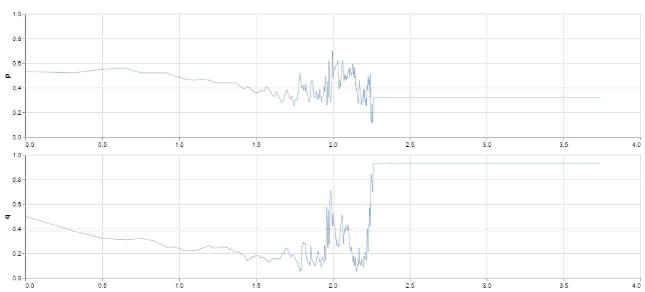

Figure 12. Evolution of fairness percentage values in the Baseline model while applying penalties to low offer values ($lowOffersPenaltyFlag = 1$). Both p and q values tend to maintain a constant value after several iterations, this happens because at a certain instant of the simulation the payoff values of each player started to become 0 due to no existing players left to exchange with. When the simulation reached its convergence point, players stopped having a fair approach and increased their acceptance threshold values and started to show a more rational behaviour.

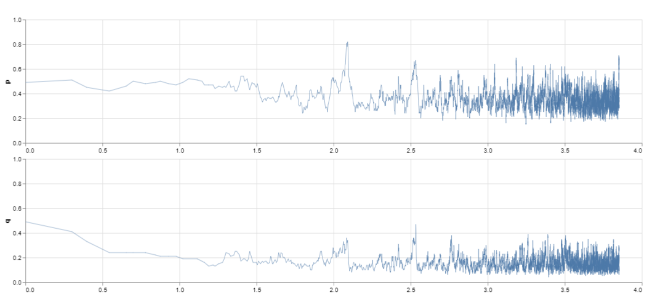

Figure 13. Evolution of p and q values in a Albert-Barabasi for $m = 5$ while applying penalties to low offer values ($lowOffersPenaltyFlag = 1$).

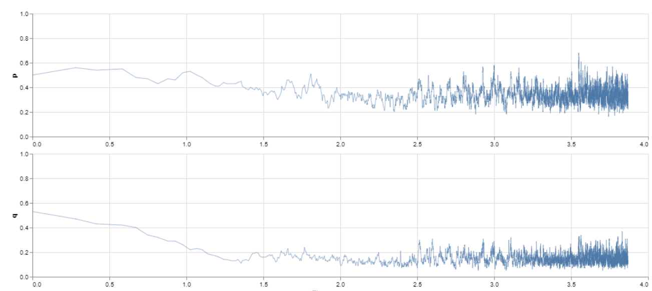

Figure 14. Evolution of p and q values in a Random network for $prob = 0.1$ while applying penalties to low offer values ($lowOffersPenaltyFlag = 1$).

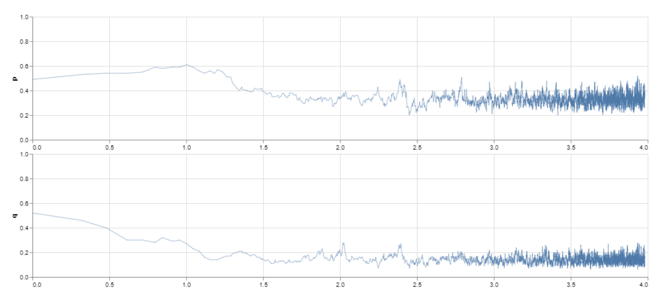

Figure 15. Evolution of p and q values in a Watts-Strogatz network for $m = 5$ and $prob = 0.5$ while applying penalties to low offer values ($lowOffersPenaltyFlag = 1$).

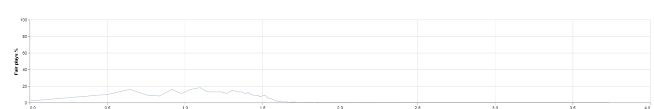

Figure 16. Evolution of fairness percentage values in in the Baseline model while applying penalties ($lowOffersPenaltyFlag = 1$).

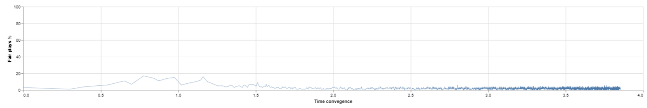

Figure 17. Evolution of fairness percentage values in a Albert-Barabasi for $m = 5$ while applying penalties ($lowOffersPenaltyFlag = 1$).

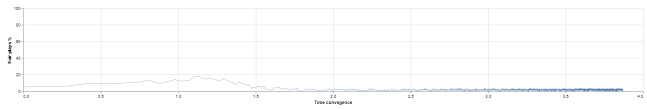

Figure 18. Evolution of fairness percentage values in a Random network for $prob = 0.1$ while applying penalties ($lowOffersPenaltyFlag = 1$).

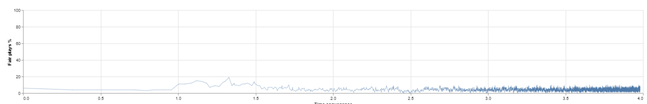

Figure 19. Evolution of fairness percentage values in a Watts-Strogatz network for $m = 5$ and $prob = 0.5$ ($lowOffersPenaltyFlag = 1$).

the simulations with no penalties. The baseline has 0 fairness levels by the end of the simulation, and in a network environment the fairness levels tend to near 0 values. In contrast, the simulations that were carried out without a penalty had higher fairness levels.

From the previous results it is possible to conclude that applying penalties led to a lot of players being unable to receive offers from other players and, as a consequence, the fairness levels of every environment decreased a lot, the same applies to its payoff values. Also the players started to demonstrate a more rational kind behaviour.

4 Conclusion

In short, we conclude that all of the environments showed a tendency towards the same type of behavior when penalties were not being applied. In this scenario, the players began to gravitate towards more fair and empathetic solutions until they reached their convergence point. When penalties were added to the system and players had to deal with being penalized for making a low offer, players started having more rational behaviors and decreased the general level of fairness of the model. Another conclusion is that the evolution of fairness in this system does not depend on how may players play with each other but rather on how the player's strategies are updated.

5 Future Work

With more time we would try to alter the way the model updates the strategies of each player and explore if and how strategy updates depending on neighbours instead of all players of the population would affect the model. We would also execute the system with more generations in order to get more accurate results.

References

[1] K. Page, M. Nowak, and K. Sigmund, "The spatial ultimatum game," *Proceedings. Biological sciences / The Royal Society*, vol. 267, pp. 2177–82, 12 2000.

[2] R. Sinatra, J. Iranzo, J. Gómez-Gardeñes, L. Floría, V. Latora, and Y. Moreno, "The ultimatum game in complex networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, p. P09012, 09 2009.

[3] A. Hagberg, P. Swart, and D. S Chult, "Exploring network structure, dynamics, and function using networkx," Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2008.

[4] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2

[5] J. VanderPlas, B. Granger, J. Heer, D. Moritz, K. Wongsuphasawat, A. Satyanarayan, E. Lees, I. Timofeev, B. Welsh, and S. Sievert, "Altair: Interactive statistical visualizations for python," *Journal of open source software*, vol. 3, no. 32, p. 1057, 2018.