Ivan Prskalo

Mansury SSCI 381

Computational Social Science

12/6/2019

Apathy in The Minority Game and The El Farol Bar Model

I. Abstract

In this study, we introduce an *apathy* variable to the El Farol Bar model. We explore how apathy plays a role in the decision making of a set of agents whose primary objective is to decide whether or not to go to a bar. The agents want to go to the bar if they believe the bar's attendance will not be over a certain threshold. If the agents predict the bar's attendance to be over this threshold, going to the bar becomes unenjoyable and is considered the "wrong" choice. The *apathy* variable is a random number between zero and the *apathy-max* (a user defined variable between zero and one). Each turn, a random number is generated between zero and one for each agent and if the number is less than the agent's *apathy* number, the agent will decide whether or not to go to the bar at random. We explore how varying the *apathy-max* from zero to one will affect the average bar attendance and the spread of attendance.

II. Introduction

The El Farol bar model is meant to represent how actual people will make decisions using varying strategies. While everyone will be using different strategies and using logic to decide what strategy to use in an ideal world, we don't see that in the real world. When using these varying strategies and logic, we see an emergence of an alternating pattern of attendance above and below the threshold. This is not very human-like and doesn't represent how an actual population of bar-goers would decide whether or not to go.

The way people react to logical situations is not always logical. The problem with agent-based models is when presented with more complex situations such as the way the human brain works, it is impossible to completely model such a phenomenon. What we can do, however, is represent the way a human brain functions by breaking down certain aspects and modeling those. People are not logical beings, as they let emotions and other criteria affect their perception of reality. For example, people might not care enough to devise a strategy in our El Farol bar situation.

With the introduction of the *apathy* variable, we will attempt to make the model more representative of an actual population's distribution of bar attendance. We propose that the *apathy* variable will remove the alternating pattern that emerges when the agents use their best strategies to predict whether or not to go to the bar. By increasing the *apathy-max*, we hypothesize that the alternating pattern will become less noticeable or completely go away at some point. This will be more representative of the actual population of bar-goers.

III. Literature Review

In an ideal world where everyone has access to the same information and is able to make judgements in a similar fashion, we can still see variations in decisions all across the board. Modeling such a phenomenon through agent-based modeling is fairly simple, yet unrealistic if we consider our non ideal world. Some aspects of these ideal agents are simply unable to be replicated by real people, according to Sperry-Taylor (2017), people will not be able to predict possibilities like an ideal agent. When a person takes in new information they can go through a process called belief-revision. In order to accommodate a new piece of information, a person may change the way they see a situation. This introduces uncertainty which leads to misguided judgement that doesn't encompass the full picture of the game. This is far from ideal since the game theory aspect is confounded. Damaševičius & Ašeriškis (2017) found "The goal of game theory is to find and describe the behaviors of players, which provide best response to other players' individual decision choices" (p. 108). We can see why misguided judgement will result in inaccurate results in even the most rational individuals. The most rational agents on the other hand will not have misguided judgement because they are programmed not to. Rajpal & Dhar (2018) argue that these rational agents can maximize their long-term payoff while having limited information and no communication with other agents. This would be incredibly impressive to see in actual human interaction. It would actually be impossible to maximize in heterogeneous population due to fundamental differences in people's beliefs and values. Grilli & Sfrecola (2009) took this a step further by examining a neural network's approach to game theory, and more specifically, to the minority game. Similar results arise from the neural network's approach as to the agent-based model, but as time goes on, the predictions approach the threshold. This means that the neural network is able to become even more accurate than certain agent-based models. This makes it more accurate, but it doesn't incorporate the realism of an actual person. People are not perfect and do not always follow set rules. That is why when using agent-based modeling, taking as many of these human factors into account is important. In my model of the minority game, more specifically, the El Farol Bar model, I will be adding simulated human apathy which will dampen the agent's judgment. This will replicate a person's misguided judgment and offer a more biased game theory. Agents will not be inclined to please other agents because their future placement depends on it. It will better represent the phenomenon of a human minority game.

IV. Agent-Based Methodology

A. Overview

1. Purpose

This model is attempting to replicate the social phenomenon of whether or not people decide to do something based on previous results. More specifically, this model explores whether or not people will attend a bar while knowing the amount of people that attended the bar on previous nights. The model is trying to demonstrate how people will change their strategy as new information is gathered. The purpose of the modification to the original model is to better represent human actions and thought processes in the original El Farol model.

2. Entities, State Variables, and Scales

Starting with entities, the model has two entities, the turtles (agents) and patches (coordinate system). The turtles represent people going into the bar, while the patches are organized into home patches and bar patches that represent the turtle's homes and the bar respectively.

Moving on to the state variables, we see that the turtles have a few. The strategies state variable is a list of different strategies that the turtles each have and is static. The best-strategy state variable is dynamic and is one strategy from the list of strategies that would have best predicted the attendance in previous results. The *attend?* variable is a dynamic boolean variable that changes from false to true if the turtle is going to be in attendance at the bar that night and then back to false at the start of every round. The prediction variable is a dynamic variable directly affected by the best-strategy variable. It is the predicted attendance of the bar that night using the best-strategy. Finally, the *apathy* variable is a static variable that is assigned at the start of each run during setup. This variable is between zero and the *apathy-max*, a global variable. It is used to decide whether or not the turtle cares to use a strategy or not. The patches have few variables compared to the agents. The bar patches have a variable called attendance, which is the actual attendance of turtles in the bar patches. The bar patches have one more significant state variable and that is whether or not it is crowded. If the attendance is greater than the threshold, then the bar is crowded. The bar patches also have a location variable that stays static along with home patches.

Finally, we move on to the scales. The temporal scale in our model is one week per tick, but the actions of our agents occur over a Thursday night, when the bar is known to attract the most people. The spatial scale is far more ambiguous than the temporal. We only constrict that the bar patches represent the entire bar, while the home patches represent everywhere that isn't the bar. Giving these patches a dimensional value would be inaccurate of the environment the patches are representing.

3. Process Overview and Scheduling

To give an overview of the processes, I will be stepping through the go procedure. Any processes that I do not go into detail will be fully explained in the submodels portion of this ODD. After initialization, the go procedure is called and the crowded sign on the bar is reset. We then ask the turtles to set their prediction to the best-strategy's predicted attendance, which is gained through the reporter procedure *predict-attendance*. We will then ask the turtle to change its attend? variable to true if the predicted attendance is less than the threshold and false if the predicted attendance is greater than the threshold. Now we generate a random float number between zero and one and check to see if it is less than the turtle's *apathy*. If it is less than the turtle's *apathy*, it will choose whether to attend or not at random. Next we move the turtles to either a random bar patch if they chose to attend or a random home patch if they chose not to attend. If they chose to attend we also increase attendance by one for each turtle that decided to attend. Now we move out of the context of the turtles and set the attendance equal to the number of turtles on bar patches. Then we will turn the crowded sign on if there are more turtles in attendance than the threshold. We will then add the attendance to a list called history that contains several previous night's attendances. The amount of attendances it stores is equal to that of the memory size which is set to 5 by default. Finally, we ask the turtles to update their strategies by calling the procedure *update-strategies* and increase the tick counter.

B. Design

This model was originally a modified version of the originally proposed minority game where the purpose was to choose a side that a minority chose. The El Farol model is a modified version of this as the objective is to choose to attend if the agent believes that there will be less agents than the threshold. This threshold does not need to be less than half of the population as originally proposed in the minority game model. We generate random weights to give each previous night in the agent's memory and then find which set of random weights was most accurate to the actual attendance on the previous night. During the BehaviorSpace testing, I found an emerging trend. I tested whether changing the *apathy-max* would have an effect on the average attendance and the standard deviation of the attendance. I saw that there was a decrease in average attendance and a decrease in standard deviation as the *apathy-max* went up. This means that as the *apathy-max* went up, the spread of data would slightly decrease. This is an emergent property because *apathy* is a random factor and one would think that an increase in randomness would increase spread and not decrease it. The agents are always adapting to each other's actions. They are all competing with each other and trust that their strategy will be superior to the other agent's strategies. This means that as more and more games are played, the agents will adapt and use different strategies with the new information they are given. The reason the agents act this way is because we want to better represent the way that people think. The use of a set amount of random strategies is assumed to be representative of how people will change the way they go about predicting how many people are going to go to the bar. While the change in opinion based on additional results is humanlike, the calculating seems a bit flawed. That's why I added an *apathy* variable that will add a bit of randomness to an agent's decision. People don't always care enough to do calculations or even think about the matter. That is why the *apathy* variable will better represent the way that people think about a realistic minority game like El Farol.

C. Details

1. Initialization

The setup of the model is very important to get right in order to replicate. Let us define a few variables that will be kept constant throughout the model after it starts. First off we have the memory size. It is set to five by default and can be changed from one to ten. The next variable we have is the number of strategies that each agent has. It is set to ten by default and can be changed from one to twenty. Next we have the overcrowding threshold. This is how many people can go to the bar before it is considered crowded. It is set to sixty by default and can be changed from zero to one hundred. Finally, we have the *apathy-max*. This is the maximum amount of *apathy* an agent can have. It is set to .8 by default, but can be varied anywhere from zero to one. Now we can move on to the setup procedure. The first thing that the setup procedure does is reset everything. It will then shape the agents into the form of "people." Then it will assign which patches are bar patches and which ones are home patches. It sets the top right quarter of the screen to the bar and the rest to home, but this shouldn't affect the results of the model. The history is then set with random numbers between zero and one hundred for each day. The length of the history list variable is equal to the memory size of the agents. It then sets attendance to the first element in the history list. Then it will create the text that contains the word "CROWDED" over the bar and will show it if there are more agents in the bar than the threshold. Finally, it moves onto creating the agents themselves. It creates one hundred agents and sets the color for each of them to white. It also generates a random number between zero and the *apathy-max* for each agent and sets random strategies for the agents strategies. It does this by calling the procedure, *random-strategy,* for as many times as the number-strategies indicates. It will select the best strategy to the first strategy as there is no data to work with yet. It will finally call the procedure *update-strategies* and exit the agent's context. We finish the setup by resetting the tick count.

2. **Submodels**

Update-strategies procedure:

The *update-strategies* procedure is the longest procedure aside from setup and go. It will set an initial best score equal to the memory size multiplied by one hundred plus one. Then it will iterate through each strategy and declare two variables for each strategy. These variables are score and week. It then enters a loop that will iterate as many times as memory-size. In this loop it will set the prediction equal to what the *predicted-attendance* is for that strategy by calling *predict-attendance*. It will then set the score equal to the score plus the absolute value of the attendance of the previous week minus the prediction. After iterating for the length of the memory-size it will check if the score is less than the best score and if it is it will set the best score to the score and after iterating through each strategy it will return the best score.

Random-strategy procedure:

The *random-strategy* procedure will iterate as many times as the memory-size plus one. It will then choose a random float between negative one and one to represent the weight to value each day in the history variable

Predict-attendance procedure:

The *predict-attendance* procedure has two parameters, a strategy and a subhistory. The procedure uses the weights it was given in the strategy to multiply the attendance of each week and then multiple the result by one hundred to return the expected attendance.

Move-to-empty-one-of procedure:

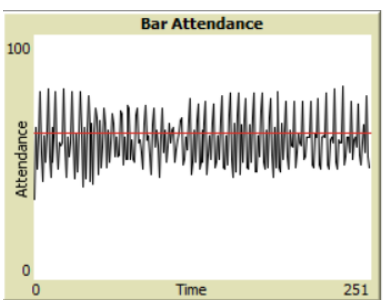
This procedure isn't necessary but makes sure that there are not more than one turtle on each patch when moving the turtles there. What it does is move a turtle to a location and then checks if there is another turtle there and if there is, it moves the original turtle to another location until it finds one with no turtles.

V. Results

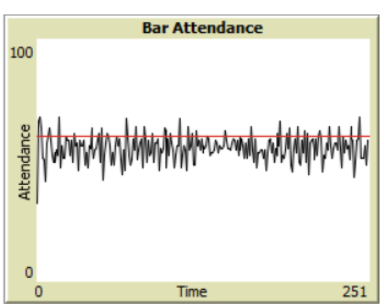
A. Simulation Output

We used BehaviorSpace to vary the *apathy-max* global variable from zero to one. We increased the variable by .05 and ran ten simulations for each *apathy-max*. We hypothesized that the attendance would lose its alternating quality that the original model exhibited. We can see the difference in the results from the two diagrams shown below. These diagrams show what a normal simulation with *apathy-max* equal to zero and one respectively might look like. In the first diagram we can see alternating peaks in the bar attendance, while the second diagram exhibits less varied peaks. In our BehaviorSpace analysis we found that the average attendance would steadily decrease as the *apathy-max* was increased. We also found that the spread of data would also steadily decrease as the *apathy-max* was increased.

apathy-max = 0



apathy-max = 1



B. Interpretation of Results

The average attendance and the spread of attendance decreased as the apathy-max increased. This was an emergent behavior that was not hypothesized. Looking back at the diagrams, one can see that there is less spread in the second diagram showing what a normal *apathy-max* of one simulation would look like. Having a higher spread means more agents are choosing incorrectly. One way to interpret this behavior is that in a less competitive environment, the tendency to go over the attendance threshold is much lower. Despite the *apathy* variable making the agent choose at random, the attendance still went down. On top of that, the spread of the average attendance went down as well. The amount of times the threshold was reached or surpassed went down, which is beneficial for the agents attending.

VI. Concluding Remarks

Through the introduction of the *apathy* variable, we were able to remove the non-human-like alternating pattern. We are not sure that with the *apathy* variable the model is an accurate representation of the way actual people would strategize going to the bar, but can confidently say it is a step in the right direction. We were able to remove the alternating pattern that would not occur naturally as well as reduce the spread of the results.

To address a few assumptions and limitations, we are not sure that flipping a coin is even closely representative of when a person decides not to come up with any strategy. That is what we are basically doing when the *apathy* variable is triggered. The way the original model works also has a few limitations. The way that strategies are devised is far from the way a human might devise a strategy. We are making the assumption that the algorithm that decides which strategy is best is representative of the way a human might devise a strategy.

For further research, we would like to see a better way to represent apathy in agents. The current 50/50 chance of going or not going when the *apathy* variable is triggered is not how people usually decide whether to go or not. A study showing the distribution of people's choices when they decide it doesn't really matter regarding the El Farol bar would be helpful in making a more representative model. Also, perhaps some sort of neural network way of devising strategies would be more accurate than our current algorithm. We speculate that having this additional information and technique would allow us to better represent the bar-goers of El Farol.

References:

Damaševičius, R., & Ašeriškis, D. (2017). Visual and Computational Modelling of Minority Games. *TEM Journal*, *6*(1), 108–116. https://doi.org/10.18421/TEM61-16

Grilli, L., & Sfrecola, A. (2009). A neural networks approach to minority game. *Neural Computing & Applications*, *18*(2), 109–113. https://doi.org/10.1007/s00521-007-0163-1

Rajpal, H., & Dhar, D. (2018). Achieving Perfect Coordination amongst Agents in the Co-Action Minority Game. *Games (20734336)*, *9*(2), 27. https://doi.org/10.3390/g9020027

Sperry-Taylor, A. T. (2017). Strategy Constrained by Cognitive Limits, and the Rationality of Belief-Revision Policies. *Games (20734336)*, *8*(1), 3. https://doi.org/10.3390/g8010003