NORTHWESTERN UNIVERSITY

Near-Optimal Non-Truthful Mechanism Design

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Computer Science

By

Samuel Taggart

EVANSTON, ILLINOIS

September 2017

ABSTRACT

Near-Optimal Non-Truthful Mechanism Design

Samuel Taggart

This dissertation considers mechanism design and redesign for markets like Internet advertising where many frequent, small transactions are organized by a principal. Mechanisms for these markets rarely have truthtelling equilibria. In contrast to previous work which analyzes existing mechanisms, we initiate the study of non-truthful mechanisms explicitly from a design perspective. We identify a family of winner-pays-bid mechanisms that exhibit three properties. First, equilibria in these mechanisms are simple. Second, the mechanisms' parameters are easily reoptimized from the bid data that the mechanism generates. Third, the performance of mechanisms in the family is near the optimal performance possible by any mechanism (not necessarily within the family). Our mechanisms are based on batching across multiple iterations of an auction environment, and our approximation bound is asymptotically optimal, with loss inversely proportional to the cube root of the number of iterations batched. Our analysis methods are of broader interest in mechanism design and, for example, we also use them to give new sample complexity bounds for mechanism design in general single-dimensional agent environments.

Acknowledgements

Over the last five years, I have grown enormously as a scholar and educator. For each achievement of mine are at least ten people who collectively deserve most of the credit.

First and foremost, I have been incredibly fortunate to have Jason Hartline as an advisor. He taught me, both by example and with attentive mentorship, to pursue excellence in every facet of scholarship. Nothing I've done as a doctoral student would have been possible without his help. I have also benefitted from the attention of numerous other mentors, including Nicole Immorlica, Brendan Lucier, Chris Wilkens, and David Parkes. They each have my gratitude.

I feel very lucky to have entered a research group at Northwestern that was also a tight-knit community. My Northwestern colleagues, including Nima Haghpanah, Bach Ha, Aleck Johnsen, Rachel Cummings, Greg Stoddard, Manolis Pountourakis, Darrell Hoy, Yorgos Askilidis, Abhratanu Dutta, Yiding Feng, Alex Wang, and Madhav Suresh were not just co-workers, but close friends. I especially want to thank Darrell and Manolis. Darrell's mentorship was an indispensible resource in my first few years of study. Manolis has made my research time more fruitful and entertaining by a huge margin.

At other institutions, there is a long list of friends who deserve thanks. This includes the students, researchers, and interns at Harvard and Microsoft Research who welcomed me during my time in Boston, Rediet Abebe, Steven Wu, Kira Goldner, Irene Lo, Jamie Morgenstern, Matt Weinberg, Debmalya Mandal, Vasilis Syrgkanis, Zihe Wang, Ben Miller, Rad Niazadeh, Giro Cavallo, and the Mechanism Design for Social Good reading group. The work in this thesis in particular has benefitted from conversations with Jamie and Vasilis.

I want to thank Aravindan Vijayaraghavan and Marciano Siniscalchi for serving on my committee.

I want to thank all of my other friends at Northwestern and elsewhere for supporting and entertaining me during my studies. Sneha Narayan and Matt Ford deserve special thanks for being such wonderful comrades.

Finally, I want to thank my parents and my brother. Every good thing I've ever done can in some way be traced back to them.

Dedication

For my parents.

Table of Contents

CHAPTER 1

Introduction

This thesis considers the design of algorithms for allocating a resource to strategic agents, i.e. *mechanisms*. The classical theory of mechanism design focuses on mechanisms in which agents are incentivized report their values for allocation truthfully. The design of such truthful mechanisms is well-understood. Mechanisms used in practice, however, are often not truthful. Common examples include the ubiquitous first-price auction (often implemented as a descending-price auction) and the auctions used to sell advertising online. These non-truthful mechanisms which appear in practice have been well-studied, and much is known about their performance under a broad range of assumptions. The proliferation of such non-truthful mechanisms, however, suggests that there is need not just for analysis, but for *design* of mechanisms which both perform well and conform to the practical needs that lead to the use of non-truthful mechanisms in the first place.

This thesis works towards such a theory for non-truthful mechanism design and inference for mechanism redesign in markets like Internet advertising where many frequent, small transactions are organized by a principal. In these environments bidders place bids in advance and, as goods become available, the bids of the relevant bidders are entered into an auction to determine the allocation of goods. Such settings are among those where market mechanisms do not have truthtelling equilibria and, thus, it is challenging to reason about their design and, when bid data is available, their redesign. Our theory therefore focuses on the canonical model of mechanism design, the independent private value model, under the additional constraint that mechanisms must satisfy the constraints of practical applications with non-truthful semantics such as winner-pays-bid/first-price.

1.1. Why Non-Truthful Mechanisms?

As discussed, much of the theory of mechanism design is developed via the *revelation principle* (Myerson, 1981) which observes that existence of a mechanism with good equilibrium implies the existence of one with a good truthtelling equilibrium. There are many practical constraints, however, that prevent mechanisms with truthtelling equilibrium from being adopted (see Ausubel and Milgrom, 2006). Practitioners instead often employ non-revelation mechanisms such as those with winner-pays-bid (i.e., first-price) semantics: agents submit bids to the mechanisms, winners are selected from bids, and winners pay the bids they submitted. Such mechanisms tend to be more robust to risk averse bidders (e.g. Ausubel and Milgrom, 2006). Moreover, winner-pays-bid mechanisms are transparent - bidders know exactly what they will pay on winning. Computation of payments is trivial for the mechanism; thus winner-pays-bid mechanisms also essentially offload the more complicated payment computation that takes place in truthful mechanisms to the agents instead, which is advantageous when transactions are frequent.

1.2. Philosophy: Non-Revelation Mechanism Design

The non-truthful mechanisms that are common in practice have been well-studied. For example, Kirkegaard (2009) and Kaplan and Zamir (2012) give detailed characterizations of the equilibria of first-price auctions. More recently, literature on the *price of anarchy* has sought to compare the social welfare and revenue of commonly used non-truthful mechanisms to those of the optimal mechanisms. (See, for example, Syrgkanis and Tardos, 2013; Hartline et al., 2014) Such results provide a deeper understanding of existing mechanisms, but do little to guide the design of non-truthful mechanisms which improve on those in use.

We note here some partial approaches to the problem of non-truthful mechanism design. For position auctions, which model the sale of advertising space, Chawla et al. (2014, 2016) develop econometric inference tools which can then be used to decide how to reparametrize rank-based auctions for environments such as those encountered in the sale of online advertising. Their approach does not design new mechanisms, however. For the sale of multiple heterogeneous items, Devanur et al. (2015) design new non-truthful mechanisms with the goal of producing auctions with simple strategy spaces simplify bidders' problem of learning optimal strategies. These mechanisms have the property that even when agents have complicated preferences for sets of items, a single first-price bid, chosen in equilibrium, suffices to capture the information necessary to allocate items approximately efficiently.

This thesis attempts to initiate the development of a theoretical framework to aid in the systematic design of non-truthful mechanisms. We do so in a special case of the single-dimensional, independent private values model for mechanism design, in which the seller has access to many transactions at once, and may make decisions for these transactions jointly. In this model, we highlight some of the theoretical properties which complicate or facilitate design and inference for non-truthful mechanisms.

In what follows, we identify a family of practical, non-revelation mechanisms for general environments, including single-minded combinatorial auctions, that exhibit three properties. First, these mechanisms use limited distributional information to obtain performance near that of the optimal mechanism, a $1 + \epsilon$ approximation for any desired ϵ. Second, these mechanisms can be easily parameterized and reoptimized from the equilibrium bid data they generate. Mechanisms from such a family can be tuned as fundamentals of the market evolve. Third, the equilibria in these mechanisms have a focal equilibrium which is simple and are therefore easy to analyze.

The *iterated population model* expands on a standard interpretation of the independent private value model. In this model there are a collection of populations, and each population consists of a continuum of bidders which induces a distribution of values. The strategies of these bidders induce a distribution of bids. In each iteration, one bidder from each population is drawn independently and uniformly at random to participate in a mechanism. For example, the mechanism might be the first price auction where the highest bidder wins and pays her bid. Notice that a single stage of this iterated population model is equivalent to the standard independent private value model that is pervasive in auction theory. Our model simply is an explicit extension of this standard model to an iterated environment.

1.3. Case Study: Single-Item Auctions

The sale of a single item by winner-pays-bid or all-pay auctions has been studied extensively. When all buyers' values are identically distributed, Chawla and Hartline (2013) show that Bayes-Nash equilibrium of the first-price auction is unique and symmetric. In this equilibrium, the highest-valued buyer wins with probability 1. Hence, equilibrium is efficient. The same holds true for all-pay auctions as well.

When agents' values are asymmetrically distributed, the strong welfare guarantees of Chawla and Hartline (2013) break down. The inefficiency of asymmetric first-price equilibria has long been known (Vickrey, 1961). Syrgkanis and Tardos (2013) show that this inefficiency is bounded: in every BNE of the first-price auction, the social welfare is within a $(1 - 1/e)$-factor of optimal. While this constant-approximation is often framed positively, there are many applications where a potential loss of over $1/3$ of the optimal welfare is unacceptable. While this bound is not known to be tight, there are examples where the first-price auction is as much as a factor of 1.15 away from optimal (Hartline et al., 2014).

1.4. Case Study: Single-Minded Combinatorial Auctions

In more general feasibility environments, there is an even stronger contrast between settings where non-truthful mechanisms perform well, and those where they fail. Consider well-studied single-minded combinatorial auction environment. There are m items for sale, and the bidders from population i desire some publicly-known bundle S_i. A bidder i obtains value v_i if they receive all the items in S_i, and 0 otherwise. A *winner-pays-bid* mechanism chooses an allocation (such as "allocate the feasible set maximizing the sum of bids") and charges winners their bids.

For the general single-minded combinatorial setting, i.e., with multiple items and bidders desire distinct bundles of items, Bayes-Nash equilibrium welfare can be even worse. The *highest-bids-win* winner-pays-bid mechanism allocates to the feasible subset of bids with the highest total. Computational issues aside, the equilibria of the winner-pays-bid mechanism with this rule can have welfare as far as a factor of m away from optimal.

The literature has considered also a greedy winner-pays-bid mechanism, where bidders are allocated greedily based on their bid weighted by $\sqrt{1/|S_i|}$. The equilibrium welfare of this greedy mechanism can be as low as a $\sqrt{m}$-factor of the optimal welfare (Borodin and Lucier, 2010). Both of these mechanisms are unparameterized by distributional knowledge; Dütting and Kesselheim (2015) prove that no unparameterized mechanism can be shown to obtain better than a $\sqrt{m}$ fraction of the optimal welfare via the standard proof method. These lower bounds apply to any algorithm, regardless of computational power. In fact, inefficiencies from non-truthful equilibrium are orthogonal to computational considerations - Dütting and Kesselheim (2015) also show that for the special case where bundles form an interval scheduling problem, $\sqrt{m}$ is still the best guarantee provable for a deterministic mechanism using standard methods. For randomized algorithms, the lower bound weakens to the still super-constant $\log m / \log \log m$. This example highlights that in the absence of symmetry or distributional knowledge, non-truthful mechanisms are limited in their capabilities.

1.5. Case Study: IID Position Auctions

Another generalization of the single-item environment is the *position environment*. Popularized by Varian (2007) and Edelman et al. (2005) to model auctions for online advertisements, position auctions exemplify a tractable setting for non-truthful mechanism design. A position auction environment consists of n slots with *position weights* $w_1 \geq \ldots \geq w_n$. Slots correspond to locations for an advertisment on a webpage. The weight w_i of slot i corresponds to the probability a visitor will click an ad in position i. Such a problem is IID if all agents have the same value distribution.

In these settings, first-price and all-pay auctions have desirable equilibria. In particular, Chawla and Hartline (2013) consider the natural first-price or all-pay position auction, which solicits bids, ranks agents by bid, and assigns agents to slots assortatively. When agents' values distributions are identical, they show that first-price and all-pay auctions have a unique Bayes-Nash equilibrium. Moreover, this equilibrium is the efficient one: the highest-valued agents receive the highest slots.

While the single-minded combinatorial auction environment was highly asymmetrical in what subsets of agents were allowed to be served, the feasibility structure in a position environment is completely symmetrical; an allocation is simply a permutation of agents. Moreover, because bidders have identical value distributions, winner-pays-bid auctions in position environments avoid even the constant-factor loss of the asymmetrically-distributed single-item setting. These comparisons suggest broader design ideas that we will apply in this thesis.

1.6. Case Study: "Un-Revelation" Mechanisms

When the value distributions of the bidders are known, there are winner-pays-bid mechanisms that achieve optimal welfare or revenue in equilibrium, even in highly asymmetrical settings such as the single-minded combinatorial auction. Identifying such a mechanism requires a delicate undoing of the revelation principle, which we outline below. The resulting mechanism is complex and parameterized by details of the value distributions. As such, these "un-revelation" mechanisms are more of a theoretical novelty than a practical suggestion.

We can construct a good first-price or all-pay mechanism for an arbitrary single-dimensional environment such as the single-minded combinatorial auction by inverting the payment function of an existing truthful mechanism. This construction applies to any revelation mechanism $\mathcal{M}$. For concreteness, imagine a applying this approach to a single-minded combinatorial auction problem where $\mathcal{M}$ is the Vickrey-Clarke-Groves (VCG) mechanism. We give the all-pay version of the construction which is slightly simpler, but exhibits the same issues.

Definition 1. The all-pay unrevelation mechanism *for a revelation mechanism $\mathcal{M}$ is:*

(1) *For each agent i and value v_i, calculate $s_i(v_i)$ as the expected payment in $\mathcal{M}$ when the agent's value is v_i and other agents' values are drawn from the distribution.*

(2) *For each agent i, given bid b_i in the un-revelation mechanism, calculate the agent's value as $v_i = s_i^{-1}(b_i)$.*

(3) *Serve the agents who are served by $\mathcal{M}$ on values $\mathbf{v} = (v_1, \ldots, v_n)$; all agents pay their bids.*

The characterization of Bayes-Nash equilibrium (Theorem 2 in Section 2.1) implies that s_i is the strategy that agents will employ in equilibrium of the constructed all-pay mechanism. Thus, the all-pay unrevelation mechanism has the same equilibrium outcome.

From this definition we can see why symmetric and ordinal environments (i.e., IID position environments) are special. For these environments all agents will have the same strategy function, this strategy function will order higher valued bidders higher (by monotonicity), and the ordinal environment then implies that all that is needed to select an outcome is the order of values not their cardinal values. Thus, the mechanism simplifies to simply ordering the bids and the strategy function does not need to be calculated.

Even absent computational issues in estimating the strategy functions so as to implement this mechanism, it is clear that very detailed distributional information is needed to run the unrevelation mechanism. Moreover, the resulting outcomes may be very sensitive to small errors with the inversion of the strategy function. This unrevelation mechanism is not to be considered practical.

1.7. Approach and Results

In this work we identify a family of practical, non-revelation mechanisms for general environments, including single-minded combinatorial auctions, that exhibit three properties. First, these mechanisms use limited distributional information to obtain performance near that of the optimal mechanism, a $1 + \epsilon$ approximation for any desired ϵ. Second, these mechanisms can be easily parameterized and reoptimized from the equilibrium bid data they generate. Mechanisms from such a family can be tuned as fundamentals of the market evolve. Third, the equilibria in these mechanisms have a simple focal equilibrium, and are therefore easy to analyze.

We work in the *iterated population model*, which expands on a standard interpretation of the independent private value model. In this model there are a collection of populations, and each population consists of a continuum of bidders which induces a distribution of values. The strategies of these bidders induce a distribution of bids. In each iteration, one bidder from each population is drawn independently and uniformly at random to participate in a mechanism. For example, the mechanism might be the first price auction where the highest bidder wins and pays her bid. Notice that a single stage of this iterated population model is equivalent to the standard independent private value model that is pervasive in auction theory. Our model simply is an explicit extension of this standard model to an iterated environment.

Our approach is based on linking decisions for bidders from each population across the iterations of the stage environment. This linking of decisions replaces competition between bidders in distinct populations, which is asymmetric, with competition between bidders in the same population, which is symmetric. This linking is achieved by considering the family of mechanisms that determine their outcome only from the rank of a bidder among others from the same population. Such a mechanism cannot know a bidder's value exactly but has a posterior distribution over values, obtained by conditioning on the bidder's rank. The optimal mechanism in this family optimizes as if the bidders' values were equal to the expectation of their respective posterior distributions given their ranks. Our approximation result then shows that little welfare (or revenue) is lost by optimizing with respect to these conditional expected values rather than the exact values. When there are n populations and decisions are linked across T iterations, the loss is bounded by a $O(\sqrt[3]{n/T})$ fraction of the optimal welfare (or revenue).

To prove our linking result, we build up a series of natural approximation results which we believe to be of independent interest. In particular, to understand the performance of mechanisms from data in non-truthful settings, we first consider the problem of learning good truthful mechanisms. This study leads us to a new sample complexity result for general feasibility environments. For regular value distributions, using $\Theta(n^5\epsilon^{-8})$ sampled profiles from the true value distributions, we give a computationally efficient procedure to obtain a mechanism with expected revenue at least a $(1 - \epsilon)$-fraction of optimal. Our results hold for arbitrary feasibility environments. Our procedure estimates a coarse approximation to each agent's revenue curve, and maximizes ironed virtual values according to this estimate.

1.8. Related Work

1.8.1. Nontruthful Mechanisms

Many previous papers have studied the welfare and revenue properties of non-truthful mechanisms, or their *price of anarchy*. Syrgkanis and Tardos (2013) give worst-case welfare approximation bounds for a large family of non-truthful auctions, including first-price and all-pay auctions. Borodin and Lucier (2010) derive similar worst-case results for winner-pays-bid mechanisms based on greedy allocation rules, and Devanur et al. (2015) design non-truthful mechanisms whose equilibria are simple to learn via no-regret algorithms. Moreover, Hartline et al. (2014) extend many of these welfare analyses to the revenue objective when there is sufficient competition or reserve prices. Many of these proofs are via the smoothness framework of Roughgarden (2009). Unfortunately, Dütting and Kesselheim (2015) prove limits on smoothness-based welfare bounds for environments such as the single-minded combinatorial auction. All of these mechanisms are unparameterized by distributional knowledge (except for the reserve price result). We get asymptotically welfare optimal mechanisms – bypassing this and other lower bounds – by allowing the designer to better adapt the stage mechanism based on bids in other stages.

1.8.2. Mechanism Design from Data

The problem of non-truthful mechanism design from data is in many ways analogous to the literature on *sample complexity in mechanism design*. Here, the designer has sample access to bidders' true value distributions, and seeks to design a truthful mechanism with high expected revenue with only these samples from the true distribution. This problem was initiated by Elkind (2007) and Balcan et al. (2008). In so far as this work solves this problem in the more complicated non-truthful setting (as we have samples from the bid distribution not the value distribution), our results have implications on that literature.

More recently, this question was studied for single-item auctions by Cole and Roughgarden (2014) and extended to downward-closed environments by Devanur et al. (2016). The latter paper shows that the sufficient number of samples is $\tilde{O}(n^2\epsilon^{-4})$. Our methods extend this result by relaxing downward closure, but the sample complexity bound worsens to $\tilde{O}(n^5\epsilon^{-8})$. All of these mechanisms are computationally tractable and apply to unbounded but regular distributions (see Section 2.1 for a definition of regularity).

Morgenstern and Roughgarden (2015) have previously considered sample complexity for bounded distributions, applying tools from computational learning theory to auction design. They identify a family of mechanisms which has low representation error (for every distribution, there is a mechanism in the family that is near-optimal) and low generalization error (with high probability, the revenue of mechanisms in the family on a sample is close to their true expected revenue). Consequently, one need only solve the empirical revenue maximization problem of finding the mechanism in the family with they highest revenue on the sample. Unfortunately, identifying an optimal mechanism for the sample can be computationally hard, even for simple environments like single-dimensional matching markets (e.g. Briest (2008)). Gonczarowski and Nisan (2017) subsequently provided a computationally efficient solution to the same problem, assuming the designer is satisfied with low additive rather than multiplicative loss. While we instead consider the unbounded distributions and multiplicative error, the mechanism in our sample complexity result is in the family of mechanisms identified by Morgenstern and Roughgarden (2015).

1.8.3. Linking Decisions

The mechanisms we consider for the iterative population model link decisions for bidders from the same population across distinct iterations. This linking of decisions results in the perceived competition in the mechanism to be among bidders within the same population. Competitions for rank among independent and identically distributed bidders are strategically simple (e.g., Chawla and Hartline, 2013). Linking of decisions has been previously considered in the context of social choice (Jackson and Sonnenschein, 2007) and principal-agent delegation (Frankel, 2014). In both of these papers the decisions being linked are the multi-dimensional reports of a single agent and this linking of decisions enables the principal to obtain the first-best outcome in the limit. Our analysis improves upon these results by bounding the convergence rate to the first-best outcome.

1.8.4. Black-Box Reductions

We provide a *black-box reduction* from non-truthful mechanism design to algorithm design, in the sense that any algorithm that can solve the allocation problem for the stage environment can be converted into a winner-pays-bid or all-pay mechanism with minimal loss of performance. To do so, our mechanism changes the nature of the competition in the iterative population model so that bidders compete across iterations but within each population instead of within an iteration but across populations. This approach has been employed in revelation mechanism design to reduce Bayesian mechanism design to Bayesian algorithm design several times before. Hartline and Lucier (2010) give such a reduction for single-dimensional agents and Hartline et al. (2011), Bei and Huang (2011), and Dughmi et al. (2017) give reductions for single- and multi-dimensional agents. Notably, the reduction of Hartline et al. (2011) for single-dimensional agents gives a mechanism that falls within the family of mechanisms that we develop, albeit with a truthtelling payment scheme. The approximation of welfare and revenue of our mechanism is independent of the payment scheme; therefore, their approximation result implies the same approximation result for our mechanisms. The approximation bound that we derive improves on theirs in that it is multiplicative and does not assume bidders values fall within a bounded range.

1.9. Structure of This Thesis

In Chapter 2, we present several models for single-dimensional mechanism design, and discuss the basic ideas and results that we will use to reason about mechanisms. In Chapter 3, we consider mechanism design and inference in the absence of the strategic issues created by non-truthful payment semantics. We show how to design nearly-optimal truthful mechanisms from samples. The analysis tools employed in this section will serve as a groundwork for the proofs in subsequent chapters. In Chapter 4, we study the iterated population environment, and define the family of mechanisms that is our main object of study. In Chapter 5, we show that with carefully chosen parameters, the mechanisms in our family perform nearly optimally, and that these parameters can be easily inferred. Finally, in Chapter 6, we conclude with open problems.

CHAPTER 2

Model and Preliminaries

This chapter contains the technical description of the mechanism design settings we consider in this thesis, as well as some key results from prior work which will serve as a foundation for our own results. We consider three main models of *single-dimensional Bayesian mechanism design*, each defined by the amount of information and power allowed to the designer. In Section 2.1, we formally describe the *classical Bayesian mechanism design* setting studied by Myerson (1981), in which the seller has full distributional knowledge, and must design a mechanism to maximize revenue or welfare. This setting is well-understood, and we will highlight several results that will prove useful.

In Section 2.1.2, we weaken the assumption of full distributional knowledge to *Bayesian mechanism design from samples*, where the seller only has access to sampled value profiles from agents and must infer a revenue-maximizing mechanism with this more limited information. This setting will serves an intermediate setting as we work towards designing near-optimal winner-pays-bid and all-pay mechanisms from data. In Chapter 3, we will show how to reduce the problem of designing a near-optimal truthful mechanism from samples to the algorithmic problem of computing a welfare-maximizing allocation.

Finally, in Section 2.2, we describe *batched Bayesian mechanism design*. Intuitively, the designer has access to several identical copies of a classical Bayesian environment, and may make decisions jointly across copies. We will design a parametrized family of winner-pays-bid and all-pay mechanisms for this setting which, for the optimal choice of parameters, approach the performance of the optimal truthful mechanism as the number of batched copies grows large.

2.1. Bayesian Mechanism Design

In the classical single-dimensional Bayesian mechanism design problem (which we also refer to as the *independent private values* model), a seller must allocate a service or good to n different agents. Each agent i derives a value for service v_i, which we assume to be drawn from a distribution F_i, with density function f_i (which we assume to exist for simplicity). We assume values are drawn independently across agents.

We consider general single-dimensional feasibility environments, where each agent has an allocation variable x_i which is 1 if the agent receives service and 0 otherwise. We take set of agents who may be feasibly served to be an arbitrary set system $\mathcal{X} \subseteq \{0, 1\}^n$. Some notable special cases include:

- the *single-item environment*: $\mathcal{X}$ consists of all allocation vectors with at most one nonzero entry
- *matroid environments*: $\mathcal{X}$ is the set of characteristic vectors of a matroid
- *single-minded combinatorial environments*: as discussed in Section 1.4, $\mathcal{X}$ is the set of collections of agents i who can be allocated their desired desired bundles S_i of items without overallocating any item.

This work reduces several different mechanism design problems to the *value maximization* problem of selecting a feasible allocation $\mathbf{x} \in \mathcal{X}$ maximizing the allocated value $\sum_i v_i x_i$. We note that in many cases, including the single-minded combinatorial environment, the value maximization problem is NP-complete.

A mechanism takes as input a profile of bids $\mathbf{b} = (b_1, \ldots, b_n)$ and outputs a feasible allocation $\mathbf{x} \in \mathcal{X}$ and agent payments $\mathbf{p}$. A mechanism consists of an *allocation rule* $\tilde{\mathbf{x}}(\mathbf{b})$, which maps bid profiles to a feasible allocation, and a *payment rule* $\tilde{\mathbf{p}}(\mathbf{b})$, which maps bid profiles to a non-negative payment for each agent. A standard allocation rule is *highest-bids-win* which is defined by $\tilde{\mathbf{x}}(\mathbf{b}) \in \operatorname{argmax}_{\mathbf{x}\in\mathcal{X}} \sum_i b_i\, x_i$. That is, the highest-bids-win solves the value maximization problem on agents' reported values. We consider payment rules defined directly from the allocation algorithm via standard payment semantics. The three payment rules we consider here are:

- The *threshold* or *critical bid* payment rule: $\tilde{p}_i(\mathbf{b}) = \min\{b \,|\, \tilde{x}_i(b, \mathbf{b}_{-i}) = 1\} \cdot \tilde{x}_i(\mathbf{b})$.
- The *winner-pays-bid* payment rule: $\tilde{p}_i(\mathbf{b}) = b_i \tilde{x}_i(\mathbf{b})$.
- The *all-pay* payment rule: $\tilde{p}_i(\mathbf{b}) = b_i$.

Given an allocation $\tilde{x}_i(\mathbf{b})$ and payment $\tilde{p}_i(\mathbf{b})$, an agent's utility is given by $\tilde{u}_i(\mathbf{b}) = v_i\tilde{x}_i(\mathbf{b}) - \tilde{p}_i(\mathbf{b})$.

We analyze mechanisms in Bayes-Nash equilibrium (BNE): each agent's report to the mechanism is a best response to the distribution of bids induced by other agents' strategies. The strategy of agent i is denoted s_i and maps the agent's value to a bid. The mechanism $(\tilde{\mathbf{x}}, \tilde{\mathbf{p}})$, the agents' strategies $\mathbf{s}$, and distribution of values $\mathbf{F}$ induce interim allocation and payment rules. Agent i's *interim allocation rule* is $x_i(v_i) = \mathbb{E}_{\mathbf{v}_{-i}}[\tilde{x}_i(\mathbf{s}(\mathbf{v}))]$, *interim payment rule* is $p_i(v_i) = \mathbb{E}_{\mathbf{v}_{-i}}[\tilde{p}_i(\mathbf{s}(\mathbf{v}))]$, and *interim utility* is $u_i(v_i) = v_i x_i(v_i) - p_i(v_i)$. Formally, BNE states that agent i's strategy s_i maximizes $u_i(v_i)$ for all v_i. Of the three payment rules mentioned above, only the threshold payment rule is *dominant strategy truthful* (henceforth *truthful* for short) in the sense that in any mechanism with a monotone allocation algorithm and the threshold payment rule, reporting one's true value is a best response no matter the actions of other agents. Winner-pays-bid mechanisms and all-pay mechanisms tend not to have such an equilibrium, and are therefore *non-truthful*. Myerson (1981) characterized the interim allocation and payment rules that arise in BNE. These results are summarized in the following theorem.

Theorem 2 (Myerson, 1981)**.** *Interim allocation and payment rules are induced by a Bayes-Nash equilibrium of a mechanism with onto strategies and values drawn from a product distribution if and only if for each agent i,*

(1) *(monotonicity) allocation rule $x_i(v_i)$ is monotone non-decreasing in v_i, and*
(2) *(payment identity) payment rule $p_i(v_i)$ satisfies $p_i(v_i) = v_i x_i(v_i) - \int_0^{v_i} x_i(z)dz + p_i(0)$.*

This paper studies the objectives of welfare and revenue. The *welfare* of a mechanism is $\mathbb{E}[\sum_i v_i\, x_i(v_i)]$. As a corollary of the above discussion, the welfare-optimal mechanism uses the highest-bids-win allocation algorithm and the threshold payment rule. The *revenue* of a mechanism is given by $\mathbb{E}[\sum_i p_i(v_i)]$. Our revenue analysis is based on the standard virtual value characterization of Myerson (1981):

Lemma 3. *In BNE, the ex ante expected payment of an agent satisfies $\mathbb{E}_{v_i}[p_i(v_i)] = E_{v_i}[\phi_i(v_i)\, x_i(v_i)]$, where $\phi_i(v_i) = v_i - \frac{1-F(v_i)}{f(v_i)}$ is the* Myerson virtual value *for value v_i.*

It follows from Lemma 3 that the equilibrium revenue of a mechanism is given by $\sum_i \mathbb{E}_{\mathbf{v}}[p_i(v_i)] = \sum_i \mathbb{E}_{\mathbf{v}}[\phi_i(v_i)\, x_i(v_i)]$. *Regular* distributions are those for which the virtual value functions are monotone non-decreasing. For regular distributions, the optimal mechanism allocates the virtual value-maximizing feasible set.

An equivalent formulation of Lemma 3 can be stated in terms of agents *quantiles* and *revenue curves*, two concepts that will prove useful in the revenue analysis of our mechanisms. An agent's quantile $q_i(v_i)$ is that agent's strength in their distribution, given by $q_i(v_i) = 1 - F_i(v_i)$ (with inverse *value function* $v_i(q_i) = F_i^{-1}(1 - q_i)$). The revenue from posting a price at quantile q_i is $q_i v_i(q_i)$, and from this, we can define the price-posting revenue curve $R_i(q_i) = q_i v_i(q_i)$. An agent's Myerson virtual value $\phi_i(v_i)$ can be expressed as $\phi_i(q_i) = R_i'(q_i)$. The revenue curve formulation also allows us to derive the revenue-optimal mechanism for agents whose distributions do not satisfy regularity. To do so, we define the *ironed revenue curve* $\overline{R}_i(q_i)$ to be the smallest concave function that upperbounds $R_i(q_i)$, and define the *ironed virtual values* to be $\overline{\phi}_i(q_i) = \overline{R}_i'(q_i)$. The revenue-optimal mechanism for general distributions maximizes ironed virtual surplus, subject to the constraint that tiebreaking in the mechanism depend only on agents' virtual value and index, and not on their value. This can be achieved, for example, by breaking ties lexicographically.

2.1.1. Position Environments

To design mechanisms for general single-dimensional environments, where winner-pays-bid and all-pay mechanisms are generally badly behaved, we will draw intuition from a setting where such mechanisms possess many desirable properties: *i.i.d. position environments*. An n-agent position environment is given by n *positions*. Each position i has a corresponding *position weight* $w_i \in [0, 1]$, with $w_1 \geq \ldots \geq w_n$. Position environments arise in models for advertising auctions, where each position represented a potential position for an ad on a webpage, and the position weight represents the probability that an ad in position i will be clicked. A feasible allocation in a position environment is an assignment of agents to positions, and an agent's allocation x_i is equal to the weight of the position to which they have been assigned.

In position environments highest-bidders-win allocation rule requires only the rank information of bidders, as it matches agents to positions assortatively. Chawla and Hartline (2013) showed that when agents' values are i.i.d., the winner-pays-bid and all-pay implementations of the highest-bidders-win rule has a single, straightforward equilibrium:

Theorem 4 (Chawla and Hartline, 2013)**.** *In i.i.d. position environments, the rank-by-bid winner-pays-bid and all-pay auctions have a unique and welfare-maximizing BNE (in which agents are assigned to positions in order of their true values).*

2.1.2. Bayesian Mechanism Design from Samples

One common criticism of the classical model for Bayesian mechanism design is that it is often unrealistic to assume that the mechanism designer has full knowledge of agents' value distributions. The *Bayesian mechanism design from samples* model, formalized by Cole and Roughgarden (2014), seeks to address this criticism by modeling the designer's acquisition of distributional information more explicitly. Rather than possessing knowledge of each agent i's prior F_i, the seller has access to a sample oracle, which can be queried for a freshly sampled value profile $\mathbf{v}^j \sim \mathbf{F}$. Using polynomially many samples, the seller's goal is to produce a mechanism which obtains at least a $(1-\epsilon)$-fraction of the expected revenue of the optimal mechanism which has full knowledge of the true distributions $\mathbf{F}$. Formally, the seller's goal is to give a procedure which takes sampled value profiles $\mathbf{v}^1, \ldots, \mathbf{v}^m$ for some m polynomial in the number of bidders n and the inverse of the target loss, $1/\epsilon$, and produce a truthful mechanism $\text{ALG}(\mathbf{v}^1, \ldots, \mathbf{v}^m)$ to be run on a fresh sample $\mathbf{v}^0$ such that:

$$\mathbb{E}_{\mathbf{v}^1,\ldots,\mathbf{v}^m \sim \mathbf{F}}[\text{Rev}(\text{ALG}(\mathbf{v}^1, \ldots, \mathbf{v}^m))] \geq (1-\epsilon)\text{Rev}(\text{OPT}_{\mathbf{F}}),$$

where for any mechanism M, $\text{Rev}(M)$ denotes the expected revenue of M with respect to the fresh sample $\mathbf{v}^0 \sim \mathbf{F}$. Additionally, we will require that the algorithm that produces the mechanism from samples run in polynomial time.

This problem serves as a useful stepping stone, as we develop winner-pays-bid and all-pay mechanisms which use bid data to obtain nearly-optimal welfare and revenue. In Chapter 3, we provide a solution to this problem which applies in any single-dimensional feasibility environment (as opposed to the single-item results of Cole and Roughgarden (2014) or the downward-closed results of Devanur et al. (2016)). Subsequent chapters can be thought of as an implementation of our scheme with winner-pays-bid or all-pay payment semantics.

2.2. The Population Model

To design near-optimal winner-pays-bid and all-pay mechanisms, we will consider the *iterated population model*. The iterated population model is a special case of single-dimensional Bayesian mechanism design, in which each distribution is comprised of a continuum of individuals, and each iteration of a mechanism draws new individuals from each population.

We consider a *batched environment* which has T independent copies of the stage environment. The stage environment is an arbitrary single-dimensional mechanism design environment, with n buyers, a set $\mathcal{X}$ of feasible allocations, and value distributions $F_1, \ldots, F_n$. The batched environment is consequently an independent private value model with nT agents. Importantly, the T agents from population i are independent and identically distributed (i.i.d.) according to F_i. The set of feasible allocations for the batched environment is $\mathcal{X}^T$. Values, allocations, payments, and utilities in the batched environment are denoted as v_i^t, x_i^t, p_i^t, and u_i^t for agent i in stage t.

We will also consider an online version of the population model, in which the designer has access to stages one at a time. The mechanism designer may use bid data from the past $T-1$ stages in designing a mechanism for the present stage, but they must commit to allocations and payments in each stage separately. We will show in Chapter 4 how reasoning about the online model may be reduced to analysis of the offline model.

CHAPTER 3

Sample Complexity

Recall the main goal of this thesis: to design a parametrized family of mechanisms with winner-pays-bid and all-pay semantics such that

(1) The revenue or welfare of the optimal mechanism from the family is close to the performance of the optimal mechanism from the family.

(2) The optimal choice of parameters can be inferred from past bid data generated by mechanisms in the family.

Both of these goals are made particularly challenging by the restriction to non-truthful payment semantics. Objective (1) is made difficult by the fact that the performance of non-truthful mechanisms must be analyzed in equilibrium, which need not generally trade off allocating bidders from different distributions optimally, or even in a straightforward way. Objective (2) is made difficult by the fact that inference must be done from bid data, which are generally not bidders' true values. In this chapter, we study the problem of revenue maximization from sampled values.[1] This setting allows us to isolate some of the design challenges related to inference, and solve them in a setting without nontruthful equilibria complicating the analysis. The approach of the later chapters will reduce the problem of nontruthful mechanism design from data to this easier truthful setting.

[1]Welfare maximization can be solved without any distributional access. Nonetheless, our analysis in this section will generally apply to the objective of welfare as well, which will allow us to design non-truthful mechanisms with near-optimal welfare, which is nontrivial.

Our mechanism from samples estimates each agent's revenue curve, and maximizes revenue with respect to the estimated revenue curves. In other words, it computes ironed virtual values as if the distributions had the estimated revenue curves and maximizes ironed virtual surplus. To estimate the revenue curve, the mechanism estimates the quantile of a small number of values, and linearly interpolates between the revenue at these points to construct a full revenue curve.

The proof that the mechanism's revenue is near-optimal consists of three main steps. First, we show that for any regular distribution, maximizing revenue with respect to a piecewise linear approximation of the true revenue curve is nearly as good as maximizing revenue with respect ot the true revenue curve. While such a result might seem intuitively obvious, note that even for regular distributions, the true revenue curve and a piecewise-linear approximation (obtained by interpolating between $R(q)$ for $q \in \{1/T, 2/T, \ldots, T/T\}$ for some positive integer T) need not be close in either a multiplicative or additive sense. The approach we use to prove this theorem involves composing several natural revenue approximation results which may be of independent interest. Given that a piecewise-linear approximation to the true revenue curve suffices to approximately maximize revenue, the remainder of the analysis consists of showing that such a piecewise-linear revenue curve is easy to estimate from samples, and that estimation error propogates cleanly to revenue loss.

The structure of the chapter is as follows. In Section 3.1, we prove the main technical insight of this section: that piecewise linear revenue curves suffice for obtaining a close approximation to the optimal revenue. In Section 3.2, we formally discuss the sample complexity implications of this observation, and give a mechanism from samples that obtains almost-optimal revenue.

3.1. Revenue Maximization with Coarse Revenue Curves

In this section, we show that if a mechanism designer has access to a piecewise-linear approximation to each agent's revenue curve, rather than the true distributions, then maximizing revenue with respect to these approximations is nearly revenue-optimal.

More formally, for each agent i, let R_i denote agent i's price-posting revenue curve. Furthermore, for some positive integer T define the piecewise-constant approximation $R_i^{\{j/T\}_{j=1,\ldots,T}}(q)$ by linearly interpolating between the points $(j/T, R(j/T))$ for $j = 1, \ldots, n$. We will prove the following theorem:

Theorem 5. *Assume agents have regular value distributions. Then for any positive integer T, the mechanism which maximizes revenue as if each agent i had revenue curve $R_i^{\{j/T\}_{j=1,\ldots,T}}$ obtains a TODO: REVENUE FRACTION of the optimal revenue.*

This result seems intuitively obvious. As T grows large, $R_i^{\{j/T\}_{j=1,\ldots,T}}$ comes to look more and more like the true revenue curve, and consequently, the designer's imagined distribution should grow close to the true distribution. Formally, however, the result is less clear. Note that even for regular distributions, there is no way to bound either the multiplicative or additive distance of $R_i^{\{j/T\}_{j=1,\ldots,T}}(q)$ from $R_i(q)$ uniformly for all $q \in [0, 1]$. This is best illustrated by considering the revenue curve of the equal-revenue distribution: $R_i(q) = 1$ for $q \in (0, 1)$ and 0 for $q \in \{0, 1\}$. For extremely low and extremely high or low quantiles, the $R_i(q) = 1$, while $R_i^{\{j/T\}_{j=1,\ldots,T}}(q)$ can be arbitrarily small.

Rather than argue in the space of revenue curves, we prove Theorem 5 indirectly. Note that for any agent i, the revenue-optimal mechanism under the piecewise-linear approximations $R_i^{\{j/T\}_{j=1,\ldots,T}}(q)$ will treat each quantile $q_i \in [j/t, (j+1)/T]$ identically. In other words, this mechanism treats types coarsely based on their "bin" in quantile space. In fact, maximizing revenue according to $R_i^{\{j/T\}_{j=1,\ldots,T}}(q)$ is the revenue-optimal way to treat agents in each bin identically, and will inherit the revenue properties of every other mechanism in the class of mechanisms that satisfy this property. We therefore proceed by constructing a different bin-based mechanism, and proving a performance guarantee for this new mechanism.

To construct an approximately-optimal bin-based mechanism, we begin with the revenue-optimal mechanism for the true distributions, and transform it into a bin-based mechanism by resampling (as in Hartline and Lucier, 2010). When a agent i reports a quantile $q_i \in [j/T, (j+1)/T]$, the mechanism instead resamples a quantile uniformly from $[j/T, (j+1)/T]$ and treats this sample as their report in the revenue-optimal mechanism for their true distributions. Note that for some regular distributions (e.g. the equal-revenue distribution mentioned above), any mechanism which attains high revenue must treat agents with extreme quantiles delicately, making sure to allocate very high quantiles and reject very low quantiles. Resampling reduces a mechanism's ability to treat extremal quantiles in this way. We show how to modify the resampling procedure to treat extreme quantiles carefully, which yields our approximately-optimal bin-based mechanism.

In Section 3.1.1, we present a more general statement of Theorem 5, which will imply Theorem 5 as well as laying the technical groundwork for the samples-based approach of Section 3.2. In Sections 3.1.2 and 3.1.3, we prove this more general theorem using the indirect approach described above. Section 3.1.2 formalizes the idea that optimizing revenue based on a piecewise-linear approximation to the true revenue curve is revenue-optimal in the family of mechanisms which treats agents based a coarsening of their distribution into "bins." In Section 3.1.3, we design a bin-based resampling mechanism with nearly-optimal revenue.

3.1.1. Technical Preliminaries

In this section, we formally state a more general version of Theorem 5, which we will prove subsequently. Theorem 5 states that knowing only $R_i(j/T)$ for some positive integer T, for all $i \in \{1, \dots, n\}$, and all $j \in \{1, \dots, T\}$, a mechanism designer can attain nearly optimal revenue. To maximize revenue from samples, one could then hope to estimate each value of $R_i(j/T)$. Because it is easier to bound the error from estimating the quantile of a given value than it is to accurately estimate the value at a given quantile, we relax the set of quantiles for which the designer knows the revenues from j/T for all $j \in \{0, \dots, T\}$ to some general set of quantiles $\hat{q}_i^j$ for $j \in \{0, \dots, T\}$, with $0 = \hat{q}_i^0 \leq \hat{q}_i^1 \leq \dots \leq \hat{q}_i^T = 1$, which we will refer to as "breakpoints." Our mechanism from samples in Section 3.2 will use values of $\hat{q}_i^J$ which are close to j/T. Throughout, we will take T to be given in advance.

As discussed, we will consider the mechanism which treats agents as if their revenue curves were a linear interpolation between $(\hat{q}_i^j, R_i(\hat{q}_i^j))$ for all j. We formally define these curves as follows:

Definition 6. *Let $\hat{Q}_i$ be a set of breakpoints for a single agent i such that $\hat{q}^0 = 0 \leq \hat{q}^1 \leq \ldots \leq \hat{q}^T = 1$. The* binned revenue curve *for $\hat{Q}$ is given by*

$$R_i^{\hat{Q}_i}(q) = \frac{q-\hat{q}_i^j}{\hat{q}_i^{j+1}-\hat{q}_i^j} R_i(\hat{q}_i^{j+1}) + \frac{\hat{q}_i^{j+1}-q}{\hat{q}_i^{j+1}-\hat{q}_i^j} R_i(\hat{q}_i^j) \text{ for } q \in [\hat{q}_i^j, \hat{q}_i^{j+1}].$$

With Definition 6 in hand, we may state our the generalization of Theorem 5:

Theorem 7. *Let $\hat{Q} = \{\hat{q}_i^j\}_{i=1,\ldots,n}^{j=1,\ldots,T}$ denote a set of breakpoints for each player. Let $\hat{q}_i^j \in [j/T - \epsilon, j/T + \epsilon]$ for all agents i and $j \in \{1, \ldots, T-1\}$, and further let $\epsilon \leq \min((nT)^{-1/2}, T^{-1})$. Then the revenue-maximizing mechanism for revenue curves $\hat{R}_i^{\hat{Q}_i}$ obtains at least a $(1 - O(\sqrt{n/T}))$-fraction of the optimal virtual surplus.*

3.1.2. Optimal Binning Mechanisms

In this section, we begin developing the necessary tools to prove Theorem 7. For any agent i, the revenue-optimal mechanism for the linear approximations $\hat{R}_i^{\hat{Q}_i}$ treats each quantile $q_i \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$ identically. In other words, it divides each agent's quantile space into "bins," given by $[\hat{q}_i^j, \hat{q}_i^{j+1}]$, and treats them conditioned on their bin. We show in what follows that this mechanism is revenue-optimal among all mechanisms which treat agents based only on the label of their bin. Consequently, this mechanism will inherit any revenue guarantees we derive for other mechanisms in this family. There remainder of the proof of Theorem 7 will be the construction of another bin-based mechanism which is approximately revenue-optimal. Formally stated, we have the following:

Theorem 8. *Let $\hat{Q}$ be a set of T breakpoints for each agent. The mechanism which maximizes revenue as if each agent i had revenue curve $\hat{R}_i^{\hat{Q}_i}$ is revenue-optimal among all mechanisms which treat each agent based only on the bin into which their quantile falls.*

Proof. Given a Bayesian mechanism design environment with distributions $F_1, \ldots, F_n$, define the *bin-based algorithm design problem for bins given by* $\hat{Q}$ as follows: the designer must choose a truthful mechanism with allocation rule $\overline{\mathbf{x}}$ which takes as inputs the indices $\mathbf{k}(\mathbf{v}) = (k_1(v_1), \ldots, k_n(v_n))$ of the bins into which each agent's quantile falls and outputs a feasible allocation. In other words, $k_i(v_i)$ is defined to be the index j such that $q_i(v_i) \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$. As a constraint, $\overline{\mathbf{x}}$ must be monotone decreasing in k_i for each agent. The objective is to maximize $\mathbb{E}_{\mathbf{v}\sim\mathbf{F}}[\sum_i \phi_i(v_i)\overline{x}_i(\mathbf{k}(\mathbf{v}))]$, where $\phi_i(\cdot)$ is the Myerson virtual value function.

The above optimization problem can be solved by inspection. Fixing an allocation algorithm, the objective can be rewritten as $\mathbb{E}[\sum_i \mathbb{E}[\phi_i(v_i) \,|\, k_i]\overline{x}_i(\mathbf{k}(\mathbf{v}))]$ by linearity of expectation. From this expression, it becomes clear that the optimal mechanism maximizes the quantity $\sum_i \mathbb{E}[\phi_i(v_i) \,|\, k_i]\overline{x}_i(\mathbf{k}(\mathbf{v}))$ pointwise. These conditional virtual surpluses are exactly the slopes of the binned revenue curves, which are the virtual values maximized by the optimal mechanism for the binned revenue curves. Note that if $\phi_i(\cdot)$ is monotone (which it will be for regular distributions), then this algorithm will be monotone as well, and therefore feasible. Hence, maximizing revenue for the binned revenue curves is the revenue-optimal mechanism which conditions an agent's treatment solely on their bin. □

3.1.3. Binning Via Resampling

We have shown that a mechanism designer who is constrained to treating agents based on a coarsening of their quantile space into "bins" should maximize virtual surplus conditioned on the bin of each agent. This is exactly the mechanism of Theorem 7. To prove Theorem 7, then, we need only construct *some* mechanism which for all bins, treats all types in that bin identically. We accomplish this task in what follows.

In addition to proving Theorem 7 and therefore enabling us to prove a sample complexity result, the tools developed in this section will also be helpful for deriving mechanisms with nontruthful payment semantics for the iterated population environment in Chapters 4 and 5. As discussed in Chapter 1, designing nontruthful mechanisms for the objectives of welfare and revenue are both challenging. This stands in contrast to truthful mechanism design from samples, where maximizing welfare is trivial - the prior-independent VCG mechanism is always welfare optimal.

Because we will apply the techniques in this section to welfare in the subsequent chapters, we will will consider not only bin-based mechanisms for revenue but also bin-based mechanisms for welfare. To accommodate multiple objectives, we will instead consider the maximization of an abstract virtual surplus quantity ϕ_i for each agent. Taking ϕ_i to be the Myerson virtual value of Lemma 3 will yield a revenue result. Taking $\phi_i(q_i) = v_i(q_i)$ will instead produce a welfare result.

3.1.3.1. Intuition. To prove Theorem 7, we will show that any monotone allocation rule can be converted into a bin-based rule without losing much virtual welfare. To do so, we take each agent i with value $v_i \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$ and *resample* their value from F_i conditioned to that interval, and then apply the original allocation rule with respect to the resampled values. Such a resampling does not change the induced allocation rule for any other agents, and replaces the allocation rule on $[\hat{q}_i^j, \hat{q}_i^{j+1}]$ with its average.

This basic approach does not directly lead to the desired approximation bound because, for the highest-valued interval (i.e. the lowest quantiles), the allocation probability at the top of the interval may be much higher than its average allocation probability, and higher values on the interval may be much higher than the interval's average. For example, if the value and allocation rule are both one for an ϵ measure and zero otherwise, then the original welfare is ϵ and the welfare from resampling is ϵ^2. A similar problem holds for extremely high quantiles, whose virtual value might be extremely negative.

To resolve this issue, we will first modify the allocation rule to treat agents with values in the top k intervals as if they had the highest value in the support of their distributions, and treat agents with values in the bottom k intervals as if they had the lowest value in the support of their distributions, for some given positive integer k. The quantiles of the remaining agents will be rescaled. Conditioned on the values not being in the top or bottom k intervals, the value distribution after rescaling will match the original unconditioned value distribution. We refer to this transformation as *extremal buffering*. Unlike the basic resampling approach, applying this method to one agent does change the mechanism for other agents. We show that this change does not have a significant impact on the outcomes other agents receive, and approximately preserves welfare and revenue from each population. Our analysis will hold for any choice of k. Theorem 7 will follow from choosing k wisely.

We analyze the top promotion procedure in Section 3.1.3.2, and the binning algorithm that results from resampling in Section 3.1.3.3.

3.1.3.2. Extremal Buffering. We have seen that naïvely resampling each agent's value based on their bin cannot yield a mechanism with a good welfare or revenue approximation. We show now how to transform an arbitrary allocation algorithm to guarantee that not too much virtual surplus is lost from mishandling agents with extreme quantiles. The procedure follows:

Definition 9. *Given a monotone allocation algorithm* $\mathbf{x}$*, breakpoints* $\hat{Q}$*, and an integer* k*, the* k-bin extremal buffering algorithm *for* $\mathbf{x}$ *and* $\hat{Q}$ *runs* $\mathbf{x}$ *on agents with quantiles transformed for each agent as follows:*

- *For any* $q_i \in [0, \hat{q}_i^k]$*, return 0.*
- *For any* $q_i \in [\hat{q}_i^k, \hat{q}_i^{T-k}]$*, return* $(q_i - \hat{q}_i^k)/(\hat{q}_i^{T-k} - \hat{q}_i^k)$*.*
- *For any* $q_i \in [\hat{q}_i^{T-k}, 1]$*, return* 1*.*

We will prove the following approximation guarantee:

Lemma 10. *Let* $\hat{q}^k = \max_i \hat{q}_i^k$ *and* $\hat{q}^{T-k} = \min_i \hat{q}_i^{T-k}$*, and for each agent* i*, let* $\phi_i : [0, 1] \to \mathbb{R}$ *be an arbitrary nonincreasing virtual value function satisfying* $\int_0^1 \phi_i(q)\, dq \geq 0$*. The* k*-bin extremal buffering algorithm for* $\hat{\mathbf{x}}$ *and* $\hat{Q}$ *attains at least a* $(1-\hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k}(1-(n-1)(\hat{q}^k + (1 - \hat{q}^{T-k})))$*-fraction of the virtual surplus of* $\mathbf{x}$*.*

The proof of Lemma 10 will proceed in two main steps. First, we will show that applying the quantile remapping procedure in Definition 9 to a single agent i (leaving other agents' quantiles untouched) cannot reduce the virtual surplus from that agent by too much. This will follow from a natural approximation result we derive, which relates the virtual surpluses of allocation rules with inverses that are multiplicatively close.

Second, we will show that subsequently applying the quantile resampling procedure to the remaining agents other than i also does not significantly reduce the expected virtual surplus from i. This will follow from the fact that the distribution of quantiles input to the base allocation algorithm is identical, conditioned on no agents having extreme quantiles.

We begin with the single-agent analysis. Note that for a single agent, the extremal buffering procedure can be thought of as two composed steps. First is a *top promotion* procedure, which remaps sufficiently low quantiles to 0 while remapping the remaining quantiles to induce a uniform distribution over $[0, 1]$. Top promotion is then composed with *bottom demotion*, which performs analogous transformation, mapping high quantiles to 1 and mapping the rest of the interval to $[0, 1]$. We formalize this as follows:

Definition 11. *Given a monotone single-agent allocation rule x and quantile $\underline{q}$, the* top promotion algorithm *for x and $\underline{q}$ runs x on the agent with quantiles transformed as follows:*

- *For any $q \in [0, \underline{q}]$, return* 0.
- *For any $q \in [\underline{q}, 1]$, return $(q - \underline{q})/(1 - \underline{q})$.*

Definition 12. *Given a monotone single-agent allocation rule x and quantile $\overline{q}$, the* bottom demotion algorithm *for x and $\overline{q}$ runs x on the agent with quantiles transformed as follows:*

- *For any $q \in [0, \overline{q}]$, return $q/\overline{q}$.*
- *For any $q \in [\overline{q}, 1]$, return* 1.

The interim allocation rule faced by an agent i after applying the extremal buffering algorithm to just i is the composition of the bottom demotion algorithm for quantile $\hat{q}_i^{T-k}$ composed with the top promotion algorithm for original allocation rule x_i and quantile $\hat{q}_i^k/\hat{q}_i^{T-k}$. Consequently, we may analyze the loss from applying these two transformations separately and multiply the losses.

We first analyze bottom demotion. While bottom demotion does not produce an allocation rule which is multiplicatively close to the original rule, it does produce one which is close in a different sense: its inverse is close to the inverse of the original rule. For the objectives of both revenue and welfare, this notion of closeness also produces a multiplicative approximation for virtual surplus. We state this as a separate technical lemma, as we will make use of the idea multiple times.

Lemma 13. *For virtual value function $\phi(\cdot)$ and cumulative virtual value $R(q) = \int_0^q \phi(r)\,dr$ satisfying $R(\alpha\, q) \geq \alpha\, R(q)$ for all quantiles q and $\alpha \in [0,1]$, and any two allocation rules $\tilde{x}$ and $\hat{x}$ that satisfy $\tilde{x}^{-1}(z) \geq \hat{x}^{-1}(z) \geq \frac{1}{\alpha}\tilde{x}^{-1}(z)$, the virtual surpluses satisfy*

$$E_{q\sim U[0,1]}[\phi(q)\hat{x}(q)] \geq \tfrac{1}{\alpha}E_{q\sim U[0,1]}[\phi(q)\tilde{x}(q)].$$

Proof. The virtual surplus can be rewritten as $\int_0^1 \phi(q)x(q)\,dq = \int_0^1 R(x^{-1}(z))dz$. This follows from an integration by parts and then change of variables to integrate the vertical axis rather than the horizontal axis as follows:

$$\int_0^1 \phi(q)x(q)\,dq = R(1)x(1) - R(0)x(0) + \int_0^1 R(q)\,(-x'(q))\,dq \tag{3.1}$$
$$= \int_0^{x(1)} R(1)\,dz - 0 + \int_{x(1)}^1 R(x^{-1}(z))\,dz.$$

Now consider two arbitrary quantiles q_1 and q_2 satisfying $\frac{1}{\alpha}q_1 \leq q_2 \leq q_1$. By assumption, we have $R(q_2) \geq q_2 R(q_1)/q_1 \geq \frac{1}{\alpha}R(q_1)$. The assumption on the approximation of the two allocation rules, namely $\tilde{x}^{-1}(z) \geq \hat{x}^{-1}(z) \geq \frac{1}{\alpha}\tilde{x}^{-1}(z)$ for all $z \in [0,1]$, and the expected virtual surplus written as $\int_0^1 R(x^{-1}(z))dz$ suffice to prove the lemma. □

Lemma 14. *Let $\phi : [0,1] \to \mathbb{R}$ be an arbitrary nonincreasing virtual value function. Given a monotone single-agent allocation rule x and quantile $\overline{q}$, the bottom demotion algorithm for x and $\overline{q}$ obtains at least a $\overline{q}$-fraction of the expected virtual surplus of x.*

Proof. The lemma will follow from a straightforward application of Lemma 13. For a quantile q receiving allocation $x(q)$ from the base algorithm, the quantile receiving this probability of allocation under the bottom demotion algorithm will be $\overline{q}q$. Hence, $x^{-1}(z) \geq \hat{x}^{-1}(z) = \overline{q}x^{-1}(z)$. Since ϕ is nonincreasing in q, we have that $R(q) = \int_0^q \phi(r)\,dr$ satisfies $R(\alpha q) \geq \alpha R(q)$ for all $\alpha \in [0,1]$. Hence, Lemma 13 implies the desired result. □

We have shown that bottom demotion results in an allocation rule which has an inverse close to that of the original rule on which it is based. To derive an approximation result for the top promotion algorithm requires a more nuanced version of the same approach, based on two observations. First, the "unallocation rules", i.e., $y(q) = 1 - x(1-q)$ for allocation rule $x(q)$, satisfy the inverse-approximation condition of the lemma. Second, the virtual surplus of the unallocation rule is given by the expected virtual value plus the negative virtual surplus of the unallocation rule. Specifically $\mathbb{E}_q[\phi(q)\,x(q)] = \mathbb{E}_q[\phi(q)] + \mathbb{E}_q[(-\phi(1-q))\,y(q)]$. While virtual values for revenue always satisfy the property that rays from the origin cross the cumulative virtual value curve from below, this property does not generally hold for the negative virtual values $-\phi(1-q)$. Regularity, i.e., monotonicity of the original virtual value function, however, implies the property for negative virtual values. These observations are formally summarized in the subsequent lemma:

Lemma 15. *Let $\phi : [0,1] \to \mathbb{R}$ be an arbitrary nonincreasing virtual value function satisfying $\int_0^1 \phi(q)\,dq \geq 0$. Given a monotone single-agent allocation rule x and quantile $\underline{q}$, the top promotion algorithm for x and $\underline{q}$ obtains at least a $(1-\underline{q})$-fraction of the expected virtual surplus of x.*

Proof. Note that the expected virtual surplus from any allocation rule x is can be written as $\int_0^1 \phi(q)x(q)\,dq = \int_0^1 \phi(q)\,dq - \int_0^1 \phi(q)(1-x(q))\,dq$. Specifically, let $\hat{x}$ be the allocation rule of the top promotion algorithm, and x the allocation rule of the original algorithm. Moreover, define $\hat{y}(q) = 1 - \hat{x}(1-q)$ and $y(q) = 1 - x(1-q)$ to be the corresponding "unallocation rules." We will show that

$$\int_0^1 -\phi(1-q)\hat{y}(q)\,dq \geq (1-\underline{q}) \int_0^1 -\phi(1-q)y(q)\,dq. \tag{3.2}$$

Since, $\int_0^1 \phi(q)\,dq \geq 0$, this will prove that $\int_0^1 \phi(q)x(q)\,dq \geq (1-\underline{q}) \int_0^1 \phi(q)\hat{x}(q)\,dq$.

To prove (3.2), note that the definition of the top promotion algorithm can be manipulated to obtain $\hat{x}^{-1}(z) = x^{-1}(z)(1-\underline{q}) + \underline{q}$. Moreover, by the definition of y and $\hat{y}$, we have $y^{-1} = 1 - x^{-1}(1-z)$ and $\hat{y}^{-1} = 1 - \hat{x}^{-1}(1-z)$. Combining these three equations yields that $y^{-1}(z) \geq \hat{y}^{-1}(z) = (1-\underline{q})y^{-1}(z)$ for all $z \in [0,1]$. Moreover, note that $-\phi(1-q)$ is decreasing in q. This implies that $R(q)/q \geq -\phi(1-q)$, where $R(q) = \int_0^q -\phi(1-q)\,dq$. We may therefore apply Lemma 13, which yields (3.2). □

Combining Lemmas 14 and 15 yields the following lemma:

Lemma 16. *Let $\phi : [0,1] \to \mathbb{R}$ be an arbitrary nonincreasing virtual value function satisfying $\int_0^1 \phi(q)\,dq \geq 0$, and consider an arbitrary agent i. The k-bin extremal buffering algorithm for $\mathbf{x}$ and $\hat{Q}$, when applied only to agent i, attains at least a $(1-\hat{q}_i^k\hat{q}_i^{T-k})\hat{q}_i^{T-k}$-fraction of the expected virtual surplus for i.*

Having derived a single-agent guarantee, we now show that applying the extremal buffering algorithm to all agents at once, rather than just to one agent, yields only a small additional loss. Intuitively, for each agent, the mechanism only appears different when another agent has an extreme quantile which is promoted or demoted by the buffering algorithm. The probability of such an event can be bounded using the union bound. Formally, we have:

Proof of Lemma 10. Lemma 16 states that the virtual surplus lost from applying the extremal buffering to a single agent is small. We now argue that applying the algorithm to all agents at once does not incur much additional loss. We argue from the perspective of an arbitrary agent i.

The key observation in our analysis is that the distribution of the quantiles of other agents is nearly unchanged by the extremal buffering algorithm. In particular, note that the probability that one or more agents other than i with quantiles set to 0 or 1 by the extremal buffering algorithm is at most $(n-1)(\hat{q}^k + (1-\hat{q}^{T-k})$, by the union bound.

Conditioned on there being no such agents, the distribution of quantiles input to the allocation algorithm remains uniform. It follows that the virtual surplus from distribution i conditioned on this event is identical to the revenue from the extremal buffering algorithm applied only to i.

In the event that there are one or more agents from populations other than i who have top quantiles (which are promoted) or bottom quantiles (which are demoted), we note that the conditional virtual surplus from population i is nonnegative. To see this, let $\tilde{x}_i$ be the interim allocation rule for agent i in the extremal buffering algorithm conditioned on the event $\mathcal{E}$ that at least one agent j other than i has a quantile in $[0, \hat{q}_j^k] \cup [\hat{q}_j^{T-k}, 1]$. Since $\mathbf{x}$ is a monotone function of its inputs, it must be that $\tilde{x}_i$ is nondecreasing. The expected virtual surplus from agent i conditioned on $\mathcal{E}$ is $\int_0^1 \phi_i(q)\tilde{x}_i(q)\,dq$. By assumption, $\int_0^1 \phi_i(q)\,dq \geq 0$, so it must also be the case that $\int_0^1 \phi_i(q)\tilde{x}_i(q)\,dq \geq 0$.

To conclude the proof, let $\hat{x}_i$ denote the interim allocation rule of the extremal buffering algorithm conditioned on the event $\overline{\mathcal{E}}$. The total virtual surplus from agent i is:

$$\Pr(\mathcal{E}) \int_0^1 \phi_i(q)\tilde{x}_i(q)\,dq + \Pr(\overline{\mathcal{E}}) \int_0^1 \phi_i(q)\hat{x}_i(q)\,dq$$

By the union bound, $\Pr(\overline{\mathcal{E}}) = 1 - \Pr(\mathcal{E}) \geq 1 - (n-1)(\hat{q}^k + \hat{q}^{T-k})$. By Lemma 16 and the fact that, conditioned on $\overline{\mathcal{E}}$, the distribution of quantiles i perceives from other agents is uniform implies that

$$\begin{aligned} \int_0^1 \phi_i(q)\hat{x}_i(q)\,dq &\geq (1 - \hat{q}_i^k/\hat{q}_i^{T-k})\hat{q}_i^{T-k} \textstyle\int_0^1 \phi_i(q)x_i(q)\,dq \\ &\geq (1 - \hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k} \textstyle\int_0^1 \phi_i(q)x_i(q)\,dq. \end{aligned}$$

Since we have shown that $\int_0^1 \phi_i(q)\hat{x}_i(q)\,dq \geq 0$, we can combine the above to obtain:

$$\begin{aligned}&\Pr(\mathcal{E})\int_0^1 \phi_i(q)\tilde{x}_i(q)\,dq + \Pr(\overline{\mathcal{E}})\int_0^1 \phi_i(q)\hat{x}_i(q)\,dq\\ \geq\ &(1-\hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k}(1-(n-1)(\hat{q}^k+(1-\hat{q}^{T-k})))\int_0^1 \phi_i(q)x_i(q)\,dq.\end{aligned}$$

Summing over agents proves the lemma. □

Since we have reasoned about abstract virtual surplus, which could be value or Myerson virtual value, we obtain revenue and welfare approximation results for the extremal buffering algorithm.

Corollary 17. *Given a monotone allocation rule* $\mathbf{x}$*, breakpoints* $\hat{Q}$*, and positive integer* k*, the* k*-bin extremal buffering algorithm for* $\mathbf{x}$ *and* $\hat{Q}$ *obtains at least a* $(1 - \hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k}(1-(n-1)(\hat{q}^k+(1-\hat{q}^{T-k})))$*-fraction of the expected welfare of* $\mathbf{x}$*. If agents' distributions are regular, then the* k*-bin extremal buffering algorithm obtains at least a* $(1-\hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k}(1-(n-1)(\hat{q}^k+(1-\hat{q}^{T-k})))$*-fraction of the revenue of* $\mathbf{x}$ *as well.*

3.1.3.3. Approximately Optimal Resampling. We observed in Section 3.1.3.1 that the resampling each agent's quantile from their bin could drastically reduce the welfare or revenue of an algorithm. Driving this loss were agents with extreme quantiles: if the base algorithm's welfare was driven primarily by allocating rare but high-valued agents while rejecting all other quantiles, it is very unlikely that the resampling procedure will give these high-valued agents priority.

Applying the extremal buffering procedure before resampling solves exactly this issue. After applying the extremal buffering procedure, resampling does not change the way the mechanism treats the top k and bottom k bins, as agents in those bins are already treated identically. Consequently, any loss from resampling must occur in the central quantiles of the distribution. For decreasing virtual value functions, such loss cannot be too high. In what follows, we formally define our resampling procedure, and then prove its performance guarantee.

Definition 18. *The* resampling algorithm *for allocation algorithm $\hat{\mathbf{x}}$, breakpoints $\hat{Q}$, and k buffered bins resamples a quantile for each agent i with quantile $q_i \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$ uniformly from $[\hat{q}_i^j, \hat{q}_i^{j+1}]$, and runs the k-bin extremal buffering algorithm for $\hat{Q}$ and $\hat{\mathbf{x}}$ on the resampled quantiles.*

Lemma 19. *For each agent i, let ϕ_i be a nondecreasing virtual value function, and let $\mathbf{x}$ be an a monotone allocation algorithm. For any $k \leq T/2$, define*

$$\alpha_k(\hat{Q}) = \min_i \min \left(\min_{k \leq j \leq T-k-1} (\hat{q}_i^j / \hat{q}_i^{j+1}), \min_{k \leq j \leq T-k-1} (1 - \hat{q}_i^{j+1})/(1 - \hat{q}_i^j) \right).$$

The resampling algorithm for $\mathbf{x}$, k, and obtains at least an $\alpha_k(\hat{Q})(1 - \hat{q}^k/\hat{q}^{T-k})\hat{q}^{T-k}(1 - (n-1)(\hat{q}^k + (1 - \hat{q}^{T-k})))$ of the i's virtual surplus under $\mathbf{x}$.

To prove Lemma 19, we first state a useful lemma formalized in Roughgarden and Schrijvers (2016), which characterizes the relationship between the virtual surplus of resampling algorithms and their base algorithms. Informally, the result states that the revenue from bin-based resampling for an allocation rule $\mathbf{x}$ is the same as if $\mathbf{x}$ was run on the binned revenue curve for those same bins. Formally:

Lemma 20 (Roughgarden and Schrijvers, 2016)**.** *Let $\hat{Q}$ be a set of breakpoints for each agent, and for every agent i, let ϕ_i be a virtual value function for each agent, with* *cumulative virtual surplus* $R_i = \int_0^q \phi_i(r)\,dr$. *For any allocation algorithm* $\mathbf{x}$, *let* $\overline{\mathbf{x}}$ *denote the algorithm which resample's each agent's quantile* $q_i \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$ *uniformly from their bin* $[\hat{q}_i^j, \hat{q}_i^{j+1}]$ *and runs* $\mathbf{x}$ *on the resampled quantiles. Then for each agent* i *we have:*

$$\mathbb{E}[-\overline{x}_i'(q_i)R_i(q_i)] = \mathbb{E}[-\overline{x}_i'(q_i)R_i^{\hat{Q}_i}(q_i)] = \mathbb{E}[-x_i'(q_i)R_i^{\hat{Q}_i}(q_i)].$$

Proof of Lemma 19. Let $\hat{\mathbf{x}}$ denote the allocation rule of the k-bin extremal buffering algorithm, and $\overline{\mathbf{x}}$ that of the resampling algorithm. Define

$$\hat{R}_i(q_i) = \begin{cases} R_i(q_i) & \text{for } q_i \in [\hat{q}_i^k, \hat{q}_i^{T-k}] \\ \frac{q_i}{\hat{q}_i^k} R_i(\hat{q}_i^k) & \text{for } q_i \in [0, \hat{q}_i^k] \\ \frac{1-q_i}{1-\hat{q}_i^{T-k}} R_i(\hat{q}_i^{T-k}) & \text{for } q_i \in [\hat{q}_i^{T-k}, 1]. \end{cases}$$

That is, $\hat{R}_i$ is equal to R_i except in $[0, \hat{q}_i^k]$ and $[\hat{q}_i^{T-k}, 1]$, where it is a linear interpolation between the revenue curve at the endpoints of those intervals.

We will argue the following sequence of inequalities for each agent:

$$\begin{aligned} \mathbb{E}[-\overline{x}_i'(q_i)R_i(q_i)] &= \mathbb{E}[-\hat{x}_i'(q_i)R_i^{\hat{Q}_i}(q_i)] \\ &\geq \alpha_k(\hat{Q})\mathbb{E}[-\hat{x}_i'(q_i)\hat{R}_i(q_i)] \\ &= \alpha_k(\hat{Q})\mathbb{E}[-\hat{x}_i'(q_i)R_i(q_i)]. \end{aligned}$$

The first equality follows from Lemma 20. We will prove shortly that $R_i^{\hat{Q}_i}(q_i) \geq \alpha_k(\hat{Q})\hat{R}_i(q_i)$ for all $q_i \in [0,1]$, which implies the middle inequality. The last equality follows from the fact that the extremal buffering algorithm treats agents in the first and last k bins identically. Since the first and last expressions in the above chain are the respective virtual surpluses of the binning and buffering algorithms, respectively, the result will follow.

To see that $R_i^{\hat{Q}_i}(q_i) \geq \alpha_k(\hat{Q})\hat{R}_i(q_i)$, note that $R_i^{\hat{Q}_i}(q_i) = \hat{R}_i(q_i)$ for all quantiles in $[0, \hat{q}_i^k] \cup [\hat{q}_i^{T-k}, 1]$. Otherwise, consider $q_i \in [\hat{q}_i^j, \hat{q}_i^{j+1}]$ for $j \in \{k, \ldots, T-k-1\}$. Assume without loss of generality that $R_i(\hat{q}_i^j) \leq R_i(\hat{q}_i^{j+1})$; a symmetric argument will apply to the case where $R_i(\hat{q}_i^j) \geq R_i(\hat{q}_i^{j+1})$. By the definition of $R_i^{\hat{Q}_i}$ and the concavity of R_i, it must be that $R_i^{\hat{Q}_i}(q_i) \geq R_i(\hat{q}_i^j)$. Moreover, the concavity of R_i implies that $\hat{R}_i(q_i) = R_i(q_i) \leq \frac{\hat{q}_i^j}{\hat{q}_i^{j+1}} R_i(\hat{q}_i^j)$. The result follows. □

Proof of Theorem 7. We proved in Theorem 8 that the mechanism which maximizes revenue for the binned revenue curves $R_i^{\hat{Q}}$ is optimal among all mechanisms which treat agents based only on their bin. It therefore inherits the revenue guarantees of the k-bin resampling mechanism for any choice of k. Choosing $k = \sqrt{n/T}$ and applying the bound of Lemma 19 yields the result. □

3.2. Revenue Maximization from Samples

In the previous section, we showed that given knowledge of each agent's revenue curve at a collection of sufficiently evenly-spaced (but not necessarily uniform) points, it is possible to nearly maximize revenue by simply interpolating linearly between these points and maximizing revenue according to this approximation. We now show how to emulate this process using *estimates* of the revenue curve at evenly-spaced points.

To estimate a each agent's revenue curve from samples, we will use a simple ranking procedure. A consequence of the Chernoff-Hoeffding inequality is that given some large positive integers T and m, the $jm-1$st-highest sample among $Tm-1$, denoted $\hat{v}_i^j$ will have quantile concentrated around j/T. It will follow that if we treat $\hat{v}_i^j$ as the true value at quantile j/T, we will obtain a close estimate to the binned revenue curve for the true quantiles $q_i(\hat{v}_i^j)$. Maximizing revenue with respect to a close estimate of the true revenue curve yields a close revenue approximation. We first formalize our estimates for each agent's revenue curve given value estimates:

Definition 21. *For any agent i, given values $\hat{v}_i^j$ for $j \in \{0, 1, \ldots, T-1\}$, define the* estimated revenue curve *$\hat{R}_i$ for $\hat{\mathbf{v}}_\mathbf{i}$ as:*

$$\hat{R}_i(q) = \begin{cases} \frac{q}{T}\hat{v}_i^1 & \text{for } q \in [0, 1/T] \\ (Tq-j)\frac{j+1}{T}\hat{v}_i^{j+1} + (j+1-Tq)\frac{j}{T}\hat{v}_i^j & \text{for } q \in [j/T, (j+1)/T] \\ & \text{with } j \in \{1, \ldots, T-2\} \\ \frac{1-q}{1-1/T}\frac{T-1}{T}\hat{v}_i^{T-1} & \text{for } q \in [(T-1)/T, 1] \end{cases} \tag{3.3}$$

It is now straightforward to define our mechanism from samples:

Definition 22. *Given a target loss $\epsilon > 0$ and sample access to regular distributions $F_1, \ldots, F_n$, the* binning mechanism from samples*:*

(1) *Sample $mT-1$ value profiles (containing one value per agent).*
(2) *For all i and j, let $\hat{v}_i^j$ be the $jm-1$st highest sample from distribution i.*
(3) *For each agent i, construct the* estimated revenue curve *$\hat{R}_i$ for $\hat{\mathbf{v}}_\mathbf{i}$.*
(4) *Maximize revenue with respect to the estimated revenue curves.*

Note that the binning mechanism from samples does indeed treat agents in each bin identically: the slope of the estimated revenue curve is constant on each interval $[j/T, (j+1)/T]$, and therefore agents in each bin have the same estimated virtual value. It follows that they will be treated the same.

In what follows, we show that the binning mechanism from samples gets a $(1-\epsilon)$-fraction of the optimal expected revenue with probability at least $(1-\epsilon)$. The analysis consists of three steps. First, we prove a straightforward technical lemma: with $Tm-1$ sampled value profiles, $q_i(\hat{v}_i^j)$ concentrates around j/T. Formally, we will show:

Lemma 23. *Let ϵ and δ be given. There exist T and m polynomial in ϵ, δ, and n such that with probability at least $(1-\delta)$, $q_i(\hat{v}_i^j) \in [j/T-\epsilon, j/T+\epsilon]$ simultaneously for all i and j.*

We prove this lemma in the appendix. Second, we show that in the event that $q_i(\hat{v}_i^j) \in [j/T-\delta, j/T+\delta]$, the estimated revenue curve is everywhere close to the binned revenue curve for $q_i(\hat{v}_i^j)$. Finally, we show that this closeness of revenue curves implies that the binning mechanism from samples performs comparably to the mechanism which maximizes revenue with respect to the binned revenue curves, which we proved in the previous sections performs almost-optimally. Together, these steps will prove the following theorem:

Theorem 24. *For regular distributions $F_1, \ldots, F_n$, any feasibility environment, and any $\epsilon > 0$, there exist choices of T and m which are polynomial in ϵ and n such that the binning mechanism from samples obtains at least $(1-\epsilon)$-fraction of the expected revenue of the optimal mechanism for $F_1, \ldots, F_n$, with probability at least $(1-\epsilon)$.*

3.2.1. Estimating Binned Revenue Curves

In this section, we show that assuming the quantiles of the breakpoints estimated by the binning algorithm from samples are sufficiently close to the uniform breakpoints $1/T, 2/T, \ldots, (T-1)/T$, the estimated revenue curve for $\mathbf{\hat{v}_i}$ will be close to the binned revenue curve for the estimated breakpoints. Formally, define $\hat{q}_i^j = q_i(\hat{v}_i^j)$ for all i and j. We will prove:

Lemma 25. *Fix an agent i. Let $Q_i = \{\hat{q}_i^j\}_{j=0,\ldots,T}$. Conditioned on the event that $\hat{q}_i^j \in [j/T - \epsilon, j/T + \epsilon]$, then for any $q_i \in [0,1]$, $|\hat{R}_i(q_i) - R^{\hat{Q}_i}(q_i)| \leq O(\epsilon T)R_i^*$, where $R_i^* = \max_q R_i(q)$.*

Proof. We we will show $\hat{R}_i(q_i) - R^{\hat{Q}_i}(q_i) \leq O(\epsilon T)R_i^*$, i.e. that the estimated revenue curve does not overestimate by too much. Bounding the magnitude of the underestimate will follow from a similar argument.

First, note that $\hat{R}_i(q_i)$ is maximized pointwise for all q_i when $\hat{q}_i^j = j/T - \epsilon$ for all j. Assume this is the case. In this situation, it is easy to show that $\hat{R}_i(q_i) \leq R_i(q_i - \epsilon) + \frac{\epsilon}{q_i}R_i(q_i - \epsilon) \leq R_i(q_i - \epsilon) + \epsilon T R^*$. This upperbounds $\hat{R}_i(q_i)$. To lowerbound $R_i^{\hat{Q}_i}(q_i)$, note that because $R_i^{\hat{Q}_i}$ is a piecewise linear interpolation between the revenue values at the breakpoints in $\hat{Q}_i$, its slope is always at least $-TR^*(q)$. Hence, $R_i^{\hat{Q}_i}(q_i) \geq R_i(q_i - \epsilon) - \epsilon T R^*$. This implies the lemma. □

3.2.2. Propagation of Error

In the previous section, we showed that the binning mechanism from samples estimates a revenue curve which is close to the binned revenue curve. We further know that maximizing revenue with respect to the binned revenue curves is almost revenue-optimal. In this section, we complete the proof of our sample complexity result by showing that error in estimation of the binned revenue curves propagates cleanly to the performance of the mechanism. In particular, we have the following:

Lemma 26. *Given a collection of breakpoints $\hat{Q}$ and estimates $\hat{v}_i^j$ of $v_i(\hat{q}_i^j)$ for each i and breakpoint $\hat{q}_i^j$, If $q_i(\hat{v}_i^j) \in [j/T - \epsilon, j/T + \epsilon]$ for all i and j, then the revenue of mechanism which maximizes revenue with respect to the binned revenue curves $R_i^{\hat{Q}}$ differs from that of the binning algorithm from samples with estimates $\hat{\mathbf{v}}$ by at most $O(\epsilon T)\sum_i R_i^*$.*

Proof. The result will follow from the following sequence of inequalities:

$$\begin{aligned}
\sum_i \mathbb{E}[-\hat{x}_i'(q_i)R_i(q_i)] &= \textstyle\sum_i \mathbb{E}[-\hat{x}_i'(q_i)R_i^{\hat{Q}_i}(q_i)] \\
&\geq \textstyle\sum_i \mathbb{E}[-\hat{x}_i'(q_i)\hat{R}_i(q_i)] - O(\epsilon T)\sum_i R_i^* \\
&\geq \textstyle\sum_i \mathbb{E}[-x_i'(q_i)\hat{R}_i(q_i)] - O(\epsilon T)\sum_i R_i^* \\
&\geq \textstyle\sum_i \mathbb{E}[-x_i'(q_i)R_i^{\hat{Q}}(q_i)] - O(\epsilon T)\sum_i R_i^* \\
&= \textstyle\sum_i \mathbb{E}[-x_i'(q_i)R_i(q_i)] - O(\epsilon T)\sum_i R_i^*.
\end{aligned}$$

The first equality follows from the the fact that the binning algorithm treats all quantiles in each bin identically. The first inequality comes from, Lemma 25k combined with the fact that $-\hat{x}_i'$ is nonnegative and integrates to at most 1. The second equality follows from the fact that $\hat{\mathbf{x}}$ maximizes revenue with respect to $\hat{R}_i$, while $\mathbf{x}$ does not. The third inequality follows from applying Lemma 25 again, and the final equality follows from the fact that the optimal mechanism for the binned revenue curves treats agents based solely on their bin. □

Proof of Theorem 24. The theorem follows from combining Lemmas 23 and 26 with Theorem 7, and noting that $\sum_i R_i^*$ is an upper bound on the optimal revenue. □

CHAPTER 4

Rank-Based Mechanism Design

In this section we define a parametrized family of winner-pays-bid and all-pay mechanisms for the batched environment and discuss their equilibria and optimization. Recall that the batched environment is a single-dimensional environment comprised of T stages. Each stage is comprised of n buyers, a set $\mathcal{X}$ of feasible allocations, and value distributions $F_1, \ldots, F_n$. Each distribution corresponds to a population, and the T agents for each population i have values which are independent and identically distributed according to distribution F_i. In this and the next chapter, we argue:

(1) Every mechanism in our parametrized family has a focal equilibrium which is easy to characterize.

(2) There exists a choice of parameters such that our mechanism's revenue (or welfare) is close to that of the revenue (or welfare-) optimal mechanism (i.e. from outside our family).

(3) The optimal choice of parameters in Part (2) is can be easily inferred from any nontrivial mechanism in our family.

Our restriction to non-truthful payment semantics introduces challenges for proving all three of the above claims. The first claim is challenging because we make no restriction on the set of feasible allocations $\mathcal{X}$ of the stage environment. For such environments, the equilibria of winner-pays-bid and all-pay mechanisms can often be complicated due to the asymmetry that might be present in the feasibility environment an distributions. Moreover, because of this asymmetry, there is no assurance that equilibrium will trade off allocating agents from distributions in a revenue- or welfare-optimal way, which makes the second claim difficult. Finally, inference must be from nontruthful equilibrium bids, rather than agents' true values.

To overcome these the challenges to establishing claims (1)-(3), we will exploit a symmetry inherent in the batched population environment: agents in each population are i.i.d., and each serve a similar role in the stage feasibility environments. By forcing agents in each population to compete with their peers, we can obtain rank information for each agent's value within their population, which we can use as a proxy for their quantile, allowing our mechanism to effectively trade off allocation both within and across distributions in an approximately revenue- or welfare-optimal way, achieving (2). Moreover, our mechanisms make each agent's incentives resemble those they would experience in a position auction against identically distributed competitors. In such environments, BNE easy to characterize, solving (1), and inference has been well-studied in Chawla et al. (2014). In particular, Chawla et al. (2014) give a procedure to infer distributional parameters which will yield the optimal mechanism in our family, solving (3).

In Section 4.1, we formally define the family of mechanisms, *surrogate ranking mechanisms*, that we will study. Surrogate ranking mechanisms can be implemented with either winner-pays-bid or all-pay semantics, and in Section 4.2, we show that both implementations have a focal and well-behaved BNE. We derive the welfare- and revenue-optimal surrogate ranking algorithms in Section 4.3.

4.1. Surrogate-Ranking Mechanisms

We now describe the family of algorithms, *surrogate-ranking algorithms*, which will serve as a base for our rank-based mechanisms in the population environment, and present notation and concepts which will aid in reasoning about this family. These algorithms treat each agent against others in their population, and uses this rank information to determine allocations. We will consider a special case of such rank-based mechanisms, which force agents to compete for *surrogate values*, which are input to an allocation algorithm instead of the agent's true value. Higher-valued agents will win higher surrogate values, which will yield higher probability of allocation when input to a monotone allocation algorithm. Formally:

Definition 27. *The* surrogate ranking algorithm *(SRA) for the batched environment is parameterized by nT surrogate values, denoted $\Psi_i = \{\psi_i^1 \geq \ldots \geq \psi_i^T\}$ for each population i, and a stage allocation algorithm $\hat{\mathbf{x}}$ that maps a profile of n surrogate values to a feasible allocation $\mathbf{x} \in \mathcal{X}$. The algorithm on bids from each of the nT agents works as follows:*

(1) *For each population i and stage t, compute the rank $r(i,t)$ of the bid of the agent in population i and stage t with respect to the $T-1$ other bids from population i.*
(2) *For each stage t, allocate to agents in stage t according to $\hat{\mathbf{x}}(\psi_1^{r(1,t)}, \ldots, \psi_n^{r(n,t)})$.*

As described in Section 2.1, we consider mechanisms that are defined by an allocation algorithm and a payment semantic. Our analysis will include the *surrogate ranking mechanisms* (SRMs) with both winner-pays-bid and all-pay semantics.

In reasoning about surrogate-ranking algorithms, it will be useful to consider stage algorithms that operate by assigning agents to surrogate values not just based on rank (as in Definition 27) but by other methods. In such algorithms, agent i's surrogate values are denoted by $\Psi_i = \{\psi_i^1 \geq \cdots \geq \psi_i^T\}$, and these are coupled with a stage allocation algorithm $\hat{\mathbf{x}}$ which maps a profile of n surrogate values to a feasible allocation $\mathbf{x} \in \mathcal{X}$. The following definition characterizes the outcome of such a stage allocation algorithm when the profile of surrogate values is uniformly distributed.

Definition 28. *For nT surrogate values, denoted $\Psi_i = \{\psi_i^1 \geq \ldots \geq \psi_i^T\}$ for each population i, and a stage allocation algorithm $\hat{\mathbf{x}}$ that maps a profile of n surrogate values to a feasible allocation in $\mathcal{X}$, the* characteristic weights $(w_i^1, \ldots, w_i^T)$ *for population i are defined by calculating the allocation probability of each surrogate when the surrogates of other populations are drawn uniformly at random, i.e., $w_i^j = \mathbb{E}[\hat{x}_i(\psi_i^j, \boldsymbol{\psi}_{-i})]$ for each surrogate j and uniform random $\boldsymbol{\psi}_{-i}$.*

In what follows, we discuss several algorithms for assigning agents to surrogate values which will induce uniform surrogate distributions. The agents in a mechanism based on such an algorithm effectively compete for allocation probabilities equal to the characteristic weights. The input distribution to the stage algorithm $\hat{\mathbf{x}}$ is jointly determined by two factors: the rule used to select surrogate values, and the distribution of bids input to this rule. Formally:

Definition 29. *Given surrogate values $\Psi_i = \{\psi_i^1, \ldots, \psi_i^T\}$ for agent i, a* surrogate selection rule *for Ψ_i is a function σ_i mapping bids for i to surrogate values in Ψ_i.*

Our analysis will focus on two particular surrogate selection rules (and their associated algorithms). The *sample ranking* rule samples bids according to some distribution and chooses a surrogate value based on the rank of agent i's bid among the samples. The *binning* rule emulates the binning algorithms of the previous chapter and divides bid space into T intervals of equal probability according to some distribution and maps a bid in the jth highest interval to the jth highest surrogate value. Formally (with the subscript for population i omitted):

Definition 30. *Given distribution G and set of surrogate values Ψ, with $\psi^1 \geq \ldots \geq \psi^T$, the* sample-ranking *selection rule for G and Ψ draws $T-1$ samples from G, computes the rank of r of input bid b among the $T-1$ samples, and outputs surrogate value ψ^r.*

Definition 31. *Given distibution G, set of surrogate values Ψ with $\psi^1 \geq \ldots \geq \psi^T$, and partitioning $\mathcal{I} = \{I^1, \ldots, I^T\}$ of G's support into intervals of equal probability, the* binning *selection rule for G and Ψ maps input bid b to ψ^j for the j for which $I^j \ni b$.*

The surrogate ranking algorithm (Definition 27) implements the sample ranking selection rule with samples being drawn from the symmetric equilibrium bid distribution for each population. If i's bid is distributed according to the same distribution, then their rank among the samples will be uniformly distributed. This property will in turn mean that agents' allocations are determined by their characteristic weights. Formally:

Definition 32. *Given a distribution over bids G and a surrogate selection rule σ with surrogates $\Psi = \{\psi^1, \ldots, \psi^T\}$, σ* induces uniformity *for G if $\mathbb{P}_{b \sim G}[\sigma(b) = \psi^j] = 1/T$ for all $j \in \{1, \ldots, T\}$.*

The following lemmas are immediate from the definitions.

Lemma 33. *The sample ranking and binning surrogate selection rules for any distribution induce uniformity for inputs drawn from that same distribution.*

Lemma 34. *The composition of a stage allocation algorithm with a surrogate selection rule that induces uniformity in its BNE bid distribution allocates to bidders according to its characteristic weights. Specifically, a bidder from population i who is assigned surrogate value ψ_i^j is allocated with probability w_i^j.*

4.2. Incentives

We now show that from each agent's perspective, surrogate-ranking mechanisms hide asymmetry that might be present in the stage settings. In particular, they induce a rank-based position auction among agents from each population, forcing agents to compete for the characteristic weights of their population's surrogate values. With pay-your-bid or all-pay payment semantics, they therefore inherit the equilibrium of rank-based position auctions, which is shown to be unique in Chawla and Hartline (2013). Formally:

Theorem 35. *For a monotone*[1] *stage allocation algorithm $\hat{\mathbf{x}}$, a set of T surrogate values $\Psi_i = \{\psi_i^1 \geq \ldots \geq \psi_i^T\}$ for each population i, and their characteristic weights w_i^j for each i and j, there is a BNE in the winner-pays-bid (resp. all-pay) SRM where for each population i, the agents in population i bid according to the unique BNE of the i.i.d. rank-based winner-pays-bid (resp. all-pay) position auction with position weights $w_i^1, \ldots, w_i^T$ and value distribution F_i.*

Proof. We argue from the perspective of agents in an arbitrary population i. Assume agents in other populations are bidding according to the position auction equilibrium for the characteristic weights of their populations. By Lemma 33, symmetric bid distributions for each population induce uniformity for each population in every stage. Lemma 34 then

[1]We will see in Chapter 5 that assuming monotonicity is without loss.

implies that the agent in population i who is assigned surrogate value ψ_i^j is allocated with probability w_i^j. Thus, the agents in population i bid according to the equilibria of the rank-based winner-pays-bid (resp. all-pay) auction with position weights $w_i^1, \ldots, w_i^T$. Monotonicity of the allocation rule implies monotonicity of the position weights; it follows that the unique equilibrium of agents from population i (as guaranteed by Theorem 4) assigns the agents to positions according to their values. We conclude that the prescribed strategies are an equilibrium for agents from all populations. □

The equilibrium of Theorem 35 is unique under the natural assumption that agents are not able to condition their strategy on the label of their stage. The uniqueness follows from the fact that such an equilibrium is necessarily symmetric within each population, and therefore induces uniformity. Hence, the equilibrium appears to agents in each population as a position auction with position weights equal to the characteristic weights. The symmetric equilibrium for such an auction is unique, by a straightforward application of revenue equivalence (i.e. the second part of Theorem 2). This yields:

Theorem 36. *The equilibria of Theorem 35 for winner-pays-bid and all-pay SRMs are unique among stage-invariant BNE.*

As the equilibrium of Theorem 35 is the unique equilibrium which is symmetric among agents in a population, we will refer to it as the *symmetric equilibrium* of the winner-pays-bid or all-pay SRM.

Definition 37. *In the* symmetric equilibrium *of the winner-pays-bid (resp. all-pay) SRM, agents in each population i bid according to the unique BNE of the i.i.d. winner-pays-bid (resp. all-pay) rank-by-bid position auction for the position environment with position weights equal to the characteristic weights $w_i^1 \geq \ldots \geq w_i^T$ and distribution F_i.*

The design of revelation mechanisms is facilitated by the fact that the assumed equilibrium has agents bidding their true values. Thus, if the outcome of the allocation algorithm has good properties with respect to its input bids, those properties also hold with respect to the agents' values. Because the equilibria of surrogate ranking mechanisms are monotone within each population, they assign agents in each population to surrogates in the order of their values. This is the same allocation that would be achieved if the agents bid their values. Thus, for analysis of welfare and virtual welfare, we are free to consider the surrogate ranking algorithm on the true values of the agents. This serves as a sort of "revelation principle" for the analysis of rank-based mechanisms.

Theorem 38. *The allocation of the symmetric equilibrium of a surrogate ranking mechanism is the same as the outcome of the corresponding surrogate ranking algorithm on the true values of the agents.*

More often than not, a mechanism designer who must design for many iterations of an environment must choose allocations and payments for each iteration one-at-a-time, rather than in a batched fashion. That is, they must design for an *online* version of the iterated population environment, where they may select each iteration's allocation and payments using past bid data from each population in previous iterations.

The surrogate ranking mechanism suggests a way to design mechanisms that use this data. Samples from the past bid distribution can be used to calculate the rank of the value of an agent in the present stage, as long as that agent's bid is from the same distribution. The surrogate ranking algorithm (Definition 27) for the batched environment obtained these samples from the agents from the same population but in different stages of the batched environment. The following algorithm for the stage environment relaxes the assumption that the stages are batched and replaces it with direct sample access to the (supposed) bid distribution. These sampled bids could be obtained, for example, by previous iterations of the stage mechanism.

Definition 39. *The* surrogate sample ranking algorithm *for the stage environment is parameterized by nT surrogate values, denoted $\Psi_i = \{\psi_i^1 \geq \ldots \geq \psi_i^T\}$ for each population i, a stage allocation algorithm $\hat{\mathbf{x}}$ that maps a profile of n surrogate values to a feasible allocation $\mathbf{x} \in \mathcal{X}$, and n bid distributions $G_1, \ldots, G_n$. The algorithm, on bids from each of the n agents, works as follows:*

(1) *For each agent i, draw $T-1$ samples from the bid distribution G_i.*

(2) *For each agent i, compute the rank $r(i)$ of the bid of the agent with respect to the $T-1$ bids sampled from G_i.*

(3) *Allocate to agents according to $\hat{\mathbf{x}}(\psi_1^{r(1)}, \ldots, \psi_n^{r(n)})$.*

The analysis of Theorem 35 shows that if for all agents i, G_i is the bid distribution for population i in the symmetric equilibrium of the surrogate ranking mechanism, then it is a best response for agent i to bid according to that same equilibrium. In fact, Theorem 36 implies that this is the unique such stationary point. To see this, assume there was another set of distributions $\{G_i\}_{i=1}^n$ with the property that the best response in the winner-pays-bid or all-pay surrogate sample ranking mechanisms for $\{G_i\}_{i=1}^n$ was for agent i to bid according to G_i. Note that this would also be an equilibrium of the corresponding SRM. By Theorem 36, the symmetric equilibrium is the unique equilibrium that is stage invariant. The following theorem summarizes this argument.

Theorem 40. *For a monotone stage allocation algorithm $\hat{\mathbf{x}}$, surrogate values $\{\Psi_i\}_{i=1}^n$, and the SRM symmetric equilibium bid distributions $G_1, \ldots, G_n$, bidding according to $\{G_i\}_{i=1}^n$ is a BNE of the surrogate sample ranking mechanism for $\hat{\mathbf{x}}$, $\{\Psi_i\}_{i=1}^n$, and $\{G_i\}_{i=1}^n$.*

Using the past $T-1$ days a the source of samples according to the distributions $G_1, \ldots, G_n$ provides a solution to the problem of online mechanism design in the population model.

4.3. Optimal Surrogate-Ranking Mechanisms

In this section we derive the optimal surrogate-ranking mechanisms for welfare and revenue, assuming agents play the symmetric equilibrium. Since agents are ranked according to their true values in this equilibrium, it suffices to optimize over the underlying surrogate ranking algorithms instead. Consequently, we may use reasoning reminiscent of the derivation of the optimal truthful binning mechanism for revenue in Chapter 3.

The free parameters for surrogate ranking algorithms are the choice of surrogate values ψ_i^j for $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, T\}$ and the choice of the stage allocation algorithm $\hat{\mathbf{x}}$. To optimize these parameters, we consider the relaxed algorithm design problem of maximizing a generic virtual surplus quantity subject to the constraint that the algorithm must be rank-based. We note that for monotone virtual value functions, there is an obvious solution to this problem which happens to be a SRM.

Given a Bayesian population environment with distributions $F_1, \ldots, F_n$ and T stages, we define the *rank-based algorithm design problem* as follows: the designer must choose a stage allocation algorithm $\overline{\mathbf{x}}$ which takes as inputs the ranks $\mathbf{r}^t = r_1^t, \ldots, r_n^t$ of agents in an arbitrary stage t within each of their respective populations and outputs a (possibly randomized) feasible allocation $\overline{\mathbf{x}}(\mathbf{r}^t)$ for that stage. As a constraint, $\overline{\mathbf{x}}$ must be monotone for each agent. The objective is to maximize $\mathbb{E}[\sum_t \sum_i \phi_i(v_i^t)\overline{x}_i^t(\mathbf{r}^t)]$ for some given virtual value function $\phi_i(\cdot)$, where ranks are drawn uniformly for each population. (For example, $\phi_i(v_i^t) = v_i^t$ corresponds to welfare maximization.) Note that the restriction to a single allocation algorithm across all stages is without loss of generality, as the stages are symmetric - one may permute the labels of the stages uniformly at random before running the algorithm.

The above optimization problem can be solved by inspection. Fixing an allocation algorithm, the objective can be rewritten as $\sum_t \sum_i \mathbb{E}[\phi_i(v_i^t) \,|\, r_i^t]\overline{x}_i^t(\mathbf{r}^t)$ by linearity of expectation. From this expression, it becomes clear that the optimal algorithm maximizes the quantity $\sum_i \mathbb{E}[\phi_i(v_i^t) \,|\, r_i^t]\overline{x}_i^t(\mathbf{r}^t)$ in every stage. Note that if $\phi_i(\cdot)$ is monotone, then this algorithm will be monotone as well, and therefore feasible.[2] The above proves:

[2] For simplicity, we do not consider non-monotone ϕ_i in what follows. Chawla et al. (2014) show how to derive the optimal rank-based mechanism for such ϕ_i. Their procedure results in a surrogate ranking mechanism and therefore extends the results of this section to the objective of revenue with irregular distributions.

Theorem 41. *The virtual welfare-optimal rank-based algorithm for the batched population setting maximizes* $\sum_i \mathbb{E}[\phi_i(v_i^t) \mid r_i^t]\overline{x}_i^t(\mathbf{r}^t)$ *in each stage. In particular, the welfare-optimal rank-based algorithm uses* $\phi_i(v_i^t) = v_i^t$*, and for regular distributions, the revenue-optimal rank-based algorithm uses* $\phi_i(v_i^t) = v_i^t - \frac{1-F(v_i^t)}{f(v_i^t)}$.

Note that the rank-based algorithm design problem is a relaxation of the surrogate ranking algorithm design problem; any surrogate ranking algorithm with a monotone stage allocation algorithm is feasible. Moreover, the algorithm prescribed by Theorem 41 is a surrogate ranking algorithm. We therefore can conclude that this algorithm is optimal among surrogate ranking algorithms.

Corollary 42. *Assuming agents play the symmetric equilibrium, the welfare-optimal surrogate-ranking mechanism uses surrogate values* $\psi_i^j = \mathbb{E}_{v\sim F_i}[v \mid r_i^j]$ *and allocation algorithm* $\hat{\mathbf{x}}(\mathbf{v}) = argmax_{\hat{\mathbf{x}}\in\mathcal{X}} \sum_i v_i x_i$. *For regular distributions, the revenue-optimal surrogate-ranking mechanism uses surrogate values* $\psi_i^j = \mathbb{E}_{v\sim F_i}\left[\left(v - \frac{1-F(v)}{f(v)}\right) \mid r_i^j\right]$ *and stage allocation algorithm* $\hat{\mathbf{x}}(\mathbf{v}) = argmax_{\hat{\mathbf{x}}\in\mathcal{X}} \sum_i v_i x_i$.

CHAPTER 5

Performance and Inference for Rank-Based Mechanisms

In Chapter 4, we defined surrogate-ranking mechanisms, a class of mechanisms for the iterated population model in which agents compete with other agents from their same distribution to be assigned surrogate values, which are used as inputs to an allocation algorithm. The choice of surrogate values determines how the mechanism will trade off allocation across distributions. In Section 4.3, we derived the optimal choice of surrogate values: to maximize welfare, set ψ_i^j to be the expected value of the jth highest of T draws from F_i, and to maximize welfare, choose the same expected order statistic of the virtual value distribution for F_i.

In this chapter, we analyze the inference of these parameters, and derive performance guarantees given (possibly noisy estimates of) these optimally chosen parameters. Since the focal equilibrium in a surrogate-ranking mechanism with winner-pays-bid or all-pay semantics is the unique BNE for the position environment each population perceives, a straightforward application of the results of Chawla et al. (2014) will allow us to infer the optimal surrogate values for welfare or revenue with low error. Moreover, we show in Section 5.2 that estimation errors propagate to welfare or revenue in a controlled way.

To derive performance guarantees, we extend the analysis of surrogate binning algorithms developed in Chapter 3 to surrogate ranking algorithms. The crux of our approach is an analysis of the ability of rank-based mechanisms to compete with posted pricing. Both welfare and revenue maximization require the designer to be able to discriminate against low-valued buyers. Posted pricing represents perfect discrimination: an agent buys if their value is above a threshold, and goes unallocated if their value is below. A ranking mechanism, which allocates the top k of T bidders, discriminates in value space, but does so less precisely - rank is not always an accurate measure of value. We show that surrogate binning algorithms can be thought of as distributions over few posted prices, and that surrogate ranking algorithms can be thought of as distributions over rank-based k-unit auctions. Consequently, a comparison of ranking to prices reduces the analysis of SRMs to the binning analysis from Chapter 3. We will explicitly construct a surrogate-ranking algorithm which approximates the welfare or revenue of any other allocation algorithm, and which our optimal SRM by definition outperforms. This yields the following:

Theorem 43. *For any monotone*[1] *stage allocation algorithm* $\hat{\mathbf{x}}$*, there exists a surrogate-ranking algorithm which attains a* $(1 - O(\sqrt[3]{n/T}))$*-fraction of the welfare and virtual welfare of* $\hat{\mathbf{x}}$*.*

Combining the theorem above with Theorem 38, i.e., that the allocation of the symmetric equilibrium of a surrogate ranking mechanism is the same as that of the surrogate ranking algorithm on the true values, we have the following corollaries.

Corollary 44. *For any feasibility setting, the welfare-optimal surrogate-ranking mechanism in its symmetric equilibrium attains a* $(1 - O(\sqrt[3]{n/T}))$*-fraction of the surplus of the optimal mechanism.*

[1]The restriction to monotonicity is essentially without loss. Hartline and Lucier (2010) show how to convert a non-monotone allocation algorithm to one which is monotone and obtains higher (virtual) welfare. Their procedure requires access to the distribution over inputs to the algorithm, which we have because the designer controls the choice of surrogate values.

Corollary 45. *For any feasibility setting, the revenue-optimal surrogate-ranking mechanism in its symmetric equilibrium attains a* $(1 - O(\sqrt[3]{n/T}))$*-fraction of the revenue of the optimal mechanism.*

A Prior Result: A weaker result follows from a theorem of Hartline et al. (2011). In this paper, the authors consider an algorithm which can be interpreted as a surrogate ranking algorithm built from an arbitrary stage allocation algorithm $\hat{\mathbf{x}}$. They show that for agents whose values are distributed on the interval $[0, 1]$, their mechanism has an additive per-stage welfare loss of at most $n/(4\sqrt{T})$. Because we derive the welfare-optimal surrogate-ranking mechanism, we inherit this welfare guarantee. In particular, we have:

Corollary 46 (Hartline et al. (2011))**.** *In any feasibility setting, if agents have value distributions on* $[0, 1]$*, then in the symmetric equilibrium, the welfare-optimal surrogate-ranking mechanism loses at most an additive* $n/(4\sqrt{T})$ *per stage with respect to the welfare of the optimal mechanism.*

We improve on this welfare guarantee in three ways. First, we derive a guarantee for arbitrary distributions, even those with unbounded support. Second, our bounds will be multiplicative. Finally, our bounds apply to revenue in addition to welfare.[2]

5.1. Inference

We saw in Section 4.3 that the optimal choice of surrogate values requires mild distributional knowledge in the form of expected order statistics of the value or virtual value

[2]With the additional assumption that virtual values are bounded below by $-\underline{\phi}$, the guarantee of Corollary 46 applies to revenue as well, with an additional factor of $1 + \underline{\phi}$ applied to the loss.

function. We now discuss how to use the tools of Chawla et al. (2014, 2016) to infer the optimal surrogate values from samples.

Chawla et al. (2016) consider inference of the expected order statistics of value and virtual value distributions (i.e. the optimal surrogate values for welfare and revenue) in i.i.d. position auctions with values distributed on $[0,1]$. Because each population of a surrogate-ranking mechanism inherits the unique BNE of the position environment with that population's characteristic weights, we may directly apply their procedure to samples from a batched or online SRM with possibly suboptimal parameters to estimate the optimal surrogate values. Their procedure only requires that for each population, some pair of characteristic weights differs by at least ϵ, so that the equilibrium is nontrivial. Denoting by P_i^k the kth highest optimal surrogate value for revenue for population i, and V_i^k the kth highest optimal surrogate value for welfare for population i, we obtain

Corollary 47 (Chawla et al. (2016)). *Consider a batched environment with T stages and values drawn from $[0,1]$ for each population. Assume surrogate values for a winner-pays-bid SRM are selected such that for every population i, there exists some j such that the characteristic weights satisfy $w_i^{j+1} - w_i^j > \epsilon$. Then given N samples of the batched environment with T stages per sample, there exist estimators $\hat{P}_i^k$ and $\hat{V}_i^k$ satisfying:*

$$\mathbb{E}[|\hat{P}_i^k - P_i^k|] \leq \Theta\left(\frac{T}{\sqrt{\min(k, T-k)}}\frac{1}{\sqrt{N}}\log\left(\frac{T}{\epsilon}\right)\right)$$

$$\mathbb{E}[|\hat{V}_i^k - V_i^k|] \leq \Theta\left(\frac{T\log T}{\sqrt{N}}\log\left(\frac{T}{\epsilon}\right) + \frac{T}{N\epsilon}\right)$$

A similar result holds for all-pay SRMs. It follows that as long as the initial choice of surrogate values is not trivial, it is easy to infer better surrogate values and reoptimize the mechanism. Note that the convergence rates of the estimators of Chawla et al. (2016) are faster than if one tried to infer the full value distributions, which would be required to implement the optimal winner-pays-bid or all-pay mechanism by undoing the revelation principle (as described in Section 1.4).

5.2. Propogation of Error

We now show that error in our estimates of the optimal surrogate values translates to a loss in performance of the optimal surrogate ranking mechanism in a straightforward way. The proof resembles that of Lemma 48: the expected revenue or welfare of a rank-based mechanism is exactly its expected surplus with respect to the optimal surrogate values. Small noise in these surrogate values can only cause the optimal surrogate-ranking mechanism to misallocate by a bounded amount.

Lemma 48. *Let* $\hat{\Psi} = \{\hat{\psi}_i^j\}_{i\in\{1,\dots,n\}}^{j\in\{1,\dots,T\}}$ *be estimates of the revenue-optimal surrogate values* ψ_i^j *satisfying* $\hat{\psi}_i^j \in [\psi_i^j - \gamma_i, \psi_i^j + \gamma_i]$ *for all* i *and* j*. Let* $Rev^{\hat{\Psi}}$ *be the revenue of the surrogate ranking mechanism using* $\hat{\Psi}$*, and let* Rev^{Ψ} *be revenue of the optimal surrogate ranking mechanism, using* Ψ*. Then:*

$$Rev^{\hat{\Psi}} \geq Rev^{\Psi} - 2\sum_i \gamma_i.$$

An analogous proposition holds for welfare.

Proof. Let $\hat{\mathbf{x}}$ be the allocation rule of the algorithm using the estimates $\hat{\Psi}$, and $\mathbf{x}$ the allocation rule of the algorithm using the true optimal surrogate values Ψ. Given a value profile $\mathbf{v}$, let $r_i^t(\mathbf{v})$ be the rank of agent i in stage t based on the value profile $\mathbf{v}$. The expected revenue from $\hat{\mathbf{x}}$ is:

$$\begin{aligned}\mathbb{E}_{\mathbf{v}}\left[\sum_t \sum_i \mathbb{E}\left[\phi_i(v)\,|\, r_i^t(\mathbf{v})\right]\hat{x}_i^t(\mathbf{v})\right] &= \mathbb{E}_{\mathbf{v}}\left[\sum_t \sum_i \psi_i^{r_i^t(v_i)}\hat{x}_i^t(\mathbf{v})\right] \\ &\geq \mathbb{E}_{\mathbf{v}}\left[\sum_t \sum_i \hat{\psi}_i^{r_i^t(\mathbf{v})}\hat{x}_i^t(\mathbf{v})\right] - \sum_i \gamma_i \\ &\geq \mathbb{E}_{\mathbf{v}}\left[\sum_t \sum_i \hat{\psi}_i^{r_i^t(\mathbf{v})}x_i^t(\mathbf{v})\right] - \sum_i \gamma_i \\ &\geq \mathbb{E}_{\mathbf{v}}\left[\sum_t \sum_i \psi_i^{r_i^t(\mathbf{v})}x_i^t(\mathbf{v})\right] - 2\sum_i \gamma_i.\end{aligned}$$

The first term in the final expression is simply the virtual surplus from $\mathbf{x}$, completing the proof. The proof for welfare is identical. □

5.3. Pricing Versus Ranking

For a mechanism to maximize welfare or revenue effectively, it must be able to discriminate between agents with high and low values. To prove Theorem 43 we must show that ranking mechanisms can do this effectively. We build towards this goal by first showing that ranking mechanisms can approximate the simplest form of discrimination: posted pricing. We will describe price-posting mechanisms in terms of the location of the price p via its quantile $q(p) = 1 - F(p)$. We in particular consider prices for which $q(p)$ is an integral multiple of $1/n$. Formally:

Definition 49. *The k/n-price posting algorithm allocates agents if and only if their quantile is below k/n, for some integer k. This can be achieved by posting the price with quantile k/n.*

We first show that price-posting can be approximated by ranking. Posting a price at the quantile k/n will result in allocation to k of the n agents in expectation. The rank-based equivalent enforces this quota pointwise, allocating the k highest-valued agents each time.

Definition 50. *The top k-of-n algorithm for n agents ranks agents by value and allocates the k agents with the highest values.*

As the law of large numbers might suggest, these two algorithms perform comparably for large n when k is bounded away from the extremes (one and $n-1$). Formally:

Lemma 51. *For any distribution F, the top k-of-n algorithm attains a $\rho(k,n)$-fraction of the welfare of the k/n-price posting algorithm with n agents. If F is regular, then it attains a $\eta(\min(k,n-k),n)$-fraction of the revenue of the k/n-price posting algorithm, where*

$$\rho(k,n) \approx 1 - \sqrt{\frac{n}{2\pi k(n-k)}} \quad \textit{and} \quad \eta(k,n) \approx 1 - \frac{1}{\sqrt{k}}\left(\frac{n}{n-k}\right)^{\frac{3}{2}},$$

with the error stemming from Stirling's approximation.

Proof. We will explicitly characterize the worst-case distributions for each objective, and analyze the per-agent contribution to each algorithm's surplus. For notational convenience, we suppress the subscripts on functions which would refer to our agent.

Key to the analysis will be two formulae for the expected surplus of an algorithm, in terms of its interim allocation rule $x(\cdot)$ and the distribution's value function $v(\cdot)$. We have that an algorithm's surplus is:

$$\mathbb{E}_{q\sim U[0,1]}[x(q)v(q)] = \mathbb{E}_{q\sim U[0,1]}[-x'(q)V(q)], \tag{5.1}$$

where $V(q) = \int_0^q v(z)\,dz$, and the equality follows from integration by parts. An analogous formula holds for virtual surplus, with $v(q)$ replaced by the Myerson virtual value at q.

We will first analyze welfare, and then highlight the changes necessary for proving the result for virtual surplus. The only real difference between the two objectives is the fact that values are always positive, whereas virtual values may be negative. This changes the nature of the approximation, as allocating the wrong agent becomes actively harmful to the performance of the algorithm.

Welfare. We begin by normalizing the per-agent surplus of the price-posting mechanism to 1. Note that for the k/n-price posting algorithm, the interim allocation rule is 1 until quantile k/n, and then drops to 0. It follows from equation (5.1) that our normalization is equivalent to the assumption that $V(k/n) = 1$.

Next, we note that because $v(\cdot)$ is positive and decreasing, $V(\cdot)$ is increasing and concave, with $V(0) = 0$. Let $x(\cdot)$ be the allocation rule of the top-k-of-n mechanism. Given our normalization, the problem of finding the worst-case distribution then becomes:

$$\begin{aligned}
\min_{V(\cdot)} \quad & \mathbb{E}_{q\sim U[0,1]}[-x'(q)V(q)] \\
\text{subject to} \quad & V(0) = 0 \\
& V(k/n) = 1 \\
& V(\cdot) \text{ concave} \\
& V(\cdot) \text{ increasing}
\end{aligned}$$

This program can be solved by inspection by noticing that there is pointwise minimal function satisfying the constraints of the program: namely, the optimal $V(q)$ is linear with slope $v(q) = n/k$ for $q \le k/n$, and constant at 1 for $q \ge k/n$. This corresponds to the distribution with k/n mass on the value n/k, and the rest on 0.

Having solved for the worst-case distribution, it remains to compute optimal value of the objective. Note that the allocation rule for the top-k-of-n algorithm is

$$x(q) = \sum_{j=0}^{k-1} \binom{n-1}{k} q^j (1-q)^{n-j-1}.$$

Combining this with our knowledge of the worst-case distribution, we can easily compute the per-agent surplus of the top-k-of-n mechanism. Omitting tedious computations, we get the following formula for the multiplicative loss per agent:

$$\text{Loss}_{\text{W}}(k,n) = 1 - \sum_{i=0}^{k-1} \binom{n}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i} (k-i).$$

To obtain the bound given in the statement of the lemma, note that the following sequence of inequalities holds:

$$\begin{aligned}
\text{Loss}_{\text{W}}(k,n) &= 1 - \sum_{i=0}^{k-1} \binom{n}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i} (k-i) \\
&= \sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i} - \sum_{i=0}^{k-1} \binom{n-1}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i-1} \\
&\le \sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i} - \sum_{i=0}^{k-1} \binom{n-1}{i} \left(\frac{k}{n}\right)^i \left(\frac{n-k}{n}\right)^{n-i} \\
&= \binom{n}{k} \left(\frac{k}{n}\right)^k \left(\frac{n-k}{n}\right)^{n-k}.
\end{aligned}$$

The result for welfare follows from applying Sterling's approximation.

Virtual Surplus. We now adapt the above proof to virtual surplus. The main difference will be the fact that the Myerson virtual value, denoted $\phi(q)$, can be negative. We will additionally use the fact that the Myerson virtual value is the derivative of the price-posting revenue curve. That is, $\phi(q) = R'(q) = \frac{d}{dq}v(q)(1-q)$. It follows that cumulative virtual value has the convenient form $R(q) = v(q)q$.

As before, we normalize the virtual surplus from price-posting to 1. This corresponds with setting $R(q) = 1$. Subject to normalization, we use properties of revenue curves to derive the worst-case distribution for virtual surplus. We assume values are regularly distributed, which implies that $R(q)$ is concave. Moreover, since $R(q) = v(q)q$, we have that $R(0) = R(1) = 0$. These properties yield the following program for the worst-case distribution:

$$\begin{aligned} \min_{R(\cdot)} \quad & \mathbb{E}_{q\sim U[0,1]}[-x'(q)R(q)] \\ \text{subject to} \quad & R(0) = R(1) = 0 \\ & R(k/n) = 1 \\ & R(\cdot) \text{ concave} \end{aligned}$$

Again, this may be solved by inspection. The worst-case $R(\cdot)$ is triangular, with its apex at $(k/n, 1)$. That is, on $[0, k/n]$, $R(q)$ has slope n/k, and on $[k/n, 1]$, it has slope $-n/(n-k)$.

Using the allocation rule of the top-k-of-n mechanism from the welfare proof, we can compute the multiplicative loss per agent for revenue as:

$$\text{Loss}_{\text{R}}(n,k) = \frac{n}{k(n-k)} \sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} (k-i). \tag{5.2}$$

The following sequence of equations yields the result:

$$\begin{aligned}
\text{Loss}_{\text{R}}(n,k) &= \frac{n}{k(n-k)} \left[k \sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} - \sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} i \right] \\
&= \frac{n}{(n-k)} \left[\sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} - \sum_{i=0}^{k-1} \binom{n-1}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i-1} \right] \\
&\leq \frac{n}{(n-k)} \left[\sum_{i=0}^{k} \binom{n}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} - \sum_{i=0}^{k-1} \binom{n-1}{i} \left(\frac{k}{n}\right)^{i} \left(\frac{n-k}{n}\right)^{n-i} \right] \\
&= \frac{n}{(n-k)} \binom{n}{k} \left(\frac{k}{n}\right)^{k} \left(\frac{n-k}{n}\right)^{n-k}
\end{aligned}$$

Applying Stirling's approximation and noting symmetry about $n/2$ yields the result. □

We can further generalize Lemma 51 by comparing distributions over pricing algorithms with the analogous distributions over top-k algorithms. As long as prices avoid the extremes of the distribution, ranking performs well with respect to pricing.

Lemma 52. *For any value distribution F, consider a distribution over k/n-price posting algorithms for n agents, where the highest price is at quantile $\underline{k}/n$ and the lowest price is at quantile $\overline{k}/n$. The same distribution over corresponding top-k-of-n algorithms* *attains a* $\rho(\min(\underline{k}, n-\overline{k}), n)$*-fraction of the welfare of the distribution over price-posting algorithms. If* F *is regular, then the distribution over top-*k*-of-*n *algorithms attains an* $\eta(\min(\underline{k}, n-\overline{k}), n)$*-fraction of the price-posting revenue as well.*

Proof. Lemma 51 implies that for each price in the distribution of the price-posting algorithm, there is a top-k-of-n algorithm which approximates it and which appears with the same probability. The approximation ratio of a distribution over pairwise approximations is at least the approximation from the worst pair. Note that the approximations from Lemma 51 are symmetric about 1/2, and are worst for very low and very high k. It follows that the approximation ratio is driven by the $\underline{k}/n$- and $\overline{k}/n$-price posting algorithm. □

5.4. Performance Guarantees for Optimal SRMS

Lemma 52 shows that in simple settings, ranking can discriminate with almost as much accuracy as pricing. We now extend this idea to the batched population model. We compare the surrogate ranking algorithm with a *uniform surrogate binning algorithm*, which, in the language of Section 4.2 composes the allocation algorithm $\hat{\mathbf{x}}$ with the binning surrogate selection rule (Definition 31) in each stage.

Definition 53. *For the stage allocation algorithm* $\hat{\mathbf{x}}$*, value distributions* $\{F_i\}_{i=1}^n$*, surrogate values* $\{\Psi_i\}_{i=1}^n$*, and binning surrogate selection rules* $\{\sigma_i\}_{i=1}^n$*, the* uniform surrogate binning algorithm *is given by computing* $\tilde{\mathbf{x}}(\mathbf{b}^t) = \hat{\mathbf{x}}(\sigma_1(b_1^t), \ldots, \sigma_n(b_n^t))$ *in each stage* t*.*

We now show that uniform surrogate-binning algorithms can be well-approximated by ranking algorithms. The allocation rules of such binning algorithms appear to agents as piecewise constant functions. Hence, each population i sees a distribution over k/T-price posting algorithms. Moreover, in the batched population environment, we may rank agents against their peers in the same distribution (as in a multi-unit auction) rather than price, just as in Lemma 52. By ranking each agent and treating the agent with rank j as if they were in the jth bin, we produce a surrogate-ranking algorithm. Lemma 52 implies that this algorithm performs almost as well as the binning algorithm, provided that the prices from the binning algorithm do not come from extreme quantiles. In particular, we assume that $\underline{k}$ lowest surrogate values and $\overline{k}$ highest surrogate values are identical. This will ensure that the comparison of our rank-based algorithm will only be to prices which are not too extreme.

Theorem 54. *For monotone stage algorithm $\hat{\mathbf{x}}$ and surrogate values $\psi_i^1 \geq \psi_i^2 \geq \ldots \geq \psi_i^T$ with $\psi_i^1 = \psi_i^2 = \ldots = \psi_i^{\underline{k}}$ and $\psi_i^{\overline{k}} = \psi_i^{\overline{k}+1} = \ldots = \psi_i^T$ for each population i, the uniform surrogate ranking algorithm attains a $\rho(\min(\underline{k}, T - \overline{k}), T)$-fraction of the welfare of the binning algorithm. If distributions are regular, then the surrogate ranking algorithm attains a $\eta(\min(\underline{k}, T - \overline{k}), T)$-fraction of the binning algorithm's virtual surplus.*

To derive Theorem 54, note that by Lemmas 33 and 34, the surrogate-binning algorithm allocates agents according to its characteristic weights. Moreover, we already showed in Section 4.2 that surrogate-ranking algorithms also allocate agents according to characteristic weights. Thus if the two algorithms use the same surrogate values, these characteristic weights will be the same. Next, note that the surrogate-binning and surrogate-ranking algorithms appear to agents as distributions over pricing and top-k algorithms, respectively, with the distributions determined by marginal characteristic weights. Formally:

Lemma 55. *Any uniform surrogate-binning (resp. surrogate-ranking) algorithm with surrogate values* $\{\psi_i^j\}_{i=1,\ldots,n}^{j=1,\ldots,T}$ *appears to agents in each distribution* i *as a distribution over price-posting (resp. top-*k*) algorithms. The probability of offering the price with quantile* $\frac{j}{T}$ *(resp. of allocating* j *units) is given by* $w_i^j - w_i^{j+1}$*, where* $w_i^0 = 1$*,* $w_i^{T+1} = 0$*, and* w_i^j *is the characteristic weight for* ψ_i^j *for* $j = 1, \ldots, T$*.*

Finally, Theorem 54 follows from applying Lemma 52. Notice that if $\psi_i^1 = \ldots = \psi_i^{\underline{k}}$, the uniform binning algorithm's allocation rule on the first $\underline{k}$ intervals of distribution i's quantile space will be constant. In terms of distribution i's randomization over posted pricings, the highest nontrivial price offered has quantile $\underline{k}/n$, and the lowest has quantile $\overline{k}/n$. These extremal quantiles drive the approximation guarantees relating pricing to ranking and, thus, good approximation bounds can be obtained via Lemma 52 if there is not much loss in restricting to binning algorithms that price at moderate quantiles.

5.4.1. Proof of Theorem 43

To prove Theorem 43, we note that our analysis of surrogate binning mechanisms in Chapter 3 provides us with a uniform surrogate binning algorithm which achieves a close approximation to the optimal welfare or revenue. In particular, let $\overline{Q}$ be the set of breakpoints which equipartition each agent's quantile space into T intervals. The resampling algorithm for with bins defined by $\overline{Q}$ and k promoted bins is a uniform surrogate binning algorithm. For every choice k of how many bins to promote, then, we will obtain an approximation result as a corollary of Lemma 19.

All that remains to prove the approximation result is to compose our lemmas and select a value for the parameter k. First, note the binning algorithm's surrogate values are the same for the intervals $[0, \frac{1}{T}], \ldots, [\frac{k-1}{T}, \frac{k}{T}]$ and the intervals $[\frac{T-k}{T}, \frac{T-k+1}{T}], \ldots, [\frac{T-1}{T}, 1]$. It follows that welfare and revenue loss from applying Theorem 54 are $\rho(k, n)$ and $\eta(k, n)$, respectively. Composing this with our approximation result and setting $k = (n/T)^{\frac{2}{3}}$ yields the theorem.

CHAPTER 6

Conclusion

In this thesis, we have attempted to build a theory to study non-truthful mechanisms explicitly from a design perspective. We identified some of the important characteristics of settings where winner-pays-bid and all-pay semantics, two canonical non-truthful mechanism families, perform well. These characteristics included *distributional symmetry*, *symmetry of feasibility structure*, and *ordinality*. In such settings, rank-based mechanisms perform particularly well with non-truthful semantics, which we demonstrated in the context of the iterated population environment. Continued work to understand what properties of an auction setting are conducive (or not) to the design of nontruthful mechanisms, as well as the development of methods for harnessing these properties, is critically important for the development of a theory of mechanism design that aligns with the realities of practical implementations.

In addition to performance and incentives, this work considered inference in non-truthful mechanisms. Again, we identified properties of mechanisms, e.g. ordinality, that facilitate inference, and designed mechanisms (and corresponding inference tools) to take advantage of these properties. We have seen that the theory of non-truthful mechanisms, especially the results of Dütting and Kesselheim (2015) suggest that non-truthful mechanisms generally require data to perform well. The constraints of practical mechanism design requires methods for making sense of bid data as well. Consequently, a theory of non-truthful mechanism design must involve the use of data to parametrize and re-optimize mechanisms. We hope that this work sets the stage for further work in this broad area of research.

There are several specific open problems left unsettled by the work in this thesis. First, a key assumption in the iterated population model is that agents in each stage are distinct, and unable to coordinate their strategies. In may practical applications, e.g. advertising auctions, the same participants play in many stages. Consequently, modifying our approach to accommodate agents who play in many stages is an interesting direction that would make this work more realistic. Additionally, because the stages in many iterated settings occur over time, there is a dynamic or learning aspect to agent's incentives. Consequently, understanding the performance of our mechanisms in learning equilibria, e.g. *Bayes coarse correlated equilibria*, would be another step towards a more realistic model. Indeed, understanding non-truthful mechanisms in Bayesian learning equilibria generally is an under-studied topic which is of great importance to extending the analysis of non-truthful mechanisms beyond delicate analyses of specific equilibria.

In most mechanism design settings, agents have some form of non-linearity in their utility functions: risk-aversion, budgets, or even lexicographic preferences (e.g. Wilkens et al., 2017) are common in practice. Understanding when and if the principles of non-truthful mechanism design laid out in this thesis extend beyond the quasilinear setting is another key direction for a more comprehensive and practical theory of mechanism design.

References

Ausubel, L. M. and Milgrom, P. (2006). The lovely but lonely vickrey auction. *Combinatorial auctions*, pages 17–40.

Balcan, M.-F., Blum, A., Hartline, J. D., and Mansour, Y. (2008). Reducing mechanism design to algorithm design via machine learning. *Journal of Computer and System Sciences*, 74(8):1245–1270.

Bei, X. and Huang, Z. (2011). Bayesian incentive compatibility via fractional assignments. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 720–733.

Borodin, A. and Lucier, B. (2010). Price of anarchy for greedy auctions. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 537–553.

Briest, P. (2008). Uniform budgets and the envy-free pricing problem. In *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008*, pages 808–819.

Chawla, S., Hartline, J., and Nekipelov, D. (2014). Mechanism design for data science. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 711–712. ACM.

Chawla, S. and Hartline, J. D. (2013). Auctions with unique equilibria. In *ACM Conference on Electronic Commerce*, pages 181–196.

Chawla, S., Hartline, J. D., and Nekipelov, D. (2016). A/B testing of auctions. *CoRR*, abs/1606.00908.

Cole, R. and Roughgarden, T. (2014). The sample complexity of revenue maximization. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 243–252. ACM.

Devanur, N., Morgenstern, J., Syrgkanis, V., and Weinberg, S. M. (2015). Simple auctions with simple strategies. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 305–322. ACM.

Devanur, N. R., Huang, Z., and Psomas, C. (2016). The sample complexity of auctions with side information. In *STOC 2016*, pages 426–439.

Dughmi, S., Hartline, J., Kleinberg, R., and Niazadeh, R. (2017). Bernoulli factories and black-box reductions in mechanism design. *arXiv preprint arXiv:1703.04143*.

Dütting, P. and Kesselheim, T. (2015). Algorithms against anarchy: Understanding non-truthful mechanisms. In *16th ACM Conference on Economics and Computation*.

Edelman, B., Ostrovsky, M., and Schwarz, M. (2005). Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. Technical report, National Bureau of Economic Research.

Elkind, E. (2007). Designing and learning optimal finite support auctions. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 736–745. Society for Industrial and Applied Mathematics.

Frankel, A. (2014). Aligned delegation. *The American Economic Review*, 104(1):66–83.

Gonczarowski, Y. A. and Nisan, N. (2017). Efficient empirical revenue maximization in single-parameter auction environments. In *Proceedings of the 49th Annual ACM*

SIGACT Symposium on Theory of Computing, pages 856–868. ACM.

Hartline, J., Hoy, D., and Taggart, S. (2014). Price of anarchy for auction revenue. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 693–710. ACM.

Hartline, J. and Lucier, B. (2010). Bayesian algorithmic mechanism design. In *ACM Symposium on Theory of Computing*, pages 301–310.

Hartline, J. D., Kleinberg, R., and Malekian, A. (2011). Bayesian incentive compatibility and matchings. In *SODA*.

Jackson, M. O. and Sonnenschein, H. F. (2007). Overcoming incentive constraints by linking decisions1. *Econometrica*, 75(1):241–257.

Kaplan, T. R. and Zamir, S. (2012). Asymmetric first-price auctions with uniform distributions: analytic solutions to the general case. *Economic Theory*, 50(2):269–302.

Kirkegaard, R. (2009). Asymmetric first price auctions. *Journal of Economic Theory*, 144(4):1617–1635.

Morgenstern, J. and Roughgarden, T. (2015). On the pseudo-dimension of nearly optimal auctions. In *Annual Conference on Neural Information Processing Systems 2015*, pages 136–144.

Myerson, R. (1981). Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73.

Roughgarden, T. (2009). Intrinsic robustness of the price of anarchy. In *ACM Symposium on Theory of Computing*, pages 513–522.

Roughgarden, T. and Schrijvers, O. (2016). Ironing in the dark. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 1–18. ACM.

Syrgkanis, V. and Tardos, E. (2013). Composable and efficient mechanisms. In *ACM Symposium on Theory of Computing*, pages 211–220.

Varian, H. R. (2007). Position auctions. *international Journal of industrial Organization*, 25(6):1163–1178.

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

Wilkens, C. A., Cavallo, R., and Niazadeh, R. (2017). Gsp: The cinderella of mechanism design. In *Proceedings of the 26th International Conference on World Wide Web*, pages 25–32. International World Wide Web Conferences Steering Committee.

Proof of Lemma 23. Consider drawing $N = mT - 1$ samples from F. Note that the expected value of the quantile of the $(jm-1)$st sample is exactly j/T. Call this value v_{jm-1}, with quantile q_{jm-1}. We will use Chernoff to bound the deviation of q_{jm-1} from its mean in terms of m. Specifically, note that for any $\epsilon \in (0,1)$, the number of samples with quantile at most $j/T + \epsilon$ is the sum of N iid Bernoulli random variables with mean $j/T + \epsilon$. Note that $q_{jm-1} > j/T + \epsilon$ only if at most $jm - 2$ samples overall have quantile at most $j/T + \epsilon$. This latter event is equivalent to the number of samples with quantile at most $j/T + \epsilon$ deviating from its mean (which is $(j/T + \epsilon)(Tm - 1)$) by at least $(j/T + \epsilon)(Tm - 1) - jm - 2 = \left(1 - \frac{jm-2}{(j/T+\epsilon)(Tm-1)}\right)(j/T + \epsilon)(Tm - 1)$. We may bound the probability of this event using Chernoff to obtain:

$$\Pr[q_{jm-1} \geq j/T + \epsilon] \leq e^{-\Omega\left(\left(1-\frac{jm-2}{(j/T+\epsilon)(Tm-1)}\right)^2 (j/T+\epsilon)(Tm-1)\right)}$$

This bound is worst for $j = 1$. Using the bound for $j = 1$ yields:

$$\Pr[q_{jm-1} \geq j/T + \epsilon] \leq e^{-\Omega\left(\left(1-\frac{m-2}{(1/T+\epsilon)(Tm-1)}\right)^2 (1/T+\epsilon)(Tm-1)\right)}$$

A similar approach can bound the probability that $q_{jm-1} < j/T - \epsilon$. This occurs only if at least $jm-1$ samples have quantile at most $j/T-\epsilon$. The latter event is equivalent to the number of samples with quantile at most $j/T - \epsilon$ deviating from its mean of $(j/T - \epsilon)(Tm-1)$ by at least $jm-1-(j/T-\epsilon)(Tm-1) = \left(\frac{jm-1}{(j/T-\epsilon)(Tm-1)} - 1\right)(j/T-\epsilon)(Tm-1)$. Chernoff yields:

$$\begin{aligned}\Pr[q_{jm-1} \leq j/T - \epsilon] &\leq e^{-\Omega\left(\left(\frac{jm-1}{(j/T-\epsilon)(Tm-1)}-1\right)^2 (j/T-\epsilon)(Tm-1)\right)} \\ &\leq e^{-\Omega\left(\left(\frac{m-1}{(1/T-\epsilon)(Tm-1)}-1\right)^2 (1/T-\epsilon)(Tm-1)\right)}.\end{aligned}$$

We may apply a union bound to obtain:

Lemma 56. *Assume the following inequality holds:*

$$(Tm-1)\min\left(\left(1 - \frac{m-2}{(1/T+\epsilon)(Tm-1)}\right)^2 \left(\tfrac{1}{T}+\epsilon\right), \left(\frac{m-1}{(1/T-\epsilon)(Tm-1)} - 1\right)^2 \left(\tfrac{1}{T}-\epsilon\right)\right) \geq \Omega\left(\ln \frac{T}{\delta}\right).$$

Then the probability that $q_{jm-1} \in [j/T - \epsilon, j/T + \epsilon]$ for all $j \in \{1, \ldots, T-1\}$ is at least $1 - \delta$.

This lemma implies Lemma 23.