Optimal Reserve Price Estimation in the Generalized First and Second Price Auctions with Best Response Dynamics

Rigel Galgana
Brown University

Advisor: Amy Greenwald, Computer Science
Secondary Reader: Anastasios Matsavinos, Applied Mathematics
Special Thanks: Takehiro Oyakawa, Enrique Areyan Viqueira
May 22, 2020

Abstract

The generalized first and second price auctions (GFP and GSP) are the primary means through which ad auctions are conducted. Several theoretical results characterizing bidding behavior, expected revenue, and optimal reserve price under both mechanisms have been shown to hold under assumptions of valuation distribution symmetry and monotone hazard rate (MHR). From the standpoint of evolutionary game theory, the stability (or uniqueness) of these equilibria are not well understood. That is, assuming that agents are expected utility maximizing, these equilibria may be unattainable using best response dynamics. In this paper, we seek to computationally analyze the properties of these equilibria. Furthermore, we attempt to recover similar characterizations of bidding behavior, expected revenue, and optimal reserve price without the additional assumptions of symmetry and MHR. To that end, we introduce fast algorithms to compute conditional and joint distributions of order statistics which inevitably play a role in rank-based mechanisms such as the GSP and GFP.

Contents

1 Introduction

Characterizing the expected revenue of auctions from a computational standpoint has been of great interest, especially in recent years. Many ordinary transactions such as those in e-commerce [7], internet service plans [14], advertisements, airplane tickets [12], hotel reservations [8] among others are indeed designed with profit in mind. Auctioneers consider past transactions, demand, reliability, necessity, budget, and other characteristics indicative of one's willingness to pay—their valuation—to determine appropriate prices. Profit optimization is straightforward in markets in which individuals are price takers as the auctioneer need not consider strategic behavior amongst the participants. Markets with either weak or no competition (monopolies) or a large number of participants (high-demand) tend to give auctioneers more freedom in price-discrimination and profit maximization. In other situations, however, the auctioneer has less market presence and participants exhibit strategic behavior that non-trivially impacts expected revenue. This is particularly true in the ad auction setting [4]. Much of recent research focuses on methods to characterize the expected revenue of these non-truthful auctions through sample complexity guarantees [3], game-theoretic solution concepts [6] [10], approximation algorithms [11].

The design and analysis of ad auctions, also known as sponsored search, is a particularly hot area of research. In these auctions, companies pay advertisers to display their advertisements given a particular search query. That is, every time a user queries, say Google, a micro-auction between companies interested in targeting the user and similar demographics is held. This auction determines not only which ads are shown but also how attractive of a location on the search page they are placed. The auction's participants then pay some fee based on what they are allocated, whether the user clicked on the ad, and if the user purchased anything. Determining this payment scheme has been of significant computational and economic interest. Historically, Generalized Second Price auctions (GSPs), in which the auction's winners pay the next highest bid, have been used, most notably by Yahoo! and Google. In the past, Yahoo! switched from Generalized First Price (GFP) to GSPs, and by the end of 2019, Google switched from GSP to the GFP, in which participants pay their own bid. As most of the theory regarding sponsored search auctions has revolved around GSPs, GFPs are not as well understood and bidder behavior is often unpredictable [4]. Nonetheless, several results characterizing Nash equilibria, expected revenue, and optimal reserve prices in both GFP and GSP, from both the ex-ante and full-information settings, have been shown in recent years [6] [10] [9]. In addition to using standard optimization and auction-theoretic tools, such as Myerson's theorem and revenue equivalence, many of these derivations call upon distributions of order statistics.

To that end, order statistics—ordered random variables—naturally arise from the rank-based allocation rules of auctions, GFPs included. The expected utility of a bidder and the expected revenue of the GSP/GFP can be expressed as a function of order statistics and their distributions. For instance, consider the expected revenue of the GFP with no reserve and m slots. Let there be n bidders with bids $b_i \sim F_{b_i}$ for all $i \in [n]$ such that $b_i \perp b_j$. Define $b_{(i)}$ to be the ith largest value in $\mathbf{b} = (b_1, \ldots, b_n)$. The expected revenue can be expressed as the sum of the expected values of the top m order statistics:

$$\mathbb{E}_{\mathbf{b}}(\mathrm{Rev}_{n,GFP}) = \mathbb{E}_{\mathbf{b}}\left(\sum_{i=1}^{m} b_{(i)}\right) = \sum_{i=1}^{m} \mathbb{E}_{b_{(i)}}\left(b_{(i)}\right) \tag{1}$$

At first glance, this expression seem relatively simple to compute. However, there are two issues which makes characterizing the expected revenue more difficult. The first is that we have direct access to the bidder behavior. The bidding distributions F_{b_i} are actually functions of some underlying valuation distribution and the mechanism type and parameters. The second issue is that the distribution of order statistics of non-identically distributed random variables is difficult to compute. In particular, while the Bapat-Beg theorem [1] provides an explicit expression for the joint cumulative distribution functions of order statistics, however, is costly to evaluate. We require alternative methods to quickly compute the quantity $\mathbb{E}_{b_{(i)}}\left(b_{(i)}\right)$ as well as more general expectations of functions of multiple order statistics as we will need to for GSP. In the i.i.d. case—an assumption of most of the aforementioned results regarding the Nash equilibria, expected revenue, and optimal reserve price—there is a simple expression for single order statistic distributions. More concretely, consider computing the distribution of the ith order statistic $X_{(i)}$ of the random variables $X_1, \ldots, X_n \overset{i.i.d.}{\sim} F$. The order statistic cumulative distribution function (cdf) denoted by $F_{X_{(i)}}$ requires a summation over a set of size $O(n)$ and the pdf is easy to compute:

$$F_{X_{(i)}}(x) = \sum_{j=i}^{n} \binom{n}{j} F(x)^j (1 - F(x))^{n-j} \tag{2}$$

$$f_{X_{(i)}}(x) = n \binom{n-1}{i-1} F(x)^{i-1} (1 - F(x))^{n-i} f(x) \tag{3}$$

We can then compute the expected value as $\mathbb{E}(X_{(i)}) = \int_{-\infty}^{\infty} x f_{X_{(i)}} dx$. If the X_i's are non-negative which they are in the case of auction bids, $\mathbb{E}(X_{(i)}) = \int_0^{\infty} d\tilde{F}_{X_{(i)}}(x)$, where $\tilde{F}$ denotes the complementary cdf. Approximating these quantities computationally is not too difficult as $f_{X_{(i)}}$ and $F_{X_{(i)}}$ are simple. Unsurprisingly, the expression becomes more complicated when dropping the identically distributed assumption. Letting each $X_i \sim F_i$ denote n independent but not necessarily identically distributed random variables, then we have:

$$F_{X_{(i)}}(x) = \sum_{j=i}^{n} \sum_{\mathcal{S} \in 2^{\{1,\ldots,n\}}, |\mathcal{S}|=j} \left[\prod_{s \in \mathcal{S}} F_s(x) \prod_{s' \in \{1,\ldots,n\} \setminus \mathcal{S}} (1 - F_{s'}(x)) \right] . \tag{4}$$

The number of terms in this expression is exponential with respect to n. The pdf $f_{X_{(i)}}$ is similarly difficult to compute. Furthermore, we have only computed the distribution of a single order statistic under the independence assumption; computing the joint order statistics distribution without the independence assumption is even more difficult. For our purposes, we will always assume independence, though there are certainly scenarios in which this does not hold, such as when there is shared information amongst some agents or when agents bid in order. However, in some of the mechanisms we consider in this paper, the expected revenue is a function of the *joint* distribution. A similar analysis can be made for computing the expected utility of a bidder participating in one of these auctions with fixed competitor bidding distributions. Some of our primary contributions in this paper are an efficient algorithms to compute joint order statistic distributions, expectations, and conditional expectations which will allow for both analysis and optimization of several single-parameter auctions.

One particular application allows us to simulate best response dynamics (BRD) over the (discretized) space of bidder behavior without having to simulate individual auctions. Interpreting learning expected utility maximizing behavior as a dynamical system over bidding strategies induced by best response dynamics, we can interpret Nash/Bayes-Nash equilibria as long-term behavior of the agents. As we seek to determine the stability of the equilibria as predicted in theory, the additional noise in Monte Carlo estimation—which is the primary means in which several papers have attempted to learn equilibria in auctions [15] [16]—complicates locating unstable equilibria. This necessitates exact computations of expected utility, which we have previously mentioned may be calculated as a function of distribution of order statistics.

The paper is outlined as follows: In Section 2, we introduce relevant notation and terminology. In Section 3, we formally define the two auctions that we will be analyzing or optimizing. In Section 4, we solve the problem of computing joint or conditional order statistic distributions through dynamic programming. In Sections 5, we apply our algorithm to each of the auctions defined in Section 3. We conclude Section 6, giving possible extensions for the work done in this paper.

2 Preliminaries

2.1 Notation

While we will be restricting our attention to the GSP and GFP for most of the paper, we will give a more general exposition of notation as our methodology works for more general rank based auctions. Recall that our objective is to either compute or optimize the expected revenue of a single-parameter auction $\mathcal{A}_\theta$ with n bidders with respect to the auction parameters θ. Each of these bidders have independent valuation distributions $F_1, \dots, F_n$. Let $v_i \sim F_i$ denote the valuation of bidder i. The auction proceeds as follows: there are m goods to be auctioned off and each bidder i submits a bid $b_i(v_i, \mathcal{A}_\theta) = b_i \in \mathbb{R}^+$. Depending on agent i's bid b_i, all other agents' bids $\mathbf{b}_{-i}$, and the auction's allocation rule, the ith agent will be assigned one of the goods and receives utility $\mu_i(b_i, v_i, \mathbf{b}_{-i}, \mathcal{A}) = v_i x_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta) - p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)$. Here, $v_i x_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)$ is one's valuation v_i multiplied by their allocation and $p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)$ is the payment of bidder i, which is some function of their own bid, all other agents' bids, and the auction parameters. The expected revenue is then a function of several objects: the payments $\mathbf{p}$, bid distributions $F_{b_1|\mathcal{A}_\theta}, \dots, F_{b_n|\mathcal{A}_\theta}$, and the auction $\mathcal{A}_\theta$. These bid distributions F_{b_i} are obtained as a transformation from valuation space to bid space $b_i = b_i(v_i, \mathcal{A}_\theta) \sim F_{b_i|\mathcal{A}_\theta}$ through the bid function. With this, we have the following two expression for the expected revenue:

$$\mathbb{E}_{\mathbf{v}}(\mathrm{Rev}_{n,\mathcal{A}_\theta}) = \mathbb{E}_{\mathbf{v}}\left(\sum_{i=1}^{n} p_i(b_i(v_i), \mathbf{b}_{-i}(\mathbf{v}_{-i}), \mathcal{A}_\theta)\right) = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{b}|\mathcal{A}_\theta}\left(p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)\right) \tag{5}$$

Intuitively, the revenue of an auction $\mathcal{A}_\theta$ and some vector of bids $\mathbf{b}$ is exactly the sum of the payments of each of the bidders. The expression seems relatively straightforward to evaluate, and indeed, if we are given access to the bidding distributions, the expected revenue can either be computed directly or estimated via Monte Carlo simulation. Now assuming that we can compute the expected revenue of a particular auction, we may be able to optimize some parameters of an auction, such as reserve or posted prices. At first glance, this seems to be a calculus problem of solving for the zeros of the derivative of the expected revenue over the set of auction parameters. In the i.i.d. case, this does not appear to be too difficult as there exist simple expressions to compute joint order statistic distributions [5]. However, we have overlooked a crucial hurdle: the bidding distributions $F_{\mathbf{b}|\mathcal{A}_\theta}$ are in general not given, and moreover, are not trivial to characterize even in the i.i.d. setting.

2.2 Solution Concepts

The bidding functions b_i can be arbitrary mappings from a valuation to some real value. We will assume that these bidding functions are rational where each agent i with valuation v_i will bid b_i in order to maximize their expected utility:

$$b_i = b_i(v_i, \mathcal{A}_\theta) = \mathrm{argmax}_b \mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}\left(\sum_{j=1}^{m} \mu_i(b_i, \mathbf{b}_{-i}, \mathcal{A})\right) \tag{6}$$

As mentioned, these bidding functions may not necessarily be uniquely determined. We can interpret the auction as a game, where there are n players with action/strategy space as the set of all possible maps from valuations to bids and reward equal to expected utility as in Equation (6). Now consider this game's set of Nash equilibria—vectors of strategies such that each agent cannot improve their expected utility by deviating from their current strategy. However, we are guaranteed neither existence or uniqueness of Nash equilibria which spells trouble for our computation of expected revenue in Equation (5). However, not all is lost. Nash equilibrium is not the only solution concept or characterization of rational agent behavior. For example, Bayes-Nash equilibria generalizes the notion of Nash equilibria by defining a probability distribution over actions such that one modifying these probabilities cannot improve one's expected utility. Similarly, from the lens of evolutionary game theory, one can define how these agents learn their bidding distribution as some deterministic function f. These functions define a dynamical system (induce a response graph if discrete) over the set of all possible vectors of strategies. One can then define a function that maps from the dynamical system (response graph) to a single, representative vector of strategies. For example, [13] defines a Markov chain over and within the set of stable limit cycles and defines the representative vector as the stationary distribution of this Markov chain. Similarly, [17] applies the same procedure except considering only the recent history of best-response vectors. In this paper, we will run BRD from random initializations for a set number of trials and average the last few entries to determine long-term bidder behavior. This allows our reserve price optimization to apply more generally as the problem of bidding behavior determination can be treated as a separate problem. While there are certainly many other useful solution concepts that allow us to uniquely determine the bid distributions, we will only consider fixed time BRD and fixed time Boltzmann weighted BRD where action probabilities are weighted exponentially with respect to their expected utility.

2.3 Learning Bid Functions

As we have noted, the auction can be seen as a game with n agents, seeking to find an optimal bidding function over the set of valuations. Constructing agents to learn these mappings is difficult as the support of each F_{v_i} may be unbounded or a continuum. There are several theoretical results that characterize the existence of the GFP and GSP's symmetric Bayes-Nash equilibrium (SBNE) with the additional assumptions of i.i.d. valuations and monotone hazard rate valuation distributions. Letting b_i, f, and F denote the universal bidding function and valuation distribution and cumulative density function, then the following two bidding behaviors constitute a symmetric BNE in the GSP and GFP respectively:

(Gomes, Sweeney 2014): If a quasi-efficient, symmetric BNE exists in GSP, then:

$$b_i(v) = v - \sum_{s=2}^{m} I_{v \geq p} \gamma_s(v) \int_0^v (v - \max(b(x), p))\, F^{n-s-1}(x) f(x) dx \tag{7}$$

$$\text{where } \gamma_s(v) = \frac{\binom{n-2}{s-1}(s-1)(1-F(v))^{s-2}\alpha_s}{\sum_{t=1}^{s} \binom{n-2}{t-1}(1-F(v))^{t-1}F^{n-t-1}(v)\alpha_t} \tag{8}$$

(Han, Liu 2015): If a quasi-efficient, symmetric BNE exists in GFP, then:

$$b_i(v) = v - \frac{\sum_{s=1}^{m}(\alpha_s - \alpha_{s+1}) \int_p^v F_{(s)}(t)dt}{\sum_{s=1}^{m}(\alpha_s - \alpha_{s+1})F_{(s)}(v)} \tag{9}$$

In terms of the dynamical system induced by best response dynamics, it is unknown whether or not these equilibria are stable. That is, these BNE may be not necessarily be attainable from bidders seeking to maximize their expected utility. To that end, one of our major goals is to determine the stability of these equilibria computationally. One of the hurdles is that the expected utility function seems difficult to calculate. However, two useful observations come to our rescue:

1. As seen in Equation (5), one need not actually consider the mappings from valuations to bids in order to compute the expected revenue of an auction. Instead, we only consider the distribution of bids induced by the bidding function and valuation distributions. This implies that in order for the ith agent to learn a bidding function, they need not consider the bidding function of other bidders, but just their bid distributions.

2. The expression in Equation (6) is similar to Equation (5). In the former, we compute the expected value over all bids of some function dependent on these bids. In the latter, we compute the expected value over all but the ith bid of some other function dependent on these bids. We can use compute the distributions of the relevant order statistics via dynamic programming to determine the expected utility quickly.

With these observations in mind, we define a simple, discretized one-step optimization function that obtains both the bidding function and bid distribution for a particular bidder, fixing all other agents' bid distributions. More specifically, we first discretize the valuation distribution F_{v_i} (if it is not already discrete) into probability mass functions over t points on the real line, $r_1 < \ldots < r_t$. The mass of the discretized valuation distribution at each point r_s is proportional to $F_{v_i}(r_s) - F_{v_i}(r_{s-1})$ for $s = 1, \ldots, t$, with $r_0 = -\infty$. Then, at each point $v_i = r_s$, we compute the expected utility of bidding $b_i = r_{s'}$ to be $\hat{\mu}_i(r_{s'}; r_s, \mathcal{A}_\theta)$. We then set the bidding function at point $v \in (r_{s-1}, r_s]$ to be the bid that maximizes $\hat{\mu}_i(\cdot, r_s)$ and update the bid distribution accordingly.

Algorithm 1 Discretized One-Step Bidding Function Learner

Require: $\mathcal{A}_\theta, F_{v_i}, F_{\mathbf{b}_{-i}}, (r_1, \ldots, r_t), \mu_i$
Ensure: Approximate bid function and bid distribution, $\tilde{b}_i(\cdot)$ and $\tilde{F}_{b_i}(\cdot)$ respectively

```
Initialize F̂_{b_i}(·) = 0
for s ← 1 to t do
    for s' ← 1 to t do
        Set μ̂_i(r_{s'}; r_s, A_θ) = E_{b_{-i}|A_θ} ( Σ_{j=1}^m μ_i(r_{s'}, b_{-i}, A; r_s) )
    end for
    Set b̃_i(v) = argmax_b μ̂_i(b; r_s, A_θ) for all v ∈ (r_{s-1}, r_s]
    Set F̃_{b_i|A_θ}(r_s) += F_{v_i}(r_s) − F_{v_i}(r_{s-1})
end for
Set b̃_i(v) = argmax_b μ̂_i(b, r_t, A_θ) for all v ∈ (r_t, ∞)
Set F̃_{b_i|A_θ}(r_t) += 1 − F_{v_i}(r_t)
return b̃_i(·), F̃_{b_i|A_θ}(·)
```

This can be seen as learning a best response to the current set of bidding distributions. We run this procedure for each agent (in some randomized fashion to avoid possible cycles), updating the bidding distributions and functions until the agents have reached equilibrium. There is one primary advantage and one disadvantage of programming agents in this fashion to compute bidding functions rather than using one of the solution concepts as mentioned last section. The advantage is that we can start from any initialization to show the stability of a particular BNE (attraction of a particular fixed point of a dynamical system). With this, we can 'jump-start' the mixing process by saving previous bidding functions and modify them slightly each time the auction parameters or bid distributions are updated. Hopefully, this initialization will allow the agents to converge more quickly. However, the main disadvantage is that the bidding behavior over all agents are not guaranteed to converge as there may be bidder behavior cycling. We will simulate best response dynamics in our experiments sequential, simultaneous, and weighted learning updates. We will verify in which auctions these agents settle into an equilibrium and those in which they do not.

2.4 Auctions

Having obtained the empirical bidding distributions, we are almost fully equipped to compute or optimize the expected revenue of any auction. The question now is which auctions we will be analyzing or optimizing. Auctions are defined by an allocation and payment scheme which are defined by key parameters such as posted or reserve prices, the number of goods m, or a tie-breaking index. Here, we describe the three auctions of interest—the GFP, GSP, and Myerson's optimal auction—and then compare the expected revenue of each of these auctions.

2.4.1 Generalized First and Second Price Auctions

GSPs, along with generalized first price mechanisms (GFP), are the prevalent mechanisms for sponsored search auctions. In the sponsored search setting, there are m items—slots—with click through rates $\alpha_{i,1} \geq \ldots \geq \alpha_{i,m}$ for each bidder i. Slots are allocated in decreasing order of bids. Bidder i obtain payoff $\alpha_{i,j}v_i$ and pays b_i if allocated the jth slot. The expected revenue of the GFP without reserve is then the expected value of the sum of the m largest bids. Similarly, in the GSP, the expected revenue is the sum of the second through $m+1$st largest bids. In practice, both auctions are defined with a reserve price π. In GFPs and GSPs with reserve, the reserve price π enforces that bidders must bid at least π to be considered at all for slot allocation. In the GSP, the winners then pay the maximum of the reserve price and the next highest bid. These reserve prices serve to both avoid the risk of receiving unusually low revenue from the auction winners and to incentivize bidders not to shade their bids too much. The expected revenue of these auctions is then the sum of the expected values of the m largest bids subject to the reserve price: $\sum_{i=1}^{m} \mathbb{E}\left(b_{(i)} 1_{b_{(i)} \geq \pi}\right)$. These reserve prices can be used to optimize the expected revenue of an auction, such as in Myerson's revenue maximizing auction. Under the theoretical SBNE as described earlier, the expected revenue of the GFP and GSP can be computed explicitly under a given reserve price π:

(Gomes, Sweeney 2014): Given reserve price π, the expected revenue of the GSP under symmetric equilibrium is given by:

$$\mathbb{E}_{\mathbf{b}}\left(\text{Revenue}(\mathbf{b}) \mid \text{GSP}(\pi)\right) = n \sum_{s=1}^{m} \int_{\pi}^{\infty} \alpha_s \binom{n-1}{s-1} (1-F(t))^{s-1} F(t)^{n-s} \left(t - \frac{1-F(t)}{f(t)}\right) dF(t)$$

(Han, Liu 2015): Let $F_{(s)|n-1}(x)$ denote the CDF of the sth order statistic of $n-1$ i.i.d. random variables drawn from F. Given reserve price π, the expected revenue of the GFP under symmetric equilibrium is given by:

$$\mathbb{E}_{\mathbf{b}}\left(\text{Revenue}(\mathbf{b}) \mid \text{GFP}(\pi)\right) = n\sum_{s=1}^{m}(\alpha_s - \alpha_{s+1})\left[\int_{\pi}^{\infty} t(1-F(t))dF_{(s)|n-1}(t) + \pi(1-F(\pi))F_{(s)|n-1}(\pi)\right]$$

Interestingly enough, the optimal reserve price in both auctions is the same under these SBNE:

$$\pi^* = \frac{1-F(\pi^*)}{f(\pi)} \tag{10}$$

Where f, F, and $\alpha_1, \ldots, \alpha_m$ denote the universal valuation distribution, cumulative distribution functions, and click through rates respectively. It is currently unknown how these results generalize to the asymmetric setting (valuations or CTRs) or without the assumption of MHR. Fortunately, there exists another auction to which we can compare expected revenue in this more general setting.

2.4.2 Myerson's Revenue Maximizing Auction

Myerson's revenue maximizing auction is a truthful auction in which players are allocated in decreasing order of non-negative virtual valuations ϕ_i where $\phi_i(v_i) = v_i - \frac{1-F_{v_i}(v_i)}{f_{v_i}(v_i)}$. Myerson showed that the expected revenue is equal to the expected virtual surplus, which is the sum of all winners' virtual valuations. Hence, with appropriate bidder behavior, this allocation rule is expected revenue maximizing. The question now is how to enforce 'appropriate bidder behavior', as in general, the auctioneer only sees bids rather than valuations or virtual valuations. In a separate result, Myerson also provides an expression to compute the payment formula under any particular monotone allocation rule such that the resultant auction is truthful. In the sponsored search setting, the payment for the winner of the kth item, say bidder i and $j < m$, comes out to be $\sum_{j=k}^{m} \phi_i^{-1}(\phi_{j+1})(\alpha_{j+1} - \alpha_j)$, where ϕ_{j+1} is the $j+1$st largest non-negative virtual valuations. Similarly, the winner of the mth item, say bidder i, then pays the reserve $\pi_i = \phi_i^{-1}(0)$. The virtual valuation order allocation rule and this payment scheme make this auction revenue maximizing over the set of all truthful auctions. One of the additional nice properties of the auction, since it is truthful, is that we have equivalence between expected revenue and expected virtual surplus. This allows us to compute the expected revenue as the sum of the m largest expected virtual valuations subject to non-negativity, which themselves can be treated as random variables. Using the same algorithm to compute order statistic distributions as in the GFP, we can back out the expected virtual surplus of Myerson's auction.

Despite the desirable properties of Myerson's auction—DSIC and expected revneue maximizing over the set of truthful auctions—it is rarely used in practice. The complex allocation and payment schemes involve not only one's own bid but also the virtual valuations of *other* bidders. We will only use this auction as a baseline to gauge the effectiveness of the GFP and GSP in our experiments. Now, we are ready to introduce the main auctions of interest and the algorithm through which we shall analyze them.

3 Problem Statement

As we have mentioned previously, the generalized first and second price auctions are the primary mechanisms of interest. We will be maximizing the expected revenue with respect to the reserve price π for which the bidders must meet in order to be considered in the auction. In particular, we will be comparing expected revenue from the non-truthful GFP and GSP with empirically optimal reserve price with that of Myerson's auction, which is provably expected revenue maximizing among the set of all truthful auctions. Our overall optimization procedure can be partitioned into four stages: we randomly initialize some reserve price $\theta = \pi$ and obtain the bidder distributions according to our bidding function learning algorithm. Then, we compute the expected revenue associated with this particular reserve price and corresponding bid distributions. We then obtain a proposal parameterization $\theta_{t+1} \in \Theta$ and repeat until some stopping condition and return the optimal reserve price π. We repeat these last three steps for different initializations to escape local optima. More generally, this procedure works for any auction except replacing the reserve price π with a different set of parameters:

Algorithm 2 General Auction Parameter Optimization Procedure

Require: $\mathcal{A}, (F_{v_i})_{i\in[n]}, (\alpha_{i,j})_{i\in[n],j\in[m]}, \Theta$

Ensure: θ^*, the revenue maximizing parameterization and R^*, the associated expected revenue

1: **for** randomly initialized θ^0 until termination **do**
2:     **for** proposal θ^t until termination condition **do**
3:         $F_{\mathbf{b}^t} \leftarrow F_{\mathbf{b}|\mathcal{A}_{\theta^t}}$
4:         Set $R_{\mathcal{A}_{\theta^t}} = \mathbb{E}_{\mathbf{b}^t}\left(\mathrm{Rev}(\mathbf{b}; \mathcal{A}_{\theta^t})\right)$
5:         **if** $R_{\mathcal{A}_{\theta^t}} \geq R^*$ **then**
6:             Set $R^* \leftarrow R_{\mathcal{A}_{\theta^t}}$
7:             Set $\theta^* \leftarrow \theta^t$
8:         **end if**
9:         $\theta^{t+1} \leftarrow \mathrm{ProposalFunction}(\theta_{0:t}, R_{0:t})$
10:     **end for**
11: **end for**
12: **return** θ^* and R^*.

It is worth noting that step (4) in the algorithm above computes the resultant bid distributions corresponding to the behavior simulated by our agents through Algorithm (1). These distributions need not be obtained through simulated agents and instead can be replaced with the behavior predicted by other solution concepts as mentioned previously. We have also already explained the merits of using our simulated agents. Similarly, the parameter updating step in line (10) we choose to be the expected improvement maximizing parameterization assuming that expected revenue is a Gaussian Process. Any general purpose local search proposals should suffice though. After applying this procedure, we will be ready to answer the following key questions: How do the expected revenues in the optimized GFP and GSP obtained via BRD compare to those as predicted by theory? Similarly, can we characterize changes in bidder behavior with respect to the reserve price, number of bidders, number of slots, or valuation distributions? Most importantly, can we cover the optimal reserve price?

4 Algorithms

The only step in Algorithm (2) we have yet to define how to do is step step 5, where we obtain the expected revenue of the auction with parameter θ^t; namely $R_{\mathcal{A}_{\theta^t}} = \mathbb{E}_{\mathbf{b}^t}\left(\mathrm{Rev}(\mathbf{b}; \mathcal{A}_{\theta^t})\right)$. For the generalized first (second) price auction with m slots and reserve price π, the expected revenue is precisely the expected sum of the m largest bids above π (sum of the maximum of π and the 2nd through $m+1$st largest bids above π). Using linearity of expectations, we obtain the following formula for the expected revenue of the GFP with reserve price π and bid distributions $F_{\mathbf{b}^t}$:

$$\mathbb{E}_{\mathbf{b}^t}(\mathrm{Rev}_{n,GFP(\pi)}) = \sum_{i=1}^{m} \mathbb{E}_{b^t_{(i)}}\left(b^t_{(i)} 1_{b^t_{(i)} \geq \pi}\right) \tag{11}$$

There are two main differences between this auction and that as defined in the introduction in Equation (1). In the latter, there is no reserve price, so our expectation is over all $b_{(i)}$ rather than just $b^t_{(i)} 1_{b^t_{(i)} \geq \pi}$. Assuming that $b_{(i)} \stackrel{d}{=} b^t_{(i)}$, then the GFP with no reserve performs better. Of course, this is not the case, as the reserve incentivizes bidders to shade their valuations less. Exactly how much better we will be able to see empirically. One major hurdle remains as computing the expectation of a function of an order statistic of non-i.i.d. random variables is non-trivial. We run into a similar problem when computing the expected revenue of Myerson's auction when treating the virtual valuations themselves as random variables.

$$\mathbb{E}_{\mathbf{b}}\left(\mathrm{Rev}_{\mathrm{Myerson}}\right) = \mathbb{E}_{\phi_1,\ldots,\phi_n}\left(\sum_{i=1}^{m} \phi_{(i)} 1_{\phi_{(i)} \geq 0}\right) \tag{12}$$

The distribution of the ϕ_i's can be backed out using f_i and F_i's and similarly, the expectation of of $\phi_{(i)} 1_{\phi_{(i)} \geq 0}$ requires obtaining the distribution of individual order statistics as in the GFP. The expected revenue in the GSP is even more complex as the payments are functions of multiple order statistics:

$$\mathbb{E}_{\mathbf{b}^t}(\mathrm{Rev}_{n,GSP,\pi}) = \sum_{i=1}^{m} \mathbb{E}_{b^t_{(i)}, b^t_{(i+1)}}\left(\max(b^t_{(i+1)}, \pi) 1_{b^t_{(i)} \geq \pi}\right) \tag{13}$$

In addition to computing the expected revenue, the expected utility of a bidder requires computing joint order statistic distributions of independent but non-necessarily identically distributed random variables:

$$\mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}\left(\mu_i(b_i; v_i, \mathcal{A}_\theta)\right) = \mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}\left(v_i x_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)\right) - \mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}\left(p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)\right) \tag{14}$$

$$= \sum_{j=1}^{m} v_i \mathbb{P}(\text{win slot } j \text{ with bid } b \mid F_{\mathbf{b}_{-i}})\alpha_j - \mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}\left(p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)\right) \tag{15}$$

In GFP, the expected payment is $\mathbb{P}(\text{win any item}) b_i$. In GSP, the expression is more complicated:

$$\mathbb{E}_{\mathbf{b}_{-i}|\mathrm{GSP}(\pi)}\left(p_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)\right) = \sum_{j=1}^{m} \mathbb{P}(\text{win slot } j \text{ with bid } b) \mathbb{E}_{\mathbf{b}_{-i}|\mathrm{GSP}(\pi)}\left(\max(\pi, b_{(j+1)}) \mid b_{(j+1)} \leq b\right) \tag{16}$$

To that end, we will need algorithms that can compute the expected revenues, payoffs, and payments in the GFP, GSP, and Myerson settings quickly. We first describe an algorithm that obtains joint cumulative distribution functions of order statistics of non-identically distributed random variables. In particular, let $X_i \sim F_i$ for all $i \in [n]$ be a collection of independent random variables. We wish to compute the joint cumulative distribution function of the order statistics indexed by $\mathcal{C} = (c_1, \dots, c_d) \in \mathbb{N}^d$ at a point $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$:

$$F_{X_\mathcal{C}}(\mathbf{x}) = \mathbb{P}\left(\bigcap_{j=1}^{d} \{X_{(c_j)} \leq x_j\}\right) \tag{17}$$

Here, we assume both $\mathcal{C}$ and $\mathbf{x}$ are strictly increasing; otherwise some indices may be redundant i.e. $\mathbb{P}(X_{(1)} < 1, X_{(2)} < 0) = \mathbb{P}(X_{(2)} < 0)$. The joint cumulative distribution function will be useful for computing the expected utility of a particular bidder. Computing the expected revenue of Myerson's optimal auction also requires a similar expression, though requires some modification to account for virtual valuations. To compute the expected revenue of GFP and Myerson's auction, we will need the expected value of individual order statistics. Fortunately, the expected value of a non-negative random variable can be expressed in terms of the complementary cumulative distribution function, which we are going to compute anyways. That is, for non-negative random variable X with F:

$$\mathbb{E}(X) = \int_0^\infty (1 - F(x))dx \tag{18}$$

4.1 Related Work

4.1.1 Bapat-Beg Theorem

The Bapat-Beg theorem [1] gives an explicit formula for this quantity, which involves the computation of matrix permanents. Letting $\mathcal{I} \equiv \{\mathbf{i} : 0 = i_0 \leq i_1 \leq \ldots \leq i_d \leq i_{d+1} = n, i_j \geq c_j \;\; \forall j = 1, \dots, d\}$, we have

$$F_{X_\mathcal{C}}(\mathbf{x}) = \sum_{\mathbf{i} \in \mathcal{I}} \frac{\mathrm{Per}_{i_1,\dots,i_d}(X_1, \dots, X_d)}{\prod_{j=1}^{d+1}(i_j - i_{j-1})!} \tag{19}$$

Here, $\mathrm{Per}_{i_1,\dots,i_d}(X_1, \dots, X_d)$ is the permanent of the block matrix $\mathcal{P}$ where $\mathcal{P}_{i,j} = F_i(X_t) - F_i(X_{t-1}) = p_{i,t}$ for $i, j = 1, \dots, n$ and $t = \min_s(\{i_s : i_s \geq j\})$. While this seems to solve our problem, the $\mathcal{I}$ grows at rate $O(n^d)$ and the fastest known procedure to compute the matrix permanent is $O(n2^{n-1})$. Even for small n and d, this procedure becomes intractable quickly.

4.1.2 Boncelet Jr. Dynamic Programming Algorithm

An improvement to this result using dynamic programming was posed in [2]. This method converts (17) into an equivalent combinatorial problem as follows.

1. We begin by defining $x_0 = -\infty$ and partition the region $(-\infty, x_d]$ into n intervals—which we will henceforth refer to as bins:

$$(-\infty, x_d] = \cup_{j=1}^{d}(x_{j-1}, x_j] = \cup_{j=1}^{d} I_j = I_{1:d} \tag{20}$$

2. Define $C_j^i = \sum_{k=1}^{i} 1_{X_k \in I_j}$ to be the number of the first i random variables $X_1, \ldots, X_i$—which we refer to as balls—that land in bin j. Let $\mathbf{C}^i = (C_1^i, \ldots, C_d^i)$ be the ball configurations of the first i balls respectively.

3. Define $D_j \equiv \{\sum_{i=1}^{j} C_i^n \geq c_j\} \equiv \{C_{1:j}^n \geq c_j\}$ to be the event that there are at least c_j balls in the first j bins. We can now rewrite Equation (17) as:

$$F_{X_\mathcal{C}}(\mathbf{x}) = \mathbb{P}\left(\bigcap_{j=1}^{d}\{X_{(c_j)} \leq x_j\}\right) = \mathbb{P}\left(\bigcap_{j=1}^{d}\{C_{1:j}^n \geq c_j\}\right) = \mathbb{P}\left(\bigcap_{j=1}^{d} D_j\right) = \mathbb{P}(D_{1:d}) \tag{21}$$

4. We define $n+1$ tables $T_0, \ldots, T_n$ where $T_i \in [n]^d$ such that $T_i(\mathbf{C})$ is the probability that $\mathbf{C}^i = \mathbf{C}$. With T_0 to be a table of 0's, Boncelet's algorithm then employs the following recurrence relationship:

$$T_i(\mathbf{C}) = \mathbb{P}\left(X_i > x_d\right) T_{i-1}(\mathbf{C}) + \sum_{\{j:C_j>0\}} \mathbb{P}(x_{j-1} < X_i \leq x_j) T_{i-1}(\mathbf{C} - \mathbf{1}_j) \tag{22}$$

5. The probability in Equation (21) can then be obtained from table T_n by summing over the entries $\mathbf{C}$ that satisfy $D_{1:d}$.

As there are n iterations, updating $O(n^d)$ table entries, where each update requires a summation over $O(d)$ elements, this algorithm has a time complexity of $O(dn^{d+1})$ and a space complexity of $O(n^{d+1})$, though the space complexity can be improved by a factor of n by discarding previous tables which we are no longer needed. Furthermore, this algorithm has an additional advantage over using the Bapat-Beg theorem as the table entries themselves are meaningful. For example, we can run the Spill-Over algorithm when $n < d$. Note that this makes the original joint distribution computation impossible, as we cannot possibly satisfy $D_{1:d}$ with only $n < d$ variables. However, applying the recurrence relation with $\mathcal{C} = (1, \ldots, d)$ and $\mathbf{x} = (x_1, \ldots, x_d)$, the table T represents the joint probability mass function of these n variables over the d intervals $I_1, \ldots, I_d$ defined by $\mathbf{x}$. Then, $T_n(\mathbf{C})$ represents the probability that there are C_j random variables that realized value in I_j. Letting d be very large, we can approximate the joint cumulative density function of these points. That being said, while algorithm is much faster than applying Bapat-Beg's theorem, the space complexity of $O(n^d)$ makes the algorithm difficult to use in practice. Our algorithm, which we dub the Spill-Over algorithm, builds off Boncelet Jr.'s dynamic program to reduce its time and space complexities.

4.2 Spill-Over Algorithm

As just mentioned, the Spill-Over algorithm is a variant of Boncelet Jr.'s algorithm. In particular, it is a compressed version of Boncelet Jr.'s algorithm. Previously, we defined ball count and configuration variables C_j^i and $\mathbf{C}^i$ respectively which denoted the *exact* number of balls in a particular bin. In this algorithm, we instead construct lower bounds B_j^i and $\mathbf{B}^i$ on the number of balls in the bins. Note that the event D_j still holds as long as $\sum_{i=1}^{j} B_i^n \geq c_j$. To more precisely show how much this algorithm improves upon Boncelet Jr.'s algorithm, we first define $\Delta_j = c_j - c_{j-1}$ for $c_0 = 0$ for all $j = 1, \ldots, d$

and $\mathbf{\Delta} = (\Delta_1, \ldots, \Delta_j)$. We then construct tables $T_0, \ldots, T_n$ where $T_i \in \bigotimes_{j=1}^{d} [\Delta_j]$. At a high level, the algorithm updates these tables using a similar recurrence relationship as in Boncelet Jr.'s algorithm with the stipulation that once the number of balls in the jth bin reaches Δ_j, it is considered to be 'full' and any additional balls that land in this bin then 'spill-over' into the next open bin. The intuition for this spill-over step is exactly that we care only that D_j is satisfied, rather than the exact number of balls in the first j bins. More specifically, we the algorithm works as follows:

1. Suppose the jth through kth bins are full and the $k+1$st bin is not from throwing the first i balls. Then, if the $i+1$st ball is thrown in any of bins j through k (or $k+1$), then it will land in bin $k+1$. That is, if the $i+1$st ball lands anywhere in bins j through $k+1$, then $B_{k+1}^{i+1} = B_{k+1}^{i} + 1$ and $B_{k'}^{i+1} = B_{k'}^{i}$ for all $k' \neq k$.

2. Define the 'spill-over' function $\psi : \mathbb{N}^d \times \mathbb{N}^d \times \mathbb{N} \to \mathbb{N}$:

$$\psi(\mathbf{B}, \mathbf{\Delta}, j) = \min(j, \{i : B_k = \Delta_k \ \ \forall k \text{ s.t } i \leq k \leq j\}) \tag{23}$$

   For our purposes, $\psi(\mathbf{B}, \mathbf{\Delta}, j)$ is the smallest index i such that a ball thrown into the ith bin results in the ball landing in the jth bin.

3. Suppose the jth through dth bins are full after the first i ball tosses. Then, if the $i+1$st ball is thrown in any of these bins, then it will 'spill-out' of all bins and will be treated as if it did not fall into any of the bins. That is, if the $i+1$st ball lands anywhere in bins j through d, then $\mathbf{B}^{i+1} = \mathbf{B}^{i}$.

4. We define the 'spill-out' function $\theta : \mathbb{N}^d \times \mathbb{N}^d \to \mathbb{N}$:

$$\theta(\mathbf{B}, \mathbf{\Delta}) = \min(d, \{i : B_k = \Delta_k \ \ \forall k \text{ s.t. } i < k \leq d\}) \tag{24}$$

   $\theta(\mathbf{B}, \mathbf{\Delta})$ represents the smallest index i such that all bins indexed larger than i are full. That is, any balls thrown into bins $i+1$ through d spill-out and are ignored. With this defined, the main recurrence relation (which we will further modify) is given by

$$T_i(\mathbf{B}) = \mathbb{P}(X_i > x_{\theta(\mathbf{B}, \mathbf{\Delta})}) T_{i-1}(\mathbf{B}) + \sum_{j=1}^{d} \sum_{k=\phi(\mathbf{B}, \mathbf{\Delta}, j)}^{j} 1_{B_j > 0} \mathbb{P}(X_i \in I_k) T_{i-1}(\mathbf{B} - \mathbf{1}_j) \tag{25}$$

4.3 Spill-Over Algorithm, Complexity Analysis, Proof

Algorithm 3 Joint CDF of selected order statistics $\mathcal{C}$ for independent r.v.'s (Spill-Over)

Require: $d \in \mathbb{N}$, $n \in \mathbb{N}$, $\mathbf{p}_i \in [0,1]^{d+1}$ for $i \in [n]$, $\mathcal{C} \in \{(c_1, \ldots, c_d) : 1 \leq c_1 < \ldots < c_d \leq n\}$
Ensure: $T(\mathbf{\Delta}) = \mathbb{P}(D_{1:d})$

```
T_0(0) = 1
Δ = (c_1, c_2 − c_1, ..., c_d − c_{d−1})
for i ← 1 to n do
    for B in decreasing order do
        T_i(B) = P(X_i > x_{θ(B,Δ)}) T_{i−1}(B) + Σ_{j=1}^{d} Σ_{k=φ(B,Δ,j)}^{j} 1_{B_j>0} P(X_i ∈ I_k) T_{i−1}(B − 1_j)
    end for
end for
return T_n(Δ) = P(D_{1:d})
```

Since there are a total of n tables (which are disposable) each containing $O(\prod_{j=1}^{d} \Delta_j)$ table entries with each update to an entry formula involving the sum over $O(d^2)$ terms, the space and time complexity of this algorithm are $O(\prod_{j=1}^{d} \Delta_j)$ and $O(nd^2 \prod_{j=1}^{d} \Delta_j)$ respectively. This is worst case a factor of $O(d^d)$ more efficient; a significant improvement over Boncelet Jr.'s original algorithm, though is still exponential in d. In the sponsored search setting, d will represent the granularity at which we discretize continuous distributions and n is the number of bidders (which we will modify this algorithm later so as to only need to track the k largest bidders to yield $n = k$). The Δ_j's for our purposes will be equal to 1, as the order statistics of interest are the top k, which means $c_j - c_{j-1} = \Delta_j = 1$.

4.4 Proof: Independent Case

Theorem 1. *For all* $\mathbf{B}$ *such that* $\mathbf{0} \leq \mathbf{B} \leq \mathbf{\Delta}$, *we have the following equality at the ith iteration:*

$$\mathbb{P}(A_{i,\mathbf{B}}) = \mathbb{P}(X_i > x_{\theta(\mathbf{B},\mathbf{\Delta})})\mathbb{P}(A_{i-1,\mathbf{B}}) + \sum_{j=1}^{d} \sum_{k=\phi(\mathbf{B},\mathbf{\Delta},j)}^{j} 1_{B_j>0}\mathbb{P}(X_i \in I_k)\mathbb{P}(A_{i-1,\mathbf{B}-\mathbf{1}_j}) \tag{26}$$

Proof. Recall that before the ith iteration of the algorithm, only $i-1$ balls have been thrown thus far, so $C^i_{1:j}$ and $B^i_{1:j}$ represent the number of and lower bound on the first i balls thrown into first j bins respectively for all j. Let $\mathbf{B}^{i-1}$ and $\mathbf{B}^i$ denote the corresponding lower bounds before and after throwing the ith ball. There are 4 possible cases:

1. If $\mathbf{B}^i = \mathbf{B}^{i-1} + \mathbf{1}_j$: The ith ball was thrown into the jth bin, $B^{i-1}_j < \Delta_j$. This event has corresponding probability $\mathbb{P}(x_{j-1} < X_i \leq x_j)\mathbb{P}(A_{i-1,\mathbf{B}-\mathbf{1}_j})$.

2. If $\mathbf{B}^i = \mathbf{B}^{i-1} + \mathbf{1}_j$: The ith ball spilled over into the jth bin from a lower indexed bin. $\psi(\mathbf{B}, \mathbf{\Delta}, j)$ corresponds to the smallest index k such that a ball thrown in k spills over into bin j. Then, the

corresponding probability is:

$$\mathbb{P}(x_{k-1} < X_i \leq x_d)\mathbb{P}(A_{i-1,\mathbf{B}-\mathbf{1}_k}) = \sum_{k=\psi(\mathbf{B},\mathbf{\Delta},j)}^{j} \mathbb{P}(x_{k-1} < X_i \leq x_k)\mathbb{P}(A_{i-1,\mathbf{B}}) \tag{27}$$

3. If $\mathbf{B}^i = \mathbf{B}^{i-1}$: The ith ball was thrown into the jth bin where $\bigcap_{k=j}^{d}\{B_k = \Delta_j\}$ (all subsequent bins are full). $\theta(\mathbf{B},\mathbf{\Delta})$ corresponds to the smallest index where all subsequent bins are full. This event then has corresponding probability:

$$\mathbb{P}(x_{k-1} < X_i \leq x_d)\mathbb{P}(A_{i-1,\mathbf{B}}) = \sum_{k=\theta(\mathbf{B},\mathbf{\Delta})}^{d} \mathbb{P}(x_{k-1} < X_i \leq x_k)\mathbb{P}(A_{i-1,\mathbf{B}}) \tag{28}$$

.

4. If $\mathbf{B}^i = \mathbf{B}^{i-1}$: The ith ball doesn't land in any bins. This event has corresponding probability $\mathbb{P}(X_i > x_d)\mathbb{P}(A_{i-1,\mathbf{B}})$.

Summing up the probability of these events yields the desired result. □

Theorem 2. *By the end of the ith iteration,* $\mathbb{P}(A_{i,\mathbf{B}}) = T_i(\mathbf{B})$ *for* $i = 0, \ldots, n$ *and* $\mathbf{0} \leq \mathbf{B} \leq \mathbf{\Delta}$.

Proof. The proof is by strong induction on i. In the base case, before the first iteration (0 balls have been thrown), we have that $\mathbb{P}(A_{0,\mathbf{0}}) = 1$. Likewise in our algorithm, $T_0(\mathbf{0}) = 1$.

We now prove the inductive step for $i \geq 1$. Assume the strong induction hypothesis, namely $\mathbb{P}(A_{i-1,\mathbf{B}}) = T_{i-1}(\mathbf{B})$ for all $0 \leq \mathbf{B} \leq \mathbf{n}$. Defining $\psi(\mathbf{B},\mathbf{\Delta},j)$ and $\theta(\mathbf{B},\mathbf{\Delta})$ to be as in (23) and (24) respectively, we have the following by Theorem (1) and the strong induction hypothesis:

$$\mathbb{P}(A_{i,\mathbf{B}}) = \mathbb{P}(X_i > x_{\theta(\mathbf{B},\mathbf{\Delta})})\mathbb{P}(A_{i-1,\mathbf{B}}) + \sum_{j=1}^{d}\sum_{k=\phi(\mathbf{B},\mathbf{\Delta},j)}^{j} \mathbb{P}(X_i \in I_k)\mathbb{P}(A_{i-1,\mathbf{B}-\mathbf{1}_k}) \tag{29}$$

□

4.5 Expected Revenue in GFP and Myerson's

In addition to the output of the algorithm having interpretation as the joint cdf of order statistics of non-identically distributed random variables evaluated at a point $\mathbf{x}$, the table entries $T(\mathbf{B})$ are also meaningful. In particular, they represent the probability that there are at least B_i balls in the first i bins (there are at least B_i random variables at most x_i). Furthermore, we have exact equality between B_i and the number of balls in bin i if $B_{i-1} < \Delta_i$ and $B_i < \Delta_i$. In the case of the first bin, we have strict equality if $B_1 < \Delta_1 = c_1$. As such, we can use this fact to compute the expected value of the ith order statistic which will be useful in characterizing the expected revenue of GFP and Myerson's auction as well as the expected payoff in the GFP and GSP settings. Tackling the issue of computing expected revenue first, we state a useful relationship that will let us compute these quantity:

Theorem 3. *Let $T_n^{m+1,x}$ denote the Spill-Over tables after n iterations with input parameters $\mathcal{C} = \{m+1\}$ at point $-x$ with underlying distributions F_{-b_i} for $i \in [n]$. Then for $B \in [m]$ we have:*

$$F_{b_{(i)}}(x) = \sum_{j=i}^{m+1} T_n^{m+1,x}(j) \tag{30}$$

Proof. Recall that we have exact equality of B and the number of balls in the first bin when $B < m+1$. Here, the first bin denotes the region $(-\infty, -x]$ for the negated versions of the bids. This yields the equivalent of the region $[x, \infty)$ for the non-negated bids. Hence, for $B < m+1$ $T_n^{m+1,x}(B)$ denotes probability that the exact number of bids in the range $[x, \infty)$ is B. Similarly, $T_n^{m+1,x}(m+1)$ denotes the probability that there are at least $m+1$ bids at least x.

$$\begin{aligned} F_{b_{(i)}}(x) &= \mathbb{P}(i\text{th largest bid at most } x) && (31)\\ &= \mathbb{P}(\text{At least } i \text{ bids at least } x) && (32)\\ &= \mathbb{P}(\text{At least } m+1 \text{ bids at least } x) + \sum_{j=i}^{m} \mathbb{P}(\text{Exactly } j \text{ bids at least } x) && (33)\\ &= T_n^{m+1,x}(m+1) + \sum_{j=i}^{m} T_n^{m+1,x}(j) && (34)\\ &= \sum_{j=i}^{m+1} T_n^{m+1,x}(j) && (35) \end{aligned}$$

□

Having an easily computable expression for the cdf the ith largest bid in the GFP setting, we can discretize (18) and evaluate the cdf multiplied by the indicator function of being greater than the reserve π at each of the lattice points. Summing over each of these points yields the expectation (up to discretization error) of the expected payment of the ith highest bidder. Summing over these expressions for $i \in \{1, \ldots, m\}$ yields the expected revenue of the GFP. A similar procedure can be applied for Myerson's auction, where instead of computing sums of the expectations of the m largest bids greater than the reserve π, we compute the sums of the expectations of the m largest non-negative virtual valuations.

4.6 Expected Utility in GFP and GSP

As we mentioned earlier, the expected payoff in the GSP and GFP auctions $\mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}(v_i x_i(b_i, \mathbf{b}_{-i}, \mathcal{A}_\theta)) = \sum_{j=1}^{m} v_i \mathbb{P}(\text{win slot } j \text{ with bid } b \mid F_{\mathbf{b}_{-i}})\alpha_j$ can also be computed using the previous observation characterizing when B_i is exact. We will need a slight modification of the Spill-Over algorithm for the case $d = 1$ and discrete distributions, as we will be discretizing the valuation and bid space in order to account for ties.

Consider computing the expected utility of bidder i given their valuation v_i, opponent bid distributions $F_{b_j|\mathcal{A}_\theta}$ for $j \neq i$, and auction $\mathcal{A}_\theta$. Let $b \in \mathcal{B}$ be some value in the support of discrete distributions $F_{b_j|\mathcal{A}_\theta}$ for $j \neq i$. Letting $b_j \sim F_{b_j|\mathcal{A}_\theta}$ be independent bids, we define $U_i^b : \mathbb{N} \times \mathbb{N} \times \mathcal{B} \to [0, 1]$ to be a dynamic programming table such that for $x \in [m-1], y \in [n], z \in \mathcal{B}$:

$$U_i^b(x, y, z) = \text{Prob}\left(x \text{ values in } \mathbf{b_{-i}} > b, y \text{ values in } \mathbf{b_{-i}} = b, z \text{ is the next largest bid in } \mathbf{b_{-i}} < b\right) \tag{36}$$

As per the Spill-Over algorithm's insight, for $y \in [n], z \in \mathcal{B}$, we can compress some information for the case $x \geq m$ into a single entry:

$$U_i^b(m, y, z) = \text{Prob}\left(\text{at least } m \text{ values in } \mathbf{b_{-i}} > b, y \text{ values in } \mathbf{b_{-i}} = b, z \text{ is the next largest bid in } \mathbf{b_{-i}} < b\right) \tag{37}$$

We initialize the table to be all zeros except at $U(0, 0, \underline{b}) = 1$. The recurrence relation is similar to that of the Spill-Over algorithm for the first two entries x and y, as this is simply updating the probability that some number of bids take on a value in a specified range; namely (b, ∞) and b respectively. The last entry allows us to obtain the conditional distribution of the next largest bid given the current bid. Iterating over $j = 1, \ldots, n$, $j \neq i$, the recurrence relation is as follows:

1. Initialize a new table $\tilde{U}_i^b$ to all zeros and for notational simplicity, assume $U_i^b(x', y', z') = 0$ for $x' \notin [m], y' \notin [n], z' \notin \mathcal{B}$. For $x \in [m-1], y \in [n], z \in \mathcal{B}$:

$$\tilde{U}_i^b(x, y, z) = U_i^b(x-1, y, z)\mathbb{P}(b_j > b) + U_i^b(x, y-1, z)\mathbb{P}(b_j = b) + \sum_{z' < z} U_i^b(x, y, z')\mathbb{P}(b_j = z') \tag{38}$$

2. For the case $x = [m]$, we have a slightly modified relation:

$$\tilde{U}_i^b(m, y, z) = \left[U_i^b(m, y, z) + U_i^b(m-1, y, z)\right]\mathbb{P}(b_j > b) \tag{39}$$

$$+ U_i^b(m, y-1, z)\mathbb{P}(b_j = b) + \sum_{z' < z} U_i^b(m, y, z')\mathbb{P}(b_j = z') \tag{40}$$

3. After all entries of $\tilde{U}_i^b$ have been updated, then set $U_i^b = \tilde{U}_i^b$. Repeat this process for all $j \neq i$. More formally, the algorithm is as follows:

Algorithm 4 Conditional Spill-Over Algorithm for $d = 1$

Require: $m, n \in \mathbb{N}$, $i \in [n]$, $\mathcal{B}$, $b \in \mathcal{B}$, $\{\mathbb{P}(b_j > b) = p_{j,+}, \mathbb{P}(b_j = b) = p_{j,=}, \mathbb{P}(b_j = z) = p_{j,z}\}$ for all $j \neq i, z < b$

Ensure: Table U_i^b

```
U_i^b(0,0,0) = 1
Ũ_i^b = U_i^b
for j ∈ 1,...,n, j ≠ i do
    for x ∈ [m], y ∈ [n], z ∈ B do
        Ũ_i^b(x,y,z) = U_i^b(x−1,y,z) p_{j,+} + U_i^b(x,y−1,z) p_{j,=} + ∑_{z'<z} U_i^b(x,y,z') p_{j,z'}
    end for
    for y ∈ [n], z ∈ B do
        Ũ_i^b(m,y,z) += U_i^b(m,y,z) p_{j,+}
    end for
    U_i^b ← Ũ_i^b
end for
return U_i^b
```

The proof is similar to that of the Spill-Over algorithm. While we do not provide the formal proof here, the proof overview relies on the disjoint-ness of events of the form

$$\{\text{Exactly } x \text{ bids in some interval, exactly } y \text{ bids in a separate interval, the next highest bid takes on value } z\} \tag{41}$$

and applying the law of total probability. Now, the table is of size $O(mn|\mathcal{B}|)$ and each update requires summing over $O(|\mathcal{B}|)$ terms for total algorithm space and time complexities of $O(mn|\mathcal{B}|)$ and $O(mn|\mathcal{B}|^2)$ respectively. Now, we can characterize the expected payoff of the GFP and GSP auctions through this algorithm. Firstly, the troublesome term $\mathbb{P}(\text{win slot } k \text{ with bid } b \mid F_{\mathbf{b}_{-i}})$ in the expression for the expected payoff for bidder i for bidding at b with valuation v_i given competitor distributions $F_{\mathbf{b}_{-i}|\mathcal{A}_\theta}$ can now be expressed as a function of the table U_i^b:

Theorem 4. *Let U_i^b denote the conditional Spill-Over table with input parameters $\mathcal{C} = \{m+1\}$ at point b with underlying distributions F_{b_j} for $j \neq i$ and $i, j \in [n]$. Assuming uniformly random tie-breaking, then we have the following relationship for $x + y \geq k - 1$:*

$$\mathbb{P}(\textit{win slot } k \textit{ with bid } b \mid F_{\mathbf{b}_{-i}}) = \sum_{x<k} \sum_{z \in \mathcal{B}} \frac{U_i^b(x,y,z)}{y+1} \tag{42}$$

Proof. In order to have a positive probability of being allocated slot k, we require that there be strictly fewer than k bids greater than b (or else bidder i will be allocated a lower slot) and also $x + y \geq k - 1$ (or else bidder i will be allocated a higher slot). In these cases, the probability that we are allocated slot k is the number of bidders, including bidder i, who tied at bid b; namely $y + 1$. By the law of total probability, we can sum over all entries $z \in \mathcal{B}$ in U_i^b to obtain the probability that there are x bids strictly greater than b and y bids at b. □

One may make the observation that z is actually unnecessary in this computation as we simply marginalize it out at the end. Indeed this is the case; and a more efficient algorithm—faster by a factor of $|\mathcal{B}|^2$—can be obtained by only considering probabilities of the form:

$$\mathbb{P}(\text{Exactly } x \text{ bids strictly greater than } b \text{ and exactly } y \text{ bids at } b) \tag{43}$$

.

We have implemented this simplified conditional algorithm for the GFP. The reason why we show this algorithm is that the expected payments $\mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}(p_i(b_i, \mathbf{b}_{-i}, \text{GSP}(\pi))$ can also be obtained from this table:

Theorem 5. *Let U_i^b denote the conditional Spill-Over table with input parameters $\mathcal{C} = \{m+1\}$ at point b with underlying distributions F_{b_j} for $j \neq i$ and $i, j \in [n]$. Assuming uniformly random tie-breaking, then the expected payment by bidding at $b \geq \pi$ given competitor bidding distributions $F_{\mathbf{b}_{-i}|\mathcal{A}_\theta}$:*

$$\mathbb{E}_{\mathbf{b}_{-i}|\mathcal{A}_\theta}(p_i(b_i, \mathbf{b}_{-i}, GSP(\pi)) = \sum_{x+y<m, z\in\mathcal{B}} \left[\frac{y}{y+1} U_i(x,y,z) b + \frac{1}{y+1} U_i(x,y,z) \max(z,\pi)\right] \tag{44}$$

$$+ \sum_{x+y\geq m, x<m, z\in\mathcal{B}} \left[\frac{m-x}{y+1} U_i(x,y,z) b\right] \tag{45}$$

Proof. Similar to the analysis done for the probability of winning a particular slot k with a bid b, we consider a particular table entry $U_i(x,y,z)$ item and multiply by the payment associated at that particular configuration of (x,y,z).

1. In the case that $x \geq m$, then bidder i is not allocated an item.

2. If $x < m$ and $x + y < m$, then there is a $\frac{y}{y+1}$ probability that bidder i is allocated any of slots $x+1, \ldots, x+y$ and must pay the bid of the winner of the next item, which in this case is b. There is a $\frac{1}{y+1}$ probability bidder i is allocated item $x+y+1$ and must pay the maximum of the reserve and the bid of the winner of the next item, which is $\max(z,\pi)$.

3. If $x < m$ and $x + y \geq m$, then there are some bidders who tied and will not be allocated an item. There are $m - x$ items remaining and $y+1$ bidders who tied with bid b, so the probability that bidder i is allocated an item is $\frac{m-x}{y+1}$. Consequently, their expected payment in this case is $\frac{m-x}{y+1}(b)$.

□

4.7 Expected Revenue in GSP

The expected revenue in GSP can be computed in a similar fashion to the expected payments in the GSP setting. That is, since we have already computed the expected payment of the ith bidder for bidding at b for a particular valuation, in order to compute their expected payment over all possible bids, we can simply enumerate over their expected utility maximizing bid for each possible valuation v_i and weigh the expected payments by $f_{v_i}(v_i)$. Summing over the expected payment of all bidders—which is a strategy we can also employ to compute the expected payment in the GFP setting—yields the expected revenue of GSP.

Theorem 6. *Let* $\mathbb{E}_{\mathbf{b}_{-i}}\left(p_i(b^*(v), \mathbf{b}_{-i}, \mathcal{A}_\theta)\right)$ *denote the expected payment under the utility maximizing bid given valuation* v*. Then the expected revenue of GSP can be computed as the sum of the expected payments of each bidder:*

$$\sum_{i=1}^{n} \mathbb{E}_{v_i}\left[\mathbb{E}_{\mathbf{b}_{-i}}\left(p_i(b^*(v_i), \mathbf{b}_{-i}, \mathrm{GSP}(\pi))\right)\right] = \sum_{i=1}^{n} \sum_{v \in \mathcal{B}} f_{v_i}(v) \mathbb{E}_{\mathbf{b}_{-i}}\left(p_i(b^*(v), \mathbf{b}_{-i}, \mathrm{GSP}(\pi))\right) \tag{46}$$

Proof. This follows immediately from linearity of expectations and the definition of expectation. $\square$

5 Experiments and Results

In this section, we describe the two experiments that we will run, the results, and provide some analysis. In particular, we will be running a toy experiment with only a single good and $n = 2,4,8$ bidders with i.i.d. Unif$(0,1)$ valuations to see whether the results match those predicted by theory for GFP and GSP. We will then run a small scale test with $k = 2$ slots with CTRs $[\alpha_1, \alpha_2] = [0.8, 0.6]$ and $n = 2,4,8$ bidders with i.i.d. valuation distributions Unif$(0,1)$. We will use discretization points $\frac{t}{100}$ for $t \in [100]$. We will then more closely look at how the expected revenue and bidder distributions change over iterations of best response dynamics and as the reserve price varies. In both experiments, we also try different initial bidding distribution initializations and variants of best response dynamics, such as sequential best response dynamics (SBRD) where bidders learn sequentially and weighted probability strategies (WPS) where bidders select bids with probability exponential in its conferred expected utility. These variants increase the 'noise' in the best response dynamics system as simultaneous BRD yields strictly symmetric bidding distributions, whereas SBRD does not. WPS allows bidders to have a mixed strategy for a fixed valuation, which combined with SBRD, yields the most constraint relaxed dynamical system over the space of strategies. In other words, SBRD relaxes the symmetric assumption and WPS relaxes the single dominant strategy assumption.

5.1 Single Item

For this experiment, we wanted to see whether best response dynamics can recover the dominant strategies in the $m = 1$ case for GSP and GFP. This corresponds to the standard Vickrey auction which has several nice properties; particularly so in the case where the bidder valuations are Unif$(0,1)$ which we will assume. In GSP, bidders have a dominant strategy for any fixed valuation which is to bid truthfully (DSIC) subject to the reserve price. Similarly, in GFP, assuming n bidders with Unif$(0,1)$ valuations, bidder i bidding at $\frac{n}{n+1}(v_i)$ constitutes a Nash equilibrium for any fixed v_i without a reserve. Similarly, for reserve price π, taking first order conditions on the utility for a bidder i with fixed valuation v_i in GFP yields:

$$\frac{d}{v_i}\left[(v - b_i(v_i))\mathbb{P}(v_i > v_j \text{forall j})\right] = 0 \rightarrow b_i'(v_i) = \frac{(n-1)(v_i - b_i(v_i))}{v_i} \tag{47}$$

Using the initial condition that $b_i(\pi) = \pi$, we obtain that the Bayes Nash equilibrium bid in the GFP with n bidders and reserve π is given by:

$$b_i(v_i) = (1 - \frac{1}{n})v_i + \frac{\pi^n}{nv_i^{n-1}} \tag{48}$$

Note though, that this is only a symmetric BNE. This is a considerable difference, seeing that GSP is DSIC and GFP does not have a dominant strategy. Since the strategies of the bidders in the one-item GSP should not depend on those of others, we should expect convergence of best response dynamics in a single iteration—regardless of the initialization—to truthful behavior subject to the reserve price. However, since the BNE (SBNE) in the GFP is dependent on the other bidders behaving the same, it might not be possible to recover this particular BNE. Indeed, we verify this in our results for the GFP. We are also interested in comparing the revenues and optimal reserve prices yielded by both GFP and GSP when running BRD and its variants as compared to the the theoretical expected revenue.

5.1.1 Learned Strategies

Here, we show figures depicting the bid distributions after running 100 iterations of BRD/SBRD/WPS. Since the bid function are monotonic, taking the inverse of the bidding distribution yields the bid function. We also include the shifted versions of the bidding distributions—learning dynamics run for an extra trial—in order to see the convergence or non-convergence of these dynamics.

5.1.2 Expected Revenue and Optimal Reserve

Plotting the expected revenue, we can see that we can recover the optimal reserve of $\frac{1}{2}$ in most of the cases, though there are some exceptions, especially for GFP using BRD or SBRD due to cycling.

5.2 Multiple Items, i.i.d. valuations

Here, we let $k = 2$ and $\alpha_1 = 0.8, \alpha_2 = 0.6$. Recall the symmetric equilibrium behavior as predicted in theory for the GSP and GFP given by Equations (9) and (7). We will show that the cycling behavior as seen in GFP's BRD and SBRD in the toy example extends into this setting as well, in addition to the GSP's BRD and SBRD. Most importantly though, we note that the expected revenues as provided in the Gomes, Sweeney 2014, and Han, Liu 2015 papers are under the assumption that $\alpha_1 = 1$. For example, in the GFP setting, it is apparent that bidders will never bid above their maximum possible valuation multiplied by the largest alpha, as this would yield non-positive utility. However, in the graphs of the theoretical equilibrium revenue, we see that these do indeed return a positive expected revenue suggesting that bidders are willing to bid *larger* than their maximum possible valuation (which in our case is 1) multiplied by the largest alpha (which is 0.8).

5.2.1 Learned Strategies

5.2.2 Expected Revenue and Optimal Reserve

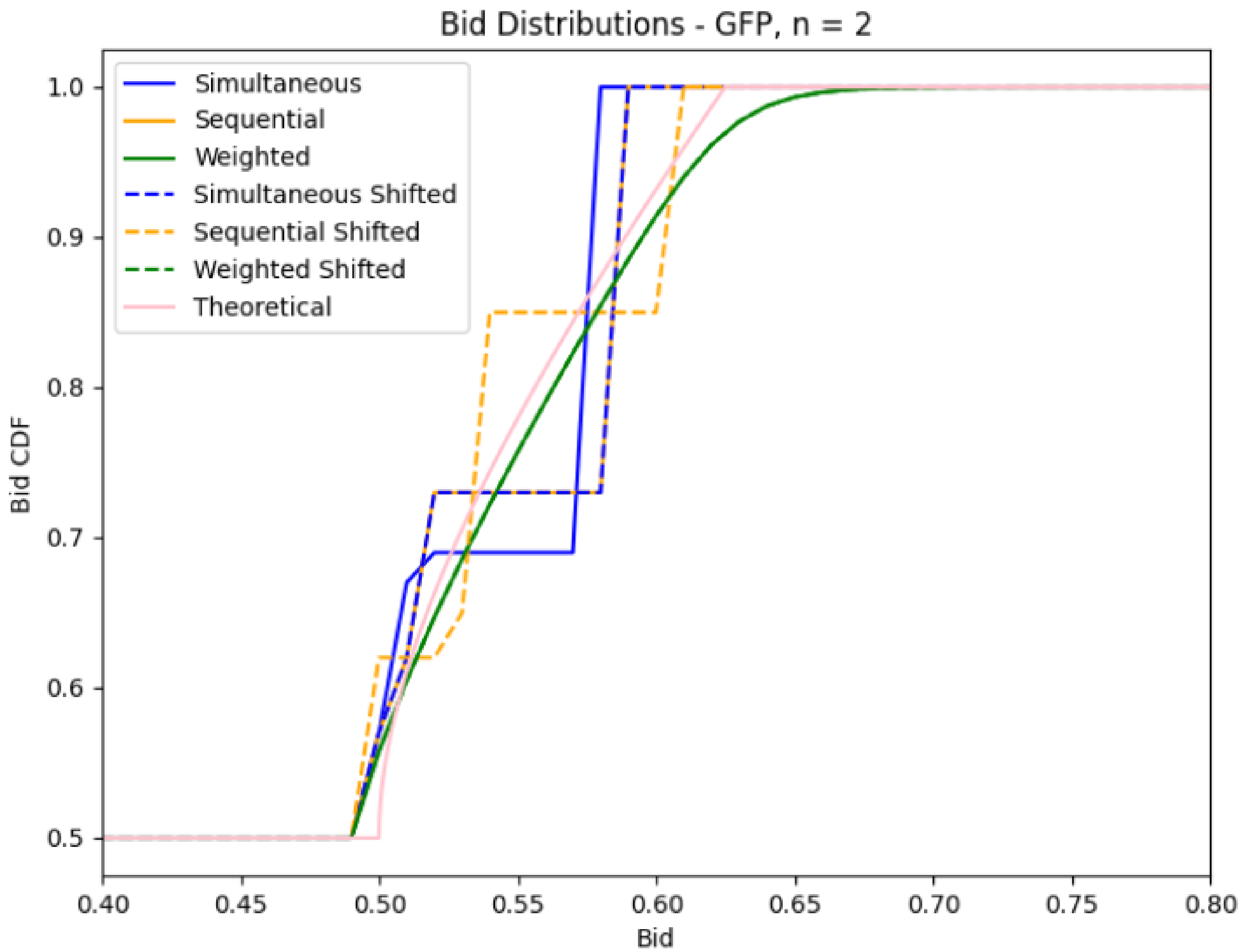

Figure 1: Bidder dynamics run for 100 and 101 trials for GFP, $n = 2$, $\pi = \frac{1}{2}$. The BNE is shown in pink. Neither simultaneous nor sequential learning can be seen to recover the BNE or converge to any equilibrium, though it is apparent that the bidder distributions cycle as the sequential and simultaneous shifted curves overlap entirely. It is interesting that both BRD and SBRD return step functions in which bidders's valuations 'pool' (must be strictly monotonic) into the same bid, which at least for the GSP, Gomes and Sweeney showed should not be the case in any BNE. This phenomena is due to the discretized nature of the experiment. We will see that sequential learning also does not return symmetric bidding functions/distributions. However, WPS, shown in green, converges, traces the theoretical optima, and is also symmetric, though this is not shown here.

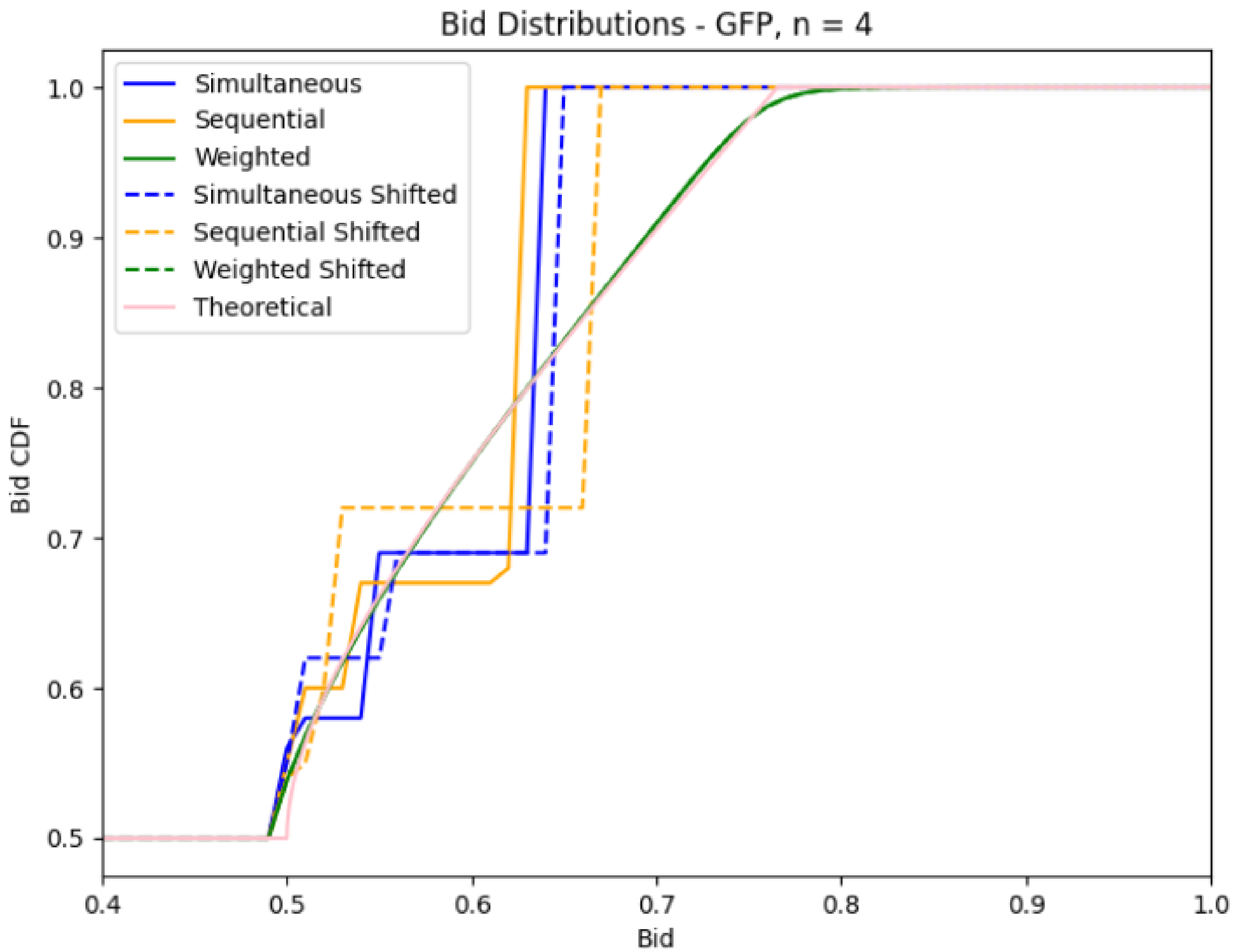

Figure 2: Bidder dynamics run for 100 and 101 trials for GFP, $n = 4$, $\pi = \frac{1}{2}$. The BNE is shown in pink. Much like the case for $n = 2$, the simultaneous and sequential do not converge to the theoretical equilibrium, or any equilibrium for that matter. WPS once again hugs the theoretical value.

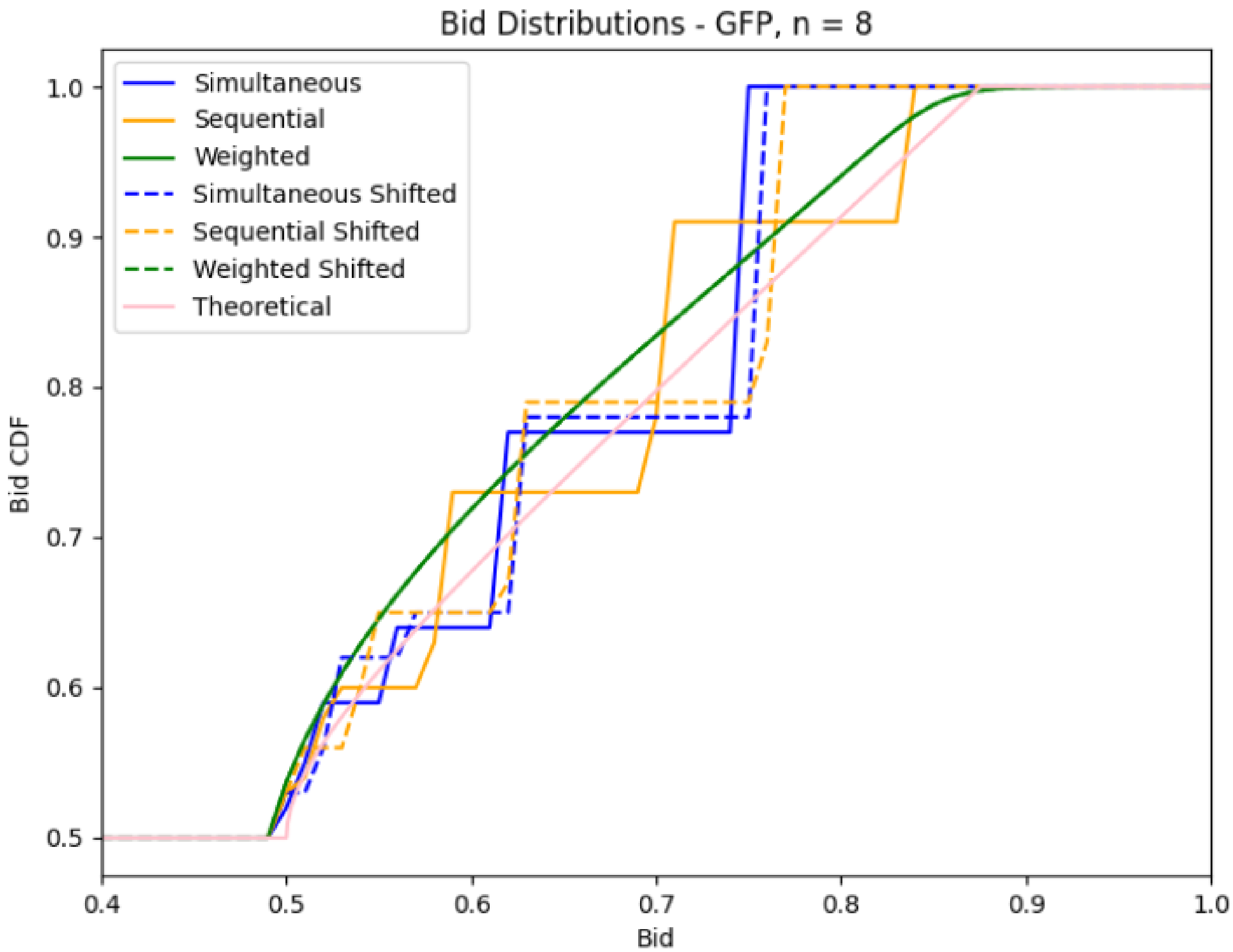

Figure 3: Bidder dynamics run for 100 and 101 trials for GFP, $n = 8$, $\pi = \frac{1}{2}$. The BNE is shown in pink. Much like the case for $n = 2$ and $n = 4$, the simultaneous and sequential do not converge to the theoretical equilibrium, or any equilibrium for that matter. WPS once again hugs the theoretical value, though is slightly above as opposed to slightly below for $n = 2$.

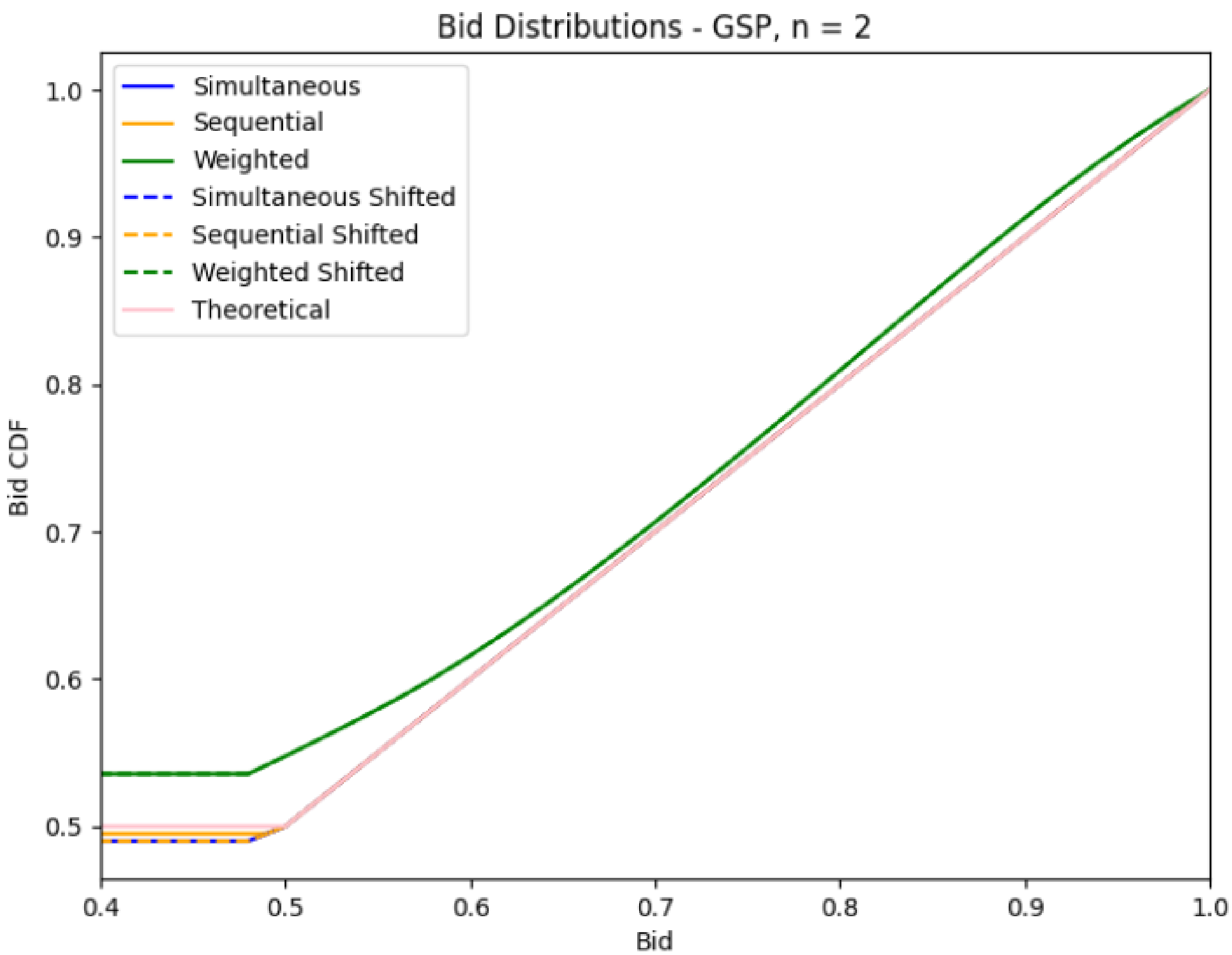

Figure 4: Bidder dynamics run for 100 and 101 trials for GSP, $n = 2$. The BNE is shown in pink. The BNE is shown in pink. Both simultaneous and sequential converge to the theoretical equilibrium whereas the weighted is slightly above the equilibrium. It is worth noting that both sequential and weighted both converge to a symmetric equilibrium.

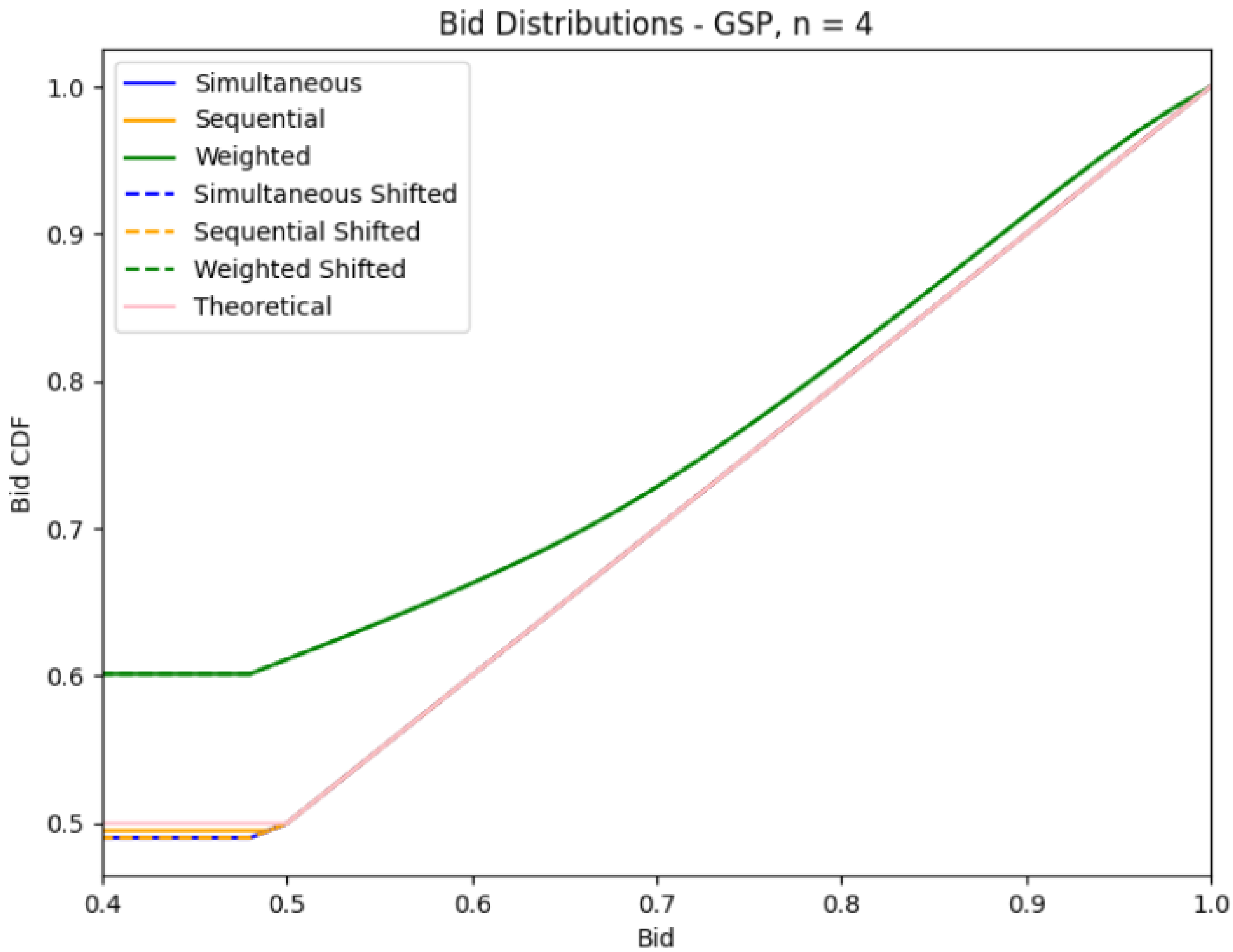

Figure 5: Bidder dynamics run for 100 and 101 trials for GSP, $n = 4$, $\pi = \frac{1}{2}$. The analysis is identical to the case for $n = 2$, except now the weighted is more so above the theoretical equilibrium.

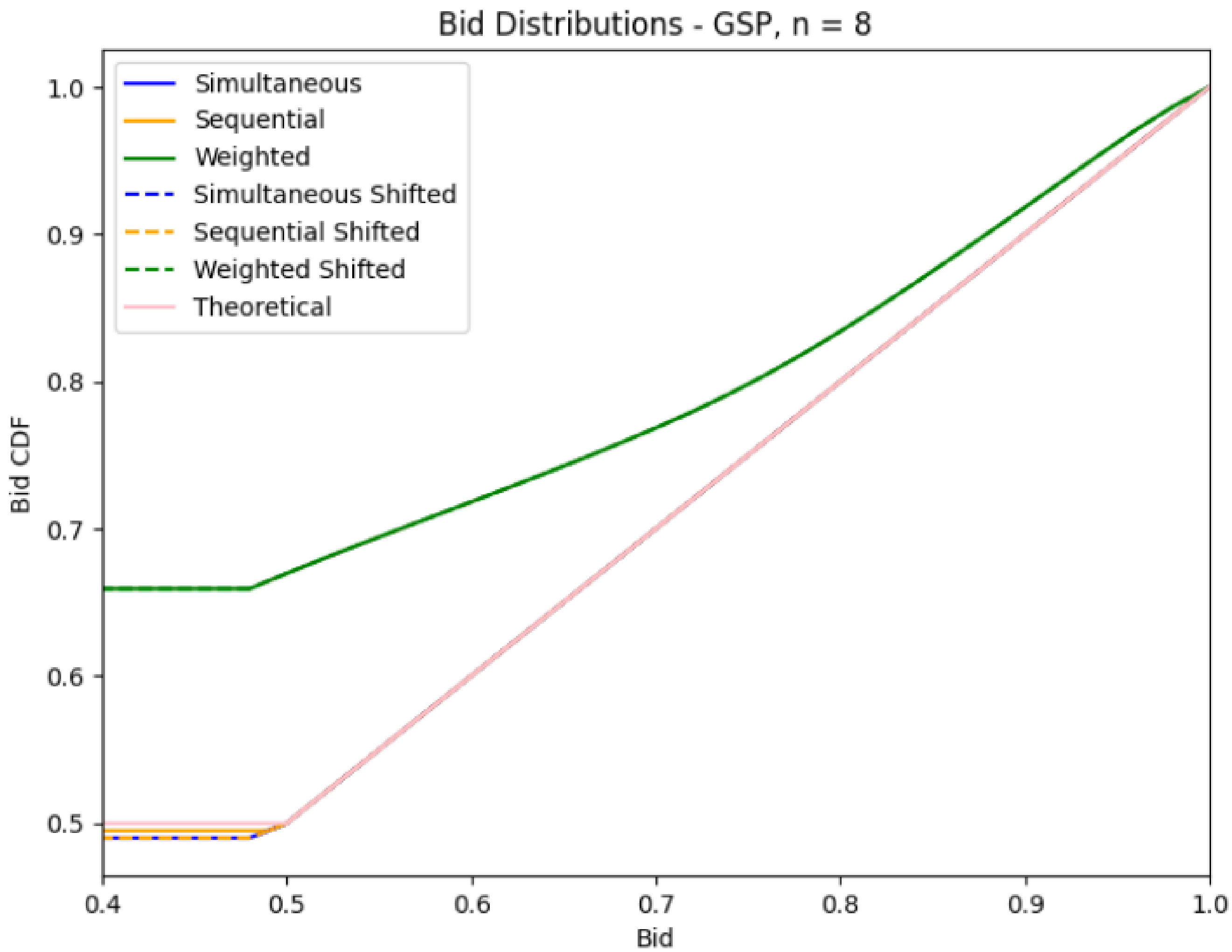

Figure 6: Bidder dynamics run for 100 and 101 trials for GSP, $n = 8$. The BNE is shown in pink. The analysis is identical to the case for $n = 2$, except now the weighted is even more so above the theoretical equilibrium than in $n = 4$.

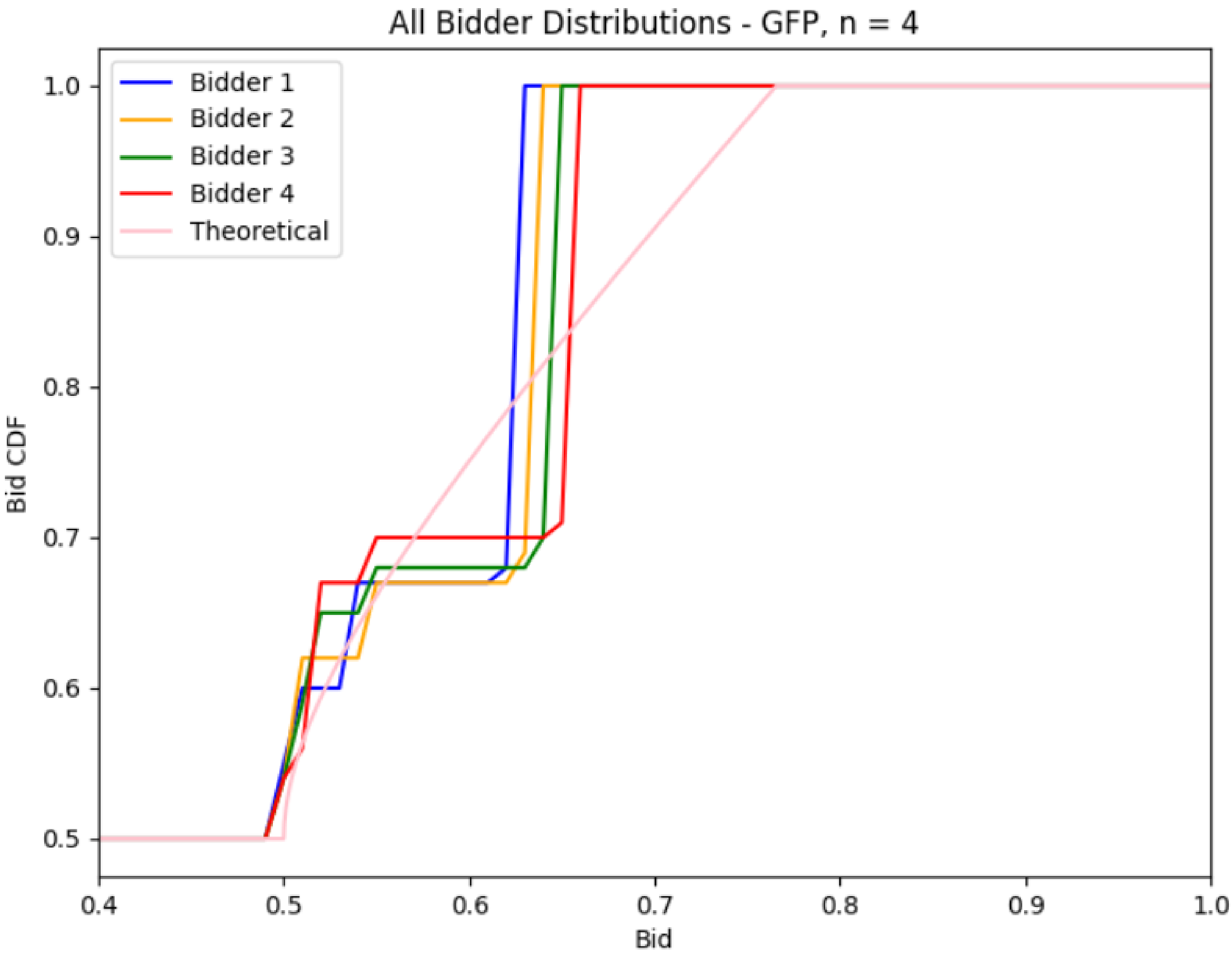

Figure 7: Bidder dynamics run for 100 and 101 trials for GFP, $n = 4$, $\pi = \frac{1}{2}$, using SBRD. The different bidder distributions are shown to be distinct step functions, somewhat centered around the equilibrium. This shows us the cycling behavior seen in GFP when using sequential learning; a similar pattern appears using the bidding distributions of time-shifted simultaneous learning.

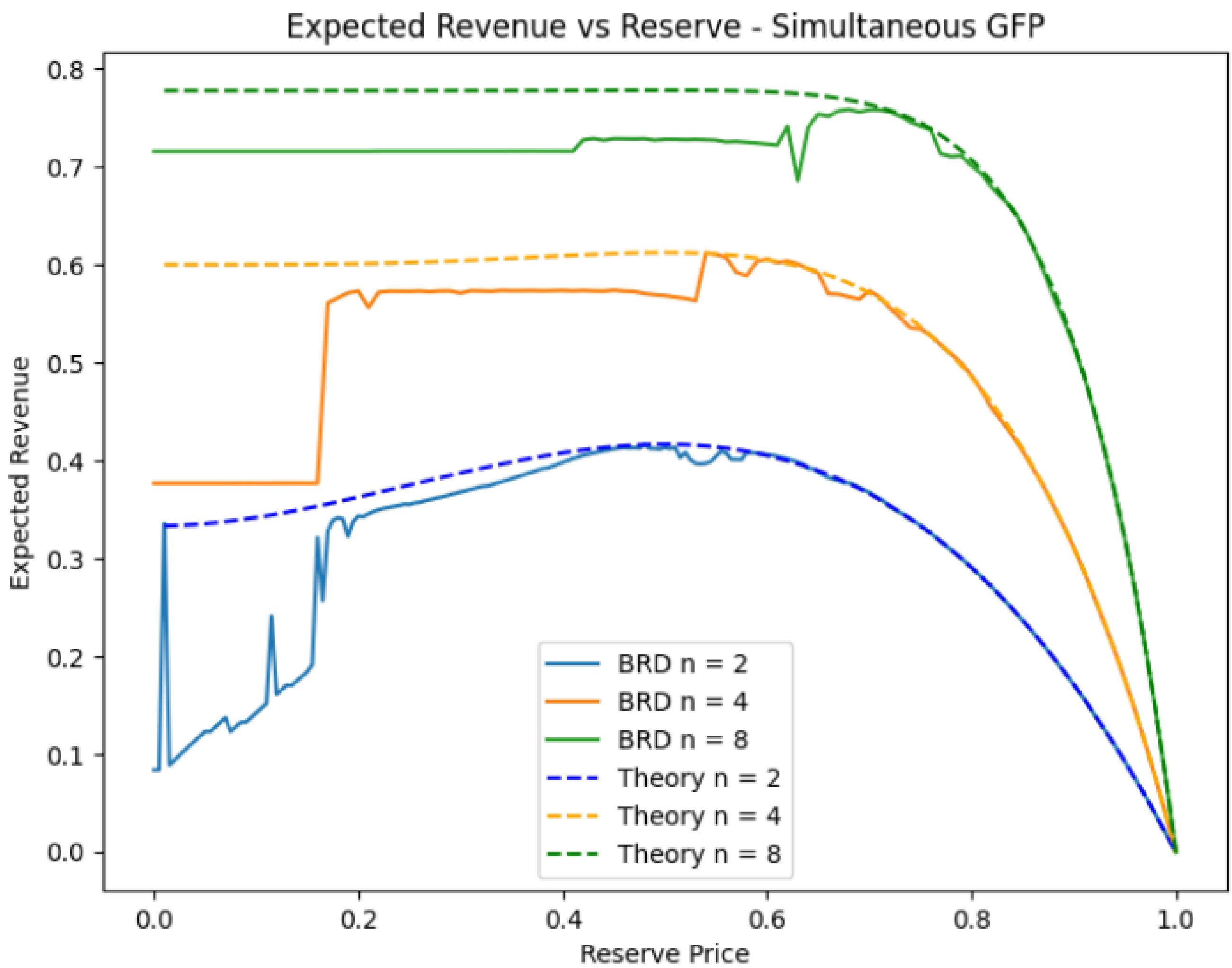

Figure 8: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using simultaneous learning in GFP. Notice the additional noise, particularly for smaller reserve. This is due to the cycling behavior as seen in the previous section being more prevalent for small reserves, as they get stuck in low revenue cycles. Increasing the reserve incentivizes bidders to shade less, so the expected revenue for higher reserves follows that of the theoretical.

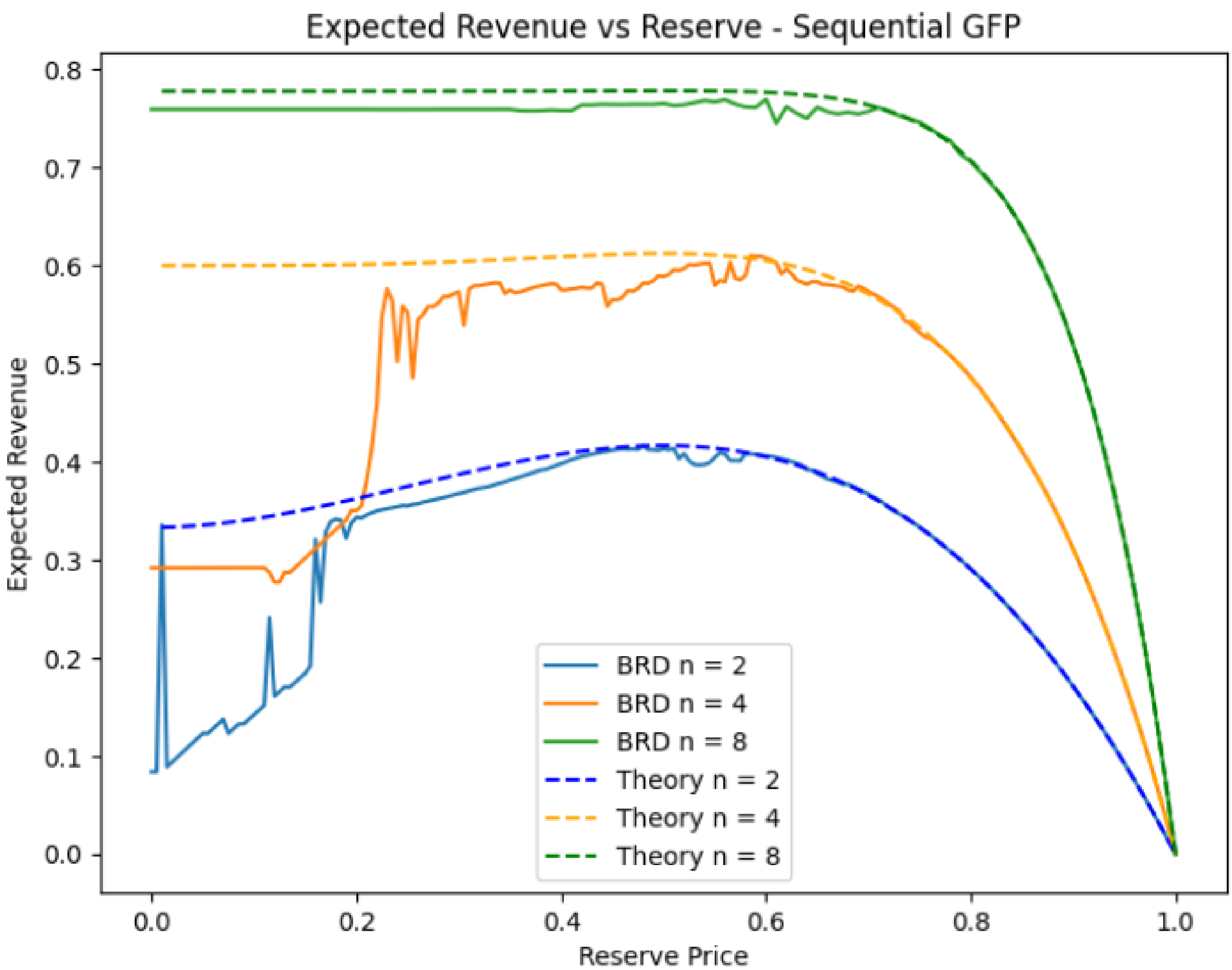

Figure 9: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using sequential learning in GFP. Much like in the simultaneous case, cycling behavior as seen in the previous section being more prevalent for small reserves, as they get stuck in low revenue cycles. Increasing the reserve incentivizes bidders to shade less, so the expected revenue for higher reserves follows that of the theoretical. It is worth noting that for a larger number of bidders, the effect of the cycling on expected revenue is lessened in the case of sequential versus simultaneous bidding.

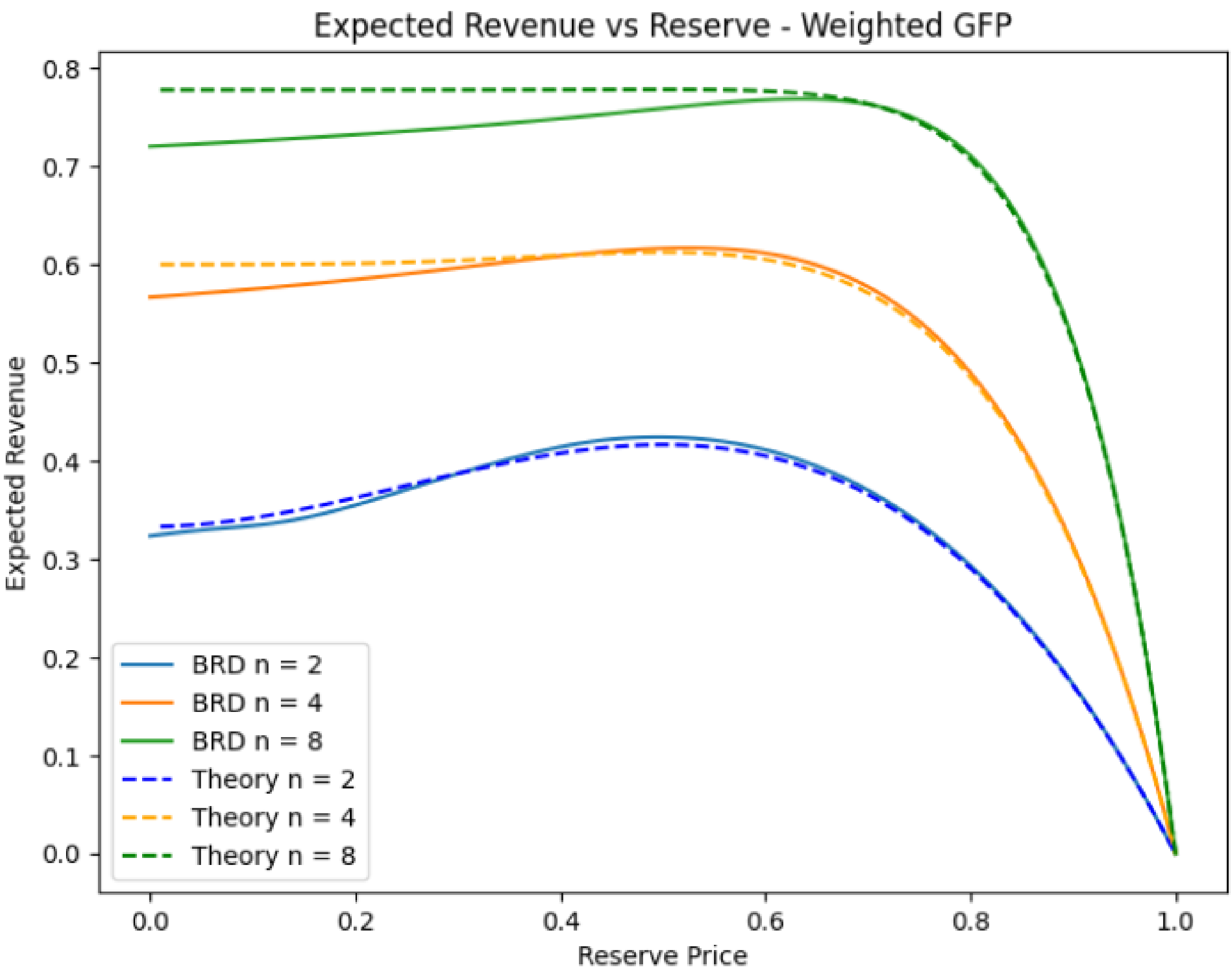

Figure 10: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using weighted learning in GFP. Unlike in the simultaneous and sequential setting, the revenue here almost traces that of the equilibrium revenue. This is due to the near-convergence of the bidding behavior to the equilibrium bidding behavior, for example, in the case for $\pi = \frac{1}{2}$ as we saw earlier.

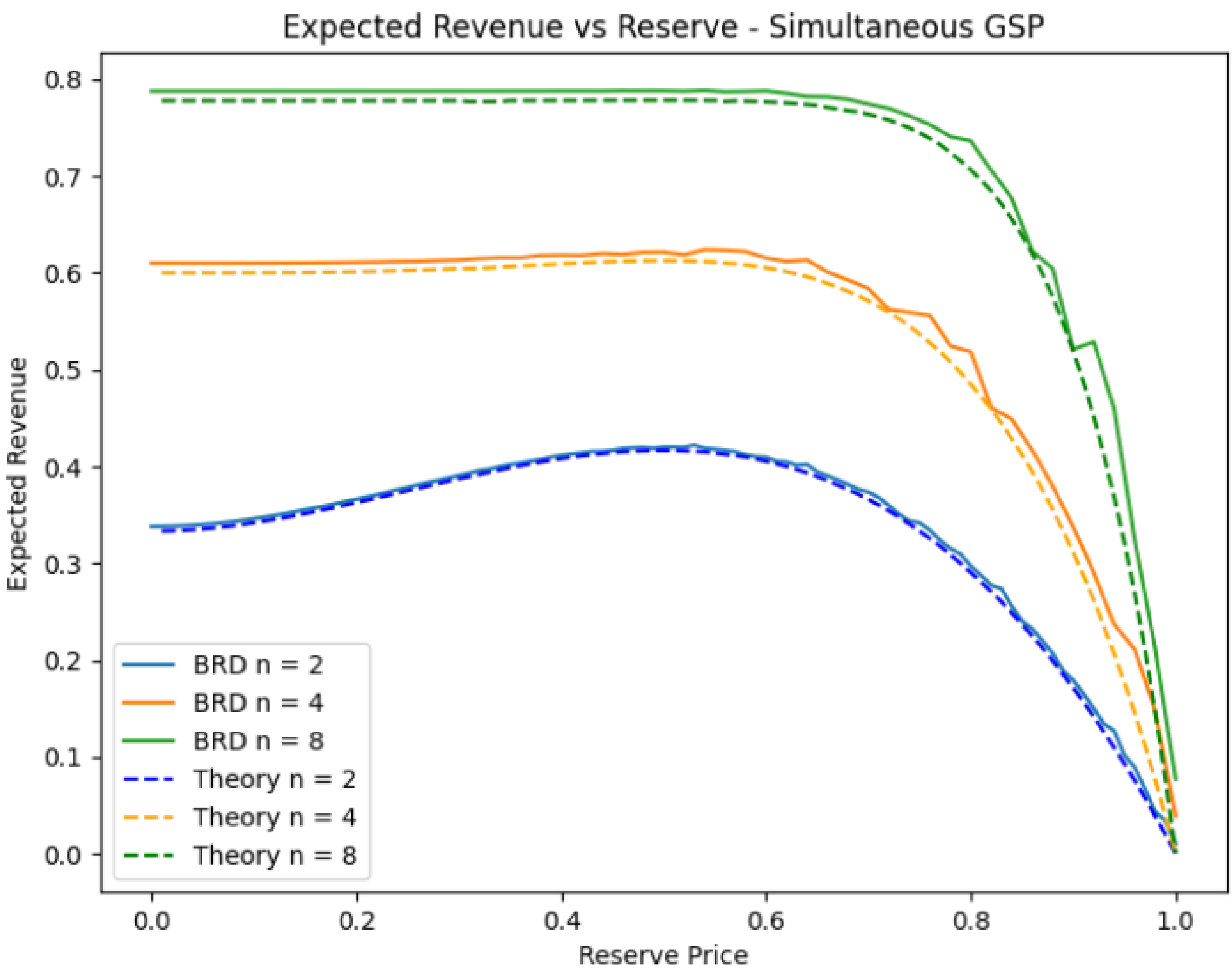

Figure 11: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using simultaneous learning in GSP. As we have shown the bidder behavior converges to the theoretical in the simultaneous setting (at least for $\pi = \frac{1}{2}$, it is not surprising to see the BRD recovering the revenue curve. Note that due to implementation specifics, bids in BRD, SBRD, and WPS are rounded up to the nearest discretization factor, which explains the slight overshoot in revenue.

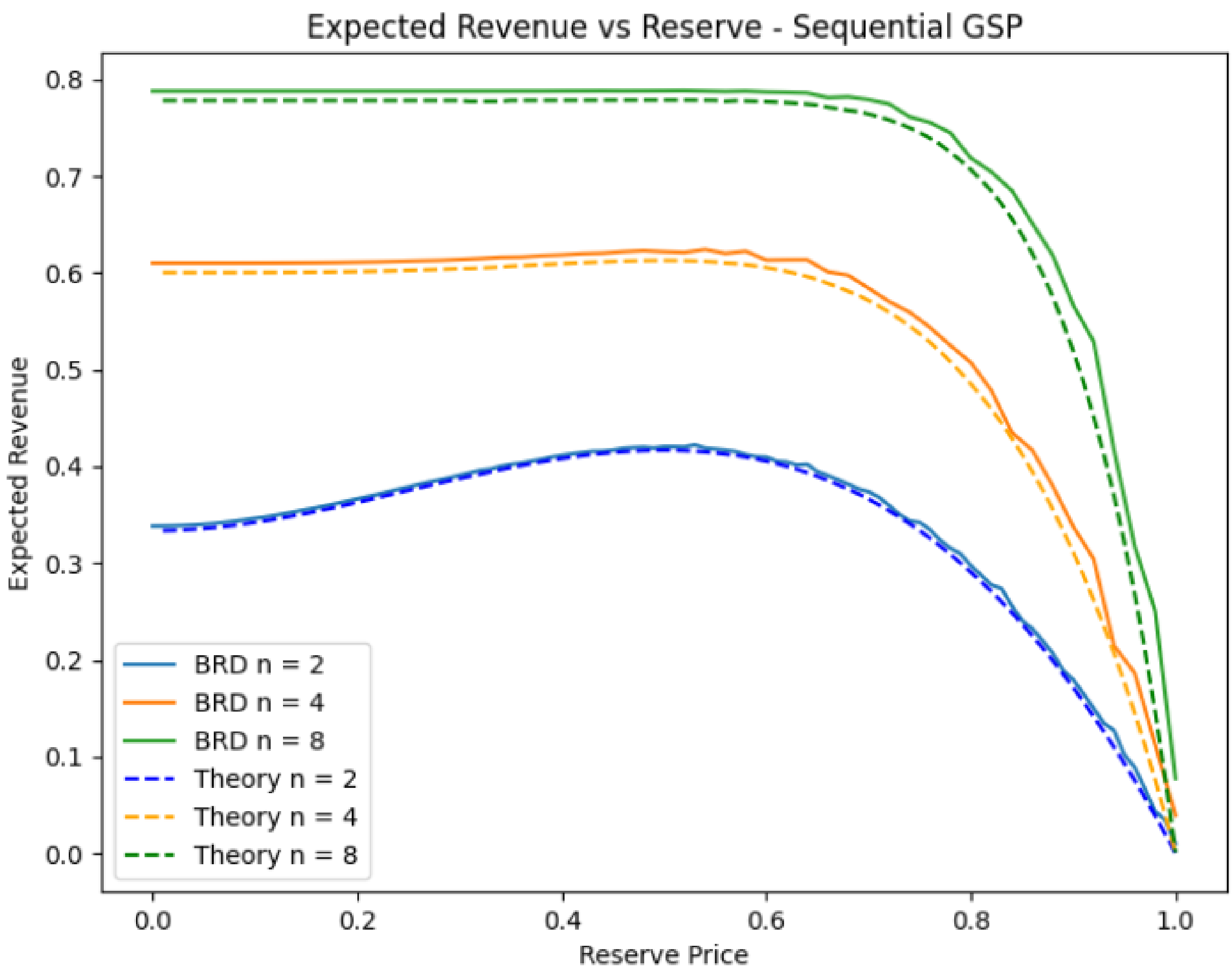

Figure 12: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using sequential learning in GSP. Like above, we have shown the bidder behavior converges to the theoretical in the sequential setting.

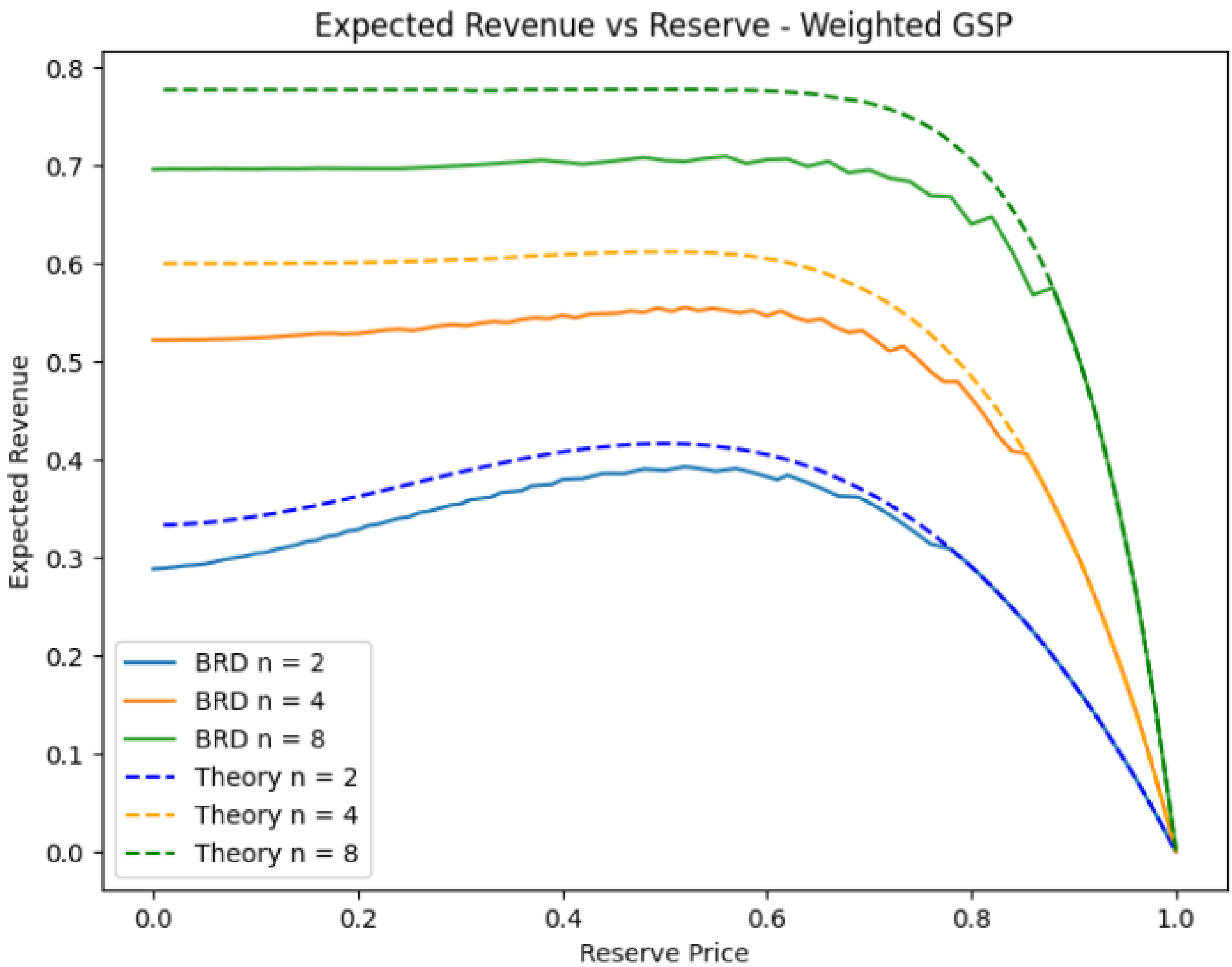

Figure 13: Shown above is the reserve vs. expected revenue for each of $n = 2, 4, 8$ using weighted learning in GSP. As the bidding distributions don't converge nicely to the BNE (at least in the $\pi = \frac{1}{2}$ case), then it is not surprising that we cannot recover the revenue curve though the shape is similar.

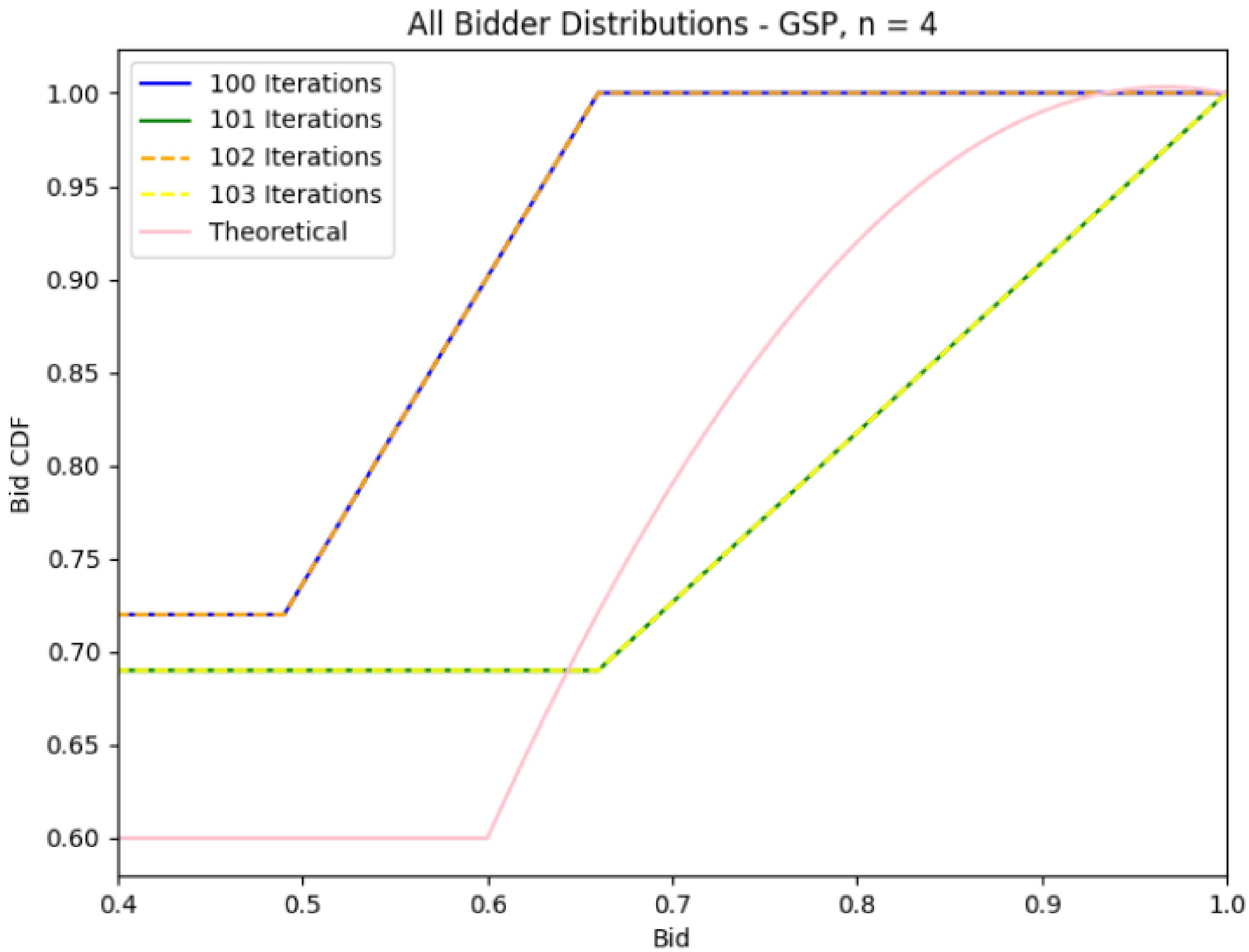

Figure 14: We run BRD on GSP for 100, 101, 102, and 103 iterations at $\pi = \frac{1}{2}$, $n = 4$ respectively to see that the bidder behavior cycles in periods of 2. Note that the theoretical equilibrium here is again seemingly off due to the scaling issue with $\alpha_1 = 0.8 < 1$.

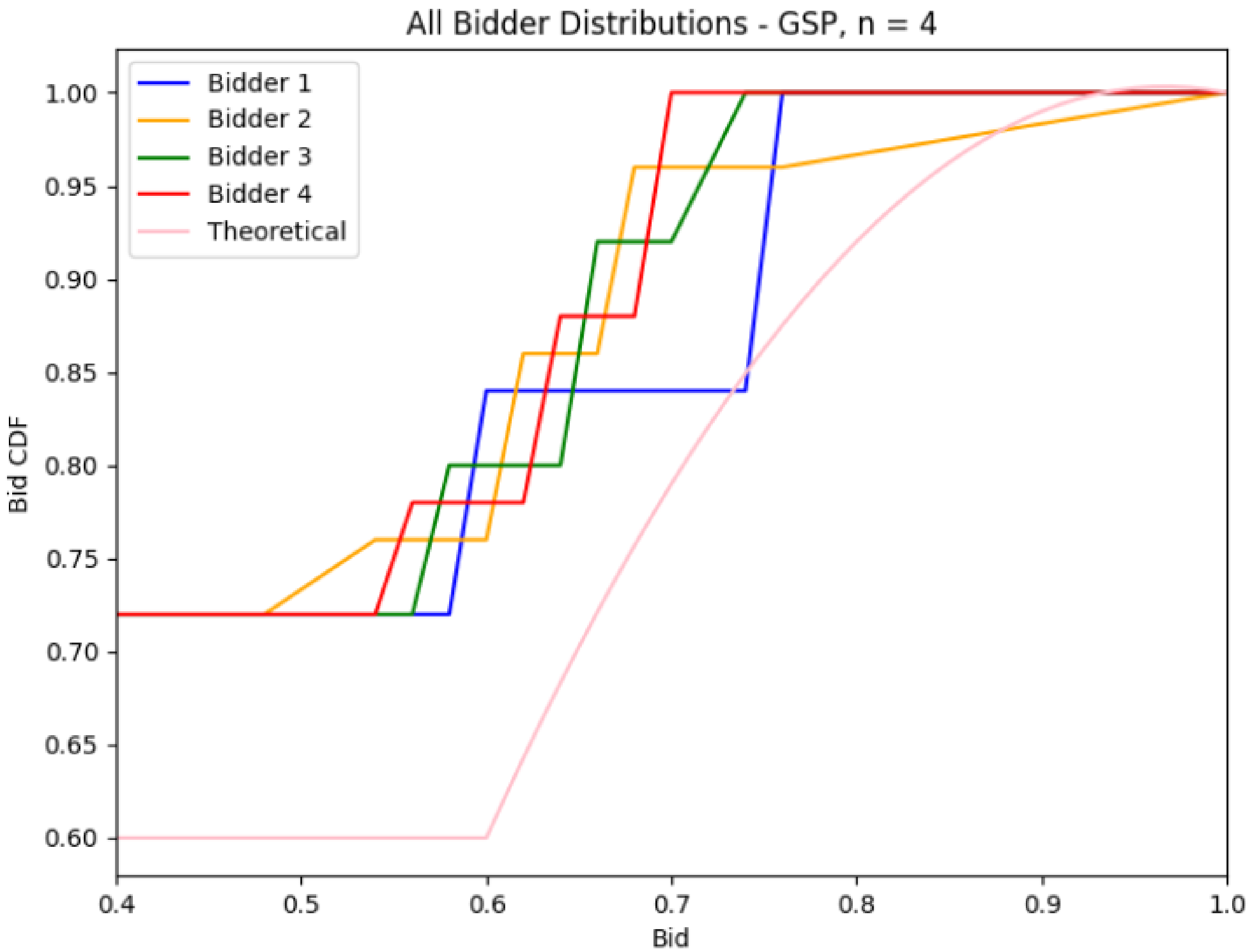

Figure 15: We run SBRD on GSP for 100 iterations at $\pi = \frac{1}{2}$, $n = 4$ and the bidder distributions do not converge, nor are they symmetric. The distributions are fairly close to the theoretically predicted and the pattern extends to the $n = 2$ and $n = 8$ case, though are the bids are generally smaller than the theoretical. As such, we will be using this learning style to plot the expected revenue versus reserve. Note that due to the scaling factor issue with $\alpha_1 = 0.8 < 1$, the theoretical distribution is significantly different.

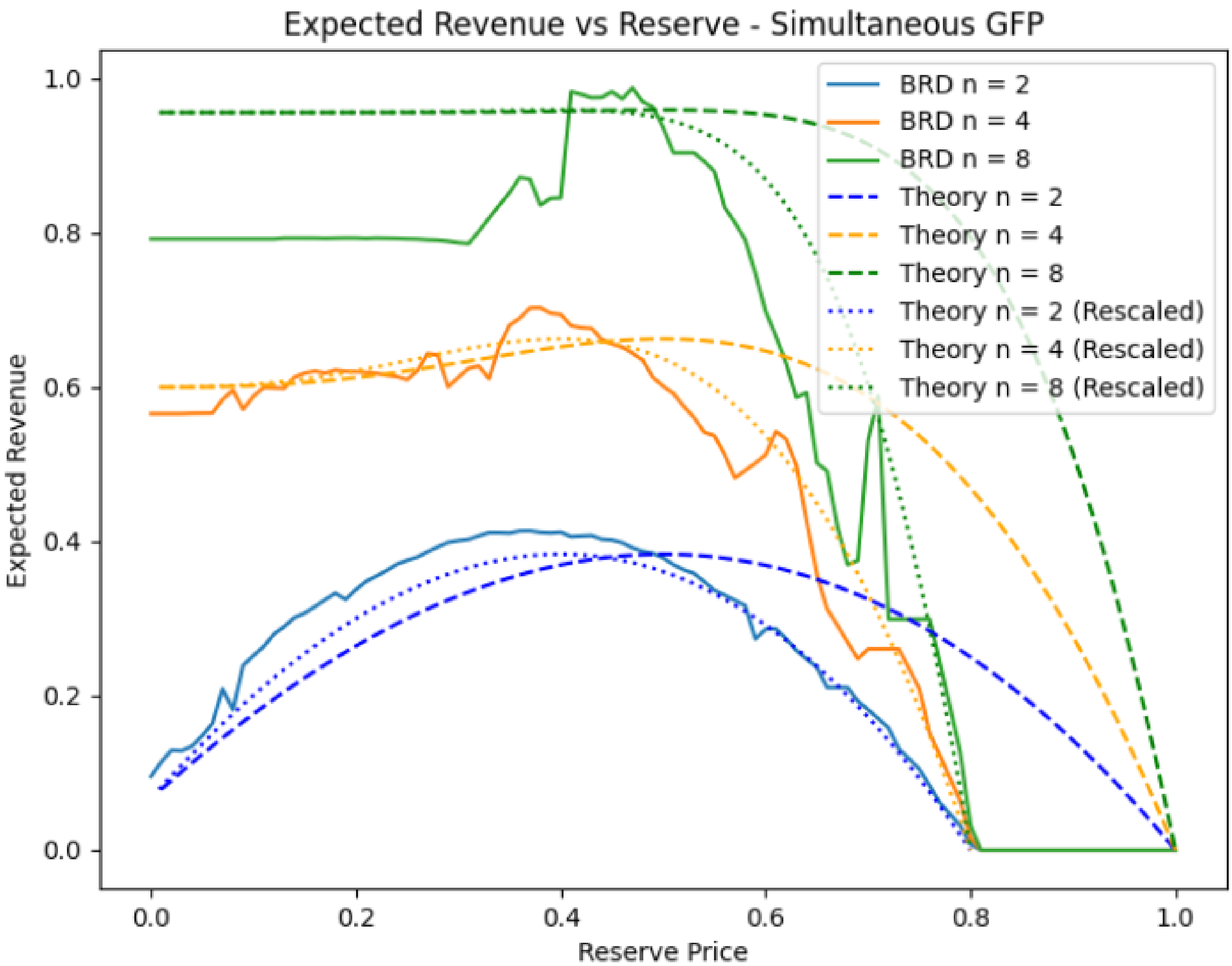

Figure 16: GFP, simultaneous learning. We note that we once again observe noise due the cycling behavior as we saw in the GFP, one item, and simultaneous learning setting. The obtained curves is a horizontally-scaled (by roughly 0.8) version of the theoretical revenue, which is due to the assumption that $\alpha_1 = 1$ in the papers in which they were derived. It is also apparent that the cycling behavior actually worsens as the number of bidders increases.

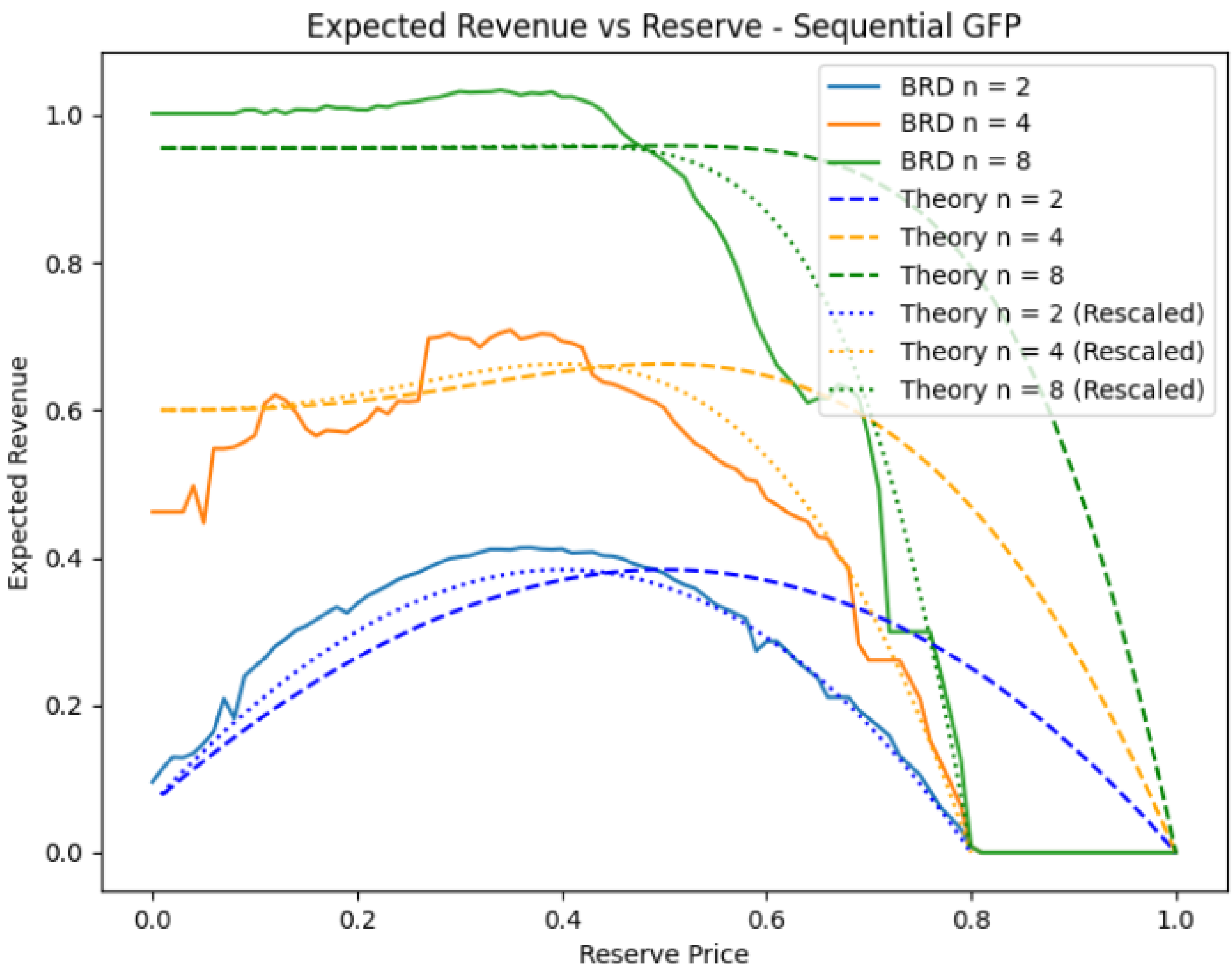

Figure 17: GFP, sequential learning. Like above, the cycling in GFP yields noisy expected revenue curves; though the effect is less apparent than in simultaneous learning. Once again, we see that the recovered revenue curves are a horizontally scaled version (also by 0.8) version of the theoretical revenue curves.

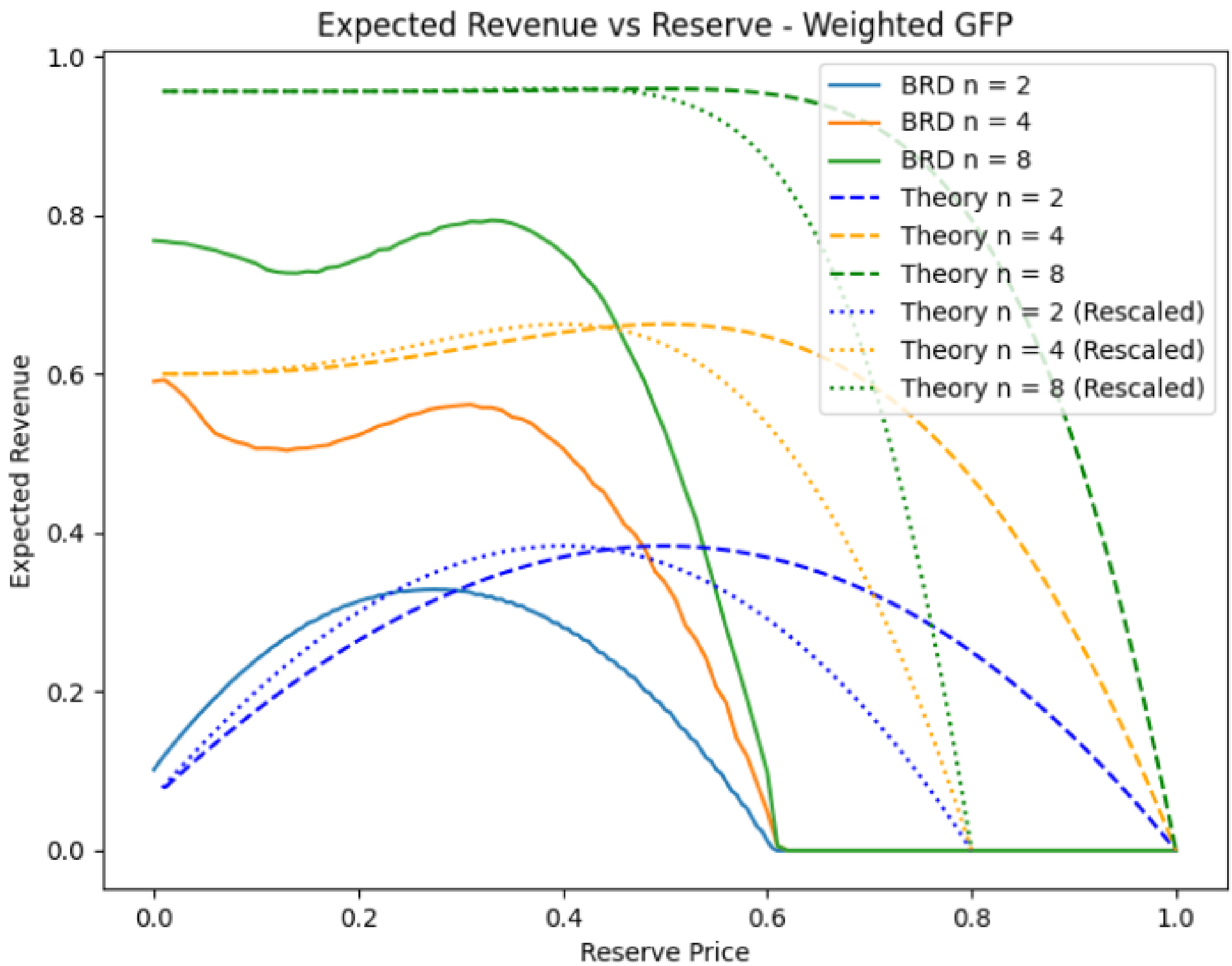

Figure 18: GFP, weighted learning. Unlike the revenue curves above, the revenue curves terminate at 0.6, though the shapes of the curves vaguely resembles that of the theoretical.

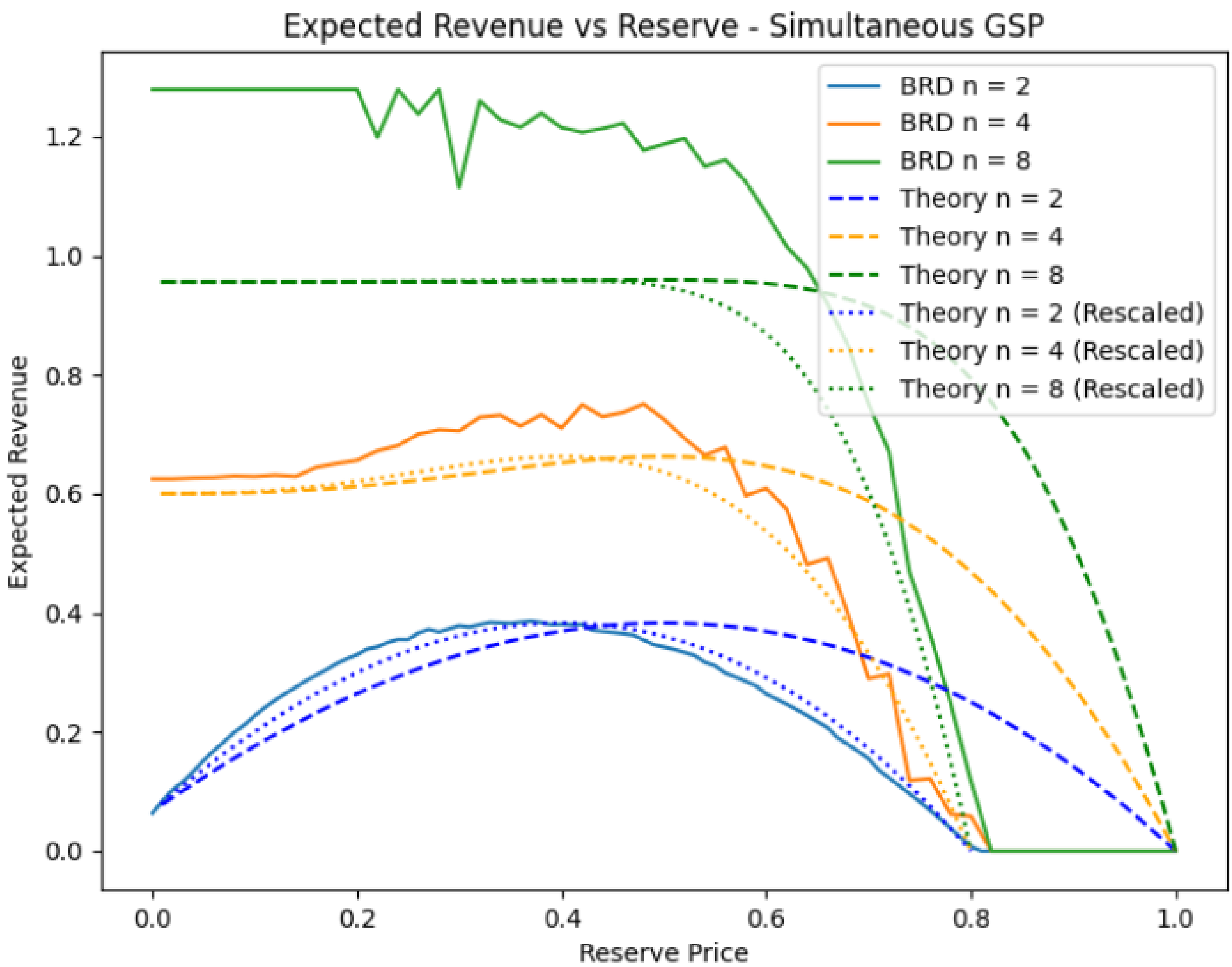

Figure 19: GSP, simultaneous learning. Here, like in the GFP simultaneous, we have cycling behavior which yields noisy revenue curves and a similar horizontal scaling factor of 0.8. The curves are noticeably less noisy than that of the GFP, though, and but also overshoot (rather than undershoot) the theoretical equilibrium revenues.

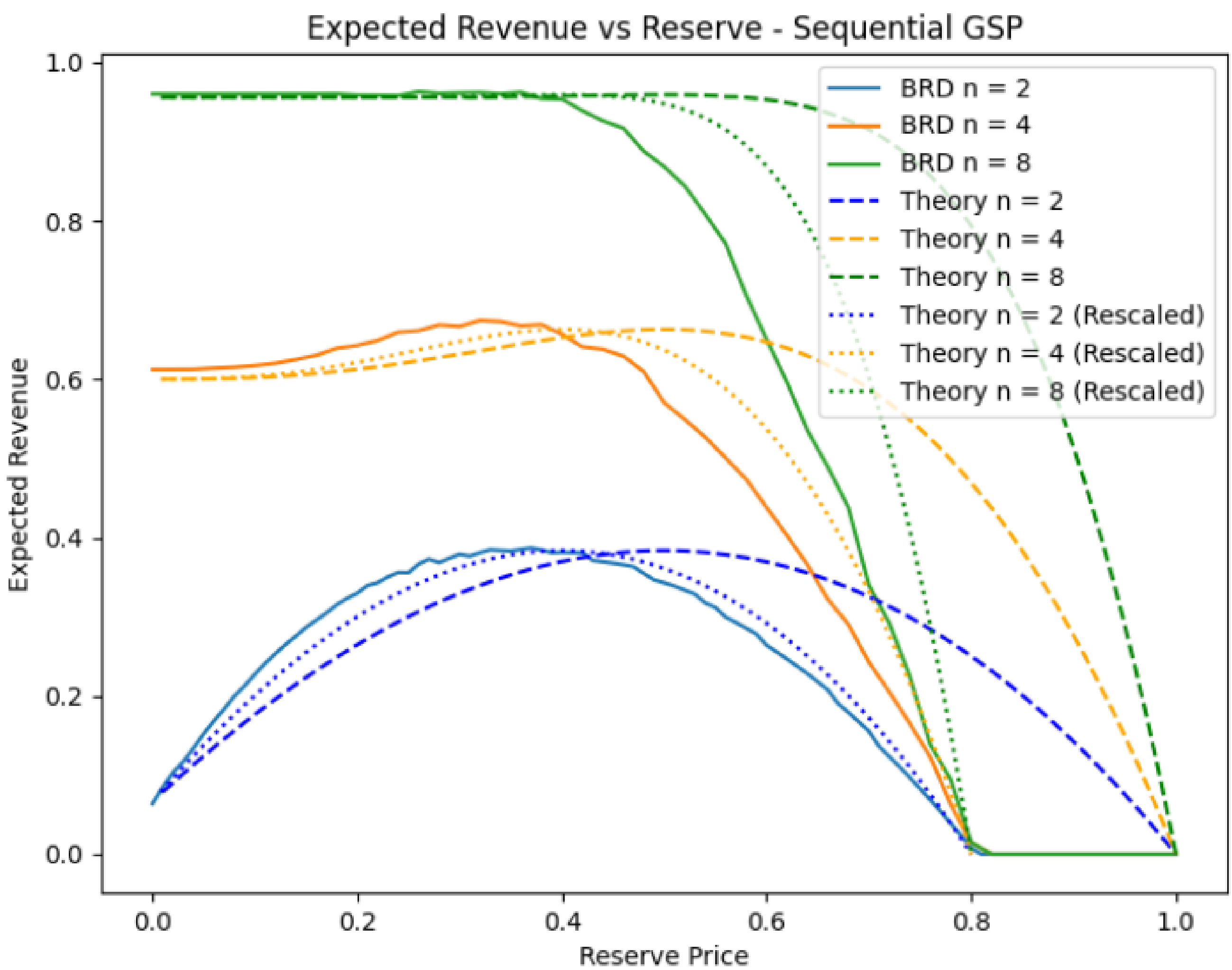

Figure 20: GSP, sequential learning. This is the more well recovered revenue curves (up to the scaling constant of 0.8) as despite the bidder behavior cycling and not converging to a symmetric equilibrium, the bidder's behave very similarly suggesting that a symmetric equilibrium is nearby.

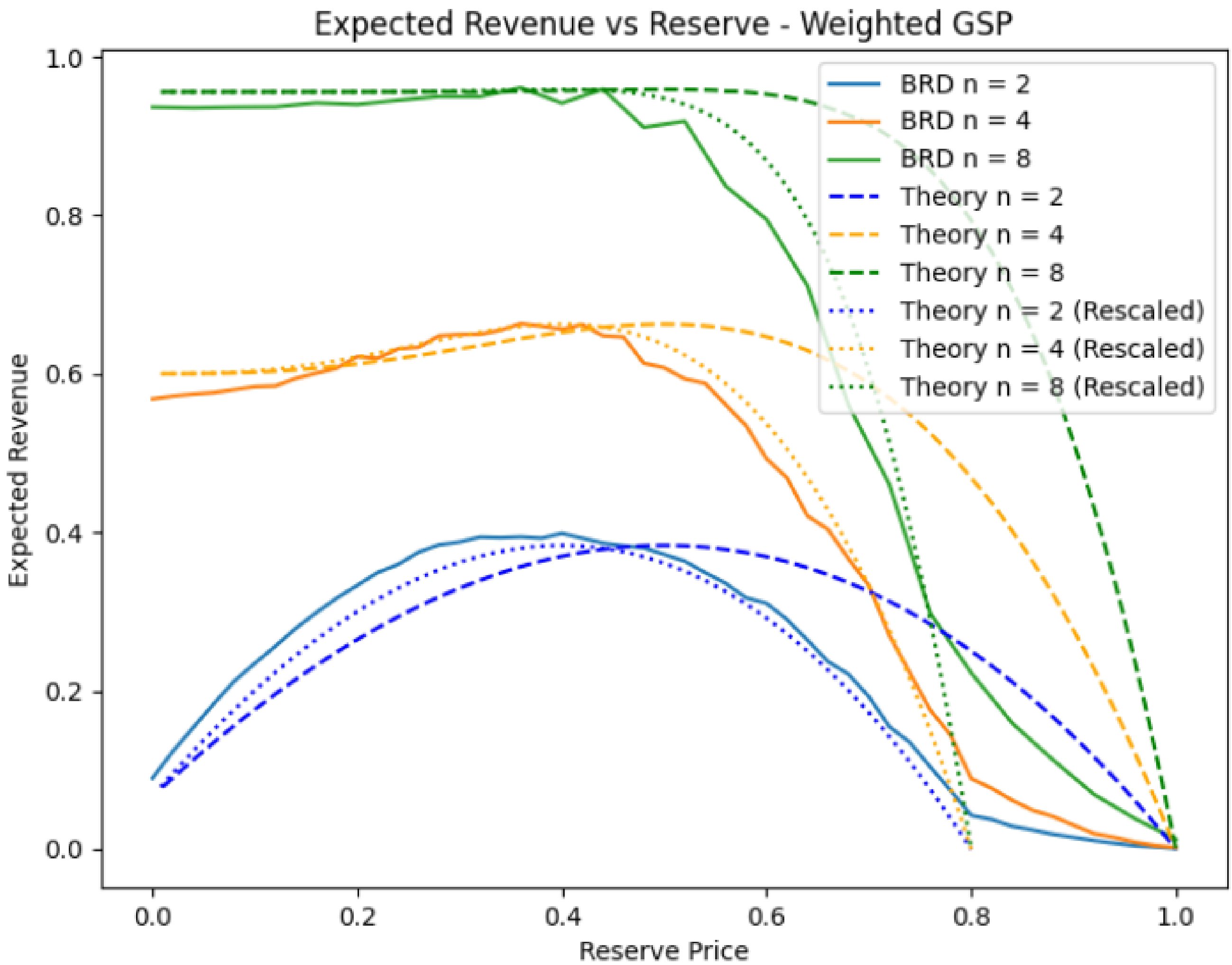

Figure 21: GSP, weighted learning. Even more so than the sequential learning version, weighted learning almost recovers the theoretical revenue curves (up to a scaling constant of 0.8). Furthermore, it immediately recovers a symmetric bidding distribution, though due to the exponentially weighted scheme, puts some non-negative mass in the bidder distributions above 0.8 which results in having positive revenue past a reserve of 0.8.

6 Conclusion

6.1 Summary

We have studied the GFP and GSP from the standpoint of best response dynamics and two slight variants involving sequential and weighted learning. In an attempt to recover some of the theoretical results as proven in [6] and [10], we have ran several experiments with varying numbers of bidders and the bidding distribution learner. Using novel, fast algorithms to compute the joint and conditional distributions of order statistics, we are able to exactly characterize the utility of bidders and revenue of auctions. Listing the following observations made in our experiments:

1. Simultaneous learning is the most prone to bidder behavior cycles, leading to valuations pooling over bids. This is due to the discretized nature of the valuation and bid space. As a consequence, the resultant revenue distributions are noisy and initialization dependent, particularly so in GFP. This pooling behavior—valuations mapping to the same bid—is not completely unrealistic; as some advertisers may not necessarily have a smooth valuation to bid utility maximizing-function. Instead, one may prefer to simplify their bid function and cluster together certain valuation ranges and map to a single bid. For example, a women's fashion company may bid some high value if the user is known to be female, bid some intermediate value if unsure of the user's gender, and bid low if the user is known male.

2. Sequential learning, while also prone to bidder behavior cycles, converges more closely to the theoretical distributions than simultaneous learning. In particular, the differences in the cycling behavior and bidder asymmetries in sequential learning are smaller than that of simultaneous, which yields a smaller noise on the revenue curves.

3. Weighted learning consistently yields convergent symmetric distributions, which seems to agree with the guaranteed existence of mixed-Nash equilibria, but not necessarily Nash equilibria. Using a weighted scheme, while is not technically best response dynamics, almost recovers the theoretical BNE in the GFP $m = 1$, and GSP $m = 1, 2$ cases. As this seems to converge to symmetric distributions, this learning scheme may be the best suited to estimating revenue curves for more complicated settings, such as larger n, m, k, asymmetric α's or valuation distributions.

4. The theoretical BNE optimal reserve price of 0.5 in the case of $m = 1$ and 0.4 for $m = 2$ was approximately recovered in all learning schemes, except for GFP $m = 2$, weighted. This is an interesting result in and of itself considering the non-convergence of many of learning schemes to that of the BNE, suggesting that there may be a connection between the optimal reserve within bidder behavior cycles and that of the true optimal reserve.

6.2 Future Work

On the topic of generalizations to higher dimensions, a fairly natural extension of our experiments is to that of the case of asymmetric click through rates, valuation distributions, non monotone hazard rate valuation distributions, or non-uniform reserve prices. Our machinery does not rely on any of these assumptions and hence can be used to estimate expected revenue and optimal reserve price (vectors) for more complicated auction settings. In the opposite direction as our computational efforts, the theoretical convergence properties of the various learning schemes remains to be shown and is of considerable interest on its own. As earlier mentioned, there may be a mathematical connection between the existence of a mixed Nash equilibrium and the symmetric convergence of the weighted learning scheme. Similarly, the reasons for the non-convergence of and valuation pooling phenomenon of the simultaneous and sequential best response dynamics is not well understood, though we speculate that this is a result of discretization. Additionally, the following few questions are also of interest:

1. Under what initial conditions do the various learning schemes settle into equilibrium? Symmetric equilibrium?
2. Does discretization necessarily lead to pooling behavior or will finer discretization eliminate this issue?
3. From the perspective of revenue and the optimal reserve, what statistical properties do the cycles in the simultaneous and sequential learning settings have?

References

[1] B. Bapat and I. Beg. Order statistics for nonidentically distributed variables and permanents. Sankhyā: The Indian Journal of Statistics, Series A, 1989.

[2] Charles Boncelet Jr. Algorithms to computer order statistic distributions. SIAM Journal of Scientific and Statistical Computing, 2006.

[3] Richard Cole and Tim Roughgarden. The sample complexity of revenue maximization. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 243–252. ACM, 2014.

[4] Benjamin Edelman and Michael Ostrovsky. Strategic bidder behavior in sponsored search auctions. Decision Support Systems, 2006.

[5] Deborah Glueck, A. Kampour-Fard, J. Mandel, L. Hunter, and K. Muller. Fast computation by block permanents of cumulative distribution functions of order statistics from several populations. Commun Stat Theory Methods, 2008.

[6] Renato Gomes and Kane Sweeney. Bayes–nash equilibria of the generalized second-price auction. Games and Economic Behavior, 2014.

[7] Oliver Hinz, II-Horn Hann, and Martin Spann. Price discrimination in e-commerce? an examination of dynamic pricing in name-your-own price markets. Management Information Systems Research Center, 2011.

[8] Kin-Keung Lai and Ng Wan-Lung. A stochastic approach to hotel revenue optimization. pages 1059–1072. Computers & Operations Research, 2005.

[9] Renato Leme and Eva Tardos. Pure and bayes-nash price of anarchy of generalized second price auction. IEEE Foundations of Computer Science, 2010.

[10] Shulin Liu and Xiaohu Han. Bayes–nash equilibria of the generalized first-price auction. Mathematical Problems in Engineering, 2015.

[11] Brendan Lucier, Renato Pase Leme, and Eva Tardos. On revenue in the generalized second price auction. Proceedings of the 21st international conference on World Wide Web, 2012.

[12] Preston McAfee and Vera Velde. Dynamic pricing in the airline industry. 2006.

[13] Christos Papadimitrou and Georgio Piliouras. Game dynamics as the meaning of a game. ACM Sigecom Exchanges, 2019.

[14] Laszlo Toka, Janos Tapolcai, George Darzanos, and Balazs Sonkoly. On pricing of 5g services. IEEE Global Communications Conference, 2017.

[15] Enrique Viqueira, Cyrus Cousins, Mohammad Yasser, and Amy Greenwald. Empirical mechanism design: Designing mechanisms from data. Uncertainty in Artificial Intelligence, 2019.

[16] Enrique Viqueira, Cyrus Cousins, and Amy Greenwald. Improved algorithms for learning equilibria in simulation-based games. International Foundation for Autonomous Agents and Multiagent Systems, 2020.

[17] John Wicks and Amy Greenwald. An algorithm for computing stochastically stable distributions with applications to multiagent learning in repeated games. Brown University, 2009.