Simulating the Price of Anarchy in Auctions

A Thesis
Presented to
The Established Interdisciplinary Committee for Mathematics and Economics
Reed College

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Arts

Robert S. Irvin

May 2020

Approved for the Committee for
(Mathematics-Economics)

Jeffery Parker

David Perkinson

Acknowledgments

Thank you to my advisors Jeff Parker and Dave Perkinson who helped me produce something I can be proud of. Jeff, who kept me on track each step of the way and diligently provided the support I needed to complete this process. Dave, who made sure the math in this thesis was as clear and understandable as possible and made sure I really understood whatever I was presenting to him. Their combined experience and gentle advice took me from a wild idea to a fully formed thesis.

Thank you to all of my friends and family who helped me reach the end. Most especially I would like to thank Ryan Gamblin whose steadfast friendship has meant everything over these past years and Jasmine Holland whose direct support on this thesis and getting through quarantine have made all of this possible. I would like to thank my parents who always encouraged me to stretch myself academically, even if that meant traveling so far away. I would like to thank my brother who patiently taught me how to code and who always answered my technical questions as they came up writing this thesis. Finally I would like to thank all of my friends, teachers, and professors who made contributions large and small to my education and my life. Without all of these people I could not be here, nor would I want to.

Contents

Abstract

The purpose of this thesis is to demonstrate via simulation the efficiency guarantees of the price of anarchy in first-price, single-item auctions. An adaptive agent-based model is used to demonstrate the efficiency properties of a first-price auction where each agent learns to play from a fixed set of bidding strategies using a no-regret learning algorithm.

The simulations find that agents using known Bayes-Nash equilibria strategies converge to a price of anarchy greater then the lower bound given by theory. Next, we see that agents who use an arbitrary bidding strategy still achieve high efficiency in both symmetric and asymmetric auctions despite their lack of intelligence, and that as the number of bidders increases these auctions approach full efficiency. Finally, we see that no-regret learning agents converge to efficiency outcomes within the bounds given by price of anarchy theory and better than their no-learning arbitrary bidding counterparts.

The results from this experiment both demonstrate the correctness of the price of anarchy theory and give high confidence in efficiency of first-price single-item auctions. While even arbitrary agents produce semi-efficient outcomes, we see that agents who are capable of learning and reflecting human behavior converge to highly efficient outcomes.

Introduction

When people act rationally, society can suffer. Everyone looking out for their own best interests and following the incentives of the structures around them can lead to an outcome no one would have chosen at the outset. Economists often talk about the tragedy of the commons, a scenario where people follow their incentives to over-utilize a public good and ruin it for society at large. In the tragedy of the commons, each agent's rational behavior is what leads the system to converge to overuse, wasted resources, and utility lower in the long run than if they had been able to preserve the resource. While the idea of the tragedy of the commons is about how incentives can lead to the overuse of public resources, there is a larger problem at the heart of it: how and why is it that when people behave strategically it often leads to worse outcomes for both the individual and society as a whole? These questions are what economics is all about. What are the effects of selfish behavior on markets and when can we expect this behavior to lead to good outcome, i.e. when do the incentives of the market line up with the public good?

The fact that selfish behavior leads to a worse outcome in the tragedy of the commons is a product of the system it takes place in. Because the incentives of this game line up with individualistic choices that are bad for society, rational agents will always choose to over-utilize the public good. In this case, it would only be if some outside force stepped in and made people behave correctly that these agents could maintain behaviors that would lead to the best outcome for society in the long run. Socially optimal choice for all agents is not an equilibrium when agents behave rationally in this case, but we can still imagine what it would be like if agents were forced to make socially responsible choices by some external force. This is the technocrat's dream, to have some *deus economica* control the actions of every agent to maximize some empirically derived social welfare function: every action pre-planned and every util maximized for the good of all. In lieu of an economic god ruling us, maybe some benevolent dictator could be set up instead?

Unfortunately, even if a benevolent dictator could be found, who is to say that they will be able to calculate the exact optimal solutions for large scale problems? Imagine a nation's government trying to calculate the socially optimal number of shoes to manufacture each year for millions of people each with different shoe sizes and preferences. It seems doubtful that any government would be capable of doing this as there is too much data to process and too many changing circumstances to react quickly to societal need. Rather, most countries leave shoe manufacturing to a free market which has firms that can pop up and respond to the demands of customers at will. That is not to say that the free market is perfect for shoes: we have no way of knowing if its allocation of resources ends up being the best possible. In fact, it is highly doubtful that it is. There is only one socially optimal outcome (which presumably our omnipotent god of economics could calculate), but there exist an infinite number of outcomes which are worse than optimal. Our question is one of trying to understand what makes some markets or systems behave in a way that is good for society and some terrible for society when everyone is behaving selfishly, or, how we can describe the effect of selfish behavior on a system. One way of approaching an answer to this question is analyzing how agents behave in the worst case (from a social welfare perspective) for a given system or game. That is, we want to bound the system's worst possible social cost when people are behaving rationally. This is called the *price of anarchy*, the worst possible price paid in terms of social welfare at the equilibria for rational agents within that system. How much worse can society's welfare get when everyone is making their own decisions looking out for their own best interests when compared with the best possible outcome where everyone's actions are controlled by a centralized planner?

The idea of reaching an equilibrium for a given system is also problematic. How are we to presume that agents converge to an equilibrium? Are they able to easily calculate it and decide to go there to begin with? No. We should be concerned with the set of outcomes or equilibria that happens not only for fully rational agents with infinite computational power, but also with agents who learn how to behave as they participate, who dynamically react to the changes in strategy by the other players.

This thesis is about simulating the price to society for anarchic behavior within a system (game) of agents who are learning how to maximize their own utility as they go. Computer scientists and economists have been proving the bounds of this price for various systems both at the fully rational equilibria and at the equilibria that certain learning agents converge to. We create a simulation to demonstrate this bound in one of the most simple markets, a sealed bid first-price, single-item auction. This is an auction where everyone bids simultaneously, one-shot, on a single item and the highest bidder gets the item. Thus, by constructing a simulation of an auction in this way, we can be more sure that when this auction format is used in real life that these auctions are producing outcomes that are at least so efficient. This will also give us a way to explore the ideas of the price of anarchy and algorithmic game theory by seeing how the theory behaves when it plays out in a simulation.

Chapter 1

Auctions and Computational Economics

Computational economics has been on the rise in the last three decades. This branch of economic research encompasses two major ideas. One is that the increasing power of computers can help solve and understand classical economic problems through increasingly more complex simulations and numerical analyses. The other is that the mathematical methods developed in the field of theoretical computer science can be used to explore the algorithmic and computational complexity of existing models in economic theory. I aim to take elements from both of these frameworks, simulation and computational theory, to better understand the efficiency of one shot (first-price, sealed-bid) auctions at equilibrium.

1.1 Auctions, Equilibria, and Anarchy

It is perhaps obvious why economists would be interested in studying auctions. Auctions are one of the most basic market structures. Auctions have roots going back to at least the ancient Greeks and are still pervasive today in places such as art auctions, car auctions, Ebay, and other markets where sellers want to see what price they can get for an item (Mochon and Saez, 2015). In the past 15 years, economists have been joined by computer scientists who are increasingly interested in the strategic interactions of agents within this setting. Christos Papadimitriou said in a 2015 lecture at the Simons Institute that it was the advent of the internet, an artifact out of computer scientist's control, that turned theoretical computer science into a "physical science." Now computer scientists had to "approach the internet with the same humility that economists approach the market..." He went on to say that "it also turned us[computer science] into a social science. It was obviously about people and incentives. Without understanding this, you cannot understand the internet"(Papadimitriou, 2015). It is within this framework that he says computer scientists first began to study auctions as they existed on the internet most notably Ebay and Google's sponsored search auctions. Now, computer scientists are moving beyond the internet, taking the mathematical tools of their discipline and applying them as a lens to understand and explain the world. This thesis is looking at the field of algorithmic game theory: the study of the algorithms and complexity of strategic interactions. For economists, this can be thought of as a new toolbox for unpacking and understanding the models and structures that already dominate the field. For example, in the case of a Walrasian auctioneer who calculates the clearing prices of a combinatorial auction, their problem was shown to be NP-complete, a computational complexity class where the best you can do usually is guess and check.[1] Calculating Nash equilibria for games was shown to be a PPAD-complete problem, a complexity class that lies somewhere between polynomial and nondeterministic-polynomial time meaning that this problem can be quite hard (Papadimitriou, 2015)! Using computa-

[1]Informally, time complexity can be thought of as how the number of operations it takes to solve the worst case instance of a problem grows as a function of its input size. The time complexity class NP stands for "nondeterministic polynomial" (time). It is the class of problems that if given solution, the correctness of that solution can be checked in polynomial time, i.e. $O(n^k)$ operations where n is the size of the input and k is any positive integer. A problem being NP-complete means that all problems in NP reduce to solving this problem. NP-complete problems grow very quickly and are when you may hear "taking more time to compute than the age of the universe" etc.

tional complexity, economists can further question the assumptions they are making about rational agents. When we assume agents in our models are calculating the Nash equilibria, is it reasonable given the complexity class of this problem? If we are assuming our agents are solving computationally *intractable* problems, ones where the required computation time grow very, very quickly with their input size, we may want to reconsider how reasonable the model is as an approximation of human behavior.

1.1.1 The Price of Anarchy

One of the major ideas to come out of algorithmic game theory is the *price of anarchy* (POA), a mathematical way of showing the difference between the social welfare in the optimal case and in the worst case equilibrium for a game. This term was first coined by two computer scientists Elias Koutsoupias and Christos Papadimitriou, who were using the price of anarchy to understand network games and more generally were looking at how this concept could be used to understand behavior on the internet (Koutsoupias and Papadimitriou, 1999; Papadimitriou, 2001). More formally, the price of anarchy for a game is the ratio of the minimum equilibrium social welfare in a game over the best possible social welfare of the game (see Definition 2.6). An illustrative example of the price of anarchy from Roughgarden (2016, 15-16) is selfish routing games played on graphs such as the one pictured below in Figure 1.1.

In this game, each player chooses which edges to take on the directed graph from the source s to reach the terminal t. Each edge has an associated cost, $c(x)$, expressed as a function of the proportion of players who take that edge, x. Player's total cost is then the sum of the costs for each of the edges they took to reach t. Player's objective is to minimize their total cost of reaching the terminal by picking the best route they can. In the game depicted in Figure 1.1, we see that routes either cost, 0, 1, or x, where x is the proportion of the players who take that route ($x \in [0, 1]$) For example if 50% of the players take that edge then $x = 0.5$. This can be seen as analogous to

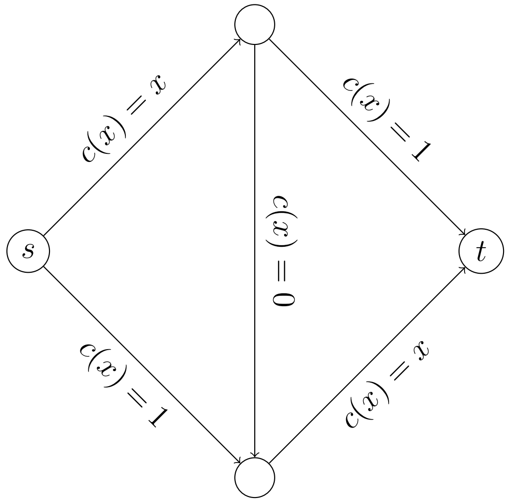

Figure 1.1: Routing game from s to t. Strategic interaction will make everyone worse off.

traffic when driving a car the more people take a road, the slower the traffic goes and the longer it takes to get somewhere. Knowing this, each player tries to choose which path to take to get to their destination as quickly as possible. As one can quickly verify, the best solution for society, minimizing the total driving time for all, is when half of the drivers take the top route, and half of the drivers take the bottom route costing in total 1.5 for each driver.

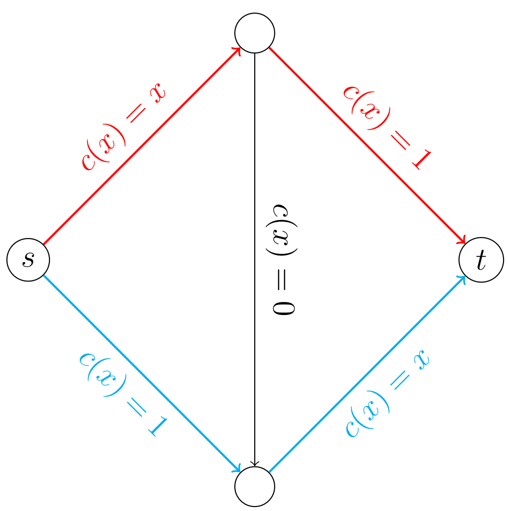

Figure 1.2: Socially optimal routing, not at equilibrium

However, this is not an equilibrium. The drivers taking the top route have incentive to instead take the middle edge to try and lower their total time traveled. In fact, the Nash equilibrium will end with all of the drivers taking the top edge, $c(x) = x$, going through the middle edge, $c(x) = 0$, and then the bottom edge $c(x) = x$. Only then will no players have reason to deviate.[2]

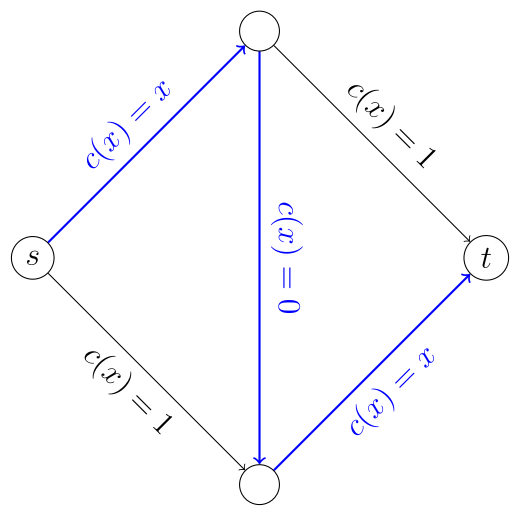

Figure 1.3: Route under strategic interaction, the only Nash equilibrium

These actions where agents try to strategically decrease their own cost end up producing an outcome that is worse for every player individually and society as a whole. It now takes time 2 for each player to reach the terminal since every player is taking both $c(x) = x$ edge. No player has incentive to deviate from this strategy since they would be no better off individually taking a different route. Thus, under strategic interaction we see that the equilibrium is *sub-optimal* for society: this set of player choices does not maximize the social welfare (by not minimizing the social cost). Moreover, in this specific case all individual players will be worse off under the equilibrium outcome than under the socially optimal one! So, the ratio of the social cost (total costs to all players) for the worst possible equilibria compared to the best

[2] A careful observer will note that if the edge with cost 0 were not there, this would not be a problem. This is called Braess's Paradox: having this extra edge counter intuitively leads to worse outcomes (Braess, 1968)! This paradox is commonly cited as a reason why building more roads can lead to worse traffic.

possible, socially optimal (but not necessarily equilibria) outcome can be computed

$$\frac{\sum_{n=1}^{N} n \cdot 2}{\sum_{n=1}^{N} n \cdot 1.5} = \frac{4}{3},$$

where we sum the cost for our N players in each of the two scenarios described above depicted in Figure 1.2, and Figure 1.3 (Roughgarden et al., 2017).[3] Since this is a cost minimization game, $POA \geq 1$, and the closer the number is to one, the closer the worst case Nash equilibrium is to optimal. For this network, there is only one Nash equilibrium for selfish routing games, however, price of anarchy bounds can be found for the class of game as a whole regardless of its individual construction. These bounds tell us how much worse then optimal we can expect a system at equilibrium to behave in the worst case. The price of anarchy bound for the entire format of selfish routing games with affine cost functions (linear + translation, i.e. $c(x) = ax + b$) is known to be 4/3. This means that our example is as bad as the price of anarchy can get for any network we construct to play a routing game (Roughgarden, 2007).

1.1.2 Auctions

Researchers in algorithmic game theory have recently been applying the lens of the price of anarchy to understand the efficiency of auctions. This pursuit should seem very familiar to economists who traditionally want to understand the efficiency of market structures. To understand how the price of anarchy framework is helpful for auctions, it is first worth understanding the basics of auction theory and how it is that inefficiencies could arise.

An *auction* is a market mechanism, operating under specific rules that determine to whom one or more items will be awarded and at what price. To do this, an

[3] This is a cost minimization game so POA = EQ-social-cost/OPT-social cost. Definition 2.6, is for auctions, a payoff maximization game, so POA = EQ-social-welfare/OPT-social-welfare. In general it is important to check if higher numbers are a good thing in that context.

auctioneer solicits a bid, or a sequence of bids from each of the players for the items being auctioned and then uses some fixed rule (or mechanism) to determine who is awarded the items and how much each bidder must pay. While we may traditionally think about ascending auctions for a single item, called an *English auction*, where the auctioneer keeps soliciting higher and higher bids until no one is willing to bid anymore, there are in fact many different kinds of auctions! There can be many different rules for determining who the winner is, or what price each of the bidders must pay (regardless of if they win or not) (Mochon and Saez, 2015).

I use the notation from Mochon and Saez (2015) to express auctions as mathematical objects. An auction consists of a set of n *bidders* who are trying to buy the object and a *seller* who is trying to sell the object. For each bidder i in an auction, their *valuation* v_i is how much they value the item. This is sometimes called a private valuation as this value is generally unknown by the other participants in the auction. The *bid*, b_i, is the offer that bidder i submits for an item. *Sincere bidding* is when $v_i = b_i$, *underbidding* is when $v_i > b_i$, and *overbidding* is when $v_i < b_i$. The highest bid made by any bidder is denoted b^*, and under the highest price mechanism, b_i^* is also the winning bid (if multiple bids equal the winning bid, then some tie-breaking rule must be used).[4] The *selling price*, p^* is the final price that the bidder actually pays for the item (which, depending on the auction type, need not equal b^*). Before the auction begins, each bidder only knows their private valuation for the item, and the probability distribution from which they believe the other bidders will be sampling their valuations. This probability distribution helps model the fact that in auctions players don't know how the exact valuation for other players, but they still generally have an idea how they might value it (see Section 2.1 for more on this). During the auction, the auctioneer solicits a number of bids from each of the player depending on

[4]We could in principle also have a second (or j-th) highest price mechanism meaning that the second (or j-th) highest bidder would win the item. Obviously knowing this will change how bidders play the game.

the format of the auction. For example in a sealed-bid auction each bidder secretly submits only a single bid to the auctioneer.

The way in which bidders decide how they are going to bid is called a *strategy*. A strategy is a mapping from their private valuation v_i to their bid b_i. After the auction, the bidder i wins the item if their bid is higher than the bid placed by any other bidder, $b_i > \max_{i \neq k}\{b_k\}$. In a single item auction, the *income of the bidder i* is equal to their value of the item:

$$\Gamma_i^* = v_i,$$

and the *surplus of the bidder i* is equal to the difference between income and price paid:

$$\Pi_i^* = \Gamma_i^* - p^*.$$

If a bid placed by a bidder is less than the winning bid, they do not win anything and their income and surplus are both zero. The *seller's revenue* in a single unit auction is equal to the price paid by the winning bidder:

$$\mathcal{R} = p^*.$$

There are multiple different pricing rules in an auction that determine to whom the item gets allocated. In a *first-price* auction the winning bidder pays the amount of their bid, which is the highest bid of the auction: $p^* = b^*$. This first-price pricing rule is sometimes referred to as *pay-what-you-bid*. In a *second-price* auction, the winning bidder pays an amount that is equal to the second highest bid for the awarded item (Vickrey, 1961).[5] In sealed-bid auctions with a first-price rule, people must bid lower than their valuation if they want to get any surplus. Thus, agents in these auction formats spend their time reasoning how much less than their true valuation

[5]The second-price, sealed-bid auction is also known as a Vickrey auction after the economist who invented it.

they should bid. Because no rational bidder would bid their true valuation and get zero surplus, this leads to strategic interaction between the agents deciding how much they are willing to shade their bids: the lower the less chance of winning, but the higher their surplus. This can potential lead to *inefficiency*, when someone who does not value the item the most wins the bid and is awarded the item. This stands in contrast with a second-price auction which encourages bidders to bid their true valuations (as they will gain positive profits if they win, no matter the second highest bid). This pricing mechanism aligns the incentives for bidders, bidding truthfully, with an efficient allocation of items (Mochon and Saez, 2015). However, I am concerned with inefficiency in auctions, not ones which guarantee optimal social outcomes. Thus, I will confine ourselves to first-price sealed bid auctions as this simple auction has sometimes inefficient bidding dynamics which price of anarchy analysis can shine a light on. Moving forward, the paper will refer to sealed bid first-price auctions simply as first-price auctions as is common in the literature.

Before discussing the known bounds on the price of anarchy for first-price auctions, it is worth further investigation into how the bidding might lead to non-optimal or inefficient outcomes, and what that means. Because the other bidders' valuations are unknown, each bidder must act under uncertainty when making their decision of how to bid. If someone bids their exact private valuation, they will get zero surplus and thus zero utility. In order for them to get some utility for the item they must be paying less then the exact amount they value the item. How much each bidder should bid less than their valuation, or *shade*, their bid is determined by how much they think the other parties value the item. To capture this interaction, auctions are represented as Bayesian games (see Definition 2.1) where each bidder is drawing their valuation from some *a priori* distribution. If one player knows that the other is drawing from a distribution with a smaller mean than they are (i.e. probably does not value the item as much), the equilibrium for this game, the point where no one wants to change their strategy, will be where the bidder who expects they have a higher valuation isn't bidding as aggressively as the bidder who expects they have a lower valuation. In each of them doing so, they can maximize their expected payoffs based off of their beliefs for how the other player will bid. Importantly, this difference in shading can lead to the person who values the item most losing if they shaded their bid too much. Hence the *social welfare*, the summed surplus of all bidders and the seller (Definition 2.4), can be less than optimal at the equilibrium.

Price of anarchy analysis for the first-price auction format has already been done, but the proofs are incomplete. Syrgkanis and Tardos (2013) proved that the lower bound of the price of anarchy in first-price auctions is at least $1-\frac{1}{e} \approx 0.63$. The exact upper bound on the price of anarchy for first-price auctions is still unknown, but it has been proven that it can be no better than 0.87 (Hartline et al., 2015). These bounds on the price of anarchy are true across the format regardless of how many bidders there are or what distributions they are drawing their bids from. Therefore, for any first-price single-item auction at equilibrium, we can say that the social welfare for any equilibrium must be at least 63% of the optimal social welfare (the maximum possible social welfare achievable in this system).

This moves us closer to understanding the social welfare produced by first-price single-item auctions, but these results do not take into account how the players arrive at these equilibria. Also, in the real world, we may expect that players play repeatedly in the same auction and learn how to bid as they play rather than come into each one with pre-computed strategies. This seems especially likely when computing the equilibrium is computationally hard and the stakes of each individual auction are small, pre-computing just might not be worth the effort. Given these observations, it is natural to ask questions about how the efficiency results carry over to adaptive game environments. One notion of learning commonly used in both theoretical computer science and game theory is *no-regret learning*. An algorithm for a player satisfies the no-regret condition if, in the limit as the number of times the game is played goes to infinity, the average reward of the algorithm is at least as good as the average reward for the best fixed-action in hindsight (assuming the sequence of actions for the other players remains unchanged).[6] The no-regret condition is not in itself a technique for learning, but rather a characteristic of learning algorithms where players learn to play from a finite set of options. A player's algorithm satisfies the no-regret condition if they perform - on average - at least as well as if they had picked the best possible fixed-action at the beginning of play. If each player incorporates this kind of learning algorithm players' strategies will converge together into a large, relatively permissive class of equilibrium called *coarse correlated equilibrium* (see Definition 2.12) where each player conditions their response on the expected action of the other player (Blum and Mansour, 2007). Remarkably, Roughgarden et al. (2017) produced a way to extend the price of anarchy bounds proven for Bayes-Nash equilibria to the larger set of coarse correlated equilibria. Hence, we know that the price of anarchy for the set of coarse correlated equilibria for first-price, single-item auctions are also at least 0.63, and that if the bidders in the auction are no-regret learners, they can arrive at this equilibrium through the course of play.

We now have enough background information to state the goal of this thesis: to simulate no-regret learning algorithms for agents in first-price auctions to demonstrate the price of anarchy of this format. Because algorithms exist that guarantee convergence to some equilibrium set, we can construct computer agents to carry out simulated auctions. Each agent will use the strategies dictated by the no-regret learning algorithm; because of the work on this literature, we know that we will see them converge to a coarse correlated equilibrium and we know that the efficiency of this learned equilibria must be at least 63% of the optimal social welfare.

[6]That is to say that the algorithm will converge to having an average utility no worse than any fixed strategy we would have rather picked in hindsight after seeing the outcome for each of their actions (see Definition 2.10).

1.2 Simulated Agents and Simulated Economies

The history of computers in economics, as outlined in Backhouse and Cherrier (2016), goes all the way back to general purpose computers invented in the 1940's. At that time, Wassily Leontif used a computer to invert a 39×39 matrix for his input output model. Since then, the use of computers in economics has exploded. With computers, economists are able to invert bigger matrices, do Monte-Carlo simulations, create multinomial probit models, use full information maximum likelihood estimation, and all of the other computationally intensive techniques that define modern econometrics. While we might say that one branch of computational economics is focused on creating stronger and stronger calculators to facilitate empirical research, another branch has focused on creating simulated economies that allow economists to construct a blended version of theory and research within a computer program. With a simulated economy, theories can be coded into the simulation which when run allow the researcher to conduct experiments that might not be practical to conduct in the real world.

One specific type of economic simulations is an *agent-based model*: simulated system of autonomous decision makers which are referred to as *agents*. These models are able to generate complex behavior even if only simple assumptions are made about the behavior of the coded agents. That is, these agents interacting with each other in complex ways are able to produce emergent phenomena in the macro structure of the system.

For example, in the late 1960's and early 1970's, Thomas Schilling created computer simulations in an attempt to understand how and why self-segregated neighborhoods formed. He coded a virtual environment where agents were given a simple preference: they are only happy if they are not in the minority population in their neighborhood and will keep moving otherwise (Schelling, 1969). This simple model can illustrate the main tenets of agent-based modeling. I review a simplified implementation of this model outlined in Thomas Sargent and John Stachurski's 2019 lectures in quantitative economics.[7] First, we have agents whose preferences are simple enough that they might be considered representative of people in the real world: they are only "happy" if half of their closest neighbors are the same as them. If they are not happy, they will move somewhere else arbitrarily. These preferences can be modeled as the short procedure shown below, where L is the space they live in:

1. draw a random location in L,

2. if happy at new location, move there,

3. else, go to step 1.

In this case, we get to choose what that environment looks like and—like Shelling—we can say that it is a one by one unit square and that agents neighbors are the ten closest people to them (in Euclidean distance) on that square. I ran this simulation (after constructing it in Python) with green and orange dots representing the types of people. There are 250 agents (125 green and 125 orange) in our simulation, and I conducted five cycles of the above procedure going through each of my 250 agents until everyone was "happy" and stopped moving.

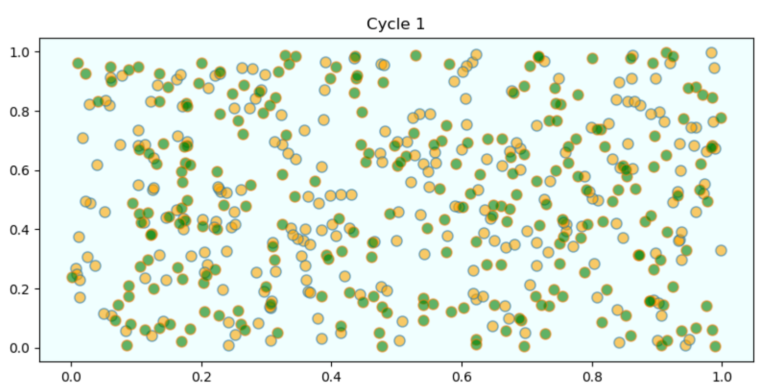

Figure 1.4: Schelling's Segregation Model: Cycle 1

As can be seen in cycle 1 (Figure 1.4), the agents are well distributed among each

[7]These lectures are a treasure trove of information on how to use Python to construct economic models. The simulation in chapter three was built in part using them as a guide.

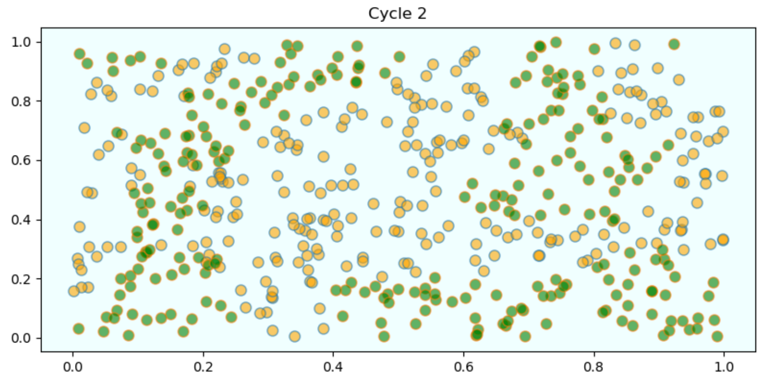

Figure 1.5: Schelling's Segregation Model: Cycle 2

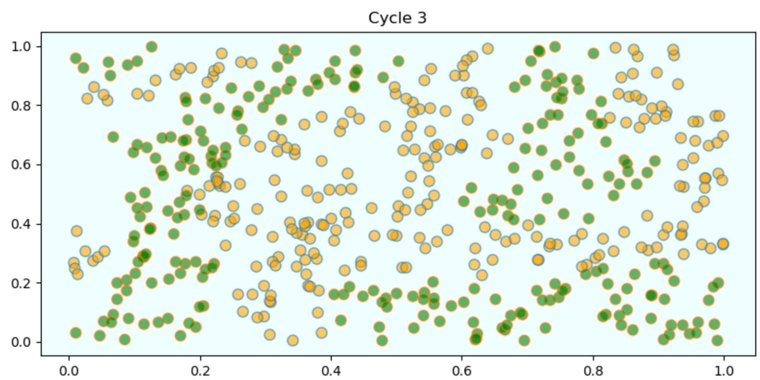

Figure 1.6: Schelling's Segregation Model: Cycle 3

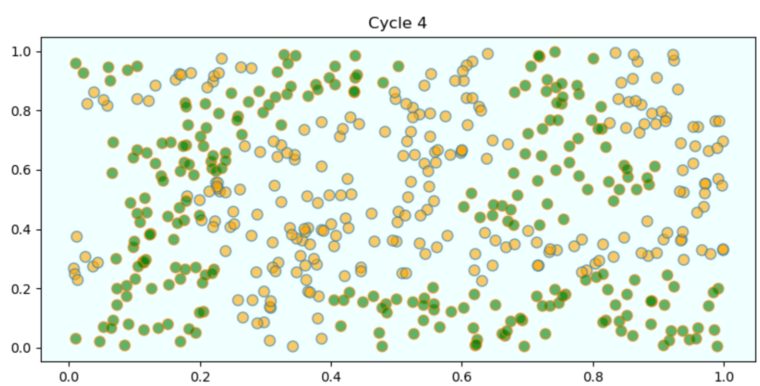

Figure 1.7: Schelling's Segregation Model: Cycle 4

other, but as they move in cycles 2-5, they become progressively more segregated. After five cycles all agents are happy and the simulation terminates. With very few assumptions about the agents' preferences, we can see the resulting emergent behavior

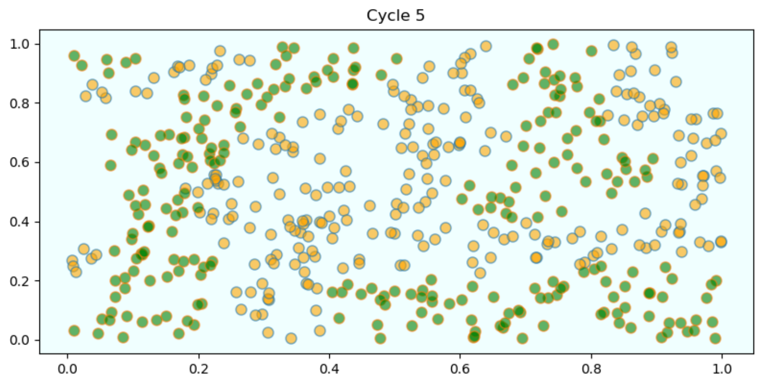

Figure 1.8: Schelling's Segregation Model: Cycle 5

of segregated neighborhoods in the system as a whole. Not only that, but the agents naturally move to an equilibrium as they adjust their behavior to what their neighbors are doing (Sargent and Stachurski, 2019).

Agent-based modeling has been used to build and understand much more complicated systems than the example illustrated above. The Santa Fe Institute in New Mexico is one of the main proponents of agent-based modeling, releasing a manifesto supporting the use of this technique to understand the *complexity* of economics from the ground up (Backhouse and Cherrier, 2016). They built the Santa Fe Artificial Stock Market in the 1990's in an attempt to simulate the behavior of agents on the stock market and investigate how using adaptive trading strategies affects the outcome of the market. The Santa Fe Artificial Stock Market is one of the first examples of agents learning and adapting to their environments as part of the simulation, and represents a move to model not only people's preferences in simulations, but also how they learn and adapt (LeBaron, 2002). In the Santa Fe Artificial Stock Market, agents used a genetic algorithm to adapt their trading strategy at each period by modifying a string (for example 00011100) where each bit in the string told the agent to use a certain behavior or not. Those trading strategies that did well were coded to survive longer, while the agents with worse strategies would randomly modify their own or copy a more successful agent's strategy (Joshi, 1998). This genetic algorithm was supposed to be representative of the learning of traders on the stock market so that the insights taken from running these agents in simulation could be applied to learn something about the real world (Arthur, 1992).

The authors of Santa Fe Stock Market paper suggest that their genetic algorithm was one of many algorithms that could be used to stand in for human behavior. They also thought that reinforcement learning or deep learning could also be used to stand in for human intelligence in a simulation.[8] More specifically, they suggest that appropriate algorithms can be chosen for agents in a simulation by employing empirical studies and seeing which algorithms give similar behaviors. If the algorithmically played games could not be told apart from the human played games, you had a good candidate (like a Turing test specific to what you are trying to simulate)(Arthur, 1991).

Unfortunately, research that has compared the behavior of human agents in strategic settings to that of these algorithms have not found any general learning algorithm that best approximates humans. From game to game, different algorithms more appropriately behave like humans who were experimentally playing the same games (Tesfatsion, 2002). Of course, having no algorithm to approximate human learning in a simulation is a problem only if the simulation is trying to replicate actual human behavior. We may instead want our agents to represent the "rational choice" in any given situation. Unfortunately, the learning algorithms we have today are not necessarily fully rational (in the economics sense of effectively maximizing their utility) either. While the artificial agents may try different options in an attempt to maximize their objective function (basically a coded utility function for AI) and end up with a solution that is really good, this is done through searching the ac-

[8] Reinforcement learning is when an algorithm plays a game repeatedly and updates its beliefs about what actions will lead to the best payoff. Deep learning uses deep neural networks to try and estimate the best outcomes in a fashion similar to regression. It's worth pointing out that deep learning hadn't exploded in popularity in the 90's when they were writing this paper. With this methods recent surge in popularity also came a lot more awareness of its drawbacks and limitations.

tion space as best they can rather than computing outright how they should behave. This generally leads to behaviors that are short of the analytically computable optimal best strategy. As Holland and Miller say in their 1991 paper "Artificial Adaptive Agents in Economic Theory", "Usually there is only one way to be fully rational, but there are many ways to be less rational" (pg 367). Since they recognize that the agents they are creating for their simulation probably fall in the "less rational" camp, they suggest to get around this is to try and build models that have robust behavior across agent learning algorithm choice. If most learning algorithms lead to similar results for whatever question the model is asking, one may be fairly confident that we are looking at a property of learning in the system that leads to the result you think you have demonstrated rather than the particular outcome reached by the specific algorithm you choose.

The literature of game theory provides another solution for trying to approximate human learning, no-regret learning. Because the learning outcome requirements that define these algorithms are relaxed, it isn't wild to suggest that human learning will perform at least as well. These algorithms are also generally quite simple, if something does well do it more, if not do it less. Most importantly, no-regret learning algorithms also allow us to combine our simulated models and analytical models. Because of the well-described properties of no-regret algorithms, one can describe and prove outcomes of the systems in a stronger way than we ever could using a simulation. With a simulation, unless the simulation tries all possible cases, it is hard to be definitive. With mathematical analysis of no-regret learning it is possible to say interesting things things about all possible outcomes. Because there exist algorithms proven to satisfy the no-regret property, it is possible to code this behavior into the agents in simulated economies and perhaps understand the system better.

Moving forward, this thesis aims to better understand the efficiency of auctions by simulating them using no-regret learning agents. The main aim of this simulation will be to demonstrate the price of anarchy bounds proven by Roughgarden et al. (2017), and then to beyond that see how efficient we might expect these auctions to be (not just in the worst case) by looking at the results of the simulated economy. I will do this with first-price, sealed-bid auction as they are the best understood theoretically and simple to code.

Chapter 2

Price of Anarchy in First-Price Auctions

This chapter lays out the mathematical framework for modeling auctions as Bayesian games of incomplete information, formally defines the price of anarchy in auctions, and shows the bounds on the price of anarchy for sealed bid first-price, single-item auctions (which I will simply refer to as first-price auctions). It then follows up on this framework of analysis to show how these bounds can be extended to agents competing in auctions using learning algorithms with a property called "no-regret" learning. This chapter follows the work of Tim Roughgarden, Vasilis Syrgkanis, and Éva Tardos, three computer scientists active in the field of algorithmic game theory who jointly authored the 2017 paper "The Price of Anarchy in Auctions," a survey of the topic. This chapter will be giving the important theorems and ideas that primarily come from this paper and the individual works of these three authors, explaining them, demonstrating proofs where appropriate, and reconstructing the important ideas that we will be using in our simulation.

2.1 Auctions as Bayesian Games

Auctions are typically modeled as *Bayesian games*—games of incomplete information. While it is obvious why an auction consisting of the strategic interaction of bidders could be modeled as a game, how it should be modeled requires some thought. After all, while each bidder knows their own valuation of the item being sold, unless for some reason (and against their own interest) the other players announced their valuation of the item before the auction began, our bidder will not know how the other players value the item. In fact, it is precisely this lack of information in first-price auctions that makes them strategically interesting! If all players came into a first-price auction knowing the valuation of other players (assuming they are all distinct), the bidder with the highest value would always win by bidding the valuation of the next highest bidder (or slightly higher). It is the uncertainty that players face about the valuation of other players that forces players to guess how much they should shade their bid (as, again, no rational player will ever bid at or above their valuation) based off of what they know about other players.

But what can we say that bidders know about the valuations of other bidders? Certainly they do not know nothing as we all have reasonable expectations about what some item is worth to others. No one is likely to value a candy bar at a million dollars. But a collector of candy bars might value it at a higher value than an ordinary person who just wants to eat the candy bar. For each bidder, we can model them as drawing their valuation from some publicly known distribution. As an example, we might expect normal people's valuations of candy bars to be a normal distribution centered at \$1, but the collector's value might be a Laplacian distribution (thicker tails on the probability distribution) centered at \$5. This sort of strategic interaction where the players know the distributions which parameters of the game are drawn from are typically called *games of incomplete information*, or *Bayesian games*.

2.1.1 Bayesian Games

Bayesian games of incomplete information, introduced by John C. Harsanyi in 1967, are games in which one or more of the players lack "full knowledge" of the game that is being played. Rather than players knowing every parameter of the game such as utility functions, possible strategies, and information held by other players, each player knows a probability distribution from which these will be drawn before the game begins. In his paper, Harsanyi says that this type of game can be thought of as a normal game, where "nature" goes first, drawing from a piori probability distributions and assigning values before play begins without the players knowing which specific variation of the game they are playing (Harsanyi, 1967, 159). Importantly, each player does know the probability distributions from which each value is selected. Formally, slightly modifying the definition given in Nisan (2007), Bayesian games are defined as follows:

Definition 2.1 (Bayesian Game)**.** A game with (independent private values and) *incomplete information* on a set of n players consists of:

1. For each player i, a set of *strategies* S_i, letting $S = S_1 \times \ldots \times S_n$.
2. For every player i, a set of *types* T_i, and a prior distribution $\mathcal{F}_i$ on T_i. A value $t_i \in T_i$ is the private information that i has, and $\mathcal{F}_i(t_i)$ is the a priori probability that i gets type t_i. $T = T_1 \times \ldots \times T_n$ and $\mathcal{F} = \mathcal{F}_1 \times \ldots \times \mathcal{F}_n$.
3. For every player i, a *utility function* $u_i : T_i \times S \to \mathbb{R}$, where $u_i(t_i, s_1, \ldots s_n)$ is the utility achieved by player i, if their type is t_i, and the profile of strategies played by all players is $s_1, \ldots s_n$.

We break down this definition before we apply it to auctions. For each player i, there is some set of strategies, or actions that they can take in the game, S_i. These strategy sets need not be finite, and they need not be the same for each player. In rock paper scissors of example $S_i = \{rock, paper, scissors\}$ for the two players $i = 1$ and $i = 2$. Now, we create a larger set S, the *strategy space*, which is the Cartesian product of each individual players strategy sets. Elements of this set will then be the different possible strategy combinations. So, in the rock paper scissors example we get $S = \{rock, paper, scissors\} \times \{rock, paper, scissors\}$, and we can sample $\mathbf{s} \in S$ such as $\mathbf{s} = (rock, paper)$ or $\mathbf{s} = (rock, rock)$. We call these vectors $\mathbf{s}$, *strategy profiles*, where $\mathbf{s} = (s_1, \ldots, s_n)$. Often, game theorists want to understand how a player i would choose their strategy based off of the strategy of others. Hence, as is common in the literature, we let $\mathbf{s}_{-i}$ denote a strategy profile with the *ith* element removed, the ordered vector of everyone else but player i's strategies. In rock paper scissors if $s_{-i} = rock$, player i would probably want to play paper, $s_i = paper$. Similarly we define the *bid profile* as the vector $\mathbf{b} = (b_1, \ldots, b_n)$, and the *valuation profile* as the vector $\mathbf{v} = (v_1, \ldots, v_n)$ where the ith element of each vector corresponds to the bid or valuation of the ith bidder.

The second part of Definition 2.1, is what separates Bayesian games from normal mathematical games. We have a set of types for each player and some probability distribution over this set that tells how likely each player is to be of a certain type. We say that the probability distribution is a priori on the set of types, which means that the probability of each type is known to the other players before the game starts. Here, for an illustrative we will examine card games such as poker or five card draw. The type of each player would be their current hand of cards. Players only know their own cards, not the other players. They do, however, a priori know how likely each possible hand is for the other player. In a card game, the probability distribution for types are not independent. If one player draws the ace of spades, the probability that the other player has a hand containing that card is zero (supposing they are using only one standard deck). To simplify talking about this, we will generally speak about drawing some type t from $\mathcal{F}$.

The third part of Definition 2.1 is the utility function, u_i for each player. This is a function that maps the strategies played by each of the players (which gives the outcome of the game), and the type of player they are to a real number—their "utility" for this outcome. We will often write $u_i(t_i, \mathbf{s})$ instead of $u_i(t_i, s_1, \dots s_n)$ for convenience. A players utility function, u_i, can be defined deterministically or probabilistically. For example as is common in the case that both players play strategies that leads to a tie, say $\mathbf{s} = (rock, rock)$ for rock-paper-scissors, one might let the tie be broken arbitrarily and have the winning player get the utility.

2.1.2 First-Price Auctions as Bayesian Games

We now move to using the formal structure of Bayesian games to model auctions. Here the types of players will be their valuation, v_i. As discussed in chapter one, v_i represents how much bidder i values the item being bid on. There is also $\mathcal{F}_i$, which for an auction is the publicly known distribution over the set of valuations from which bidders are drawing their valuation for the item being auctioned. For example, if each of the players were uniformly drawing their valuation from $[0, 1]$, then their type t_i, would be $t_i \in T_i = \{x | 0 \leq x \leq 1\}$ and $\mathcal{F}_i$ is the uniform distribution over T_i. To avoid confusion in the notation, we will simply speak of players drawing their valuation v_i from their distribution $\mathcal{F}_i$ which we will write as $v_i \sim F_i$, and avoid talking about the type of the bidder. Therefore, in Bayesian auctions, bidders know their own valuation of the item being bid on and the distribution from which each of the other players are drawing their valuations. In a first-price auction, if player i wins with bid b_i, we define their utility to be $u_i = v_i - b_i$, the difference between their valuation and their bid. The losers all get $u_i = 0$ as they did not receive the item. Using the language of auctions, this is assuming the bidders are receiving utility equal to their surplus. A strategy for a player is a function $s_i \in S_i$ that maps a valuation v_i in support of $\mathcal{F}$ to a bid, $b_i = s_i(v_i)$ (i.e. taking into account the non-zero parts of the probability distribution for the other bidders valuations).

We now move on to the idea of equilibrium in this system. In games of complete information, the central equilibrium concept is usually a Nash equilibrium, the set of strategies in which each individual player cannot increase their utility by deviating from their strategy all else constant. That is, for every player, if no one else changes strategy, their best option is to stay where they are, hence an equilibrium. This concept is now updated to give us a Bayes-Nash equilibrium where we must also incorporate the distributions which players are drawing from. For an auction, Bayesian-Nash equilibria are defined as follows by Roughgarden et al. (2017):

Definition 2.2 (Bayes-Nash Equilibrium Auctions)**.** A strategy profile $\mathbf{s}$ constitutes a *Bayes-Nash equilibrium* if for every player i and every valuation v_i that the player might have, the player chooses a bid $s_i(v_i)$ that maximizes her conditional expected utility where the expectation is over the valuations of the other players, conditioned on a bidder i's valuation being v_i.

This is an equilibrium because, for the given strategy profile $\mathbf{s}$, every player i is maximizing their expected utility with their current choice of s_i. Hence, no player has incentive to deviate from their chosen strategy with this profile.

Now, if for a bid profile $\mathbf{b} = (b_1, \dots, b_n)$, we let

$$x_i(\mathbf{b}) = \begin{cases} 1 & \text{if player } i \text{ is the winner} \\ 0 & \text{otherwise,} \end{cases}$$

then the utility a player i receives when their valuation is v_i is

$$u_i(\mathbf{b}; v_i) = (v_i - b_i) \cdot x_i(\mathbf{b}),$$

and we can say that a strategy profile $\mathbf{s} = (s_1, \dots, s_n)$ is a Bayes-Nash equilibrium if and only if

$$\mathbb{E}_{\mathbf{v}_{-i}\sim\mathcal{F}_{-i}}[u_i(\mathbf{s}(\mathbf{v});v_i) \mid v_i] \geq \mathbb{E}_{\mathbf{v}_{-i}\sim\mathcal{F}_{-i}}[u_i(b_i^{'},\mathbf{s}_{-i}(\mathbf{v}_{-i});v_i) \mid v_i]$$

where the expectation is over the prior distribution $\mathcal{F}$ (Roughgarden et al., 2017). So, the above equation is one way of symbolically writing Definition 2.1 for auctions.

2.1.3 Example First-Price Auction

We now turn to an example auction to clarify what has just been laid out. In this example we analyze an auction between two players, Alice and Bob, who are bidding on a candy bar where each draw their valuations from the uniform distribution on $[0, 1]$. This is a first-price, sealed-bid auction where they each submit a bid for a (single) candy bar simultaneously. How are Alice and Bob supposed to decide what to bid on this auction? Supposing that Alice and Bob can calculate a Bayes-Nash equilibrium for this example, and that this equilibria is unique, they would know that their best strategy would be to bid their strategy at this equilibrium. But how would bidders go about calculating the equilibrium? We recreate the proof of the existence of such an equilibrium as from Nisan (2007). An example of proving the uniqueness of this equilibria appears in Levin (2002).

Proposition 2.3. *In a first-price auction among two players with prior distributions distributions of their valuations v_1, v_2, uniform over the interval $[0, 1]$ the unique Bayesian-Nash equilibrium is $\boldsymbol{s} = (s_1(v_1) = v_1/2, s_2(v_2) = v_2/2)$.*

Proof. (Nisan, 2007) First we show that this is a Bayesian-Nash equilibrium. Let us consider which bid x is Alice's best response if Bob uses bidding strategy $s(v_2) = v_2/2$, where Alice's valuation is v_1 and Bob's is v_2. The utility of Alice if she wins is $v_1 - x$ if she wins and pays x, and 0 if she loses. Thus, her expected utility from a bid x is $\mathbb{E}[u_1] = \Pr[\text{Alice wins with bid } x] \cdot (v_1 - x)$, where the probability is over $\mathcal{F}_2$, the prior distribution of v_2. Now, Alice wins if she outbids Bob so, $x \geq s_2(v_2) = v_2/2$.

Since v_2 is distributed uniformly in $[0,1]$ we can calculate the probability she wins with a bid x:

$$\Pr[\text{Alice wins with bid } x] = \begin{cases} 0 & \text{if } x < 0, \\ 2x & \text{if } 0 \leq x \leq 1/2, \\ 1 & \text{if } x > 1/2. \end{cases}$$

We see that the optimal value of x is in range $0 \leq x \leq 1/2$ as $x = 1/2$ gives higher utility than any $x > 1/2$, and since any $x < 0$ will give utility 0. Thus, to optimize the value of x, we find the maximum of the function $2x(v_1 - x)$ over the interval $0 \leq x \leq 1/2$. Taking the derivative with respect to x and setting this equal to zero, we get $2v_1 - 4x = 0$ (noting that the second derivative is negative, and thus this is a maximum). This equation has solution $x = v_1/2$. $\square$

2.1.4 Known Bayes-Nash Equilibria of First-Price Auctions

Bayes-Nash equilibria for single-item auctions have been solved for various choices of the number of players n, and prior valuation distributions $\mathcal{F}$. For example, in *symmetric auctions* there are n bidders who have uniform prior valuation distributions over the same interval, Chawla and Hartline (2013) showed that the Bayes-Nash equilibrium is when every bidder i uses strategy $s_i(v_i) = \frac{n-1}{n}v_i$. That is, if each player bids $b_i = s_i(v_i) = \frac{n-1}{n}v_i$, then no one will have any reason to deviate. This is a generalization of what we showed above in the case of two players; if we plug in $n = 2$ we get a resulting strategy for each player i ($i \in \{1, 2\}$) of $s_i(v_i) = v_i/2$. The Bayes-Nash equilibria for symmetric first-price auctions means that as the number of players increases, each player must shade their bid less and less, bidding closer and closer to their true valuation. The Bayes-Nash equilibria for symmetric auctions are *efficient*, meaning that the item will always be allocated to the player with the highest valuation. Efficiency is what economists want out of their auctions. When the item is always awarded to the person who values the item the most, it seems like a reasonably fair way to allocate the item, which is what economics is all about! However, in general, efficiency is not guaranteed at Bayes-Nash equilibria of first-price auctions. For example if we conduct an auction with two bidders, one choosing from the uniform distribution $(0, 1]$ and the other from the uniform distribution $(0, 2]$ as demonstrated by Krishna (2002), the Bayes-Nash equilibrium for this auction is:

$$
\begin{aligned}
s_1(v_1) &= \frac{4}{3v_1}\left(1 - \sqrt{1 - \frac{3v_1^2}{4}}\right) \\
s_2(v_2) &= \frac{4}{3v_2}\left(\sqrt{1 + \frac{3v_2^2}{4}} - 1\right)
\end{aligned}
$$

Here, bidder one (who has the smaller prior valuation distribution) knows that bidder two is more likely to have a higher valuation than them. Thus, player one must bid higher relative to their given valuation if they expect to win, and so they shade their bid less than bidder two. This can lead to bidder one drawing a lower valuation than bidder two, but outbidding them regardless and winning the item. For example if $v_1 = 1$ and $v_2 = 1.1$, then $s_1(1) \approx 0.667$ and $s_2(1.1) \approx 0.462$. This is inefficient since bidder one wins the item even though they have a lower valuation. From an economist's perspective this presents a problem—sometimes the structure of this market leads to inefficient outcomes, and sometimes it doesn't. How can we figure out how inefficient this market structure is?

Unfortunately, it has been shown that solving many of these asymmetric Bayes-Nash equilibrium requires finding a solution to a system of partial differential equations which have no closed-form solution (Roughgarden et al., 2017). Thus, even with infinite time we could not compute the Bayes-Nash equilibria for all possible variation of the auction. Having no closed form solution also suggests it may be unreasonable to expect bidders to "do their homework before an auction."[1] Thus, it would seem hard to characterize the efficiency outcomes of first-price auctions even though it seems like a simple format. However, the price of anarchy literature gives us a workaround. Instead of spending the time to characterize the efficiency of (solvable) Bayes-Nash equilibria, we can find a bound on the worst case equilibrium outcome!

2.1.5 Price of Anarchy in First-Price Auctions

In order to understand how inefficient first-price auctions can be, computer scientists have been applying a concept known as the price of anarchy (see Definition 2.6) to characterize system this system at equilibrium.[2] The price of anarchy is a way to compare the social welfare of a system or a game at its best possible value to that of its worst possible equilibrium under strategic play. Before moving on, we must mathematically define what we mean by "social welfare" if we are going to use it in our definitions. In the case of an auction, the social welfare is the sum of the utilities of the players, $u_i(\mathbf{s}, v_i) = (v_i - b_i) \cdot x_i(b_i)$, plus the revenue of the auctioneer, $\sum_i^n b_i \cdot x_i$.

Definition 2.4 (Social Welfare)**.** The *social welfare* of a bid profile $\mathbf{b}$ when the valuation profile is $\mathbf{v} = (v_1, \dots, v_n)$ is,

$$SW(\mathbf{b}; \mathbf{v}) = \sum_{i=1}^{n} v_i \cdot x_i(\mathbf{b}).$$

The price the winning bidder pays does not appear in this equation since the winning bidder is paying exactly as much as the auctioneer is getting, and this term

[1] How many times have you been at a sealed bid auction when someone shouts "Help. Help. My Bayes-Nash equilibrium strategy for this auction has no closed form?"

[2] Not that economists were uninterested in these questions of efficiency and welfare analysis in auctions before the computer scientists started studying them.

cancels out. Hence, the social welfare is maximized when the bidder with the highest valuation wins the auction. Since the social welfare is the valuation of the winning bidder regardless of their bid, this can lead to outcomes that we might find a little strange. For example, even if the player who wins overbids and receives negative utility, the social welfare is still positive. This is because the utility function for each agent is modeled as their surplus. So, this metric is good for measuring how much surplus is created in a system but not necessarily how "happy" everyone is with the outcome.

Moving on, we let $x_i^*(\mathbf{v})$ be an indicator variable for whether or not a player i is the player with the highest valuation (ties broken arbitrarily). Now, we can define what we mean by optimal social welfare.

Definition 2.5 (Optimal Social Welfare)**.** The *optimal social welfare* in a single-item auction is,

$$\mathrm{OPT}(\mathbf{v}) = \sum_{i=1}^{n} v_i \cdot x_i^*(\mathbf{v}).$$

Hence, the optimal social welfare for any auction will is the highest valuation of any player. In a perfect world, the person who values the item the most would always be given it and we could go home knowing that our auctions were allocating resources in an efficient way. However, as we have discussed, the knowledge of how much each player values the item is private knowledge. Unless each player bids their true valuation, which is not a property of strategic play for this auction format, or the auctioneer is a mind reader, there is no way for the auctioneer to know how much each bidder truly values the item based off of the bids they receive, and cannot guarantee they are maximizing the social welfare. Hence, we introduce the concept of the price of anarchy to formalize how much worse than optimal we are.

Definition 2.6 (Price of Anarchy)**.** The *price of anarchy* of an auction, with a valuation distribution $\mathcal{F}$, is the smallest value of the ratio:

$$\frac{\mathbb{E}_{\mathbf{v}}[SW(\mathbf{s}(\mathbf{v}); \mathbf{v})]}{\mathbb{E}_{\mathbf{v}}[\mathrm{OPT}(\mathbf{v})]}, \tag{2.1}$$

ranging over all Bayes-Nash equilibria $\mathbf{s}$ of the auction.

The above definition applies to individual auctions which are dependent on the choice of $\mathcal{F}$ and n. We generally only discuss the price of anarchy for the format of the auction which—in our case—is the first-price auction.

Definition 2.7 (Price of Anarchy First-Price Auction Format)**.** *The price of anarchy for the first-price auction format* is then the worst possible price of anarchy for any choice of the number of players n or valuation distributions $\mathcal{F}$.

Note that the price of anarchy (for either an individual auction or the format of auction) is a number between 0 and 1, and that the closer it is to one, the "better" we can guarantee the system's social welfare will be.[3]

Incredibly, bounds on the price of anarchy for the format of first-price auctions have been found. Again, this allows us to characterize how much worse the social welfare for the system could be at (Bayes-Nash) equilibrium no matter how many players we have or what distributions they are choosing their valuations from. This sort of guarantee is useful, especially for systems where we may not want, or it may not be feasible to have, a central authority pre-calculate the way to optimize social welfare in a system. Rather, we can trust that the system will perform at least so well under strategic interaction.

Theorem 2.8. *(Syrgkanis and Tardos, 2013) The price of anarchy in first-price single-item auctions format is at least* $1 - \frac{1}{e} \approx 0.63$.

[3]Much of the literature for POA (including the paper introducing the idea) defines it as the opposite ratio, Optimal/Worst-EQ where smaller values indicate better systems (Koutsoupias and Papadimitriou, 1999). For some reason the auction literature defines it as Worst-EQ/Optimal, so I will remain consistent with them. This is confusing as *price* of anarchy makes you think it should be a number that gets bigger as it gets worse.

Theorem 2.8 tells us that no matter how many players we have or what distributions we give them, we cannot construct a first-price auction that will achieve less than 0.63% of the optimal social welfare at Bayes-Nash equilibrium.

2.2 Extending Results to No-Regret Agents

The results of Theorem 2.8 hold for first-price auctions, but it is natural to ask questions about how robust these results are. First of all, how do people arrive at a Bayes-Nash equilibria if there doesn't exist a closed form for it? What if there is no Bayes-Nash equilibria at all for this choice of $\mathcal{F}$ and n. Also, do these results hold for mixed Bayes-Nash equilibria (randomizing between bidding strategies) or other larger, more realistic equilibrium concepts? To address these questions, Roughgarden et al. (2017) create what they call an "extension theorem" which allows them to "extend" the price of anarchy results from the small set of Bayes-Nash equilibria to the much larger set of coarse correlated equilibria (Roughgarden et al., 2017, p. 60). This extension will take us to a concept of no-regret learning. We briefly sketch key theorems from this thesis that lead us to an equilibrium concept that applies to learning agents.

2.2.1 Smooth Auctions

In proving the price of anarchy bounds for a variety of different types of games, Tim Roughgarden noticed that many of the proofs followed the same formula:

1. Given a pure strategy Nash equilibrium $\mathbf{s}$ and optimal strategy $\mathbf{s}^*$, for each agent i invoke the equilibrium hypothesis for a hypothetical deviation to $\mathbf{s}_i^*$ to obtain the inequality $u_i(\mathbf{s}) \geq u_i(\mathbf{s}_i^*, \mathbf{s}_{-i})$.

2. Sum the inequalities over the n agents in the game to get $\sum_{i=1}^{n} u_i(\mathbf{s}) \geq \sum_{i=1}^{n} u_i(\mathbf{s}_i^*, \mathbf{s}_{-i})$.

3. Relate the sum of the cost of deviations for each agent to the sum of the utilities for each agent at equilibrium and at optimal getting something of the form, $\sum_{i=1}^{n} u_i(\mathbf{s}_i^*, \mathbf{s}_{-i}) \geq \lambda \sum_{i=1}^{n} u(\mathbf{s}^*) - \mu \sum_{i=1}^{n} u(\mathbf{s})$ for some $\lambda \geq 0$ and $\mu \geq 1$. You do not invoke the equilibrium hypothesis for this step.

4. Put together the inequalities in step 2 and 3 to get $\sum_{i=1}^{n} u_i(\mathbf{s}) \geq \lambda \sum_{i=1}^{n} u(\mathbf{s}^*) - \mu \sum_{i=1}^{n} u(\mathbf{s})$.

5. Solve above equation for the price of anarchy.

In fact, he eventually realized that this generic recipe could be formulated as a precise definition, something he called *smooth games*. Smooth games are ones in which you can prove the third step of Roughgarden's formulas. Since you are not invoking the equilibrium hypothesis for either $\mathbf{s}$ or $\mathbf{s}^*$, this means that this relationship actually applies to any $\mathbf{s}, \mathbf{s}^* \in S$. That is where the name "smoothness" comes from: games that satisfy this property can have their utilities related between deviations of any possible strategy profiles by a simple inequality. Syrgkanis and Tardos (2013) adapted this to create a definition for a smooth auction as follows:

Definition 2.9 (Smooth Auctions)**.** For parameters $\lambda \geq 0$ and $\mu \geq 1$, an auction is (λ, μ)-smooth if for every valuation profile $\mathbf{v} \in \mathcal{F}$ there exists strategy distributions $D_1^*(\mathbf{v}), \ldots, D_n^*(\mathbf{v})$ over $S_1, \ldots, S_n$, such that for every strategy profile $\mathbf{s}$,

$$\sum_i \mathbb{E}_{s_i^* \sim D_i^*(\mathbf{v})}[u_i(s^*, \mathbf{s}_{-i}; v_i)] \geq \lambda \mathrm{OPT}(\mathbf{v}) - \mu \mathcal{R}(\mathbf{s}).$$

The strategy distributions $D_i^*(\mathbf{v})$ in Definition 2.9 are possible a priori distributions on the outcome, or strategy space of the game. While we have not discussed it yet in this thesis, there are more equilibrium concepts than the *pure strategy* Nash equilibrium that we have mainly referred to in this thesis so far: outcomes where you don't randomize your strategy choice. There are also *mixed strategy* Nash equilibria (and Bayes-Nash equilibria) where players randomize their strategy choice from a set of possible strategies. An obvious example of this is in rock-paper-scissors; the best strategy is to pick each strategy (rock, paper, or scissors) with equal probability. With players playing mixed strategies, the resulting distribution on the entire strategy space S is a product distribution, the product of each players individual distributions (the probability that we both play paper is $\frac{1}{3} \cdot \frac{1}{3} = \frac{1}{9}$). Another type of equilibrium that makes use of this distribution on the strategy space is a *correlated equilibrium*, one where given some strategy from prior distribution, one's best strategy is to follow it. The classic example here is a stop light or intersection game. If people see the traffic light telling them to stop or go, their best expected outcome is to follow that advice. Each of these equilibrium concepts makes use of this distribution over the set of outcomes. In the case of pure strategy Nash equilibrium we can just think of it as choosing that particular strategy with probability one.

Now, following from the details of the proof for Theorem 2.8 corresponding to step 3 of Roughgarden's general price of anarchy formula, Roughgarden et al. (2017) show that first-price auctions are $(1 - \frac{1}{e}, 1)$-smooth. They then go on to develop a series of extension theorems that allow us to characterize the price of anarchy for equilibria of auctions when they are not just for pure-strategies. Specifically, we will be looking at how these can be extended a set of outcomes for agents who learn to play the game as they go.

2.2.2 No-Regret Learning

Consider an auction with n players that is repeated for T time steps. At each iteration t, bidder i draws a valuation v_i from $\mathcal{F}_i$ and chooses an action a_i^t which can depend on the history of play. After each iteration, the players observe the actions taken by the other players. A player i is said to use a no-regret learning algorithm if, in hindsight their average regret (difference between average utility of strategy vs algorithm) for any alternative strategy $a_i^{'}$ goes to zero or becomes negative as $T \to \infty$. When all players use this algorithm it results in a vanishing regret sequence.

Definition 2.10 (Vanishing Regret)**.** A sequence of action profiles $\mathbf{a}^1, \mathbf{a}^2, \ldots, \mathbf{a}^T$ is a *vanishing regret sequence* if for every player i and action $a_i^{'} \in \mathcal{A}_i$,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} (u_i(a_i^{'}, \mathbf{a}_{-i}^t; v_i) - u_i(\mathbf{a}^t; v_i)) \leq 0$$

Theorem 2.11 (Roughgarden (2007))**.** *If an auction is (λ, μ)-smooth, then for every valuation profile $\boldsymbol{v}$, every vanishing regret sequence of the auction has expected welfare at least $\frac{\lambda}{\mu} \cdot OPT(\boldsymbol{v})$ as $T \to \infty$.*

The end result is that since single-payer first-price auctions are $(1 - \frac{1}{e}, 1)$ smooth, we know that this bound will hold for auctions with no regret learning agents. Thus, we will try and build a simulation that uses one of the algorithms that fulfills this property and demonstrate that for any arbitrary number of players and distributions from which they pick their valuations from, they converge to an equilibrium social welfare greater than $1 - \frac{1}{e}$.

2.2.3 Coarse Correlated Equilibria

The set of outcomes that no-regret algorithms converge to are not arbitrary. It has been shown that when all of the agents independently run algorithms satisfying the no-regret condition they converge to a set of outcomes called *coarse correlated equilibria*. Informally, one can think of this as the equilibria where a player's best option is agreeing to take advice from some trusted outside source before they actually hear what that advice is. For example, if I were taking a math exam that I was completely unprepared for and I was given the option of putting down the same answer as the best student in the class, my expected grade would be higher for putting down their solution. Following the definition from Roughgarden (2016),

Definition 2.12 (Coarse Correlated Equilibria)**.** A distribution σ on the set $S_1 \times \ldots \times S_n$ of outcomes of a game is a *coarse correlated equilibrium* (CCE) if for every $i \in 1, \ldots n$ and every unilateral deviation $s_i^{'} \in S_i$,

$$\mathbb{E}_{\mathbf{s}\sim\sigma}[u_i(\mathbf{s})] \geq \mathbb{E}_{\mathbf{s}\sim\sigma}[u_i(s_i^{'}, \mathbf{s}_{-i})].$$

Interpreting our math test game example in light of Definition 2.12, we interpret the star student as sampling $\mathbf{s}$ from σ when she selects her answer and gives it to me. The distribution on possible outcome, σ, is generally interpreted as being publicly known (i.e I know that she is probably submitting the right answer). Hence, the best I can do here is just follow her advice. The difference between this definition and the (unprovided) definition for a correlated equilibria is that expectations would be conditioned on s_i, that is, player i gets to "see" the advised strategy first, perform a Bayesian-update on their expectation, and then decide what to do.

It is perhaps easy to see how this could generalize to no-regret outcomes, and Roughgarden (2016) shows exactly that. Specifically, he proves that if each player implements a no-regret algorithm, than their expected utility after T steps with regret ϵ will be,

$$\mathbb{E}_{\mathbf{s}\ \sigma}[u_i(\mathbf{s})] \geq \mathbb{E}_{\mathbf{s}\ \sigma}[u_i(s_i^{'}, \mathbf{s}_{-i})] + \epsilon.$$

Because these players are implementing vanishing regret sequences, their regret grows arbitrarily smaller as the number of iterations approaches infinity, i.e., they grow arbitrarily closer and closer to a coarse correlated equilibria.

The convergence of no-regret bidders in (λ, μ)-smooth auctions to coarse correlated equilibria above were proved for when the valuation profiles are fixed. That is, over the course of repeated play with $\mathbf{v}$ sampled from $\mathcal{F}$ once the beginning of the game. Hartline et al. (2015) expanded these results to hold when each player samples their valuations again at random for each round of play.[4] Since having bidders learn to bid when draw their valuations at the beginning of each round is more interesting to simulate, that is how I will build my simulation.

[4]This paper is phenomenal and a natural next read if you found this chapter intriguing.

Chapter 3

Simulating the Price of Anarchy

This chapter constructs a novel simulation of first-price single-payer auctions to demonstrate that the ratio of actual social welfare to optimal social welfare is within the bounds given by the price of anarchy for this format. First, the construction of this simulation is discussed. Next, I demonstrate the results of running the simulation on bidders using known Bayes-Nash equilibrium strategies and an arbitrary bidding strategy. Finally, I demonstrate that these bounds also hold for no-regret learning agents in a finite action variant of the first-price auction.

3.1 A Simulated Auction Environment

To simulate sequential first-price auctions, I construct a program in Python that allows us to create an arbitrary number of bidders, each with valuation distributions of our choice who simultaneously bid on an item being auctioned for as many sequential auctions, or rounds, as I choose. Each round represents an auction for a new, but similar item where the valuation distributions for each bidder remains the same. At the beginning of each round, every bidder draws a new valuation from their distribution. The bidders then each simultaneously submit a bid to the auction and the winner is determined by the highest bid (ties are broken with equal probability among those with the same bid). After the winner is selected, they are given utility $u(v, b) = v - b$, the difference between their valuation and bid for that round. At each round the total social welfare is v_i, the valuation of the winning bidder as per Definition 2.4. The optimal social welfare each round then is the highest valuation of any bidder. These values are summed across auctions to get the total social welfare for this sequence of auctions and to see what the optimal social welfare would have been. This is laid out in the pseudo-code version of the sequential auction below where I use the super script t to denote which round each variable is from.[1]

Algorithm 1: Sequential First-Price Single-Item Auction

Initialize SW, OPT, and POA to zero

for $t = 1, \ldots, T$ **do**

- Each bidder draws their valuation v_i^t from their distribution F_i;
- Each bidder uses their strategy s_i^t to submit a bid;
- The highest bidder is assigned the object and they pay their bid, if tie, a winner is choosen randomly among them;
- Each player has their utilites updated according to if they won the object;
- **if** *player i wins the auction* **then**
  - $SW \leftarrow SW + v_i^t$
- **if** *player j has the highest valuation* **then**
  - $OPT \leftarrow OPT + v_j^t$
- $POA \leftarrow \frac{SW}{OPT}$

For the simulation results I will refer to the ratio of the produced social welfare to the optimal social welfare as the POA (as per the last line of Algorithm 1). It is worth remembering that the price of anarchy is actually very specifically the worst case equilibria welfare over the optimal welfare (see Definition 2.6), not the social

[1]The "$\leftarrow$" symbol used in the algorithm means assignment of value. For example $x \leftarrow 1$ is the variable x is assigned a value of 1. This reduces the ambiguity of using the "=" symbol which could be also be a statement or proposition in pseudo-code.

welfare for any particular outcome compared to the optimal. However, it is useful to have a quick way to refer to the produced ratio of SW/OPT in the simulation, so I will use this notation for the tables and figures for lack of an obvious substitute.

3.1.1 Simulating Bayes-Nash Equilibria

Using the simulation outlined above, I first demonstrate that bidders using known Bayes-Nash equilibrium strategies (see Section 2.1.4) have an average price of anarchy greater than 0.63. First, I simulate the case of two bidders each drawing their valuation from the uniform distribution $[0, 1]$. The agents are using hard-coded strategies $s(v_i^t) = \frac{v_i^t}{2}$ for each bidder as was shown to be the unique Bayes-Nash Equilibrium in section 2.1.4. The results are shown in Table 3.1 below, where the cumulative price of anarchy is given up to the specified round specified.[2]

Round	POA
1	1.0000
10	1.0000
100	1.0000
1,000	1.0000
10,000	1.0000
100,000	1.0000

Table 3.1: POA in two player symmetric auction

This example is obviously silly to simulate, because this equilibrium is fully efficient. As each player bids half of their valuation, the winner is always the player with the highest valuation. Hence the actual social welfare is always the same as the optimal social welfare:

$$\frac{SW(\mathbf{s}(\mathbf{v}); \mathbf{v})}{\mathrm{OPT}(\mathbf{v})} = 1.$$

This does however give us a simple check of correctness for my simulation. Let us now move to simulate the more interesting example of two bidders with asymmetric

[2]The POA is calculated as per Algorithm 1.

distributions. I let bidder one choose their valuation from the uniform distribution over the interval $[0, 1]$ and bidder two choose their valuation from the uniform distribution on the interval $[0, 2]$. As stated in Section 2.1.4, the unique Bayes-Nash equilibrium for this auction is:

$$s_1(v_1) = \frac{4}{3v_1}\left(1 - \sqrt{1 - \frac{3v_1^2}{4}}\right)$$
$$s_2(v_2) = \frac{4}{3v_2}\left(\sqrt{1 + \frac{3v_2^2}{4}} - 1\right)$$

Again, this auction is not fully efficient, as the bidder who is drawing their valuation from the smaller distribution has to shade their bid less (bid higher relative to their valuation) which can sometimes result in the bidder who values the item less winning it. I simulate this sequentially 100,000 times and get the following results, as shown in Table 3.2:

Round	POA
1	1.0000
10	1.0000
1,000	0.9919
10,000	0.9924
100,000	0.9935

Table 3.2: POA in two player asymmetric auction

While this auction is not fully efficient, it is highly efficient. The price of anarchy in this auction never drops below 0.99. The main reason this happens is that while it is possible for the bidder drawing from the smaller distribution to win even if they have the smaller valuation, this only occurs when their two valuations are relatively close. This means that the social welfare lost in this case is not much, even if it is not fully efficient. In both of these auctions we see that they are well above the 0.63 lower bound guaranteed for all first-price single-item auction formats. The fact that they perform so well is not surprising given that this is such a simple system. We would expect people who are rationally maximizing their own utility to win most of the time if they have the highest bid. If anything, this simulation highlights the need to find an exact lower bound on the price of anarchy for Bayes-Nash equilibria as it would be more surprising if we could find an instance of this auction that produces highly inefficient outcomes.

3.2 Simulating No-Regret Bidders

With the results from the Bayes-Nash equilibrium demonstrated, I move on to simulating the bounds of no-regret learning agents in first-price auctions. Again, the equilibria they converge to should be arbitrarily close to a coarse correlated equilibrium since each agent is using a no-regret learning algorithm. In order to use these algorithms, I must make a concession to the game I am simulating: I must now simulate an auction where the bidders are only allowed a finite number of strategies that they must learn to choose between.

3.2.1 Multiplicative Weights Algorithm

The no-regret learning algorithm I will use to train my bidders is called the *multiplicative weights* algorithm. It has been shown to satisfy the no-regret property in papers such as Littlestone and Warmuth (1994) or Freund and Schapire (1999), but my implementation of it comes from Roughgarden (2016). If each of our bidders use the no-regret learning algorithm, our auction should converge to a coarse correlated equilibrium with efficiency greater than the 0.63 price of anarchy bound from Section 2.2.2.

Before giving the algorithm, a few words are probably necessary to understand where it comes from and what it is trying to do. First, this algorithm is what is called an *online* algorithm. That is an algorithm that takes its inputs sequentially as it goes rather than getting all of its inputs up front, i.e. it learns how to play as it is playing. Next, this algorithm and many other algorithms for players learning in games are based around the player only having a fixed number of actions they can choose from. For each round in the game, the player gets outside advice from "experts" who recommend to the player what to do at each round and the player picks between the experts' advice to decide what to do (Roughgarden, 2016). For us, these experts are our strategies that will map a valuation to a bid. At each time step t, the player picks the action to play and then after that, some adversary picks the utilities to assign for each action that could have been taken. This is a stronger condition than we will need as the adversary in a first-price auction is the cumulative action of the other players, where the highest bid determines which strategies (if any) the player could have taken and won. That is, the adversary from the perspective of any individual agent playing this game is the collective actions of the other players. Based off of the actions these other players choose (how they bid), this particular agent will either win the item and receive utility, or not. In the general case beyond auctions, this algorithm has been shown to be no-regret in the face of an adversary directly picking the utilities the learning agent receives for any given action (see Lugosi (2006) for a broad overview of prediction with expert advice and learning in games).

Algorithm 2: Multiplicative Weights (MW) Algorithm

Initialize $w^1(a) = 1$ for every $a \in A$

for $t = 1, 2, \ldots, T$ **do**

  Use distribution $p^t = \frac{w^t}{\sum_{a \in A} w^t(a)}$ over actions to pick $a \in A$ and output a.

  Given the utility vector u^t, for every action $a \in A$ use the formula

   $w^{t+1}(a) = w^t(a) \cdot (1 + \eta u^t(a))$ to update its weight.

The logic of the multiplicative weights algorithm (Algorithm 2) is simple. At each time step we see how well each of the possible actions we could have taken performed; then proportionally increase or decrease the probability of picking that action next time accordingly. The adjustment of the probability is done proportionally to the utility that would have been received if that action had been played in that round multiplied by some *learning rate*, η, which is a tuning parameter chosen before starting the procedure. As demonstrated in Roughgarden (2016) and Blum and Mansour (2007), the MW algorithm is no-regret if $\eta = \sqrt{(\ln n)/T}$ where n is the number of actions that this agent can choose from, and T is the number of rounds that will be played (yes, this assumes that the player knows T up front).

This algorithm fulfills our purpose of simulating agents learning in auctions, but in some sense it is unsatisfying that the learning algorithm requires a finite set of actions that the bidder does not even get to choose. It would be more interesting if we were able to give our agents some reasonable algorithm that allowed them to formulate their own strategies or mappings between their valuation and bid rather than choosing from a pre-made set. Unfortunately, these other learning algorithms would introduce a whole host of other problems from the reasonableness of expecting agents to implement such algorithms to the ability to prove that such algorithms converge to an equilibrium. Part of the beauty of using multiplicative weights is that it is simple *and* has nice, predictable mathematical properties. Using advanced machine learning techniques, these properties might no longer hold and it would be harder to argue it is a useful agent-based model.[3]

[3]This thesis grew out of an interest in doing just that, throwing "smarter" algorithms such as neural networks and machine learning into existing multi-agent simulations. It becomes hard to tell what the point of such simulations are when the dynamics might just be properties of the interaction of the specific algorithms used and not of the system itself.

3.2.2 No-Regret Learners

The first simulations I run are symmetric auctions with agents learning using the multiplicative weights algorithm (Algorithm 2). I create 101 strategies that bidders can choose from to shade their bid, from bidding zero percent of their valuation with one percent increases up to bidding 100 percent of their valuation. That is, $S = \{0, 0.01{\cdot}v_i, 0.02{\cdot}v_i, \ldots, 0.99{\cdot}v_i, v_i\}$. First, I run the simulation with two symmetric bidders each choosing their valuation from the uniform distribution over $[0, 1]$. The results are shown below in Table 3.3.

Round	POA
1	1.0000
10	0.9517
1,000	0.9129
10,000	0.9578
100,000	0.9947

Table 3.3: POA in two player asymmetric auction with no-regret learning

Here we can see that after 100,000 rounds the price of anarchy moves to 0.99 and near-perfect efficiency as the agents learn how to play the game. It's important to point out here that the above table is not an average, but simply one run of the simulation with 100,000 sequential auctions. Importantly, each time I run the simulation it is possible for the bidders to learn to converge to a different equilibrium. Hence, the POA values for each simulation can be different depending on how the bidders in that simulation happened to learn. While we are mainly concerned with the lowest possible POA that the system converges to as a demonstration of the proven bounds, it is also interesting to see what the average and best case simulations resulted in, as these will give us more confidence about the robust efficiency of the system. I now repeat this using 2 through 100 agents, each symmetric and drawing from a uniform $[0, 1]$. I run 100 simulations with each number of agents, each for 100,000 rounds. We also run the same set of simulations for a set of bidders I call

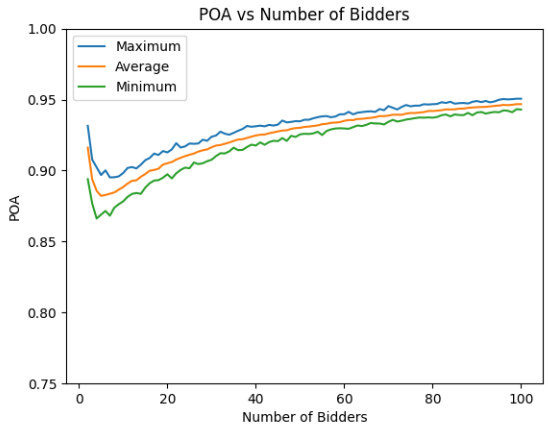

Figure 3.1: POA for symmetric, MI-bidders

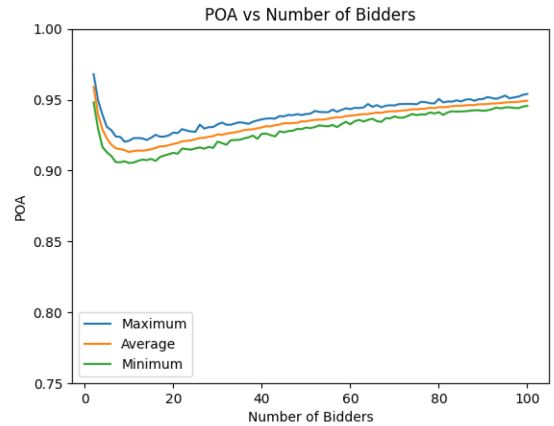

Figure 3.2: POA for symmetric, no-regret bidders

minimally intelligent (MI-bidders), where bidder i draws their bid uniformly from the interval $[0, v_i^t]$ for each round t. They are called minimally intelligent because they are choosing their bids in range based off of their valuation and in this case not overbidding.[4] This gives us a baseline to compare the results of the learning against to see how much of the efficiency is a product of being at an equilibrium and how much is based off of the system. Taking the minimum, maximum, and averages of the efficiency outcome produced in these simulations gives us the results shown in Figures 3.1 and 3.2.

In both the learning and non-learning case, the efficiency non-monotonically increases with the number of bidders. This pattern of decreasing POA from 2 to 10 bidders and then increasing again almost like a check mark is something we will see in every simulation. It seems to be a pattern caused by the probability that someone with a high valuation wins by chance has a minimum at about 10 players. I have not yet figured out definitively why that is.

Moving on, in both cases, we see that the system is near optimal efficiency. The efficiency of the outcome for the learning agents is higher than for the minimally intelligent ones, but not significantly so. Because the minimally intelligent bidders

[4]In computer science literature there are often zero-intellince agents. I figured it would be rude to call my agents that as they are not overbidding.

are only bidding in the range of their current valuation, it is still most likely for the bidder with the highest valuation to win. Similarly, as the number of bidders increases (past 10) it becomes more likely that someone with a relatively high valuation will randomly choose a high bid resulting in a fairly efficient outcome (Someone with a high valuation winning the item).

Since it is theoretically possible in some games that social welfare will be worse if everyone learns to play strategically (such as in the routing game from Section 1.1.1), it is good to see that the social welfare actually increases in this auction as agents learn how to bid. This should give economists some confidence that allowing agents to compete strategically in this auction format leads to near efficient outcomes. Also, we see that the efficiency results are well above the lower bound on the price of anarchy of 0.66 showing that whatever auction set up that causes this might be some strange case and that maybe we should expect the efficiency to be much higher than this bound.

Next, I simulate the case of having two bidders with asymmetric valuation distributions where one draws uniformly from $[0, 1]$ and the other draws from $[0, 2]$. We can see the result of simulating this 100,000 times in Table 3.4 below.

Round	POA
1	1.0000
10	0.8901
1,000	0.9314
10,000	0.9770
100,000	0.9961

Table 3.4: POA in two player asymmetric auction with no-regret learning

Here we see the agents converge to a very high efficiency similar to the behavior we saw when using the Bayes-Nash equilibrium for this setting and better than in the two bidder minimum-intelligence case. Now again I conduct this simulation with 2 to 100 asymmetric bidders. For each number of bidders I simulate this 100 times

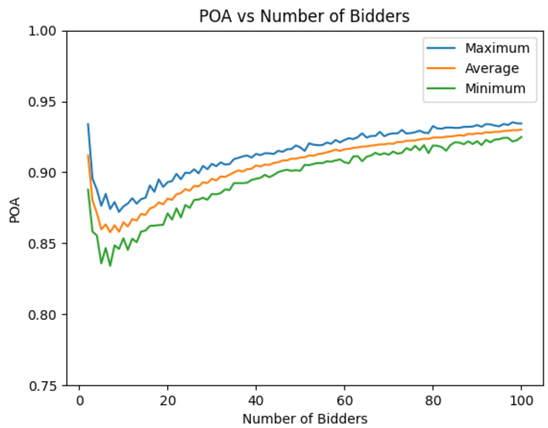

Figure 3.3: POA for asymmetric, MI-bidders

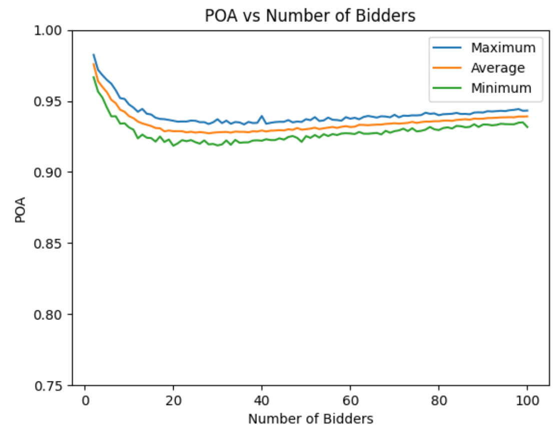

Figure 3.4: POA for asymmetric, no-regret bidders

and when there are an odd number of bidders there is an extra $[0, 1]$ bidder. Each sequential auction consists of 100,000 rounds. The results are shown in Figure 3.4. As before, I also conduct the same experiment with minimally intelligent bidders who choose their bids uniformly from $[0, v_i^t]$ each round.

Figures 3.3 and 3.4 are similar to the results for the symmetric bidding counterparts, but with slightly lower efficiency as the number of bidders increases. Again, there is very high efficiency for two or three bidders, and then a dramatic decrease. We then see the numbers slowly climb back towards being fully efficient in the best, worst, and average cases.

Comparing the results of the no-regret symmetric and asymmetric simulations, we see that they have similar efficiency resulst for 20 bidders and less. However, after 20 bidders we see the symmetric bidders converge to a higher efficiency as n increases. This makes sense as the symmetric case has a fully efficient Bayes-Nash equilibrium that it can converge to and the asymmetric case does not (or at least it doesn't for $n = 2$). In general it seems like asymmetry in the valuation distributions of agents just causes there to be more opportunity for inefficient allocations.

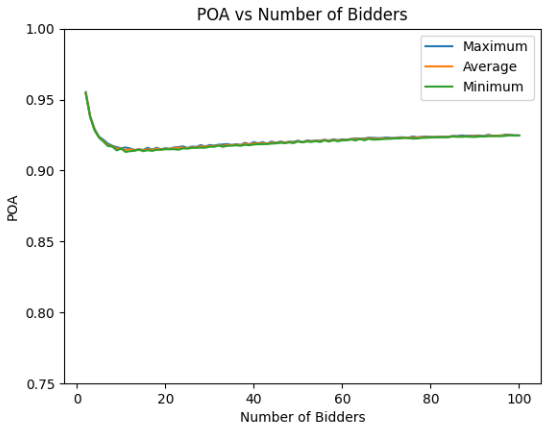

Figure 3.5: POA for normal, MI-bidders

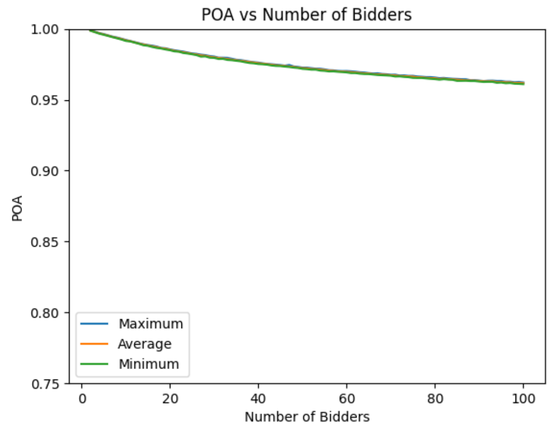

Figure 3.6: POA for normal, no-regret bidders

3.2.3 Variations With Less Rounds

This section continues the simulation no-regret agents in first-price auctions, but with fewer rounds so as to have time to try a wider variety of auction parameters. Each of these variations was conducted 10 times for $10,000$ rounds, with from 2 to 100 bidders in each auction. These simulations represent a desire to test the efficiency bounds given by the price of anarchy and are situations I predicted to lead to more inefficient outcomes than their symmetric counterpart.

The first variation conducted used bidders who draw their valuations from a normal distribution with a mean of 10 and a standard deviation of 1, $\mathcal{N}(\mu = 10, \sigma^2 = 1)$. This distribution might be considered more representative of the distribution of valuations for many items. Most people value it around the same amount, and extreme valuations become more and more unlikely. The mean of 10 with standard deviation 1 is chosen to make it nearly impossible for someone to draw a negative distribution.

As can be seen in Figure 3.6, when drawing from a normal distribution, the difference between the minimum, maximum and average efficiency collapses and the simulation produces approximately the same results every time. Most interestingly, the POA decreases monotonically for the learning agents as the number of bidders increases. Something about this setup teaches people to bid more and more ineffi-

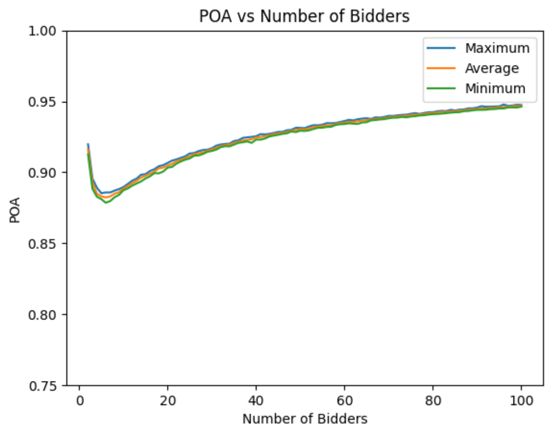

Figure 3.7: POA for overbidding, MI-bidders

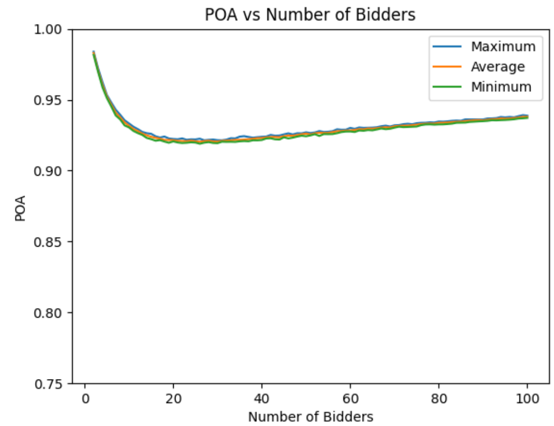

Figure 3.8: POA for overbidding, no-regret bidders

ciently as they compete against each other.[5] Here again we also see that the learning agents lead to outcomes with higher efficiency than in the minimally intelligent case.

Next, we move to the case of allowing symmetric agents to overbid. For the no-regret agents, this meant choosing from strategies between zero and up to two times their valuation, $S = \{0 \cdot v_i^t, 0.01 \cdot v_i^t, \ldots, v_i^t, \ldots 1.99 \cdot v_i^t\}$. For the minimally intelligent bidders this meant choosing their bids uniformly from $[0, 2v_i^t]$. In both cases, all bidders draw their valuations uniformly from $[0, 1]$ at the beginning of each round.

The results for symmetric over-bidders shown in Figures 3.7 and 3.8 are nearly identical to the non overbidding symmetric bidder case shown in Section 3.2.2. This is because overbidding does not hurt the social welfare as defined in Definition 2.4. The bid of the winner has nothing to do with the social welfare as I have defined it: the valuation of the winning bidder. This highlights a fault of measuring the utility of each individual as being equal to their surplus in that auction. In this system, no matter how much the winner overbids, we say it is efficient and maximized social welfare if the winner had the highest valuation. In one sense, we should be happy that the bidder who values the item the most was awarded it. However, we should

[5] When I first saw this graph I assumed I had coded the simulation wrong. Text tools comparison showed the code was correct and the only difference between this and the symmetric uniform no-regret simulation was the distribution that agents draw their valuations from.

be concerned that we are saying even with this person having negative utility at the price they paid this is a good outcome. This metric seems good at measuring surplus in a system, but it would be good to do this analysis with different utility functions.

Returning to the simulation results, because the minimally intelligent bidders are bidding in a range defined by their valuation, as in the symmetric uniform simulation, the people who value it more are more likely to win. In fact, the minimally intelligent bidders in Figure 3.7 produce POA efficiencies slightly higher than those who were not allowed to overbid in Figure 3.1. This is because bidding uniformly in the range $[0, 2v_i^t]$ makes it even more likely for people with higher valuations to draw the winning bid. In the case of the learning agents, they also behave similarly to their counterparts who could not overbid, but in this case it is no doubt because they quickly learned to overbid; as such Figure 3.8 and Figure 3.2 are almost identical. So, throwing in bad options doesn't seem to effect the efficiency for learning agents.

The final simulation I conduct is one with the hopes of producing the worst efficiency in the minimally intelligent case to demonstrate the effectiveness of the learning. Up to this point, we have seen minimally intelligent agents produce near efficient outcomes simply as a product of their bidding range being a function of their valuation (which set the maximum on the interval they were drawing from). Agents who have higher valuations are more likely to win simply because they are drawing from a larger interval. One way to create highly inefficient auctions would be to have bidders who draw their bids uniformly from some large distribution that has nothing to do with players' valuations. Say that bidders draw their valuation uniformly from $[0, 1]$, and their bid uniformly from $[0, 100]$. The problem with this is that there no point of comparison in the no-regret agent counterpart to this of strategies. The strategies no-regret agents learn should be functions of their valuation for it to be meaningful, not just random numbers on the interval of $[0, 100]$.[6] To avoid this issue, we instead

[6]This simulation actually should still work. Theory tells us the price of anarchy bound should hold for this auction format regardless of the strategy sets that we give the players. The question

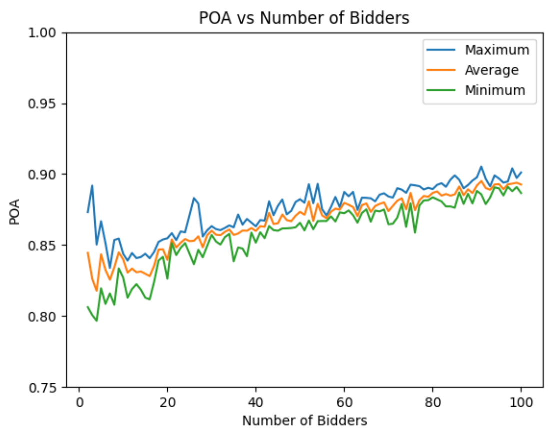

Figure 3.9: POA for random over-bidding factor, MI-bidders

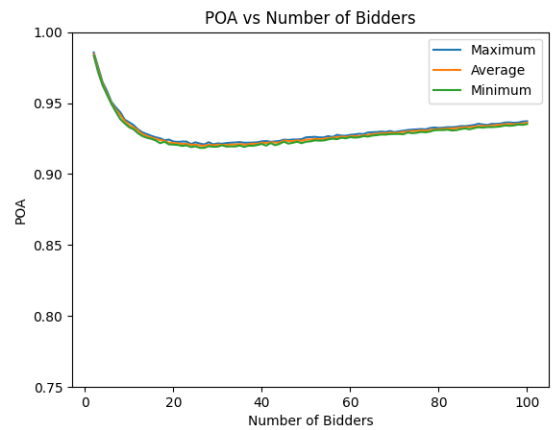

Figure 3.10: POA for random overbidding factor, no-regret bidders

do what I call a "random overbidding" simulation.

For the random overbidding simulation, at the beginning of each simulation the agent i draws a number α_i uniformly from the interval $[0, 100]$. The minimally intelligent agents then arbitrarily choose their bid from the interval $[0, \alpha v_i^t]$, and the no-regret agents learn how to bid from the strategy set $S = \{0, 0.01v_i^t, \ldots, \alpha_i \cdot 0.99v_i^t\}$, again going up by increments of 0.01 starting from 0 and now going all of the way up to $\alpha_i v_i^t$. The results from running these simulations are shown in Figures 3.9 and 3.10.

The minimally intelligent auction in this case yields wildly inefficient POA. The jaggedness in Figure 3.9 is caused by the possible variations in the difference between bidding factors between rounds. Some of these combinations produce outcomes that are more inefficient than others. The worst case efficiency in the minimally intelligent case is the worst case efficiency produced. This is because with such variation in the bidding factors, bidders with high α are likely to win the bid regardless of if they have the highest valuation or not. Looking at the no-regret learners in Figure 3.10, we see results again similar to that of the original symmetric bidders in Figure 3.2.

at that point is what is being simulated when we assume that agents cannot use their valuation as part of their bidding strategy?

Bidders learn to not overbid and the results converge to efficiency outcomes similar to if they did not have the option.

Conclusion

This simulation demonstrated efficiency greater than the theoretical price of anarchy bounds for first-auctions with no-regret bidders. In fact, by all indications with both learning and non-learning agents, first-price single-item auctions are incredibly efficient![7] These results beg for a proof for a tighter bound on the price of anarchy for the first-price auction format. The simulated results produced POAs much higher than the lower bound set by theory. This avenue is especially compelling by the high efficiency of the system even without learning seeming to indicate that near-optimal outcomes are just a property of the system. The Price of Anarchy of near 0.66 could still be correct, perhaps there is some worst case set up out there that produces efficiencies at equilibria lower than we even saw with random bidders.

Regardless of how low the price of anarchy actually is for this format, combining the results from the learning and non-learning agents we should feel incredibly confident in the efficiency of first-price auctions implemented in the real world. It seems they they behave well as a way to allocate resources to those who value them the most (and have the means to bid whatever they wish). However, given that the second-price, sealed-bid auction is fully efficient, if your goal is to maximize social welfare you would be better off just using that format. That being said, this style of auction is implemented from time to time by auctioneers who believe that this might be the best way to maximize their own revenue. When this happens, the results of

[7] Albeit, these simulation results are only for a limited number of valuation distributions.

this simulation and price of anarchy analysis indicate that the winner will probably be one of the people who values the item the most.

Moving forward using simulations to understand the price of anarchy in auctions, there are two obvious ways to extend this research. One is to build a simulation for simultaneous first-price auctions or all-pay auctions (where everyone must pay regardless of winning) where the price of anarchy bounds for no-regret agents have also been proven using smooth-auctions (Roughgarden et al., 2017). For simultaneous auctions, or any sort of auction where bidders must make many decisions at once (in this case, bids), simulating no-regret learning becomes much harder as the number of actions increases. The simulations presented here were already slow: it is hard to imagine how long it would take to run a simulation of such a procedure. Another obvious extension is to try simulating this again using different algorithms. Specifically, we could try different no-regret algorithms to see if they converge to similar efficiency outcomes, or perhaps modern machine learning techniques (while opening up the bids to be any real number). Again, if one uses techniques that tend to operate as a black box it might be hard to know how sensitive the results are to particular parameters or what results of the simulation might possibly be in every scenario. No-regret learning is only useful as a stand in for humans because it is easy to implement, and well behaved (the outcome satisfies the no-regret condition). Because of these properties, we should also expect human agents learning in auctions to make choices that lead to mostly efficient outcomes.

Economists interested in agent-based simulations should also strongly consider using no-regret learning techniques for their simulations if there is a reasonable fixed action variant of the environment they are trying to simulate. Because these learning algorithms converge to coarse correlated equilibria it allows analytical math to help characterize the outcome of the situation more strongly then a simulation might be able to. For example, with the price of anarchy bounds one can say "these auctions did quite well in simulation, and we know for a fact it can never get worse than x even for the most exotic setup that we did not test in simulation.” Interestingly, Arthur (1991), suggests using the multiplicative weights algorithm as a learning algorithm for agents. If there are agent based models out there that use this algorithm, they will converge to a coarse correlated equilibria: a fact that might prove helpful for further analyzing the system.

Finally, the results of this simulation could mostly be summed up as “the price of anarchy bounds are correct for this format.” The most interesting places to explore the ideas of the price of anarchy lie in the math behind it, not in a simulation of such a simple system. There are a plethora of open questions in the intersection of auction theory and algorithmic game theory which Roughgarden et al. (2017) lay out at the end of their work. The simulation produced here can help us understand the theory and grapple with what it is and is not saying, but the strongest and most compelling results lie down the road of more theory.

Appendix A

Simulation Code

The code for this simulation is available at the public GitHub repository:

https://github.com/15rsirvin/Computational-Economics

in the sequential auction folder. There are different Python files for each of the simulations where the relevant parts are changed to make them minimally intelligent bidders or asymmetric. The simulation is coded in Python and mostly consists of loops to sequentially run each auction, update each bidder, and then allow each bidder to learn. I ran each simulation in parallel on a different processor to try and speed up the process, but it is still quite slow for large numbers of simulations, rounds, or bidders. That is because this process is essentially $O(T \cdot n \cdot m)$, where T is the number of rounds, n the number of bidders, and m is the number of simulations at that number of bidders.

I ran my code on a google cloud platform “compute engine” server to speed up the simulation process with a faster computer and to allow the simulation to run continuously. This enabled us to choose high values of T, m, and n, despite how long the code takes to execute with such values.

If someone were to try and replicate parts of this thesis, I would recommend re-writing the code in a faster language like C or C++. Because this code scales linearly with any individual input, optimizing the speed at which the loop executes should have a massive effect on the total time it takes to compute. It might also be possible to algorithmically optimize this simulation so that it can scale sub-linearly, but I have no idea how one would do that. Ultimately I decided not to spend to much time optimizing the code as letting it run on a remote server was not a huge issue.

Bibliography

Arthur, W. Brian. 1991. Designing Economic Agents that act like Human Agents: A Behavioral Approach to Bounded Rationality. *The American Economic Review* 81: 353–359.

Arthur, W Brian. 1992. On Learning and Adaptation in the Economy. *Queen's University, Department of Economics, Working Papers* (854).

Backhouse, Roger, and Beatrice Cherrier. 2016. 'It's Computerization, Stupid!' The Spread of Computers and the Changing Roles of Theoretical and Applied Economics. *SSRN Electronic Journal* .

Blum, Avrim, and Yishay Mansour. 2007. Learning, Regret Minimization, and Equilibria. In *Algorithmic Game Theory*, edited by Noam Nisan, Tim Roughgarden, and Vijay Vazirani, chap. 4, 79–100.

Braess, D. 1968. Uber ein paradoxen der verkehrsplanung. *Unternehmensforschung* 12: 258–268.

Chawla, Shuchi, and Jason D. Hartline. 2013. Auctions with unique equilibria. In *Proceedings of the ACM Conference on Electronic Commerce*, vol. 1, 181–195.

Freund, Yoav, and Robert E. Schapire. 1999. Adaptive Game Playing Using Multiplicative Weights. *Games and Economic Behavior* 29(1): 79–103.

Harsanyi, John C. 1967. Games with Incomplete Information Played by "Bayesian" Players. *Management Science* 14(3): 159–182.

Hart, Sergiu, and Andreu Mas-Colell. 2000. A simple adaptive procedure leading to correlated equilibrium. *Econometrica* 68(5): 1127–1150.

Hartline, Jason, Vasilis Syrgkanis, and Éva Tardos. 2015. No-regret learning in Bayesian games. *arXiv* .
URL https://arxiv.org/abs/1507.00418

Holland, John, and John Miller. 1991. Artificial Adaptive Agents in Economic Theory. *American Economic Review* .

Jamal, Karim, Michael Maier, and Shyam Sunder. 2017. Simple Agents, Intelligent Markets. *Computational Economics* 49(4): 653–675.

Joshi, Shareen. 1998. *Is Technical Trading A Menace? Resuslts From An Agent-Based Artificial Model*, Reed College.

Koutsoupias, Ellias, and Christos H Papadimitriou. 1999. Worst-case equilibria. In *16th Annual Symposium on Theoritical Aspects of Computer Science*, 404–413.

Krishna, Vijay. 2002. *Auction Theory*. Academic Press.

LeBaron, Blake. 2002. Building the Santa Fe artificial stock market. *Physica A* (June).

Levin, Jonathan. 2002. Games of Incomplete Information.
URL web.stanford.edu/~jdlevin/Econ203/Bayesian.pdf

Littlestone, N., and M. K. Warmuth. 1994. The Weighted Majority Algorithm. *Information and Computation* 108(2): 212–261.

Lugosi, Gábor. 2006. *Prediction, learning, and games*. Cambridge University Press.

Mochon, Asuncion, and Yago Saez. 2015. *Understanding Auctions*. Springer International Publishing Switzerland.

Nisan, Noam. 2007. Introduction to Mechanism Design (for Computer Scientists). In *Algorithmic Game Theory*, edited by Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, chap. 9, 209–241.

Papadimitriou, Christos. 2015. On Algorithmic Game Theory I.
URL https://www.youtube.com/watch?v=ggi2PZbO0oQ{\&}t=814s

Papadimitriou, Christos H. 2001. Algorithms , Games , and the Internet. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, 749–753.

Rejeb, Lilia, and Zahia Guessoum. 2005. Firms Adaptation in Dynamic Economic Systems. In *Artificial Economics*, 53–64.

Roughgarden, Tim. Beyond Worst-Case Analysis I.
URL https://www.youtube.com/watch?v=thHt1lhLqJA

Roughgarden, Tim. 2007. Routing games. In *Algorithmic Game Theory*, edited by Noam Nisan, Tim Roughgarden, and Vijay Vazirani, chap. 18, 461–486.

Roughgarden, Tim. 2016. *Twenty lectures on algorithmic game theory*. Cambridge University Press.

Roughgarden, Tim, Vasilis Syrgkanis, and Éva Tardos. 2017. The price of anarchy in auctions. *Journal of Artificial Intelligence Research* 59: 59–101.

Russel, Stuart J, and Peter Norvig. 2010. *Artificial Intelligence: A Modern Approach*. Upper Saddle River, New Jearsy: Prentice Hall, 3 edn.

Sargent, Thomas J, and John Stachurski. 2019. Lectures in Quantitative Economics
.
URL https://python.quantecon.org/

Schelling, Thomas. 1969. Models of Segregation. *American Economic Review* 59: 488–93.

Shoham, Yoav. 2008. Computer science and game theory. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*.

Shoham, Yoav, and Kevin Leyton-Brown. 2008. Multiagent systems: Algorithmic, Game-Theoretic, and logical foundations. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations* .

Syrgkanis, Vasilis, and Éva Tardos. 2013. Composable and efficient mechanisms. In *Proceedings of the Annual ACM Symposium on Theory of Computing*, 211–220.

Tesfatsion, Leigh. 2002. Agent-based computational economics: growing economies from the bottom up. *Artificial life* .

Vickrey, William. 1961. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance* 16(1): 8–37.