Polynomial-Time Approximation Schemes via Utility Alignment: Unit-Demand Pricing and More*

Robin Bowers†1, Marius Garbea‡2, Emmanouil Pountourakis§2, and Samuel Taggart¶3

1University of Colorado, Boulder
2Drexel University
3Oberlin College

Abstract

This paper derives polynomial-time approximation schemes for several NP-hard stochastic optimization problems from the algorithmic mechanism design and operations research literatures. The problems we consider involve a principal or seller optimizing with respect to a subsequent choice by an agent or buyer. These include posted pricing for a unit-demand buyer with independent values (Chawla et al. [19], Cai and Daskalakis [16]), assortment optimization with independent utilities (Talluri and van Ryzin [53]), and delegated choice (Khodabakhsh et al. [36]). Our results advance the state of the art for each of these problems. For unit-demand pricing with discrete distributions, our multiplicative PTAS improves on the additive PTAS of Cai and Daskalakis [16], and we additionally give a PTAS for the unbounded regular case, improving on the latter paper's QPTAS. For assortment optimization, no constant approximation was previously known. For delegated choice, we improve on both the 3-approximation for the case with no outside option and the super-constant-approximation with an outside option.

A key technical insight driving our results is an economically meaningful property we term *utility alignment*. Informally, a problem is utility aligned if, at optimality, the principal derives most of their utility from realizations where the agent's utility is also high. Utility alignment allows the algorithm designer to focus on maximizing performance on realizations with high agent utility, which is often an algorithmically simpler task. We prove utility alignment results for all the problems mentioned above, including strong results for unit-demand pricing and delegation, as well as a weaker but very broad guarantee that holds for many other problems under very mild conditions.

1 Introduction

We consider algorithms for a family of agent-principal problems from economics and operations research. The problems we consider involve a principal (e.g. a seller) presenting a choice to an agent (e.g. a buyer). The principal offers a set of options to the agent, who selects their favorite.

*This work was supported by NSF awards CCF 2218813 and CCF 2218814.

†robin.bowers@colorado.edu
‡mgarbea@drexel.edu
§manolis@drexel.edu
¶staggart@oberlin.edu

The two main hallmarks of the problems we consider are: (1) the agent's preferences differ from those of the principal, and (2) the agent's preferences are unknown (and modeled probabilistically). The principal's goal is to maximize their expected utility with respect to the agent's choice, in expectation over their uncertain preferences.

Several well-studied problems satisfy the hallmarks above. For example, in pricing problems with a single buyer, the seller chooses a price p_i for each item i. The buyer's choice is dictated by their values for the different items, which are unknown to the seller. The seller seeks to maximize revenue, taken in expectation over the buyer's utility-maximizing choice. Another example is the model of delegated choice of Khodabakhsh et al. [36], inspired by similar classical problems in economics. In the delegated choice problem, a principal presents a menu of options to an agent, who selects their favorite. This captures a broad range of deterministic mechanisms for single-agent problems. Finally, in assortment optimization, a fundamental problem from operations research, a seller chooses an inventory of fixed-price items to stock, again for a single buyer.

We consider versions of these problems where the agent chooses a single option, and where their utilities for the different options are independently distributed. For pricing, this corresponds to the canonical unit-demand pricing problem studied in Chawla et al. [19] and Cai and Daskalakis [16]. For assortment optimization, this single-choice random-utility model generalizes the well-studied multinomial logit model of e.g. Talluri and van Ryzin [53].

Prior to this work, none of the problems discussed above had polynomial-time approximation schemes. The strongest prior approximation results are for unit-demand pricing. For unconditional multiplicative approximation, Chawla et al. [19] give a constant-approximation, which was subsequently improved by Jin and Lu [35]. Meanwhile, Cai and Daskalakis [16] give an additive PTAS for discrete distributions and a multiplicative QPTAS for unbounded distributions satisfying the standard Myerson regularity condition, which improves to a multiplicative PTAS only when the instance satisfies the much stronger condition of monotone hazard rate distributions. We improve these results to a multiplicative PTAS which holds for both discrete and unbounded regular distributions. For delegation and assortment optimization, only a handful of even constant approximations were known, and for some variants of the problems, not even that.

Two main hurdles complicate the design of a PTAS for the problems we consider. First, the typical approach to designing a PTAS is to coarsen the space of values involved, usually by rounding to a grid. For problems where the algorithm's performance is determined by the choice of an agent, this approach is especially problematic. While the designer would like to coarsen the space of agent utilities, even small changes to the agent utility can cause them to switch their choice (e.g. from a high-priced, high-value item to a low-priced but low-value item), leading to large changes in principal utility. For pricing, Cai and Daskalakis [16] resolved this issue with careful discretization schemes. The second hurdle is that even with careful discretization, existing approximation schemes all required enumerating over an exponentially-sized set of objects, such as utility distributions associated with solutions. Indeed, the runtime exponent of the unit-demand pricing algorithm in Cai and Daskalakis [16] depends on the number of discretized prices, which is in turn logarithmic in the number of items, leading to a QPTAS.

To solve these issues, our main technical innovation is to show that for all of the optimization problems we study, the principal can ignore outcomes where the agent's utility is low. This takes the form of a property we call *utility alignment*. In more detail, utility alignment states that in the optimal solution, most of the principal's utility comes from realizations that have a strong quantile in the agent's utility distribution. Since the principal and agent tend to get high utility from the same set of realizations, their utilities are aligned. Utility alignment implies that near-ties are not important for the principal utility: if the agent's utility for different options is independent, then the probability that multiple of these utilities fall in the same high-quantile subinterval is very small: e.g. the probability that action i and j both fall into the top $1/n$ quantile of the agent's utility distribution is at most $1/n^2$. Hence, near-ties mostly occur at low quantiles in the agent's utility distribution, and by utility alignment, these quantiles are unimportant to the principal. Algorithmically, this means that many of the probability computations in the algorithm can be coarsend to first-order approximations that ignore ties, and can be optimized without enumerating an exponentially-sized set. The rest of the introduction makes these ideas and problems above more precise.

1.1 Utility Configuration

The problems mentioned above are all special cases of a general template problem, which we call *utility configuration.* In utility configuration, a principal (e.g. the seller) considers a set of n possible actions that the agent will subsequently choose from. Each option can be "configured" one of several ways, corresponding to setting the price of an item, or including/excluding an action from a delegation set. This configuration then determines the distribution over agent and principal utilities for this item. Given the configurations, the agent realizes their utilities (and hence the principal utilities as well) and picks their favorite action. We define the problem below, before defining pricing, delegation, and assortment optimization, and showing that these are special cases.

Definition 1 (Utility Configuration). *An instance of the* utility configuration problem *is given by n* actions. *Each action i has at most m* configurations. *Configuration j for action i induces a joint distribution F_{ij} over pairs (u_i^A, u_i^P), of agent and principal utilities. A solution to the utility configuration problem (a* configuration*) is a choice $\mathbf{C} = (C_1, \ldots, C_n)$ of configuration $C_i \in \{1, \ldots, m\}$ for each action i. Given a configuration $\mathbf{C}$, the agent draws (u_i^A, u_i^P) independently from F_{iC_i} for each action i. Denote their favorite action given the realized utilities by the random variable $i^*(\mathbf{C}) = \arg\max_i u_i^A$. The principal's objective is to maximize their expected utility, given by $\mathbb{E}[u^P_{i^*(\mathbf{C})}]$.*

Remark on notation. When context makes the choice of configuration clear, we often omit the subscript $i^*(\mathbf{C})$ to denote the agent's preferred choice, and instead denote the utilities from the agent's favorite choice by $u^A = u^A_{i^*(\mathbf{C})}$ and $u^P = u^P_{i^*(\mathbf{C})}$.

For computational purposes, we consider *discrete* instances of utility configuration, where the distributions F_{ij} are discrete distributions with finite supports for every i and j, given as input in the form of a list of mass points with respective probabilities. In principle, though, it is also natural to consider variants with continuous utility distributions, as we do for unit-demand pricing at certain points in the paper. To enable approximation, we restrict to nonnegative principal utilities $u_i^P \geq 0$, but do not need to place any such restriction on agent utilities, as only the order of agent utilities matters to the principal.

We now formalize the three problems discussed in the introduction — pricing, delegation, and assortment optimization — and show how they can be cast as utility configuration problems.

Delegated Choice. In the delegated choice problem of Khodabakhsh et al. [36], a principal must select a subset $S \subseteq \{1, \ldots, n\}$ of actions to present to an agent. The agent realizes their utilities for the actions, and chooses their favorite. Khodabakhsh et al. [36] consider a simple utility model, where the principal's utility is given by v_i, which is unknown to the principal and distributed according to some F_i, and the agent's utility is $v_i + b_i$, for a bias term which is constant for each i and known to the principal. This can be cast as utility configuration in the following way: each action i can be configured as *in*, in which case $(u_i^A, u_i^P) = (v_i + b_i, v_i)$ with $v_i \sim F_i$, or *out*, in which case $(u_i^A, u_i^P) = (-\infty, 0)$. One can define further variants of this problem, e.g. where the biases b_i are also random, or where the agent has an outside option, an action 0 that cannot be excluded by the principal. We consider these additional variants in Section 4, and reduce them to utility configuration as well.

Unit-demand Pricing. In the unit-demand pricing problem, a seller prices n items for consideration by a unit-demand buyer. The buyer's value v_i for each item i is drawn independently from a known distribution F_i. Given a pricing $\mathbf{p} = (p_1, \ldots, p_n)$, the buyer picks the item that maximizes their utility $v_i - p_i$ (yielding revenue p_i for the seller). If no item gives positive utility, they choose not to purchase, yielding revenue 0 for the seller.

This can be framed as a utility configuration problem in the following way: each item i corresponds to an action, and each possible price p_i for item i corresponds to a configuration for that item. Notice that this implies an infinite number of configurations per item. While this is well-defined according to Definition 1, it clearly poses computational issues, which we show how to circumvent in Section 5. In the configuration for action i corresponding to price p_i, the utilities are given by $(u_i^A, u_i^P) = (v_i - p_i, p_i)$ if $v_i - p_i \geq 0$, or $(u_i^A, u_i^P) = (0, 0)$ if $v_i - p_i < 0$, with $v_i \sim F_i$.

Assortment Optimization. In assortment optimization, a seller chooses a subset $S \subseteq \{1, \ldots, n\}$ of fixed-price items to stock for a unit-demand buyer, who wishes to purchase at most one item. Item i has price p_i, and yields revenue p_i to the seller if i is purchased. While there is an extensive literature solving this problem for a dizzying range of choice models for the buyer, we consider a simple *utility-based model* with independent utilities, which is natural to state and generalizes the best-known multinomial logit model (Talluri and van Ryzin [53]) as well as variants (Aouad et al. [5]). Using notation similar to unit-demand pricing, we have the following: for each item i, the buyer draws a value v_i independently from a distribution F_i. They select the item maximizing $v_i - p_i$ from the set S of available items. One can consider a variant of the problem where the buyer chooses not to purchase if $v_i - p_i < 0$ for all $i \in S$. In this case, for $i \in S$, we have $(u_i^A, u_i^P) = (v_i - p_i, p_i)$ if $v_i - p_i \geq 0$, or $(u_i^A, u_i^P) = (0, 0)$ if $v_i - p_i < 0$, and for $i \notin S$, $(u_i^A, u_i^P) = (-\infty, 0)$. It is more common in the literature, however, to consider the no-purchase option as yielding random utility u_0 for the buyer. We discuss how to incorporate this variant in Section 4.3.3.

1.2 Utility Alignment and Main Results

We give polynomial-time approximation schemes for all problems mentioned in the previous section. In fact, we give a single algorithm, an approximation scheme for utility configuration, which can be applied via the reductions described above. The algorithm's runtime is polynomial in the size of the utility configuration instance, and its performance is parametrized by a fineness parameter M (which also governs the polynomials in the runtime) and the *utility alignment* of the instance, which is a parameter we define formally below. Intuitively, a utility configuration instance is utility-aligned if, in the optimal solution, the agent and principal derive most of their utility from the same realizations. More precisely, when we consider the bottom q-quantile of the agent's utility distribution in the optimal configuration, it has to be that this quantile range does not constitute too much of the principal's overall expected utility. Equivalently, the principal's expected utility conditioned on this range of agent utilities should not be too much higher than their expected utility overall. Formally, we define utility alignment as follows:

Definition 2 (Utility Alignment)**.** *Given a non-increasing function f, an instance of utility configuration is* $f(q)$-utility aligned *if for any utility threshold U such that in the optimal configuration $\mathbf{C}^*$, $Pr[u^A \leq U] = q$, the principal utilities under $\mathbf{C}^*$ satisfy*

$$\mathbb{E}[u^P \mid u^A \leq U] \leq f(q)\mathbb{E}[u^P],$$

where u^A and u^P are the agent and principal utilities from the agent's preferred action under $\mathbf{C}^*$.

For intuition, first consider delegation and unit-demand pricing. In Sections 4.1 and 4.2, we prove the following:

Lemma 3. *The delegation problem (with deterministic biases and no outside option) and unit-demand pricing are both* 2*-utility aligned.*

Consequently, under the optimal solution to each of these problems, the principal's utility conditioned on the bottom q-quantile of agent utilities is no more than twice their overall expected utility. Equivalently, the contribution to their overall utility from the bottom q quantile range is at most a $2q$-fraction overall. For example, the bottom 10% of agent utilities can contribute at most 20% of the principal's optimal utility. We emphasize that utility alignment is a property of optimal solutions: it is easy to find non-optimal solutions with extremely misaligned utilities, e.g. where nearly all of the principal's utility comes from the agent's least favorite realizations.

For a more complicated example, we prove the following theorem in Section 4.3, which we state very informally here:

Lemma 4 (Informal)**.** *If agent and principal utilities are positively correlated for each individual action, then the instance satisfies $4/\sqrt{q}$-utility alignment.*

This lemma will hold for the more complicated variants of delegated choice discussed above, as well as for assortment optimization with independent utilities. Notice that $4/\sqrt{q}$-utility alignment is weaker: it implies that, conditioned on the very weakest quantiles of the agent's utility, the principal's utility may be very large. However, it still implies that not too much of the principal's utility comes from low agent quantiles. For example, for $q = .01$, no more than $q \cdot 4/\sqrt{q} \cdot \mathbb{E}[u^P] = .01 \cdot 4/.1 \cdot \mathbb{E}[u^P] = .4 \cdot \mathbb{E}[u^P]$ can come from the bottom q-quantile, i.e. the bottom 1% of agent utilities can generate no more than 40% of the principal's utility.

The performance of our approximation scheme is parametrized by the level of utility alignment of the utility configuration instance. At a high level, the algorithm coarsens the quantile space of agent utilities into M bins, then uses additive approximations to compute probabilities and ensure that the agent's utility falls into each bin with probability approximately $1/M$. It also maintains additive estimates of the principal's expected utility conditioned on each bin. These additive estimates will be inaccurate for low quantiles of agent utility, but accurate for high ones. Utility alignment then implies that these larger errors from low quantiles do not seriously harm the algorithm's performance. We obtain the following result, stated informally.

Theorem 5 (Informal)**.** *Let M be a positive integer. There is an algorithm that runs in polynomial time for any utility configuration instance, such that if the instance is 2-utility aligned, the algorithm is a $(1 - O(\log(M)/M))$-approximation, and if the instance is $4/\sqrt{q}$-utility aligned, the algorithm is $(1 - O(1/\sqrt{M}))$-approximation.*

More generally, our approximation scheme is a PTAS for any instances that are $c/q^{1-\epsilon}$-utility aligned for some $c, \epsilon > 0$.

1.3 Structure of the Paper

We discuss previous work on the various problems we consider in Section 2. Section 3 then gives our algorithm for utility configuration, and analyzes its performance as a function of the number of bins (the "fineness parameter") M and the level of utility alignment in the instance. In Section 4, we then consider the applications of our framework. In addition to the problems defined in Section 1.1, we show how to reduce several other variants to the utility configuration problem. For each, we prove a bound on the utility alignment. For all problems except unit-demand pricing, these immediately imply that the algorithm of Section 3 is a PTAS. The one exception is unit-demand pricing. Because the set of prices is a continuum, a straightforward reduction would yield an infinite number of configurations per action. Instead, we apply a price discretization result from Cai and Daskalakis [16], which implies that polynomially many prices suffice to obtain near-optimal solutions. This then combines with our results to produce a multiplicative PTAS for the discrete version of unit-demand pricing. Along the way, we also show how to use the value discretization and truncation results from that same paper to obtain a PTAS for the case of unbounded regular distributions as well. These results can be found in Section 5.

2 Related Work

Our PTAS applies to a variety of problems, each with its respective body of research. We group our discussion of related work accordingly.

Unit-demand Pricing. Both pricing and more general mechanism design for unit-demand buyers have been important testbeds for algorithmic mechanism design. The most directly related works design approximation algorithms for a unit-demand buyer with independently distributed values. Chawla et al. [19] give a 4-approximation for this problem, which has recently been improved to a 3-approximation in Jin and Lu [35]. Subsequent work by Cai and Daskalakis [16] derives an additive PTAS for the discrete version of the problem, a multiplicative PTAS for the version with continuous distributions that may be unbounded but satisfy the MHR assumption, and a multiplicative QPTAS for the unbounded regular case. Additionally, they prove several powerful truncation and extreme value results that we use in our analysis of unit-demand pricing in Section 5. Our results are the first multiplicative PTAS for both the discrete and unbounded regular cases. On the hardness side, Chen et al. [22] prove that even with independent values, the discrete version of the problem is NP-hard. With correlation across values, Briest and Krysta [15] further prove logarithmic hardness-of-approximation results; our techniques have no bearing on this much harder setting.

Several natural extensions beyond pricing with independent values have been well-studied. In revenue management and operations research, past work such as Aggarwal et al. [1], Rusmevichientong et al. [48] has considered more general demand models. There has also been extensive work on the more general mechanism design/lottery pricing problems, where the principal can populate their menu with distributions over items (also called lotteries) for the buyer to choose from. This work includes Kothari et al. [38], who derive near-optimal mechanisms, and Chen et al. [23], who prove computational hardness. An interesting question is whether our techniques have bearings on any of these settings. Other work has considered multi-buyer versions of the problem, where the seller designs sequential posted pricing mechanisms or more general multi-buyer mechanisms. In these settings, welfare is also a non-trivial objective. These include Chawla et al. [20], Yao [55], Chawla and Miller [18], Cai and Zhao [17], Dütting et al. [28, 27]. Finally, one can consider other families of combinatorial valuations for items, such as additive or subadditive valuations. Some papers on such models include Hart and Nisan [32], Li and Yao [41], Babaioff et al. [8], Rubinstein and Weinberg [47], Chawla and Miller [18], Cai and Zhao [17], Dütting et al. [28].

Delegation. Delegation is a classical problem in microeconomic theory (Holmström [33]). Typical work from economics focuses on deriving explicit optimal delegation mechanisms in settings where the set of actions is the real line, and where the agent and principal have single-peaked preferences with differing biases between the agent and principal. Papers in this vein include Martimort and Semenov [43], Melumad and Shibano [45], Alonso and Matouschek [2], and Amador and Bagwell [3]. The combinatorial model we consider is due to Khodabakhsh et al. [36], who formulate a richer model more amenable to algorithmic analysis. They prove that simple threshold policies are a 3-approximation under independent utilities and no outside option, and give a parametrized super-constant approximation when the agent has an outside option that the principal can't exclude. They also show that even with no outside option, the case with independent utilities is NP-hard, and the case with correlated utilities is APX-hard. We obtain a PTAS for both settings, as well as the more general setting where biases are also random.

Recent work has focused on connections between the delegated project choice model of Armstrong and Vickers [6] and classic optimal search results like prophet inequalities (Krengel and Sucheston [39], Samuel-Cahn [50]) and Pandora's box (Weitzman [54]). These include Kleinberg and Kleinberg [37] and followups Bechtel and Dughmi [10, 11], Bechtel et al. [12]. These papers observe that for delegation problems where the principal can verify their utility from the agent's choice before accepting the agent's selected option, the principal can approximate the utility they would obtain if they controlled the agent's behavior directly. Other papers have included an online variant of the problem (Braun et al. [14]) and a multi-agent variant (Shin et al. [52]). In the Khodabakhsh et al. [36] variant of delegation, the comparisons to first-best made by the papers above are not possible, as the principal is not able to verify their utility for the agent's choice. Nonetheless, several of our technical lemmas, including the utility alignment result for delegation, at a high level resemble arguments from the prophet inequalities literature.

Assortment Optimization. Assortment optimization is a fundamental problem in revenue management, and has been studied under a wide range of demand models for the buyer. The simplest model is the multinomial logit (MNL) model, which can be expressed as a random utility model with independent utilities. Talluri and van Ryzin [53] show that for this restricted class, simple threshold-based policies ("revenue-ordered assortments") are optimal. Other utility models maintain this focus on narrowly-tailored parametrized families: Aouad et al. [5] consider the "exponomial choice model" (but do not give approximation algorithms), which, like MNL, has independent utilities across items. Many other papers introduce correlation: Rusmevichientong et al. [49] consider a mixture over MNL models (corresponding to correlation across utilities for different items), and prove NP-hardness, while Davis et al. [25], Li et al. [40] consider nested variants of the logit model, and consider both algorithms and hardness. No prior work has shown that any model with independent utilities is NP-hard. However, in Appendix A, we show that this in fact follows from a similar argument to one in Khodabakhsh et al. [36]:

Theorem 6. *The assortment optimization problem with independent item utilities is NP-hard, even when the no-purchase option gives the buyer utility zero deterministically.*

No prior work has obtained constant-approximations or better for the case with general independent utilities, either. Our PTAS is the first such result.

Several other papers have gone beyond heavily parametrized utility models. Aouad et al. [4] consider choices generated distributions over rankings (equivalently, fully-correlated utilities), and show that the problem is as hard to approximate as the maximum independent set problem. Rieger and Segev [46] consider the special case of Mallows models for rankings, and give a QPTAS for that problem. Feldman and Topaloglu [30] consider a variant where preferences are drawn by following a Markov chain. Berbeglia and Joret [13] give approximation algorithms for general choice models, with parametrized performance depending on the model. Several other papers have considered versions of assortment optimization with additional constraints or multi-item purchases. These include Barré et al. [9], Feldman et al. [29], Désir et al. [26], Chen et al. [24], Immorlica et al. [34]. The latter two papers are notable for proving hardness results as well. An interesting direction for future work would be extending our results to some of these related models.

Stochastic Probing. At a high level, the problems we consider resemble some of those recently considered in the literature on *stochastic probing*. These papers typically consider an algorithm designer who must choose a limited number of random variables to observe from a larger set. Upon realizing the values of the chosen random variables, the designer picks the one with the best reward. Segev and Singla [51] and Mehta et al. [44] consider a cardinality-constrained version of this problem (the designer must pick k variables), and Goel et al. [31] consider a version with probing costs. Chen et al. [21] consider a version embedded in a bandits problem. These models resemble delegation, except that the decision-maker selecting the variables is different from the one observing the realizations and picking the final reward. Our techniques may have future applications in this literature as well. Two related papers that bear final mention are Mahdian et al. [42] and Azizzadenesheli et al. [7], which consider a regret-minimizing decision-maker who observes only noisy observations of rewards. The decision-maker knows the noise distributions but not the rewards: there are loose connections to the model of delegation with random bias we introduce in Section 4.3.1, where random bias can be viewed as observation error.

3 PTAS for Utility Configuration

This section gives the main PTAS framework for utility configuration. The algorithm takes as input discrete instances of the problem, that is, instances of utility configuration where for each action i and configuration j, the joint distribution F_{ij} over pairs (u_i^A, u_i^P) is given by a list of point masses and their respective probabilities. The algorithm is parametrized by a fineness parameter M, which governs the approximation ratio: the configuration the algorithm computes will be a multiplicative

$(1 - g(M))$-approximation, where $g(M) \to 0$ as $M \to \infty$, with the precise value of g depending on the degree of utility alignment in the instance or family of instances. For example, for utility configuration problems corresponding to unit-demand pricing, we will have $g(M) = O(\log M/M)$. Meanwhile, the runtime of the algorithm will grow with M; the growth will be exponential in M, meaning that the PTAS is not an FPTAS. (However, as is needed for a PTAS, the runtime is polynomial in the input size for any fixed M.)

Before making the PTAS framework fully precise, we give a high-level overview of the main ideas here. We then prove the necessary preliminary results about the key quantities used in the algorithm in Sections 3.1-3.4, before presenting and analyzing the complete algorithm in Sections 3.5 and 3.6. We defer discussion of applications to specific problems of interest (and the corresponding analyses of those problems' utility alignment properties) to Section 4.

The algorithm has an outer loop and an inner optimization procedure. In the outer loop, the algorithm attempts to approximately guess the structure of the distribution of the agent's utility for their favorite action, $\max_i u_i^A$, under the optimal configuration. For each such guess, the inside optimization solves a dynamic program, which maximizes an approximation to the true objective function, subject to the constraint that the distribution of $\max_i u_i^A$ approximately matches the algorithm's guess. The algorithm then returns the best solution obtained from the dynamic program, over all the guesses. We cover each piece in more detail below.

Guessing the agent's utility distribution. In the optimal configuration $\mathbf{C}^*$, each action i's agent utility u_i^A is drawn from a distribution F_{i,C_i^*}. The agent chooses the action that yields the highest utility given the realizations $(u_1^A, \ldots, u_n^A)$, yielding utility $\max_i u_i^A$. For brevity, denote this maximum utility by $u^A = \max_i u_i^A$. The algorithm's outer loop guesses the j/M quantile of the distribution of u^A for every $j \in \{1, \ldots, M\}$. More precisely, we guess utilities $\hat{u}_1 \leq \ldots \leq \hat{u}_M$, which define utility bins $B_j = [\hat{u}_{j-1}, \hat{u}_j)$ for all j, such that for any $j \in \{1, \ldots, M\}$, we have $\Pr[u^A \in B_j] = 1/M$. Assume, for preliminary discussion, an exact partition of the probability space into size $1/M$ bins is possible; with discrete distributions, there will be errors, which we handle in the complete technical exposition.

Optimization procedure. Given a guess of the bins, we now lay out the algorithm's optimization problem. Rather than maximizing the principal's utility exactly, subject to exact constraints on the probability of the agent's utility falling in each bin B_j, the algorithm uses additive estimates. For the bin probabilities, we consider the conditional probabilities $q_j^{\leq} = \Pr[u^A \in B_j \mid u^A \in B_{\leq j}]$, where $B_{\leq j} = \cup_{k \leq j} B_j$. In the optimal configuration, $q_j^{\leq} = 1/j$ for each $j \in \{1, \ldots, M\}$. The algorithm maintains additive estimates of these conditional probabilities. Given a configuration $\mathbf{C}$, each action has a utility distribution of u_i^A and conditional probability $q_{ij}^{\leq} = \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}]$. We then can consider the additive estimate $q_j^{\text{est}} = \sum_{i=1}^n q_{ij}^{\leq}$. Note that if q_j^{est} is close to $1/j$, then each of the individual $q_{ij}^{\leq}$s must be small, and hence $q_j^{\text{est}} \approx q_j^{\leq}$, as first-order terms dominate the calculation of $q_j^{\leq}$. Thus, for B_j corresponding to high quantiles of the agent's utility distribution, requiring q_j^{est} to be close to $1/j$ effectively enforces that the agent's utility distribution is close to that under $\mathbf{C}^*$.

A similar principle applies to the principal's expected utility. We define $u_j^{\leq} = \mathbb{E}[u^P \mid u^A \in B_j] \Pr[u^A \in B_j \mid u^A \in B_{\leq j}]$, where u^P is the principal's utility for the action selected by the agent. Informally, $u_j^{\leq}$ is the principal's expected utility from bin j conditioned on the agent's utility being in bin at most j. Similarly, under configuration $\mathbf{C}$, each action i has a corresponding action-wise $u_{ij}^{\leq} = \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}]$. For similar reasons as with the $q_j^{\leq}$s, the additive estimates $u_j^{\text{est}} = \sum_{i=1}^n u_{ij}^{\leq}$ are a good approximation to $u_j^{\leq}$ when j is large. Moreover, the contribution of an individual action to the overall objective can be approximated as a weighted sum of its contributions to the additive estimates. Consequently, maximizing the additive utility estimate subject to the constraint that the additive probability estimates $q_j^{\leq}$ be close to $1/j$ yields an optimization problem that is structurally similar to the multidimensional knapsack problem, albeit with both upper and lower bounds on weights, and with multiple choices for each item (instead of "in" or "out"). As is natural for knapsack-like problems, we then solve the problem approximately using rounding and dynamic programming.

Analysis. As discussed, q_j^{est} will be close to $q_j^{\leq}$ for high j, and u_j^{est} will be an accurate estimate of the principal's utility as well. With careful accounting, we are able to show that an optimal solution to the optimization above will capture nearly all of the principal's expected utility from high quantiles of the agent's utility distribution. To account for the low quantiles, then, we rely on utility alignment. Utility alignment states that under the optimal configuration $\mathbf{C}^*$, these low quantiles of agent utility contribute little to the principal's objective. In the algorithm, this manifests by down-weighting terms corresponding to low j, and shows that the resulting approximate objective value will be close to the true optimal expected principal utility.

Section outline. The rest of this section precisely defines and analyzes the estimates above, describes the algorithm in full detail, and analyzes the approximation ratio and runtime of the algorithm. In Section 3.1, we first define a preprocessing procedure to handle the discrete nature of the utility configuration instances we consider. This preprocessing step will ensure that it is possible to define utility bins that nearly evenly divide the agent's probability space. Sections 3.2 and 3.3 define the additive estimates for the $q_j^{\leq}$s and $u_j^{\leq}$s, and bound their errors. Section 3.4 then defines the objective function for the algorithm's optimization problem, which approximates the principal's true utility. Finally, we give the full algorithm in Section 3.5 and analyze its runtime in Section 3.6.

3.1 Preprocessing

The formulation of utility configuration given in Section 1.1 is for a discrete version of the problem. Recall that in this discrete formulation, action i has m possible utility configurations. Configuration j for action i is defined by a joint distribution F_{ij} over pairs (u_i^A, u_i^P) of principal utilities, given as a list of K_{ij} triples $(u_{ij}^{A,k}, u_{ij}^{P,k}, p_{ij}^k)$, where $u_{ij}^{A,k}$ (resp. $u_{ij}^{P,k}$) are the agent (resp. principal) utilities for this mass point, which is realized with probability p_{ij}^k. In this discrete formulation, probabilities p_{ij}^k are arbitrary numbers in the range $(0, 1]$.

The first step of the PTAS is to guess M agent utilities $-\infty = \hat{u}_0 \leq \ldots \leq \hat{u}_M$ such that under the optimal configuration, the agent's utility is roughly equally likely to fall in each of M bins $B_j = [\hat{u}_{j-1}, \hat{u}_j)$ for $j \in \{1, \ldots, M\}$ (with $B_M = [\hat{u}_{M-1}, \hat{u}_M]$). With general probability masses $p_{ij}^k \in (0, 1]$ for utilities, this may not be possible, as large probability masses cause the agent's utility to take particular values with probability significantly greater than the $1/M$ mass allowed in each bin. To enable us to partition the agent's utility distribution anyway, we now define a preprocessing procedure for breaking larger point masses into smaller masses of size at most $1/M^2$.

In the process, we will perturb agent utilities by a small amount. These perturbations will cause each of the smaller masses we construct to yield a distinct utility for the agent, allowing us to find the desired partition of the agent's utility space. Meanwhile, these perturbations will not change the expected principal utility from any configuration of the actions. The preprocessing algorithm is given in Algorithm 1, and explained informally below.

Informally, the preprocessing algorithm splits each point mass into smaller point masses with the same principal utility. Given a point mass $(u_{ij}^{A,k}, u_{ij}^{P,k}, p_{ij}^k)$ for configuration j of action i, we split it into $\lfloor p_{ij}^k/M^2 \rfloor$ point masses of size $1/M^2$, plus one more of size $p_{ij}^k - \lfloor p_{ij}^k/M^2 \rfloor$ to capture the remaining mass. The principal utilities for these new point masses will be $u_{ij}^{P,k}$, unchanged from the original. To induce the agent utilities of each point mass to be distinct, we perturb each new point mass by $\delta\ell$ for some extremely small $\delta > 0$, and with ℓ assuming a distinct value for each new point mass. By splitting point masses starting with those with the lowest principal utility, the perturbed utilities enforce tiebreaking in the principal's favor. We note that it is possible to select a δ with bit complexity polynomial in the original input size and such that the agent's choice between any two realizations of any two actions with distinct agent utilities is the same as it was before preprocessing; consequently, the principal's utility for any configuration of actions is the same before and after preprocessing.

Algorithm 1: Preprocessing Algorithm

```
Input  : Distributions $F_{ij}$ for each $i \in [n]$, $j \in [m]$, perturbation $\delta > 0$.
Output: New distributions $\tilde{F}_{ij}$ for each $i \in [n]$, $j \in [m]$.

/* Make full sorted list of point masses.                                  */
$L \leftarrow$ empty list;
for $i \in [n]$, $j \in [m]$, $k \in [K_{ij}]$ do
    Append $(u_{ij}^{A,k}, u_{ij}^{P,k}, p_{ij}^k)$ to $L$;
Sort $L$ in non-decreasing order by $u_{ij}^{P,k}$;

/* Empty list for each new distribution.                                   */
for $i \in [n]$, $j \in [m]$ do
    $\tilde{F}_{ij} \leftarrow$ empty list;

/* Split and perturb point masses.  Append to appropriate distribution.   */
$\ell \leftarrow 0$;
for $(u_{ij}^{A,k}, u_{ij}^{P,k}, p_{ij}^k)$ in $L$ do
    for $k' \in \{1, \ldots, \lfloor p_{ij}^k/M^2 \rfloor\}$ do
        Append $(u_{ij}^{A,k} + \delta\ell, u_{ij}^{P,k}, 1/M^2)$ to $\tilde{F}_{ij}$;       // Add whole increments of $1/M^2$.
        $\ell \leftarrow \ell + 1$;
    Append $(u_{ij}^{A,k} + \delta\ell, u_{ij}^{P,k}, p_{ij}^k - \lfloor p_{ij}^k/M^2 \rfloor)$ to $\tilde{F}_{ij}$;   // Add remainder of point mass.
    $\ell \leftarrow \ell + 1$;
Return $\tilde{F}_{ij}$ for each $i \in [n]$, $j \in [m]$;
```

For the remainder of the analysis, we assume that every realization of every action has a distinct agent utility and that each realization has probability at most $1/M^2$. One consequence of these facts is that the distribution of agent utilities in any solution, including the optimal solution, can now be split into bins of nearly equal probability, stated formally as follows.

Definition 7 (Utility bins). *Given optimal configuration* $\mathbf{C}^*$ *after preprocessing, let* $\hat{u}_0 = -\infty$, *and* $\hat{u}_j$ *be the smallest agent utility such that* $Pr[u^A_{OPT} \leq \hat{u}_j] \geq j/M$, *for all* $j \in \{1, \ldots, M\}$. *Given the* $\hat{u}_j$*s, we may define* utility bins $B_j = [\hat{u}_{j-1}, \hat{u}_j)$ *for* $j \in \{1, \ldots, M-1\}$ *and* $B_M = [\hat{u}_{M-1}, \hat{u}_M]$ *and* cumulative bins $B_{\leq j} = \cup_{k \leq j} B_k$.

After the preprocessing step, the bins constructed by Definition 7 divide the probability space of the agent's utility distribution nearly equally: Lemma 8 shows that the agent's utility falls into each of the M bins with probability approximately $1/M$.

Lemma 8. *The bins of Definition 7 satisfy:*

$$\Pr[u^A_{OPT} \in B_j] \in \left[\frac{1}{M} - \frac{1}{M^2}, \frac{1}{M} + \frac{1}{M^2}\right].$$

Proof. The cumulative bin probabilities satisfy $\Pr[u^A_{\text{OPT}} \in B_j] = \Pr[u^A_{\text{OPT}} \in B_{\leq j}] - \Pr[u^A_{\text{OPT}} \in B_{\leq j-1}]$. Since $\Pr[u^A_{\text{OPT}} \in B_{\leq j}] \in [j/M, j/M + 1/M^2]$ and $\Pr[u^A_{\text{OPT}} \in B_{\leq j-1}] \in [(j-1)/M, (j-1)/M + 1/M^2]$, we can apply these upper and lower bounds:

$$\Pr[u^A_{\text{OPT}} \in B_j] \geq j/M - ((j-1)/M + 1/M^2) = 1/M - 1/M^2$$

$$\Pr[u^A_{\text{OPT}} \in B_j] \leq j/M + 1/M^2 - (j-1)/M = 1/M + 1/M^2.$$

This implies the lemma. □

Many of the subsequent analyses of the optimal configuration rely on conditional probabilities of the form $\Pr[u^A_{\text{OPT}} \in B_k \mid u^A_{\text{OPT}} \in B_{\leq j}]$. The following lemma characterizes these probabilities. Since the bins nearly evenly divide the probability space, we should expect these probabilities to be roughly $1/j$. Especially important is the case where $k = j$, as $q^{\leq}_j = \Pr[u^A_{\text{OPT}} \in B_j \mid u^A_{\text{OPT}} \in B_{\leq j}]$ is a key quantity in the structure of the PTAS. The Lemma 9 below gives especially tight bounds for these probabilities.

Lemma 9. *For any* $k \leq j$, *the bins of Definition 7 satisfy*

$$\Pr[u^A_{OPT} \in B_k \mid u^A_{OPT} \in B_{\leq j}] \in \left[\frac{1 - 1/M}{j + 1/M}, \frac{1 + 1/M}{j}\right]. \tag{1}$$

Moreover, for $k = j$ *the following tighter bound holds:*

$$q^{\leq}_j = \Pr[u^A_{OPT} \in B_j \mid u^A_{OPT} \in B_{\leq j}] \in \left[\frac{1 - 1/M}{j}, \frac{1 + 1/M}{j + 1/M}\right]. \tag{2}$$

Proof. The definition of the bins implies

$$\Pr[u^A_{\text{OPT}} \in B_k \mid u^A_{\text{OPT}} \in B_{\leq j}] = \frac{\Pr[u^A_{\text{OPT}} \in B_{\leq k}] - \Pr[u^A_{\text{OPT}} \in B_{\leq k-1}]}{\Pr[u^A_{\text{OPT}} \in B_{\leq j}]}. \tag{3}$$

To prove (1), note that $\Pr[u^A_{\text{OPT}} \in B_{\leq k}] \in [k/M, k/M+1/M^2]$, $\Pr[u^A_{\text{OPT}} \in B_{\leq k-1}] \in [(k-1)/M, (k-1)/M + 1/M^2]$, and $\Pr[u^A_{\text{OPT}} \in B_{\leq j}] \in [j/M, j/M + 1/M^2]$. Plugging in the appropriate upper and lower bounds and simplifying yields the quantities in (1).

To prove (2), note that when $k = j$, the term $\Pr[u^A_{\text{OPT}} \in B_{\leq j}]$ appears in both the numerator and denominator of the right-hand side of (3). This function is increasing in $\Pr[u^A_{\text{OPT}} \in B_{\leq j}]$ and decreasing in $\Pr[u^A_{\text{OPT}} \in B_{\leq j-1}]$. Applying the bounds $\Pr[u^A_{\text{OPT}} \in B_{\leq j}] \in [j/M, j/M + 1/M^2]$ and $\Pr[u^A_{\text{OPT}} \in B_{\leq j-1}] \in [(j-1)/M, (j-1)/M + 1/M^2]$ and simplifying, we obtain the desired inequality. □

3.2 Probability Estimates

The high-level objective of the PTAS is to construct a configuration with approximately the same distribution for the agent's utility from their favorite action as the optimal solution. To maintain this approximation, the algorithm aims to match the probabilities that the agent's utility falls into the bins B_j defined as in Definition 9. More specifically, the algorithm tries to match the conditional probabilities $\Pr[u^A_{\text{ALG}} \in B_j \mid u^A_{\text{ALG}} \in B_{\leq j}]$ over the cumulative bins to the optimal conditional probabilities, $\Pr[u^A_{\text{OPT}} \in B_j \mid u^A_{\text{OPT}} \in B_{\leq j}]$. Rather than exactly track these conditional probabilities, the algorithm will maintain estimates, obtained in two steps, by rounding and then summing the action-wise probabilities $\Pr[u^A_i \in B_j \mid u^A_i \in B_{\leq j}]$. In more detail, we will analyze the relationship between the following quantities, which may be defined for any configuration $\mathbf{C}$:

- Conditional bin probabilities: $q^{\leq}_j = \Pr[u^A \in B_j \mid u^A \in B_{\leq j}]$.
- Action-wise conditional bin probabilities: $q^{\leq}_{ij} = \Pr[u^A_i \in B_j \mid u^A_i \in B_{\leq j}]$.
- Rounded action-wise probabilities: $\overline{q}^{\leq}_{ij} = \lfloor q^{\leq}_{ij} \cdot M^2 n \rfloor / M^2 n$.
- Additive estimates: $q^{\text{est}}_j = \sum_{i=1}^n q^{\leq}_{ij}$.
- Rounded additive estimates $\overline{q}^{\text{est}}_j = \sum_{i=1}^n \overline{q}^{\leq}_{ij}$.

We give three analyses of these quantities. First, we give a preliminary lemma comparing $q^{\leq}_j$ and q^{est}_j, which holds for any configuration. Next, we incorporate the rounding into analyses of the optimal configuration. We show that in the optimal configuration, $\overline{q}^{\text{est}}_j$ will be close to $1/j$, with the estimation error being large for small j corresponding to low-utility buckets, and small for high-utility buckets with large j. (Note that if the distributions were continuous, the true values of $\Pr[u^A_{\text{OPT}} \in B_j \mid u^A_{\text{OPT}} \in B_{\leq j}]$ would be $1/j$ for all j.) Finally, we show that any other solution for which the rounded additive estimates $\overline{q}^{\text{est}}_j$ fall within these same bounds, the conditional bin probabilities $q^{\leq}_j$ are also close to $1/j$. This will imply that tracking the rounded additive estimates $\overline{q}^{\text{est}}_j$ will achieve the desired goal of approximating the structure of the optimal configuration's distribution of agent utility.

The first task is a general analysis of the unrounded additive estimates. That is, for an arbitrary configuration $\mathbf{C}$, we compare $q^{\leq}_j = \Pr[u^A \in B_j \mid u^A \in B_{\leq j}]$ to the additive estimate $q^{\text{est}}_j = \sum_{i=1}^n \Pr[u^A_i \in B_j \mid u^A_i \in B_{\leq j}]$, where u^A is the agent's utility from their favorite action under $\mathbf{C}$. The following lemma gives the approximation:

Lemma 10. *For any configuration, the corresponding additive estimates satisfy:*

$$(1 - q_j^{\leq})q_j^{est} \leq q_j^{\leq} \leq q_j^{est}.$$

Proof. The rightmost inequality ($q_j^{\leq} \leq q_j^{\mathrm{est}}$) follows from the union bound. The leftmost inequality is proved by the following sequence of inequalities, explained after their statement:

$$\begin{aligned} q_j^{\leq} &= \Pr[u^A \in B_j \mid u^A \in B_{\leq j}] \\ &\geq \sum\nolimits_{i=1}^{n} \Pr[u_{i'}^A \notin B_j, \forall i' \neq i \mid u^A \in B_{\leq j}] \Pr[u_i^A \in B_j \mid u^A \in B_{\leq j}] && (4) \\ &\geq \sum\nolimits_{i=1}^{n} \Pr[u^A \notin B_j \mid u^A \in B_{\leq j}] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}] && (5) \\ &= (1 - q_j^{\leq}) \sum\nolimits_{i=1}^{n} \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}] \\ &= (1 - q_j^{\leq}) q_j^{\mathrm{est}}. \end{aligned}$$

Equation (4) follows from the fact that a sufficient condition for action i to be chosen is that all other actions fall in bins with lower utility, and from the independence of the actions. Equation (5) follows from the fact that the event $u^A \notin B_j$ is contained in the event $u_{i'}^A \notin B_j, \forall i' \neq i$. □

We now analyze the relationship between the rounded estimates $\overline{q}_j^{\leq}$ and the true probability $q_j^{\leq}$ for the optimal configuration $\mathbf{C}^*$. If the utility distributions were continuous, then $q_j^{\leq}$ would be $1/j$ for all j. The following lemma shows that even with discrete distributions and rounding, the additive estimates are close. In particular, we obtain low error when j is high, while for constant j, the error might be large.

Lemma 11. *The rounded additive estimates for the optimal configuration satisfy:*

$$\overline{q}_j^{est} \in \left[\frac{1 - 1/M}{j} - \frac{1}{M^2}, \frac{1 + 1/M}{j - 1} \right]. \tag{6}$$

Proof. We first lower bound $\overline{q}_j^{\mathrm{est}}$ with the following sequence of inequalities, explained after their statement:

$$\begin{aligned} \overline{q}_j^{\mathrm{est}} &\geq q_j^{\mathrm{est}} - n \cdot \frac{1}{M^2 n} && (7) \\ &= q_j^{\mathrm{est}} - \frac{1}{M^2} \\ &\geq q_j^{\leq} - \frac{1}{M^2} && (8) \\ &\geq \frac{1 - 1/M}{j} - \frac{1}{M^2}. && (9) \end{aligned}$$

Line (7) follows from the fact that there are n actions and for each action i, $q_{ij}^{\leq} - \overline{q}_{ij}^{\leq} \leq 1/(M^2 n)$, and line (8) follows from Lemma 10. Line (9) then follows from Lemma 9.

Similarly, we may upper bound $\overline{q}_j^{\mathrm{est}}$ as follows:

$$\overline{q}_j^{\mathrm{est}} \leq q_j^{\mathrm{est}} \tag{10}$$

$$
\begin{aligned}
&\le \frac{q_j^{\le}}{1-q_j^{\le}} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (11)\\
&\le \frac{1+1/M}{j-1}. \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (12)
\end{aligned}
$$

Line (10) follows because the rounded estimates are obtained by rounding down. Line (11) follows from Lemma 10. Finally, line (12) from noting that the expression in (11) is increasing in $q_j^{\le}$, plugging in the upper bound from Lemma 9, and simplifying. □

Lemma 11 implies that there exists at least one configuration, namely $\mathbf{C}^*$, which induces $\overline{q}_j^{\text{est}}$ satisfying equation (6). The PTAS will optimize an estimate of the principal's expected utility over configurations satisfying these bounds. We next show that any configuration satisfying these bounds also induces values of $q_j^{\le} = \Pr[u^A \in B_j \mid u^A \in B_{\le j}]$ which are close to the $1/j$.

Lemma 12. *For any configuration $\mathbf{C}$ satisfying equation (6) for all $j \in \{1, \ldots, M\}$, the resulting conditional probabilities satisfy:*

$$
q_j^{\le} \in \left[\frac{1-2/M}{j+1-2/M}, \frac{1+1/M}{j-1} + 1/M^2\right].
$$

Proof. We first lower bound $q_j^{\le}$. The following sequence of inequalities is explained after its statement:

$$
\begin{aligned}
q_j^{\le} &\ge \frac{q_j^{\text{est}}}{1+q_j^{\text{est}}} && (13)\\
&\ge \frac{\overline{q}_j^{\text{est}}}{1+\overline{q}_j^{\text{est}}} && (14)\\
&\ge \frac{\frac{1-1/M}{j} - 1/M^2}{1+\frac{1-1/M}{j} - 1/M^2} && (15)\\
&= \frac{1-1/M-j/M^2}{j+1-1/M-j/M^2}\\
&\ge \frac{1-2/M}{j+1-2/M}. && (16)
\end{aligned}
$$

Line (13) follows from rearranging the bound from Lemma 10. Line (14) holds because the right-hand side of line (13) is an increasing function of q_j^{est}, and so we can obtain a lower bound by noting that $\overline{q}_j^{\text{est}} \le q_j^{\text{est}}$. Line (15) then follows from plugging in the lower bound from equation (6). Finally, line (16) holds because $j \le M$.

Next, we can upper bound $q_j^{\le}$. Again, we explain the inequalities after their statement.

$$
\begin{aligned}
q_j^{\le} &\le q_j^{\text{est}} && (17)\\
&\le \overline{q}_j^{\text{est}} + 1/M^2 && (18)\\
&\le \frac{1+1/M}{j-1} + 1/M^2.
\end{aligned}
$$

Line (17) follows from Lemma 10. Line (18) follows from the fact that $q_j^{\text{est}} \le \overline{q}_j^{\text{est}} + 1/M^2$ due to the rounding increment. This implies the lemma. □

3.3 Utility Estimates

The previous section analyzed additive estimates of conditional bin probabilities, which will form the constraints of the dynamic program at the heart of the PTAS. The next step is to define and analyze additive estimates of the principal's utility. Specifically, given an arbitrary configuration $\mathbf{C}$, we consider the conditional contributions to the principal's utility defined as

$$u_j^{\le} = \mathbb{E}[u^P \mid u^A \in B_j] \Pr[u^A \in B_j \mid u^A \in B_{\le j}].$$

We will approximate this with the additive estimate

$$u_j^{\text{est}} = \sum_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\le j}].$$

To analyze the PTAS, we will combine these bin-wise estimates into a complete estimate of the principal's expected utility.

The main lemma of this section gives a general approximation relationship which depends only on the conditional bin probability $q_j^{\le} = \Pr[u^A \in B_j \mid u^A \in B_{\le j}]$, stated below.

Lemma 13. *For any configuration inducing conditional bin probabilities $q_j^{\le} = \Pr[u^A \in B_j \mid u^A \in B_{\le j}]$, the following holds:*

$$(1 - q_j^{\le}) u_j^{est} \le u_j^{\le} \le u_j^{est}.$$

Proof. Let i^* be a random variable denoting the agent's favorite action in $\mathbf{C}$. We use the $\mathbb{I}[A]$ to denote the indicator that event A occurs. Spelling out the dependence of the principal's utility on i^*, the upper bound follows from the reasoning below:

$$\begin{align}
u_j^{\le} &= \mathbb{E}[u_{i^*}^P \mid u_{i^*}^A \in B_j] \Pr[u_{i^*}^A \in B_j \mid u_{i^*}^A \in B_{\le j}] \tag{19}\\
&= \mathbb{E}[u_{i^*}^P \mathbb{I}[u_{i^*}^A \in B_j] \mid u_{i^*}^A \in B_{\le j}] \tag{20}\\
&\le \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mathbb{I}[u_i^A \in B_j] \mid u_{i^*}^A \in B_{\le j}] \tag{21}\\
&\le \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mathbb{I}[u_i^A \in B_j] \mid u_1^A \in B_{\le j}, \dots, u_n^A \in B_{\le j}] \tag{22}\\
&= \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mathbb{I}[u_i^A \in B_j] \mid u_i^A \in B_{\le j}] \tag{23}\\
&= \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\le j}] \tag{24}\\
&= u_j^{\text{est}}.
\end{align}$$

Line (19) follows from the definition of $u_j^{\le}$. Lines (20) and (24) follow from basic probability. Line (21) follows from the fact that the previous line counts only contributions from i^*, instead of those from all actions. Line (22) holds by the definition of i^*, and line (23) from the independence of the actions. This proves the upper bound.

Meanwhile, the lower bound follows from the following analysis:

$$\begin{align}
u_j^{\le} &= \mathbb{E}[u_{i^*}^P \mid u_{i^*}^A \in B_j] \Pr[u_{i^*}^A \in B_j \mid u_{i^*}^A \in B_{\le j}] \notag\\
&\ge \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_{i'}^A \notin B_j, \forall i' \neq i \mid u_{i^*}^A \in B_{\le j}] \Pr[u_i^A \in B_j \mid u_{i^*}^A \in B_{\le j}] \tag{25}\\
&\ge \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_{i^*}^A \notin B_j \mid u_{i^*}^A \in B_{\le j}] \Pr[u_i^A \in B_j \mid u_{i^*}^A \in B_{\le j}] \tag{26}
\end{align}$$

$$= \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_{i^*}^A \notin B_j \mid u_{i^*}^A \in B_{\leq j}] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}] \tag{27}$$

$$= (1 - q_j^{\leq}) \sum\nolimits_{i=1}^{n} \mathbb{E}[u_i^P \mid u_i^A \in B_j] \Pr[u_i^A \in B_j \mid u_i^A \in B_{\leq j}] \tag{28}$$

$$= (1 - q_j^{\leq}) u_j^{\text{est}}.$$

Line (25) follows from the fact that a sufficient condition for action i to be selected is that $u_i^A \in B_j$ while all other actions i' have $u_{i'}^A \in B_{\leq j-1}$, along with independence across actions. Line (26) comes from the fact that $\Pr[u_{i'}^A \notin B_j, \forall i' \neq i \mid u_{i^*}^A \in B_{\leq j}] \geq \Pr[u_{i^*}^A \notin B_j \mid u_{i^*}^A \in B_{\leq j}]$ by the definition of i^*. Line (27) follows from independence across actions. Finally, line (28) follows from the definition of $q_j^{\leq}$. This proves the lower bound and thus the lemma. □

3.4 Approximate Objective Function

We now define the objective function for the dynamic program at the heart of the PTAS and analyze its relationship to the true objective. To define the objective function, we consider the following weighted sum of the utility estimates from Section 3.3:

$$\overline{\text{Util}}(\mathbf{C}) = \sum_{j=6}^{M} c_j u_j^{\text{est}},$$

where u_j^{est} implicitly depends on $\mathbf{C}$ for all j, and where $c_j = \frac{j-5}{M-1}$. We give two analyses of $\overline{\text{Util}}$. First, we show that for any configuration $\mathbf{C}$ where $q_j^{\leq}$ satisfies the bounds of Lemma 12 for all $j \in \{1, \ldots, M\}$, $\overline{\text{Util}}(\mathbf{C})$ is a lower bound on the principal's true utility under that same configuration. This will serve to lower-bound the objective value of the configurations computed by the PTAS. We present this analysis in Section 3.4.1. Second, we show that for the *optimal configuration* $\mathbf{C}^*$ *in particular*, the approximate objective $\overline{\text{Util}}(\mathbf{C}^*)$ is also an approximate upper bound. This step relies critically on the utility alignment property (and is the only place in the analysis where the property comes into play). We give this analysis in Section 3.4.2. After proving these two lemmas, the stage will be set for the rigorous description of the full PTAS, as well as the final pieces of its analysis, presented in Section 3.5.

3.4.1 Lower Bounding PTAS Solutions

This section analyzes configurations such that for all j, $q_j^{\leq}$ satisfies the bounds of Lemma 12, restated below

$$q_j^{\leq} \in \left[\frac{1 - 2/M}{j + 1 - 2/M}, \frac{1 + 1/M}{j - 1} + 1/M^2\right]. \tag{29}$$

The result is the following:

Lemma 14. *Let* $\mathbf{C}$ *be a configuration such that* $q_j^{\leq}$ *satisfies the bounds of Lemma 12. Then the principal's expected utility from* $\mathbf{C}$ *is at least* $\overline{\mathit{Util}}(\mathbf{C})$.

Proof. We lower bound the principal's expected utility by the following sequence of relations, justified after their statement:

$$\mathbb{E}[u^P] = \sum\nolimits_{j=1}^{M} \mathbb{E}[u^P \mid u^A \in B_j] \Pr[u^A \in B_j] \tag{30}$$

$$
\begin{aligned}
&\geq \sum\nolimits_{j=6}^{M} \mathbb{E}[u^P \mid u^A \in B_j] \Pr[u^A \in B_j] && (31)\\
&= \sum\nolimits_{j=6}^{M} \mathbb{E}[u^P \mid u^A \in B_j] \Pr[u^A \in B_j \mid u^A \in B_{\leq j}] \frac{\Pr[u^A \in B_j]}{\Pr[u^A \in B_j \mid u^A \in B_{\leq j}]}\\
&= \sum\nolimits_{j=6}^{M} u_j^{\leq} \frac{\Pr[u^A \in B_j]}{q_j^{\leq}} && (32)\\
&\geq \sum\nolimits_{j=6}^{M} (1-q_j^{\leq}) u_j^{\text{est}} \frac{\Pr[u^A \in B_j]}{q_j^{\leq}} && (33)\\
&\geq \sum\nolimits_{j=6}^{M} (1-q_j^{\leq}) u_j^{\text{est}} \frac{q_j^{\leq} \prod_{k>j}(1-q_k^{\leq})}{q_j^{\leq}} && (34)\\
&= \sum\nolimits_{j=6}^{M} \prod_{k \geq j} (1-q_k^{\leq}) u_j^{\text{est}}\\
&\geq \sum\nolimits_{j=6}^{M} c_j u_j^{\text{est}} && (35)\\
&= \overline{\text{Util}}(\mathbf{C}).
\end{aligned}
$$

Line (30) follows from the law of total expectation, and line (31) from the fact that the principal's utilities are nonnegative. Line (32) follows from the definitions of $q_j^{\leq}$ and $u_j^{\leq}$. Line (33) holds because of Lemma 13. Line (34) follows by Lemma 15, proved below, which states that $\Pr[u^A \in B_j] = q_j^{\leq} \prod_{k>j}(1-q_k^{\leq})$. Finally, Line (35) holds because of Lemma 16, also proved below, which states that when $q_j^{\leq}$ satisfies (29) for all j, $\prod_{k \geq j}(1-q_k^{\leq}) \geq \frac{j-5}{M-1} = c_j$ for $j \geq 6$. □

We conclude by stating and proving the technical lemmas required for Lemma 14.

Lemma 15. *For all $j \in \{1, \ldots, M\}$, the bin probabilities satisfy:*

$$\Pr[u^A \in B_j] = q_j^{\leq} \prod_{k>j}(1-q_k^{\leq}) = \Pr[u^A \in B_j \mid u^A \in B_{\leq j}] \prod_{k>j} \Pr[u^A \in B_{\leq k-1} \mid u^A \in B_{\leq k}].$$

Proof. The right-hand expression follows from the middle one by the definition of the $q_j^{\leq}$s. To see that the right-hand expression is equal to $\Pr[u^A \in B_j]$, we use the law of iterated expectation. Consider the following sequential process for realizing u^A: first determine whether $u^A \in B_M$, which occurs with probability equal to $\Pr[u^A \in B_M] = \Pr[u^A \in B_M \mid u^A \in B{\leq}\, M] = q_M^{\leq}$. If this does not occur, then determine whether $u^A \in B_{M-1}$, which, conditioned on reaching this point, occurs with probability $\Pr[u^A \in B_{M-1} \mid u^A \in B_{\leq M-1}] = q_{M-1}^{\leq}$, and so on for each box from $M-2$ down to j. For $u^A \in B_j$, then, u^A must have failed to lie in B_k for all $k > j$, which happens with overall probability $\prod_{k>j}(1-q_k^{\leq})$. The probability of finally falling in B_j given these previous failures is then $\Pr[u^A \in B_j \mid u^A \in B_{\leq j}] = q_j^{\leq}$, for an overall probability of $q_j^{\leq} \prod_{k>j}(1-q_k^{\leq})$. □

Lemma 16. *For any configuration inducing $q_j^{\leq}$s satisfying (29) for all $j \in \{1, \ldots, M\}$, it holds that $\prod_{k \geq j}(1-q_k^{\leq}) \geq \frac{j-5}{M-1}$ for $j \geq 6$.*

Proof. At a high level, the result will follow from the fact that $q_k^{\leq} \approx 1 - 1/k = (k-1)/k$ for all $k \geq j$. Were this to hold exactly, then the product would telescope, yielding $(j-1)/M$. The crux of the analysis, then, is showing that multiplying the errors introduced in the $q_k^{\le}$s does not cause significant loss. Formally, the lemma follows from the inequalities below, explained after their statement.

$$
\begin{align*}
\prod_{k\ge j}(1-q_k^{\le}) &\ge \prod_{k\ge j}\left(1-\frac{1+1/M}{k-1}-\frac{1}{M^2}\right) \tag{36}\\
&= \prod_{k\ge j}\left(\frac{k-2-1/M}{k-1}-\frac{1}{M^2}\right)\\
&\ge \prod_{k\ge j}\left(\frac{k-2-1/M}{k-1}-\frac{1/M}{k-1}\right) \tag{37}\\
&= \prod_{k\ge j}\frac{k-2-2/M}{k-1}\\
&= \frac{j-2-2/M}{j-1}\cdot\frac{j-1-2/M}{j}\cdot\ldots\cdot\frac{M-3-2/M}{M-2}\cdot\frac{M-2-2/M}{M-1}\\
&= \frac{j-2-2/M}{M-1}\prod_{k=j-1}^{M-2}\left(\frac{k-2/M}{k}\right) \tag{38}\\
&= \frac{j-2-2/M}{M-1}\prod_{k=j-1}^{M-2}\left(1-\frac{2}{Mk}\right)\\
&\ge \frac{j-2-2/M}{M-1}\left(1-\sum_{k=j-1}^{M-2}\frac{2}{Mk}\right)\\
&= \frac{j-2-2/M}{M-1}\left(1-\frac{2}{M}(H_{M-2}-H_{j-2})\right)\\
&\ge \frac{j-2-2/M}{M-1}\left(1-\frac{2}{M}(\log(M-2)+1-\log(j-2))\right) \tag{39}\\
&= \frac{j-2-2/M}{M-1}-\frac{j-2-2/M}{M-1}\cdot\frac{2}{M}\left(\log\left(\frac{M-2}{j-2}\right)+1\right)\\
&\ge \frac{j-2-2/M}{M-1}-\frac{2}{M-1}\cdot\frac{j-2}{M-2}\left(\log\left(\frac{M-2}{j-2}\right)+1\right) \tag{40}\\
&= \frac{j-2-2/M}{M-1}-\frac{2}{M-1}\left(\frac{j-2}{M-2}\log\left(\frac{M-2}{j-2}\right)+\frac{j-2}{M-2}\right)\\
&\ge \frac{j-2-2/M}{M-1}-\frac{2}{M-1} \tag{41}\\
&\ge \frac{j-5}{M-1}. \tag{42}
\end{align*}
$$

Line (36) follows from the lower bound (29). Line (37) follows from the fact that $k-1\le M$. Line (38) follows from rearranging the numerators and denominators of the expanded product on the previous line. Line (39) holds because $H_k\in[\log n,\log n+1]$. Line (40) follows from rearranging the fractions in the second term, and noting that $j-2-2/M\le j-2$ on top while $M-2\le M$ on bottom. Finally, line (41) follows because $x\log x+x\le 1$ for all $x\in[0,1]$, and line (42) because

$M \geq 2$. All other lines follow from algebra or well-known inequalities. □

3.4.2 Upper Bounding Optimal Solutions

We now analyze the relationship between the true principal utility under the optimal configuration $\mathbf{C}^*$ and the approximate objective value $\overline{\mathrm{Util}}(\mathbf{C}^*)$. The approximate objective disproportionately down-weights the utility contribution from lower bins but the error from down-weighting disappears as the index j of the bucket gets closer to M, corresponding to stronger quantiles of the agent's utility distribution. Utility alignment implies that most of the optimal utility comes from these stronger quantiles. Hence, the approximate objective $\overline{\mathrm{Util}}(\mathbf{C}^*)$ is not much less than the principal's true utility in $\mathbf{C}^*$. The upper bound on the optimal utility will depend on the degree of utility alignment in the utility configuration instance. The following notation will allow us to state the bound.

Definition 17. *Given an instance of utility configuration satisfying $f(q)$-utility alignment, the* utility alignment coefficients *are defined as $r_j = f(j/M)$ for $j \in \{1, \dots, M\}$.*

Utility alignment implies that for the optimal configuration and any $j \in \{1, \dots, M\}$, the conditional expectation $\mathbb{E}[u^P_{\mathrm{OPT}} \mid u^A_{\mathrm{OPT}} \in B_{\leq j}]$ is at most $r_j \mathbb{E}[u^P_{\mathrm{OPT}}]$. We then have the following bound on the optimal utility.

Lemma 18. *Let $\mathbf{C}^*$ be the optimal configuration. Then the principal's optimal utility $\mathbb{E}[u^P_{OPT}]$ is approximately upper-bounded by the objective $\overline{Util}(\mathbf{C}^*)$, namely:*

$$\overline{Util}(\mathbf{C}^*) \geq \frac{M-1}{M+1} \left(\frac{M-5}{M-1} - \frac{5}{6} \frac{r_5}{M-1} - \frac{5}{M-1} \sum_{j=6}^{M} \frac{r_j}{j-1} \right) \mathbb{E}[u^P_{OPT}]. \tag{43}$$

To parse the complicated bound of (43), note that if the r_js are constant (e.g. $r_j = 2$ for all j), we may conclude that $\overline{\mathrm{Util}}(\mathbf{C}^*) \geq (1 - O(\log M/M))\mathbb{E}[u^P_{\mathrm{OPT}}]$. Similarly non-constant r_js that are smaller than M/j (e.g. $r_j = \sqrt{M/j}$) will yield weaker approximations that still approach 1 as M grows. Before proving the lemma, note that all the expressions considered so far for the utility of a configuration have been in terms of the utility conditioned on the agent's utility falling in a particular bin, $\mathbb{E}[u^P \mid u^A \in B_j]$. Utility configuration is stated in terms of the event that the agent's utility falls into cumulative bins, $\mathbb{E}[u^P \mid u^A \in B_{\leq j}]$. The following lemma will bridge the gap between these two accounting strategies and allow us to use summation by parts in the proof of Lemma 18.

Lemma 19. *In the optimal configuration $\mathbf{C}^*$, for any bin j, the principal's expected utility satisfies:*

$$\mathbb{E}[u^P_{OPT} \mid u^A_{OPT} \in B_j] \geq \frac{j + 1/M}{1 - 1/M} \mathbb{E}[u^P_{OPT} \mid u^A_{OPT} \in B_{\leq j}] - \frac{j-1}{1 + 1/M} \mathbb{E}[u^P_{OPT} \mid u^A_{OPT} \in B_{\leq j-1}].$$

Proof. We start by rewriting each of the terms on the right-hand side, using the bounds from Lemma 9, which show that $\Pr[u^A_{\mathrm{OPT}} \in B_k | u^A_{\mathrm{OPT}} \in B_{\leq j}] \in [\frac{1-1/M}{j+1/M}, \frac{1+1/M}{j}]$ for any $k \leq j$.

$$\mathbb{E}[u^P_{\mathrm{OPT}} \mid u^A_{\mathrm{OPT}} \in B_{\leq j}] = \sum_{k=1}^{j} \mathbb{E}[u^P_{\mathrm{OPT}} \mid u^A_{\mathrm{OPT}} \in B_k] \cdot \Pr[u^A_{\mathrm{OPT}} \in B_k | u^A_{\mathrm{OPT}} \in B_{\leq j}]$$

$$\geq \frac{1-1/M}{j+1/M} \sum_{k=1}^{j} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_k].$$

$$\begin{aligned}
\mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j-1}] &= \sum_{k=1}^{j-1} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_k] \cdot \Pr[u^A_{\text{OPT}} \in B_k | u^A_{\text{OPT}} \in B_{\leq j-1}] \\
&\leq \frac{1+1/M}{j-1} \sum_{k=1}^{j-1} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_k].
\end{aligned}$$

Combining these two inequalities completes the proof. □

Proof of Lemma 18. The proof consists of three main steps. First, we note that the objective is defined in terms of the utility estimates u^{est}_j, which are lower bounded by the true conditional utilities $u^{\leq}_j = \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_j] \Pr[u^A_{\text{OPT}} \in B_j \mid u^A_{\text{OPT}} \in B_{\leq j}]$. Next, we use Lemma 19 to rewrite the conditional utilities in terms of differences of the cumulative utilities $\mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j}]$. Finally, we use summation by parts to rewrite the quantity in a form where we may apply utility alignment, yielding a bound in terms of the true optimal utility $E[u^P_{\text{OPT}}]$. Formally, we have the following, explained after the statement:

$$\begin{aligned}
\overline{\text{Util}}(\mathbf{C}^*) &= \sum_{j=6}^{M} c_j u^{\text{est}}_j \\
&\geq \sum_{j=6}^{M} c_j u^{\leq}_j && (44) \\
&= \sum_{j=6}^{M} c_j \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_j] \Pr[u^A_{\text{OPT}} \in B_j \mid u^A_{\text{OPT}} \in B_{\leq j}] && (45) \\
&\geq \left(1 - \frac{1}{M}\right) \sum_{j=6}^{M} \frac{c_j}{j} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_j] && (46) \\
&\geq \left(1 - \frac{1}{M}\right) \sum_{j=6}^{M} \frac{c_j}{j} \Big(\frac{j+1/M}{1-1/M} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j}] \\
&\qquad - \frac{j-1}{1+1/M} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j-1}] \Big) && (47) \\
&\geq \left(1 - \frac{1}{M}\right) \sum_{j=6}^{M} \frac{c_j}{j} \Big(\frac{j}{1+1/M} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j}] \\
&\qquad - \frac{j-1}{1+1/M} \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j-1}] \Big) \\
&= \frac{M-1}{M+1} \sum_{j=6}^{M} \frac{c_j}{j} \Big(j \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j}] - (j-1) \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq j-1}] \Big) \\
&= \frac{M-1}{M+1} \Bigg(c_M \mathbb{E}[u^P_{\text{OPT}}] - \frac{c_6}{6} \cdot 5 \mathbb{E}[u^P_{\text{OPT}} \mid u^A_{\text{OPT}} \in B_{\leq 5}]
\end{aligned}$$

$$
\begin{aligned}
&\qquad - \sum_{j=6}^{M} \left(\frac{c_j}{j} - \frac{c_{j-1}}{j-1} \right) j \cdot \mathbb{E}[u^P_{\mathrm{OPT}} \mid u^A_{\mathrm{OPT}} \in B_{\le j}] \Bigg) \qquad (48)\\
&\ge \frac{M-1}{M+1} \left(c_M \mathbb{E}[u^P_{\mathrm{OPT}}] - \frac{5}{6} c_6 r_5 \mathbb{E}[u^P_{\mathrm{OPT}}] - \sum_{j=6}^{M} \left(\frac{c_j}{j} - \frac{c_{j-1}}{j-1} \right) r_j \cdot j \cdot \mathbb{E}[u^P_{\mathrm{OPT}}] \right) \qquad (49)\\
&= \frac{M-1}{M+1} \left(c_M - \frac{5}{6} c_6 r_5 - \sum_{j=6}^{M} \left(\frac{c_j}{j} - \frac{c_{j-1}}{j-1} \right) r_j \cdot j \right) \mathbb{E}[u^P_{\mathrm{OPT}}]\\
&= \frac{M-1}{M+1} \left(\frac{M-5}{M-1} - \frac{5}{6} \frac{r_5}{M-1} - \sum_{j=6}^{M} \left(\frac{j-5}{j(M-1)} - \frac{j-6}{(j-1)(M-1)} \right) r_j \cdot j \right) \mathbb{E}[u^P_{\mathrm{OPT}}]\\
&= \frac{M-1}{M+1} \left(\frac{M-5}{M-1} - \frac{5}{6} \frac{r_5}{M-1} - \frac{1}{M-1} \sum_{j=6}^{M} \left(\frac{j-5}{j} - \frac{j-6}{j-1} \right) r_j \cdot j \right) \mathbb{E}[u^P_{\mathrm{OPT}}]\\
&= \frac{M-1}{M+1} \left(\frac{M-5}{M-1} - \frac{5}{6} \frac{r_5}{M-1} - \frac{5}{M-1} \sum_{j=6}^{M} \frac{r_j}{j-1} \right) \mathbb{E}[u^P_{\mathrm{OPT}}].
\end{aligned}
$$

Line (44) follows from Lemma 13, and line (45) from the definition of the $u_j^{\le}$s. Line (46) follows from Lemma 9, which states that $\Pr[u^A_{\mathrm{OPT}} \in B_j \mid u^A_{\mathrm{OPT}} \in B_{\le j}] \ge (1 - 1/M)/j$. Line (47) then follows from Lemma 19. In line (48), we apply summation by parts, and in line (49), we apply utility alignment. The remaining lines are algebra. □

3.5 PTAS Description and Approximation Analysis

We now present and analyze the full PTAS for utility configuration. Recall that an input to the utility configuration problem consists of n actions, each of which has m possible configurations, each represented by a discrete distribution over (u_i^A, u_i^P) pairs. Before executing the algorithm's main loop, we preprocess the discrete distributions to obtain an equivalent instance where each probability mass in each utility distribution has size at most $1/M^2$, and such that agent utilities are distinct across all point masses for all configurations for all actions. Full details and analysis of the preprocessing algorithm can be found in Section 3.1.

For a constant M (which governs the approximation factor), the main loop of the algorithm iterates through all profiles of bin boundaries $\hat{u}_1 \le \cdots \le \hat{u}_M$, chosen from the union of all supports of all agent utility distributions for all actions and configurations. Note that the number of such profiles is exponential in M, but for constant M, it is polynomial in the input size. These bin boundaries then define bins $B_1, \ldots, B_M$ in agent utility space, which then allow us to define the key quantities analyzed in Sections 3.2, 3.3, and 3.4. These quantities then define the optimization we perform for each profile of bins.

For each choice of $\hat{u}_1 \le \cdots \le \hat{u}_M$, we solve a dynamic program, with the objective of maximizing the approximate objective $\overline{\mathrm{Util}}$ defined in Section 3.4, subject to constraint that the agent utilities induced by the chosen configuration should be approximately evenly distributed between the buckets $B_1, \ldots, B_M$, which we enforce using the (rounded) additive estimates $\overline{q}_j^{\mathrm{est}}$ of $\Pr[u^A \in B_j \mid u^A \in B_{\le j}]$, defined in Section 3.2. Formally, we solve the following problem for each choice of $\hat{u}_1 \leq \cdots \leq \hat{u}_M$:

$$\begin{aligned} &\text{maximize} && \overline{\mathrm{Util}}(\mathbf{C}) \\ &\text{subject to} && \overline{q}_j^{\mathrm{est}} \in \left[\frac{1-1/M}{j} - \frac{1}{M^2}, \frac{1+1/M}{j-1}\right] && \forall j \in \{1, \ldots, M\}, \end{aligned}$$

where recall that each $\overline{q}_j^{\mathrm{est}}$ depends on the choice of configuration. We discuss the details of the dynamic programming formulation of this problem below. While the program above may not be feasible for all choices of $\hat{u}_1 \leq \cdots \leq \hat{u}_M$, the analysis of the previous sections implies that it will be feasible for at least one choice, corresponding to a correct guess of the bin boundaries in the true optimal configuration. The overall algorithm computes the expected principal utility from each configuration obtained by solving the program above for each choice of $\hat{u}_1 \leq \cdots \leq \hat{u}_M$, and returns the best one overall.[1]

We next show that for constant M, the optimization problem above can be solved in polynomial time via dynamic programming. Given a configuration $\mathbf{C}$, the objective $\overline{\mathrm{Util}}(\mathbf{C})$ is defined as follows, making the dependence on the chosen configuration explicit:

$$\begin{aligned} \sum\nolimits_{j=6}^{M} c_j u_j^{\mathrm{est}} &= \sum\nolimits_{j=6}^{M} c_j \sum\nolimits_{i=1}^{n} \mathbb{E}_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^P \mid u_i^A \in B_j] \Pr_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^A \in B_j \mid u_i^A \in B_{\leq j}] \\ &= \sum\nolimits_{i=1}^{n} \sum\nolimits_{j=6}^{M} c_j \mathbb{E}_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^P \mid u_i^A \in B_j] \Pr_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^A \in B_j \mid u_i^A \in B_{\leq j}]. \end{aligned}$$

Hence, if we define

$$\psi_i(C_i) = \sum\nolimits_{j=6}^{M} c_j \mathbb{E}_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^P \mid u_i^A \in B_j] \Pr_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^A \in B_j \mid u_i^A \in B_{\leq j}]$$

to be the contribution of action i to $\overline{\mathrm{Util}}(\mathbf{C})$, then $\overline{\mathrm{Util}}(\mathbf{C}) = \sum_i \psi_i(C_i)$.

To define the subproblem and recurrence for the dynamic program, recall the definitions of the probability estimates $\overline{q}_j^{\mathrm{est}}$, where we now make dependence on the configuration $\mathbf{C}$ explicit: we have $q_{ij}^{\leq}(C_i) = \Pr_{(u_i^A, u_i^P) \sim F_{i,C_i}}[u_i^A \in B_j \mid u_i^A \in B_{\leq j}]$, $\overline{q}_{ij}^{\leq}(C_i) = \lfloor q_{ij}^{\leq}(C_i)/(M^2 n) \rfloor$, and $\overline{q}_j^{\mathrm{est}}(\mathbf{C}) = \sum_{i=1}^{n} \overline{q}_{ij}^{\leq}(C_i)$. From these definitions, each action contributes to $\overline{q}_j^{\mathrm{est}}$ in increments of $1/M^2 n$. Thus, given $i \in \{1, \ldots, n\}$ and $\mathbf{k} = (k_1, \ldots, k_M) \in [M^2 n]^M$, define $\mathrm{OPT}(i, \mathbf{k})$ to be the maximum value of $\sum_{i'=i}^{M} \psi_{i'}(C_{i'})$ over all configurations of actions $i, \ldots, M$ such that for each $j \in \{1, \ldots, M\}$, it holds that

$$\frac{k_j}{M^2 n} + \sum_{i'=i}^{M} \overline{q}_{ij}^{\leq}(C_{i'}) \in \left[\frac{1-1/M}{j} - \frac{1}{M^2}, \frac{1+1/M}{j-1}\right].$$

Then $\mathrm{OPT}(1, \mathbf{0})$ corresponds to the optimal solution to the optimization problem above. The base cases are $\mathrm{OPT}(n+1, \mathbf{k})$ for all $\mathbf{k} \in [M^2 n]^M$, with value either 0 or $-\infty$ depending on whether $\mathbf{k}/M^2 n$ satisfies the program's constraints on $\overline{q}_j^{\mathrm{est}}$. To define the recurrence, consider configuration ℓ of action i, and let $k_j^{i,\ell} = \overline{q}_{ij}^{\leq}(\ell) \cdot M^2 n$ be the number of increments action i would contribute to bin j in configuration ℓ. Then we have the following recurrence:

$$\mathrm{OPT}(i, \mathbf{k}) = \max_{\ell \in \{1, \ldots, m\}} \psi_i(\ell) + \mathrm{OPT}(i+1, \mathbf{k} - (k_1^{i,\ell}, \ldots, k_M^{i,\ell})).$$

[1] Given a configuration, computing the principal's true expected utility can be done by emulating the approach of Khodabakhsh et al. [36]: let $a^*(i)$ denote the agent's favorite action among $1, \ldots, i$. Let $\mathrm{Util}_i(u) = \mathbb{E}[u_{a^*(i)}^P \mid u_{a^*(i)}^A = u]$ and $p_i(u) = \Pr[u_{a^*(i)}^A = u]$. Both can be computed by dynamic programming, and the expected principal utility for any configuration can be computed using knowledge of the $\mathrm{Util}_M(u)$s, $p_M(u)$s, and the law of total expectation.

We analyze the runtime more carefully in Section 3.6, but for now note that each k_j can range from 1 to $M^2 n$ and there are at most m choices at each stage of the optimization, which for any constant M will yield a polynomial-time algorithm in n, m, and the bit complexity of the utilities and probabilities.

We can now analyze the approximation ratio of the algorithm described above. The approximation ratio will depend on the choice of M and the utility alignment of the utility configuration instance, which we summarize by the coefficients $r_1, \ldots, r_M$ (Definition 17).

Theorem 20. *For an instance of utility configuration with alignment coefficients $r_1, \ldots, r_M$, the algorithm of this section achieves an α-approximation to the optimal principal utility, where*

$$\alpha = \frac{M-1}{M+1}\left(\frac{M-5}{M-1} - \frac{5}{6}\cdot\frac{r_5}{M-1} - \frac{5}{M-1}\sum_{j=6}^{M}\frac{r_j}{j-1}\right).$$

Proof. Consider the choice of $\hat{u}_1, \ldots, \hat{u}_M$ corresponding to the optimal configuration $\mathbf{C}^*$ as in Definition 7, and define the bins $B_1, \ldots, B_M$ and $B_{\leq 1}, \ldots, B_{\leq M}$ accordingly. These, in turn, set the approximate objective function $\overline{\text{Util}}$, which depends on the bins to be well-defined. Let $\mathbf{C}$ denote the configuration computed by the algorithm for this choice of $\hat{u}_1, \ldots, \hat{u}_M$. By Lemma 11, the optimal configuration is feasible for this choice of $\hat{u}_1, \ldots, \hat{u}_M$, and hence the algorithm does indeed find some configuration on this iteration. Let $\mathbb{E}[u^P_{\text{ALG}}]$ denote the expected principal utility for $\mathbf{C}$. Then we have the following:

$$\begin{aligned} \mathbb{E}[u^P_{\text{ALG}}] &\geq \overline{\text{Util}}(\mathbf{C}) && (50)\\ &\geq \overline{\text{Util}}(\mathbf{C}^*) && (51)\\ &\geq \alpha\mathbb{E}[u^A_{\text{OPT}}]. && (52) \end{aligned}$$

Line (50) follows from the fact that the feasibility constraints of the dynamic program imply that $\mathbf{C}$ satisfies the bounds of Lemma 11 and hence the true $q_j^{\leq}$s satisfy the bounds of Lemma 12. Lemma 14 implies that $\overline{\text{Util}}$ is a lower bound on the expected principal utility for such configurations. Line (51) follows from Lemma 11, which implies that the optimal configuration is feasible for the dynamic program, and therefore can have no higher objective value than $\mathbf{C}$. Finally, (52) follows from Lemma 18. Since the algorithm of this section returns a configuration that has at least as high expected principal utility as $\mathbf{C}$ does, the result follows. □

We conclude the section by applying Theorem 20 with the two sets of alignment coefficients that will apply for the applications in this paper, and work out the resulting approximation ratios as a function of M. In each case, we obtain an approximation ratio which goes to 1 as M grows, implying (given the runtime analysis of the next section) that the algorithm is a PTAS.

Corollary 21. *For utility configuration instances satisfying 2-utility alignment, the algorithm of this section is a $(1 - O(\log M/M))$-approximation.*

Proof. For 2-utility aligned instances, we have $r_j = 2$ for all j. Plugging these r_j into the formula for the approximation ratio, we obtain:

$$\frac{M-1}{M+1}\left(\frac{M-5}{M-1} - \frac{5}{6}\cdot\frac{2}{M-1} - \frac{5}{M-1}\sum_{j=6}^{M}\frac{2}{j-1}\right).$$

The final term is the dominant source of loss, and is $O(\log M/M)$, implying the result. □

Corollary 22. *For utility configuration instances satisfying $1/(c\sqrt{q})$-utility alignment for constant c the algorithm of this section is a $(1 - O(1/\sqrt{M}))$-approximation.*

Proof. For such instances, we have $r_j = c\sqrt{M/j}$ for all j. Therefore, the approximation ratio is:

$$\frac{M-1}{M+1}\left(\frac{M-5}{M-1} - \frac{5}{6}\cdot\frac{c\sqrt{M}}{5(M-1)} - \frac{5}{M-1}\sum_{j=6}^{M}\frac{c\sqrt{M}}{(j-1)\sqrt{j}}\right).$$

$$= \frac{M-1}{M+1}\left(\frac{M-5}{M-1} - \frac{5}{6}\cdot\frac{c\sqrt{M}}{5(M-1)} - \frac{5c\sqrt{M}}{M-1}\sum_{j=6}^{M}\frac{1}{(j-1)\sqrt{j}}\right). \tag{53}$$

The sum in (53) is now $O(1)$. The second and third terms in parentheses are now the dominant loss terms, and are both $O(1/\sqrt{M})$. □

3.6 Runtime Analysis

We conclude the section with a runtime analysis of the algorithm and show that for fixed M, the runtime is polynomial in the number of actions n, the number of configurations per action m, the number of masses K per distribution, and the bit complexity of the utilities and probabilities given as inputs to the problem.

First, consider preprocessing (Section 3.1). The preprocessing step takes each probability mass, splits it into at most M^2 smaller probability masses, and adds a perturbation to each agent's utility in such a way that the perturbations have polynomial bit complexity in the size of the original input. This blows up the number of mass points by at most a factor of M^2, while keeping the number of actions and configurations per action the same. Hence, the size of the output to preprocessing is polynomial in the original input size, and in particular has at most $nmKM^2$ different masses across all actions and configurations, each with a different agent utility.

Next, consider the runtime of the main loop. The algorithm tries each choice of $\hat{u}_1, \ldots, \hat{u}_M$, and after preprocessing, there are $nmKM^2$ choices for each $\hat{u}_j$, leading to at most $(nmKM^2)^M$ choices of $\hat{u}_1, \ldots, \hat{u}_M$. We next consider the relevant aspects of the dynamic program, which is solved once per choice of $\hat{u}_1, \ldots, \hat{u}_M$. It is straightforward to verify that the computation of the ψ_is and $\overline{q}_{ij}^{\leq}$s can be done in polynomial time, and is not the bottleneck to the computation. The dimension of the memo table, meanwhile, is $n(M^2n)^M$, and each entry requires m constant-time operations to fill. Therefore, the runtime of the dynamic program is $O(mn(M^2n)^M)$. Since this is run once per loop, we obtain a runtime of $O(mn(M^2n)^M \cdot (nmKM^2)^M) = O(m^{M+1}n^{2M+1}K^M M^{4M})$, which is polynomial when M is constant.

4 Applications and Utility Alignment Analyses

This section now applies the utility configuration PTAS framework from Section 3 to the problems discussed in the introduction. For each problem, we first show how to reduce the problem to utility configuration. We then prove that each problem satisfies utility alignment (Definition 2), which we restate below:

Definition (Utility Alignment, restated)**.** *Given a non-increasing function f, an instance of utility configuration is* $f(q)$-utility aligned *if for any utility threshold U such that in the optimal configuration $\mathbf{C}^*$, $Pr[u^A \leq U] = q$, the principal utilities under $\mathbf{C}^*$ satisfy*

$$\mathbb{E}[u^P \mid u^A \leq U] \leq f(q)\mathbb{E}[u^P],$$

where u^A and u^P are the agent and principal utilities from the agent's preferred action under $\mathbf{C}^*$.

In Sections 4.1 and 4.2, we consider the delegation and unit demand pricing problems discussed in the introduction. For each of these problems, we show 2-utility alignment. In Section 4.3, we then present a broad result that implies utility alignment for several similar problems. The result shows that if utility alignment holds for individual actions, then it also holds for optimal configurations, albeit at a weaker level (though still strong enough to imply a PTAS). This result implies that the algorithm of the previous section is a PTAS for two variants of the delegated choice problem (one with randomly distributed bias, and another with an outside option), as well as the assortment optimization problem. We cover these in detail in Sections 4.3.1-4.3.3.

4.1 Delegated Choice

Recall that in the delegated choice problem, there are n actions, each with an associated bias b_i and randomly (independently) distributed nonnegative value v_i with distribution F_i. The principal does not know v_i but knows the biases, which are not random. We consider a generalization with random biases in Section 4.3.1. The principal must select a nonempty subset $S \subseteq [n]$. After the principal selects S, the agent observes $v_1, \dots, v_n$ and selects the action i^* maximizing $v_i + b_i$. The principal's utility is v_{i^*}.

This problem reduces to utility configuration as follows: for each action $i \in [n]$, create a corresponding action in the utility configuration problem, with two configurations. *in* and *out*. If action i is *in*, then (u_i^A, u_i^P) are drawn by drawing v_i from F_i and setting $(u_i^A, u_i^P) = (v_i + b_i, v_i)$. If i is *out*, then $(u_i^A, u_i^P) = (-\infty, 0)$ deterministically. There is a natural bijection between configurations and solutions to delegated choice: the principal includes i if and only if it is *in*. We next prove that delegated choice is 2-utility aligned.

Lemma 23. *Delegated choice is* 2*-utility aligned.*

Proof. Given an arbitrary delegation set S, let $i^*(S)$ denote the random variable representing the agent's favorite action from S, given the realization of $(v_1, \dots, v_n)$. Let OPT denote the optimal delegation set, and consider any utility level U for the agent. Let $X = \mathbb{E}[v_{i^*(\text{OPT})} \mid v_{i^*(\text{OPT})} + b_{i^*(\text{OPT})} \leq U]$ denote the principal's expected utility from the event that the agent's utility is below U. We will show that there exists a delegation set that gives the principal expected utility at least $X/2$, and hence $\mathbb{E}[v_{i^*(\text{OPT})}] \geq X/2$ as well, as OPT is optimal. To do so, we partition OPT into *good* actions $G = \{i \mid b_i \leq U - X/2\}$ and *bad* actions $B = \{i \mid b_i > U - X/2\}$. We claim that $\mathbb{E}[v_{i^*(G)}] \geq X/2$.

Let $\mathcal{E}$ denote the event that $v_{i^*(\text{OPT})} + b_{i^*(\text{OPT})} \leq U$, and $\mathcal{E}_G, \mathcal{E}_B$ the events that the agent's choice from OPT lies in G or B, respectively. First, note that conditioned on $\mathcal{E}$, it must be that $v_i \leq X/2$ for all $i \in B$:

$$v_i \leq U - b_i < U - U + X/2 = X/2.$$

One consequence of this fact is that good actions represent at least half of X. That is:

$$X = \mathbb{E}[v_{i^*(\text{OPT})} \mid \mathcal{E}]$$

$$
\begin{aligned}
&= \mathbb{E}[v_{i^*(\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + \mathbb{E}[v_{i^*(\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_B] \Pr[\mathcal{E}_B \mid \mathcal{E}] \\
&\le \mathbb{E}[v_{i^*(\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + \frac{X}{2},
\end{aligned}
$$

and hence $\mathbb{E}[v_{i^*(\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] \ge X/2$.

Meanwhile, let $\mathcal{E}_G^U$ denote the event that $v_{i^*(G)} + b_{i^*(G)} \le U$. Then conditioned on $\overline{\mathcal{E}_G^U}$, note that every good action yields utility at least $X/2$:

$$v_i > U - b_i \ge U - U + X/2 = X/2.$$

We can now show that $\mathbb{E}[v_{i^*(G)}] \ge X/2$. We have:

$$
\begin{aligned}
\mathbb{E}[v_{i^*(G)}] &= \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[v_{i^*(G)} | \mathcal{E}_G^U] + (1 - \Pr[\mathcal{E}_G^U]) \cdot \mathbb{E}[v_{i^*(G)} | \overline{\mathcal{E}_G^U}] \\
&\ge \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[v_{i^*(G)} | \mathcal{E}_G^U] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 \\
&= \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[v_{i^*(G)} | \mathcal{E}] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (54) \\
&= \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[v_{i^*(G)} | \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (55) \\
&= \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[v_{i^*(\mathrm{OPT})} | \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (56) \\
&\ge \Pr[\mathcal{E}_G^U] \cdot X/2 + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 \\
&= X/2.
\end{aligned}
$$

Line (54) follows from the independence of good and bad actions' values. Line (55) follows from the fact that values are nonnegative. Line (56) then follows from the fact that $\mathcal{E}_G$ implies that $i^*(\mathrm{OPT}) = i^*(G)$. The remaining lines follow from the facts above. □

We also show that the analysis above is tight.

Lemma 24. *Delegated choice is not c-utility aligned for any $c < 2$.*

Proof. Consider the following instance with 2 actions. The biases for these actions are $b_1 = M - 1 + 1/M$ and $b_2 = 0$.

$$
v_1 = \begin{cases} 1, & w.p.\ 1 - 1/M \\ 1 - 2/M, & w.p.\ 1/M \end{cases}
\qquad
v_2 = \begin{cases} M, & w.p.\ 1/M \\ 0, & w.p.\ 1 - 1/M \end{cases}
$$

In this case, it can be easily checked that the optimal delegation set includes both actions, which results in the following principal utility distribution:

$$
v_{i^*(\mathrm{OPT})} = \begin{cases} 1, & w.p.\ 1 - 1/M \\ M, & w.p.\ (1/M)^2 \\ 1 - 2/M, & w.p.\ (1 - 1/M)/M \end{cases}
$$

The total principal utility is equal to

$$(1 - 1/M) + (1/M) + (1 - 1/M)(1 - 2/M)/M = 1 + (1 - 1/M)(1 - 2/M)/M$$

Notice that the agent's utility from the $1/M$-quantile comes from the last two realizations of $v_{i^*(\text{OPT})}$. The principal's utility conditioned on the last two realizations is

$$(1/M) \cdot M + (1 - 1/M) \cdot (1 - 2/M) = 1 + (1 - 1/M)(1 - 2/M).$$

If delegated choice were c-utility aligned for some $c < 2$, it would have to be that

$$\frac{1 + (1 - 1/M)(1 - 2/M)}{1 + (1 - 1/M)(1 - 2/M)/M} \leq c,$$

which is a contradiction, since the limit of this expression (as $M \to \infty$) is 2. □

Applying the analysis of the PTAS from the previous section— in particular, Corollary 21— we immediately obtain the following:

Corollary 25. *For any fixed M, the algorithm from Section 3 with M bins is a $(1 - O(\log M/M))$-approximation for the delegated choice problem and runs in polynomial time.*

4.2 Unit-demand Pricing

We next consider the problem of pricing n items for sale to a unit-demand agent. Recall that for this problem, we have n items. Item i has value v_i for the buyer, which has distribution F_i. The buyer's values for the different items are independent. The principal (seller) selects a price $p_i \geq 0$ for each item i. The buyer then realizes their values and picks the single item i^* that maximizes their profit $v_{i^*} - p_{i^*}$, or leaves without buying an item if $v_i - p_i < 0$ for all i. The seller's utility is their revenue, equal to p_{i^*} (or 0 if no item is sold).

Unit-demand pricing reduces to utility configuration: each item will represent an action in the configuration instance. When the seller can use any positive real-valued price, the resulting utility configuration problem will have a continuum of configurations per action. While this is a mathematically well-defined optimization problem, the PTAS of the previous section requires there to be a finite number of configurations per action. This can be solved via standard price discretization techniques, e.g. those from Cai and Daskalakis [16]. We discuss the reduction from continuous to discrete pricing in Section 5, where we also show how to obtain a PTAS for unit-demand pricing with unbounded regular value distributions. For the time being, we prove that unit-demand pricing is 2-utility aligned for any set of prices available to the seller, as long as this set includes ∞ (i.e., not selling an item).

The reduction from unit-demand pricing to utility configuration is the following. Each item i corresponds to an action in the configuration problem. Each price p_i for item i corresponds to a configuration for action i. To draw the utilities for item i priced at p_i, draw v_i from F_i. If $v_i \geq p_i$, then $(u_i^A, u_i^P) = (v_i - p_i, p_i)$. Otherwise, $= (u_i^A, u_i^P) = (0, 0)$. Using a very similar argument to the one for delegation (Lemma 23), we can prove the following:

Lemma 26. *Unit-demand pricing is 2-utility aligned as long as the seller's set of allowable prices contains ∞.*

Proof. Given price vector $\mathbf{p}$, let $i^*(\mathbf{p})$ denote the buyer's utility-maximizing choice of item, or 0 if no item yields nonnegative utility. Then the seller's revenue is $\mathbb{E}[p_{i^*(\mathbf{p})}]$, where we adopt the convention that $p_0 = 0$ and $v_0 = 0$. Now let $\mathbf{p}$ denote the optimal price vector, and consider a utility level U for the buyer. Let $\mathcal{E}$ denote the event that $v_{i^*(\mathbf{p})} - p_{i^*(\mathbf{p})} \leq U$. Let $X = \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E}]$. We will show that there is a pricing $\mathbf{p}'$ where the seller earns at least $X/2$ in expectation. To pick $\mathbf{p}'$, let $p'_i = p_i$ for all i such that $p_i \geq X/2$, and $p'_i = \infty$ for all other items. Call the former set of items the *good* items G and the latter set the *bad* items B.

The claim will follow from two key observations. First, we observe that the good items represent at least half the seller's revenue conditioned on $\mathcal{E}$. To see this, let $\mathcal{E}_G$ denote the event that the agent buys a good item given $\mathbf{p}$, and $\mathcal{E}_B$ similarly for a bad item. We have:

$$\begin{aligned} X &= \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E}] \\ &= \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E} \cap \mathcal{E}_B] \Pr[\mathcal{E}_B \mid \mathcal{E}] \\ &\leq \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + \frac{X}{2}, \end{aligned}$$

and hence $\mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] \geq X/2$.

Now let $\mathcal{E}_G^U$ denote the event that the buyer's favorite item under $\mathbf{p}'$ yields utility less than U, i.e. $v_{i^*(\mathbf{p}')} - p_{i^*(\mathbf{p}')} \leq U$. The second observation is that conditioned on $\mathcal{E}_G^U$, the buyer is more likely to choose any $i \in G$ given $\mathbf{p}'$ than they are to choose i given $\mathbf{p}$ conditioned on $\mathcal{E}$. Intuitively, in the former scenario, item i has the same distribution but less competition. Formally, we have the following for all $i \in G$:

$$\begin{aligned} \Pr[v_i - p'_i \geq v_{i'} - p'_{i'}\ \forall i' \in G \mid \mathcal{E}_G^U] &= \Pr[v_i - p_i \geq v_{i'} - p_{i'}\ \forall i' \in G \mid \mathcal{E}_G^U] && (57) \\ &= \Pr[v_i - p_i \geq v_{i'} - p_{i'}\ \forall i' \in G \mid \mathcal{E}] && (58) \\ &\geq \Pr[v_i - p_i \geq v_{i'} - p_{i'}\ \forall i' \in [n] \mid \mathcal{E}]. && (59) \end{aligned}$$

Line (57) holds as $p_i = p'_i$ for all $i \in G$. Line (58) follows from the independence of values between items. Line (59) follows because we have added more constraints compared to the line before.

The following inequalities then prove the lemma:

$$\begin{aligned} \mathbb{E}[p'_{i^*(\mathbf{p}')}] &= \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[p'_{i^*(\mathbf{p}')} | \mathcal{E}_G^U] + (1 - \Pr[\mathcal{E}_G^U]) \cdot \mathbb{E}[p'_{i^*(\mathbf{p}')} | \overline{\mathcal{E}_G^U}] \\ &\geq \Pr[\mathcal{E}_G^U] \cdot \mathbb{E}[p'_{i^*(\mathbf{p}')} | \mathcal{E}_G^U] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (60) \\ &= \Pr[\mathcal{E}_G^U] \sum_{i \in G} p'_i \Pr[v_i - p'_i \geq v_{i'} - p'_{i'}\ \forall i' \in G \mid \mathcal{E}_G^U] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (61) \\ &\geq \Pr[\mathcal{E}_G^U] \sum_{i \in G} p_i \Pr[v_i - p_i \geq v_{i'} - p_{i'}\ \forall i' \in [n] \mid \mathcal{E}] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 && (62) \\ &= \Pr[\mathcal{E}_G^U] \mathbb{E}[p_{i^*(\mathbf{p})} \mid \mathcal{E} \cap \mathcal{E}_G] \Pr[\mathcal{E}_G \mid \mathcal{E}] + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 \\ &\geq \Pr[\mathcal{E}_G^U] \cdot X/2 + (1 - \Pr[\mathcal{E}_G^U]) \cdot X/2 \\ &= X/2. \end{aligned}$$

Line (60) holds because whenever there is some good item i with $v_{i^*(\mathbf{p}')} - p_{i^*(\mathbf{p}')} \geq U \geq 0$, the buyer purchases a good item under $\mathbf{p}'$. Line (61) rewrites the seller's revenue under $\mathcal{E}_G^U$ in terms of the purchase probabilities for the individual items. Line (62) holds because item i is only more likely to be bought if the bad actions are priced at ∞ than not, and because $p_i = p'_i$ for all $i \in G$. Since $\mathbf{p}'$ yields at least $X/2$, the optimal pricing must yield at least this much revenue as well, proving the lemma. □

While the framework of Section 3 is not compatible with infinitely many prices (and hence configurations of each action), for any fixed set of m allowable prices per item (including ∞), we obtain the following by applying Corollary 21.

Corollary 27. *For any fixed M, the algorithm from Section 3 with M bins is a $(1 - O(\log M/M))$-approximation for the unit-demand pricing problem with m possible prices, and runs in polynomial time.*

We show in Section 5 how to reduce the infinite-prices problem to the version with finitely many prices. For discrete value distributions, the number of prices required will be polynomial in the bit complexity of the values, leading to a PTAS for that problem as well. For unbounded regular distributions, the number of prices required will be polynomial in the number of items.

4.3 Local to Global Utility Alignment

Sections 4.1 and 4.2 showed that both delegated choice and unit-demand pricing were 2-utility aligned. Both proofs relied heavily on the structures of the problems: both involved imposing a threshold on an existing solution (either based on bias or price) and discarding all items or actions below the threshold. For general instances of utility configuration, such an approach may not be possible. However, in this section, we show a weaker utility alignment result that applies to a wide range of natural problems. The guarantee will still be strong enough to obtain a PTAS using the algorithm of Section 3.

Definition 2 of utility alignment is a statement about optimal solutions: an instance is utility aligned if, under the optimal solution, the principal's and agent's utilities are aligned. The result of this section considers an item-wise version of this same condition, defined below.

Definition 28. *For non-increasing function f, an instance of utility configuration is* locally $f(q)$-utility aligned *if for every action i, configuration j and utility level U such that* $\Pr[u_i^A \leq U] = q$, *we have:*

$$\mathbb{E}_{(u_i^A, u_i^P) \sim F_{ij}}[u_i^P | u_i^A \leq U] \leq f(q) \cdot \mathbb{E}_{(u_i^A, u_i^P) \sim F_{ij}}[u_i^P].$$

Note that Definition 28 is satisfied by both delegated choice (Section 4.1) and unit-demand pricing (Section 4.2), each with $f(q)$ identically equal to 1 (i.e. both problems are *locally* 1-*utility aligned*). For example, in delegated choice, the agent's and principal's utilities are perfectly and positively correlated: for actions which are included in the delegation set, $u_i^P = v_i$ increases only if $u_i^A = v_i + b_i$ increases. The following result shows that local utility alignment (Definition 28) is sufficient for utility alignment overall (Definition 2), albeit with a weaker dependence of f on q than the results for delegated choice and pricing. The result also requires one more natural property — which again holds for all problems in this paper — and which states that the principal can choose to completely discard any action. For example, in delegation, the principal's main decision is exactly this choice of including or excluding actions.

Definition 29. *A utility configuration instance has* excludable actions *if for every action i, there is a configuration such that $u_i^A = -\infty$ deterministically in that configuration.*

We say action i is *excluded* in a configuration $\mathbf{C}$ if C_i is such that $u_i^A = -\infty$ deterministically. The result is then the following:

Lemma 30. *Any instance of utility configuration with excludable actions that satisfies local $f(q)$-utility alignment also satisfies* $\max(4f(q), 4/\sqrt{q})$*-utility alignment.*

As a consequence of Lemma 30, local $O(1/\sqrt{q})$-utility alignment also translates into overall $O(1/\sqrt{q})$-utility alignment. We will show in Sections 4.3.1-4.3.3 that two generalizations of the delegated choice problem, as well as the assortment optimization problem from the revenue management literature, all in fact satisfy local 1-utility alignment, and hence are $O(1/\sqrt{q})$-aligned overall.

Proof of Lemma 30. Let OPT denote the optimal configuration, and given any configuration $\mathbf{C}$, let $i^*(\mathbf{C})$ be the random variable denoting the agent's choice under $\mathbf{C}$. Given utility threshold U, let $\mathcal{E}$ denote the event that $u^A_{i^*(\mathrm{OPT})} \leq U$, and let $q = \Pr[\mathcal{E}]$. Further let $X = \mathbb{E}[u^P_{i^*(\mathrm{OPT})} | u^A_{i^*(\mathrm{OPT})} \leq U]$. The proof considers two cases. The main goal is to break the actions into two groups with roughly equal contribution to X, and to show that discarding the worse of these two groups guarantees the principal utility at least $\sqrt{q}$. The only case where this strategy fails is when one action contributes a large fraction of X. In this case, local utility alignment implies that this action alone will provide high utility.

Case 1. Assume that for every action i, the contribution to X given by $\mathbb{E}[u^P_{i^*(\mathrm{OPT})} | \mathcal{E} \cap i^*(\mathrm{OPT}) = i] \Pr[i^*(\mathrm{OPT}) = i \mid \mathcal{E}]$ is at most $X/4$. Then we may partition the actions into two sets S_1 and S_2 such that for $j \in \{1, 2\}$,

$$\mathbb{E}[u^P_{i^*(\mathrm{OPT})} | \mathcal{E} \cap \mathcal{E}_j] \Pr[\mathcal{E}_j \mid \mathcal{E}] \geq X/4,$$

where $\mathcal{E}_j$ denotes the event that $i^*(\mathrm{OPT}) \in S_j$. Let $\mathbf{C}^j$ denote the configuration where actions not in S_j are excluded, and those in S_j are configured as in OPT. Let $i^*(S_j, \mathrm{OPT})$ denote the agent's favorite action from S_j under OPT, and recall that $i^*(\mathbf{C}^j)$ is the agent's favorite action under $\mathbf{C}^j$. Note that $i^*(S_j, \mathrm{OPT})$ and $i^*(\mathbf{C}^j)$ have the same distribution. Let $\mathcal{E}^U_j$ denote the event that the agent's favorite action under $\mathbf{C}^j$ gives them utility at most U, or equivalently that $u^A_{i^*(S_j)} \leq U$, and let $\epsilon_j = \Pr[\mathcal{E}^U_j]$.

The result then follows from two observations. First, $\mathbf{C}^j$ guarantees the principal at least utility $X/4 \cdot \epsilon_j$. In particular, we may express the principal's utility from this configuration as:

$$\begin{aligned}
\mathbb{E}[u^P_{i^*(\mathbf{C}^j)}] &\geq \mathbb{E}[u^P_{i^*(\mathbf{C}^j)} \mid \mathcal{E}^U_j] \cdot \epsilon_j && (63)\\
&= \mathbb{E}[u^P_{i^*(S_j,\mathrm{OPT})} \mid \mathcal{E}] \cdot \epsilon_j && (64)\\
&\geq \mathbb{E}[u^P_{i^*(S_j,\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_j] \Pr[\mathcal{E}_j \mid \mathcal{E}] \cdot \epsilon_j && (65)\\
&\geq \mathbb{E}[u^P_{i^*(\mathrm{OPT})} \mid \mathcal{E} \cap \mathcal{E}_j] \Pr[\mathcal{E}_j \mid \mathcal{E}] \cdot \epsilon_j && (66)\\
&\geq X/4 \cdot \epsilon_j && (67)
\end{aligned}$$

Line (63) follows from the fact that the principal utility is nonnegative and the definition of ϵ_j. Line (64) follows from the independence of utilities across actions. Line (65) then again follows from the non-negativity of principal utilities. Line (66) follows from the fact that in event $\mathcal{E}_j$, the agent chooses an action from S_j, so $i^*(\mathrm{OPT}) = i^*(S_j, \mathrm{OPT})$. Finally, Line (67) holds because $\mathbb{E}[u^P_{i^*(\mathrm{OPT})} | \mathcal{E} \cap \mathcal{E}_j] \Pr[\mathcal{E}_j \mid \mathcal{E}] \geq X/4$.

To conclude the proof of utility alignment in this case, notice that since S_1 and S_2 partition the actions, $q = \epsilon_1 \cdot \epsilon_2$. Hence, $\max_{j \in \{1,2\}} \epsilon_j \geq \sqrt{q}$. Hence, choosing the j with the larger ϵ_j yields a set with expected principal utility at least $X\sqrt{q}/4$, as desired.

Case 2. Assume some action i contributes at least a quarter of X, i.e. $\mathbb{E}[u^P_{i^*(\mathrm{OPT})}|\mathcal{E} \cap i^*(\mathrm{OPT}) = i] \Pr[i^*(\mathrm{OPT}) = i \mid \mathcal{E}] \geq X/4$. Consider the configuration $\hat{\mathbf{C}}^i$ where i is configured as in OPT, but all other actions are excluded. Then $\hat{\mathbf{C}}^i$ guarantees the principal utility at least $X/(4f(q))$. To see this, let $\mathcal{E}^U_i$ denote the event that $u^A_i \leq U$, and $\mathcal{E}_i$ the event that $i^*(\mathrm{OPT}) = i$. Then we have the following inequalities, justified after their statement:

$$\begin{aligned}
\mathbb{E}[u^P_i] &\geq \frac{1}{f(\Pr[u^A_i \leq U])} \mathbb{E}[u^P_i \mid \mathcal{E}^U_i] && (68)\\
&\geq \frac{1}{f(q)} \mathbb{E}[u^P_i \mid \mathcal{E}^U_i] && (69)\\
&= \frac{1}{f(q)} \mathbb{E}[u^P_i \mid \mathcal{E}] && (70)\\
&\geq \frac{1}{f(q)} \mathbb{E}[u^P_i \mid \mathcal{E} \cap i^*(\mathrm{OPT}) = i] \Pr[i^*(\mathrm{OPT}) = i \mid \mathcal{E}] && (71)\\
&\geq \frac{X}{4f(q)}. && (72)
\end{aligned}$$

Line (68) follows from local utility alignment. Line (69) holds because f is non-increasing, and because $\Pr[u^A_i \leq U] \geq \Pr[u^A_{i^*(\mathrm{OPT})} \leq U] = q$. Line (70) holds because of the independence between actions. Line (71) holds because principal utilities are nonnegative. Finally, Line (72) holds because i contributes at least $1/4$ of X. The overall utility alignment is the worse of the two guarantees from the two cases. This leads to the stated bound. □

In the rest of this section, we prove that several further applications of interest can be reduced to utility configuration, and furthermore that these applications are all locally 1-utility aligned. Consequently, all the applications below are also $4/\sqrt{q}$-utility aligned overall and hence the algorithm of Section 3 is a PTAS. In Sections 4.3.1 and 4.3.2, we consider two generalizations of the delegated choice problem. In Section 4.3.1, we allow each action i to have bias b_i, which is random, but independent from the action's value v_i, as well as the biases and values of other actions. Section 4.3.2 then considers a variant of delegation where the agent has an outside option, an action which gives the principal no utility and cannot be excluded from the delegation set. Finally, Section 4.3.3 considers the well-studied assortment optimization problem from revenue management.

4.3.1 Delegation with Independent Random Biases

In the delegated choice problem considered in Section 4.1, each action's value v_i was randomly distributed, and the action's bias b_i was fixed and known to the principal. We now allow the bias b_i to be random as well, with the distribution also known to the principal. This can capture scenarios where the bias stems from, e.g. measurement error, as well as from uncertainty of the principal regarding the agent's preferences. For example, one could consider $b_i \sim N(0, \sigma^2_i)$, with variance that varies across actions. The agent then myopically chooses the action that maximizes the measured value.

This problem reduces to utility configuration in exactly the same way that the fixed-bias version does. Each action has an *in* configuration and an *out* configuration. The *out* configuration gives agent $-\infty$ and principal 0 utility deterministically, as in the vanilla delegation model. The distribution of (u_i^A, u_i^P) in the *in* configuration is equal to that of $(v_i + b_i, v_i)$, drawing v_i and b_i independently from their respective distributions. We then have the following local utility alignment guarantee:

Lemma 31. *Delegation with randomly distributed bias is locally* 1*-utility aligned.*

Proof. We only need to consider the *in* configuration for any action i. Fixing a utility threshold U, we have:

$$\begin{aligned}\mathbb{E}[v_i \mid v_i + b_i \leq U] &= \mathbb{E}_{b_i}\left[\mathbb{E}_{v_i}[v_i \mid v_i \leq U - b_i] \mid b_i \leq U\right] \\ &\leq \mathbb{E}_{b_i}\left[\mathbb{E}_{v_i}[v_i] \mid b_i \leq U\right] \\ &= \mathbb{E}[v_i].\end{aligned}$$

This immediately implies the desired local utility alignment result. □

Note that Lemma 31 will also tend to hold when v_i and b_i satisfy natural positive correlation conditions, via essentially the same proof. Combining Lemma 31 with the local-to-global result of Lemma 30 and the guarantee of Corollary 22, we obtain:

Corollary 32. *For any fixed M, the algorithm from Section 3 with M bins is a $(1 - O(1/\sqrt{M}))$-approximation for the delegated choice problem with random biases and runs in polynomial time.*

4.3.2 Delegation with An Outside Option

The second generalization to delegated choice we consider is delegation with an *outside option.* We assume the agent has an action that they may take, which yields utility $v_0 = 0$ for the principal, but utility b_0 for the agent. The agent then chooses the action maximizing $v_i + b_i$ if this quantity exceeds b_0, or takes the outside option otherwise, yielding 0 utility for the principal. As in Section 4.3.1, we allow biases, including b_0, to be random and independent of each other and the values.

The obvious approach to reduce this problem to utility alignment fails: if we follow the reduction of Section 4.3.1, then add a 0-th action corresponding to the outside option, this 0-th action needs to have one configuration (the "*in*" configuration), as the principal cannot exclude the outside option. However, this construction now fails to satisfy the excludable actions assumption (Definition 29).

Instead, we adjust the utility distributions for non-outside actions in the following way. As before, we let the *out* configuration for each action yield utility $-\infty$ for the agent and 0 for the principal. Now for action i in the *in* configuration, draw (u_i^A, u_i^P) in the following way: first draw v_i and b_i from their respective distributions. Set $u_i^A = v_i + b_i$, as before. Let u_i^P be set as $u_i^P = v_i \cdot \Pr_{b_0}[v_i + b_i \geq b_0]$. Given a configuration of actions in this utility configuration instance, it is straightforward to see that the principal's expected utility is the same as if they took the corresponding actions in the delegation instance. Intuitively, the utilities above simulate the following two-step process: first, the agent chooses their favorite action other than the outside option, then switches to the outside option if b_0 is high enough. Intuitively, this preserves local utility alignment: if the agent's utility is higher, then they are less likely to switch to the outside option and deny utility to the principal.

Lemma 33. *Delegation with an outside option and randomly distributed bias is locally* 1*-utility aligned.*

Proof. The proof generalizes that of Lemma 31. Let $a(v_i, b_i) = v_i \Pr_{b_0}[v_i + b_i \geq b_0]$ denote the contribution of action i after factoring in the outside option. Note that $a(v_i, b_i)$ is nondecreasing in both v_i and b_i. Fixing a utility threshold U, let G_i^U be the conditional distribution of b_i where $b_i \leq U$ and let $F^{(U-b_i)}$ be the conditional distribution of v_i where $v_i \leq U - b_i$. Then can write

$$\begin{aligned}
\mathbb{E}[a(v_i, b_i) \mid v_i + b_i \leq U] &= \mathbb{E}_{b_i \sim G_i^U}\left[\mathbb{E}_{v_i \sim F_i^{(U-b_i)}}[a(v_i, b_i)]\right] \\
&\leq \mathbb{E}_{b_i \sim G_i^U}\left[\mathbb{E}_{v_i \sim F_i}[a(v_i, b_i)]\right] \\
&\leq \mathbb{E}_{b_i \sim G_i}\left[\mathbb{E}_{v_i \sim F_i}[a(v_i, b_i)]\right] \\
&= \mathbb{E}[a(v_i, b_i)],
\end{aligned}$$

where the inequalities follow from the fact G and F respectively first-order stochastically dominate G^U and $F^{(U-b)}$, and because $a(v_i, b_i)$ is nondecreasing in v_i and b_i. This implies local 1-utility alignment. □

Lemmas 33 and Lemma 30 combine with Corollary 22 to imply:

Corollary 34. *For any fixed M, the algorithm from Section 3 with M bins is a $(1 - O(1/\sqrt{M}))$-approximation for the delegated choice problem with an outside option, and runs in polynomial time.*

4.3.3 Assortment Optimization

The final application we consider is assortment optimization, with a random utility model and independent utilities across items. In the assortment optimization problem, there are n fixed-price items, where item i has price p_i. The seller's decision is to pick an inventory, a subset of $\{1, \ldots, n\}$ to offer for sale to a single buyer. The buyer's value for item i is v_i, drawn independently from distribution F_i, yielding utility $v_i - p_i$ for buying item i. The buyer also has an outside option, which yields utility $u_0 \sim F_0$ for the buyer and 0 for the seller. The seller seeks to maximize their revenue in expectation over the random choice of the agent.

This problem reduces to delegation with an outside option (and, by composition, to utility configuration) in the following way. We define delegation value $v_i' = p_i$ and (random) delegation bias b_i' distributed as $v_i - 2p_i$. Note that since p_i is fixed, v_i' is fixed and hence v_i' and b_i' are independent. Include in the delegation instance an outside option with agent utility distributed as F_0. The resulting instance is equivalent to the original assortment optimization problem. Through the reduction and utility alignment results of Section 4.3.2, then, we obtain the following:

Lemma 35. *Assortment optimization with independent utilities is locally* 1*-utility aligned.*

Corollary 36. *For any fixed M, the algorithm from Section 3 with M bins is a $(1 - O(1/\sqrt{M}))$-approximation for assortment optimization, and runs in polynomial time.*

5 Unit-demand Pricing

The algorithm for utility configuration in Section 3 assumed a finite number m of configurations per action. Its runtime was polynomial in this number, among the other parameters of the input size. However, the usual statement of the unit-demand pricing problem allows the seller to offer arbitrary nonnegative prices, and the reduction from unit-demand pricing to utility configuration translates every price for each item i into a configuration for i's corresponding action, leading to an instance with a continuum of configurations. Since this is clearly not computationally tractable, in this section we re-purpose lemmas from prior work on the problem [16] to show that a number of prices per item — and hence configurations per action — that is polynomial in the input size suffices for a $(1-\epsilon)$-approximation, and can be computed in polynomial time. Combined with our results, this implies a PTAS for the case with discrete values but unrestricted prices. This yields the first multiplicative PTAS for this case, improving on the additive PTAS of Cai and Daskalakis [16]. These results are in Section 5.1.

We also study the case where values are continuously distributed but satisfy the standard Myerson regularity condition. In this case, Cai and Daskalakis [16] give a quasi-polynomial time approximation scheme for the regular case, based on a novel lemma for truncating the value distributions, and another lemma for discretizing it into polynomially many mass points. We combine these with our framework to produce the first PTAS for this problem. These results can be found in Section 5.2.

5.1 Finite Value Distributions

In the discrete-values version of unit-demand pricing, each item i's value v_i is drawn independently from a finitely supported discrete distribution F_i. Note that even with discrete distributions, it may be in the seller's interest to offer prices that are not equal to any mass for any distribution. (See, e.g. Cai and Daskalakis [16], Appendix J for an example where this holds.) Consequently, it is necessary to show that finitely many prices suffice for a $(1-\epsilon)$-approximation, and that these prices can be computed efficiently. Fortunately, we may borrow the following directly from Cai and Daskalakis [16]:

Lemma 37 (Cai and Daskalakis [16], Lemma 25)**.** *Suppose that the value distributions in an instance of the item pricing problem are independent and supported on $[u_{\min}, u_{\max}] \subset \mathbb{R}_+$. For any $\varepsilon \in (0, 1/2)$, consider the following finite set of prices:*

$$\mathcal{P}_\varepsilon = \left\{ p_i \mid p_i = \frac{1+\varepsilon^2-\varepsilon}{(1-\varepsilon^2)^i} \cdot u_{\min}, i \in [\lfloor \log_{\frac{1}{1-\varepsilon^2}} (u_{\max}/u_{\min}) \rfloor] \right\}.$$

For any price vector $P \in [u_{\min}, u_{\max}]^n$, there exists a price vector P' such that $p'_i \in \mathcal{P}_\varepsilon$ and $p'_i \in [1-\varepsilon, 1+\varepsilon^2-\varepsilon] \cdot p_i$, for all i. The expected revenue achieved by the two price vectors satisfies $\mathcal{R}_{P'} \geq (1-2\varepsilon)\mathcal{R}_P$.

Given that prices below $u_{\min}$ are clearly suboptimal, and above $u_{\max}$ are dominated by $u_{\max}$, Lemma 37 implies that we may obtain a multiplicative PTAS for the discrete-values case by first discretizing the prices, then applying the algorithm from Section 3 to the version with discrete prices. Hence, we have:

Corollary 38. *There exists a multiplicative PTAS for the unit-demand pricing problem with independent, finitely supported, discretely distributed values.*

5.2 Unbounded Regular Distributions

We now consider a version of the unit-demand pricing problem where values are continuously distributed and possibly unbounded. Without some tail condition on the distribution, this problem is intractable. A standard condition is *Myerson regularity*, which states that the virtual value function for each distribution F_i with density f_i, given by $\phi_i(v_i) = v_i - (1 - F_i(v_i))/f_i(v_i)$, is non-decreasing in v_i. For regular distributions, Cai and Daskalakis [16] give a QPTAS for unit-demand pricing, computing a $(1-\epsilon)$-approximation in time polynomial in $n^{\text{polylog}(n)}$. In addition to the price discretization lemma in the previous section, they give a *truncation lemma* and a *value discretization lemma*. The number of prices they produce in the end is dependent on the number of items, and the dependence of their algorithm's runtime on the number of prices is exponential. Our algorithm has a polynomial dependence on the number of prices, which allows us to improve the result to a PTAS for the regular case.

We adopt the same model of distributional access as [16]: their results (and hence our applications of these results as well) hold as long as the algorithm has access to one of the following: (1) explicit access to the CDFs, (2) oracle access to the CDFs, along with information about a fixed quantile of the distributions (e.g. the median); such an anchoring point is necessary, or else the algorithm would face the hopeless task of guessing the distribution's scale, (3) sample access to the distribution, in which case the guarantees hold with high probability over the samples. For more details, see Appendix B of Cai and Daskalakis [16].

Given any of the three forms of access above, the first step is to truncate the potentially unbounded regular distribution. The following lemma, again from Cai and Daskalakis [16], accomplishes this goal. Given an instance with independent, unbounded regular distributions, Cai and Daskalakis [16] show that it is possible to create a corresponding instance with distributions bounded in a polynomially-sized interval. Given near-optimal prices for this new instance, it is possible to construct near-optimal prices for the original instance in polynomial time. Formally:

Lemma 39 (Cai and Daskalakis [16], Theorem 20)**.** *Let $\mathcal{V}$ be a collection of mutually independent regular random variables. Then there exists some $\alpha(\mathcal{V}) > 0$ such that for any $\epsilon \in (0,1)$, there is a reduction from $(1-\epsilon)$-approximately optimal pricing for $\mathcal{V}$ to $(1-\theta(\epsilon/n))$-approximately optimal pricing for $\tilde{\mathcal{V}}$, where $\tilde{\mathcal{V}}$ is a collection of mutually independent random variables supported on $[\frac{\epsilon\alpha}{4n^4}, \frac{4n^4\alpha}{\epsilon^3}]$. Moreover, α is efficiently computable from the distributions of v_is, and for all ϵ, the runtime of the reduction is polynomial in the input size and $1/\epsilon$.*

Finally, Cai and Daskalakis [16] give a discretization lemma for bounded value distributions. As with Lemma 39 for truncating the distributions, they show that it is possible to convert continuous distributions into discrete distributions and to then convert approximately optimal prices for the discrete distributions into approximately optimal prices for the original instance. The number of values in the support of the distributions is polylogarithmic in the ratio of the highest to the lowest value in the original distribution. Formally:

Lemma 40 (Cai and Daskalakis [16], Lemma 30)**.** *Let $\{v_i\}_{i\in[n]}$ be a product distribution over values supported on range $[u_{min}, u_{max}] \subset \mathbb{R}_+$, and $r = \frac{u_{max}}{u_{min}}$. Then for any $\delta \in (0, (4\lceil\log_2 r\rceil)^{4/3})$, there exists a collection of mutually independent random variables $\{\hat{v}_i\}_{i\in[n]}$ which are supported on a set of $O(\log r/\delta^2)$ values that satisfy the following properties:*

1. *The optimal revenue under $\{\hat{v}_i\}_{i\in[n]}$ is at least a $(1-3\delta^{1/8})$-fraction of that under $\{v_i\}_{i\in[n]}$.*
2. *For any $\rho \in (0, 1/2)$ and any price vector $\mathbf{p}$ that yield revenue at least a $(1-\rho)$-fraction of optimal for $\{\hat{v}_i\}_{i\in[n]}$, we can construct in polynomial time in the description of $\mathbf{p}$ and $1/\delta$ another price vector $\hat{\mathbf{p}}$ such that $\hat{\text{p}}$rices obtain at least a $(1-7\delta^{1/8}-\rho)$ fraction of the optimal for revenue for $\{v_i\}_{i\in[n]}$.*

If u_{min} and u_{max} are explicitly known, we can compute the distribution of $\{\hat{v}_i\}_{i\in[n]}$s and their support in time polynomial in the description of $\{v_i\}_{i\in[n]}$, the bit complexity of u_{min} and u_{max}, and in $1/\delta$.

We may combine Lemmas 37, 39, and 40 with our framework to obtain a PTAS for the unbounded regular case of unit-demand pricing. In more detail, given distributions $\{v_i\}_{i\in[n]}$, we may apply Lemma 39 to truncate the value distributions to new distributions $\{\hat{v}_i\}_{i\in[n]}$ to range $[\frac{\epsilon\alpha}{4n^4}, \frac{4n^4\alpha}{\epsilon^3}]$. Given the truncated distributions $\{\hat{v}_i\}_{i\in[n]}$, we may discretize the value distributions using Lemma 40 to obtain new distributions $\{\tilde{v}_i\}_{i\in[n]}$ with support polynomial in n and $1/\epsilon$. Then given $\{\tilde{v}_i\}_{i\in[n]}$, we may apply Lemma 37 to obtain a restricted set of $\text{poly}(n, 1/\epsilon)$ prices for the problem. We may then apply our framework to obtain near-optimal prices for $\{\tilde{v}_i\}_{i\in[n]}$ from the restricted set in time polynomial in n and the number of prices and the number of mass points in $\{\tilde{v}_i\}_{i\in[n]}$ (which, in turn, are polynomial in n for any fixed approximation error ϵ). Then, given the prices computed by our framework, we may back out prices for the original instance using Lemmas 40 and 39. We conclude with the following:

Corollary 41. *For any ϵ, there is an algorithm which, for any independent regular distributions, finds prices within a $(1-\epsilon)$-factor of the optimal in time polynomial in the number of items n.*

References

[1] Gagan Aggarwal, Tomás Feder, Rajeev Motwani, and An Zhu. Algorithms for multi-product pricing. In *Automata, Languages and Programming: 31st International Colloquium, ICALP 2004, Turku, Finland, July 12-16, 2004. Proceedings 31*, pages 72–83. Springer, 2004.

[2] Ricardo Alonso and Niko Matouschek. Optimal delegation. *The Review of Economic Studies*, 75(1):259–293, 2008.

[3] Manuel Amador and Kyle Bagwell. Tariff revenue and tariff caps. *American Economic Review*, 102(3):459–65, 2012.

[4] Ali Aouad, Vivek Farias, Retsef Levi, and Danny Segev. The approximability of assortment optimization under ranking preferences. *Operations Research*, 66(6):1661–1669, 2018.

[5] Ali Aouad, Jacob Feldman, and Danny Segev. The exponomial choice model for assortment optimization: An alternative to the mnl model? *Management Science*, 69(5):2814–2832, 2023.

[6] Mark Armstrong and John Vickers. A model of delegated project choice. *Econometrica*, 78 (1):213–244, 2010.

[7] Kamyar Azizzadenesheli, Trung Dang, Aranyak Mehta, Alexandros Psomas, and Qian Zhang. Reward selection with noisy observations. *arXiv preprint arXiv:2307.05953*, 2023.

[8] Moshe Babaioff, Nicole Immorlica, Brendan Lucier, and S. Matthew Weinberg. A simple and approximately optimal mechanism for an additive buyer. *J. ACM*, 67(4), 2020.

[9] Théo Barré, Omar El Housni, and Andrea Lodi. Assortment optimization with visibility constraints. In *Integer Programming and Combinatorial Optimization: 25th International Conference, IPCO 2024, Wroclaw, Poland, July 3–5, 2024, Proceedings*, page 124–138, 2024.

[10] Curtis Bechtel and Shaddin Dughmi. Delegated stochastic probing. In *12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference*, volume 185 of *LIPIcs*, pages 37:1–37:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[11] Curtis Bechtel and Shaddin Dughmi. Efficient multi-agent delegated search. *CoRR*, abs/2411.00181, 2024.

[12] Curtis Bechtel, Shaddin Dughmi, and Neel Patel. Delegated pandora's box. In *EC '22: The 23rd ACM Conference on Economics and Computation, Boulder, CO, USA, July 11 - 15, 2022*, pages 666–693. ACM, 2022.

[13] Gerardo Berbeglia and Gwenaël Joret. Assortment optimisation under a general discrete choice model: A tight analysis of revenue-ordered assortments. *Algorithmica*, 82(4):681–720, 2020.

[14] Pirmin Braun, Niklas Hahn, Martin Hoefer, and Conrad Schecker. Delegated online search. *Artif. Intell.*, 334:104171, 2024.

[15] Patrick Briest and Piotr Krysta. Buying cheap is expensive: Hardness of non-parametric multi-product pricing. In *SODA*, volume 7, pages 716–725. Citeseer, 2007.

[16] Yang Cai and Constantinos Daskalakis. Extreme-value theorems for optimal multidimensional pricing. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 522–531. IEEE Computer Society, 2011.

[17] Yang Cai and Mingfei Zhao. Simple mechanisms for subadditive buyers via duality. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, page 170–183, 2017.

[18] Shuchi Chawla and J. Benjamin Miller. Mechanism design for subadditive agents via an ex ante relaxation. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, page 579–596, 2016.

[19] Shuchi Chawla, Jason D. Hartline, and Robert Kleinberg. Algorithmic pricing via virtual valuations. In *Proceedings of the 8th ACM Conference on Electronic Commerce*, EC '07, page 243–251, 2007.

[20] Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, page 311–320. Association for Computing Machinery, 2010.

[21] Wei Chen, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu. Combinatorial multi-armed bandit with general reward functions. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 1659–1667, 2016.

[22] Xi Chen, Ilias Diakonikolas, Dimitris Paparas, Xiaorui Sun, and Mihalis Yannakakis. The complexity of optimal multidimensional pricing. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, page 1319–1328, 2014.

[23] Xi Chen, Ilias Diakonikolas, Anthi Orfanou, Dimitris Paparas, Xiaorui Sun, and Mihalis Yannakakis. On the complexity of optimal lottery pricing and randomized mechanisms. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 1464–1479, 2015.

[24] Xi Chen, Will Ma, David Simchi-Levi, and Linwei Xin. Assortment planning for recommendations at checkout under inventory constraints. *Mathematics of Operations Research*, 49(1): 297–325, 2024.

[25] James M. Davis, Guillermo Gallego, and Huseyin Topaloglu. Assortment optimization under variants of the nested logit model. *Operations Research*, 62(2):250–273, 2014.

[26] Antoine Désir, Vineet Goyal, and Jiawei Zhang. Technical note—capacitated assortment optimization: Hardness and approximation. *Operations Research*, 70(2):893–904, 2022.

[27] Paul Dütting, Michal Feldman, Thomas Kesselheim, and Brendan Lucier. Prophet inequalities made easy: Stochastic optimization by pricing nonstochastic inputs. *SIAM Journal on Computing*, 49(3):540–582, 2020.

[28] Paul Dütting, Thomas Kesselheim, and Brendan Lucier. An o(log log m) prophet inequality for subadditive combinatorial auctions. *SIGecom Exch.*, 18(2):32–37, 2020.

[29] Jacob Feldman, Alice Paul, and Huseyin Topaloglu. Technical note—assortment optimization with small consideration sets. *Operations Research*, 67(5):1283–1299, 2019.

[30] Jacob B. Feldman and Huseyin Topaloglu. Revenue management under the markov chain choice model. *Operations Research*, 65(5):1322–1342, 2017.

[31] Ashish Goel, Sudipto Guha, and Kamesh Munagala. Asking the right questions: model-driven optimization using probes. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, PODS '06, page 203–212, 2006.

[32] Sergiu Hart and Noam Nisan. The menu-size complexity of auctions. In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce*, EC '13, page 565–566, 2013.

[33] Bengt Holmström. *On Incentives and Control in Organizations.* Stanford University, 1977.

[34] Nicole Immorlica, Brendan Lucier, Jieming Mao, Vasilis Syrgkanis, and Christos Tzamos. Combinatorial assortment optimization. *ACM Trans. Econ. Comput.*, 9(1), 2021.

[35] Yaonan Jin and Pinyan Lu. Benchmark-tight approximation ratio of simple mechanism for a unit-demand buyer. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1251–1259. IEEE Computer Society, 2024.

[36] Ali Khodabakhsh, Emmanouil Pountourakis, and Samuel Taggart. Simple delegated choice. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 569–590, 2024.

[37] Jon Kleinberg and Robert Kleinberg. Delegated search approximates efficient search. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, page 287–302. Association for Computing Machinery, 2018.

[38] Pravesh Kothari, Sahil Singla, Divyarthi Mohan, Ariel Schvartzman, and S. Matthew Weinberg. Approximation Schemes for a Unit-Demand Buyer with Independent Items via Symmetries . In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 220–232, 2019.

[39] Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bulletin of the American Mathematical Society*, 83(4):745 – 747, 1977.

[40] Guang Li, Paat Rusmevichientong, and Huseyin Topaloglu. The d-level nested logit model: Assortment and price optimization problems. *Operations Research*, 63(2):325–342, 2015.

[41] Xinye Li and Andrew Chi-Chih Yao. On revenue maximization for selling multiple independently distributed items. *Proc. Natl. Acad. Sci. USA*, 110(28):11232–11237, 2013.

[42] Mohammad Mahdian, Jieming Mao, and Kangning Wang. Regret minimization with noisy observations. In *Proceedings of the 24th ACM Conference on Economics and Computation*, EC '23, 2023.

[43] David Martimort and Aggey Semenov. Continuity in mechanism design without transfers. *Economics Letters*, 93(2):182–189, 2006.

[44] Aranyak Mehta, Uri Nadav, Alexandros Psomas, and Aviad Rubinstein. Hitting the high notes: subset selection for maximizing expected order statistics. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, 2020.

[45] Nahum D. Melumad and Toshiyuki Shibano. Communication in settings with no. transfers. *RAND Journal of Economics*, 22(2):173–198, 1991.

[46] Alon Rieger and Danny Segev. Quasi-polynomial time approximation schemes for assortment optimization under mallows-based rankings. *Math. Program.*, 208(1):111–171, 2024.

[47] Aviad Rubinstein and S. Matthew Weinberg. Simple mechanisms for a subadditive buyer and applications to revenue monotonicity. *ACM Trans. Econ. Comput.*, 6(3–4), 2018.

[48] Paat Rusmevichientong, Benjamin Van Roy, and Peter W Glynn. A nonparametric approach to multiproduct pricing. *Operations Research*, 54(1):82–98, 2006.

[49] Paat Rusmevichientong, David Shmoys, Chaoxu Tong, and Huseyin Topaloglu. Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management*, 23(11):2023–2039, 2014.

[50] Ester Samuel-Cahn. Comparison of threshold stop rules and maximum for independent random variables. *Annals of Probability*, 12(4):1212–1216, 1984.

[51] Danny Segev and Sahil Singla. Efficient approximation schemes for stochastic probing and prophet problems. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, EC '21, page 793–794, 2021.

[52] Suho Shin, Keivan Rezaei, and MohammadTaghi Hajiaghayi. Delegating to multiple agents. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*, pages 1081–1126. ACM, 2023.

[53] Kalyan Talluri and Garrett van Ryzin. Revenue management under a general discrete choice model of consumer behavior. *Management Science*, 50(1):15–33, 2004.

[54] Martin L. Weitzman. Optimal search for the best alternative. *Econometrica*, 47(3):641–654, 1979.

[55] Andrew Chi-Chih Yao. An n-to-1 bidder reduction for multi-item auctions and its applications. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, page 92–109, 2015.

Appendix

A Assortment Optimization is NP-hard

We follow the same strategy as Khodabakhsh et al. [36] and reduce from the Integer Partition problem. Integer Partition takes a set of integers $\{c_1, \ldots, c_n\}$ and seeks a subset $S \subseteq [n]$ such that $\sum_{i \in S} c_i = \frac{1}{2} \sum_{i=1}^{n} c_i$. We denote the sum of all integers by $C = \sum_i^n c_i$.

We follow closely the reduction of Khodabakhsh et al. [36], and rather than formally replicating their proof, we instead provide a reduction to assortment optimization which has identical utilities for a given solution to the Khodabakhsh et al. [36] reduction, and then refer the reader to their analysis of the optimal solution of the delegation reduction.

We construct an instance of assortment optimization with $n+1$ items and an outside option as follows:

- For item 0 (the outside option), we set $p_0 = 0$ and $v_0 = 0$
- For items $1, \ldots, n$, we set $p_i = 1$ and $v_i = \begin{cases} 2+\delta, & \text{w.p.} \frac{c_i}{M} - \frac{c_i^2}{2M^4(1-\frac{C}{2M})} \\ 2-\delta, & \text{w.p.} \frac{c_i}{M} \\ 0, & \text{otherwise} \end{cases}$
- For items $n+1$, we set $p_{n+1} = M^2(1-\frac{C}{2M})+1+\delta$ and $v_{n+1} = \begin{cases} M^2(1-\frac{C}{2M})+2+\delta, & \text{w.p.} \frac{1}{2} \\ 0, & \text{w.p.} \frac{1}{2} \end{cases}$

Here, M is a large constant picked to satisfy the constraints imposed in the proof of Theorem 2 of Khodabakhsh et al. [36] ($M \geq 128 n^3 c_{max}^3$ suffices, where $c_{max} = \max_i c_i$). Lastly, δ is a tiny constant used for tie-breaking and thus excluded from the principal utility computation.

We argue that this assortment optimization instance is structurally equivalent to the instance used by Khodabakhsh et al. [36] to prove that the delegated choice problem is NP-hard:

- For actions $1, \ldots, n$, we set $b_i = M^2(1-\frac{C}{2M})$ and $v_i = \begin{cases} 1+2\delta, & \text{w.p.} \frac{c_i}{M} - \frac{c_i^2}{2M^4(1-\frac{C}{2M})} \\ 1, & \text{w.p.} \frac{c_i}{M} \\ 0, & \text{otherwise} \end{cases}$
- For action $n+1$, we set $b_{n+1} = 0$ and $v_{n+1} = \begin{cases} M^2(1-\frac{C}{2M})+1+\delta, & \text{w.p.} \frac{1}{2} \\ 0, & \text{w.p.} \frac{1}{2} \end{cases}$

Consider any subset of actions $S \subseteq \{1, \ldots, n\}$ in the delegation instance. We claim that the distribution of the principal's utility in the delegation instance induced by S is identical to that in the assortment optimization instance induced by item set $S \cup \{0\}$. To see this, first observe that each action $i \in \{1, \ldots, n+1\}$ has the same probability distribution as the corresponding item in the assortment optimization instance. Second, observe that for all realizations, the principal's utility is pairwise identical for both instances: if one of the first n actions/items realizes to a positive value, the principal gets utility 1 in both instances; if the $(n+1)$-th item/action realizes to the positive value, the principal gets utility $M^2(1-\frac{C}{2M})+1+\delta$ in both instances; if all actions realize to 0, then the delegation instance will provide the principal a utility of 0, no matter what action is selected by the agent, while in the assortment optimization instance, the agent will select the outside option which provides the principal a utility of 0 as well.

Then, the NP-hardness proof for delegation (Theorem 2 of Khodabakhsh et al. [36]) can be directly translated to the assortment optimization problem, completing the reduction.