Kryptonite for Squeezers
CSNS Project 2021/22

Lorenzo Niccolai
ID: 0001004845

University of Bologna
lorenzo.niccolai@studio.unibo.it

Abstract. In this project I tackled a variation of the classic Rock-Paper-Scissors game on a population of fixed agents. Then I experimented with different settings of spatial contexts and noise to see what the model could produce. Finally I gathered the results and confronted them with my suppositions.

1 Chosen Phenomenon

The phenomenon I modeled in this project is quite simple and is based on the famous Rock-Paper-Scissors (**RPS**) game[1], I named it **Krypto, Mediators and Squeezers** (KMS). It can be applied to many different contexts but to start I like to relate it to a mating contest between males of a species[2]. So the game between two players will be to mate with a given female and every player will try to maximize his chances to mate and prosper.

The game has a maximum reward of 4 which means having the certainty of mating and a minimum of 0 which means having the certainty of not mating. For example if two players play and the first will always mate for sure the payoff would be 4/0, if the payoff is 3/1 means that 75% of the times the first player will mate and the remaining 25% the other will. So in this contest there are 3 strategies:

Squeezer a Squeezer can crack the game and obtain an unique outcome only if plays with another Squeezer, the female will mate with both contestants. In the other cases it will lose quite surely.

Krypto I named this strategy Krypto because it is the enemy of the Squeezers, in fact when a Krypto plays against a Squeezer it always win. When playing with another Krypto they both have the same chances.

Mediator this strategy is the one that absorbs changes in populations between the other 2. In fact a Mediator will probably win with a Krypto but probably lose with a Squeezer. When playing with another Mediator they both have the same chances.

1.1 Why is interesting

So I though of this problem as an RPS game but with a few tweaks. Starting with increasing the reward domain[1] and doing such introducing a probability distribution on the outcomes, for example when a Mediator plays with a Krypto the payoff will be 3/1, this encoding that 3 times out of 4 the Mediator will mate.

The other idea I wanted to try was a strategy that can obtain more than any other when playing "against" an akin: This meaning that the Squeezers are more smart than the others and when they find another Squeezer in the mating contest they recognize it and cooperate to obtain both the goal.

One could argue that there is no reason to not adopt a Squeezer strategy a priori, and this is true but I will address this in Section 3.

1.2 Applications

Some of this phenomenon applications that I though about are: for example a free-for-all online game (et similia) where players who collaborate (Squeezers)

[1] In RPS winning is 1 and losing is -1, both are sure events

can win against the others, but there is also the possibility that one takes advantage of the other (Krypto) when not expected, while the Mediator plays the game as it is supposed to be. Another case that came to my mind was two companies that share the know-how and both benefits from it, but again one of the two at a certain point can stop sharing knowledge to try and overcome on the other.

2 Modeling

To develop this project I started from a basis, the code from `evol.nlogo` provided by the Professor Angelo Trotta. This assume the agents have fixed positions, in fact they are represented by 1089 patches. Every time unit (tick) each agent play the game with another agent and a payoff for each is returned by the payoff matrix. After that - with a given probability - each agent revise his strategy by looking at another player payoff from last outcome and in case is better than his, adopt his strategy. In the next subsections I will comment any important addition or modification made.

2.1 Game Definition

The game models the phenomenon stated in Section 1, the matrix is reported in Table 1

	Krypto(K)	Mediator(M)	Squeezer(S)
Krypto(K)	2	1	4
Mediator(M)	3	2	1
Squeezer(S)	0	3	4

Table 1: KMS Game Payoff Matrix

2.2 New Features

Globals The globals I added are:

1. `tot-`i, with $i \in (1, 2, 3)$: these are three globals which accumulate the sum of the payoffs obtained by a single strategy. The update is done each tick, like the graph.
2. `m`i, with $i \in (1, 2, 3)$: these variables store the mean payoff for each strategy per each tick.
3. `east, west`: also called *conferences*, are `agentsets` made of `patches` which will be used in the Probabilistic Spatial Context extension (treated in subsection 2.3). In particular they divide the world into 2 equal portions, taking the central column (where `xcor` is 0) as reference.

4. `e-north, e-south, w-north, w-south`: also called *divisions*, similar to `east` and `west` but are subsets of them. In particular they divide a conference (e is east, w is west) into two equal portions.
5. `e-border, w-border`: these 2 strips of patches are used to prevent bordering agents to play with the other conference in the probabilistic spatial case.

These variables are defined in the `setup` procedure of the model. Since the number of columns is odd one of the conferences has one patch more.

Owned Variables The patches owned variables I added are:

1. `my-conf`: it defines the *conference* to which a single patch belongs.
2. `my-div`: it defines the *division* to which a single patch belongs.
3. `other-conf`: it is the border of the conference.

These variables are defined in the `setup-players` procedure of the model.

Plots In the interface I added a few outputs to make the model behavior easier to understand:

1. **Total payoffs of strategies**: as said before it displays the `tot` globals to represent each strategy cumulative fitness.
2. **means**: in this plot the `m` variables trends are plotted over time to see how the different strategies evolve.
3. **Populations**: this plot use the same information of Strategy Distribution, but shows the variation of each strategy on a different pen. With this is easy to visualize the typical RPS trend.
4. **East, West**: these plot are set like Populations but use data only from the respective *conferences*.

2.3 Spatial Context

It can be noticed from the interface that one model features is the possibility to use different degrees of Spatial Context, by using the chooser named `spatial`. Talking about the actual structure of the spatial interactions of the agents, I wanted to structure the interactions in a more complex way than just using neighbors, here is when the conferences and divisions come into play.

The spatiality is treated in a probabilistic fashion, in the sense that the pair of agents playing the game is decided by a `random-float` that goes from 0 to 1. Then the outcome has 3 ranges (which are arbitrarily decided) that will pair each agent with:

– a **neighbor** 50% of the times, removing neighboring patches which are not in the same division
– an agent in the same **division** 35% of the times
– an agent in the same **conference** (except the division) 15% of the times

This can be chose with the option 2 in the chooser. This kind of process can be thought as modeling a matchmaking mechanism (for example) that gives priority to locality but in some cases can resort to more distant agents. The set from which is extracted the other player is also the same set which will be used for extracting the agent to observe in the revision.

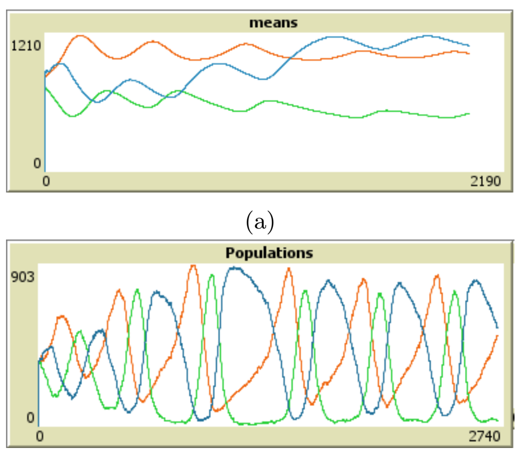
Also the basic spatiality option of making each agent play only with his neighbors is present, with option 1. Option 0 is random choice.

3 Evolution Analysis

Here I will discuss what I expected from the model and what I found running simulations on NetLogo. From now on the three strategies will have fixed colors in graphs or images:

- Orange for Krypto
- Green for Mediators
- Blue for Squeezers

3.1 No Spatial Context



(a)

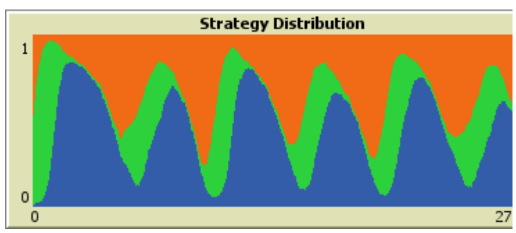
(b)

Fig. 1: (a) The mean payoff over time for the strategies, (b) the count of agents for strategy over time

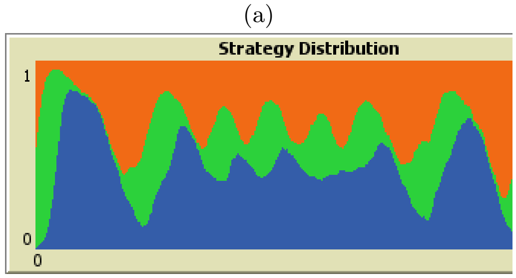
Just looking at Table 1 we can make a few assumptions on the possible evolution of a population, for example that taking as starting points populations with only 2 strategies without noise, one will inevitably take over, this is expected from the RPS nature of the game. We can also see (still without noise) that if initially we set the same number of agents for each strategy, surely the Mediators will disappear. The unclear thing is who will prevail? The Kryptos or the Squeezers? Actually the Kryptos has the upper hand, because their payoff will be higher than the observed more often than the Squeezers, so is very rare to see a Krypto transition to a Squeezer, they tend to stabilize. The tipping point for having the Kryptos not taking over is less than 12, so at least 1077 Squeezers (the total agents playing are 1089). Of course less Kryptos means they will propagate later.

The introduction of noise in revision protocols change the evolution considerably, in particular the Mediators start to reappear and act as a counter to the Kryptos, with this setting the model returns to the expected RPS behavior but with a difference. In fact as a result of the variation I introduced in the matrix the Squeezers have longer periods of dominance in the RPS succession - as showed in Figure 1.b - and from the Figure 1.a is possible to appreciate how the mean payoff starts lower than the Kryptos but then grows and remains higher.

3.2 Probabilistic Spatial Context



(a)



(b)

Fig. 2: (a) The attended evolution of this RPS, (b) An example of chaotic sequence

Now that the random matching has been addressed, we can test the model with the Probabilistic Spatial Context - described in subsection 2.3. The population evolution does not change much from the previous example, the only difference is that sometimes the attended alternation of the strategies (Fig 2.a) is replaced by a chaotic sequence (Fig 2.b). This is probably due to a fraction of the period in one of the 2 conferences (Fig 3), this resulting in a weird distribution graph.

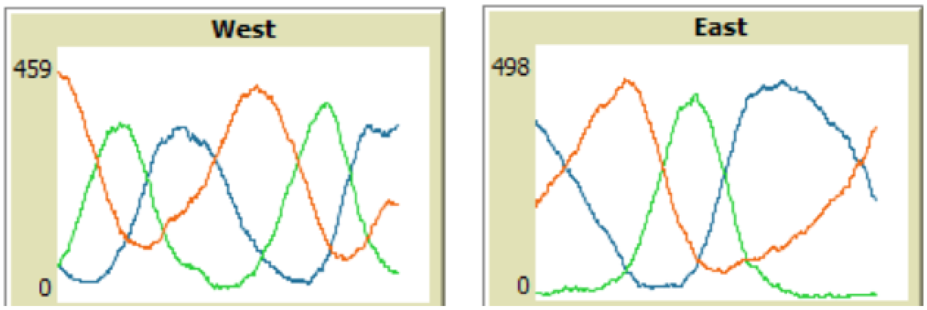


Fig. 3: In the West conference there are 2 different successions of peaks, while in the East only one cycle is completed

3.3 Local Spatial Context

Here the findings related to the local spatial context (stated in subsection 2.3) are reported. The behavior of the model when agents play only with their neighbors strays from the RPS trend and becomes more stable in terms of mean payoffs, even if the population counts still oscillate the means plot suggests that the 3 strategies have a clear constant rank.

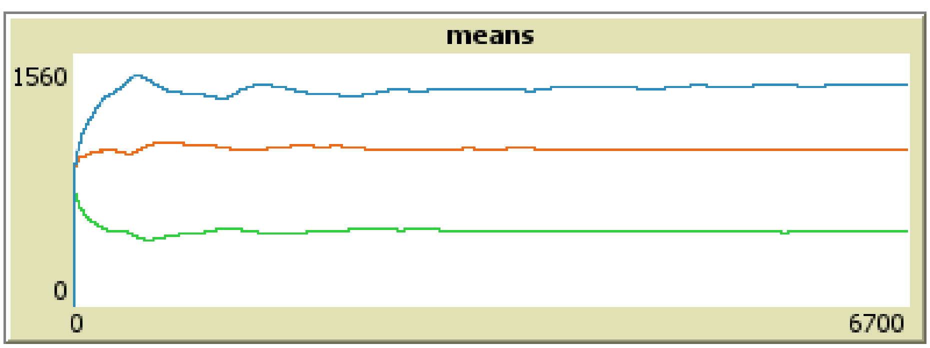


Fig. 4: The mean payoffs after the initial period stays constant

4 Results

I tried to come up with a variation to the classic Rock-Paper-Scissor game which can be found often in real life scenarios[3]. This approach showed various differences from the original game - which are stated in section 3 - and I also tried to apply different spatial contexts to this phenomenon to see how it would adapt to such situations or if some strategy would come as better in such cases. In conclusion the findings are somewhat in line from what I expected, of course further research could be done by increasing the population count for example or applying new kinds of spatiality.

References

1. Timothy N. Cason, D. Friedman, Ed Hopkins: Cycles and Instability in a Rock–Paper–Scissors Population Game: A Continuous Time Experiment. In: The Review of Economic Studies, Volume 81, Issue 1, January 2014, Pages 112–136,https://doi.org/10.1093/restud/rdt023
2. Sinervo, B., Lively, C. M.: The rock-scissors-paper game and the evolution of alternative male strategies. In: Nature (1996)
3. Kerr, B., Riley, M., Feldman, M. et al.: Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors. In: Nature 418, 171–174 (2002). https://doi.org/10.1038/nature00823