Simulating Auctions

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

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

Robert S. Irvin

May 2020

Approved for the Committee for
(Economics-Mathematics)

Jeffery Parker

David Perkinson

Acknowledgments

The cat, Pinoe.

Preface

“For if we could suppose a great multitude of men to consent in the observation of justice and other laws of nature without a common power to keep them in awe, we might as well suppose all mankind to do the same; and then there neither would be, nor need to be, any civil government or commonwealth at all, because there would be peace without subjection.” -Thomas Hobbes (Leviathan, chapter XVII)

List of Abbreviations

AI	Artificial Intelligence
MAS	Multi-Agent System
ML	Machine Learning
POA	Price of Anarchy

Contents

List of Tables

List of Figures

Abstract

The preface pretty much says it all.

Dedication

You can have a dedication here if you wish.

Introduction

This thesis lays out the literature on the price of anarchy bounds for single item, first price auctions and then constructs a simulation to demonstrate these bounds for no-regret agents. Chapter one is a broad overview of related fields to give context for this discussion covering the price of anarchy, auctions, and simulations. Chapter two goes through the mathematical theory in a more rigorous way laying out what sort of behavior we expect and why for no-regret agents in these auctions as well as how the price of anarchy for first price, single payer auctions were derived. Chapter three constructs a novel simulation of first-price auctions which demonstrates the high efficiency of this format both at equilibria and in general.

Chapter 1

Auctions and Computational Economics

Within the last three decades, computational economics has been on the rise. 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 analysis. Two, that the mathematical methods developed in the field of theoretical computer science can be used to gain better understanding of existing models in terms of their algorithmic and computational complexity properties. We aim to take elements from both of these frameworks to better understand the social welfare of 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 that have roots going back to at least the ancient Greeks and still exist today in places such as art auctions and Ebay (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 scientists 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"(**?**). It is within this framework that he says computer scientists first began to study auctions as they existed on the internet such as 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 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 complexity class usually called "intractable" due to the time it takes to solve these problems (the best algorithms here are generally guess and check, which gets out of hand for large inputs i.e. possible combinations)(Papadimitriou, 2015)[1]. Using the lens of computational complexity, analysis of how many operations or how much memory a problem must take to solve, is one way of assessing what assumptions we are making about the computational power of our rational agents.

[1]NP is the class of problems that a given 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. NP-complete means that all problems in NP reduce to solving this problem. This is when you will usually hear "taking more time to compute than the age of the universe" etc...

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 scientist 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 (The actual formal definition is given in definition 2.1). An illustrative example of the price of anarchy from Roughgarden (2016, 15-16) are 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

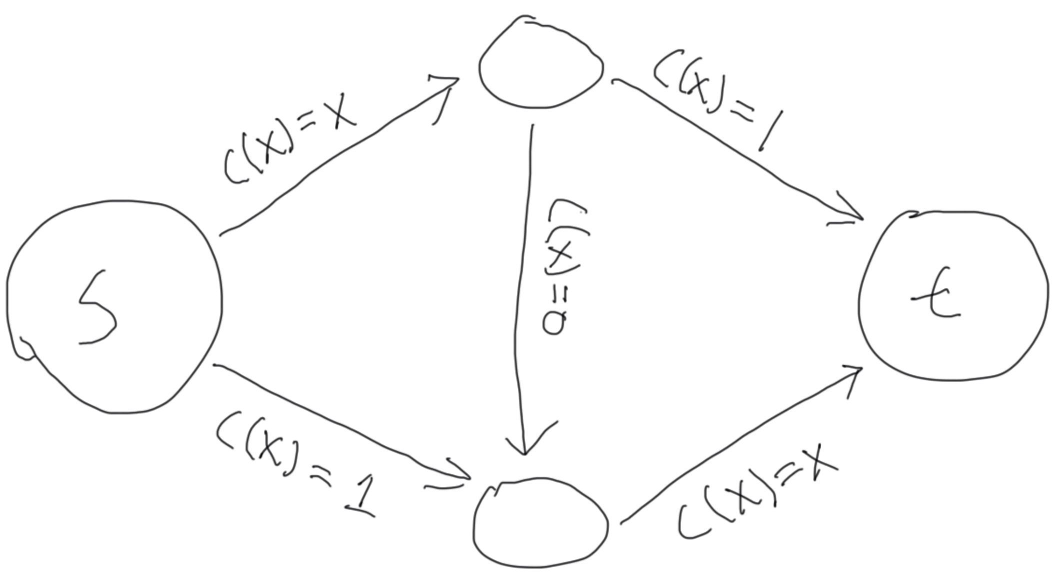

Figure 1.1: Routing game from s to t. Strategic interaction will make everyone worse off.

on the directed graph from the source s to reach the terminal t and tries to do it with the least cost, total weight of edges taken, as possible. The cost of taking these edges it is a function of the proportion of players who take that edge in the game. Some of them do not depend on how many players take it, but always cost a constant 1 or 0 where others are exactly proportional to how many players take it with a possible range $(0, 1]$. For example if 50% of the players take that edge then $x = 0.5$. This can be seen as analogous to 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 try to choose which path to take to get to their destination as quickly as possible. As you 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.

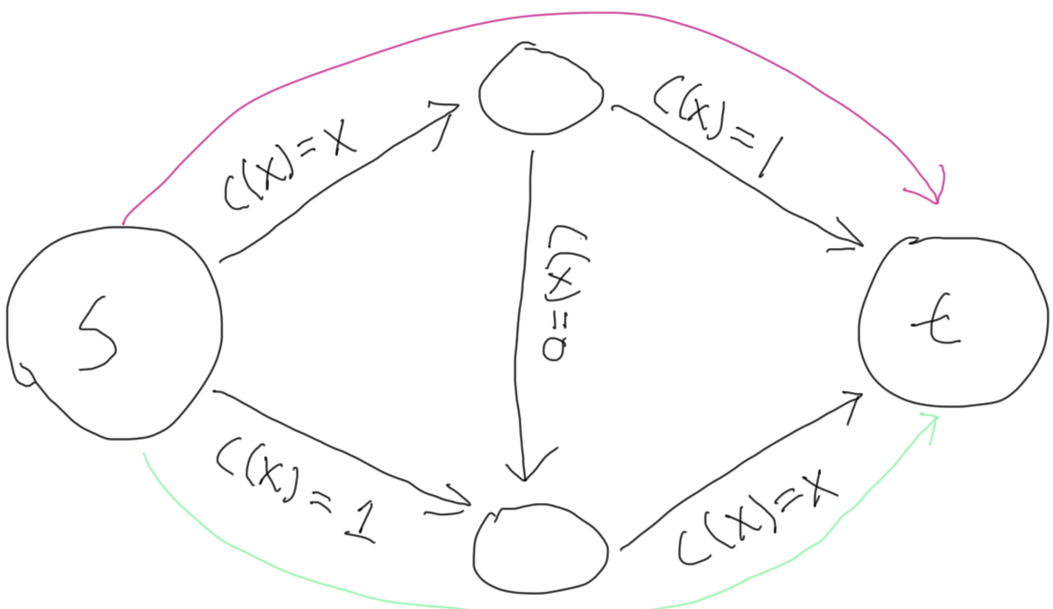

Figure 1.2: Socially optimal Routing

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 x, going through the middle, and then the bottom x. Only then will no players have reason to deviate. These actions to individually and strategically try to decrease their cost end up producing a worse outcome for all players and society as a whole (the social welfare)[2].

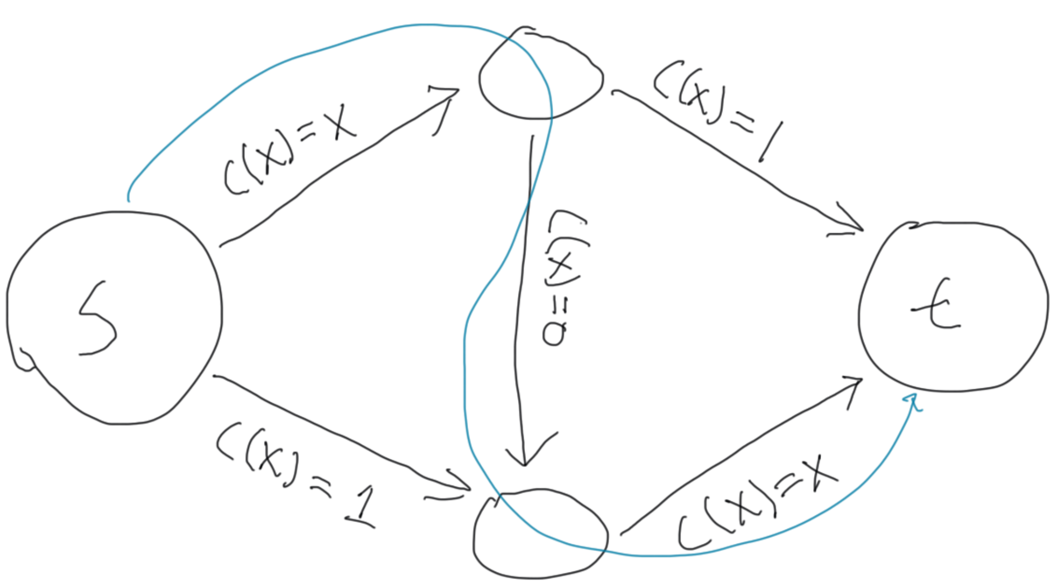

Figure 1.3: Route under strategic interaction

Now, it will take time 2 for each player to reach the terminal. Thus, under strategic interaction we see that the equilibrium is sub-optimal for society, and in this case all players as well. In this case, the price of anarchy can be computed as

$$POA = 3/4$$

(Roughgarden et al., 2017). For this network, there is only one (Nash) equilibrium for selfish routing games, however, price of anarchy bounds can be found for the class

[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 where having this extra edge counter intuitively leads to worse outcomes (Braess, 1968).

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 these selfish routing games with affine cost functions is know to be 3/4, meaning that our example is as bad as the price of anarchy could get for any network (Roughgarden, 2007).

1.1.2 Auctions

This framework of the price of anarchy is now being applied to understand auctions by researchers in the algorithmic game theory field. Where for selfish routing games, the price of anarchy has been shown to have the tight bound above, the price of anarchy for single price auctions only has approximate upper and lower bounds set on it, and is not known at all for double auctions. Before exploring this further, we should set up the mathematical framework of auctions that is necessary to compute such bounds.

An *auction* is a market mechanism, operating under specific rules that determines to whom one or more items will be awarded and at what price. For a bidder in an auction, the *value* v_i is how much the bidder values 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 is the winning bid (if multiple bids equal the winning bid, then some tie-breaking rule must be used). 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^*). Under the *first-price* rule, the bid submitted by the winner is equal to the selling price. Before the auction begins, each bidder knows their personal or private value for the item. An auction consists of a set of bidders, $I = (1, 2, ..., N)$ and a seller. After the auction, the bidder i wins the item if their bid is higher than the bid placed by any other bidder k, $b_i > max_{k \neq b_k}$. In a single unit 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

$$R^* = p^*.$$

There are multiple different pricing rules in an auction that determine who gets allocated the item. In a *first-price* auction the winning bidder pays the amount of their bid, which is the highest bid of the auction: $p^* = b^*$. Also called *pay-what-you-bid* (PWYB). 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) [3]. People tend to bid lower than their private valuation in first price auctions since if they bid that value, their profit is zero regardless of if they win. Thus, the first price rule the item could be awarded to someone who values the item less than other bidders. Second price auctions encourages bidder to bid their true values (as they will gain positive profits if they win no matter the second highest bid). This encourages an efficient allocation of items (Mochon and Saez, 2015). For the moment we will confine ourselves to first price auctions as this is where most of the strong results in price of

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

anarchy analysis of auctions currently are.

Before discussing the known bounds on the price of anarchy for first-price, single-item auctions, it is worth understanding how the bids might lead to a non-optimal outcome and what we mean by that. Due to only knowing their own valuation of the good, each bidder must act under uncertainty as to how much they should bid to beat out the unknown valuations of the other bidders. However, if they bid their exact private valuation then they will get a surplus of zero, or 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 that the other party values the item. To capture this interaction auctions are represented as Bayesian games where each bidder is drawing their bids from distributions known to the other player (that need not be the same). If one player knows that the other is drawing from a distribution with a smaller mean than they are (i.e. probably doesn't value the item as much), the Bayes-Nash equilibrium will have them shade their bid less and this other person will shade their bid more. This can lead to the person who values the item less winning the auction and creating less social welfare (the summed surplus of all bidders and the seller).

Syrgkanis and Tardos proved in 2013 that the lower bound of the price of anarchy in first-price, single-item auctions is at least $1 - \frac{1}{e} \approx 0.63$. The exact upper bound is unknown, but it has been proven that it can be no better than 0.87 (Hartline et al., 2015). This bound is true regardless of how many bidders there are or what distributions they are drawing their bids from so for any first-price single payer auction at equilibrium, we can say that it must be performing at least 63% as well as it could in the best case scenario.

This moves closer to answering the question for what is the price of anarchy at equilibrium, but these results do not pay attention to how the players arrive at these equilibria. In real world applications, we expect that players might play repeatedly in the same auction and learn as they play rather than come in with pre-computed strategies. This is especially true for when computing the equilibrium is computationally hard and the stakes of each individual auction is small. Given these observations, it is natural to ask questions about how the efficiency results carry over to adaptive game environments. The model for learning agents that is commonly used in the field 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) [4]. If each player incorporates this kind of learning algorithm, then it has been shown that these can converge to a larger class of equilibrium called correlated equilibrium where each player conditions their response on the expected action of the other player (Blum and Mansour, 2007). Luckily, the previous theorem has been extended so that we know that the price of anarchy for the set of coarse correlated equilibria of first price auctions are also at least 0.63 (Roughgarden et al., 2017).

With all of this set up, we now state our goal: to simulate no-regret learning algorithms for agents in a first price, single payer auction to see how well the price of anarchy holds. We also want to compare this with other learning algorithms that aren't no regret to see if this framework of using these algorithms is appropriate for making generalizations about equilibrium under learning.

[4]That is to say that the algorithm will converge to having a loss no worse than any fixed strategy we would have rather picked in hindsight as the limit goes to infinity (see definition. A more precise definition, example algorithms, and uses will be shown in chapter 2 to clarify what "regret" is and how this converges to zero (see definition 2.8).

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 being invented in the 1940's. Wassily Leontif used a computer to invert a 39 x 39 matrix to help solve his input output model. Since then, computers use in economics has exploded. With computers, economists are able to solve bigger matrices, do Monte-Carlo simulations, create multinomial probit models, and use full information maximum likelihood estimation. While 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. Within these simulated economies, theories can be coded into the simulation which, when run, can allow the researcher to conduct experiments that might not be practical to conduct in the real world.

One kind of a simulation that can be run is called an agent-based model (ABM), a simulated system of autonomous decision makers (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 to try and 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 the minority in their neighborhood and will keep moving otherwise (Schelling, 1969). This simple model can illustrate the main tenets of agent based modeling. The implementation we illustrate comes from Thomas Sargent and John Stachurski's 2019 lectures in quantitative economics [5]. First, we have

[5]These lectures are a treasure trove of information on how to use python to construct economic

agents who are representative of people in the real world. Their preferences to this respect are simple and easy to understand where 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 represented as the short procedure, or algorithm, shown below where S is just the space they live in.

1. Draw a random location in S

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 just say that it is a one by one unit square and we can say that their neighbors are the ten closest people to them (in Euclidean distance) on that square. When you run this simulation with green and orange dots representing the types of people you get the following behavior cycling through each of our 250 agents with the above procedure until every agent is happy.

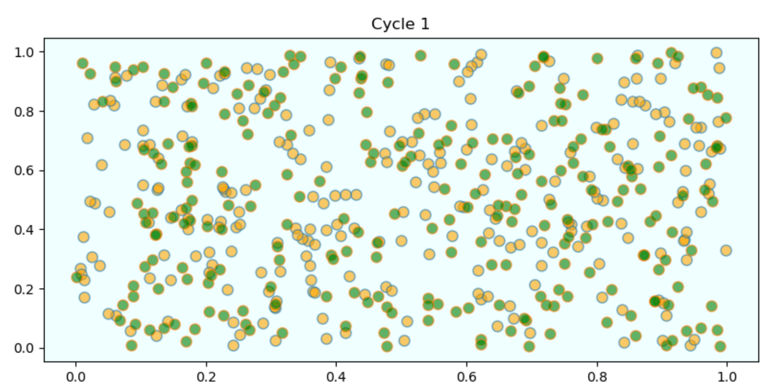

Figure 1.4: Schelling's Segregation Model: Cycle 1

As can be seen, in cycle 1 (Figure 1.4), the agents are well distributed among each other, but as they move in cycles 2-5, they become progressively more segregated. After 5 cycles all agents are happy and the simulation terminates. With very few

models. The simulation in chapter 3 was built in part using them as a guide.

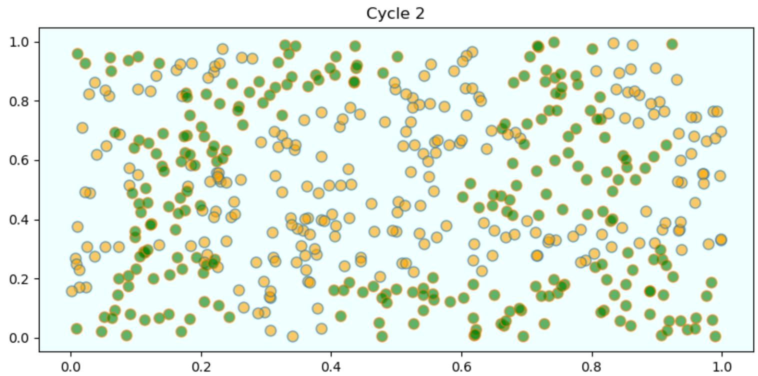

Figure 1.5: Schelling's Segregation Model: Cycle 2

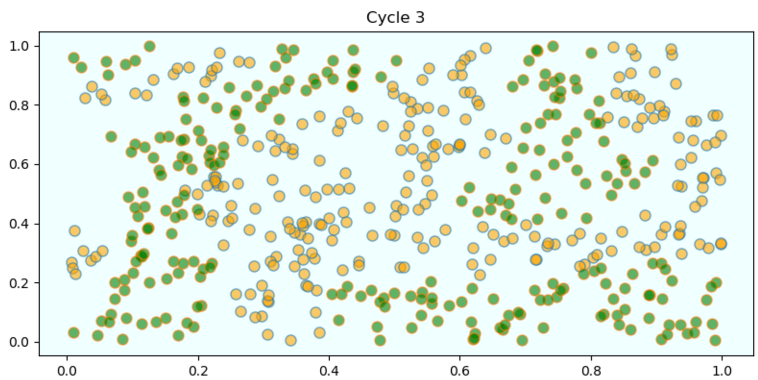

Figure 1.6: Schelling's Segregation Model: Cycle 3

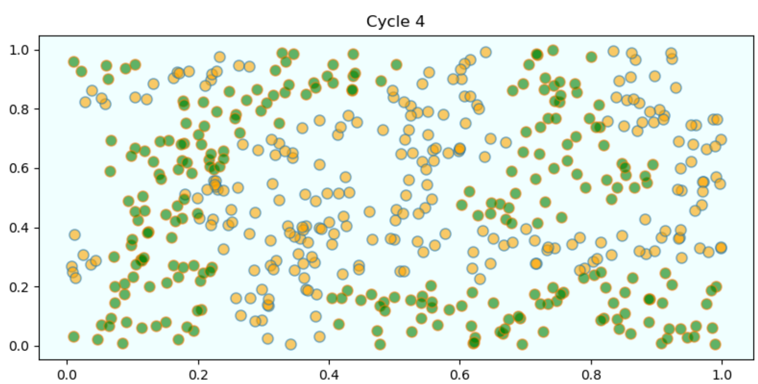

Figure 1.7: Schelling's Segregation Model: Cycle 4

assumptions about the agents' preferences, we can see the resulting emergent behavior of segregated neighborhoods in the system as a whole. Not only that, but the agents

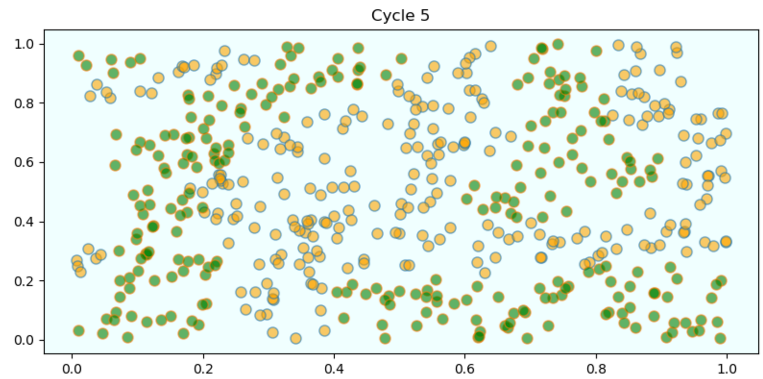

Figure 1.8: Schelling's Segregation Model: Cycle 5

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 then 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 using it 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 to try and simulate the behavior of agents on the stock market and how they adapt their adapting trading strategies effects the outcome of the market. This is one of the first examples of agents learning and adapting to their environments as part of the model (LeBaron, 2002). In this model 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 where the agents with worse strategies would randomly modify their own or take a more successful agents strategy (Arthur, 1992). This 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.

The authors of Santa Fe Stock Market paper at the time suggested that this was one of many algorithms that could be used to stand in for human behavior saying that reinforcement learning or deep learning could also be used to stand in for human intelligence[6]. More specifically they suggest that an appropriate algorithms can be chosen so long as they behave reasonably well like humans in that scenario, something like a Turing test (Arthur (1991)). Recent research in the field as well as in the field of multi-agent systems, a similar branch of computer science suggest that this might not be possible in the general case with such simple (or not fully intelligent) algorithms. Because these simulations have agents competing in non-stationary environments that are changing from the perspective of any individual agent every time some other agent changes their behavior, the choice of algorithm dramatically changes how the system behaves (Rejeb and Guessoum (2005), Shoham and Leyton-Brown (2008)). Further, research that has compared the behavior of human agents in strategic settings to that of these algorithms have found that there is no general algorithm that best approximates human learning(Tesfatsion (2002)). From situation to situation, different algorithms more appropriately behave like humans. This is a problem only assuming that you want your agents to in some way represent human behavior, we might simply want our agents to represent the "rational choice" in any given situation. These algorithms aren't necessarily doing that either. 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." The way they suggest to get around this is to try and build models that have robust behavior across algorithm choice. This is probably not true for their own artificial stock market as it is not even robust across the choice of parameters to program the market.

The literature of game theory provides a nice solution to this. The no-regret algo-

[6]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.

rithms they use have the simple property of doing better in the long run than fixing their strategy randomly in the beginning. This is a simple learning requirement that means that this kind of algorithm should be more robust across representing human behavior. It also allows us to combine our simulated models and analytical models in a nice way as we can mathematically describe the processes of such algorithms behavior and we can code agents to use algorithms that have that behavior. This allows the learning process to be less of a black box and instead to be as simple and thus generalizable as possible.

Moving forward this thesis aims to better understand the price of anarchy in auctions by simulating auctions using these no-regret learning algorithms and seeing how they behave compared to the socially optimal equilibrium. These simulations will try to answer two main questions, what equilibrium do these algorithms converge to (if they converge at all) and what is the calculated price of anarchy in these systems compared to the bounds that theory tells us should exist. Using this framework to explore auctions, we do this with single-item first price auction as it is the best understood theoretically and is the easiest 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 first price, single payer 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 primarily follows from the work of Tim Roughgarden, Vasilis Syrgkanis, and Éva Tardos who not only are all individually active in the field of algorithmic game theory and auctions, but also jointly authored the 2017 paper "The Price of Anarchy in Auctions" that is a survey of the entire topic. This chapter will be giving the important theorems and ideas that primarily come from this work and the individual work of these three authors, explaining them, demonstrating proofs where appropriate, and reconstruct some of the results that we will demonstrate with a simulation.

2.1 Auctions as Bayesian Game

Auctions are typically modeled as Bayesian games, also known as games with incomplete information. As you would expect, this is simply a type of game where the players don't know everything. 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 sealed bid auctions that makes them interesting! If all players came into an auction knowing the valuation of other players, for first-price, single-item auctions assuming there was no tie, the bidder with the highest value would always win by bidding the valuation of the next highest bidder. Rather it is the uncertainty that players face about the valuation of other players that make things interesting as players must guess how much they should shade their bid (as again no player will ever bid above their valuation) based off of what they know about other players.

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 will 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 want to eat the candy bar. For each bidder, we could say that this bidder has a probability distribution from which they are drawing their valuation from that is known to the other bidders. 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 (allowing for more black swan events) centered at \$5. This sort of strategic interaction where the players know the distributions that 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 are games in which one or more of the players don't have "full knowledge" of the game that is being played. Introduced by John C. Harsanyi in 1967, rather than players knowing every parameter of the game situation such as utility functions, possible strategies, and information held by other players, each player knows a probability distribution from which these will be drawn. In his paper, Harsanyi says that this type of game can be thought of as a normal game, where "nature" goes first drawing from these probability distributions and assigning values before play begins without the players knowing which specific variation of the game they are playing (Harsanyi, 1967). 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 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. Letting $T = T_1 \times \ldots \times T_n$ and $F = F_1 \times \ldots \times F_n$.

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 $s_1, \ldots s_n$.

Using this definition to model our auction the types of players will consist of the publicly known distribution from which they are drawing their valuation. That is, our auction will consist of bidders who know their own valuation of the item being bidded on and the distribution from which each of the other players is drawing their own valuations. In a first-price, single item auction if player i wins with bid b_i, we define the utility of the winner to be $u_i = v_i - b_i$, the difference between their valuation and their bid and the losers all get $u_i = 0$ since they did not receive the item. 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 $s_i(v_i)$ (i.e. taking into account the probability distribution for the other players).

We move 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 for all players in which each individual player cannot increase their utility by deviating from their strategy fixing the strategy of all the other players. 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 for which players are drawing from.

Definition 2.2. (Roughgarden et al., 2017) A strategy profile 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.

2.1.2 Formally Defining First-Price Auctions

To analyze an first-price auction as a game, we give the notation we will be using. This notation comes from Roughgarden et. al's 2017 survey of the subject of the price of anarchy in auctions. For a bid profile $\mathbf{b} = (b_1, \ldots, b_n)$, we let

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

.

We let $p(\mathbf{b}) = \max_{i \in \{1,\ldots,n\}} b_i$ denote the selling price. The utility that a player i receives when their valuation is v_i is

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

.

We denote the *strategy profile*, or vector of strategies played by each player, by $\mathbf{s} = (s_1, \ldots, s_n)$, where each s_i is a function for player i's valuation v_i to their bid. We then let $\mathbf{s}(\mathbf{v})$ denote the strategy vector resulting from the vector of valuations $\mathbf{v}$. For any given vector $\mathbf{x}$, we use $\mathbf{x}_{-i}$ to denote the vector $\mathbf{x}$ with the ith element removed. Using this notation a first-price auction we can say that a strategy profile $\mathbf{s} = (s_1, \ldots s_n)$ is a Bayes-Nash equilibrium if and only if

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

(Roughgarden et al., 2017).

2.1.3 Example Auction

We now turn to an example auction to clarify what has just been laid out. In this example we analyze the an auction between two players, Alice and Bob who are bidding on a candy bar where each select their valuations from the uniform distribution $[0, 1]$. This is a first-price, sealed bid auction where they each submit a bid for the candy bar simultaneously. How are Bob and Alice supposed to decide what to bid on this auction?

Proposition 2.3. *In the first-price, sealed bid auction with valuation distributed on* $[0, 1]$*, the unique Bayesian-Nash equilibrium is* $\boldsymbol{s} = (s_1(v_1) = v_1/2, s_2(v_2) = v_2/2)$*.*

Proof. 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 $v_1 - x$, and if she losses, 0. 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 F_2, the prior distribution of v_2. Now, Alice wins if $x \geq v_2/2$, and since v_2 is distributed uniformly in $[0, 1]$ we can calculate the probability: $2x$ for $0 \leq x \leq 1/2$, 1 for $x \geq 1/2$, and 0 for $x \leq 0$. We see that the optimal value of x is in range $0 \leq x \leq 1/2$ since $x = 1/2$ is better 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 range $0 \leq x \leq 1/2$. Taking the derivative and setting this equal to zero, we get $2v_1 - 4x = 0$, which has solution $x = v_1/2$ (Nisan, 2007). □

Proving the uniqueness of this equilibrium requires the use of a fair bit of algebra and solving a differential equation. To see an example of proving the uniqueness see Levin (2002).

2.1.4 Efficiency of First-Price Auctions

Examples of the Bayes-Nash equilibrium have been solved for various combinations of the number of players and distributions from which they draw their valuations. With n bidders it has been shown that the Bayes-Nash equilibrium strategy vector is composed of $s_i(v_i) = \frac{n-1}{n} v$ for all players i (Chawla and Hartline, 2013). Here the equilibrium is easy to calculate and efficient (meaning that the item will always be allocated to the player with the highest valuation). Neither efficiency nor ease of calculation are guaranteed for Bayes-Nash equilibria in this auction format. 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 $[0, 2]$ it has been shown by Krishna (2002) that the Bayes-Nash equilibrium for this auction is:

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

Here bidder one with the smaller 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. This is inefficient. Moreover, it has been shown that solving many of these asymmetric Bayes-Nash equilibrium requires finding a solution to a system of partial differential equations many of which have no closed-form solution (Roughgarden et al., 2017). So even if we expect bidders to do their homework before an auction, they still might not know what to do. Given this, it is extremely hard to characterize or say things about what solutions to this format of auctions look like in general. However, just because we are not able to give a closed form for all of these equilibria, that does not mean we aren't able to characterize them in other ways.

2.1.5 Price of Anarchy in First-Price Auctions

To try and get a sense of how inefficient first-price auctions can be, computer scientists have been applying a concept known as the *price of anarchy*, one way to characterize systems at equilibrium[1]. 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. We must first define these concepts and then move to the point at hand. In the case of an auction, the social welfare is the sum of the utilities of the players plus the revenue of the auctioneer.

Definition 2.4. The *social welfare* of a bid profile $\mathbf{b}$ when the valuation profile is $\mathbf{v} = (v_1, \ldots, v_n)$ 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 cancels out. Hence, the social welfare is maximized when the bidder with the highest valuation wins the auction. If 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), the maximum possible social welfare in a single-item auction is

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

Now that we have mathematically described the social welfare in our system, we can define the price of anarchy.

Definition 2.5 (Price of Anarchy)**.** The *price of anarchy* of an auction, with a

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

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 equilibrium $\mathbf{s}$ 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 first-price single-item auctions. 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[2].

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 incredible, 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 2.6. *(Syrgkanis and Tardos, 2013) The price of anarchy in first-price single-item auctions format is at least* $1 - \frac{1}{e} \approx 0.63$

Theorem 2.6 tells us that no matter how many players we have or what weird

[2] 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.

distributions we try and 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 (if it exists).

2.2 Extending Results to No-Regret Agents

These results hold for simple 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 way to express it? Secondly, 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 use a set of extension theorems that take us through a general mechanism design setting and allow the class of equilibria our bounds hold for to be expanded. This extension will take us to a concept of no-regret learning. We briefly sketch key theorems from this paper that lead us to an equilibrium concept that applies to learning agents.

2.2.1 General Auction Mechanisms

In order to understand (or even state) the proofs and theorems required to take us to our expanded set of equilibria, we must introduce more notation and the idea of a general mechanism design setting and a general auction setting (this is unfortunate for us due to an expansion of scope, but actually makes these theorems quite powerful!). Mechanism design can be thought of as reverse game theory. Instead of thinking about how to play a game to maximize your utility, you are now thinking about how to design a game so that rational agents will behave in a certain way. This influence on players is generally exerted through payoffs and cost to play in the game. While we will not be spending much time with this math, it is useful to note that this framework reflects an ability for the auctioneer to fine tune how the game is played. In a general mechanism design setting, the auctioneer solicits an action a_i from all of the players, i, from some action space $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$. Given an action profile $\mathbf{a} = (a_1, \ldots, a_n) \in \mathcal{A}$, the auctioneer decides an outcome $o(\mathbf{a})$ among the set of possible outcomes $\mathcal{O}$. This outcome includes a payment $p_i(o)$ that each player must give to the auctioneer. Denote the revenue of the auctioneer $\mathcal{R}(O) = \sum_i p_i(o)$. Players receive some utility as a function of their valuation, v_i and the outcome which we write $u_i(o; v_i)$. Let $\mathcal{V} = \mathcal{V}_1 \times \cdots \times \mathcal{V}_n$.

2.2.2 Smooth Auctions

Smooth auctions are a way of describing auctions in the general mechanism design framework that allow all possible deviations by players to be better accounted for in the math. The details of this are omitted from this paper, but for the complete details on this see Roughgarden et al. (2017). The definition of a smooth auction is as follows, where D_i^* is a new concept of an action distribution that is just a priori distribution over the belief about what actions the other players will take

Definition 2.7 (Smooth Auctions)**.** For parameters $\lambda \geq 0$ and $\mu \geq 1$, an auction is (λ, μ)-smooth if for every valuation profile $\mathbf{v} \in \mathcal{V}$ there exist action distribution $D_1^*(\mathbf{v}), \ldots, D_n^*(\mathbf{v})$ over $\mathcal{A}_1, \cdots, \mathcal{A}_n$ such that for every action profile $\mathbf{a}$,

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

Now, following from the proof of theorem 2.6, Roughgarden et. al show that first price, single payer auctions are $(1 - \frac{1}{e}, 1)$-smooth. This is done by using the last step of their proof of theorem 2.6. They then go on to develop a series of extension theorems that allow us to characterize 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.

2.2.3 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 observes 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 2.8 (Vanishing Regret)**.** A sequence of action profiles $\mathbf{a}^1, \mathbf{a}^2, \ldots, a^T$ 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 2.9 (Extension to Vanishing Regret Sequences)**.** *((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 $1 - \frac{1}{e}$.

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 the price of anarchy for this format. First, the construction of this simulation is discussed. Next, we demonstrate the results of running the simulation on bidders using known Bayes-Nash equilibrium strategies and an arbitrary bidding strategy. Finally, we demonstrate that these bounds also hold for no-regret learning agents in a fixed action variant of the first-price auction.

3.1 A Simulated Auction Environment

To simulate sequential first-price auctions, we 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 we 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 where 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 that round. At each round the total social welfare is v_i, the valuation of the winning bidder as per definition 3. 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 we use the super script t to denote which round each variable is from[1].

Algorithm 1: Sequential First-Price Single-Item Auction

```
Initialize SW, OW, and POA to zero
for t = 1, ..., 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 assgined 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 ← SW + v_i^t
    if player j has the highest valuation then
        OW ← OW + v_j^t
    POA ← SW/OW
```

3.1.1 Simulating Bayes-Nash Equilibria

Using the simulation outlined above, we first demonstrate that bidders using known Bayes-Nash equilibrium strategies have an average price of anarchy greater than 0.66. First, we simulate the case of two bidders each drawing their valuation from the

[1]The "←" 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.

uniform distribution $[0,1]$. We are using the hard coded strategies that $s(v_i^t) = \frac{v_i^t}{2}$ for each bidder which was shown to be the unique Bayes-Nash Equilibrium in chapter 2. 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: Price of anarchy in two player symmetric auction

As can be seen this example is somewhat silly to simulate since this equilibrium is fully efficient. As each player bids half 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:

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

We now move to simulate the more interesting example of two bidders with asymmetric distributions. We let bidder one chose their valuation from the uniform distribution over the interval $[0,1]$ and bidder two chooses their valuation from from the uniform distribution on the interval $[0,2]$. As stated in chapter 2, the unique Bayes-Nash equilibrium for this auction is:

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

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

As stated in chapter 2, 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) in this equilibrium. We 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: Price of anarchy 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 auction we see that they are well above the 0.66 lower bound guaranteed for all first-price single-item auction formats.

3.1.2 Minimal Intelligence Bidders

Before going on to simulate the auction using no-regret bidders, we first move to establish a baseline for how well each auction setting (based on number of bidders and distribution choice) performs with agents that are not learning. To do this we construct agents that bid randomly between zero and their valuation every round i.e. each players bid is chosen uniformly from the distribution $[0, v^t]$. We call these agents minimally intelligent since they are not overbidding, but they are also clearly using a nonsensical strategy for an auction. One should note that while this is a bad strategy, it is possible to formulate much worse strategies for our bidders from both utility maximization and efficiency standpoints [3]. This rather represents agents who are incapable of learning or doing their homework and thus randomly select from all options. The results for simulating sequential first price auctions with 2, 10, and 100 agents using this strategy are shown in the table 3.3 below.

Round	2 Agent POA	10 Agent POA	100 Agent POA
1	0.9000	0.8410	0.9859
10	0.9312	0.8987	0.9419
1,000	0.9279	0.8812	0.9480
10,000	0.9150	0.8880	0.9467
100,000	0.9164	0.8887	0.9470

Table 3.3: Price of anarchy in two player asymmetric auction

Table 3.3 shows how the price of anarchy evolves through the rounds as the number of rounds goes to 100,000. Since the agents are not learning anything and the bids are randomly sampled each time, these efficiency results should approximately converge to the expected value as $T \to \infty$. To better illustrate the relationship between price of anarchy in each of these sequential auctions and the number of bidders, we graph the above simulation now run on 2 to 100 bidders. Each of these simulations is run

[3] It's rather fun to think up such strategies! For example, if each bidder shaded little when they had a low draw and shaded a lot when they had a high valuation, this can lead to highly inefficient outcomes

100 times, and the min, max, and average POA are graphed below in figure 3.1.

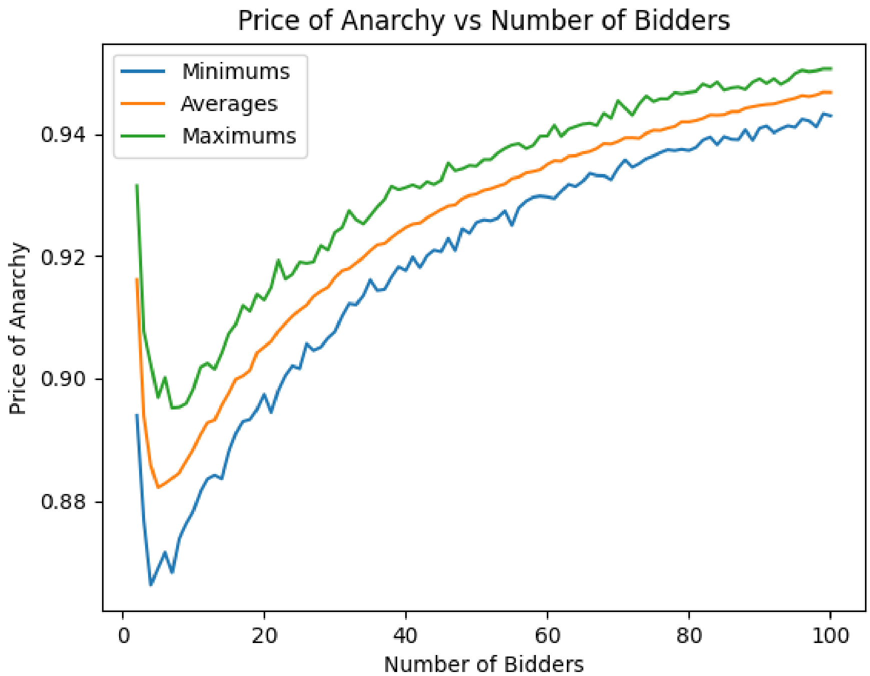

Figure 3.1: POA for 2 to 100 symmetric bidders

We see three things, one is that when all bidders are drawing from the same uniform distribution the market is incredibly efficient regardless of how smart the bidders are when choosing their strategy. That is, even when their strategies are arbitrary, this auction still performs reasonably well. The second thing to note is that in this setting, as the number of bidders increases the social welfare also increases. This makes sense as we would expect the probability of a reasonably high valuation winning to increase as there are more valuations per round. The third thing we notice, and most surprising of all, is that the increase in the price of anarchy as the number of players increases is not monotonic. At lower level of bidders, the efficiency actually decreases as we add more bidders. The minimum POA here is especially low.

We now conduct a similar simulation but in the case of asymmetric bidders. In this case half of the bidders are drawing uniformly from $[0, 1]$ and half from $[0, 2]$. The results are shown in table 3.4.

Round	2 Agent POA	10 Agent POA	100 Agent POA
1	0.1.000	1.0000	0.9726
10	0.9115	0.8178	0.9331
1,000	0.9054	0.8658	0.9310
10,000	0.9092	0.8630	0.9297
100,000	0.9112	0.8640	0.9304

Table 3.4: Price of anarchy in two player asymmetric auction

Again we see that the market is quite efficient in this case regardless of how smart the bidders are. We again also see that it seems to converge to optimal (i.e. to 1) as the number of bidders increases but non-monotonically. We simulate this for 2 to 100 bidders again 10 times at each number of bidders, where on odd numbers we allow the extra bidder to have distribution $[0, 1]$. This is shown is figure 3.2.

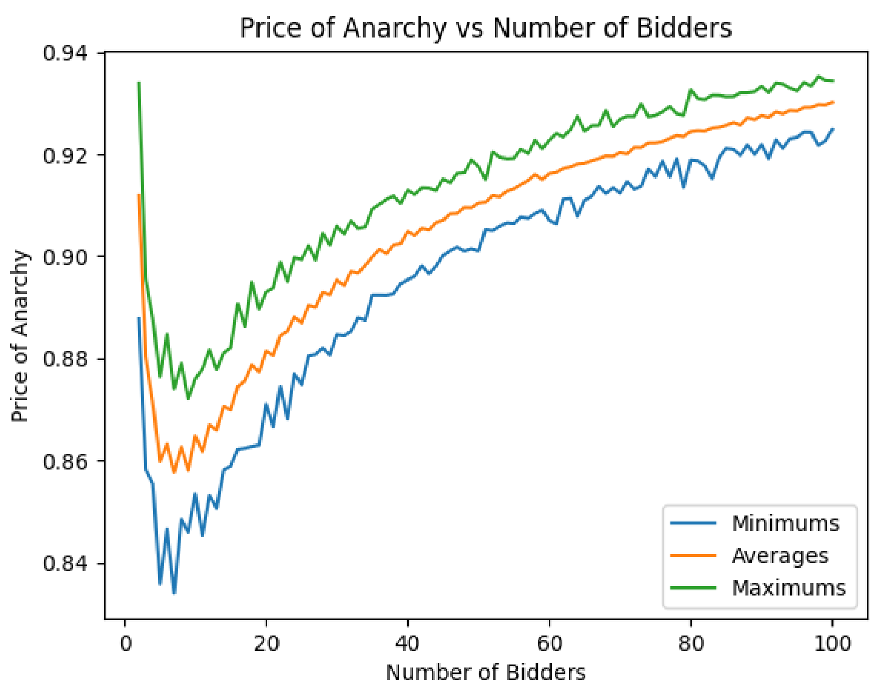

Figure 3.2: POA for 2 to 100 asymmetric bidders

Note that the baseline POA for minimal-intelligence agent auctions changes as we change the distributions. While the format seems quite efficient and well above the price of anarchy bounds (which again are only guaranteed for equilibria which this is not) even with agents following these minimally intelligent strategies, we have no guarantee that there is not some way to construct this that these agents wouldn't do much worse. Given that it seems as the number of agents increases, the efficiency also seems to increase, one would expect that this would need to be done by picking more interesting distributions for the bidders to choose from and only having two bidders.

3.2 Simulating No-Regret Bidders

With the results from the Bayes-Nash equilibrium demonstrated and an efficiency baseline established, we move on simulating the bounds of no-regret learning agents in first-price auctions. Again, this is the equilibria we expect auctions to converge to if each agent is using a no-regret learning algorithm. To use these algorithms we do however have to make a concession to the environment we are simulating: we must now simulate an auction where the bidders are only allowed a finite number of actions.

3.2.1 Multiplicative Weights Algorithm

The no-regret learning algorithm we will use to train our bidders is called the multiplicative weights algorithm. It has been shown to satisfy the no-regret property in papers such as TODO and TODO but our implementation of it comes from Roughgarden (2016) and thus if each of our bidders use it our auction should converge to a coarse correlated equilibrium.

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. 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 among them to decide what to do. 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 then 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. However, in the general case this algorithm has been shown to be no-regret in the face of an adversary directly picking the utilities the learning agent receives.

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 this algorithm is simple. At each time step we see how well each of the possible actions performed and increase the weight, or probability of selecting that action in the future. This increase is done proportionally to how well the action did as determined and *learning rate*, η chosen before starting the procedure. As demonstrated in Roughgarden 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 know that 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. This 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 equilibria. Part of the beauty of using multiplicative weights is that it is simple and has nice mathematical properties. Using more interesting learning techniques these properties might no longer hold and then we would have to wonder what our simulation is really showing[4]?

3.2.2 Uniform Distribution Simulations

The first simulation we run is a repeat of the symmetric auction setting, but now using agents learning with the multiplicative weights algorithm. We 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, we 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 3.5

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

Table 3.5: Price of anarchy in two player asymmetric auction with no-regret learning

[4]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.

Here we can see that after 100,000 rounds the price of anarchy converges 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. Since each time we run the simulation it is possible for the bidders to learn to converge to a new equilibrium, the POA values for each simulation can be different. Averages don't make sense in this context as we are more concerned with the lowest possible POA that the system converges too. We now repeat this using 2 through 100 agents, each symmetric and drawing from a uniform $[0, 1]$ as above. We run 100 simulations with each number of agents, each for 100,000 rounds[5]. This gives us the following results as shown in figure 3.3

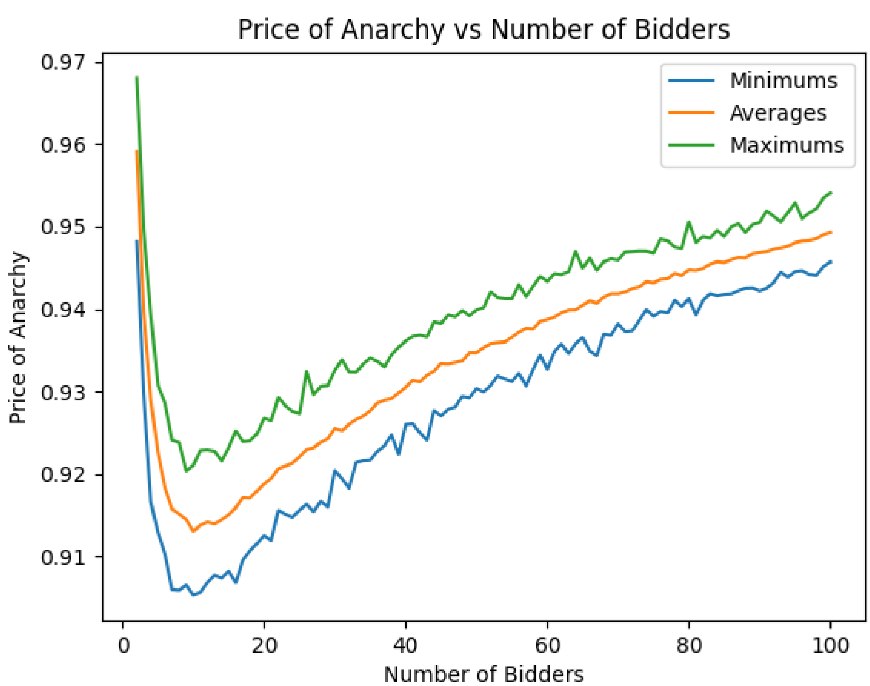

Figure 3.3: POA for 2 to 100 symmetric bidders

We can see here that our efficiency in the minimum case with our learning agents is about as good as the best-case with the random guessing agents. It makes since

[5]This is a relatively small number, but necessarily chosen for the sake of computation time with 1,000 agents

that the efficiency would increase as we would expect people with higher valuations to win more often as people learn to minimize their regret. It is theoretically possible in some games that social welfare could be better off with everyone choosing arbitrary strategies then under learning agents, but we see in this game, and with our learning agents that is not the case.

Finally, we 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 3.6 below.

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

Table 3.6: Price of anarchy in two player asymmetric auction with no-regret learning

Here we see the agents converge to a very good 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 we conduct this simulation with 2 to 100 asymmetric bidders. For each number of bidders we simulate this 100 times, 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

Figure 3.4 is similar to the figures we have seen for every other auction. We see very high efficiency for two or three bidders, and then see a dramatic decrease as it goes to 9. We then see the numbers slowly climb back towards being fully efficient in the best, worst, and average cases.

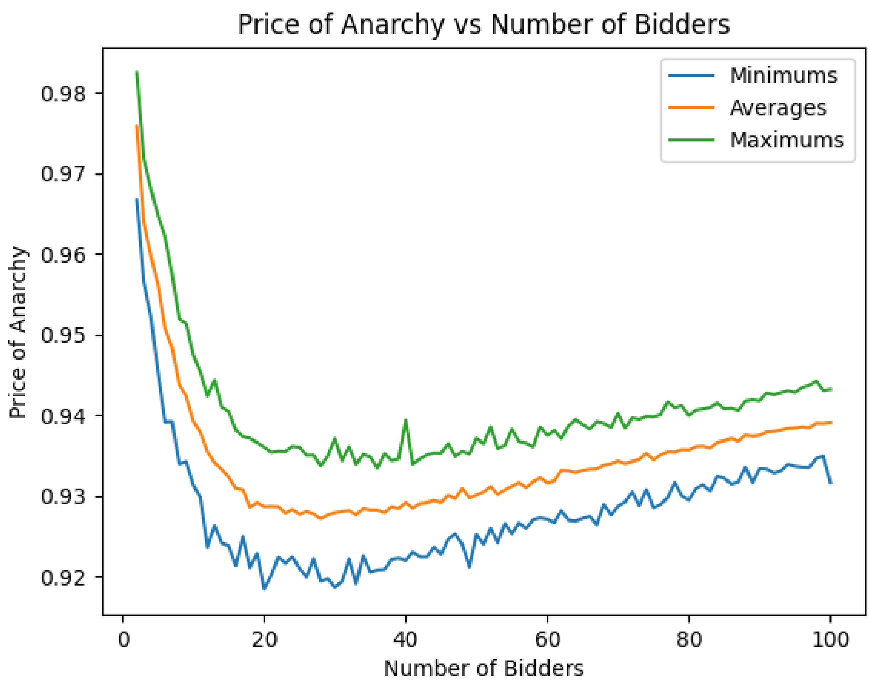

Figure 3.4: POA for 2 to 100 asymmetric bidders

Conclusion

Here's a conclusion, demonstrating the use of all that manual incrementing and table of contents adding that has to happen if you use the starred form of the chapter command. The deal is, the chapter command in LaTeX does a lot of things: it increments the chapter counter, it resets the section counter to zero, it puts the name of the chapter into the table of contents and the running headers, and probably some other stuff.

So, if you remove all that stuff because you don't like it to say "Chapter 4: Conclusion", then you have to manually add all the things LaTeX would normally do for you. Maybe someday we'll write a new chapter macro that doesn't add "Chapter X" to the beginning of every chapter title.

4.1 More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

Appendix A

The First Appendix

Appendix B

The Second Appendix, for Fun

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). URL https://ideas.repec.org/p/qed/wpaper/854.html

Auchincloss, Amy H., and Leandro Martin Totaro Garcia. 2015. Brief introductory guide to agent-based modeling and an illustration from urban health research. *Cadernos de Saúde Pública* .

Axtell, Robert. 2005. The complexity of exchange. *Economic Journal* 115(504): 193–210.

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* 1–36.

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.

Blume, Lawrence, David Easley, Jon Kleinberg, Robert Kleinberg, and Éva Tardos.

2015. Introduction to computer science and economic theory. *Journal of Economic Theory* 156(November): 1–13.
URL http://dx.doi.org/10.1016/j.jet.2014.11.002

Bonabeau, Eric. 2002. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences of the United States of America* 99(SUPPL. 3): 7280–7287.

Boutilier, Craig, Yoav Shoham, and Michael P. Wellman. 1997. Economic principles of multi-agent systems. *Artificial Intelligence* 94(1-2): 1–6.

Braess, D. 1968. Uber ein paradoxen der verkehrsplanung. *Unternehmensforschung* 12: 258–268.
URL http://homepage.ruhr-uni-bochum.de/Dietrich.Braess/paradox.pdf

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

Harsanyi, John C. 1967. Games with Incomplete Information Played by "Bayesian" Players, I–III Part I. The Basic Model. *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. *Advances in Neural Information Processing Systems* 2015-Janua: 3061–3069.

Hartline, Jason D., and Anna R. Karlin. 2007. Profit maximization in mechanism design. In *Algorithmic Game Theory*, vol. 9780521872, 331–362.

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.

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, 1 edn.

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

Lugosi, Gábor. 2006. *Prediction, learning, and games*, vol. 9780521841.

MacAl, C. M., and M. J. North. 2010. Tutorial on agent-based modelling and simulation. *Journal of Simulation* 4(3): 151–162.

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, 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 .

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* 9780521899: 1–483.

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.

Tesfatsion, Leigh. 2005. AGENT-BASED COMPUTATIONAL ECONOMICS : A CONSTRUCTIVE APPROACH TO ECONOMIC THEORY 18 December 2005. *Computational Economics* 2(October 2003): 831–880.
URL http://linkinghub.elsevier.com/retrieve/pii/S1574002105020162

van der Hoog, Sander. 2016. Deep Learning in Agent-Based Models: A Prospectus. *SSRN Electronic Journal* (02).

Vickrey, William. 1961. American Finance Association Counterspeculation , Auctions , and Competitive Sealed Tenders Author (s): William Vickrey Source : The Journal of Finance , Vol . 16 , No . 1 (Mar ., 1961), pp . 8-37 Published by : Wiley for the American Finance Associat 16(1): 8–37.