

Randomized Robust Price Optimization

Xinyi Guan

Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University, Hong Kong,
xinyi.guan@polyu.edu.hk

Velibor V. Mišić

UCLA Anderson School of Management, University of California, Los Angeles, California 90095, United States,
velibor.misic@anderson.ucla.edu

The robust multi-product pricing problem is to determine the prices of a collection of products so as to maximize the worst-case revenue, where the worst case is taken over an uncertainty set of demand models that the firm expects could be realized in practice. A tacit assumption in this approach is that the pricing decision is a deterministic decision: the prices of the products are fixed and do not vary. In this paper, we consider a *randomized* approach to robust pricing, where a decision maker specifies a distribution over potential price vectors so as to maximize its worst-case revenue over an uncertainty set of demand models. We formally define this problem – the *randomized robust price optimization* problem – and analyze when a randomized price scheme performs as well as a deterministic scheme versus when it yields a benefit. We also propose solution methods for obtaining an optimal randomization scheme over a discrete set of candidate price vectors and show how these methods are applicable for common demand models, such as the linear, semi-log and log-log demand models. We numerically compare the randomized and deterministic approaches on a variety of synthetic and real problem instances; on instances derived from a real grocery retail scanner dataset, we show that the improvement in worst-case revenue can be as high as 92%. Using the same grocery retail scanner dataset, we also show that the randomized approach can produce price prescriptions that achieve higher out-of-sample revenue than the nominal and deterministic robust approaches.

Key words: pricing, randomization, robust optimization

1. Introduction

Price optimization is a key problem in modern business. The price optimization problem can be stated as follows: we are given a collection of products. We are given a demand model which tells us, for each product, what the expected demand for that product will be as a function of the price of that product as well as the price of the other products. Given this demand model, the price optimization problem is to decide a price vector – i.e., what price to set for each product – so as to maximize the total expected revenue arising from the collection of products.

The primary input to a price optimization approach is a demand model, which maps the price vector to the vector of expected demands of a product. However, in practice, the demand model is never known exactly, and must be estimated from data. This poses a challenge because data is

typically limited, and thus a firm often faces uncertainty as to what the demand model is. This is problematic because a mismatch between the demand model used for price optimization – the nominal demand model – and the demand model that materializes in reality can lead to suboptimal revenues.

As a result, there has been much research in how to address demand model uncertainty in pricing. In the operations research community, a general framework for dealing with uncertainty is robust optimization. The idea of robust optimization is to select an uncertainty set, which is a set of values for the uncertain parameter that we believe could plausibly occur, and to optimize the worst-case value of the objective function, where the worst-case is taken over the uncertainty set. In the price optimization context, one would construct an uncertainty set of potential demand models and determine the prices that maximize the worst-case expected revenue, where the worst-case is the minimum revenue over all of the demand models in the uncertainty set. In applying such a procedure, one can ensure that the performance of the chosen price vector is good under a multitude of demand models, and that one does not experience the deterioration of a price vector optimized from a single nominal demand model.

Typically in robust optimization, one seeks the *single* best decision that optimizes the worst-case value of the objective function; stated differently, one *deterministically* implements a single decision. However, a recent line of research (Delage et al. 2019) has revealed that with regard to the worst-case objective, it is possible to obtain better performance than the traditional deterministic robust optimization approach by *randomizing* over multiple solutions. Specifically, instead of selecting a single decision in some feasible set to optimize the worst-case objective, one selects a *distribution* supported on the feasible set that informs the decision maker how to randomize.

In this work, we propose a methodology for robust price optimization that is based on randomization. In particular, we propose solving a randomized robust price optimization (RRPO) problem, which outputs a probability distribution that specifies the frequency with which the firm should use different price vectors. From a practical perspective, such a randomization scheme has the potential to be implemented in modern retailing as a strategy for mitigating demand uncertainty. In particular, in an ecommerce setting, randomization is already used for A/B testing, which involves randomly assigning some customers to one experimental condition and other customers to a different experimental condition. Thus, it is plausible that the same form of randomization could be used to display different price vectors with certain frequencies. In the brick-and-mortar setting, one can potentially implement randomization by varying prices geographically or temporally. For example, if the RRPO solution is the three price vectors $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ with probabilities 0.2, 0.3, 0.5, then for a set of 50 regions, we would choose 10 ($= 50 \times 0.2$) to assign to price vector \mathbf{p}_1 , 15 ($= 50 \times 0.3$) to assign to \mathbf{p}_2 , and 25 ($= 50 \times 0.5$) to assign to \mathbf{p}_3 . Similarly, if one were to implement

the same randomization scheme temporally, then for a selling horizon of 20 weeks, one would use the price vector \mathbf{p}_1 for 4 ($= 20 \times 0.2$) weeks, \mathbf{p}_2 for 6 ($= 20 \times 0.3$) weeks and \mathbf{p}_3 for 10 ($= 20 \times 0.5$) weeks.

We make the following specific contributions:

1. **Benefits of randomization.** We formally define the RRPO problem and analyze it under different conditions to determine when the underlying robust price optimization problem is *randomization-receptive* – there is a benefit from implementing a randomized decision over a deterministic decision – versus when it is *randomization-proof* – the optimal randomized and deterministic decisions perform equally well. We show that the robust price optimization problem is randomization-proof in several interesting settings, which can be described roughly as follows: (1) when the set of feasible price vectors is convex and the set of uncertain revenue functions is concave; (2) when the set of feasible price vectors and the set of uncertain demand parameters are convex, and the revenue function obeys certain quasiconvexity and quasiconcavity properties with respect to the price vector and the uncertain parameter vector; and (3) when the set of feasible price vectors is finite, and a certain strong duality/saddle point property holds. We showcase a number of examples of special cases that satisfy the hypotheses of these results and consequently are randomization-proof. We also present several examples showing how these results can fail to hold when certain assumptions are relaxed and the problem thus becomes randomization-receptive.
2. **Tractable solution algorithms.** We propose algorithms for solving the RRPO problem in two different settings:
 - (a) In the first setting, we assume that the set of possible price vectors is a finite set and that the uncertainty set of demand function parameters is a convex set. In this setting, when the revenue function is convex in the uncertain parameters, we show that the RRPO problem can be solved via delayed constraint generation. The separation problem that is solved to determine which constraint to add is exactly the nominal pricing problem for a fixed uncertain parameter vector of the demand function. For the log-log and semi-log demand models, we show that this nominal pricing problem can be reformulated and solved to global optimality as a mixed-integer exponential cone program. We believe these reformulations are of independent interest as to the best of our knowledge, these are the first exact mixed-integer convex formulations for these problems in either the marketing or operations literatures, and they leverage recent advances in solution technology for mixed-integer conic programs (as exemplified in the conic solver Mosek). In this setting, we also prove that the complexity of randomization, which is the size of the support of the randomized price vector, is at most $M + 1$, where M is the dimension of the uncertain

parameter vector. Lastly, we also show how our method in this setting can be extended to incorporate moment constraints on the price distribution.

- (b) In the second setting, we assume that both the price set and the uncertainty set are finite sets. In this setting, we show how the RRPO problem can be solved using a double column generation method, which involves iteratively generating new uncertainty realizations and price vectors by solving primal and dual separation problems, respectively. We show how the primal and dual separation problems can be solved exactly for the linear, semi-log and log-log demand models.

3. **Numerical evaluation with synthetic and real data.** We evaluate the effectiveness of randomized pricing on different problem instances generated synthetically and problem instances calibrated with real data. We show that randomized pricing can significantly improve worst-case revenues over deterministic pricing, with the benefit in our real data instances being as high as 92%. Additionally, we show that for instances of realistic size (up to 20 products), our algorithm can solve the RRPO problem in a reasonable amount of time (no more than two minutes on average). We also consider a data-driven experiment to evaluate the out-of-sample prescriptive ability of randomized robust pricing using a retail scanner data set. We show that randomized robust pricing outperforms the nominal pricing approach in the log-log and semi-log demand models, is comparable to deterministic robust pricing in the log-log model, and outperforms deterministic robust pricing in the semi-log model.

The rest of this paper is organized as follows. Section 2 reviews the related literature on pricing, robust optimization and randomized robust optimization. Section 3 formally defines the nominal price optimization problem, the deterministic robust price optimization problem and the randomized robust price optimization problem. Section 4 analyzes the robust price optimization problem and provides conditions under which the price optimization problem is randomization-proof. Section 5 presents our constraint generation approach for solving the RRPO problem when the price set is a finite set and the uncertainty set is a convex set, discusses how this approach can be adapted for different families of demand models and presents our result on the complexity of randomization. Section 6 provides a brief overview of our methodology for solving the RRPO problem when the price set and uncertainty sets are finite sets, with the details provided in Section EC.7 of the companion. Section 7 presents our numerical experiments. Lastly, in Section 8, we conclude and highlight some directions for future research.

2. Literature review

Our paper is closely related to three streams of research: pricing, robust optimization and the use of randomized strategies in optimization. We discuss each of these three streams below.

Pricing optimization and demand models. Optimal pricing has been extensively studied in many fields such as revenue management and marketing research; for a general overview of this research area, we refer readers to Soon (2011) and Gallego and Topaloglu (2019). An important stream of pricing literature is on static pricing, which involves setting a fixed price for a product. The most commonly considered demand models in the static pricing literature are the linear and log-log models. For example, the papers of Zenor (1994) and Bernstein and Federgruen (2003) assume linear demand functions in the study of pricing strategies. The papers of Reibstein and Gatignon (1984), and Montgomery and Bradlow (1999) use log-log (multiplicative) demand functions to represent aggregate demand. The paper of Kalyanam (1996) considers a semi-log demand model. Besides linear, semi-log and log-log demand functions, another type of demand form that is extensively discussed in pricing literature is based on an underlying discrete choice model. For example, Hanson and Martin (1996), Aydin and Ryan (2000), and Hopp and Xu (2005) consider the product line pricing problem under the multinomial logit (MNL) model. Keller et al. (2014) consider attraction demand models which subsume MNL models. Li and Huh (2011) study the pricing problem with the nested logit (NL) models, and show the concavity of the profit function with respect to market share holds. Gallego and Wang (2014) characterize the optimal pricing structure under the general nested logit model with product-differentiated price sensitivities and arbitrary nest coefficients. The papers of Keller (2013) and Zhang et al. (2018) study the multiproduct pricing problem under the family of generalized extreme value (GEV) models which includes MNL and NL models as special cases. Our work differs from this prior work on multiproduct pricing in that the demand model is not assumed to be known, and that there is an uncertainty set of plausible demand models that could actually materialize. Correspondingly, the firm is concerned not with expected revenue under a single, nominal demand model, but with the worst-case revenue with respect to this uncertainty set of demand models.

Robust Optimization. Within the literature mentioned above, the demand models are assumed to be known exactly or are estimated from historical data. In practice, the decision maker often does not have complete knowledge of the demand model. Also, in many real applications, the lack of sales data makes it hard to obtain a good estimation of demand models, which leads to model misspecification and thus suboptimal pricing decisions. In operations research, this type of challenge is most commonly addressed using the framework of robust optimization, where uncertain parameters are assumed to belong to some uncertainty set and one optimizes for the worst-case objective of parameters within the set. We refer readers to Ben-Tal et al. (2009), Bertsimas et al. (2011), Gabrel et al. (2014) and Bertsimas and den Hertog (2022) for a detailed overview of this approach. Robust optimization has been widely applied in various problem settings such as

assortment optimization (Rusmevichientong and Topaloglu 2012, Bertsimas and Mišić 2017, Sturt 2024), inventory management (Bertsimas and Thiele 2006, Govindarajan et al. 2021) and financial option pricing (Bandi and Bertsimas 2014, Sturt 2023).

Within this literature, our paper contributes to the substream that considers robust optimization for pricing. Thiele (2009) consider tractable robust counterparts to the deterministic multiproduct pricing problem with the budget-of-resource-consumption constraint in the case of additive demand uncertainty, and investigate the impact of uncertainty on the optimal prices of multiple products sharing capacitated resources. Mai and Jaillet (2019) consider robust multiproduct pricing optimization under the generalized extreme value (GEV) choice model and characterize the robust optimal solutions for unconstrained and constrained pricing problems. Hamzei et al. (2021) study the robust pricing problem with interval uncertainty of the price sensitivity parameters under the multi-product linear demand model. For robust dynamic pricing problems, Lim and Shanthikumar (2007) and Lim et al. (2008) use relative entropy to represent uncertainty in the demand rate and thus the demand uncertainty can be expressed through a constraint on relative entropy. Perakis and Sood (2006), Adida and Perakis (2010) and Chen and Chen (2018) all study robust dynamic pricing problems with demand uncertainty modeled by intervals. Harsha et al. (2019) study robust dynamic price optimization on an omnichannel network with cross-channel interactions in demand and supply where demand uncertainty is modeled through budget constraints. From a different perspective, Cohen et al. (2018) develop a data-driven framework for solving the robust dynamic pricing problem by directly using samples in the optimization. Specifically, the paper considers three types of robust objective (max-min, min-max regret and max-min ratio), and uses the given sampled scenarios to approximate the uncertainty set by a finite number of constraints. Another paper that considers data-driven pricing is the paper of Yan et al. (2022), which models customer demand using a representative agent model. In a representative agent model, choice probabilities arise as solutions to a constrained concave optimization problem over the unit simplex, where one finds a discrete probability distribution that maximizes the expected utility minus a convex perturbation function. By appropriately setting the perturbation function, one can recover various types of discrete choice models. The paper of Yan et al. (2022) considers a method for identifying the perturbation function nonparametrically using price-demand data, and then subsequently solving a mixed-integer program to obtain the optimal prices using the estimated representative agent model. While our paper shares a high-level similarity with this work in that one seeks to address model misspecification, this prior work does not incorporate uncertainty around the estimated demand model, and does not consider randomness in the pricing decision.

Our work differs from the majority of this body of work that takes a robust optimization approach to pricing in that the decision we seek to make is no longer a deterministic decision, but a randomized one. Within this body of work, the papers closest to our work are Allouah et al. (2021)

and Allouah et al. (2022). The paper of Allouah et al. (2021) considers a pricing problem under a valuation-based model of demand, where each customer has a valuation drawn from an unknown cumulative distribution function on the positive real line, and the firm only has one historical data point. The paper considers the pricing problem from a max-min ratio standpoint, where the firm seeks to find a pricing mechanism that maximizes the percentage attained of the true maximum revenue under the worst-case (minimum) valuation distribution consistent with the data point. The paper shows the approximation rate that is attainable given knowledge of different quantiles of the valuation distribution when the valuation distribution is a regular distribution or a monotone non-decreasing hazard rate distribution. The mechanisms that are proposed in the paper, which are mappings from the data point to a price, include deterministic ones that offer a fixed price, as well as randomized ones that offer different prices probabilistically. The paper of Allouah et al. (2022) considers a similar setting, where instead of knowing a point on the valuation CDF exactly or to within an interval, one has access to an IID sample of valuations drawn from the unknown valuation CDF, and similarly proposes deterministic and randomized mechanisms for this setting.

With regard to Allouah et al. (2021) and Allouah et al. (2022), our setup differs in a number of ways. First, our methodology focuses on a max-min revenue objective, as opposed to a max-min ratio objective that considers performance relative to oracle-optimal revenue. Our methodology also does not start from a valuation model, but instead starts from an aggregate demand model, and additionally considers the multi-product case in the general setting. Additionally, we do not take data as a starting point, but instead assume that the aggregate demand model is uncertain. Lastly, the overarching goals are different: while Allouah et al. (2021, 2022) seek to understand the value of data, and how well one can do with limited data, our goal is to demonstrate that from the perspective of worst-case revenue, a randomized pricing strategy can be preferable over a deterministic fixed price, and to develop tractable computational methods for computing such strategies under commonly used demand models in the multi-product setting.

Randomized strategies in optimization under uncertainty. The conventional robust optimization problems mentioned above only consider deterministic solutions. In recent years, the benefit of using randomized strategies has received increasing attention in the literature on decision making under uncertainty and robust optimization. Mastin et al. (2015) study randomized strategies for min-max regret combinatorial optimization problems in the cases of interval uncertainty and uncertainty representable by discrete scenarios, and provide bounds on the gains from randomization for these two cases. Bertsimas et al. (2016b) consider randomness in a network interdiction min-max problem where the interdictor can benefit from using a randomized strategy to select arcs to be removed.

The paper of Delage et al. (2019) considers the problem of making a decision whose payoff is uncertain and minimizing a risk measure of this payoff, and studies under what circumstances a randomized decision leads to lower risk than a deterministic decision. The paper characterizes the classes of randomization-receptive and randomization-proof risk measures in the absence of distributional ambiguity (i.e., classical stochastic programs), and discusses conditions under which problems with distributional ambiguity can benefit from randomized decisions.

Subsequently, the paper of Delage and Saif (2022) studies the value of randomized solutions for mixed-integer distributionally robust optimization problems. The paper develops bounds on the magnitude of improvement given by randomized solutions over deterministic solutions, and proposes a two-layer column generation method for solving single-stage and two-stage linear DRO problems with randomization. Our paper relates to Delage and Saif (2022) in that we apply a similar two-layer column generation approach for solving the randomized robust pricing problem when the price set and the uncertainty set are both finite; we discuss this connection in more detail in our discussion of the paper of Wang et al. (2024) below. The paper of Sadana and Delage (2023) develops a randomization approach for solving a distributionally robust maximum flow network interdiction problem with a conditional-value-at-risk objective, which is also solved using a column generation approach.

Our work is most closely related to the paper of Wang et al. (2024). The paper of Wang et al. (2024) introduces randomization into the robust assortment optimization and characterizes the conditions under which a randomized strategy strictly improves worst-case expected revenues over a deterministic strategy. The paper proposes several different solution methods for finding an optimal distribution over assortments for the MNL, Markov chain and ranking-based models. Our paper shares a high-level viewpoint with the paper of Wang et al. (2024) in that revenue management decisions, such as assortment decisions and pricing decisions, are subject to uncertainty and from an operational point of view, have the potential to be randomized and to benefit from randomization. From a technical standpoint, several of our results on the benefit of randomization when the price set is discrete that are stated in Section 4.3 are generalizations of results in Wang et al. (2024) to the pricing setting that we study. In terms of methodology, the solution approach we apply when the price set and uncertainty set are discrete in Section 6 (described more fully in Section EC.7) is related to the approach in Wang et al. (2024) for the MNL model, as we also use the two-layer column generation scheme of Delage and Saif (2022). The main difference between the method in Wang et al. (2024) and our method lies in the nature of the subproblems. In the paper of Wang et al. (2024), the primal subproblem is a binary sum of linear fractional functions problem, and the dual subproblem is essentially a mixture of multinomial logits assortment problem that can be reformulated as a mixed-integer linear program. In our paper,

the primal and dual subproblems that are used to generate new price vectors and uncertainty realizations come from the underlying pricing problem and the structure of different demand models (linear, semi-log and log-log), which lead to different subproblems than in the assortment setting. In particular, in the semi-log and log-log cases, both the primal and dual subproblems can be formulated as mixed-integer exponential cone programs. In the semi-log and log-log cases specifically, the formulations of the dual subproblems, which are used to identify new price vectors to add, require a logarithmic transformation together with a biconjugate representation of the log-sum-exp function. This technique is also used to develop a constraint generation scheme for the randomized robust pricing problem when the price set is discrete and the uncertainty set is convex (Section 5); the subproblem in this case involves solving a nominal pricing problem under the semi-log or log-log model, which we are also able to reformulate exactly as a mixed-integer exponential cone program. As noted in the introduction, these are the first exact formulations of these problems using mixed-integer conic programming.

Other work on randomization. Lastly, we comment on several streams of work that use randomization but are unrelated to our paper. Within the revenue management community, there are instances where randomization is an operational aspect of the algorithm. For example, in network revenue management, the heuristic of probabilistic allocation control involves using the primal variable values in the deterministic linear program (DLP) to decide how frequently requests should be accepted or rejected (Jasin and Kumar 2012). Here, randomization is used to ensure that the long run frequency with which different requests are accepted or rejected is as close as possible to the DLP solution, which corresponds to an idealized upper bound on expected revenue. As another example, randomization is often also a part of methods for problems that involve learning. For example, Ferreira et al. (2018) propose a method for network revenue management where there is uncertainty in demand rates based on Thompson sampling, which is a method from the bandit literature that involves taking a random sample from the posterior distribution of an uncertain parameter and taking the action that is optimal with respect to that sample. Here, randomization is a way of ensuring that the decision maker explores possibly suboptimal actions. In our work, the focus is not on using randomization to achieve better expected performance or using randomization to achieve a balance between exploration and exploitation, but rather to operationalize randomization to achieve better worst-case performance.

3. Problem definition

In this section, we begin by defining the nominal price optimization problem in Section 3.1. We subsequently define the deterministic robust price optimization problem in Section 3.2. Lastly, we define the randomized robust price optimization problem in Section 3.3.

3.1. Nominal price optimization problem

We assume that the firm offers I products, indexed from 1 to I . We let p_i denote the price of product $i \in [I]$, where we use the notation $[n] = \{1, \dots, n\}$ for any positive integer n . We use $\mathbf{p} = (p_1, \dots, p_I)$ to denote the vector of prices. We assume that the price vector \mathbf{p} is constrained to lie in the set $\mathcal{P} \subseteq \mathbb{R}_+^I$, where \mathbb{R}_+ is the set of nonnegative real numbers.

We let d_i denote the demand function of product i , so that $d_i(\mathbf{p})$ denotes the demand of product i when the price vector \mathbf{p} is chosen. The revenue function $R(\cdot)$ can then be written as $R(\mathbf{p}) = \sum_{i=1}^I p_i \cdot d_i(\mathbf{p})$.

The nominal price optimization (NPO) problem can be written simply as

$$\text{NPO: } \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}).$$

There are numerous demand models that can be used in practice, which lead to different price optimization problems; we briefly review some of the more popular ones here.

1. *Linear demand model*: A linear demand model is defined by parameters $\boldsymbol{\alpha} \in \mathbb{R}^I$, $\boldsymbol{\beta} \in \mathbb{R}^I$, $\boldsymbol{\gamma} = (\gamma_{i,j})_{i,j \in [I], i \neq j} \in \mathbb{R}^{I \cdot (I-1)}$. The demand function $d_i(\cdot)$ of each product $i \in [I]$ has the form

$$d_i(\mathbf{p}) = \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j, \quad (1)$$

where $\beta_i \geq 0$ is the own-price elasticity parameter of product i , which indicates how much demand for product i is affected by the price of product i , whereas $\gamma_{i,j}$ is a cross-price elasticity parameter that describes how much demand for product i is affected by the price of a different product j . Note that $\gamma_{i,j}$ can be positive, which generally corresponds to products i and j being substitutes (i.e., when the price of product j increases, customers tend to switch to product i), or negative, which corresponds to products i and j being complements (i.e., products i and j tend to be purchased together, so when the price of product j increases, this causes a decrease in demand for product i). The corresponding revenue function $R(\cdot)$ is then

$$R(\mathbf{p}) = \sum_{i=1}^I p_i \cdot (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j). \quad (2)$$

2. *Semi-log demand model*: A semi-log demand model is defined by parameters $\boldsymbol{\alpha} \in \mathbb{R}^I$, $\boldsymbol{\beta} \in \mathbb{R}^I$, $\boldsymbol{\gamma} = (\gamma_{i,j})_{i,j \in [I], i \neq j} \in \mathbb{R}^{I \cdot (I-1)}$. In a semi-log demand model, the logarithm of the demand function $d_i(\cdot)$ of each product $i \in [I]$ has a linear form in the prices p_1, \dots, p_I :

$$\log d_i(\mathbf{p}) = \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j. \quad (3)$$

This implies that the demand function is

$$d_i(\mathbf{p}) = e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}. \quad (4)$$

The corresponding revenue function $R(\cdot)$ is then

$$R(\mathbf{p}) = \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}. \quad (5)$$

3. *Log-log demand model*: A log-log demand model is defined by parameters $\alpha \in \mathbb{R}^I$, $\beta \in \mathbb{R}^I$, $\gamma = (\gamma_{i,j})_{i,j \in [I], i \neq j} \in \mathbb{R}^{I \cdot (I-1)}$. In a log-log demand model, the logarithm of the demand function $d_i(\cdot)$ of each product $i \in [I]$ has a linear form in the log-transformed prices $\log p_1, \dots, \log p_I$:

$$\log d_i(\mathbf{p}) = \alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j. \quad (6)$$

This implies that the demand function for product i is

$$d_i(\mathbf{p}) = e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \quad (7)$$

$$= e^{\alpha_i} p_i^{-\beta_i} \cdot \prod_{j \neq i} p_j^{\gamma_{i,j}}, \quad (8)$$

and that the revenue function is therefore

$$R(\mathbf{p}) = \sum_{i=1}^I p_i \cdot e^{\alpha_i} p_i^{-\beta_i} \cdot \prod_{j \neq i} p_j^{\gamma_{i,j}} \quad (9)$$

$$= \sum_{i=1}^I e^{\alpha_i} \cdot p_i^{1-\beta_i} \cdot \prod_{j \neq i} p_j^{\gamma_{i,j}}. \quad (10)$$

3.2. Deterministic robust price optimization problem

We now define the deterministic robust price optimization (DRPO) problem. To define this problem abstractly, we let \mathcal{R} denote an uncertainty set of possible revenue functions. The DRPO problem is to then maximize the worst-case revenue, where the worst-case is the minimum revenue of a given price vector taken over all revenue functions in \mathcal{R} . Mathematically, this problem can be written as

$$\text{DRPO: } \max_{\mathbf{p} \in \mathcal{P}} \min_{R \in \mathcal{R}} R(\mathbf{p}).$$

We use Z_{DR}^* to denote the optimal objective value of the DRPO problem.

Although \mathcal{R} can be defined in many different ways, we will now focus on one general case that we will assume for most of our subsequent results in Sections 4, 5 and 6. Suppose that we fix the demand model to a specific parametric family, such as a log-log demand model. Let \mathbf{u} denote the vector of demand model parameters. For example, for log-log, \mathbf{u} would be the tuple $\mathbf{u} = (\alpha, \beta, \gamma)$. Let $\mathcal{U} \subseteq \mathbb{R}^M$ denote a set of possible values of \mathbf{u} , where the integer M is the dimension of \mathbf{u} ; \mathcal{U} is then an uncertainty set of model parameters. With a slight abuse of notation, let $d_i(\mathbf{p}, \mathbf{u})$ denote the demand for product i when the demand model parameters are specified by \mathbf{u} . Then \mathcal{R} can be defined as:

$$\mathcal{R} = \{R(\mathbf{p}) \equiv \sum_{i=1}^I p_i \cdot d_i(\mathbf{p}, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}, \quad (11)$$

i.e., it is all of the possible revenue functions spanned by the uncertain parameter vector \mathbf{u} in \mathcal{U} .

To express the DRPO problem in this setting more conveniently, we will abuse our notation slightly and use $R(\mathbf{p}, \mathbf{u})$ to denote the revenue function evaluated at a price vector \mathbf{p} with a particular parameter vector \mathbf{u} specified. With this abuse of notation, the DRPO problem can be written as

$$\text{DRPO: } \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}).$$

3.3. Randomized robust price optimization problem

In the DRPO problem, we assume that the decision maker will deterministically implement a single price vector \mathbf{p} in the face of uncertainty in the revenue function. In the RRPO problem, we instead assume that the decision maker will randomly select a price vector \mathbf{p} according to some distribution F over the feasible price set \mathcal{P} . Under this assumption, we can write the RRPO problem as

$$\text{RRPO: } \max_{F \in \mathcal{F}} \min_{R \in \mathcal{R}} \int_{\mathcal{P}} R(\mathbf{p}) dF(\mathbf{p}),$$

where \mathcal{F} is the set of all distributions supported on \mathcal{P} . We use Z_{RR}^* to denote the optimal objective value of the RRPO problem. Note that $Z_{\text{RR}}^* \geq Z_{\text{DR}}^*$. This is because for every $\mathbf{p}' \in \mathcal{P}$, the distribution $F(\cdot) = \delta_{\mathbf{p}'}(\cdot)$, where $\delta_{\mathbf{p}'}(\cdot)$ is the Dirac delta function at \mathbf{p}' , is contained in \mathcal{F} . For this distribution, $\min_{R \in \mathcal{R}} \int_{\mathcal{P}} R(\mathbf{p}) dF(\mathbf{p}) = \min_{R \in \mathcal{R}} R(\mathbf{p}')$, which is exactly the worst-case revenue of deterministically selecting \mathbf{p}' .

A special instance of this problem arises when \mathcal{P} is a discrete, finite set. In this case, F is a discrete probability distribution, and one can re-write the outer problem as an optimization problem over a discrete probability distribution $\boldsymbol{\pi} = (\pi_{\mathbf{p}})_{\mathbf{p} \in \mathcal{P}}$ over \mathcal{P} :

$$\text{RRPO-D: } \max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}}} \min_{R \in \mathcal{R}} \sum_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}) \pi_{\mathbf{p}},$$

where we use Δ_S to denote the $(|S| - 1)$ -dimensional unit simplex, i.e., $\Delta_S = \{\boldsymbol{\pi} \in \mathbb{R}^S \mid \sum_{i \in S} \pi_i = 1, \pi_i \geq 0 \forall i \in S\}$.

Lastly, under the assumption that \mathcal{R} is the set of revenue functions of a fixed demand model family whose parameter vector \mathbf{u} belongs to a parameter uncertainty set \mathcal{U} , we can restate the RRPO problem when \mathcal{P} is a generic set and when \mathcal{P} is finite as

$$\text{RRPO: } \max_{F \in \mathcal{F}} \min_{\mathbf{u} \in \mathcal{U}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}), \quad (12)$$

$$\text{RRPO-D: } \max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) \pi_{\mathbf{p}}. \quad (13)$$

4. Benefits of randomization

In this section, we analyze when randomization can be beneficial. To aid us, we introduce some additional nomenclature in this section, which follows the terminology established in the prior literature on randomized robust optimization (Delage et al. 2019, Delage and Saif 2022, Wang et al. 2024). We say that a robust price optimization (RPO) problem is *randomization-receptive* if $Z_{\text{RR}}^* > Z_{\text{DR}}^*$, that is, randomizing over price vectors leads to a higher worst-case revenue than deterministically selecting a single price vector. Otherwise, we say that a RPO problem is *randomization-proof* if $Z_{\text{RR}}^* = Z_{\text{DR}}^*$, that is, there is no benefit from randomizing over price vectors.

In the following three sections, we derive three classes of results that establish when the RPO problem is randomization-proof. The first condition (Section 4.1) for randomization-proofness applies in the case when \mathcal{P} is a convex set and \mathcal{R} is an arbitrary set of revenue functions each of which is concave in \mathbf{p} . The second and third conditions apply to the case where \mathcal{R} arises out of a single demand model, where the parameter vector \mathbf{u} belongs to an uncertainty set. The second condition (Section 4.2) applies in the case when \mathcal{P} and \mathcal{U} are compact, convex sets and $R(\mathbf{p}, \mathbf{u})$ obeys certain quasiconvexity and quasiconcavity properties. The third condition (Section 4.3) is for the case when \mathcal{P} is finite and involves a certain minimax condition being met; as corollaries, we show that randomization-proofness occurs if the DRPO problem satisfies a strong duality property, and that randomization-proofness occurs if the DRPO solution solves the nominal price optimization problem solution at the worst-case \mathbf{u}^* . Along the way, we also give examples where our results establish that certain families of RPO problems are randomization-proof, and also highlight how the results fail to hold when certain hypotheses are relaxed.

The main takeaway from these results is that the set of RPO problems that are randomization-proof is small. As we will see, the conditions under which a RPO problem will be randomization-proof are delicate and quite restrictive, and are satisfied only for certain very special cases; in most other realistic cases, the RPO problem will be randomization-receptive. Consequently, in Sections 5 and 6, we will develop algorithms for solving the RRPO problem when the candidate price vector \mathcal{P} is finite, and in Sections 7 we will show a wide range of both synthetic and real data instances in which the RPO problem is randomization-receptive.

Before we move on to our results, we comment on the connection between randomization-proofness and results from game theory. In particular, the robust pricing problem that we study can be viewed as a two-player zero-sum game: the first player is the firm, who chooses an action \mathbf{p} from the strategy space \mathcal{P} , and the second player is nature, who chooses an action \mathbf{u} from the strategy space \mathcal{U} . The payoff function that the first player maximizes is $R(\mathbf{p}, \mathbf{u})$, while the payoff function that the second player maximizes is $-R(\mathbf{p}, \mathbf{u})$, which is equivalent to minimizing $R(\mathbf{p}, \mathbf{u})$. In game theory, it is common to analyze when a game has a *pure strategy Nash equilibrium*. A pure

strategy Nash equilibrium in our setting is a pair of actions $(\mathbf{p}^*, \mathbf{u}^*)$ such that each player cannot improve their payoff by unilaterally deviating from their action; mathematically, it satisfies

$$R(\mathbf{p}, \mathbf{u}^*) \leq R(\mathbf{p}^*, \mathbf{u}^*), \quad \forall \mathbf{p} \in \mathcal{P}, \quad (14)$$

$$R(\mathbf{p}^*, \mathbf{u}) \leq R(\mathbf{p}^*, \mathbf{u}^*), \quad \forall \mathbf{u} \in \mathcal{U}. \quad (15)$$

Note that a pure strategy Nash equilibrium $(\mathbf{p}^*, \mathbf{u}^*)$ is exactly a saddle point of the function R . By standard optimization arguments, the existence of a saddle point is equivalent to strong duality holding, that is,

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) = \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}). \quad (16)$$

When strong duality holds, the RPO problem is randomization-proof, because

$$Z_{\text{RR}}^* = \max_{F \in \mathcal{F}} \min_{\mathbf{u} \in \mathcal{U}} \int R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \leq \min_{\mathbf{u} \in \mathcal{U}} \max_{F \in \mathcal{F}} \int R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) = \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) = \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) = Z_{\text{DR}}^*, \quad (17)$$

i.e., $Z_{\text{RR}}^* \leq Z_{\text{DR}}^*$, and together with $Z_{\text{RR}}^* \geq Z_{\text{DR}}^*$, this immediately implies that $Z_{\text{RR}}^* = Z_{\text{DR}}^*$. Thus, when one views the RPO problem as a two-player zero-sum game, if a pure strategy Nash equilibrium exists for this game, then the RPO problem is randomization-proof. We leverage this connection to establish one of our results, Theorem 2, where we invoke the well-known Debreu-Glicksberg-Fan theorem (see Theorem 1.2 of Fudenberg and Tirole 1991) to assert the existence of a pure strategy Nash equilibrium. However, the two conditions are not equivalent: it can be the case that a pure strategy Nash equilibrium does not exist, and yet the RPO problem is randomization-proof. In Section 4.1, we note an instance where the RPO problem is randomization-proof but a pure strategy Nash equilibrium does not exist, and in this instance the price set \mathcal{P} is a continuous interval. In Section 4.3, we note a similar instance where the price set \mathcal{P} is finite.

4.1. Concave revenue function uncertainty sets

Our first major result is for the case where \mathcal{R} consists of concave revenue functions.

THEOREM 1. *Suppose that \mathcal{P} is a convex set and that \mathcal{R} is such that every $R \in \mathcal{R}$ is a concave function of \mathbf{p} . Then the RPO problem is randomization-proof, that is, $Z_{\text{RR}}^* = Z_{\text{DR}}^*$.*

The proof of this result (see Section EC.1.1 of the ecompanion) follows from a simple application of Jensen's inequality. We pause to make a few important comments about this result. First, one aspect of this result that is special is that \mathcal{R} can be a very general set: it could be countable or uncountable, and it could consist of revenue functions corresponding to different families of demand models. This will not be the case for our later results in Section 4.2 and 4.3, which require that \mathcal{R} is defined based on a single demand model family.

Second, we remark that the conditions that all functions in \mathcal{R} be concave and that \mathcal{P} be a convex set cannot be relaxed in general. In Section EC.2.1 of the ecompanion, we provide examples to illustrate how Theorem 1 can fail to hold if \mathcal{R} contains a non-concave function, or if \mathcal{P} is a non-convex set.

Third, we remark on the connection between Theorem 1 and results in game theory. As alluded to at the end of Section 4, Theorem 1 is *not* a consequence of any known result on the existence of pure strategy Nash equilibria in game theory. The reason for this is that in general, strong duality does not hold in this setup, i.e., it is *not* the case that

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{R \in \mathcal{R}} R(\mathbf{p}) = \min_{R \in \mathcal{R}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}). \quad (18)$$

In Section EC.2.1 of the ecompanion we provide an example (Example EC.3) of an RPO problem that is randomization-proof, but for which strong duality does not hold (and hence, the corresponding game does not have a pure strategy Nash equilibrium).

To provide a different perspective, as we previously noted, the RPO problem can be viewed as a game between the firm and nature. In Theorem 1, the set of revenue functions is not indexed by any uncertain parameter \mathbf{u} , but one can artificially identify each function $R \in \mathcal{R}$ with some \mathbf{u} , i.e., with some abuse of notation $R(\mathbf{p}) = R(\mathbf{p}, \mathbf{u})$ for some \mathbf{u} . If one takes this viewpoint, then the set \mathcal{U} of indexing \mathbf{u} 's can be arbitrary, and need not be convex or compact. Additionally, for a fixed \mathbf{p} , the function $R(\mathbf{p}, \mathbf{u})$ could be arbitrary in \mathbf{u} and need not be convex/quasi-convex or even continuous. We are not aware of any general result asserting that a pure strategy Nash equilibrium exists in this setting, and in light of the example presented in Section EC.2.1, such a result cannot exist.

Lastly, Theorem 1 has a number of implications for different classes of demand models. In Section EC.2.2, we document several applications of Theorem 1. In particular, we present conditions under which single-product pricing under linear demand, multi-product pricing under linear demand, single-product pricing under semi-log demand and single-product pricing under log-log demand are randomization-proof.

4.2. Quasiconcavity in \mathbf{p} and quasiconvexity in \mathbf{u}

The second result we establish concerns the RRPO problem when there is a demand parameter uncertainty set \mathcal{U} . In this case, the RRPO and DRPO problems are

$$\begin{aligned} \text{RRPO} : & \max_{F \in \mathcal{F}} \min_{\mathbf{u} \in \mathcal{U}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}), \\ \text{DRPO} : & \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}). \end{aligned}$$

We then have the following result.

THEOREM 2. *Suppose that $\mathcal{P} \subseteq \mathbb{R}^I$ and $\mathcal{U} \subseteq \mathbb{R}^M$ are compact convex sets. Suppose that $R(\mathbf{p}, \mathbf{u})$ is quasiconcave in \mathbf{p} on \mathcal{P} for any $\mathbf{u} \in \mathcal{U}$, quasiconvex in \mathbf{u} on \mathcal{U} for any $\mathbf{p} \in \mathcal{P}$ and continuous in (\mathbf{p}, \mathbf{u}) . Then, the robust price optimization problem is randomization-proof, that is, $Z_{\text{DR}}^* = Z_{\text{RR}}^*$.*

The proof of Theorem 2 (see Section EC.1.2 of the ecompanion) follows from viewing the robust price optimization problem as a two-player zero-sum game, and applying the Debreu-Glicksberg-Fan theorem (see Fudenberg and Tirole 1991) to assert the existence of a pure-strategy Nash equilibrium. As we noted in Section 4, such a Nash equilibrium is a saddle point of the function R and automatically implies that strong duality holds, which establishes that the problem is randomization-proof.

This result allows us to show that a larger number of RPO problems are randomization-proof. In Section EC.2.3 of the ecompanion, we show that single-product pricing under semi-log and log-log demand is randomization-proof even in settings where the revenue function is not concave in \mathbf{p} for some values of \mathbf{u} and one cannot invoke Theorem 1. With regard to these two examples, we note that they critically rely on the revenue function being log-concave and therefore quasiconcave, which is only the case for single product price optimization problems. Log-concavity and quasiconcavity are in general not preserved under addition (i.e., the sum of quasiconcave functions is not always quasiconcave, and the sum of log-concave functions is not always log-concave), and so Theorem 2 will in general not be applicable for multiproduct pricing problems involving the semi-log or log-log demand model.

4.3. Finite price set \mathcal{P}

In this section, we analyze randomization-proofness when \mathcal{P} is a finite set. To study this setting, let us define the set \mathcal{Q} as the set of all probability distributions supported on \mathcal{U} . We note that these results are adaptations of several results from Wang et al. (2024) to the pricing setting that we study, which develop analogous conditions for randomization-proofness for the robust assortment optimization problem.

Our first result establishes that randomization-proofness is equivalent to the existence of a distribution Q over \mathcal{U} under which any price vector's expected performance is no better than the deterministic robust optimal value.

THEOREM 3. *A robust price optimization problem with finite \mathcal{P} is randomization-proof if and only if there exists a distribution $Q \in \mathcal{Q}$ such that for all $\mathbf{p} \in \mathcal{P}$,*

$$\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \leq Z_{\text{DR}}^*. \quad (19)$$

To prove this result, we use Sion's minimax theorem to establish that

$$Z_{\text{RR}}^* = \inf_{Q \in \mathcal{Q}} \max_{\mathbf{p} \in \mathcal{P}} \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}); \quad (20)$$

with that result in hand, the condition in Theorem 3 is equivalent to establishing that

$$Z_{\text{DR}}^* \geq \inf_{Q \in \mathcal{Q}} \max_{\mathbf{p} \in \mathcal{P}} \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) = Z_{\text{RR}}^*,$$

which, together with the inequality $Z_{\text{DR}}^* \leq Z_{\text{RR}}^*$ immediately yields randomization-proofness. We note that this result is analogous to Theorem 1 in Wang et al. (2024), which provides a similar necessary and sufficient condition for randomization-proofness in the context of robust assortment optimization. Our proof, which relies on Sion's minimax theorem, is perhaps slightly more direct than the proof of Theorem 1 in Wang et al. (2024), although this is a matter of taste.

Our next two results are consequences of this theorem. The first states that a RPO problem will be randomization-proof if it obeys strong duality. The second states that, a robust price optimization problem is randomization-proof if the deterministic robust price vector \mathbf{p}_{DR}^* is an optimal solution of the nominal price optimization problem under a worst-case \mathbf{u}^* that attains the worst-case objective under \mathbf{p}_{DR}^* , and also presents a partial converse if one additionally assumes that \mathbf{u}^* is unique. We note that these results are both analogous to Corollaries 1 and 2 in Wang et al. (2024).

COROLLARY 1. *Suppose that \mathcal{P} is finite. A robust price optimization problem with finite \mathcal{P} is randomization-proof if it satisfies strong duality:*

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) = \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}). \quad (21)$$

COROLLARY 2. *Suppose that \mathcal{P} is finite. Suppose that $\mathbf{p}_{\text{DR}}^* \in \arg \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u})$ is an optimal solution of the deterministic robust price optimization problem, and suppose that $\mathbf{u}^* \in \arg \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$ is an optimal solution of the worst-case problem for \mathbf{p}_{DR}^* .*

- a) *If $\mathbf{p}_{\text{DR}}^* \in \arg \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}^*)$, then the robust price optimization problem is randomization-proof.*
- b) *If the robust price optimization problem is randomization-proof and \mathbf{u}^* is the unique solution of $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$, then $\mathbf{p}_{\text{DR}}^* \in \arg \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}^*)$.*

Corollary 1 follows because the left hand side in (21) is exactly Z_{DR}^* , while the right hand side is an upper bound on Z_{RR}^* . Part a) of Corollary 2 follows because $(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*)$ is a saddle point, which implies that strong duality holds. Part b) of Corollary 2 follows by using Theorem 3 to assert the existence of a distribution Q over \mathcal{U} satisfying (19), using the uniqueness of \mathbf{u}^* to argue that this distribution must place unit mass on \mathbf{u}^* , and simplifying (19) using this fact to show that \mathbf{p}_{DR}^* is the required optimal solution.

With regard to part (b) of Corollary 2, we note that the uniqueness requirement for \mathbf{u}^* cannot in general be relaxed. In Section EC.2.4 of the ecompanion, we show an instance where $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$ has multiple optimal solutions and the problem is randomization-proof, but \mathbf{p}_{DR}^* does not solve $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}')$ for every \mathbf{u}' that solves $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$. Additionally, this instance also illustrates that even when \mathcal{P} is finite, it is possible that the robust price optimization problem may be randomization-proof without strong duality holding.

An interesting follow-on question to these results is under what conditions are strong duality and randomization-proofness equivalent when \mathcal{P} is finite. It turns out that this is the case when $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} and \mathcal{U} is convex. We state and prove this result in Section EC.1.6 of the ecompanion. Additionally, we demonstrate through an example in Section EC.2.5 that the deterministic robust objective Z_{DR}^* , the randomized robust objective Z_{RR}^* and the dual objective $\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ can in general be distinct for the finite \mathcal{P} setting.

We conclude this section with the remark that the sufficient conditions for randomization-proofness in Corollaries 1 and 2 are rather stringent and demanding. With regard to Corollary 1, strong duality is in general unlikely to hold given that \mathcal{P} is a finite set. With regard to Corollary 2, we note that in general, the solution of the deterministic robust price optimization problem is unlikely to also be an optimal solution of an appropriately defined non-trivial nominal price optimization problem; this is frequently not the case in many applications of robust optimization outside of pricing.¹ Given this, these conditions are suggestive of the fact that most robust price optimization problems will be randomization-receptive. This motivates our study of solution algorithms for numerically solving the RRPO problem in the next two sections.

5. Solution algorithm for finite price set \mathcal{P} , convex uncertainty set \mathcal{U}

In this section, we describe a general solution algorithm for solving the RRPO problem when the price set \mathcal{P} is a finite set, and the uncertainty set \mathcal{U} is a general convex uncertainty set. Section 5.1 describes the general solution algorithm, which is a constraint generation algorithm that involves solving a nominal pricing problem over \mathcal{P} as a subroutine. Sections 5.2, 5.3 and 5.4 describe how the solution algorithm specializes to the cases of the linear, semi-log and log-log demand models, respectively, and in particular, how the nominal pricing problem can be solved for each of these three cases; the formulations we present for the semi-log and log-log models here

¹ For example, the nominal problem $\max\{\mathbf{c}^T \mathbf{x} \mid \mathbf{1}^T \mathbf{x} = 1, \mathbf{x} \geq \mathbf{0}\}$ usually has solutions of the form $\mathbf{x}^* = \mathbf{e}_i$, where \mathbf{e}_i is the standard unit vector in \mathbb{R}^n , but the robust problem $\max\{\min_{\mathbf{c} \in \mathcal{U}} \mathbf{c}^T \mathbf{x} \mid \mathbf{1}^T \mathbf{x} = 1, \mathbf{x} \geq \mathbf{0}\}$ could in general have solutions that are in the interior of the unit simplex, which would never be optimal for the nominal problem except for the trivial case where all c_i 's are the same. As another example, Section 4.5 of Bertsimas and Mišić (2017) describes an experiment involving a robust assortment optimization problem with a finite uncertainty set of three choice models, where the robust problem and the nominal problems under each of the choice models are solved heuristically. The robust solution is not an optimal solution for the nominal problem under any of the three choice models (in fact, the robust solution in this case turns out to be a “blend” of the three nominal product lines).

may be of independent interest as they are, to the best of our knowledge, the first exact mixed-integer convex formulations for the multi-product pricing problem under a finite price set for these demand models. In Section 5.5, we develop a theoretical result that characterizes the support of the optimal price distribution in the finite \mathcal{P} , convex \mathcal{U} setting. In Section 5.6, we briefly describe how to extend the methodology in Section 5.1 to the case where the price distribution satisfies additional constraints, with further details provided in the ecompanion (Section EC.5). Lastly, in Section 5.7, we discuss our focus on a finite set \mathcal{P} versus a convex price set \mathcal{P} .

5.1. General solution approach

The first general solution scheme that we consider is when \mathcal{P} is a discrete set and the uncertainty set \mathcal{U} is a convex uncertainty set. In this case, if the revenue function $R(\mathbf{p}, \mathbf{u})$ is convex in $\mathbf{u} \in \mathcal{U}$, then the RRPO problem can be reformulated as follows:

$$\begin{aligned} & \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \\ &= \min_{\mathbf{u} \in \mathcal{U}} \max_{\pi \in \Delta_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \\ &= \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}), \end{aligned}$$

where the first equality follows by Sion's minimax theorem, and the second equality follows by the fact that the inner maximum is attained by setting $\pi_{\mathbf{p}} = 1$ for some \mathbf{p} and setting $\pi_{\mathbf{p}'} = 0$ for all $\mathbf{p}' \neq \mathbf{p}$. This last problem can be written in epigraph form as

$$\underset{t, \mathbf{u}}{\text{minimize}} \quad t \tag{22a}$$

$$\text{subject to} \quad t \geq R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}, \tag{22b}$$

$$\mathbf{u} \in \mathcal{U}. \tag{22c}$$

Problem (22) can be solved using constraint generation. In such a scheme, we start with constraint (22b) enforced only at a subset $\hat{\mathcal{P}} \subset \mathcal{P}$, and solve problem (22) to obtain a solution (t, \mathbf{u}) . At this solution, we solve the problem $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$, and compare this objective value to the current value of t . If it is less than or equal to t , we conclude that (t, \mathbf{u}) satisfies constraint (22b) and terminate with (t, \mathbf{u}) as the optimal solution. Otherwise, if it is greater than t , we have identified a \mathbf{p} for which constraint (22b) is violated and we add the new constraint to $\hat{\mathcal{P}}$. We then re-solve problem (22) with the new $\hat{\mathcal{P}}$ to obtain a new solution (t, \mathbf{u}) and repeat the process until we can no longer identify any violated constraints. To recover the optimal randomization scheme from the solution of this problem (i.e., the distribution π), we simply consider the optimal dual variable of each constraint $t \geq R(\mathbf{p}, \mathbf{u})$.

The viability of this solution approach critically depends on our ability to solve the separation problem $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ efficiently, and to solve the problem (22) efficiently for a fixed subset $\hat{\mathcal{P}} \subset \mathcal{P}$. In what follows, we shall demonstrate that this problem can actually be solved practically for the linear, semi-log and log-log problems.

To develop our approaches for the linear, semi-log and log-log models, we will make the following assumption about the price set \mathcal{P} , which simply states that \mathcal{P} is a Cartesian product of finite sets of prices for each of the products.

ASSUMPTION 1. $\mathcal{P} = \mathcal{P}_1 \times \cdots \times \mathcal{P}_I$, where \mathcal{P}_i is a finite subset of \mathbb{R}_+ for each i .

5.2. Linear demand model

We begin by showing how our solution approach for convex \mathcal{U} applies to the linear demand model case. Recall that the linear model revenue function is

$$R(\mathbf{p}, \mathbf{u}) = \sum_{i=1}^I p_i (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j). \quad (23)$$

For a fixed \mathbf{p} , the function $R(\mathbf{p}, \mathbf{u})$ is linear and therefore convex in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Thus, given a subset $\hat{\mathcal{P}} \subset \mathcal{P}$, the problem (22) should be easy to solve, assuming that \mathcal{U} is also a sufficiently tractable convex set. For example, if \mathcal{U} is a polyhedron, then since each constraint (22b) is linear in \mathbf{u} , problem (22) would be a linear program.

The separation problem for the linear demand model case is

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \sum_{i=1}^I p_i (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \sum_{i=1}^I p_i \alpha_i - \sum_{i=1}^I \beta_i p_i^2 + \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} p_i p_j \end{aligned}$$

Since $\mathcal{P} = \mathcal{P}_1 \times \cdots \times \mathcal{P}_I$, we can formulate this as a mixed-integer program. Let $x_{i,t}$ be a binary variable that is 1 if product i has price $t \in \mathcal{P}_i$, and 0 otherwise. Similarly, let y_{i,j,t_1,t_2} be a binary decision variable that is 1 if product i is given price t_1 and product j is given price t_2 for $i \neq j$, and 0 otherwise. Then the separation problem can be straightforwardly written as

$$\begin{aligned} & \underset{\mathbf{x}, \mathbf{y}}{\text{maximize}} \quad \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \alpha_i \cdot t \cdot x_{i,t} - \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} t^2 \cdot \beta_i \cdot x_{i,t} + \sum_{i=1}^I \sum_{j \neq i} \sum_{t_1 \in \mathcal{P}_i} \sum_{t_2 \in \mathcal{P}_j} \gamma_{i,j} \cdot t_1 \cdot t_2 \cdot y_{i,j,t_1,t_2} \quad (24a) \end{aligned}$$

$$\text{subject to} \quad \sum_{t \in \mathcal{P}_i} x_{i,t} = 1, \quad \forall i \in [I], \quad (24b)$$

$$\sum_{t_2 \in \mathcal{P}_j} y_{i,j,t_1,t_2} = x_{i,t_1}, \quad \forall i, j \in [I], j \neq i, t_1 \in \mathcal{P}_i, \quad (24c)$$

$$\sum_{t_1 \in \mathcal{P}_i} y_{i,j,t_1,t_2} = x_{j,t_2}, \quad \forall i, j \in [I], j \neq i, t_2 \in \mathcal{P}_j, \quad (24d)$$

$$x_{i,t} \in \{0, 1\}, \quad \forall i \in [I], t \in \mathcal{P}_i, \quad (24e)$$

$$y_{i,j,t_1,t_2} \in \{0, 1\}, \quad \forall i, j \in [I], i \neq j, t_1 \in \mathcal{P}_i, t_2 \in \mathcal{P}_j, \quad (24f)$$

where the first constraint simply enforces that exactly one price is chosen for each product, while the second and third constraints require that the y_{i,j,t_1,t_2} variables are equal to $x_{i,t_1} \cdot x_{j,t_2}$. In terms of the number of variables and constraints, if we assume that $|\mathcal{P}_1| = \dots = |\mathcal{P}_I| = P$ (all price sets are of the same size), then the number of constraints is $I + 2I(I-1) \cdot P$ and the number of variables is $I \cdot P + I(I-1) \cdot P^2$.

5.3. Semi-log demand model

We will now show how the solution approach we have defined earlier applies to the semi-log demand model. Recall that the semi-log revenue function is

$$R(\mathbf{p}, \mathbf{u}) = \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}. \quad (25)$$

Observe that for a fixed \mathbf{p} , the function $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} , since it is the nonnegative weighted combination of exponentials of linear functions of $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Thus, given a subset $\hat{\mathcal{P}} \subset \mathcal{P}$, solving problem (22) should again be “easy”, assuming also that \mathcal{U} is a sufficiently tractable convex set. (In particular, the function $R(\mathbf{p}, \mathbf{u})$ can be represented using I exponential cones; assuming that \mathcal{U} is also representable using conic constraints, problem (22) will thus be some type of continuous conic program.)

We now turn our attention to the separation problem, $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$. Specifically, this problem is

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}.$$

Observe that since the function $f(t) = \log(t)$ is monotonic, the set of optimal solutions remains unchanged if we consider the same problem with a log-transformed objective

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{i=1}^I e^{\alpha_i + \log p_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right). \end{aligned} \quad (26)$$

To now further re-formulate this problem, we observe that the objective function can be re-written using the function $g(\mathbf{y}) = \log(\sum_{i=1}^I e^{y_i})$. The function g is what is known as the *log-sum-exp* function, which is a convex function (Boyd and Vandenberghe 2004). More importantly, a standard result in convex analysis is that any proper, lower semi-continuous, convex function is equivalent to its *biconjugate* function, which is the convex conjugate of its convex conjugate (Rockafellar 1970). For the log-sum-exp function, this in particular means that $g(\mathbf{y})$ can be written as

$$g(\mathbf{y}) = \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \{ \boldsymbol{\mu}^T \mathbf{y} - \sum_{i=1}^I \mu_i \log \mu_i \}.$$

The function $h(x) = x \log x$ is the negative entropy function (Boyd and Vandenberghe 2004), and is a convex function; thus, the function inside the $\max\{\cdot\}$ is a linear function minus a sum of convex functions, and is a concave function.

For our problem, this means that (26) can be re-written as

$$\begin{aligned}
& \max_{\mathbf{p} \in \mathcal{P}} \log R(\mathbf{p}, \mathbf{u}) \\
&= \max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{i=1}^I e^{\alpha_i + \log p_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right) \\
&= \max_{\mathbf{p} \in \mathcal{P}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i (\alpha_i + \log p_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\
&= \max_{\mathbf{p} \in \mathcal{P}, \boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i (\alpha_i + \log p_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\}. \tag{27}
\end{aligned}$$

To further reformulate this problem, we now make use of Assumption 1, which states that \mathcal{P} is the Cartesian product of finite sets. Let us introduce a new binary decision variable $x_{i,t}$ which is 1 if product i 's price is set to price $t \in \mathcal{P}_i$, and 0 otherwise. Using this new decision variable, observe that we can replace p_i wherever it occurs with $\sum_{t \in \mathcal{P}_i} t \cdot x_{i,t}$. We can also similarly replace $\log p_i$ with $\sum_{t \in \mathcal{P}_i} \log t \cdot x_{i,t}$. Therefore, problem (27) can be further reformulated as

$$\begin{aligned}
& \underset{\mathbf{x}, \boldsymbol{\mu}}{\text{maximize}} \quad \sum_{i=1}^I \mu_i \left(\alpha_i + \sum_{t \in \mathcal{P}_i} \log t \cdot x_{i,t} - \beta_i \cdot \sum_{t \in \mathcal{P}_i} t \cdot x_{i,t} + \sum_{j \neq i} \gamma_{i,j} \sum_{t \in \mathcal{P}_j} t \cdot x_{j,t} \right) - \sum_{i=1}^I \mu_i \log \mu_i \tag{28a}
\end{aligned}$$

$$\begin{aligned}
& \text{subject to} \quad \sum_{i=1}^I \mu_i = 1, \tag{28b}
\end{aligned}$$

$$\begin{aligned}
& \sum_{t \in \mathcal{P}_i} x_{i,t} = 1, \quad \forall i \in [I], \tag{28c}
\end{aligned}$$

$$\begin{aligned}
& x_{i,t} \in \{0, 1\}, \quad \forall i \in [I], t \in \mathcal{P}_i, \tag{28d}
\end{aligned}$$

$$\begin{aligned}
& \mu_i \geq 0, \quad \forall i \in [I]. \tag{28e}
\end{aligned}$$

This last problem is *almost* a mixed-integer convex program: as noted earlier, the expression $-\sum_{i=1}^I \mu_i \log \mu_i$ is concave in $\boldsymbol{\mu}$. The main wrinkle is the presence of the bilinear terms in the objective function, specifically terms of the form $\mu_i \cdot x_{j,t}$. Fortunately, we can circumvent this difficulty by introducing a new decision variable, $w_{i,j,t}$, which is the linearization of $\mu_i \cdot x_{j,t}$, for each $i, j \in [I], t \in \mathcal{P}_j$. By adding this new decision variable and additional constraints, we arrive at our final formulation, which is a mixed-integer convex program.

$$\begin{aligned}
& \underset{\boldsymbol{\mu}, \mathbf{w}, \mathbf{x}}{\text{maximize}} \quad \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \sum_{i=1}^I \beta_i \cdot \sum_{t \in \mathcal{P}_i} t \cdot w_{i,i,t} + \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \cdot \left(\sum_{t \in \mathcal{P}_j} t \cdot w_{i,j,t} \right) - \sum_{i=1}^I \mu_i \log \mu_i \tag{29a}
\end{aligned}$$

$$\text{subject to } \sum_{t \in \mathcal{P}_j} w_{i,j,t} = \mu_i, \quad \forall i \in [I], j \in [I], \quad (29b)$$

$$\sum_{i=1}^I w_{i,j,t} = x_{j,t}, \quad \forall j \in [I], t \in \mathcal{P}_j, \quad (29c)$$

$$\sum_{i=1}^I \mu_i = 1, \quad (29d)$$

$$\sum_{t \in \mathcal{P}_i} x_{i,t} = 1, \quad \forall i \in [I], \quad (29e)$$

$$w_{i,j,t} \geq 0, \quad \forall i \in [I], j \in [I], t \in \mathcal{P}_j, \quad (29f)$$

$$x_{i,t} \in \{0, 1\}, \quad \forall i \in [I], t \in \mathcal{P}_i, \quad (29g)$$

$$\mu_i \geq 0, \quad \forall i \in [I]. \quad (29h)$$

There are a few important points to observe about this formulation. First, note that because the μ_i 's sum to 1 over i , and the $x_{j,t}$'s are binary and sum to 1 over $t \in \mathcal{P}_j$ for any j , then ensuring that $w_{i,j,t} = \mu_i \cdot x_{j,t}$ can be done simply through constraints (29b) and (29c). This is different from the usual McCormick envelope-style linearization technique, which in this case would involve the four inequalities:

$$w_{i,j,t} \leq x_{j,t}, \quad (30)$$

$$w_{i,j,t} \leq \mu_i, \quad (31)$$

$$w_{i,j,t} \geq x_{j,t} + \mu_i - 1, \quad (32)$$

$$w_{i,j,t} \geq 0, \quad (33)$$

for every $i \in [I]$, $j \in [I]$, $t \in \mathcal{P}_j$. It is not difficult to show that these constraints are implied by constraints (29b), (29c) and (29f).

Second, at the risk of belaboring the obvious, the optimal objective value of problem (29) is the value of $\max_{\mathbf{p} \in \mathcal{P}} \log R(\mathbf{p}, \mathbf{u})$, where R is the semi-log revenue function. Upon solving problem (29) to obtain the objective value Z' , we can obtain the optimal objective value of the untransformed problem $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ as $e^{Z'}$.

Third, this formulation is notable because, to our knowledge, this is the first exact mixed-integer convex formulation of the nominal multi-product pricing problem under semi-log demand and a price set defined as the Cartesian product of finite sets. To date, virtually all research that has considered solving this type of problem in the marketing and operations management literatures has involved heuristics (see, for example, Section EC.3 of Mišić 2020, which solves log-log and semi-log multi-product pricing problems for a collection of stores using local search). From this perspective, although we developed this formulation as part of the overall solution approach for the RRPO problem, we believe it is of more general interest.

Building on the previous point, problem (29) can be formulated as a mixed-integer exponential cone program. We state the exact formulation in Section EC.3 of the ecompanion; assuming that $|\mathcal{P}_1| = \dots = |\mathcal{P}_I| = P$, this formulation has $I^2 \cdot P + I \cdot P + 2I$ variables, $I^2 + I \cdot P + I + 1$ linear constraints and I exponential cone constraints. Such problems are garnering increasing attention from the academic and industry sides. In particular, since 2019, the MOSEK solver (Mosek ApS 2022) supports the exponential cone and can solve mixed-integer conic programs that involve the exponential cone to global optimality. Although the solution technology for mixed-integer conic programs is not as developed as that of mixed-integer linear programs (as exemplified by state-of-the-art solvers such as Gurobi and CPLEX), it is reasonable to expect that these solvers will continue to improve and allow larger and larger problem instances to be solved to optimality in the future.

Lastly, we comment that the same reformulation technique used above – taking the logarithm, replacing the log-sum-exp function with its biconjugate, and then linearizing the products of the binary decision variables and the probability mass function values (the μ_i variables) that arise from the biconjugate – can also be used to derive an exact formulation of the deterministic robust price optimization problem. By taking the same approach, one obtains a max-min-max problem, and one can use Sion’s minimax theorem again to swap the inner maximization over $\boldsymbol{\mu}$ with the minimization over \mathbf{u} to obtain a robust counterpart that can then be further reformulated using duality or otherwise solved using delayed constraint generation. We provide the details of this derivation in Section EC.4.1 of the ecompanion.

5.4. Log-log demand model

To now show how the solution scheme in Section 5.1 applies to the log-log approach, we again recall the form of the log-log revenue function:

$$R(\mathbf{p}, \mathbf{u}) = \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \quad (34)$$

$$= \sum_{i=1}^I e^{\alpha_i + \log p_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \quad (35)$$

Using the same biconjugate trick as with the semi-log approach, we can show that

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \log R(\mathbf{p}, \mathbf{u}) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{i=1}^I e^{\alpha_i + \log p_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \right) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i (\alpha_i + \log p_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \end{aligned} \quad (36)$$

If we now invoke Assumption 1, then we can introduce the same decision variables $x_{i,t}$ and $w_{i,j,t}$ as in problem (29) to obtain a mixed-integer convex formulation of the log-log price optimization problem, which has the same feasible region as the semi-log formulation (29):

$$\begin{aligned} \underset{\mathbf{w}, \mathbf{x}, \boldsymbol{\mu}}{\text{maximize}} \quad & \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \sum_{i=1}^I \beta_i \cdot \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} + \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \sum_{t \in \mathcal{P}_j} \log t \cdot w_{i,j,t} - \sum_{i=1}^I \mu_i \log \mu_i \\ \text{subject to} \quad & \text{constraints (29b) – (29h).} \end{aligned} \tag{37a}$$

(37b)

While the feasible region of problem (37) is the same as that of (29), the objective function of (37) is different. Just like problem (29), problem (37) can be written as a mixed-integer exponential cone program (see Section EC.3 of the ecompanion for the exact formulation), and the number of variables, linear constraints and exponential cone constraints is the same as that of (29). Similarly, to the best of our knowledge, this is the first exact mixed-integer convex formulation of the log-log multi-product price optimization problem under a Cartesian product price set. Lastly, just like the semi-log problem (29), one can easily modify the formulation to obtain an exact formulation of the deterministic robust price optimization problem under log-log demand (see Section EC.4.2 of the ecompanion).

We note that the log-log separation problem has an interesting property, which is that there exist optimal solutions that are extreme, in the sense that each product's price is set to either its lowest or highest allowable price. This property is formalized in the following proposition (see Section EC.1.7 for the proof).

PROPOSITION 1. *Suppose that Assumption 1 holds. There exists an optimal solution $(\boldsymbol{\mu}, \mathbf{p})$ of problem (36) such that for each $i \in [I]$, either $p_i = \min \mathcal{P}_i$ or $p_i = \max \mathcal{P}_i$.*

The proof follows because when $\boldsymbol{\mu}$ is fixed, the log-transformed separation problem (36) is linear in $(\log p_1, \dots, \log p_I)$; thus, given an optimal but possibly non-extremal solution $(\boldsymbol{\mu}^*, \mathbf{p}^*)$, we can optimize over \mathbf{p} with $\boldsymbol{\mu}^*$ fixed to obtain a new solution $(\boldsymbol{\mu}^*, \mathbf{p}')$ that must also be optimal, and for which \mathbf{p}' is extremal.

5.5. Complexity of randomized robust pricing with finite \mathcal{P}

In this section, we discuss the *complexity* of randomized robust pricing strategies. By complexity, we refer the size of the support of the resulting distribution $\boldsymbol{\pi}$ that solves the RRPO problem. This is an important quantity to analyze for a number of reasons. First, a randomized pricing strategy that randomizes over a small number of price vectors will be easier to implement compared to one that randomizes over a large number of price vectors. Second, from the perspective of the customer, randomized pricing can potentially raise concerns around fairness and price discrimination, as some

customers may be shown more favorable prices for some products than other customers. In this regard, a randomized pricing strategy that randomizes over a smaller set of price vectors could potentially be regarded as more fair than one that randomizes over a larger set of price vectors.

Our key result in this section is an upper bound on the complexity of randomization in the context of RRPO with a finite \mathcal{P} and convex \mathcal{U} .

THEOREM 4. *Suppose $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} for any $\mathbf{p} \in \mathcal{P}$, that \mathcal{U} is a non-empty convex compact subset of \mathbb{R}^M , and that \mathcal{P} is a finite set. Then there exists an optimal solution $\boldsymbol{\pi}^*$ to the RRPO problem $\max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$, such that $\boldsymbol{\pi}^*$ has a support of at most $M + 1$, i.e., the number of price vectors \mathbf{p} for which $\pi_{\mathbf{p}} > 0$ is at most $M + 1$.*

The proof of this result is provided in Section EC.1.8 of the electronic companion. The proof of this result is rather involved, and requires the careful application of results from the paper of Calafiore and Campi (2005) on support constraints. In particular, in a generic convex program with a linear objective function, a *support constraint* is a constraint that, once removed, results in a change in the optimal objective value of the problem. A key result in Calafiore and Campi (2005) is that any convex program with an n -dimensional decision variable has at most n support constraints. In the context of the dual, any active constraint in the dual corresponds to a potentially non-zero primal variable (i.e., a $\pi_{\mathbf{p}}$ for some $\mathbf{p} \in \mathcal{P}$) by complementary slackness. However, while every support constraint is an active constraint at any optimal solution, an active constraint at a given optimal solution need not be a support constraint. The proof of our result thus requires (1) iteratively reducing the dual problem (22) by removing constraints one at a time to obtain an equivalent dual problem with a smaller number of constraints, (2) asserting that all active constraints in this equivalent dual problem are support constraints, and (3) asserting that the number of such constraints is no more than $M + 1$.

Besides the significance of this result in our randomized pricing context, the proof technique is general and could potentially be applied in the context of other convex programs with a linear objective function. On this point, we note that the analogous result for the finite \mathcal{P} and finite \mathcal{U} case (Section EC.7.4 of the ecompanion) is considerably simpler to establish, because the RRPO problem can be written as a linear program and one can invoke standard properties of basic feasible solutions to bound the support of $\boldsymbol{\pi}$. In contrast, problem (22) is not a linear program, and to the best of our knowledge, we are not aware of any characterization of the number of active constraints in general convex programs that one could trivially invoke to obtain the above result.

This result immediately allows us to understand the complexity of randomization for the three demand models that we study. In particular, the dimension M for the linear, semi-log and log-log demand models is at most equal to $I + I + I(I - 1) = I^2 + I$, where I is the number of products.

Thus, Theorem 4 implies that any solution to the RRPO problem when \mathcal{P} is finite and \mathcal{U} is convex will not have a support of size greater than $I^2 + I + 1$, which scales polynomially in the number of products. In practice, the complexity of a randomized robust pricing strategy could be much smaller than this. For example, in Section 7.2, we show an example of an optimal solution to the RRPO problem for the log-log demand model with $I = 11$ products, where the complexity of randomization is only 6.

5.6. Constrained price distributions

Besides the complexity of randomization, it is also interesting to consider solving the RRPO problem with constraints on the set of allowable distributions. By incorporating constraints on the set of distributions that one randomizes over, one can more directly control how prices are randomized and control the variability of the prices. It turns out that moment constraints on the distribution π can be easily incorporated through a suitable modification of the dual problem (22). Due to space constraints, we relegate our discussion of this to Section EC.5 of the ecompanion, which also includes a small set of numerical experiments to show that the price of incorporating such constraints is not too large.

5.7. Convex versus finite price sets

We close Section 5 by discussing our assumption of \mathcal{P} being finite. Thus far, we have assumed that \mathcal{P} is a finite set, rather than a convex set; a natural question is whether it is possible to generalize or modify the approach to handle the case where \mathcal{P} is a convex set.

From a technical standpoint, it is not straightforward to handle the case where \mathcal{P} is a convex set. To see why, observe that the RRPO problem can be reformulated in the following way using Sion's minimax theorem:

$$\begin{aligned} & \max_{F \in \mathcal{F}} \min_{\mathbf{u} \in \mathcal{U}} \int R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \\ &= \min_{\mathbf{u} \in \mathcal{U}} \max_{F \in \mathcal{F}} \int R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \\ &= \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) \end{aligned}$$

where we make the additional assumption that $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} and that \mathcal{U} is a convex compact set. This last problem can be rewritten as

$$\begin{aligned} & \underset{t, \mathbf{u}}{\text{minimize}} && t \end{aligned} \tag{38a}$$

$$\text{subject to} \quad t \geq R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}, \tag{38b}$$

$$\mathbf{u} \in \mathcal{U}, \tag{38c}$$

which is similar to problem (22). The difficulty now arises from \mathcal{P} being a convex set. For the log-log and semi-log demand models, when \mathcal{P} is discrete, the separation problem $\max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ can be reformulated using the biconjugate technique, because one can linearize products of binary variables and the biconjugate variables (μ_1, \dots, μ_I) using standard integer programming ideas. For the present case where \mathcal{P} is convex, applying the biconjugate reformulation immediately leads to a roadblock because now one has to solve a problem with bilinear terms. It is not clear whether an exact reformulation exists in this case. Although one could solve the separation problem heuristically, this would not guarantee that one would obtain a provably optimal solution.

Beside this technical issue, we believe that the restriction to a finite \mathcal{P} is not overly restrictive, and is compatible with practice. First, in most retail scenarios, prices are naturally discrete due to how currencies are denominated. For example, the smallest denomination for US dollars is \$0.01 (the cent), and one does not see prices that are quoted to a finer precision than a hundredth of a dollar in typical retail settings (e.g., at grocery stores). Second, even beside this natural discretization, some retailers will usually consider coarser discretizations due to marketing and behavioral reasons. For example, in marketing it is well-recognized that consumers often respond more positively to prices that end in the number 9 (for example, a price such as \$149 versus \$150; see Anderson and Simester 2003). Relatedly, there is research in the marketing literature that suggests that customers anchor on the first few significant digits of the price and de-emphasize the remaining digits (Bizer and Schindler 2005, Strulov-Shlain 2022); for a product priced in the thousands, this suggests that one should use prices such as \$1999 or \$1499, and not prices such as \$1937 or \$1415. Similarly, other research suggests that customers regard round prices as indicative of higher quality (Stiving 2000). Finally, it is unlikely that a retailer will consider candidate prices that span multiple orders of magnitude for a product. A computer retailer like Apple might price laptops in the thousands of dollars, but likely will not consider prices in the tens of thousands or hundreds of thousands of dollars. This also limits how large a discrete set of price vectors \mathcal{P} can be.

Aside from the practical considerations we have described, we note that it is also possible to show that one can approximate a continuous RRPO problem with a discrete RRPO problem, and to understand theoretically how fine of a discretization one needs to consider. In Section EC.6 of the ecompanion, we show theoretically that for any RRPO problem over a continuous price set, there exists a finite price set such that the objective value of the resulting discrete RRPO problem is arbitrarily close to the true continuous RRPO problem. In particular, the proof of this result implies that restricting the price of each product to a uniform grid, where the coarseness of the grid is informed by the Lipschitz constant of the revenue function, is sufficient to obtain a good approximation of the true continuous problem.

6. Solution method for finite \mathcal{P} , finite \mathcal{U}

In addition to the case where \mathcal{U} is convex, we also consider the case where \mathcal{U} is a finite discrete set. Due to page limitations, our presentation of our solution method for this case is relegated to Section EC.7 of the ecompanion. At a high level, the foundation of our approach is double column generation, which alternates between solving the primal version of the RRPO problem, which is $\max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$, and the dual version of the RRPO problem, which is $\min_{\lambda \in \Delta_{\mathcal{U}}} \max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \mathcal{U}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u})$. In each iteration, we solve the primal problem with \mathcal{P} replaced by a subset $\hat{\mathcal{P}} \subseteq \mathcal{P}$, where we use constraint generation to handle the inner minimization over $\mathbf{u} \in \mathcal{U}$; this gives rise to a finite set of uncertainty realizations $\hat{\mathcal{U}} \subseteq \mathcal{U}$. We then solve the dual problem with \mathcal{U} replaced by $\hat{\mathcal{U}}$, where we use constraint generation to handle the inner maximization over $\mathbf{p} \in \mathcal{P}$, which gives rise to a finite set of price vectors $\hat{\mathcal{P}} \subseteq \mathcal{P}$. At each step of the algorithm, the objective value of the primal problem restricted to $\hat{\mathcal{P}}$ is a lower bound on the true optimal objective, while the objective value of the dual problem restricted to $\hat{\mathcal{U}}$ is an upper bound on the optimal objective; the algorithm terminates when these two bounds are equal or are otherwise within a pre-specified tolerance.

To implement this approach for the demand models that we consider, one needs to be able to solve the primal separation problem (solve $\min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$) and the dual separation problem (solve $\max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot R(\mathbf{p}, \mathbf{u})$). We show how both of these problems can be reformulated as mixed-integer exponential cone programs for the semi-log and log-log demand models, and as mixed-integer linear programs for the linear demand model.

7. Numerical experiments

In this section, we conduct several sets of experiments involving synthetic problem instances to understand the tractability of the RRPO approach and the improvement in worst-case revenue of the randomized robust pricing strategy over the deterministic robust pricing strategy. In Section 7.1, we consider instances involving the linear, semi-log and log-log models where the uncertainty set \mathcal{U} is a convex set. We also consider instances involving the linear, semi-log and log-log models where the uncertainty set \mathcal{U} is a discrete set; due to space considerations, these results are presented in Section EC.8.1 of the ecompanion. In Section 7.2, we consider log-log and semi-log robust price optimization instances derived from a real data set on sales of orange juice products from a grocery store chain. Lastly, in Section 7.3, we consider a data-driven experiment using the same orange juice data set that illustrates the benefit of randomized robust price strategies over deterministic robust and non-robust price prescriptions in out-of-sample performance.

All of our code is implemented in the Julia programming language (Bezanson et al. 2017). All optimization models are implemented using the JuMP package (Lubin and Dunning 2015). All

linear and mixed-integer linear programs are solved using Gurobi (Gurobi Optimization, LLC 2023) and all mixed-integer exponential cone programs are solved using Mosek (Mosek ApS 2022), with a maximum of 8 threads per program. All of our experiments are conducted on Amazon Elastic Compute Cloud (EC2), on a single instance of type `m6a.48xlarge` (AMD EPYC 7R13 processor, with 192 virtual CPUs and 768 GB of memory).

7.1. Experiments with convex \mathcal{U} and linear, log-log and semi-log demand models

In our first set of experiments, we consider the log-log and semi-log demand models, and specifically consider a $L1$ -norm uncertainty set \mathcal{U} :

$$\mathcal{U} = \left\{ \mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \mid \|\tilde{\mathbf{u}}\|_1 \leq \theta; \tilde{\mathbf{u}}_k = \frac{\mathbf{u}_k - \mathbf{u}_{0k}}{\mathbf{u}_{0k}}, \forall k \in \{1, \dots, I + I^2\} \right\}, \quad (39)$$

where \mathbf{u}_0 is the vector of nominal values of the uncertain parameters $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$, and θ is the budget of the uncertainty set. Informally, the rationale for this type of uncertainty set is that the nominal demand model parameter estimates are almost certainly wrong when they are estimated from a small sample of data. The perspective that one takes with this type of uncertainty set is that while there could be error in any individual demand model parameter, we expect that the aggregate relative error should not be too large (for example, we do not expect a large number of demand model parameters to exhibit large relative errors). We note that this type of budget uncertainty set is common in the robust optimization literature (see for example Bertsimas and Sim 2004, Bertsimas and Thiele 2006).

For each of the three demand models (linear, semi-log and log-log), the number of products I varies in $\{5, 10, 15, 20\}$. For each value of I , we generate 24 random instances, where the values of $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ are independently randomly generated as follows:

1. *Linear demand.* Each $\alpha_i \sim \text{Uniform}(100, 200)$, $\beta_i \sim \text{Uniform}(5, 15)$, $\gamma_{i,j} \sim \text{Uniform}(-0.1, +0.1)$.
2. *Semi-log demand.* Each $\alpha_i \sim \text{Uniform}(8, 10)$, $\beta_i \sim \text{Uniform}(1.5, 2)$, $\gamma_{i,j} \sim \text{Uniform}(-0.5, +0.5)$.
3. *Log-log demand.* Each $\alpha_i \sim \text{Uniform}(10, 14)$, $\beta_i \sim \text{Uniform}(1.5, 2)$, $\gamma_{i,j} \sim \text{Uniform}(-0.8, +0.8)$.

For each product $i \in [I]$, we set $\mathcal{P}_i = \{1, 2, 3, 4, 5\}$.

For each instance, we solve the nominal problem, the DRPO problem and the RRPO problem. For DRPO and RRPO, we vary the budget parameter θ that defines the uncertainty within the set $\{0.1, 0.5, 1, 1.5, 2\}$. To solve the RRPO problem for each instance, we execute the constraint generation solution algorithm described in Section 5. For instances with log-log demand, we take advantage of Proposition 1 and thus simplify the price set \mathcal{P} to contain the highest and lowest price levels for each product only. To solve the DRPO problem, we formulate it as either a mixed-integer linear program (for linear demand) or a mixed-integer exponential cone program (for semi-log and log-log demand) via the log-sum-exp biconjugate-based technique described in Section EC.4 of

the ecompanion, and use standard LP duality techniques to reformulate the objective function of the resulting problem (formulation (EC.46) and formulation (EC.47) in Section EC.4). Due to the prohibitive computation times that we encountered for the DRPO problem with log-log and semi-log demand, we impose a computation time limit of 20 minutes. From our experimentation with the DRPO problem for log-log and semi-log, it is often the case that an optimal or nearly optimal solution is found early on, and the bulk of the remaining computation time, which can be in the hours, is required by Mosek to prove optimality and close the gap. Finally, to solve the nominal problem for each instance, we either solve it as a mixed-integer linear program (for linear demand) or otherwise use the biconjugate technique to formulate it as a mixed-integer exponential cone program (for log-log and semi-log demand).

We present the objective value as well as the computation time of each RRPO, DRPO and nominal problem. We additionally compute several other metrics. We compute $\mathbb{E}[R(\mathbf{p}_{\text{RR}}^*, \mathbf{u}_0)]$, which is the expected revenue of the randomized RPO solution assuming that the nominal parameter values are realized. We also compute $R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}_0)$, the nominal revenue of DRPO solution, and $Z_{\text{N,WC}} = \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{N}}^*, \mathbf{u})$, the worst-case revenue of the nominal solution. We use the following metric, RI (relative improvement), to show the benefit of randomized strategy in robust price optimization:

$$\text{RI} = (Z_{\text{RR}}^* - Z_{\text{DR}}^*) / Z_{\text{DR}}^* \times 100\% \quad (40)$$

We additionally record the solution time for the randomized robust, deterministic robust and nominal approaches, which we denote by t_{RR} , t_{DR} and t_{N} , respectively. For each metric, we compute its average over the 24 instances for each value of I and θ .

Tables 1, 2 and 3 shows the results for the linear, semi-log and log-log demand models, respectively. For linear demand, we find that the improvement by randomized robust pricing over deterministic robust pricing is modest; the largest average improvement is 8.49% for $I = 5$, $\theta = 2$. We note that we experimented with other forms of uncertainty sets and choices of the nominal parameter values for the linear demand model, but we generally did not encounter large improvements of the same size as we did for the other two demand models.

Besides linear demand, these results also show that for semi-log and log-log demand, there can be a very large difference between the randomized and deterministic robust pricing schemes. The benefit of randomization, quantified by the metric RI, ranges from about 5% to as much as 8875% for semi-log instances, and from about 8% to 925% for log-log instances. Note that the magnitude of RI for the semi-log instances is larger than that for log-log, because the logarithm of demand in the semi-log model has a linear dependence on price which results in an exponential dependence of demand on price, but in log-log, the logarithm of demand is linear in the logarithm

I	θ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
5	0.1	0.07	2382.23	2473.30	0.03	2382.23	0.00	2473.30	0.03	2473.30	2382.23
5	0.5	0.10	2022.13	2464.54	0.03	2019.55	0.16	2468.20	—	—	2017.94
5	1.0	0.24	1602.35	2397.08	0.04	1583.01	1.34	2424.11	—	—	1562.58
5	1.5	0.39	1231.02	2271.26	0.04	1186.52	3.97	2321.85	—	—	1135.17
5	2.0	0.61	911.15	2084.73	0.04	844.02	8.49	2114.73	—	—	707.77
10	0.1	0.20	4912.69	5009.03	0.11	4912.69	0.00	5009.03	0.10	5009.03	4912.69
10	0.5	0.25	4530.01	5001.22	0.11	4527.34	0.06	5009.03	—	—	4527.34
10	1.0	0.53	4072.51	4969.01	0.12	4050.19	0.57	4980.27	—	—	4045.64
10	1.5	0.91	3639.15	4901.17	0.12	3594.08	1.28	4941.42	—	—	3592.30
10	2.0	1.44	3229.58	4836.17	0.14	3155.97	2.38	4864.44	—	—	3138.96
15	0.1	0.36	7416.52	7513.95	0.22	7416.52	0.00	7513.95	0.17	7513.95	7416.52
15	0.5	0.47	7028.22	7508.13	0.23	7026.80	0.02	7513.95	—	—	7026.80
15	1.0	1.03	6557.32	7483.57	0.25	6539.66	0.28	7513.95	—	—	6539.66
15	1.5	1.70	6105.29	7445.90	0.26	6061.74	0.74	7472.86	—	—	6071.54
15	2.0	2.56	5664.92	7365.34	0.29	5593.44	1.31	7459.25	—	—	5603.43
20	0.1	0.62	9840.40	9936.95	0.37	9840.40	0.00	9936.95	0.29	9936.95	9840.40
20	0.5	0.83	9455.48	9932.58	0.40	9454.20	0.01	9936.95	—	—	9454.20
20	1.0	1.96	8983.11	9921.01	0.43	8971.45	0.13	9936.95	—	—	8971.45
20	1.5	3.25	8518.85	9893.43	0.45	8489.54	0.35	9931.17	—	—	8498.79
20	2.0	4.77	8067.97	9856.06	0.47	8008.99	0.75	9931.17	—	—	8026.12

Table 1 Results for linear instances with convex \mathcal{U} .

I	θ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
5	0.1	0.15	9.37×10^4	2.21×10^5	0.10	9.20×10^4	4.89	2.25×10^5	0.05	2.27×10^5	9.15×10^4
5	0.5	0.31	1.06×10^4	1.14×10^5	0.16	6.58×10^3	64.55	7.74×10^4	—	—	3.61×10^3
5	1.0	0.37	2.88×10^3	7.89×10^4	0.17	1.50×10^3	88.40	4.72×10^4	—	—	3.64×10^2
5	1.5	0.40	1.07×10^3	6.99×10^4	0.18	4.90×10^2	106.08	2.23×10^4	—	—	8.44×10^1
5	2.0	0.41	4.24×10^2	6.65×10^4	0.18	1.84×10^2	116.57	1.44×10^4	—	—	2.94×10^1
10	0.1	0.70	3.01×10^7	6.33×10^7	0.44	2.87×10^7	12.02	7.16×10^7	0.17	7.16×10^7	2.87×10^7
10	0.5	1.31	4.37×10^6	3.90×10^7	1.47	1.30×10^6	202.19	4.51×10^7	—	—	9.48×10^5
10	1.0	1.91	9.33×10^5	3.28×10^7	4.45	1.39×10^5	442.91	1.62×10^7	—	—	5.68×10^4
10	1.5	3.05	3.15×10^5	1.99×10^7	10.29	3.31×10^4	652.95	7.48×10^6	—	—	9.56×10^3
10	2.0	3.94	1.36×10^5	1.84×10^7	18.12	1.13×10^4	844.92	1.1×10^7	—	—	4.05×10^3
15	0.1	1.80	2.04×10^9	4.99×10^9	1.77	1.99×10^9	9.24	5.05×10^9	0.51	5.05×10^9	1.99×10^9
15	0.5	3.66	2.02×10^8	3.84×10^9	13.40	6.27×10^7	295.04	3.52×10^9	—	—	5.05×10^7
15	1.0	7.39	3.03×10^7	2.23×10^9	133.51	4.65×10^6	879.80	1.40×10^9	—	—	1.55×10^6
15	1.5	14.07	9.14×10^6	1.61×10^9	595.51	9.40×10^5	1451.79	2.66×10^9	—	—	3.80×10^5
15	2.0	18.21	3.99×10^6	1.25×10^9	800.14	2.81×10^5	1957.47	2.17×10^9	—	—	1.52×10^5
20	0.1	6.25	1.57×10^{11}	4.1×10^{11}	5.47	1.57×10^{11}	13.12	4.11×10^{11}	1.97	4.11×10^{11}	1.57×10^{11}
20	0.5	18.29	1.05×10^{10}	2.06×10^{11}	217.95	3.36×10^9	577.40	4.09×10^{11}	—	—	3.34×10^9
20	1.0	24.63	1.53×10^9	1.92×10^{11}	972.80	1.03×10^8	2848.23	2.55×10^{10}	—	—	3.78×10^7
20	1.5	41.60	4.73×10^8	1.32×10^{11}	1126.14	1.90×10^7	6113.91	2.05×10^{10}	—	—	1.38×10^7
20	2.0	46.11	2.01×10^8	1.00×10^{11}	1140.78	6.22×10^6	8874.96	3.66×10^{11}	—	—	5.33×10^6

Table 2 Results for semi-log instances with convex \mathcal{U} .

of price, resulting in a milder polynomial dependence of demand on price. For semi-log and log-log demand, both the worst-case revenue of RRPO solution and the worst-case revenue of DRPO

I	θ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
5	0.1	0.12	1.77×10^6	3.88×10^6	0.11	1.66×10^6	8.35	4.02×10^6	0.07	4.31×10^6	1.55×10^6
5	0.5	0.18	3.95×10^5	2.64×10^6	0.15	3.44×10^5	18.10	2.45×10^6	–	–	2.21×10^5
5	1.0	0.19	1.08×10^5	2.45×10^6	0.15	9.42×10^4	18.67	2.25×10^6	–	–	5.42×10^4
5	1.5	0.18	3.20×10^4	2.41×10^6	0.15	2.74×10^4	19.99	2.25×10^6	–	–	1.57×10^4
5	2.0	0.19	9.50×10^3	2.41×10^6	0.16	8.09×10^3	20.17	2.15×10^6	–	–	4.63×10^3
10	0.1	0.51	2.87×10^7	6.03×10^7	1.41	2.32×10^7	27.15	7.44×10^7	0.37	7.66×10^7	2.31×10^7
10	0.5	1.15	6.16×10^6	3.67×10^7	15.88	3.14×10^6	90.19	2.61×10^7	–	–	1.87×10^6
10	1.0	1.81	2.20×10^6	2.58×10^7	46.27	1.02×10^6	111.15	1.77×10^7	–	–	5.52×10^5
10	1.5	2.19	9.97×10^5	2.30×10^7	59.82	4.66×10^5	113.12	1.35×10^7	–	–	2.35×10^5
10	2.0	2.54	5.04×10^5	2.12×10^7	63.96	2.38×10^5	113.39	1.42×10^7	–	–	1.14×10^5
15	0.1	1.59	2.51×10^8	7.46×10^8	8.39	2.35×10^8	19.38	8.28×10^8	1.71	8.38×10^8	2.32×10^8
15	0.5	5.71	4.33×10^7	2.86×10^8	728.17	1.67×10^7	202.98	2.46×10^8	–	–	8.21×10^6
15	1.0	9.62	1.67×10^7	2.02×10^8	1155.56	5.07×10^6	257.70	1.82×10^8	–	–	2.29×10^6
15	1.5	13.02	8.59×10^6	1.56×10^8	1200.60	2.31×10^6	289.39	1.20×10^8	–	–	1.03×10^6
15	2.0	16.49	4.99×10^6	1.36×10^8	1200.66	1.25×10^6	309.56	5.15×10^7	–	–	5.44×10^5
20	0.1	4.82	2.52×10^9	7.86×10^9	79.04	2.27×10^9	41.87	8.63×10^9	5.17	8.63×10^9	2.27×10^9
20	0.5	14.16	3.72×10^8	3.04×10^9	1200.35	6.68×10^7	512.22	1.39×10^9	–	–	2.77×10^7
20	1.0	24.35	1.42×10^8	1.75×10^9	1200.74	1.85×10^7	745.22	7.42×10^8	–	–	5.84×10^6
20	1.5	37.00	7.44×10^7	1.34×10^9	1200.87	8.69×10^6	870.06	3.42×10^8	–	–	2.75×10^6
20	2.0	44.84	4.45×10^7	1.13×10^9	1200.96	4.74×10^6	925.14	3.24×10^8	–	–	1.57×10^6

Table 3 Results for log-log instances with convex \mathcal{U} .

solution decrease as the uncertainty set becomes larger, and the rate of reduction becomes less as the uncertainty budget θ is larger. In addition, for linear, semi-log and log-log demand, the RI generally increases as the uncertainty budget θ increases. Also, as we expect, $Z_{RR}^* \geq Z_{DR}^* \geq Z_{N,WC}$. Interestingly, the randomized robust pricing scheme can sometimes achieve better performance than the deterministic robust scheme under the nominal demand model (for example, compare $\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$ and $R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$ for log-log demand with $I = 10$, $\theta \geq 0.5$); this appears to be the case for almost all (I, θ) combinations for log-log, and for a smaller set of (I, θ) combinations for semi-log.

With regard to the computation time, we observe that the computation time generally grows with the number of products for both RRPO and DRPO. For linear demand, both RRPO and DRPO can be solved extremely quickly (no more than 5 seconds on average, even with $I = 20$ products). For log-log and semi-log, when the number of products is held constant, the amount of time required to solve either RRPO generally becomes larger as the uncertainty set becomes larger, but in all cases the average time is under a minute. What we also find is that for both log-log and semi-log demand, RRPO generally requires much less time to solve to complete optimality than DRPO; this is likely because the nominal problem (which is a key piece of the constraint generation method for RRPO when \mathcal{U} is convex) can be solved rapidly, whereas the robust version of this mixed-integer exponential cone program is more challenging for Mosek.

7.2. Results using real data instances

In our next set of experiments, we evaluate the effectiveness of solution algorithms on problem instances calibrated with real data. For these experiments, we consider the `orangeJuice` data set from Montgomery (1997), which was accessed via the `bayesm` package in R (Rossi 2022). This data set contains price and sales data for $I = 11$ different orange juice brands at the Dominick's Finer Foods chain of grocery stores in the Chicago area. Each observation in the data set consists of: the store s ; the week t ; the log of the number of units sold $\log(q_{t,s,i})$ for brand i ; the prices $p_{t,s,1} \dots p_{t,s,11}$ of the eleven orange juice brands; the dummy variable $d_{t,s,i}$ indicating whether brand i had any in-store displays at store s in week t ; and the variable $f_{t,s,i}$ indicating if brand i was featured/advertised at store s in week t . We fit log-log and semi-log regression models for each brand i according to the following specifications:

$$(\text{semi-log}) \quad \log(q_{t,s,i}) = \alpha_i - \beta_i p_{t,s,i} + \sum_{j \neq i} \gamma_{ij} p_{t,s,j} + \psi_i d_{t,s,i} + \phi_i f_{t,s,i} + \epsilon_{t,s,i}, \quad (41)$$

$$(\text{log-log}) \quad \log(q_{t,s,i}) = \alpha_i - \beta_i \log(p_{t,s,i}) + \sum_{j \neq i} \gamma_{ij} \log(p_{t,s,j}) + \psi_i d_{t,s,i} + \phi_i f_{t,s,i} + \epsilon_{t,s,i}, \quad (42)$$

where $\{\epsilon_{t,s,i}\}_{t,s,i}$ is a collection of IID normally distributed error terms. The point estimates of the model parameters are provided in Section EC.8.4 of the ecompanion. We note that prior work has considered the estimation of both of these types of models (see the examples in Rossi 2022; see also Montgomery 1997 and Mišić 2020).

We consider the problem of obtaining a price vector $\mathbf{p} = (p_1, p_2, \dots, p_{11})$ for this collection of 11 products. To formulate the price vector set \mathcal{P} , we assume that each product i has five allowable prices, which are shown in Table 4. These prices correspond to the 0th (i.e., minimum), 25th, 50th, 75th and 100th (i.e., maximum) percentiles of the observed prices in the dataset.

Product	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
	1.29	2.86	1.25	0.99	0.88	2.76	0.91	0.91	0.69	0.52	1.99
	2.49	4.19	2.69	1.99	1.99	3.67	1.99	1.99	1.79	1.58	2.99
	2.99	4.75	2.89	2.35	2.17	3.96	2.39	2.19	1.99	1.59	3.59
	3.19	4.99	3.12	2.49	2.49	4.49	2.56	2.39	2.36	1.99	3.99
	3.87	5.82	3.35	3.06	3.17	5.09	3.07	2.69	3.08	2.69	4.99

Table 4 Possible price levels for products in `orangeJuice` experiment instances.

For each type of demand model, we consider two forms of uncertainty set: a convex $L1$ -norm uncertainty set (as in equation (39)) and a discrete budget uncertainty set (as in equation (EC.115)). Specifically, for the discrete budget uncertainty set, we assume that $\bar{\alpha} = 1.2\alpha$, $\underline{\alpha} = 0.8\alpha$, $\bar{\beta} = 1.3\beta$, $\underline{\beta} = 0.7\beta$, $\bar{\gamma} = 1.4\gamma$, and $\underline{\gamma} = 0.6\gamma$. Tables 5 and 6 below present the results

under the convex $L1$ -norm uncertainty set for the semi-log and log-log demand models, respectively. Due to page considerations, the results for the discrete \mathcal{U} case are provided in Section EC.8.5 of the ecompanion.

We can see from Tables 5 and 6 that the randomized pricing strategy performs significantly better than the deterministic pricing solution under the worst-case demand model, with the RI ranging from 17.86% to 47.81% for semi-log demand and from 27.71% to 92.31% for log-log demand. In addition, for the same demand type and uncertainty set, the computation time of RRPO is comparable to that of DRPO. With regard to the discrete uncertainty set case, the results shown in Section EC.8.5 are qualitatively similar, with the randomized robust pricing strategy similarly outperforming the deterministic robust solution. We do also observe that under both demand models, the time to solve the RRPO problem is generally smaller than for the DRPO problem (RRPO requires no more than about 17 seconds, whereas DRPO can take up to 300 seconds on average in the worst case).

θ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
0.10	16.86	342357.06	481125.25	15.98	290474.67	17.86	590546.51	0.83	590547.01	290474.76
0.50	41.42	197517.06	373973.40	47.65	147748.35	33.68	294483.28	—	—	96016.90
0.80	42.61	149709.04	352742.22	67.83	105734.14	41.59	294483.28	—	—	67924.78
1.00	67.47	125987.02	349644.33	74.03	86977.24	44.85	265173.68	—	—	55394.70
1.50	61.70	82880.96	348467.74	44.65	56474.64	46.76	265173.68	—	—	34864.43
2.00	65.21	54665.15	348466.26	52.10	37164.75	47.09	265173.68	—	—	22615.70

Table 5 Results for orangeJuice pricing problem with semi-log demand and convex \mathcal{U} .

θ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
0.10	0.71	722647.22	1051269.80	1.97	565866.71	27.71	922172.97	0.88	1112050.59	560812.30
0.50	1.42	342614.34	687632.84	4.99	233387.10	46.80	782893.68	—	—	152881.89
0.80	1.91	260049.66	672481.74	6.77	162276.97	60.25	782893.68	—	—	102893.20
1.00	2.08	217580.86	683576.99	9.41	128220.45	69.69	782893.68	—	—	81427.57
1.50	2.26	142307.66	670759.12	13.02	75897.66	87.50	599315.12	—	—	48983.56
2.00	2.39	94847.37	670758.38	13.49	49319.21	92.31	377932.71	—	—	31055.19

Table 6 Results for orangeJuice pricing problem with log-log demand and convex \mathcal{U} .

Lastly, it is also interesting to compare the randomized robust pricing strategy to the deterministic robust price vector. Taking the log-log demand model and the convex $L1$ uncertainty set

with $\theta = 0.8$ as an example, the solution of the RRPO problem is the following randomized pricing strategy:

$$\mathbf{p} = \begin{cases} (3.87, 5.82, 1.25, 0.99, 3.17, 5.09, 3.07, 0.91, 0.69, 2.69, 1.99) & \text{w.p. 0.1628,} \\ (1.29, 5.82, 3.35, 3.06, 0.88, 2.76, 3.07, 2.69, 3.08, 2.69, 4.99) & \text{w.p. 0.1752,} \\ (3.87, 2.86, 1.25, 3.06, 3.17, 5.09, 0.91, 2.69, 3.08, 0.52, 4.99) & \text{w.p. 0.2658,} \\ (3.87, 5.82, 3.35, 3.06, 0.88, 2.76, 3.07, 0.91, 3.08, 2.69, 4.99) & \text{w.p. 0.0381,} \\ (3.87, 2.86, 1.25, 3.06, 3.17, 2.76, 3.07, 2.69, 0.69, 2.69, 4.99) & \text{w.p. 0.3258,} \\ (3.87, 2.86, 1.25, 3.06, 3.17, 5.09, 0.91, 0.91, 3.08, 0.52, 4.99) & \text{w.p. 0.0323.} \end{cases} \quad (43)$$

Observe that in this randomized pricing strategy, each price vector is such that the product is set to either its lowest or highest allowable price. This is congruent with Proposition 1, which suggests that the nominal problem under the log-log demand model will always have a solution that involves setting each product to its highest or lowest price; since our solution algorithm is based on constraint generation using this nominal problem as a separation procedure, it makes sense that the randomized price vector will be supported on such extremal price vectors. On the other hand, the solution of the DRPO problem is the price vector $\mathbf{p}_{\text{DR}} = (3.87, 2.86, 1.25, 3.06, 3.17, 2.76, 0.91, 2.69, 0.69, 0.52, 4.99)$, for which we observe that the chosen prices are also either the lowest or highest for each product.

7.3. Data-driven evaluation of out-of-sample performance

In this section, we present the results of a data-driven experiment to demonstrate the benefit of randomized robust decisions in terms of out-of-sample revenue. The previous numerical experiments show that from a risk standpoint, randomized robust pricing decisions can lead to better worst-case performance than deterministic robust pricing decisions and nominal/non-robust deterministic pricing decisions. In contrast, this section focuses on a different question, which we state as follows. Suppose we are given a data set of historical price vectors and demands, and we use this data set to craft an appropriate uncertainty set for the randomized robust price optimization problem; how does the resulting randomized robust pricing decision perform with respect to the true, unknown demand model?

To understand this question, we will again consider the **orangeJuice** data set used previously in Section 7.2, and compare the randomized robust, deterministic robust and nominal approaches. The high-level idea of this experiment is as follows: for each store, divide the available price-demand data into training, validation and test sets. We use the training set to estimate a nominal demand model. We use the validation set to tune the uncertainty set for the randomized robust and deterministic robust approaches. We then use the test set to evaluate the randomized robust, deterministic robust and nominal approaches.

A key challenge that we face in using the **orangeJuice** data set (or any real data set, for that matter) is that the historical data only gives demand (and hence revenue) information on a small,

finite number of price vectors. For any price vector that was not offered in a given store, we do not know what the revenue would have been. Thus, to perform the validation and test steps in our experiment, we proceed by considering a *restricted pricing problem*. Namely, in the validation step, we solve restricted versions of the RRPO and DRPO problems where we can only select from price vectors that appear in the validation set. Then, given the pricing strategy that is returned by RRPO/DRPO, we evaluate its true revenue using the *real* revenues of the corresponding price vectors in the validation set. Similarly, in the test step, we solve restricted versions of the RRPO, DRPO and NPO problems where we can only use price vectors that appear in the test set, where RRPO and DRPO are solved using the uncertainty sets designed in the validation step. Given the pricing strategies produced by the restricted RRPO, DRPO and NPO problems, we calculate their revenues using the true revenues for their price vectors in the test set.

To formalize this experiment, we let $\mathcal{D} = \{(\mathbf{p}_i, \mathbf{d}_i)\}_{i=1}^{n'}$ be the historical data set for a single store in the data set, where \mathbf{p}_i is the vector of prices in week i and \mathbf{d}_i is the vector of demands in week i ; consequently, $\mathbf{p}_i^T \mathbf{d}_i$ is the revenue garnered in week i . Let $\mathcal{P} = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ be the set of unique price vectors corresponding to \mathcal{D} .

Let $\mathcal{P}_{\text{train}}, \mathcal{P}_{\text{val}}, \mathcal{P}_{\text{test}}$ be a partition of the set of indices $[n] = \{1, \dots, n\}$, such that $\mathcal{P} = \mathcal{P}_{\text{train}} \cup \mathcal{P}_{\text{val}} \cup \mathcal{P}_{\text{test}}$, where each \mathcal{P} set indicates which price vectors are assigned to the training, validation and test sets. Let the functions $\bar{R}: \mathcal{P} \rightarrow \mathbb{R}$ denote the average revenue of each price vector in the data set, which is defined as

$$\bar{R}(\mathbf{p}) = \frac{1}{|\{i \in [n'] \mid \mathbf{p}_i = \mathbf{p}\}|} \cdot \sum_{\substack{i=1: \\ \mathbf{p}_i = \mathbf{p}}}^{n'} \mathbf{p}^T \mathbf{d}_i, \quad (44)$$

Note that in cases where \mathbf{p} only occurs in a single week in the validation or test sets, then $\bar{R}(\mathbf{p})$ will be the revenue of \mathbf{p} in that single historical week. We also point out here that $\bar{R}(\mathbf{p})$ is distinct from $R(\mathbf{p}, \mathbf{u})$: the former is our data-driven estimate of the average revenue of the price vector \mathbf{p} , whereas $R(\mathbf{p}, \mathbf{u})$ is the predicted revenue from a demand model with parameters \mathbf{u} . We additionally let $\mathcal{D}_{\text{train}} = \{(\mathbf{p}, \mathbf{d}) \in \mathcal{D} \mid \mathbf{p} \in \mathcal{P}_{\text{train}}\}$ denote the price-demand observations for the price vectors corresponding to the training set.

Let us assume that the uncertainty set for the DRPO and RRPO approaches is parametrized by a vector \mathbf{w} , so that $\mathcal{U}(\mathbf{w})$ corresponds to the uncertainty set with the control parameters \mathbf{w} . We let \mathcal{W} denote the set of control parameters that we will choose from in our validation step.

We now describe our overall experimental procedure as Algorithm 1.

We apply Algorithm 1 to the price-demand data of each store in the orange juice data set. For each store, we partition the set of n historical price vectors so that approximately the first 50% of price vectors constitute the training set $\mathcal{P}_{\text{train}}$, the next 25% of price vectors constitute the validation

Algorithm 1 Data-driven experiment procedure for `orangeJuice` data set.

-
- 1: Estimate nominal demand model parameter vector \mathbf{u}^N using $\mathcal{D}_{\text{train}}$.
 - 2: **for** $\mathbf{w} \in \mathcal{W}$ **do**
 - 3: Solve $\mathbf{p}^{\text{DR},\mathbf{w}} \in \arg \max_{\mathbf{p} \in \mathcal{P}_{\text{val}}} \min_{\mathbf{u} \in \mathcal{U}(\mathbf{w})} R(\mathbf{p}, \mathbf{u})$.
 - 4: Solve $\boldsymbol{\pi}^{\text{RR},\mathbf{w}} \in \arg \max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}_{\text{val}}}} \min_{\mathbf{u} \in \mathcal{U}(\mathbf{w})} \sum_{\mathbf{p} \in \mathcal{P}_{\text{val}}} \pi_{\mathbf{p}} \cdot R(\mathbf{p}, \mathbf{u})$.
 - 5: **end for**
 - 6: Set $\mathbf{w}_{\text{DR}} = \arg \max_{\mathbf{w} \in \mathcal{W}} \bar{R}(\mathbf{p}^{\text{DR},\mathbf{w}})$.
 - 7: Set $\mathbf{w}_{\text{RR}} = \arg \max_{\mathbf{w} \in \mathcal{W}} \sum_{\mathbf{p} \in \mathcal{P}_{\text{val}}} \pi_{\mathbf{p}}^{\text{RR},\mathbf{w}} \bar{R}(\mathbf{p})$.
 - 8: Solve $\mathbf{p}^{\text{N},\text{test}} \in \arg \max_{\mathbf{p} \in \mathcal{P}_{\text{test}}} R(\mathbf{p}, \mathbf{u}^N)$.
 - 9: Calculate $R^{\text{N},\text{test}} = \bar{R}(\mathbf{p}^{\text{N},\text{test}})$.
 - 10: Solve $\mathbf{p}^{\text{DR},\text{test}} \in \arg \max_{\mathbf{p} \in \mathcal{P}_{\text{test}}} \min_{\mathbf{u} \in \mathcal{U}(\mathbf{w}_{\text{DR}})} R(\mathbf{p}, \mathbf{u})$.
 - 11: Calculate $R^{\text{DR},\text{test}} = \bar{R}(\mathbf{p}^{\text{DR},\text{test}})$.
 - 12: Solve $\boldsymbol{\pi}^{\text{RR},\text{test}} \in \arg \max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}_{\text{test}}}} \min_{\mathbf{u} \in \mathcal{U}(\mathbf{w}_{\text{RR}})} \sum_{\mathbf{p} \in \mathcal{P}_{\text{test}}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$.
 - 13: Calculate $R^{\text{RR},\text{test}} = \sum_{\mathbf{p} \in \mathcal{P}_{\text{test}}} \pi_{\mathbf{p}}^{\text{RR},\text{test}} \bar{R}(\mathbf{p})$.
 - 14: **return** $R^{\text{N},\text{test}}, R^{\text{DR},\text{test}}, R^{\text{RR},\text{test}}$.
-

set \mathcal{P}_{val} , and the final 25% of price vectors constitute the test set $\mathcal{P}_{\text{test}}$. Note that the price vectors in \mathcal{P} are sorted in order of temporal appearance; thus, the first occurrence in time of each price vector in $\mathcal{P}_{\text{train}}$ is before that of any price vector in \mathcal{P}_{val} , and the first occurrence in time of each price vector in \mathcal{P}_{val} is before that of any price vector in $\mathcal{P}_{\text{test}}$.

For the uncertainty set \mathcal{U} , we consider the following budget uncertainty set:

$$\mathcal{U}(\mathbf{w}) = \left\{ \mathbf{u} = (\mathbf{a}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \mid \|\tilde{\mathbf{u}}\|_1 \leq w_1; \|\tilde{\mathbf{u}}\|_\infty \leq w_2; \tilde{u}_k = \frac{u_k - u_k^N}{u_k^N}, \forall k \in \{1, \dots, I + I^2\} \right\}, \quad (45)$$

where the term \tilde{u}_k measures the relative deviation of the k th demand model parameter from its nominal value, the parameter w_1 is an upper bound on the aggregate deviation of all of the demand model parameters (i.e., the budget of uncertainty), and the parameter w_2 is an upper limit on the allowable relative deviation of each demand model parameter from its nominal value. We set $\mathcal{W} = \{0, 0.2, 0.4, \dots, 10.0\} \times \{0, 0.1, \dots, 1.0\}$. We test both the semi-log and log-log demand models.

In addition to the deterministic robust and randomized robust approaches, we also consider an approach that combines the two, which we refer to as the hybrid approach. The hybrid approach works as follows. In the validation phase, we calculate $R^{\text{RR},\text{val}}$ and $R^{\text{DR},\text{val}}$ as the best values of the randomized robust and deterministic robust approaches, respectively, on the validation set:

$$R^{\text{RR},\text{val}} = \max_{\mathbf{w} \in \mathcal{W}} \bar{R}(\mathbf{p}^{\text{RR},\mathbf{w}}), \quad (46)$$

$$R^{\text{DR},\text{val}} = \max_{\mathbf{w} \in \mathcal{W}} \bar{R}(\mathbf{p}^{\text{DR},\mathbf{w}}). \quad (47)$$

Then in the test phase, we implement the randomized robust approach if $R^{\text{RR},\text{val}} \geq R^{\text{DR},\text{val}}$ and the deterministic robust approach otherwise (i.e., if $R^{\text{RR},\text{val}} < R^{\text{DR},\text{val}}$). The performance of this hybrid method on the test set can thus be written as

$$R^{\text{H},\text{test}} = R^{\text{RR},\text{test}} \cdot \mathbb{I}\{R^{\text{RR},\text{val}} \geq R^{\text{DR},\text{val}}\} + R^{\text{DR},\text{test}} \cdot \mathbb{I}\{R^{\text{RR},\text{val}} < R^{\text{DR},\text{val}}\}. \quad (48)$$

The rationale of this approach is that when the randomized robust pricing outperforms deterministic robust pricing on the validation set, it is reasonable to expect that randomized robust pricing would outperform deterministic robust pricing on the test set; and conversely, when the deterministic approach outperforms the randomized approach on the validation set, one would expect this to also be true for the test set. We denote the hybrid method by H.

To compare the performance of all of the approaches, we compute a number of result metrics. For each store in the data set, the metric $\text{RI}_{\text{N} \rightarrow m}$, where $m \in \{\text{DR}, \text{RR}, \text{H}\}$, is defined as

$$\text{RI}_{\text{N} \rightarrow m} = \frac{R^{m,\text{test}} - R^{\text{N},\text{test}}}{R^{\text{N},\text{test}}} \times 100\%, \quad (49)$$

which represents the relative improvement of approach m over the nominal approach.

For each store, we also calculate the gap metric G_m for a given method $m \in \{\text{N}, \text{DR}, \text{RR}, \text{H}\}$ as

$$G_m = \frac{R^{\text{max},\text{test}} - R^{m,\text{test}}}{R^{\text{max},\text{test}}} \times 100\%, \quad (50)$$

where $R^{\text{max},\text{test}} = \max_{m' \in \{\text{N}, \text{DR}, \text{RR}, \text{H}\}} R^{m',\text{test}}$ is the best test set performance of the four methods.

Table 7 below displays the mean values of the $\text{RI}_{\text{N} \rightarrow m}$ and G_m metrics. In addition to the ordinary mean, we also report the trimmed mean with $\alpha = 0.05$ and $\alpha = 0.10$, where α denotes the fraction of observations to remove from the lower and upper tails (i.e., $\alpha = 0.05$ would mean removing the bottom 2.5% of values and the top 2.5% values, and taking the mean of the remaining middle 95% of values). The reason for considering the trimmed mean is that for some stores, both the DR and RR price prescriptions can exhibit large under or overperformance relative to the nominal pricing strategy. By removing the extreme values at the tails, we obtain a more representative measure of the performance.

From this table, we can see that for the semi-log demand model, the randomized robust approach leads to an improvement over the nominal approach, as the ordinary mean and the two trimmed means for $\text{RI}_{\text{N} \rightarrow \text{RR}}$ are all greater than zero. Considering the ordinary mean, the randomized robust approach leads to an average improvement over nominal of 4.65%; after removing the most extreme 5% and 10% of values, this average improvement is smaller but still large (1.76% for the trimmed mean with $\alpha = 0.05$ and 1.40% for the trimmed mean with $\alpha = 0.10$). Additionally, the randomized robust approach also leads to a greater improvement over nominal compared to the deterministic

Demand model	Type of mean	$RI_{N \rightarrow RR}$ (%)	$RI_{N \rightarrow DR}$ (%)	$RI_{N \rightarrow H}$ (%)	G_N (%)	G_{DR} (%)	G_{RR} (%)	G_H (%)
Semi-log	Mean	4.65	1.51	3.09	5.67	5.78	3.08	4.35
Semi-log	Trimmed Mean (0.05)	1.76	0.17	1.20	3.50	3.79	1.31	2.06
Semi-log	Trimmed Mean (0.10)	1.40	0.21	1.08	2.43	2.48	0.85	1.12
Log-log	Mean	11.88	14.31	16.61	13.29	8.06	8.25	6.52
Log-log	Trimmed Mean (0.05)	8.52	8.53	11.07	11.36	6.43	6.18	4.80
Log-log	Trimmed Mean (0.10)	5.37	3.80	6.57	9.44	5.18	4.56	3.73

Table 7 Relative improvement and gap metrics for the data-driven experiment in Section 7.3.

robust approach, as the mean and two trimmed means of $RI_{N \rightarrow RR}$ are always greater than the corresponding means for $RI_{N \rightarrow DR}$. In addition, the randomized robust approach is on average within about 3% of the performance of the best method on the test set, compared to about 6% for the nominal and deterministic robust approaches.

For the log-log demand model, the randomized robust approach still leads to an improvement over the nominal approach, as the ordinary mean of $RI_{N \rightarrow RR}$ is 11.88%. The deterministic robust approach, on the other hand, leads to a higher improvement over the nominal approach, with an ordinary mean of $RI_{N \rightarrow DR}$ of 14.31%. However, comparing the randomized robust and deterministic robust approaches using the two trimmed means of the RI metrics, we see that the trimmed mean with $\alpha = 0.05$ suggests that the two methods are similar (relative improvement of 8.52% and 8.53% for randomized and deterministic), whereas for $\alpha = 0.10$ the randomized robust approach is better (5.37% for randomized vs 3.80% for deterministic). These results suggest that while the deterministic robust approach can lead to very large gains over the nominal approach for some stores, leading to a large mean of the RI metric, the randomized robust approach does a better job of lifting the out-of-sample revenue for most stores and more uniformly improves on the nominal approach. In addition, when we consider the means of the gap metrics, we find that the deterministic and randomized robust approaches exhibit similar gap metrics. For example, using the ordinary mean, the deterministic robust approach is within 8.06% of the best prescription, whereas the randomized robust approach is within 8.25%, and both of these are better than the nominal approach (gap of 13.29%). Using the trimmed mean with $\alpha = 0.10$, the randomized robust approach is slightly better than the deterministic robust approach and the nominal approach (5.18% for deterministic robust and 9.44% for nominal versus 4.56% for randomized robust).

It is also interesting to consider the hybrid method described earlier. With regard to the semi-log demand model, we find that the hybrid method improves on the deterministic robust approach, but is not as good as the randomized robust approach. For example, when considering the trimmed mean with $\alpha = 0.05$, the hybrid method has an average improvement of 1.20% over nominal, which is higher than the deterministic robust approach (0.17%) but smaller than the randomized robust

approach (1.76%). However, for the log-log demand model, the hybrid method actually outperforms both the deterministic and randomized robust approaches. For example, when considering the trimmed mean with $\alpha = 0.05$, the relative improvement of the hybrid approach is 11.07% compared to 8.52% and 8.53% for the randomized robust and deterministic robust approaches respectively. The improvement over the pure robust approaches is also reflected in the gap metric, which is smaller for the hybrid method compared to the nominal, deterministic robust and randomized robust approaches, for all three types of means. These results suggest that from the perspective of out-of-sample performance, deterministic and randomized robust pricing can be synergistic and complementary. This contrasts with our theoretical results, which suggest that from the standpoint of worst-case performance, a randomized robust pricing strategy is always at least as good as a deterministic pricing strategy.

We conclude this section by commenting on the limitations of this type of experiment. First, the revenue estimates that one obtains using the function $\bar{R}(\cdot)$ are inherently noisy, because each price vector only appears a small number of times for each store. For each store, the number of weeks of data available ranges from 87 to 121 with a mean of 116.3, while the number of unique price vectors ranges from 86 to 121 with a mean of 115.6, which implies that most price vectors only appear once, and thus for most stores and most price vectors, the function $\bar{R}(\cdot)$ reflects the revenue of a single week in which a price vector was offered. Second, as noted earlier, the test phase of Algorithm 1 compares the randomized robust, deterministic robust and nominal approaches by considering a restricted pricing problem where one is restricted to those price vectors in $\mathcal{P}_{\text{test}}$. It is possible that by considering a different set of price vectors, one might see a different ordering of the methods.

Notwithstanding these limitations, we believe that a strength of the approach is that it is entirely data-driven. We believe that this approach could be generalized to other contexts where the data corresponds to different historical decisions and their (noisy) objective values, and where the ground truth is unknown and inaccessible to us. The application of this data-driven evaluation approach in other settings is an interesting direction for future research.

8. Conclusions

In this paper, we considered the problem of designing randomized robust pricing strategies to maximize worst-case revenue. We presented idealized conditions under which such randomized pricing strategies fare no better than the deterministic robust pricing approach, and subsequently we developed solution methods for obtaining the randomized pricing strategies in different settings (when the price set is finite, and when the uncertainty set is either convex or discrete). We showed using both synthetic instances and real data instances that such randomized pricing strategies can

lead to large improvements in worst-case revenue over deterministic robust price prescriptions, and using a data-driven experiment based on real data, we also demonstrate that randomized robust pricing can lead to large improvements in out-of-sample revenue over deterministic robust and non-robust pricing.

With regard to future research, there are several interesting directions to consider. First, a potential limitation of randomization in practice is that customers may engage in strategic behavior. For example, in a brick-and-mortar setting where prices are randomized by store or over time, strategic customers may travel to stores with lower prices, or they may wait for better prices before making their purchase. Although this type of customer behavior may not arise in every context (it may be costly for a customer to wait or to travel long distances to obtain a better price on a product), an interesting question is whether it is possible to design a randomized pricing strategy that accounts for this type of strategic behavior.

Second, it would also be interesting to consider a version of the robust price optimization problem that incorporates contextual information. For example, in the ecommerce setting, different customers who log onto a retailer’s website will differ in characteristics (age, web browser, operating system, etc.). This information could be used to craft a richer uncertainty set, and to motivate randomization strategies that randomize differently based on user characteristics. More generally, we hope that this work, which was inspired by the paper of Wang et al. (2024), motivates further study in how randomization can be used to mitigate risk in revenue management applications.

Acknowledgments

The authors thank the department editor Chung Piaw Teo, the associate editor and two anonymous reviewers for their comments and suggestions that have helped to significantly strengthen the paper.

References

- E. Adida and G. Perakis. Dynamic pricing and inventory control: robust vs. stochastic uncertainty models—a computational study. *Annals of Operations Research*, 181(1):125–157, 2010.
- A. Allouah, A. Bahamou, and O. Besbes. Optimal pricing with a single point. *arXiv preprint arXiv:2103.05611*, 2021.
- A. Allouah, A. Bahamou, and O. Besbes. Pricing with samples. *Operations Research*, 70(2):1088–1104, 2022.
- E. T. Anderson and D. I. Simester. Effects of \$9 price endings on retail sales: Evidence from field experiments. *Quantitative marketing and Economics*, 1:93–110, 2003.
- G. Aydin and J. K. Ryan. Product line selection and pricing under the multinomial logit choice model. In *Proceedings of the 2000 MSOM conference*. Citeseer, 2000.

- C. Bandi and D. Bertsimas. Robust option pricing. *European Journal of Operational Research*, 239(3): 842–853, 2014.
- A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88(3):411–424, 2000.
- A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust optimization*, volume 28. Princeton university press, 2009.
- F. Bernstein and A. Federgruen. Pricing and replenishment strategies in a distribution system with competing retailers. *Operations Research*, 51(3):409–426, 2003.
- D. Bertsimas and D. den Hertog. *Robust and adaptive optimization*. Dynamic Ideas LLC, 2022.
- D. Bertsimas and V. V. Mišić. Robust product line design. *Operations Research*, 65(1):19–37, 2017.
- D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.
- D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research*, 54(1):150–168, 2006.
- D. Bertsimas and J. N. Tsitsiklis. *Introduction to linear optimization*. Dynamic Ideas LLC, 1997.
- D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011.
- D. Bertsimas, I. Dunning, and M. Lubin. Reformulation versus cutting-planes for robust optimization: A computational study. *Computational Management Science*, 13:195–217, 2016a.
- D. Bertsimas, E. Nasrabadi, and J. B. Orlin. On the power of randomization in network interdiction. *Operations Research Letters*, 44(1):114–120, 2016b.
- J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.
- G. Y. Bizer and R. M. Schindler. Direct evidence of ending-digit drop-off in price information processing. *Psychology & Marketing*, 22(10):771–783, 2005.
- S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- G. Calafiore and M. C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102:25–46, 2005.
- M. Chen and Z.-L. Chen. Robust dynamic pricing with two substitutable products. *Manufacturing & Service Operations Management*, 20(2):249–268, 2018.
- M. C. Cohen, R. Lobel, and G. Perakis. Dynamic pricing through data sampling. *Production and Operations Management*, 27(6):1074–1088, 2018.
- E. Delage and A. Saif. The value of randomized solutions in mixed-integer distributionally robust optimization problems. *INFORMS Journal on Computing*, 34(1):333–353, 2022.

- E. Delage, D. Kuhn, and W. Wiesemann. “Dice”-sion-making under uncertainty: When can a random decision reduce risk? *Management Science*, 65(7):3282–3301, 2019.
- K. J. Ferreira, D. Simchi-Levi, and H. Wang. Online network revenue management using thompson sampling. *Operations Research*, 66(6):1586–1602, 2018.
- D. Fudenberg and J. Tirole. *Game theory*. MIT Press, 1991.
- V. Gabrel, C. Murat, and A. Thiele. Recent advances in robust optimization: An overview. *European Journal of Operational Research*, 235(3):471–483, 2014.
- G. Gallego and H. Topaloglu. *Revenue management and pricing analytics*, volume 209. Springer, 2019.
- G. Gallego and R. Wang. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Operations Research*, 62(2):450–461, 2014.
- A. Govindarajan, A. Sinha, and J. Uichanco. Distribution-free inventory risk pooling in a multilocation newsvendor. *Management Science*, 67(4):2272–2291, 2021.
- Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023. URL <https://www.gurobi.com>.
- M. Hamzeei, A. Lim, and J. Xu. Robust price optimization of multiple products under interval uncertainties. *Journal of Revenue and Pricing Management*, pages 1–13, 2021.
- W. Hanson and K. Martin. Optimizing multinomial logit profit functions. *Management Science*, 42(7):992–1003, 1996.
- P. Harsha, S. Subramanian, and J. Uichanco. Dynamic pricing of omnichannel inventories. *Manufacturing & Service Operations Management*, 21(1):47–65, 2019.
- W. J. Hopp and X. Xu. Product line selection and pricing with modularity in design. *Manufacturing & Service Operations Management*, 7(3):172–187, 2005.
- S. Jasin and S. Kumar. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research*, 37(2):313–345, 2012.
- K. Kalyanam. Pricing decisions under demand uncertainty: A bayesian mixture model approach. *Marketing Science*, 15(3):207–221, 1996.
- P. W. Keller. *Tractable multi-product pricing under discrete choice models*. PhD thesis, Massachusetts Institute of Technology, 2013.
- P. W. Keller, R. Levi, and G. Perakis. Efficient formulations for pricing under attraction demand models. *Mathematical Programming*, 145(1):223–261, 2014.
- H. Li and W. T. Huh. Pricing multiple products with the multinomial logit and nested logit models: Concavity and implications. *Manufacturing & Service Operations Management*, 13(4):549–563, 2011.
- A. Lim, J. G. Shanthikumar, and T. Watewai. Robust multi-product pricing. *Available at SSRN 1078012*, 2008.

- A. E. B. Lim and J. G. Shanthikumar. Relative entropy, exponential utility, and robust dynamic pricing. *Operations Research*, 55(2):198–214, 2007.
- M. Lubin and I. Dunning. Computing in operations research using Julia. *INFORMS Journal on Computing*, 27(2):238–248, 2015.
- T. Mai and P. Jaillet. Robust multi-product pricing under general extreme value models. *arXiv preprint arXiv:1912.09552*, 2019.
- A. Mastin, P. Jaillet, and S. Chin. Randomized minmax regret for combinatorial optimization under uncertainty. In *International Symposium on Algorithms and Computation*, pages 491–501. Springer, 2015.
- V. V. Mišić. Optimization of tree ensembles. *Operations Research*, 68(5):1605–1624, 2020.
- A. L. Montgomery. Creating micro-marketing pricing strategies using supermarket scanner data. *Marketing Science*, 16(4):315–337, 1997.
- A. L. Montgomery and E. T. Bradlow. Why analyst overconfidence about the functional form of demand models can lead to overpricing. *Marketing Science*, 18(4):569–583, 1999.
- Mosek ApS. *The MOSEK optimization toolbox for C manual. Version 10.0.*, 2022. URL <http://docs.mosek.com/10.0/capi/index.html>.
- G. Perakis and A. Sood. Competitive multi-period pricing for perishable products: A robust optimization approach. *Mathematical Programming*, 107(1):295–335, 2006.
- D. J. Reibstein and H. Gatignon. Optimal product line pricing: The influence of elasticities and cross-elasticities. *Journal of Marketing Research*, 21(3):259–267, 1984.
- R. T. Rockafellar. *Convex analysis*, volume 18. Princeton University Press, 1970.
- P. E. Rossi. *bayesm: Bayesian inference for marketing/micro-econometrics*, 2022. URL <http://CRAN.R-project.org/package=bayesm>. R package version 3.1-5.
- N. Rujeerapaiboon, D. Kuhn, and W. Wiesemann. Robust growth-optimal portfolios. *Management Science*, 62(7):2090–2109, 2016.
- P. Rusmevichientong and H. Topaloglu. Robust assortment optimization in revenue management under the multinomial logit choice model. *Operations Research*, 60(4):865–882, 2012.
- U. Sadana and E. Delage. The value of randomized strategies in distributionally robust risk-averse network interdiction problems. *INFORMS Journal on Computing*, 35(1):216–232, 2023.
- W. Soon. A review of multi-product pricing models. *Applied Mathematics and Computation*, 217(21):8149–8165, 2011.
- M. Stiving. Price-endings when prices signal quality. *Management Science*, 46(12):1617–1629, 2000.
- A. Strulov-Shlain. More than a penny’s worth: Left-digit bias and firm pricing. *The Review of Economic Studies*, 90(5):2612–2645, 2022.

- B. Sturt. A nonparametric algorithm for optimal stopping based on robust optimization. *Operations Research*, 71(5):1530–1557, 2023.
- B. Sturt. The value of robust assortment optimization under ranking-based choice models. *Management Science*, Forthcoming, 2024.
- A. Thiele. Multi-product pricing via robust optimisation. *Journal of Revenue and Pricing Management*, 8(1):67–80, 2009.
- Z. Wang, H. Peura, and W. Wiesemann. Randomized assortment optimization. *Operations Research*, 72(5):2042–2060, 2024.
- Z. Yan, K. Natarajan, C. P. Teo, and C. Cheng. A representative consumer model in data-driven multiproduct pricing optimization. *Management Science*, 68(8):5798–5827, 2022.
- M. J. Zenor. The profit benefits of category management. *Journal of Marketing Research*, 31(2):202–213, 1994.
- H. Zhang, P. Rusmevichientong, and H. Topaloglu. Multiproduct pricing under the generalized extreme value models with homogeneous price sensitivity parameters. *Operations Research*, 66(6):1559–1570, 2018.
- S. Zymmler, D. Kuhn, and B. Rustem. Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, 137:167–198, 2013.

EC.1. Omitted proofs

EC.1.1. Proof of Theorem 1

To prove this, we prove that $Z_{\text{DR}}^* \geq Z_{\text{RR}}^*$. For any $R \in \mathcal{R}$, and any distribution F supported on \mathcal{P} , we have

$$\int_{\mathcal{P}} R(\mathbf{p}) dF(\mathbf{p}) \leq R\left(\int_{\mathcal{P}} \mathbf{p} dF(\mathbf{p})\right), \quad (\text{EC.1})$$

which follows by Jensen's inequality and the concavity of R . This implies that for any $F \in \mathcal{F}$,

$$\inf_{R \in \mathcal{R}} \int_{\mathcal{P}} R(\mathbf{p}) dF(\mathbf{p}) \leq \inf_{R \in \mathcal{R}} R\left(\int_{\mathcal{P}} \mathbf{p} dF(\mathbf{p})\right). \quad (\text{EC.2})$$

Therefore, we have that

$$\begin{aligned} Z_{\text{RR}}^* &= \max_{F \in \mathcal{F}} \left\{ \inf_{R \in \mathcal{R}} \int_{\mathcal{P}} R(\mathbf{p}) dF(\mathbf{p}) \right\} \\ &\leq \max_{F \in \mathcal{F}} \left\{ \inf_{R \in \mathcal{R}} R\left(\int_{\mathcal{P}} \mathbf{p} dF(\mathbf{p})\right) \right\} \\ &\leq \max_{\mathbf{p} \in \mathcal{P}} \left\{ \inf_{R \in \mathcal{R}} R(\mathbf{p}) \right\} \\ &= Z_{\text{DR}}^*, \end{aligned}$$

where the second inequality follows because \mathcal{P} is assumed to be convex, and thus for any $F \in \mathcal{F}$, $\int_{\mathcal{P}} \mathbf{p} dF(\mathbf{p})$ is contained in \mathcal{P} . \square

EC.1.2. Proof of Theorem 2

We will establish this result by drawing a connection with game theory. Consider a two-player zero-sum game, where player 1 and 2 have strategy spaces \mathcal{P} and \mathcal{U} , respectively, and payoff functions g_1 and g_2 given by

$$\begin{aligned} g_1(\mathbf{p}, \mathbf{u}) &= R(\mathbf{p}, \mathbf{u}), \\ g_2(\mathbf{p}, \mathbf{u}) &= -R(\mathbf{p}, \mathbf{u}). \end{aligned}$$

Player 1 can choose the action $\mathbf{p} \in \mathcal{P}$ (a pure strategy) or some distribution F over \mathcal{P} (a mixed strategy), and player 2 chooses \mathbf{u} or some distribution Q over \mathcal{U} . Observe that by assumption, g_1 and g_2 are both continuous. Additionally, g_1 is quasiconcave in \mathbf{p} by assumption, and for g_2 , recall that the negative of a quasiconvex function is quasiconcave, so g_2 is also quasiconcave because $R(\mathbf{p}, \mathbf{u})$ is quasiconvex in \mathbf{u} . We now invoke the Debreu-Glicksberg-Fan theorem (see Theorem 1.2 in Fudenberg and Tirole 1991), which asserts that there exists a pure strategy Nash equilibrium for this game. In particular, there exists a $(\mathbf{p}^*, \mathbf{u}^*) \in \mathcal{P} \times \mathcal{U}$ such that

$$\begin{aligned} g_1(\mathbf{p}^*, \mathbf{u}^*) &\geq g_1(\mathbf{p}, \mathbf{u}^*), \quad \forall \mathbf{p} \in \mathcal{P}, \\ g_2(\mathbf{p}^*, \mathbf{u}^*) &\geq g_2(\mathbf{p}^*, \mathbf{u}), \quad \forall \mathbf{u} \in \mathcal{U}. \end{aligned}$$

Note that this is equivalent to $(\mathbf{p}^*, \mathbf{u}^*)$ being a saddle point of $R(\mathbf{p}, \mathbf{u})$:

$$R(\mathbf{p}, \mathbf{u}^*) \leq R(\mathbf{p}^*, \mathbf{u}^*) \leq R(\mathbf{p}^*, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}, \mathbf{u} \in \mathcal{U}. \quad (\text{EC.3})$$

This implies that

$$\begin{aligned} R(\mathbf{p}^*, \mathbf{u}^*) &\leq \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}^*, \mathbf{u}) \\ &\leq \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}), \end{aligned}$$

$$\begin{aligned} R(\mathbf{p}^*, \mathbf{u}^*) &\geq \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}^*) \\ &\geq \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}), \end{aligned}$$

which together imply that

$$\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) \leq \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}). \quad (\text{EC.4})$$

Observe now that

$$\begin{aligned} Z_{\text{RR}}^* &= \max_{F \in \mathcal{F}} \min_{\mathbf{u} \in \mathcal{U}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \\ &\leq \min_{\mathbf{u} \in \mathcal{U}} \max_{F \in \mathcal{F}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \\ &= \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) \\ &\leq \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) \\ &= Z_{\text{DR}}^*, \end{aligned}$$

and hence $Z_{\text{RR}}^* = Z_{\text{DR}}^*$. \square

EC.1.3. Proof of Theorem 3

To establish this result, observe that

$$\begin{aligned} &\inf_{Q \in \mathcal{Q}} \max_{\mathbf{p} \in \mathcal{P}} \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \\ &= \inf_{Q \in \mathcal{Q}} \max_{\pi \in \Delta_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \\ &= \max_{\pi \in \Delta_{\mathcal{P}}} \inf_{Q \in \mathcal{Q}} \sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \\ &= \max_{\pi \in \Delta_{\mathcal{P}}} \inf_{Q \in \mathcal{Q}} \int_{\mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \\ &= \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot R(\mathbf{p}, \mathbf{u}) \\ &= Z_{\text{RR}}^*. \end{aligned}$$

In the above, the steps are justified as follows. The first step follows because maximizing a function of \mathbf{p} over the finite set \mathcal{P} is the same as maximizing the expected value of that same function with respect to all probability mass functions $\boldsymbol{\pi}$ supported on \mathcal{P} .

The second step follows by Sion's minimax theorem, because the quantity $\sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u})$ is linear and therefore quasiconcave in $\boldsymbol{\pi}$ for a fixed Q , and is linear and therefore quasiconvex in Q for a fixed $\boldsymbol{\pi}$; additionally, the set $\Delta_{\mathcal{P}} = \{\boldsymbol{\pi} \in \mathbb{R}^{|\mathcal{P}|} \mid \mathbf{1}^T \boldsymbol{\pi} = 1, \boldsymbol{\pi} \geq \mathbf{0}\}$ is a compact convex set, and \mathcal{Q} is a convex set. Additionally, note that $\sum_{\mathbf{p} \in \mathcal{P}} \pi(\mathbf{p}) \cdot \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u})$ is clearly continuous in $\boldsymbol{\pi}$. It is also continuous in Q , because each term $\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u})$ is continuous in Q when \mathcal{Q} is endowed with the topology of weak convergence, and there are finitely many such terms.

The third step follows by the linearity of integration. The fourth step follows by the fact that \mathcal{Q} contains the Dirac delta distribution that places unit mass on \mathbf{u} , for every $\mathbf{u} \in \mathcal{U}$. The final step just follows from the definition of Z_{RR}^* .

With this result in hand, observe that the existence of a $Q \in \mathcal{Q}$ such that for all $\mathbf{p} \in \mathcal{P}$, $\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \leq Z_{\text{DR}}^*$ is equivalent to the existence of $Q \in \mathcal{Q}$ such that

$$\max_{\mathbf{p} \in \mathcal{P}} \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \leq Z_{\text{DR}}^*,$$

which is equivalent to

$$\inf_{Q \in \mathcal{Q}} \max_{\mathbf{p} \in \mathcal{P}} \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \leq Z_{\text{DR}}^*.$$

Since the left-hand side of this inequality is equal to Z_{RR}^* , the existence of the distribution $Q \in \mathcal{Q}$ as in the theorem statement is equivalent to $Z_{\text{RR}}^* \leq Z_{\text{DR}}^*$; since it is always the case that $Z_{\text{RR}}^* \geq Z_{\text{DR}}^*$, this is equivalent to the problem being randomization-proof. \square

EC.1.4. Proof of Corollary 1

Observe that since $\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) \leq \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ always holds, equation (21) is equivalent to

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) \geq \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}),$$

or equivalently,

$$Z_{\text{DR}}^* \geq \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}).$$

Observe that the condition $\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) \leq Z_{\text{DR}}^*$ is exactly equivalent to the condition that there exists a $\mathbf{u} \in \mathcal{U}$ such that for all $\mathbf{p} \in \mathcal{P}$, $R(\mathbf{p}, \mathbf{u}) \leq Z_{\text{DR}}^*$.

To connect this to Theorem 3, consider $Q = \delta_{\mathbf{u}}$, where $\delta_{\mathbf{u}}$ is the Dirac delta distribution centered at \mathbf{u} . For any $\mathbf{p} \in \mathcal{P}$, $R(\mathbf{p}, \mathbf{u}) = \int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}') dQ(\mathbf{u}')$. Thus, for this choice of Q , it is the case that for all $\mathbf{p} \in \mathcal{P}$, $\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}') dQ(\mathbf{u}') \leq Z_{\text{DR}}^*$. By Theorem 3, this is equivalent to randomization-proofness. Thus, it follows that the strong duality condition (21) implies that the RPO problem is randomization-proof. \square

EC.1.5. Proof of Corollary 2

Proof of a): Observe that by the definition of \mathbf{u}^* , we have that $R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*) \leq R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$ for all $\mathbf{u} \in \mathcal{U}$. By the hypothesis that \mathbf{p}_{DR}^* solves the nominal problem under \mathbf{u}^* , that is, $\mathbf{p}_{\text{DR}}^* \in \arg \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}^*)$, we also have that $R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*) \geq R(\mathbf{p}, \mathbf{u}^*)$ for all $\mathbf{p} \in \mathcal{P}$. This automatically implies that $(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*)$ is a saddle point of R , which implies that strong duality holds, i.e.,

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) = \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.5})$$

and thus that the problem is randomization-proof.

Proof of b): By using Theorem 3, we know that there exists a distribution Q such that for every $\mathbf{p} \in \mathcal{P}$,

$$\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) \leq Z_{\text{DR}}^*. \quad (\text{EC.6})$$

Recall that $Z_{\text{DR}}^* = \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}) = R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*)$. Since the minimizer of this problem is unique, we can examine the above condition when $\mathbf{p} = \mathbf{p}_{\text{DR}}^*$:

$$\int_{\mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}) dQ(\mathbf{u}) \leq \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}). \quad (\text{EC.7})$$

As a result, we must have that $Q = \delta_{\mathbf{u}^*}$, as any other Q would result in $\int_{\mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}) dQ(\mathbf{u})$ being strictly more than $R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*)$, due to the uniqueness of \mathbf{u}^* .

With this insight in hand, observe that for any $\mathbf{p} \in \mathcal{P}$, we have

$$\int_{\mathcal{U}} R(\mathbf{p}, \mathbf{u}) dQ(\mathbf{u}) = R(\mathbf{p}, \mathbf{u}^*).$$

Using the property of Q in (EC.7) at any arbitrary \mathbf{p} and the fact that $Z_{\text{DR}}^* = R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*)$, we have

$$R(\mathbf{p}, \mathbf{u}^*) \leq R(\mathbf{p}_{\text{DR}}^*, \mathbf{u}^*) \quad (\text{EC.8})$$

for all \mathbf{p} , which completes the proof. \square

EC.1.6. Equivalence of strong duality and randomization-proofness when \mathcal{P} is finite

In this section, we develop the following result, which provides a condition under which strong duality and randomization-proofness are equivalent.

PROPOSITION EC.1. *Suppose that \mathcal{P} is finite, $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} and \mathcal{U} is a convex set. Then the robust price optimization problem is randomization-proof if and only if strong duality holds.*

Proof: Observe that

$$\begin{aligned}
Z_{\text{RR}}^* &= \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \\
&= \min_{\mathbf{u} \in \mathcal{U}} \max_{\pi \in \Delta_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \\
&= \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}),
\end{aligned}$$

where the second equality follows by Sion's minimax theorem, which follows because (1) $\sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$ is linear (and hence continuous and quasiconcave) in π , and convex (and hence continuous and quasiconvex) in \mathbf{u} , (2) $\Delta_{\mathcal{P}}$ is a compact convex set, and (3) \mathcal{U} is a convex set; and the third equality follows because the inner maximum is attained at a deterministic distribution that places all mass on a single \mathbf{p} . Observe that this last min-max expression is an upper bound on $Z_{\text{DR}}^* = \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u})$ by weak duality, and thus that the problem is exactly randomization-proof if and only if strong duality holds. \square

EC.1.7. Proof of Proposition 1

Let (μ^*, \mathbf{p}^*) be an optimal solution of problem (36), which is guaranteed to exist because the objective function is continuous and the feasible set $\Delta_{[I]} \times \mathcal{P}$ is compact. Observe that the objective function in (36) can be re-arranged as

$$\begin{aligned}
&\max_{\mu \in \Delta_{[I]}, \mathbf{p} \in \mathcal{P}} \left\{ \sum_{i=1}^I \mu_i (\alpha_i + \log p_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\
&= \max_{\mu \in \Delta_{[I]}} \max_{\mathbf{p} \in \mathcal{P}} \left\{ \sum_{i=1}^I \mu_i \cdot \alpha_i + \sum_{i=1}^I \left[\mu_i \cdot (1 - \beta_i) + \sum_{j \neq i} \mu_j \gamma_{j,i} \right] \cdot \log p_i - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\
&= \max_{\mu \in \Delta_{[I]}} \left[\sum_{i=1}^I \mu_i \cdot \alpha_i + \sum_{i=1}^I \max_{p_i \in \mathcal{P}_i} \left\{ \left[\mu_i \cdot (1 - \beta_i) + \sum_{j \neq i} \mu_j \gamma_{j,i} \right] \cdot \log p_i \right\} - \sum_{i=1}^I \mu_i \log \mu_i \right]
\end{aligned}$$

where the first step follows by algebra, and the second by the separability of the objective in p_1, \dots, p_I and Assumption 1 (since the price set is a Cartesian product and the objective is separable, each product's price can be optimized independently). Consider now holding μ fixed, and optimizing over \mathbf{p} . With μ^* fixed, the optimal value of p_i for the above objective depends on the sign of $(1 - \beta_i + \sum_{j \neq i} \gamma_{j,i})$. If this coefficient is positive, then since $\log p_i$ is increasing in p_i , it is optimal to set $p'_i = \max \mathcal{P}_i$. If this coefficient is negative, then it is optimal to set $p'_i = \min \mathcal{P}_i$. It thus follows that if (μ^*, \mathbf{p}^*) is an optimal solution, then (μ^*, \mathbf{p}') is also an optimal solution, where $\mathbf{p}' \in \prod_{i=1}^I \{\min \mathcal{P}_i, \max \mathcal{P}_i\}$. \square

EC.1.8. Proof of Theorem 4

Let t_{floor} be any real number that satisfies

$$t_{\text{floor}} < \min_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}). \quad (\text{EC.9})$$

Note that the right-hand side is well-defined: $R(\mathbf{p}, \mathbf{u})$ is convex and therefore continuous in \mathbf{u} , and \mathcal{U} is a compact set, and hence $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u})$ is finite by the extreme value theorem. Since \mathcal{P} is finite, the quantity $\min_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u})$ is therefore finite.

Observe that the RRPO problem $\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u})$ is equivalent to

$$\text{RR}(\mathcal{P}): \quad \underset{t, \mathbf{u}}{\text{minimize}} \quad t \quad (\text{EC.10a})$$

$$\text{subject to} \quad t \geq R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}, \quad (\text{EC.10b})$$

$$t \geq t_{\text{floor}}, \quad (\text{EC.10c})$$

$$\mathbf{u} \in \mathcal{U}, \quad (\text{EC.10d})$$

Note that this formulation is almost the same as formulation (22) in Section 5.1, with the exception of the constraint $t \geq t_{\text{floor}}$; we shall discuss the reason for including this constraint momentarily, after problem (EC.11) is presented. Note that since t_{floor} by construction is always strictly less than $R(\mathbf{p}, \mathbf{u})$ for any $\mathbf{p} \in \mathcal{P}$ and any $\mathbf{u} \in \mathcal{U}$, this constraint is vacuous and cannot become active/binding (i.e., hold at equality).

Consider now the following generalization of the above problem, where we enforce the constraint $t \geq R(\mathbf{p}, \mathbf{u})$ at a specific set of price vectors, $\bar{\mathcal{P}} \subseteq \mathcal{P}$:

$$\text{RR}(\bar{\mathcal{P}}): \quad \underset{t, \mathbf{u}}{\text{minimize}} \quad t \quad (\text{EC.11a})$$

$$\text{subject to} \quad t \geq R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \bar{\mathcal{P}}, \quad (\text{EC.11b})$$

$$t \geq t_{\text{floor}}, \quad (\text{EC.11c})$$

$$\mathbf{u} \in \mathcal{U}, \quad (\text{EC.11d})$$

Let us denote the optimal value by $Z_{\text{RR}}^*(\bar{\mathcal{P}})$, and the problem itself with this set of constraints by $\text{RR}(\bar{\mathcal{P}})$. Similarly to the previous formulation, which is $\text{RR}(\mathcal{P})$ and has objective value $Z_{\text{RR}}^*(\mathcal{P})$, the constraint $t \geq t_{\text{floor}}$ in $\text{RR}(\bar{\mathcal{P}})$ cannot be active so long as $\bar{\mathcal{P}}$ is nonempty. Here, we note that the reason for having the constraint $t \geq t_{\text{floor}}$ is to avoid a pathological case where the problem becomes unbounded if $\bar{\mathcal{P}}$ is empty; the inclusion of this constraint facilitates our analysis of the constructive procedure (Algorithm 2) that we will discuss shortly (see, in particular, the proof of Lemma EC.2).

With these definitions, to now appreciate our high-level proof strategy, observe that if we could succeed in showing that $Z_{\text{RR}}^*(\mathcal{P}) = Z_{\text{RR}}^*(\mathcal{P}')$ for a small set of prices $\mathcal{P}' \subseteq \mathcal{P}$ that satisfies $|\mathcal{P}'| \leq$

$M + 1$, then a similar application of Sion's minimax theorem used to derive the convex program in Section 5.1 would allow us to show the following:

$$\max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.12})$$

$$= Z_{\text{RR}}^*(\mathcal{P}) \quad (\text{EC.13})$$

$$= Z_{\text{RR}}^*(\mathcal{P}') \quad (\text{EC.14})$$

$$= \min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}'} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.15})$$

$$= \min_{\mathbf{u} \in \mathcal{U}} \max_{\pi \in \Delta_{\mathcal{P}'}} \sum_{\mathbf{p} \in \mathcal{P}'} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.16})$$

$$= \max_{\pi \in \Delta_{\mathcal{P}'}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}'} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.17})$$

which would imply that there exists a price distribution $\tilde{\pi}$ over \mathcal{P}' that achieves the same objective as the optimal solution of the RRPO problem when one considers all price vectors. In what follows, we will constructively establish that such a set \mathcal{P}' exists.

Before proceeding, we first recall the concept of a support constraint from the paper of Calafiore and Campi (2005).

DEFINITION EC.1. Consider the convex optimization problem

$$\text{CONV} : \quad \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad \mathbf{c}^T \mathbf{x} \quad (\text{EC.18a})$$

$$\text{subject to} \quad \mathbf{x} \in \mathcal{X}_i, \quad \forall i \in [m], \quad (\text{EC.18b})$$

where m and n are positive integers, and each \mathcal{X}_i is a closed convex set. Consider the problem CONV_k , defined as

$$\text{CONV}_k : \quad \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad \mathbf{c}^T \mathbf{x} \quad (\text{EC.19a})$$

$$\text{subject to} \quad \mathbf{x} \in \mathcal{X}_i, \quad \forall i \in [m] \setminus \{k\}, \quad (\text{EC.19b})$$

which is the problem obtained from removing the k th constraint $\mathbf{x} \in \mathcal{X}_k$ from CONV. Let \mathbf{x}^* and \mathbf{x}_k^* be optimal solutions of CONV and CONV_k , respectively. We say that constraint k is a *support constraint*, if $\mathbf{c}^T \mathbf{x}^* > \mathbf{c}^T \mathbf{x}_k^*$, i.e., if removing constraint k from CONV causes a change to the optimal objective value of the problem.

We now recall a critical result about support constraints from Calafiore and Campi (2005).

THEOREM EC.1 (**Theorem 2 of Calafiore and Campi (2005)**). *The number of support constraints for problem CONV is at most n .*

In our case, note that each inequality $t \geq R(\mathbf{p}, \mathbf{u})$ in $\text{RR}(\bar{\mathcal{P}})$ can be equivalently regarded as the constraint $(t, \mathbf{u}) \in \{(t', \mathbf{u}') \in \mathbb{R}^{M+1} \mid t' \geq R(\mathbf{p}, \mathbf{u}')\}$, and the set in this latter definition is clearly a closed, convex set.

With a slight abuse of terminology, we shall say that \mathbf{p} is a *support constraint* for $\text{RR}(\bar{\mathcal{P}})$ if the constraint $(t, \mathbf{u}) \in \{(t', \mathbf{u}') \in \mathbb{R}^{M+1} \mid t' \geq R(\mathbf{p}, \mathbf{u}')\}$ is a support constraint. Note that if \mathbf{p} is a support constraint, then the convex inequality $t \geq R(\mathbf{p}, \mathbf{u})$ must hold at equality for any optimal solution of $\text{RR}(\bar{\mathcal{P}})$. (This is true, because if there were an optimal solution (t, \mathbf{u}) at which the constraint for \mathbf{p} was non-binding, then one could take the improved optimal solution (t', \mathbf{u}') of $\text{RR}(\bar{\mathcal{P}} \setminus \{\mathbf{p}\})$ and take a convex combination with (t, \mathbf{u}) to obtain a better solution to $\text{RR}(\bar{\mathcal{P}})$.)

Additionally, given an optimal solution (t^*, \mathbf{u}^*) , we shall say that \mathbf{p} is an *active non-support constraint* for $\text{RR}(\bar{\mathcal{P}})$ if the constraint $t \geq R(\mathbf{p}, \mathbf{u})$ is active at (t^*, \mathbf{u}^*) , i.e., if $t^* = R(\mathbf{p}, \mathbf{u}^*)$, and \mathbf{p} is not a support constraint of $\text{RR}(\bar{\mathcal{P}})$.

Lastly, given an optimal solution (t^*, \mathbf{u}^*) , we shall say that \mathbf{p} is a *non-active constraint* for $\text{RR}(\bar{\mathcal{P}})$ if the constraint $t \geq R(\mathbf{p}, \mathbf{u})$ is not active at (t^*, \mathbf{u}^*) , i.e., if $t^* > R(\mathbf{p}, \mathbf{u}^*)$.

With these definitions, let us consider the following procedure, which we denote as Algorithm 2. In words, this algorithm starts with all of the price vectors in \mathcal{P} , solves the randomized robust pricing problem with this set, and then decomposes the price vector set into sets corresponding to support constraints, active non-support constraints and non-active constraints. The non-active constraints are also discarded, and if there are any active non-support constraints, one price vector from that set of active non-support constraints is also discarded. The optimization problem is solved again with the remaining smaller set of constraints, and the remaining price vectors are then decomposed again. This procedure repeats until one has a set of price vectors for which all active constraints are support constraints, or equivalently, the set of active non-support constraints \mathcal{N} is empty.

We now establish a few key properties of this procedure.

LEMMA EC.1. *Algorithm 2 terminates in a finite number of iterations.*

Proof: Observe that after line 7 of Algorithm 2, $|\bar{\mathcal{P}}| = |\mathcal{N}| - 1 + |\mathcal{S}|$, whereas immediately before line 7, $|\bar{\mathcal{P}}| = |\mathcal{N}| + |\mathcal{S}| + |\mathcal{O}|$, which is strictly higher than $|\mathcal{N}| - 1 + |\mathcal{S}|$. Since the size of $\bar{\mathcal{P}}$ is decreasing by at least 1 in each iteration, the initial set \mathcal{P} is finite, and \mathcal{N} is always updated to be a subset of $\bar{\mathcal{P}}$, it must be that \mathcal{N} eventually becomes empty. \square

LEMMA EC.2. *The set $\bar{\mathcal{P}}$ is non-empty throughout the execution of Algorithm 2.*

Algorithm 2 Constructive procedure for reducing the price vector set \mathcal{P} .

-
- 1: Initialize $\bar{\mathcal{P}} = \mathcal{P}$.
 - 2: Solve $\text{RR}(\bar{\mathcal{P}})$ to obtain an optimal solution (t^*, \mathbf{u}^*) .
 - 3: Set $\mathcal{S} \subseteq \bar{\mathcal{P}}$ to be the set of support constraints for $\text{RR}(\bar{\mathcal{P}})$.
 - 4: Set $\mathcal{N} \subseteq \bar{\mathcal{P}}$ to be the set of active non-support constraints for $\text{RR}(\bar{\mathcal{P}})$ and (t^*, \mathbf{u}^*) .
 - 5: Set $\mathcal{O} \subseteq \bar{\mathcal{P}}$ to be the set of non-active constraints for $\text{RR}(\bar{\mathcal{P}})$ and (t^*, \mathbf{u}^*) .
 - 6: **while** $|\mathcal{N}| > 0$ **do**
 - 7: Select any $\mathbf{p} \in \mathcal{N}$, and set $\bar{\mathcal{P}} \leftarrow \mathcal{N} \setminus \{\mathbf{p}\} \cup \mathcal{S}$.
 - 8: Solve $\text{RR}(\bar{\mathcal{P}})$ to obtain an optimal solution (t^*, \mathbf{u}^*) .
 - 9: Set $\mathcal{S} \subseteq \bar{\mathcal{P}}$ to be the set of support constraints for $\text{RR}(\bar{\mathcal{P}})$.
 - 10: Set $\mathcal{N} \subseteq \bar{\mathcal{P}}$ to be the set of active non-support constraints for $\text{RR}(\bar{\mathcal{P}})$ and (t^*, \mathbf{u}^*) .
 - 11: Set $\mathcal{O} \subseteq \bar{\mathcal{P}}$ to be the set of non-active constraints for $\text{RR}(\bar{\mathcal{P}})$ and (t^*, \mathbf{u}^*) .
 - 12: **end while**
 - 13: **return** $\mathcal{S}^* \equiv \mathcal{S}$.
-

Proof: Suppose for the sake of contradiction that $\bar{\mathcal{P}}$ becomes empty. This would mean that in some iteration of the while loop, at line 7, one would have that $\mathcal{N} \setminus \{\mathbf{p}\} \cup \mathcal{S}$ is empty, which would imply that $\mathcal{N} \setminus \{\mathbf{p}\} = \emptyset$ and $\mathcal{S} = \emptyset$. Since \mathbf{p} is chosen from \mathcal{N} , this would imply that $\mathcal{N} = \{\mathbf{p}\}$. This means that prior to line 7, $\text{RR}(\bar{\mathcal{P}})$ has only one active constraint, which is at \mathbf{p} , and that constraint is not a support constraint. This, however, leads to a contradiction, because by removing this constraint, one would be free to set $t = t_{\text{floor}}$, and recalling that t_{floor} is strictly smaller than $R(\mathbf{p}, \mathbf{u})$ for all $\mathbf{u} \in \mathcal{U}$, this would mean we are able to improve the objective value. (In other words, \mathbf{p} has to be a support constraint.) \square

LEMMA EC.3. *Let \mathcal{S}^* be the output of Algorithm 2. Then \mathcal{S}^* is not empty, and $|\mathcal{S}^*| \leq M + 1$.*

Proof: Observe that Algorithm 2 terminates by Lemma EC.1, and that by Lemma EC.2, $\bar{\mathcal{P}}$ never becomes empty during Algorithm 2. Consider now what happens in lines 9 and 10 of Algorithm 2 immediately prior to termination. Since $\bar{\mathcal{P}}$ is not empty, then $\text{RR}(\bar{\mathcal{P}})$ must have at least one active constraint from among the constraints in $\bar{\mathcal{P}}$ (i.e., $\mathcal{S} \cup \mathcal{N}$ is not empty); since the algorithm terminates and the loop exits, \mathcal{N} must be empty. This implies that \mathcal{S}^* is not empty.

To see that $|\mathcal{S}^*| \leq M + 1$, observe that \mathcal{S}^* is the set of support constraints of $\text{RR}(\bar{\mathcal{P}})$ for some $\bar{\mathcal{P}} \subseteq \mathcal{P}$, which is a $(M + 1)$ -dimensional convex optimization problem; by Theorem 2 of Calafiore and Campi (2005), $|\mathcal{S}^*|$ can be at most $M + 1$. \square

LEMMA EC.4. *Let \mathcal{S}^* be the output of Algorithm 2. Then $\text{RR}(\mathcal{S}^*) = \text{RR}(\mathcal{P})$.*

Proof: We shall establish this result by induction. Let $\bar{\mathcal{P}} \subseteq \mathcal{P}$ be any non-empty set of price vectors, and let \mathcal{S} , \mathcal{N} and \mathcal{O} be defined as in lines 9, 10, and 11 respectively. Let (t^*, \mathbf{u}^*) be any optimal solution of $\text{RR}(\bar{\mathcal{P}})$. We claim that

$$Z_{\text{RR}}^*(\bar{\mathcal{P}}) \underset{(a)}{=} Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N}) \underset{(b)}{=} Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\}), \quad (\text{EC.20})$$

where \mathbf{p}' is any price vector in \mathcal{N} . We now establish each of the two equalities, (a) and (b).

Proof of (a). Observe that we trivially have $Z_{\text{RR}}^*(\bar{\mathcal{P}}) \geq Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N})$, because $\mathcal{S} \cup \mathcal{N}$ is a subset of $\bar{\mathcal{P}}$, and so the objective value of the minimization problem with $\mathcal{S} \cup \mathcal{N}$ is either equal or lower than with $\bar{\mathcal{P}}$.

To see why $Z_{\text{RR}}^*(\bar{\mathcal{P}}) > Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N})$ cannot hold, suppose for the sake of contradiction that this was the case, and let (t', \mathbf{u}') be any optimal solution of $\text{RR}(\mathcal{S} \cup \mathcal{N})$. Observe that by taking any convex combination of (t^*, \mathbf{u}^*) and (t', \mathbf{u}') , we would obtain a new solution $(\tilde{t}, \tilde{\mathbf{u}})$ that satisfies the constraint $t \geq R(\mathbf{p}, \mathbf{u})$ for all $\mathbf{p} \in \mathcal{S} \cup \mathcal{N}$, because the function $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} and both (t^*, \mathbf{u}^*) and (t', \mathbf{u}') satisfy those constraints. Note also that while (t', \mathbf{u}') might not necessarily satisfy $t \geq R(\mathbf{p}, \mathbf{u})$ for every $\mathbf{p} \in \mathcal{O} = \bar{\mathcal{P}} \setminus (\mathcal{S} \cup \mathcal{N})$, by a suitable choice of the weight on (t^*, \mathbf{u}^*) , the convex combination $(\tilde{t}, \tilde{\mathbf{u}})$ can be made to satisfy these because by assumption, $t^* > R(\mathbf{p}, \mathbf{u}^*)$ for all $\mathbf{p} \in \mathcal{O}$. (Because of the slack in these constraints for (t^*, \mathbf{u}^*) , and because there are finitely many of these constraints, we can choose the weight in the convex combination so that none of these constraints are violated in the new solution $(\tilde{t}, \tilde{\mathbf{u}})$.) Lastly, since the objective function is linear (in fact, it is exactly t), any convex combination of t^* and t' with a positive weight on t' would ensure that \tilde{t} is strictly smaller than t^* . Since we can construct a new solution $(\tilde{t}, \tilde{\mathbf{u}})$ that is feasible for $\text{RR}(\bar{\mathcal{P}})$ but has an objective value smaller than t^* , we reach a contradiction, because (t^*, \mathbf{u}^*) was assumed to be optimal for $\text{RR}(\bar{\mathcal{P}})$.

Proof of (b). We now establish the second equality. We shall do this by showing that $Z_{\text{RR}}^*(\bar{\mathcal{P}}) = Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\})$; by then using (a), we will have that $Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N}) = Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\})$.

As in the case of (a), we again trivially have that $Z_{\text{RR}}^*(\bar{\mathcal{P}}) \geq Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\})$, because $\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\} \subseteq \bar{\mathcal{P}}$.

To see why $Z_{\text{RR}}^*(\bar{\mathcal{P}}) > Z_{\text{RR}}^*(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\})$ cannot hold, let us suppose that it does, and let (t', \mathbf{u}') be any optimal solution of $\text{RR}(\mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\})$. We thus have $t^* > t'$. Now, consider a new solution $(\tilde{t}, \tilde{\mathbf{u}})$, which is obtained by taking a convex combination of (t^*, \mathbf{u}^*) and (t', \mathbf{u}') . Observe that any convex combination of (t^*, \mathbf{u}^*) and (t', \mathbf{u}') satisfies $t \geq R(\mathbf{p}, \mathbf{u})$ for all $\mathbf{p} \in \mathcal{S} \cup \mathcal{N} \setminus \{\mathbf{p}'\}$, because both

(t^*, \mathbf{u}^*) and (t', \mathbf{u}') have to satisfy these constraints and $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} . Observe also that because the constraints in \mathcal{O} are not binding at (t^*, \mathbf{u}^*) and \mathcal{O} is finite, we can choose the positive weight on (t', \mathbf{u}') in the convex combination so that $(\tilde{t}, \tilde{\mathbf{u}})$ satisfies $t \geq R(\mathbf{p}, \mathbf{u})$ for all $\mathbf{p} \in \mathcal{O}$ (this is similar to the argument in the case of equality (a)). Lastly, by choosing a suitable positive weight on (t', \mathbf{u}') , the new solution $(\tilde{t}, \tilde{\mathbf{u}})$ can also be made to have an objective value strictly lower than t^* . Thus, by choosing a suitable weight on (t', \mathbf{u}') , we are able to construct the new solution $(\tilde{t}, \tilde{\mathbf{u}})$ which is a feasible solution of $\text{RR}(\bar{\mathcal{P}} \setminus \{\mathbf{p}'\})$, and has objective value $\tilde{t} < t^* = Z_{\text{RR}}^*(\bar{\mathcal{P}})$. This implies that $Z_{\text{RR}}^*(\bar{\mathcal{P}} \setminus \{\mathbf{p}'\}) < Z_{\text{RR}}^*(\bar{\mathcal{P}})$, which immediately leads to a contradiction, because it implies that \mathbf{p}' is a support constraint of $\text{RR}(\bar{\mathcal{P}})$. (By dropping the \mathbf{p}' constraint from $\text{RR}(\bar{\mathcal{P}})$, we get $Z_{\text{RR}}^*(\bar{\mathcal{P}} \setminus \{\mathbf{p}'\})$ which we have shown to be lower than $Z_{\text{RR}}^*(\bar{\mathcal{P}})$.) This contradicts the assumption that \mathbf{p}' is from \mathcal{N} and not a support constraint.

Having established equation (EC.20), suppose that the loop in Algorithm 2 performs k iterations, and let $\bar{\mathcal{P}}_\ell$ be the set $\bar{\mathcal{P}}$ after line 7 is performed in the ℓ th iteration of the loop. Recall that \mathcal{S}^* is the output of Algorithm 2 in the hypothesis of the lemma. Then by applying equation (EC.20) inductively, we have that

$$Z_{\text{RR}}^*(\mathcal{P}) = Z_{\text{RR}}^*(\bar{\mathcal{P}}_1) = Z_{\text{RR}}^*(\bar{\mathcal{P}}_2) = Z_{\text{RR}}^*(\bar{\mathcal{P}}_3) = \cdots = Z_{\text{RR}}^*(\bar{\mathcal{P}}_k) = Z_{\text{RR}}^*(\mathcal{S}^*), \quad (\text{EC.21})$$

where each equality except for the last follows by using both (a) and (b) in (EC.20), while the last equality $Z_{\text{RR}}^*(\bar{\mathcal{P}}_k) = Z_{\text{RR}}^*(\mathcal{S}^*)$ follows by using (a) in (EC.20) together with the fact that $\mathcal{N} = \emptyset$ (this is the case because it is the final iteration and the loop is exited). \square

We now complete the proof of Theorem 4. To establish the result, we apply Algorithm 2. The resulting \mathcal{S}^* that we obtain is nonempty and has at most $M + 1$ elements (Lemma EC.3); moreover, we have that $Z_{\text{RR}}^*(\mathcal{S}^*) = Z_{\text{RR}}^*(\mathcal{P})$ (Lemma (EC.4)). By our earlier argument (equations (EC.12) - (EC.17)), this implies that

$$\max_{\pi \in \Delta_{\mathcal{S}^*}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{S}^*} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) = \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.22})$$

which establishes the theorem. \square

EC.2. Accompanying examples for Section 4

In this section, we provide a number of examples that accompany our discussion of our theoretical results in Section 4. Section EC.2.1 provides examples to illustrate that certain conditions in Theorem 1 cannot be relaxed, and an example to show that strong duality does not hold in the setting

of this theorem. Section EC.2.2 documents a number of examples of applications of Theorem 1. Section EC.2.3 documents a number of examples of applications of Theorem 2. Section EC.2.4 documents an example to show that uniqueness is necessary for part (b) of Corollary 2, and also to show that randomization-proofness can hold without strong duality holding when \mathcal{P} is finite. Lastly, Section EC.2.5 documents an example that illustrates that Z_{DR}^* , Z_{RR}^* and the dual bound $\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ can in general all be different.

EC.2.1. Accompanying examples for Theorem 1

As noted in Section 4.1, the condition that all functions in \mathcal{R} be concave cannot be relaxed, which we illustrate using the following example.

EXAMPLE EC.1. Consider a single-product RPO problem, and suppose that the revenue function uncertainty set $\mathcal{R} = \{R_1, R_2\}$, where $R_1(\cdot)$ and $R_2(\cdot)$ are defined as

$$\begin{aligned} R_1(p) &= p(10 - 2p), \\ R_2(p) &= p \cdot 10p^{-2}. \end{aligned}$$

Note that $R_1(p)$ is the revenue function corresponding to the linear demand function $d_1(p) = 10 - 2p$, while $R_2(p)$ is the revenue function corresponding to the log-log demand function $d_2(p) = \exp(\log(10) - 2\log(p)) = 10p^{-2}$. Note also that $R_1(\cdot)$ is concave, while $R_2(\cdot)$ is convex. Suppose additionally that $\mathcal{P} = [1, 4]$.

We first calculate the optimal value of the DRPO problem. Observe that in the interval $[1, 4]$, the only root of the equation $10 - 2p = 10p^{-2}$ is $p \approx p' = 1.137805\dots$. For $p < p'$, $d_2(p) > d_1(p)$, and for $p > p'$, $d_1(p) > d_2(p)$. Therefore, the optimal value of the of the DRPO problem can be calculated as

$$\begin{aligned} & \max_{p \in [1, 4]} \min_{R \in \mathcal{R}} R(p) \\ &= \max\left\{ \max_{p \in [1, p']} \min_{R \in \mathcal{R}} R(p), \max_{p \in [p', 4]} \min_{R \in \mathcal{R}} R(p) \right\} \\ &= \max\left\{ \max_{p \in [1, p']} p \cdot (10 - 2p), \max_{p \in [p', 4]} p \cdot 10p^{-2} \right\} \\ &= 10p' - 2p'^2 \\ &= 8.78885 \end{aligned}$$

In the above, the first step follows because the best value of the worst-case revenue over $[1, 4]$ is equivalent to taking the higher of the best worst-case revenue over either $[1, p']$ or $[p', 4]$. The second step follows because for every $p \in [1, p']$, $d_1(p) < d_2(p)$, and so $R_1(p) = p \cdot d_1(p) < p \cdot d_2(p) = R_2(p)$; similarly, for every $p \in [p', 4]$, $d_1(p) > d_2(p)$, and so $R_2(p) < R_1(p)$. The third step follows by carrying out the maximization of each of the two functions from the prior step over its corresponding interval.

Now, let us lower bound the optimal value of the RRPO problem. Consider a distribution F that randomizes over prices in the following way:

$$p = \begin{cases} 1 & \text{with probability } 17/21, \\ 2.5 & \text{with probability } 4/21. \end{cases} \quad (\text{EC.23})$$

The worst-case revenue for this distribution is

$$\begin{aligned} & \min_{R \in \mathcal{R}} \int_1^4 R(p) dF(p) \\ &= \min \left\{ \frac{17}{21} \cdot R_1(1) + \frac{4}{21} \cdot R_1(2.5), \frac{17}{21} \cdot R_2(1) + \frac{4}{21} \cdot R_2(2.5) \right\} \\ &= \min \left\{ \frac{17}{21} \cdot 8 + \frac{4}{21} \cdot 12.5, \frac{17}{21} \cdot 10 + \frac{4}{21} \cdot 4 \right\} \\ &= \min \left\{ \frac{62}{7}, \frac{62}{7} \right\} \\ &= \frac{62}{7} \\ &= 8.857143. \end{aligned}$$

This implies that $Z_{\text{RR}}^* \geq 8.857143$, whereas $Z_{\text{DR}}^* = 8.78885$, and thus $Z_{\text{RR}}^* > Z_{\text{DR}}^*$. \square

The condition that \mathcal{P} be a convex set also cannot be relaxed in general, which we illustrate using the next example.

EXAMPLE EC.2. Consider again a single-product RPO problem. Suppose that $\mathcal{R} = \{R_1, R_2\}$, where $R_1(p) = p(10 - p)$, $R_2(p) = p(4 - 0.2p)$; R_1 and R_2 correspond to linear demand functions $d_1(p) = 10 - p$, $d_2(p) = 4 - 0.2p$. Suppose that $\mathcal{P} = \{p_1, p_2\}$, where $p_1 = 5$, $p_2 = 10$. From this data, observe that:

$$\begin{aligned} R_1(p_1) &= 5(10 - 5) = 25, \\ R_1(p_2) &= 10(10 - 10) = 0, \\ R_2(p_1) &= 5(4 - 0.2(5)) = 15, \\ R_2(p_2) &= 10(4 - 0.2(10)) = 20. \end{aligned}$$

We first calculate the optimal value of the DRPO problem:

$$\begin{aligned} Z_{\text{DR}}^* &= \max_{p \in \{p_1, p_2\}} \min\{R_1(p), R_2(p)\} \\ &= \max\{\min\{25, 15\}, \min\{0, 20\}\} \\ &= \max\{15, 0\} \\ &= 15. \end{aligned}$$

For the RRPO problem, the optimal value is given by the following LP:

$$\underset{\eta, \pi}{\text{maximize}} \quad \eta \tag{EC.24a}$$

$$\text{subject to} \quad \eta \leq \pi_{p_1} \cdot p_1 \cdot (10 - p_1) + \pi_{p_2} \cdot p_2 \cdot (10 - p_2) \tag{EC.24b}$$

$$\eta \leq \pi_{p_1} \cdot p_1 \cdot (4 - 0.2p_1) + \pi_{p_2} \cdot p_2 \cdot (4 - 0.2p_2) \tag{EC.24c}$$

$$\pi_{p_1} + \pi_{p_2} = 1 \tag{EC.24d}$$

$$\pi_{p_1}, \pi_{p_2} \geq 0. \tag{EC.24e}$$

The optimal distribution over $\mathcal{P} = \{p_1, p_2\}$ is given by $\pi_{p_1} = 2/3$, $\pi_{p_2} = 1/3$, which leads to $Z_{\text{RR}}^* = 50/3 = 16.6667$. Since this is higher than Z_{DR}^* , we conclude that this particular instance is randomization-receptive. \square

Lastly, as noted in our discussion after Theorem 1, this result is not being driven by strong duality. The next example illustrates that a RPO problem can be randomization-proof without strong duality holding. (Note that this example is the same as our previous example, Example EC.2, with the price set $\{5, 10\}$ replaced by its convex hull, $[5, 10]$.)

EXAMPLE EC.3. Consider a single-product RPO problem, where:

$$\mathcal{P} = [5, 10], \tag{EC.25}$$

$$R_1(p) = p(10 - p), \tag{EC.26}$$

$$R_2(p) = p(4 - 0.2p). \tag{EC.27}$$

Observe that \mathcal{P} is convex and both R_1 and R_2 are concave quadratic functions, and so by Theorem 1, the problem is randomization-proof. However, observe that

$$\begin{aligned} & \max_{p \in [5, 10]} \min\{R_1(p), R_2(p)\} \\ &= 18.75, \end{aligned}$$

whereas

$$\begin{aligned} & \min\left\{ \max_{p \in [5, 10]} R_1(p), \max_{p \in [5, 10]} R_2(p) \right\} \\ &= \min\{25, 20\} \\ &= 20, \end{aligned}$$

which exactly implies that $\max_{\mathbf{p} \in \mathcal{P}} \min_{R \in \mathcal{R}} R(\mathbf{p}) \neq \min_{R \in \mathcal{R}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p})$. \square

EC.2.2. Applications of Theorem 1

Theorem 1 allows us to assert that a number of RPO problems are randomization-proof under certain conditions, which we illustrate through the following four examples.

EXAMPLE EC.4. (Single-product pricing under linear demand). Suppose that $I = 1$, which corresponds to a single-product pricing problem. Let $\mathbf{u} = (\alpha, \beta) \in \mathbb{R}^2$ denote the vector of linear demand model parameters, and let $\mathcal{U} \subseteq \mathbb{R}^2$ be an uncertainty set of possible values of (α, β) . Let $\mathcal{R} = \{R(\cdot, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}$ be the set of revenue functions that arise from the uncertainty set \mathcal{U} . Note that each revenue function is of the form $R(p) = \alpha p - \beta p^2$. Therefore, the condition that each $R \in \mathcal{R}$ is concave implies that $R''(p) = -2\beta \leq 0$. Thus, if \mathcal{U} is such that $\inf\{\beta \mid (\alpha, \beta) \in \mathcal{U}\} \geq 0$, then the robust price optimization problem is randomization-proof. \square

EXAMPLE EC.5. (Multi-product pricing under linear demand). In the more general multi-product pricing problem, let $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in \mathbb{R}^I \times \mathbb{R}^I \times \mathbb{R}^{I(I-1)}$ denote the vector of linear demand model parameters, and let \mathcal{U} be an arbitrary uncertainty set of these model parameter vectors. Let $\mathcal{R} = \{R(\cdot, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}$ be the set of revenue functions that arise from the uncertainty set \mathcal{U} . Observe that each revenue function $R(\cdot, \mathbf{u})$ is of the form

$$\begin{aligned} R(\mathbf{p}) &= \sum_{i=1}^I p_i (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) \\ &= \boldsymbol{\alpha}^T \mathbf{p} - \mathbf{p}^T \mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}} \mathbf{p}, \end{aligned}$$

where $\mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}}$ is the matrix

$$\mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}} = \begin{bmatrix} \beta_1 & -\gamma_{1,2} & -\gamma_{1,3} & \cdots & -\gamma_{1,I-1} & -\gamma_{1,I} \\ -\gamma_{2,1} & \beta_2 & -\gamma_{2,3} & \cdots & -\gamma_{2,I-1} & -\gamma_{2,I} \\ \vdots & \vdots & \ddots & & \vdots & \vdots \\ -\gamma_{I,1} & -\gamma_{I,2} & -\gamma_{I,3} & \cdots & -\gamma_{I,I-1} & \beta_I \end{bmatrix}.$$

This implies that

$$\nabla^2 R(\mathbf{p}) = -2\mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}}.$$

The function R is therefore concave if the matrix $\mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}}$ is positive semidefinite. Therefore, if \mathcal{U} is such that $\inf\{\lambda_{\min}(\mathbf{M}_{\boldsymbol{\beta}, \boldsymbol{\gamma}}) \mid (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in \mathcal{U}\} \geq 0$, where $\lambda_{\min}(\mathbf{A})$ denotes the minimum eigenvalue of a symmetric matrix \mathbf{A} , then the robust price optimization problem is randomization proof. \square

EXAMPLE EC.6. (Single-product pricing under semi-log demand). For the single-product pricing problem under semi-log demand, $d(p) = \exp(\alpha - \beta p)$ is the demand function given the parameter

vector $\mathbf{u} = (\alpha, \beta)$. Let $\mathcal{U} \subseteq \mathbb{R}^2$ be an uncertainty set of possible values of (α, β) , and assume that β is bounded away from zero, that is, $\inf\{\beta \mid (\alpha, \beta) \in \mathcal{U}\} \geq 0$. $\mathcal{R} = \{R(\cdot, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}$ be the revenue function uncertainty set. For a given $R \in \mathcal{R}$, its second derivative is $R''(p) = \beta(\beta p - 2)e^{\alpha - \beta p}$. Thus, for $R''(p)$ to be nonpositive, we need $\beta p - 2 \leq 0$ or equivalently $\beta p \leq 2$ (since β is assumed to be nonnegative) for all $p \in \mathcal{P}$ in order for $R(p)$ to be concave. Thus, if $\sup_{p \in \mathcal{P}} \sup_{(\alpha, \beta) \in \mathcal{U}} \{\beta p\} \leq 2$ and $\inf\{\beta \mid (\alpha, \beta) \in \mathcal{U}\} \geq 0$, then the RPO problem is randomization-proof. \square

EXAMPLE EC.7. (Single-product pricing under log-log demand). For the single-product pricing problem under log-log demand, $d(p) = \exp(\alpha - \beta \log p) = e^\alpha \cdot p^{-\beta}$ is the demand function and $\mathbf{u} = (\alpha, \beta) \in \mathbb{R}^2$ is the vector of uncertain demand model parameters. Let $\mathcal{U} \subseteq \mathbb{R}^2$ be an uncertainty set of possible values of (α, β) , and assume that β is bounded away from zero from below, that is, $\inf\{\beta \mid (\alpha, \beta) \in \mathcal{U}\} \geq 0$. Let $\mathcal{R} = \{R(\cdot, \mathbf{u}) \mid \mathbf{u} \in \mathcal{U}\}$ be the revenue function uncertainty set. For a given $R \in \mathcal{R}$, its second derivative is $R''(p) = e^\alpha \cdot (\beta - 1)(\beta) \cdot p^{-\beta-1}$. Thus, for $R''(p)$ to be nonpositive, we need $\beta - 1 \leq 0$, or equivalently $\beta \leq 1$. Thus, if $\sup_{(\alpha, \beta) \in \mathcal{U}} \beta \leq 1$ and $\inf_{(\alpha, \beta) \in \mathcal{U}} \beta \geq 0$, then the RPO problem is randomization-proof. \square

EC.2.3. Applications of Theorem 2

Theorem 2 allows us to establish that a single product RPO problem under semi-log and log-log demand will be randomization-proof, even when the revenue function is not necessarily concave (and one cannot rely on Theorem 1).

EXAMPLE EC.8. Consider a single-product price optimization problem where the demand follows a semi-log model. The uncertain parameter is therefore $\mathbf{u} = (\alpha, \beta)$.

Observe that $R(p, \mathbf{u}) = pe^{\alpha - \beta p}$ is convex in \mathbf{u} . Thus, it is also quasiconvex in \mathbf{u} for a fixed p . Note also that the function R is quasi-concave in p . To see this, observe that $\log R(p, \mathbf{u}) = \log p + \alpha - \beta p$, which is concave in p ; this means that R is log-concave in p . Since any log-concave function is quasiconcave, it follows that R is quasiconcave in p .

Thus, if $\mathcal{P} \subseteq \mathbb{R}$ and $\mathcal{U} \subseteq \mathbb{R}^2$ are compact and convex, then Theorem 2 asserts that the RPO problem is randomization-proof. \square

EXAMPLE EC.9. Consider a single-product price optimization problem where the demand follows a log-log model. The uncertain parameter is $\mathbf{u} = (\alpha, \beta)$, and $R(p, \mathbf{u}) = pe^{\alpha - \beta \log p}$. Assume that $\mathcal{P} \subseteq \mathbb{R}$ is a compact convex set, and that $\min\{p \mid p \in \mathcal{P}\} > 0$.

Observe that $R(p, \mathbf{u}) = p \cdot e^{\alpha - \beta \log p}$ is convex in \mathbf{u} , and therefore quasiconvex in \mathbf{u} for a fixed p . Additionally, with regard to quasiconcavity in p , observe that $\log R(p, \mathbf{u}) = \log p + \alpha - \beta \log p =$

$(1 - \beta) \log p + \alpha$, which means that R is log-concave in p whenever $1 - \beta > 0$ or equivalently $\beta < 1$. Therefore, R will also be quasiconcave whenever $\beta < 1$.

Thus, if $\mathcal{P} \subseteq \mathbb{R}$ and $\mathcal{U} \subseteq \mathbb{R}^2$ are compact and convex, and $\max\{\beta \mid (\alpha, \beta) \in \mathcal{U}\} < 1$, then Theorem 2 guarantees that the RPO problem is randomization-proof. \square

EC.2.4. Example of necessity of uniqueness assumption in part (b) of Corollary 2

The following example illustrates that the implication in part (b) of Corollary 2 may fail to hold if \mathbf{u}^* is not the unique solution of $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$. Additionally, this example also illustrates that even when \mathcal{P} is finite, the RPO problem can be randomization-proof in the absence of strong duality.

EXAMPLE EC.10. Consider a single product pricing instance, i.e., $I = 1$, which we define as follows. Let $\mathcal{P} = \{p_1, p_2, p_3\}$ where $p_1 = 5$, $p_2 = 8$, $p_3 = 9$. Let the demand model d be a linear demand model, so that the uncertain parameter $\mathbf{u} = (\alpha, \beta)$ and $d(p, \mathbf{u}) = \alpha - \beta p$. Finally, let $\mathcal{U} = \{(\alpha_1, \beta_1), (\alpha_2, \beta_2), (\alpha_3, \beta_3)\}$, where $(\alpha_1, \beta_1) = (10, 1)$, $(\alpha_2, \beta_2) = (3, 0.1)$, $(\alpha_3, \beta_3) = (3.6, 0.2)$.

We first calculate $\min_{\mathbf{u} \in \mathcal{U}} R(p, \mathbf{u})$ for each $p \in \mathcal{P}$. We have:

- For $p_1 = 5$: $p_1(\alpha_2 - \beta_2 p_1) = 12.5 < p_1(\alpha_3 - \beta_3 p_1) = 13 < p_1(\alpha_1 - \beta_1 p_1) = 25$. Hence, $\min_{\mathbf{u} \in \mathcal{U}} R(p_1, \mathbf{u}) = \min\{12.5, 13, 25\} = 12.5$.
- For $p_2 = 8$: $p_2(\alpha_1 - \beta_1 p_2) = p_2(\alpha_3 - \beta_3 p_2) = 16 < p_2(\alpha_2 - \beta_2 p_2) = 17.6$. Hence, $\min_{\mathbf{u} \in \mathcal{U}} R(p_2, \mathbf{u}) = \min\{16, 16, 17.6\} = 16$, and note also that the minimizing \mathbf{u} is not unique (the minimum is attained at both (α_1, β_1) and (α_3, β_3)).
- For $p_3 = 9$: $p_3(\alpha_1 - \beta_1 p_3) = 9 < p_3(\alpha_3 - \beta_3 p_3) = 16.2 < p_3(\alpha_2 - \beta_2 p_3) = 18.9$. Hence, $\min_{\mathbf{u} \in \mathcal{U}} R(p_3, \mathbf{u}) = \min\{9, 16.2, 18.9\} = 9$.

From this, we can see that the optimal deterministic robust price is $p_{\text{DR}}^* = p_2 = 8$ and the optimal deterministic robust objective value is $Z_{\text{DR}}^* = 16$. At $p = 8$, we can see that $\arg \min_{\mathbf{u} \in \mathcal{U}} R(p_2, \mathbf{u}) = \{(\alpha_1, \beta_1), (\alpha_3, \beta_3)\}$.

Let us now consider the RRPO problem. When we write the problem $\max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{p \in \mathcal{P}} \pi_p R(\mathbf{p}, \mathbf{u})$ as a linear program, we get the following problem:

$$\begin{aligned} & \underset{\pi, t}{\text{maximize}} && t \end{aligned} \tag{EC.28a}$$

$$\text{subject to} \quad t \leq \pi_{p_1} \cdot p_1(\alpha_1 - \beta_1 p_1) + \pi_{p_2} \cdot p_2(\alpha_1 - \beta_1 p_2) + \pi_{p_3} \cdot p_3(\alpha_1 - \beta_1 p_3), \tag{EC.28b}$$

$$t \leq \pi_{p_1} \cdot p_1(\alpha_2 - \beta_2 p_1) + \pi_{p_2} \cdot p_2(\alpha_2 - \beta_2 p_2) + \pi_{p_3} \cdot p_3(\alpha_2 - \beta_2 p_3), \tag{EC.28c}$$

$$t \leq \pi_{p_1} \cdot p_1(\alpha_3 - \beta_3 p_1) + \pi_{p_2} \cdot p_2(\alpha_3 - \beta_3 p_2) + \pi_{p_3} \cdot p_3(\alpha_3 - \beta_3 p_3), \tag{EC.28d}$$

$$\pi_{p_1} + \pi_{p_2} + \pi_{p_3} = 1, \tag{EC.28e}$$

$$\pi_{p_1}, \pi_{p_2}, \pi_{p_3} \geq 0, \tag{EC.28f}$$

or equivalently,

$$\begin{aligned} \underset{\pi, t}{\text{maximize}} \quad & t \end{aligned} \tag{EC.29a}$$

$$\text{subject to} \quad t \leq 25\pi_{p_1} + 16\pi_{p_2} + 9\pi_{p_3} \tag{EC.29b}$$

$$t \leq 12.5\pi_{p_1} + 17.6\pi_{p_2} + 18.9\pi_{p_3}, \tag{EC.29c}$$

$$t \leq 13\pi_{p_1} + 16\pi_{p_2} + 16.2\pi_{p_3}, \tag{EC.29d}$$

$$\pi_{p_1} + \pi_{p_2} + \pi_{p_3} = 1, \tag{EC.29e}$$

$$\pi_{p_1}, \pi_{p_2}, \pi_{p_3} \geq 0, \tag{EC.29f}$$

for which the optimal objective value is $Z_{\text{RR}}^* = 16$, which is the same as Z_{DR}^* . Thus, we can see that if the problem is randomization-proof, but the uniqueness condition on $\min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}_{\text{DR}}^*, \mathbf{u})$ is not met, then p_{DR}^* is not necessarily a maximizer of the nominal problem at any \mathbf{u}^* . (For $\mathbf{u}^* = (\alpha_1, \beta_1)$, the p that maximizes $R(\cdot, \mathbf{u}^*)$ is p_1 , but for $\mathbf{u}^* = (\alpha_3, \beta_3)$, the p that maximizes $R(\cdot, \mathbf{u}^*)$ is p_3 .)

Additionally, observe that in this instance, strong duality does not hold. If we compute $\min_{\mathbf{u} \in \mathcal{U}} \max_{p \in \mathcal{P}} R(p, \mathbf{u})$, we get:

$$\begin{aligned} & \min_{\mathbf{u} \in \mathcal{U}} \max_{p \in \mathcal{P}} R(p, \mathbf{u}) \\ &= \min \left\{ \max_{p \in \mathcal{P}} R(p, \mathbf{u}_1), \max_{p \in \mathcal{P}} R(p, \mathbf{u}_2), \max_{p \in \mathcal{P}} R(p, \mathbf{u}_3) \right\} \\ &= \min \left\{ \max\{25, 16, 9\}, \max\{12.5, 17.6, 18.9\}, \max\{13, 16, 16.2\} \right\} \\ &= \min\{25, 18.9, 16.2\} \\ &= 16.2, \end{aligned}$$

which is strictly higher than $Z_{\text{RR}}^* = 16$. \square

EC.2.5. Example to illustrate that Z_{DR}^* , Z_{RR}^* and dual objective can be distinct when \mathcal{P} is finite

In this section, we revisit Example EC.2 from Section EC.2.1, which can be viewed as an instance of a finite \mathcal{P} robust price optimization problem.

EXAMPLE EC.11. Consider a single-product RPO problem, and suppose that $\mathcal{P} = \{p_1, p_2\}$ where $p_1 = 5$, $p_2 = 10$. Consider $\mathbf{u} = (u_1)$, where $\mathcal{U} = \{0, 1\}$, and let $R(\mathbf{p}, \mathbf{u})$ be defined as

$$R(p, \mathbf{u}) = u_1 \cdot p(10 - p) + (1 - u_1) \cdot p(4 - 0.2p). \tag{EC.30}$$

Note that this is the same instance as Example EC.2, with the only difference being that we have re-defined the uncertain parameter \mathbf{u} to be a binary variable that toggles between the two revenue

functions $p(10 - p)$ and $p(4 - 0.2p)$. For this instance, $Z_{\text{DR}}^* = 15$ and $Z_{\text{RR}}^* = 50/3 = 16.6667$, as before, and the dual objective $\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ is

$$\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) = \min\{\max\{25, 0\}, \max\{15, 20\}\} \quad (\text{EC.31})$$

$$= \min\{25, 20\} \quad (\text{EC.32})$$

$$= 20, \quad (\text{EC.33})$$

which is strictly larger than Z_{RR}^* . \square

Note that in the previous example, the difference between $\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$ and Z_{RR}^* is driven by \mathcal{U} being a discrete set. If we replaced \mathcal{U} with its convex hull, then Z_{RR}^* and the dual objective would be identical, which is consistent with Proposition EC.1.6 (when \mathcal{U} is convex and $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} , then Z_{RR}^* and the dual objective are equal, and strong duality is equivalent to randomization-proofness).

EXAMPLE EC.12. Consider the same single-product RPO problem as in the preceding example, but let $\mathcal{U} = [0, 1]$. In this case, both Z_{RR}^* and Z_{DR}^* are unchanged because the objective of the inner minimization over $\mathbf{u} \in \mathcal{U}$ is unchanged when we replace $\mathcal{U} = \{0, 1\}$ with $\mathcal{U} = [0, 1]$. For the dual objective, we get

$$\min_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) = \min_{0 \leq u_1 \leq 1} \max\{u_1 \cdot 25 + (1 - u_1)15, (1 - u_1) \cdot 20\} \quad (\text{EC.34})$$

$$= \min_{0 \leq u_1 \leq 1} \{10u_1 + 15, 20 - 20u_1\} \quad (\text{EC.35})$$

$$= 50/3, \quad (\text{EC.36})$$

which is the same as Z_{RR}^* , as predicted by Proposition EC.1.6. \square

EC.3. Mixed-integer exponential cone formulations for Section 5

In this section, we describe how the two mixed-integer convex programs in Section 5, for the semi-log demand model (problem (29) in Section 5.3) and the log-log demand model (problem (37) in Section 5.4) can be reformulated as mixed-integer exponential cone programs.

We first recall that the standard exponential cone is defined as

$$\mathcal{K}_{\text{exp}} = \{(r, 0, t) \in \mathbb{R}^3 \mid r \geq 0, t \leq 0\} \cup \{(r, s, t) \in \mathbb{R}^3 \mid s > 0, r \geq s \exp(t/s)\}. \quad (\text{EC.37})$$

We also recall that the function $f(\mu) = \mu \log \mu$ is convex, for $\mu > 0$. To model this function using the exponential cone, let us introduce the epigraph variable q . We can then model the epigraph of this function as follows:

$$q \geq \mu \log \mu \quad (\text{EC.38})$$

$$\Leftrightarrow \frac{q}{\mu} \geq \log \mu \quad (\text{EC.39})$$

$$\Leftrightarrow e^{q/\mu} \geq \mu \quad (\text{EC.40})$$

$$\Leftrightarrow 1 \geq \mu e^{-q/\mu} \quad (\text{EC.41})$$

$$\Leftrightarrow (1, \mu, -q) \in \mathcal{K}_{\text{exp}}. \quad (\text{EC.42})$$

Therefore, the semi-log demand formulation (29) can be written using the auxiliary epigraph variables q_1, \dots, q_I as

$$\begin{aligned} \underset{\mu, \mathbf{q}, \mathbf{w}, \mathbf{x}}{\text{maximize}} \quad & \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \sum_{i=1}^I \beta_i \cdot \sum_{t \in \mathcal{P}_i} t \cdot w_{i,i,t} + \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \cdot \left(\sum_{t \in \mathcal{P}_j} t \cdot w_{i,j,t} \right) - \sum_{i=1}^I q_i \\ & (\text{EC.43a}) \end{aligned}$$

$$\text{subject to} \quad (1, \mu_i, -q_i) \in \mathcal{K}_{\text{exp}}, \quad \forall i \in [I], \quad (\text{EC.43b})$$

$$\text{constraints (29b) -- (29h)}. \quad (\text{EC.43c})$$

In a similar fashion, the log-log demand formulation (37) can be written as

$$\begin{aligned} \underset{\mu, \mathbf{q}, \mathbf{w}, \mathbf{x}}{\text{maximize}} \quad & \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \sum_{i=1}^I \beta_i \cdot \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} + \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \sum_{t \in \mathcal{P}_j} \log t \cdot w_{i,j,t} - \sum_{i=1}^I q_i \\ & (\text{EC.44a}) \end{aligned}$$

$$\text{subject to} \quad (1, \mu_i, -q_i) \in \mathcal{K}_{\text{exp}}, \quad \forall i \in [I], \quad (\text{EC.44b})$$

$$\text{constraints (29b) -- (29h)}. \quad (\text{EC.44c})$$

EC.4. Deterministic robust price optimization for finite \mathcal{P} , convex \mathcal{U} under the semi-log and log-log demand models

In this section, we describe how to formulate the DRPO problem as a mixed-integer exponential cone program for the semi-log and log-log demand models. In both cases, we assume that \mathcal{U} is a convex uncertainty set, and that Assumption 1 on the structure of \mathcal{P} holds.

EC.4.1. Semi-log model

For the semi-log demand model, we can write the DRPO problem as

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) \\ & = \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}. \end{aligned} \quad (\text{EC.45})$$

To accomplish our reformulation, we will make use of the fact that the optimal solution set of the DRPO problem is unchanged upon log-transformation, that is,

$$\arg \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) = \arg \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \log R(\mathbf{p}, \mathbf{u}).$$

Thus, instead of problem (EC.45), we can focus on the following problem:

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \log \left(\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \log \left(\sum_{i=1}^I e^{\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right) \end{aligned}$$

Here, we can again use the log-sum-exp biconjugate technique to reformulate the objective function in the following way:

$$\begin{aligned} & \log \left(\sum_{i=1}^I e^{\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right) \\ &= \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i \cdot (\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\}. \end{aligned}$$

Thus, the overall problem becomes the following max-min-max problem:

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i \cdot (\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\}.$$

Here, we observe that the objective function is linear in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$, and is concave in $\boldsymbol{\mu}$; additionally, the feasible region of \mathbf{u} is assumed to be convex, and the feasible region of $\boldsymbol{\mu}$ is convex and compact (being just the $(|I| - 1)$ -dimensional unit simplex). Therefore, we can use Sion's minimax theorem to interchange the minimization over \mathbf{u} and the maximization over $\boldsymbol{\mu}$, which gives us

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \mu_i \cdot (\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\ &= \max_{\mathbf{p} \in \mathcal{P}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \min_{\mathbf{u} \in \mathcal{U}} \left\{ \sum_{i=1}^I \mu_i \cdot (\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\ &= \max_{\mathbf{p} \in \mathcal{P}, \boldsymbol{\mu} \in \Delta_{[I]}} \min_{\mathbf{u} \in \mathcal{U}} \left\{ \sum_{i=1}^I \mu_i \cdot (\log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \end{aligned}$$

Under Assumption 1, this final problem can then be reformulated as robust mixed-integer exponential cone program, just as in Section 5.3. We introduce the same binary decision variable $x_{i,t}$ which is 1 if product i is offered at price $t \in \mathcal{P}_i$, and 0 otherwise, and we use $w_{i,j,t}$ to denote the linearization of $\mu_i \cdot x_{j,t}$ for $i, j \in [I]$, $t \in \mathcal{P}_j$. This gives rise to the following program:

$$\begin{aligned} & \underset{\boldsymbol{\mu}, \mathbf{w}, \mathbf{x}}{\text{maximize}} \quad \min_{\mathbf{u} \in \mathcal{U}} \left\{ \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \left(\sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \beta_i \cdot \sum_{t \in \mathcal{P}_i} t \cdot w_{i,i,t} + \sum_{j \neq i} \gamma_{i,j} \sum_{t \in \mathcal{P}_j} t \cdot w_{i,j,t} \right) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\ & \hspace{15cm} \text{(EC.46a)} \end{aligned}$$

$$\begin{aligned} & \text{subject to} \quad \sum_{t \in \mathcal{P}_j} w_{i,j,t} = \mu_i, \quad \forall i \in [I], j \in [I], \\ & \hspace{15cm} \text{(EC.46b)} \end{aligned}$$

$$\sum_{i=1}^I w_{i,j,t} = x_{j,t}, \quad \forall j \in [I], t \in \mathcal{P}_j, \quad (\text{EC.46c})$$

$$\sum_{i=1}^I \mu_i = 1, \quad (\text{EC.46d})$$

$$\sum_{t \in \mathcal{P}_i} x_{i,t} = 1, \quad \forall i \in [I], \quad (\text{EC.46e})$$

$$w_{i,j,t} \geq 0, \quad \forall i \in [I], j \in [I], t \in \mathcal{P}_j, \quad (\text{EC.46f})$$

$$x_{i,t} \in \{0, 1\}, \quad \forall i \in [I], t \in \mathcal{P}_i, \quad (\text{EC.46g})$$

$$\mu_i \geq 0, \quad \forall i \in [I]. \quad (\text{EC.46h})$$

Note that the feasible region of this problem is identical to that of problem (29), which appeared in our discussion of the separation problem for the RRPO problem when \mathcal{U} is convex and \mathcal{P} is finite. The difference here is that the objective is now a robust objective; it is the worst-case value of the objective of problem (29), taken over the convex uncertainty set \mathcal{U} . Depending on the structure of \mathcal{U} , the overall problem can remain in the mixed-integer convex program problem class. For example, if \mathcal{U} is a polyhedron, then one can use LP duality to reformulate the robust problem exactly by introducing additional variables and constraints, as is normally done in robust optimization (Bertsimas and Sim 2004, Ben-Tal and Nemirovski 2000, Bertsimas et al. 2011). Similarly, if \mathcal{U} is a second-order cone representable set, then one can again use conic duality to reformulate the problem. Alternatively, one can also consider solving the problem using a cutting plane method/delayed constraint generation approach, whereby one reformulates the program in epigraph form and then solves the inner minimization over \mathbf{u} to identify new cuts to add (Bertsimas et al. 2016a).

EC.4.2. Log-log model

For the log-log demand model, we can write the DRPO problem as

$$\begin{aligned} & \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} R(\mathbf{p}, \mathbf{u}) \\ &= \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \\ &= \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I e^{\alpha_i + (1 - \beta_i) \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \end{aligned}$$

Again, as with the semi-log model, solving the above problem is equivalent to solving the same problem with a log-transformed objective. Taking this log-transformed problem as our starting

point, replacing the log-sum-exp function with its biconjugate and applying Sion's minimax theorem gives us:

$$\begin{aligned}
& \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \log \left(\sum_{i=1}^I e^{\alpha_i + (1-\beta_i) \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \right) \\
&= \max_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{u} \in \mathcal{U}} \max_{\boldsymbol{\mu} \in \Delta_{[I]}} \left\{ \sum_{i=1}^I \left[\alpha_i \mu_i + (1-\beta_i) \mu_i \cdot \log p_i + \sum_{j \neq i} \gamma_{i,j} \mu_i \cdot \log p_j \right] - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\
&= \max_{\mathbf{p} \in \mathcal{P}, \boldsymbol{\mu} \in \Delta_{[I]}} \min_{\mathbf{u} \in \mathcal{U}} \left\{ \sum_{i=1}^I \left[\alpha_i \mu_i + (1-\beta_i) \mu_i \cdot \log p_i + \sum_{j \neq i} \gamma_{i,j} \mu_i \cdot \log p_j \right] - \sum_{i=1}^I \mu_i \log \mu_i \right\}.
\end{aligned}$$

Under Assumption 1, this last problem can be re-written as the following robust version of problem (37), with the decision variables defined identically:

$$\begin{aligned}
& \underset{\boldsymbol{\mu}, \mathbf{w}, \mathbf{x}}{\text{maximize}} \quad \min_{\mathbf{u} \in \mathcal{U}} \left\{ \sum_{i=1}^I \mu_i \alpha_i + \sum_{i=1}^I \left(\sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} - \beta_i \cdot \sum_{t \in \mathcal{P}_i} \log t \cdot w_{i,i,t} + \sum_{j \neq i} \gamma_{i,j} \sum_{t \in \mathcal{P}_j} \log t \cdot w_{i,j,t} \right) - \sum_{i=1}^I \mu_i \log \mu_i \right\} \\
& \hspace{25em} \text{(EC.47a)}
\end{aligned}$$

subject to constraints (29b) – (29h).

(EC.47b)

Again, this problem has exactly the same feasible region as the log-log separation problem (37) and the semi-log separation problem (29). Additionally, just as with the deterministic robust problem (EC.46) for the semi-log model, this problem can be further reformulated by exploiting the structure of \mathcal{U} , or otherwise one can design a cutting plane method that generates violated uncertain parameter vectors $\mathbf{u} \in \mathcal{U}$ on the fly.

EC.5. Incorporating constraints on price distributions

In this section, we present an extension of our framework to consider the possibility of constrained distributions over price vectors in the finite \mathcal{P} , convex \mathcal{U} case. In Section EC.5.1, we show how a particular type of constrained RRPO problem can be reformulated as a large-scale convex program. In Section EC.5.2, we present results of a small numerical experiment to show that incorporating constraints on the price distribution produced by the RRPO approach does not result in a large deterioration in performance compared to the DRPO approach.

EC.5.1. Randomized pricing with constraints

Suppose that we consider the following more general instance of the RRPO problem:

$$\text{RRPO-C: } \max_{\boldsymbol{\pi} \in \tilde{\Delta}_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \tag{EC.48}$$

where $\tilde{\Delta}_{\mathcal{P}}$ is the following subset of the probability simplex $\Delta_{\mathcal{P}}$:

$$\tilde{\Delta}_{\mathcal{P}} = \left\{ \boldsymbol{\pi} \in \Delta_{\mathcal{P}} \mid \sum_{\mathbf{p} \in \mathcal{P}} g_k(\mathbf{p}) \pi_{\mathbf{p}} = c_k, \forall k \in [K_1], \sum_{\mathbf{p} \in \mathcal{P}} h_k(\mathbf{p}) \pi_{\mathbf{p}} \leq d_k, \forall k \in [K_2], \right\}, \tag{EC.49}$$

where K_1 and K_2 are nonnegative integers. In words, $\tilde{\Delta}_{\mathcal{P}}$ is the set of all probability distributions that satisfy a set of *moment conditions* with respect to a set of moment functions $\{g_k(\cdot)\}_{k=1}^{K_1}$, $\{h_k(\cdot)\}_{k=1}^{K_2}$; in particular, the expected value of $g_k(\tilde{\mathbf{p}})$, where $\tilde{\mathbf{p}}$ is the random price vector that follows the distribution π , must be equal to c_k , for K_1 different moment functions. Similarly, the expected value of $h_k(\tilde{\mathbf{p}})$ must be less than or equal to d_k for K_2 different moment functions.

Let us now see how to reformulate this as a tractable problem. Let us assume that \mathcal{U} is convex and that $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} . In this case, Sion's minimax theorem again lets us assert

$$\max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.50})$$

$$= \min_{\mathbf{u} \in \mathcal{U}} \max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.51})$$

where we are also using the fact that $\tilde{\Delta}_{\mathcal{P}}$ is convex, as it is a polyhedron, and is compact, as it is a closed subset of the compact set $\Delta_{\mathcal{P}}$.

Now, let us introduce a K_1 -dimensional Lagrange multiplier vector λ and a K_2 -dimensional nonpositive Lagrange multiplier vector ρ . Observe that the inner maximization over π can be re-written equivalently as

$$\max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.52})$$

$$= \max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k \cdot \left(\sum_{\mathbf{p} \in \mathcal{P}} g_k(\mathbf{p}) \pi_{\mathbf{p}} - c_k \right) + \sum_{k=1}^{K_2} \rho_k \cdot \left(\sum_{\mathbf{p} \in \mathcal{P}} h_k(\mathbf{p}) \pi_{\mathbf{p}} - d_k \right) \right\} \quad (\text{EC.53})$$

$$= \max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.54})$$

$$= \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \left\{ \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.55})$$

$$= \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \max_{\pi \in \tilde{\Delta}_{\mathcal{P}}} \left\{ \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) \right\} - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.56})$$

$$= \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \max_{\mathbf{p} \in \mathcal{P}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.57})$$

where the interchange of maximization over $\pi \in \tilde{\Delta}_{\mathcal{P}}$ and minimization over λ and ρ follows by linear programming strong duality, and the final equality follows because the optimal solution of

the inner maximization over $\pi \in \Delta_{\mathcal{P}}$ occurs when all of the probability mass is placed at a single $\mathbf{p} \in \mathcal{P}$. Combining (EC.57) with (EC.51), we obtain

$$\min_{\mathbf{u} \in \mathcal{U}} \max_{\pi \in \Delta_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.58})$$

$$= \min_{\mathbf{u} \in \mathcal{U}} \min_{\lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \max_{\mathbf{p} \in \mathcal{P}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.59})$$

$$= \min_{\mathbf{u} \in \mathcal{U}, \lambda \in \mathbb{R}^{K_1}, \rho \leq \mathbf{0}} \left\{ \max_{\mathbf{p} \in \mathcal{P}} \left(R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right) - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \right\} \quad (\text{EC.60})$$

This final problem can be re-written as the following convex optimization problem, where we introduce an epigraph variable t to model the innermost maximum over \mathbf{p} :

$$\begin{aligned} \text{minimize}_{\lambda, \rho, t, \mathbf{u}} \quad & t - \sum_{k=1}^{K_1} \lambda_k c_k - \sum_{k=1}^{K_2} \rho_k d_k \end{aligned} \quad (\text{EC.61a})$$

$$\text{subject to} \quad t \geq R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}), \quad \forall \mathbf{p} \in \mathcal{P}, \quad (\text{EC.61b})$$

$$\mathbf{u} \in \mathcal{U}, \quad (\text{EC.61c})$$

$$\lambda \in \mathbb{R}^{K_1}, \quad (\text{EC.61d})$$

$$\rho \leq \mathbf{0}, \quad (\text{EC.61e})$$

$$t \in \mathbb{R}. \quad (\text{EC.61f})$$

Like the unconstrained problem (22), this problem can again potentially be solved via constraint generation. The challenge in solving this problem now becomes how one solves the modified separation problem

$$\max_{\mathbf{p} \in \mathcal{P}} \left\{ R(\mathbf{p}, \mathbf{u}) + \sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p}) + \sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p}) \right\}. \quad (\text{EC.62})$$

This separation problem is more challenging than the separation problem in the unconstrained distribution setting, because of the presence of the $\sum_{k=1}^{K_1} \lambda_k g_k(\mathbf{p})$ and $\sum_{k=1}^{K_2} \rho_k h_k(\mathbf{p})$ terms in the objective function. In particular, because of these terms and the fact that the λ_k and ρ_k variables can be negative, one cannot straightforwardly apply the biconjugate reformulation technique for the case of log-log and semi-log demand model. However, one can solve the problem heuristically, using the same random improvement approach that we use in Section EC.8.1. We adopt this approach in our numerical experiment in Section EC.5.2.

With regard to the moment functions $g_k(\cdot)$ and $h_k(\cdot)$, one natural choice of moment functions correspond to the first and second moments of each product's random price. In particular, letting $\tilde{\mathbf{p}}$ be the random price vector, one may wish to enforce that the distribution π over \mathcal{P} is such that

$$\mathbb{E}_{\pi}[\tilde{p}_i] = \mu_i, \quad \forall i \in [I], \quad (\text{EC.63})$$

$$\mathbb{E}_{\pi}[\tilde{p}_i^2] \leq \mu_i^2 + \sigma_i^2, \quad \forall i \in [I], \quad (\text{EC.64})$$

where μ_i and σ_i are user-specified values for the desired mean and standard deviation of each product's random price. This corresponds to specifying $2I$ moment functions, where $g_i(\mathbf{p}) = p_i$ for $1 \leq i \leq I$, and $h_i(\mathbf{p}) = p_i^2$ for $1 \leq i \leq I$. The resulting instance of the RRPO problem is one where we seek a distribution π that maximizes the worst-case revenue, with the additional restriction that the mean of each product's random price be equal to a predefined value μ_i , and the standard deviation is at most σ_i . By appropriately setting μ_i and σ_i , one can obtain randomized pricing schemes with controlled behavior, e.g., where the average price of a product is fixed to a certain value and to ensure that the variability around that value is not too large. As another example, one could set μ_i and σ_i so that the resulting price distribution π matches historical price data for a collection of products, i.e., on average, π results in the same average prices and the same variability in those prices as in the empirical distribution of \mathbf{p} in a historical data set.

We close this section by noting that moment constraints have a rich history in the distributionally robust optimization (DRO) literature; see, for example, Zymler et al. (2013), Rujeerapaiboon et al. (2016) and the references therein. We remark here that our use of moment constraints differs from how it is typically used in DRO. In DRO, one seeks to solve a stochastic program where the distribution of a parameter is uncertain, and one constructs an ambiguity set of possible distributions that are consistent with known information and data through moment constraints (e.g., the first and second moments of the uncertain distribution should be equal to certain known values); thus, moment constraints serve to limit how conservative the decision maker is. This contrasts with our use of moment constraints, where the moment constraints are on the firm's decision (the distribution of the price vector $\tilde{\mathbf{p}}$), and their purpose is to facilitate the implementation of the randomized pricing strategy.

EC.5.2. Numerical experiment with orangeJuice data set

In this section, we demonstrate the value of constrained price distributions using the `orangeJuice` problem instances studied in Section 7.2. We consider the same set of robust price optimization problems as in Section 7.2, with the modification that \mathcal{P}_i includes $\hat{\mu}_i$, which is the target mean for product i ; the reason for this will be discussed shortly.

For each value of θ , we solve the constrained randomized problem, with the moment constraints (EC.63) and (EC.64). We specifically impose that

$$\mathbb{E}_{\pi}[\tilde{p}_i] = \hat{\mu}_i, \quad \forall i \in [I], \quad (\text{EC.65})$$

$$\mathbb{E}_{\pi}[\tilde{p}_i^2] \leq \hat{\mu}_i^2 + \alpha^2 \hat{\sigma}_i^2, \quad \forall i \in [I], \quad (\text{EC.66})$$

where $\hat{\mu}_i$ and $\hat{\sigma}_i$ are the mean and standard deviation, respectively, of the price of product i , calculated with respect to the optimal unconstrained randomized robust price distribution π' . In

other words, we seek a distribution π that has the same marginal means as π' , but has less variance in each product's price than π' . We test values of $\alpha \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. Note that when $\alpha = 1$, the constrained RRPO problem is equivalent to the unconstrained RRPO problem. Additionally, we note that for any $\alpha \geq 0$, the set of distributions $\tilde{\Delta}_{\mathcal{P}}$ defined according to the moment constraints is non-empty, because \mathcal{P}_i includes $\hat{\mu}_i$ (as noted earlier).

Tables EC.1 and EC.2 display the results for the log-log and semi-log demand models. From these tables, we can see that in general, when the uncertainty set is small, constraining the distribution using the moment constraints described above results in weaker randomized pricing schemes than the unconstrained approach, and the resulting distributions do not improve on the deterministic robust price solution. For example, with the log-log demand model and $\theta = 0.1$, all of the constrained distributions with $\alpha \leq 0.9$ have a negative RI metric, indicating that their worst-case revenue is worse than the deterministic robust price solution. However, as the uncertainty set gets larger, there are values of $\alpha < 1$ for which the constrained RRPO solution yields a positive RI. For example, with the log-log demand model and $\theta = 0.5$, the constrained RRPO solution with α set to 0.8 or 0.9 outperforms the deterministic robust solution. As θ gets larger, it is possible to find distributions with less variability (i.e., with smaller α) that outperform the deterministic robust solution. For example, with the log-log demand model and $\theta = 1.0$, the constrained RRPO solution with $\alpha = 0.6$ continues to give a positive RI. These results suggest that by considering a constrained form of the RRPO problem, it is possible to trade-off some of the worst-case performance of the unconstrained RRPO solution to obtain price distributions that are more structured (in the examples shown above, ones with less variability than the unconstrained RRPO solution). Of course, the moment constraints that we have considered are only one way in which one can constrain the price distribution. An interesting direction for future research is to consider other types of constraints that could lead to structured price distributions that continue to provide a benefit over the deterministic robust scheme.

θ	Method m	Z_m^*	RI (%)
0.10	RRPO-C, $\alpha = 0.5$	237386.69	-58.05
0.10	RRPO-C, $\alpha = 0.6$	262704.91	-53.57
0.10	RRPO-C, $\alpha = 0.7$	285955.77	-49.47
0.10	RRPO-C, $\alpha = 0.8$	313211.80	-44.65
0.10	RRPO-C, $\alpha = 0.9$	415820.50	-26.52
0.10	RRPO	722647.22	27.71
0.50	RRPO-C, $\alpha = 0.5$	162117.27	-30.54
0.50	RRPO-C, $\alpha = 0.6$	190673.79	-18.30
0.50	RRPO-C, $\alpha = 0.7$	224890.46	-3.64
0.50	RRPO-C, $\alpha = 0.8$	251484.85	7.75
0.50	RRPO-C, $\alpha = 0.9$	286534.25	22.77
0.50	RRPO	342614.34	46.80
0.80	RRPO-C, $\alpha = 0.5$	131098.05	-19.21
0.80	RRPO-C, $\alpha = 0.6$	157189.92	-3.13
0.80	RRPO-C, $\alpha = 0.7$	175564.41	8.19
0.80	RRPO-C, $\alpha = 0.8$	200577.13	23.60
0.80	RRPO-C, $\alpha = 0.9$	228397.76	40.75
0.80	RRPO	260049.66	60.25
1.00	RRPO-C, $\alpha = 0.5$	111909.24	-12.72
1.00	RRPO-C, $\alpha = 0.6$	131534.71	2.58
1.00	RRPO-C, $\alpha = 0.7$	149969.36	16.96
1.00	RRPO-C, $\alpha = 0.8$	172717.05	34.70
1.00	RRPO-C, $\alpha = 0.9$	191272.60	49.17
1.00	RRPO	217580.86	69.69
1.50	RRPO-C, $\alpha = 0.5$	78581.62	3.54
1.50	RRPO-C, $\alpha = 0.6$	92204.17	21.48
1.50	RRPO-C, $\alpha = 0.7$	103533.52	36.41
1.50	RRPO-C, $\alpha = 0.8$	116139.87	53.02
1.50	RRPO-C, $\alpha = 0.9$	127482.81	67.97
1.50	RRPO	142307.66	87.50
2.00	RRPO-C, $\alpha = 0.5$	53377.06	8.23
2.00	RRPO-C, $\alpha = 0.6$	61205.36	24.10
2.00	RRPO-C, $\alpha = 0.7$	70425.84	42.80
2.00	RRPO-C, $\alpha = 0.8$	78122.02	58.40
2.00	RRPO-C, $\alpha = 0.9$	87114.25	76.63
2.00	RRPO	94847.37	92.31

Table EC.1 Results of comparison between RRPO, constrained RRPO and DRPO on orangeJuice instances with log-log demand model.

θ	Method m	Z_m^*	RI (%)
0.10	RRPO-C, $\alpha = 0.5$	153375.29	-47.20
0.10	RRPO-C, $\alpha = 0.6$	147513.70	-49.22
0.10	RRPO-C, $\alpha = 0.7$	173292.78	-40.34
0.10	RRPO-C, $\alpha = 0.8$	171790.80	-40.86
0.10	RRPO-C, $\alpha = 0.9$	205353.74	-29.30
0.10	RRPO	342357.06	17.86
0.50	RRPO-C, $\alpha = 0.5$	117253.37	-20.64
0.50	RRPO-C, $\alpha = 0.6$	131225.11	-11.18
0.50	RRPO-C, $\alpha = 0.7$	133415.93	-9.70
0.50	RRPO-C, $\alpha = 0.8$	155148.87	5.01
0.50	RRPO-C, $\alpha = 0.9$	171173.27	15.85
0.50	RRPO	197517.06	33.68
0.80	RRPO-C, $\alpha = 0.5$	87201.93	-17.53
0.80	RRPO-C, $\alpha = 0.6$	98469.28	-6.87
0.80	RRPO-C, $\alpha = 0.7$	104328.93	-1.33
0.80	RRPO-C, $\alpha = 0.8$	111791.18	5.73
0.80	RRPO-C, $\alpha = 0.9$	122249.99	15.62
0.80	RRPO	149709.04	41.59
1.00	RRPO-C, $\alpha = 0.5$	74421.62	-14.44
1.00	RRPO-C, $\alpha = 0.6$	77826.59	-10.52
1.00	RRPO-C, $\alpha = 0.7$	90332.98	3.86
1.00	RRPO-C, $\alpha = 0.8$	103167.94	18.61
1.00	RRPO-C, $\alpha = 0.9$	107505.80	23.60
1.00	RRPO	125987.02	44.85
1.50	RRPO-C, $\alpha = 0.5$	53040.97	-6.08
1.50	RRPO-C, $\alpha = 0.6$	56740.90	0.47
1.50	RRPO-C, $\alpha = 0.7$	62651.62	10.94
1.50	RRPO-C, $\alpha = 0.8$	66842.72	18.36
1.50	RRPO-C, $\alpha = 0.9$	73892.42	30.84
1.50	RRPO	82880.96	46.76
2.00	RRPO-C, $\alpha = 0.5$	35416.94	-4.70
2.00	RRPO-C, $\alpha = 0.6$	38862.06	4.57
2.00	RRPO-C, $\alpha = 0.7$	41834.74	12.57
2.00	RRPO-C, $\alpha = 0.8$	45750.84	23.10
2.00	RRPO-C, $\alpha = 0.9$	50007.72	34.56
2.00	RRPO	54665.15	47.09

Table EC.2 Results of comparison between RRPO, constrained RRPO and DRPO on orangeJuice instances with semi-log demand model.

EC.6. Approximation of convex price sets by discrete price sets

In this section, we show that under certain conditions, the optimal randomized robust pricing scheme over a convex price set \mathcal{P} can be well-approximated by the optimal randomized robust pricing scheme over a finite subset of \mathcal{P} .

Let $Z_{\text{RR}}^*(\mathcal{P}')$ denote the optimal objective value of the RRPO problem with respect to the price set \mathcal{P}' , which could be a finite set or an uncountable set:

$$Z_{\text{RR}}^*(\mathcal{P}') = \sup_{F \in \mathcal{F}_{\mathcal{P}'}} \inf_{\mathbf{u} \in \mathcal{U}} \int R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}). \quad (\text{EC.67})$$

We additionally make the following assumption regarding the revenue function R .

ASSUMPTION EC.1. *There exists $L > 0$ such that for all $\mathbf{u} \in \mathcal{U}$, $R(\cdot, \mathbf{u})$ is Lipschitz continuous with constant L , i.e.,*

$$|R(\mathbf{p}, \mathbf{u}) - R(\mathbf{p}', \mathbf{u})| \leq L \|\mathbf{p} - \mathbf{p}'\|_{\infty}, \quad (\text{EC.68})$$

for all $\mathbf{p}, \mathbf{p}' \in \mathcal{P}$.

Our main result can be stated as follows.

THEOREM EC.2. *Suppose that \mathcal{P} and \mathcal{U} are nonempty compact convex sets. Suppose that $R(\mathbf{p}, \mathbf{u})$ is convex in \mathbf{u} . Suppose also that Assumption EC.1 holds.*

Then, for any $\epsilon > 0$ there exists a finite subset $\hat{\mathcal{P}} \subseteq \mathcal{P}$, such that

$$Z_{\text{RR}}^*(\mathcal{P}) - Z_{\text{RR}}^*(\hat{\mathcal{P}}) \leq \epsilon. \quad (\text{EC.69})$$

To establish the result, we need to establish several auxiliary results. The first result states that if we are given a finite δ -net of \mathcal{P} , the nominal problem for a given \mathbf{u} with the exact set \mathcal{P} and the δ -net of \mathcal{P} will be close in their optimal objective values.

LEMMA EC.5. *Let $\mathcal{P} \subseteq \mathbb{R}^I$ be a nonempty set of price vectors, and let $\hat{\mathcal{P}} \subseteq \mathcal{P}$ be a finite set such that for every $\mathbf{p} \in \mathcal{P}$, there exists $\mathbf{p}' \in \hat{\mathcal{P}}$ such that $\|\mathbf{p} - \mathbf{p}'\|_{\infty} \leq \delta$. Suppose also that Assumption EC.1 holds. Then*

$$\sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \leq L\delta \quad (\text{EC.70})$$

for all $\mathbf{u} \in \mathcal{U}$.

Proof: Fix any $\epsilon > 0$ and any $\mathbf{u} \in \mathcal{U}$. Let \mathbf{p}^* be such that $R(\mathbf{p}^*, \mathbf{u}) \geq \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \epsilon$ or equivalently, $R(\mathbf{p}^*, \mathbf{u}) + \epsilon \geq \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$, which is guaranteed to exist because of the definition of the supremum.

Let $\hat{\mathbf{p}} \in \arg \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u})$. Note that $\hat{\mathbf{p}}$ is well-defined, because $\hat{\mathcal{P}}$ is a finite set.

Let $\tilde{\mathbf{p}} \in \hat{\mathcal{P}}$ be a price vector such that $\|\tilde{\mathbf{p}} - \mathbf{p}^*\|_\infty \leq \delta$, which is guaranteed to exist by the hypotheses of the lemma. We now have

$$\begin{aligned}
& \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \\
&= R(\mathbf{p}^*, \mathbf{u}) + \epsilon - R(\hat{\mathbf{p}}, \mathbf{u}) \\
&= R(\mathbf{p}^*, \mathbf{u}) - R(\tilde{\mathbf{p}}, \mathbf{u}) + R(\tilde{\mathbf{p}}, \mathbf{u}) - R(\hat{\mathbf{p}}, \mathbf{u}) + \epsilon \\
&\leq |R(\mathbf{p}^*, \mathbf{u}) - R(\tilde{\mathbf{p}}, \mathbf{u})| + 0 + \epsilon \\
&\leq L\delta + \epsilon,
\end{aligned}$$

where the first step follows by the definitions of \mathbf{p}^* and $\hat{\mathbf{p}}$; the second step follows by algebra; the third step follows by basic properties of absolute values, and also the fact that $\tilde{\mathbf{p}}$ is a feasible but not necessarily optimal solution to $\max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u})$, and hence $R(\tilde{\mathbf{p}}, \mathbf{u}) \leq R(\hat{\mathbf{p}}, \mathbf{u})$ (or equivalently, $R(\tilde{\mathbf{p}}, \mathbf{u}) - R(\hat{\mathbf{p}}, \mathbf{u}) \leq 0$); and the last step follows by Assumption EC.1.

Since $\epsilon > 0$ is arbitrary, it must therefore be that $\sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \leq L\delta$, as required. \square

The second result says that given a δ -net of \mathcal{P} , the objective value of a certain min-max problem, where the outer minimization is over \mathbf{u} and the inner maximization is over \mathcal{P} , will not change too much if the inner maximization is performed over the δ -net.

LEMMA EC.6. *Let $\mathcal{P} \subseteq \mathbb{R}^I$ be a nonempty set of price vectors, and let $\hat{\mathcal{P}} \subseteq \mathcal{P}$ be a finite set such that for every $\mathbf{p} \in \mathcal{P}$, there exists $\mathbf{p}' \in \hat{\mathcal{P}}$ such that $\|\mathbf{p} - \mathbf{p}'\|_\infty \leq \delta$. Suppose also that Assumption EC.1 holds. Then*

$$\inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \leq L\delta. \quad (\text{EC.71})$$

Proof: Fix any $\epsilon > 0$. Let $\hat{\mathbf{u}} \in \mathcal{U}$ be an uncertain parameter vector such that $\max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \hat{\mathbf{u}}) \leq \inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) + \epsilon$. Such a $\hat{\mathbf{u}}$ is guaranteed to exist by the definition of the infimum.

We now have:

$$\inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.72})$$

$$\leq \inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \hat{\mathbf{u}}) + \epsilon \quad (\text{EC.73})$$

$$\leq \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \hat{\mathbf{u}}) - \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \hat{\mathbf{u}}) + \epsilon \quad (\text{EC.74})$$

$$\leq L\delta + \epsilon, \quad (\text{EC.75})$$

where the first step follows by the definition of $\hat{\mathbf{u}}$ as an ϵ -approximate solution to the discretized problem $\inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u})$; the second step follows because $\hat{\mathbf{u}}$ is a feasible solution to the continuous (non-discretized) problem $\inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$; and the final step by Lemma EC.5. Since ϵ was arbitrary, it must be that $\inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \leq L\delta$ as required. \square

Finally, we prove Theorem EC.2.

Proof of Theorem EC.2: Let $\delta = \epsilon/L$. Since \mathcal{P} is compact, there exists a finite set $\hat{\mathcal{P}}$ which is a δ -net, that is, for every $\mathbf{p} \in \mathcal{P}$, there exists $\mathbf{p}' \in \hat{\mathcal{P}}$ such that

$$\|\mathbf{p} - \mathbf{p}'\|_{\infty} \leq \delta. \quad (\text{EC.76})$$

We now have

$$Z_{\text{RR}}^*(\mathcal{P}) - Z_{\text{RR}}^*(\hat{\mathcal{P}}) \quad (\text{EC.77})$$

$$= \sup_{F \in \mathcal{F}(\mathcal{P})} \inf_{\mathbf{u} \in \mathcal{U}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) - \sup_{F \in \mathcal{F}(\hat{\mathcal{P}})} \inf_{\mathbf{u} \in \mathcal{U}} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \quad (\text{EC.78})$$

$$= \inf_{\mathbf{u} \in \mathcal{U}} \sup_{F \in \mathcal{F}(\mathcal{P})} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) - \inf_{\mathbf{u} \in \mathcal{U}} \sup_{F \in \mathcal{F}(\hat{\mathcal{P}})} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p}) \quad (\text{EC.79})$$

$$= \inf_{\mathbf{u} \in \mathcal{U}} \sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u}) - \inf_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \hat{\mathcal{P}}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.80})$$

$$\leq L\delta \quad (\text{EC.81})$$

$$= \epsilon, \quad (\text{EC.82})$$

where the first step follows by the definition of $Z_{\text{RR}}^*(\mathcal{P})$, the second by applying Sion's minimax theorem, the third by the equivalence of $\sup_{F \in \mathcal{F}(\mathcal{P})} \int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p})$ and $\sup_{\mathbf{p} \in \mathcal{P}} R(\mathbf{p}, \mathbf{u})$, the fourth by Lemma EC.6, and the final step by the definition of δ . (Note that in the second step, Sion's minimax theorem applies as $\int_{\mathcal{P}} R(\mathbf{p}, \mathbf{u}) dF(\mathbf{p})$ is convex in \mathbf{u} (and hence both quasiconvex and continuous in \mathbf{u}), and linear in F (and hence both quasiconcave and continuous in the topology of weak convergence in F), and $\mathcal{F}(\mathcal{P})$ and \mathcal{U} are both convex, and \mathcal{U} is additionally compact.) \square

We now pause to make a few comments about Theorem EC.2. First, the Lipschitz continuity assumption on the revenue function R is not particularly restrictive, and is satisfied by many

popular demand models. In particular, when \mathcal{P} and \mathcal{U} are compact sets and the gradient $\nabla_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$ with respect to \mathbf{p} is continuous in both \mathbf{p} and \mathbf{u} , then R will satisfy the requirement of the theorem for a Lipschitz constant L that is defined as

$$L = \max_{\mathbf{u} \in \mathcal{U}} \max_{\mathbf{p} \in \mathcal{P}} \|\nabla_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})\|_1. \quad (\text{EC.83})$$

For the semi-log and log-log demand models, the revenue function R has a gradient $\nabla_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$ that is a continuous function of \mathbf{p} and \mathbf{u} .

Second, this theorem implies that when the revenue function is Lipschitz continuous with the same constant across every \mathbf{u} , discretization will perform well. In particular, for a price set of the form $\mathcal{P} = [p_{\min,i}, p_{\max,i}]^I$, a uniform discretization of the form $\hat{\mathcal{P}} = \{\delta \lceil p_{\min,i}/\delta \rceil, \delta(\lceil p_{\min,i}/\delta \rceil + 1), \delta(\lceil p_{\min,i}/\delta \rceil + 2), \dots, \delta \lfloor p_{\max,i}/\delta \rfloor\}^I$ for some $\delta > 0$ will yield the δ -net in the ℓ_∞ norm required in the proof of Theorem EC.2. By setting δ to be suitably small, we can thus ensure that the discretized RRPO problem returns a randomized pricing scheme that is arbitrarily close to the optimal randomized pricing scheme over the continuous price set \mathcal{P} . This result thus provides a theoretical justification for our focus on discrete price sets in Sections 5 and EC.7 of the paper.

EC.7. Solution method for finite \mathcal{P} , finite \mathcal{U}

The second solution approach we consider is for the case where both \mathcal{P} and \mathcal{U} are finite sets. In particular, we assume that the uncertainty set \mathcal{U} is a binary representable set. For fixed positive integers m and n , we let \mathcal{U} be defined as

$$\mathcal{U} = \{\mathbf{u} = \mathbf{F}\mathbf{z} \mid \mathbf{A}\mathbf{z} \leq \mathbf{b}, \mathbf{z} \in \{0, 1\}^n\}, \quad (\text{EC.84})$$

where \mathbf{b} is a m -dimensional real vector, \mathbf{A} is a m -by- n real matrix and \mathbf{F} is a M -by- n real matrix, where M is the dimension of the uncertain parameter vector \mathbf{u} . We note here that any finite set can be represented in this way: if $\mathcal{U} = \{\mathbf{u}_1, \dots, \mathbf{u}_H\}$, then one can define \mathbf{z} as $\mathbf{z} = (z_1, \dots, z_H) \in \{0, 1\}^H$, use the linear inequality system $\mathbf{A}\mathbf{z} \leq \mathbf{b}$ to represent the constraints $\mathbf{1}^T \mathbf{z} \leq 1$, $-\mathbf{1}^T \mathbf{z} \leq -1$, and define $\mathbf{F} = [\mathbf{u}_1 \dots \mathbf{u}_H]$ (i.e., the matrix obtained from stacking the vectors $\mathbf{u}_1, \dots, \mathbf{u}_H$ as columns).

From a modeling standpoint, a discrete set \mathcal{U} can be used to model the firm's uncertainty in different ways. One could use this type of uncertainty set to model a finite set of scenarios in terms of the customer population's demand response to the firm's prices. Another possibility is that the firm faces a collection of different customer segments, where each one corresponds to a particular type of demand model, and at most one segment will respond to the firm's products; in this setting, the firm could use the uncertainty set to model its uncertainty as to which segment will respond. Given the factor structure $\mathbf{u} = \mathbf{F}\mathbf{z}$ within \mathcal{U} , one could also consider a setting where the firm estimates a richer demand model where the elasticity parameters (represented by \mathbf{u}) are dependent

on macroeconomic factors (represented by \mathbf{z}) that are uncertain and represented in a discrete way; for example, one could use binary variables z_1, z_2, z_3 to indicate three different unemployment scenarios, and z_4, z_5, z_6 to indicate three different interest rate scenarios. The firm would then obtain a randomized pricing strategy that accounts for uncertainty in these macroeconomic factors.

Recall that when \mathcal{P} is finite, then the RRPO problem is

$$Z_{\text{RR}}^* = \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}). \quad (\text{EC.85})$$

We can transform this problem into a dual problem where the outer problem is to optimize a distribution over uncertain parameter vectors, and the inner problem is to optimize over the price vector, as follows:

$$Z_{\text{RR}}^* = \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.86})$$

$$= \max_{\pi \in \Delta_{\mathcal{P}}} \min_{\lambda \in \Delta_{\mathcal{U}}} \sum_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \mathcal{U}} \pi_{\mathbf{p}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.87})$$

$$= \min_{\lambda \in \Delta_{\mathcal{U}}} \max_{\pi \in \Delta_{\mathcal{P}}} \sum_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \mathcal{U}} \pi_{\mathbf{p}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}) \quad (\text{EC.88})$$

$$= \min_{\lambda \in \Delta_{\mathcal{U}}} \max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \mathcal{U}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.89})$$

where the first equality follows because minimization of a function of \mathbf{u} over the finite set \mathcal{U} is the same as minimizing the expected value of that function over all probability mass functions supported on \mathcal{U} ; the second equality follows by linear programming duality; and the final equality follows because maximization of a function of \mathbf{p} over \mathcal{P} is the same as maximizing the expected value of that function over all probability mass functions supported on \mathcal{P} . We refer to problem (EC.85) as the *primal* problem and (EC.89) as the *dual* problem.

Consider now the *restricted primal problem*, where we replace \mathcal{P} with a subset $\hat{\mathcal{P}} \subseteq \mathcal{P}$ in problem (EC.85), and the *restricted dual problem*, where we replace \mathcal{U} with a subset $\hat{\mathcal{U}} \subseteq \mathcal{U}$ in problem (EC.89). Let us denote the objective values of these two problems with $Z_{P, \hat{\mathcal{P}}}$ and $Z_{D, \hat{\mathcal{U}}}$, respectively. These two problems are:

$$Z_{P, \hat{\mathcal{P}}} = \max_{\pi \in \Delta_{\hat{\mathcal{P}}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad (\text{EC.90})$$

$$Z_{D, \hat{\mathcal{U}}} = \min_{\lambda \in \Delta_{\hat{\mathcal{U}}}} \max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}). \quad (\text{EC.91})$$

Observe that $Z_{P, \hat{\mathcal{P}}}$ and $Z_{D, \hat{\mathcal{U}}}$ bound Z_{RR}^* from below and above, that is,

$$Z_{P, \hat{\mathcal{P}}} \leq Z_{\text{RR}}^* \leq Z_{D, \hat{\mathcal{U}}}.$$

In the above, the justification for the first inequality is because maximizing over distributions supported on the smaller set of price vectors $\hat{\mathcal{P}}$ cannot result in a higher worst-case objective than solving the full primal problem with \mathcal{P} , which gives the value Z_{RR}^* . The second inequality similarly follows because minimizing over distributions supported on the smaller set of uncertainty realizations $\hat{\mathcal{U}}$ cannot result in a lower worst-case objective than solving the full dual problem with \mathcal{U} , which gives Z_{RR}^* .

The idea of double column generation is as follows. Let us pick some subset of price vectors $\hat{\mathcal{P}} \subseteq \mathcal{P}$ and some subset of uncertainty realizations $\hat{\mathcal{U}} \subseteq \mathcal{U}$. Observe that the restricted primal problem (EC.90) for $\hat{\mathcal{P}}$ can be written in epigraph form as

$$\underset{\pi, t}{\text{maximize}} \quad t \tag{EC.92a}$$

$$\text{subject to} \quad t \leq \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{u} \in \mathcal{U}, \tag{EC.92b}$$

$$\sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} = 1, \tag{EC.92c}$$

$$\pi_{\mathbf{p}} \geq 0, \quad \forall \mathbf{p} \in \hat{\mathcal{P}}. \tag{EC.92d}$$

This problem has a huge number of constraints (one for each $\mathbf{u} \in \mathcal{U}$). However, we can solve it using delayed constraint generation, starting from the set $\hat{\mathcal{U}}$. Upon solving it in this way, at termination we will have a subset \mathcal{U}' of uncertainty realizations from \mathcal{U} that were found during the constraint generation process. We update $\hat{\mathcal{U}}$ to be equal to \mathcal{U}' .

With this (updated) subset $\hat{\mathcal{U}}$ in hand, we now solve the restricted dual problem (EC.91) for $\hat{\mathcal{U}}$. This problem can be written in epigraph form as

$$\underset{\lambda, \rho}{\text{minimize}} \quad \rho \tag{EC.93a}$$

$$\text{subject to} \quad \rho \geq \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}, \tag{EC.93b}$$

$$\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} = 1, \tag{EC.93c}$$

$$\lambda_{\mathbf{u}} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}. \tag{EC.93d}$$

This problem also has a huge number of constraints, but again we can solve it using delayed constraint generation, with the initial subset of price vectors set to $\hat{\mathcal{P}}$. At termination, we will have a new subset \mathcal{P}' of price vectors, which will contain the original set of price vectors in $\hat{\mathcal{P}}$. We then update $\hat{\mathcal{P}}$ to \mathcal{P}' , and go back to solving the restricted primal problem. The process then repeats: after solving the restricted primal, we will have a new (bigger) $\hat{\mathcal{U}}$; we then solve the restricted dual, after which we have a new (bigger) $\hat{\mathcal{P}}$; we then go back to the restricted primal, and so on.

After each iteration of solving the restricted primal and restricted dual, the set $\hat{\mathcal{P}}$ expands and the set $\hat{\mathcal{U}}$ expands. Thus, the bounds $Z_{P,\hat{\mathcal{P}}}$ and $Z_{D,\hat{\mathcal{U}}}$ get closer and closer to Z_{RR}^* . The algorithm can then be terminated either when $Z_{P,\hat{\mathcal{P}}} = Z_{D,\hat{\mathcal{U}}}$, which would imply that both restricted primal and restricted dual objective values exactly coincide with Z_{RR}^* ; or otherwise, one can terminate when $Z_{D,\hat{\mathcal{U}}} - Z_{P,\hat{\mathcal{P}}} < \epsilon$, where $\epsilon > 0$ is a user specified tolerance.

The overall algorithmic approach is formalized as Algorithm 3. This algorithm invokes two procedures, PRIMALCG (Algorithm 4) and DUALCG (Algorithm 5), which are delayed constraint generation algorithms for solving the restricted primal and dual problems respectively. We note that Algorithm 3 is an adaptation of the double column generation algorithm of Wang et al. (2024) for the randomized robust assortment optimization problem, which is itself adapted from the double column generation algorithm of Delage and Saif (2022) for solving mixed-integer distributionally robust optimization problems. The proof of correctness of this procedure follows similarly to Delage and Saif (2022), and is omitted for brevity. The novelty in our approach lies in how we handle the separation problems which are at the heart of PRIMALCG and DUALCG, which we discuss next. Sections EC.7.1, EC.7.2 and EC.7.3 provide the details for the linear, semi-log and log-log demand models. Section EC.7.4 discusses the complexity of randomization and provides a result that is analogous to Theorem 4 for the case of a finite \mathcal{P} and finite \mathcal{U} .

Algorithm 3 Double column generation method for solving the finite \mathcal{P} , finite \mathcal{U} RRPO problem.

- 1: Initialize $\hat{\mathcal{P}}$ to be a non-empty subset of \mathcal{P} , and $\hat{\mathcal{U}}$ to be a non-empty subset of \mathcal{U} .
 - 2: Set $\text{LB} \leftarrow -\infty$, $\text{UB} \leftarrow +\infty$
 - 3: **repeat**
 - 4: Run PRIMALCG($\hat{\mathcal{P}}, \hat{\mathcal{U}}$) to solve the restricted primal problem with $\hat{\mathcal{P}}$ and with $\hat{\mathcal{U}}$ as the initial uncertainty set. Let the objective value be $Z_{P,\hat{\mathcal{P}}}$ and the new uncertainty set be \mathcal{U}' .
 - 5: Set $\hat{\mathcal{U}} \leftarrow \mathcal{U}'$.
 - 6: Set $\text{LB} \leftarrow Z_{P,\hat{\mathcal{P}}}$.
 - 7: Run DUALCG($\hat{\mathcal{P}}, \hat{\mathcal{U}}$) to solve the restricted dual problem with $\hat{\mathcal{U}}$ and with $\hat{\mathcal{P}}$ as the initial price vector set. Let the objective value be $Z_{D,\hat{\mathcal{U}}}$ and the new price vector set be \mathcal{P}' .
 - 8: Set $\hat{\mathcal{P}} \leftarrow \mathcal{P}'$.
 - 9: Set $\text{UB} \leftarrow Z_{D,\hat{\mathcal{U}}}$.
 - 10: **until** $\text{UB} - \text{LB} < \epsilon$
-

Note that the doubly restricted primal and dual problems (EC.94) and (EC.96) solved in PRIMALCG and DUALCG are both linear programs, and can be thus be solved easily. The principal

Algorithm 4 PRIMALCG procedure.

- 1: Initialize $\mathcal{U}' \leftarrow \hat{\mathcal{U}}$
- 2: **repeat**
- 3: Solve the doubly restricted primal problem:

$$\underset{\boldsymbol{\pi}, t}{\text{maximize}} \quad t \tag{EC.94a}$$

$$\text{subject to} \quad t \leq \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{u} \in \mathcal{U}', \tag{EC.94b}$$

$$\sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} = 1, \tag{EC.94c}$$

$$\pi_{\mathbf{p}} \geq 0, \quad \forall \mathbf{p} \in \hat{\mathcal{P}}. \tag{EC.94d}$$

Let $(\boldsymbol{\pi}, t^*)$ be the optimal solution of the doubly restricted problem.

- 4: Solve the primal separation problem:

$$\min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} \cdot R(\mathbf{p}, \mathbf{u}). \tag{EC.95}$$

Let t' and \mathbf{u}^* be the optimal objective value and solution of this separation problem.

- 5: **if** $t^* > t'$ **then**
 - 6: Set $\mathcal{U}' \leftarrow \mathcal{U}' \cup \{\mathbf{u}^*\}$
 - 7: **end if**
 - 8: **until** $t^* \leq t'$
 - 9: Set $Z_{P, \hat{\mathcal{P}}} \leftarrow t^*$
 - 10: **return** $(Z_{P, \hat{\mathcal{P}}}, \mathcal{U}')$.
-

difficulty in these procedures comes from the primal and dual separation problems (EC.95) and (EC.97), which require optimizing over an uncertain parameter vector $\mathbf{u} \in \mathcal{U}$ and a price vector $\mathbf{p} \in \mathcal{P}$ respectively. In the following sections, we discuss how these two separation problems can be tackled for the linear, semi-log and log-log demand models. Note that in all three sections, we continue to make Assumption 1, which states that \mathcal{P} can be written as the Cartesian product of finite sets of prices for each product, i.e., $\mathcal{P} = \mathcal{P}_1 \times \cdots \times \mathcal{P}_I$, where $\mathcal{P}_1, \dots, \mathcal{P}_I$ are finite sets.

EC.7.1. Primal and dual subproblems for linear demand model

For the linear demand model, the primal separation problem is

$$\min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} \cdot \left[\sum_{i=1}^I p_i \cdot (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) \right]. \tag{EC.98}$$

Algorithm 5 DUALCG procedure.

-
- 1: Initialize $\mathcal{P}' \leftarrow \hat{\mathcal{P}}$
 - 2: **repeat**
 - 3: Solve the doubly restricted dual problem:

$$\underset{\boldsymbol{\lambda}, \rho}{\text{minimize}} \quad \rho \tag{EC.96a}$$

$$\text{subject to} \quad \rho \geq \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{p} \in \mathcal{P}', \tag{EC.96b}$$

$$\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} = 1, \tag{EC.96c}$$

$$\lambda_{\mathbf{u}} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}. \tag{EC.96d}$$

Let $(\boldsymbol{\lambda}, \rho^*)$ be the optimal solution of the doubly restricted problem.

- 4: Solve the dual separation problem:

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot R(\mathbf{p}, \mathbf{u}) \tag{EC.97}$$

Let ρ' and \mathbf{p}^* be the optimal objective value and solution of this separation problem.

- 5: **if** $\rho^* < \rho'$ **then**
 - 6: Set $\mathcal{P}' \leftarrow \mathcal{P}' \cup \{\mathbf{p}^*\}$
 - 7: **end if**
 - 8: **until** $\rho^* \geq \rho'$
 - 9: Set $Z_{D, \hat{\mathcal{U}}} \leftarrow \rho^*$
 - 10: **return** $(Z_{D, \hat{\mathcal{U}}}, \mathcal{P}')$.
-

Note that this objective function is linear in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Therefore, the whole problem can be expressed as

$$\underset{\mathbf{p} \in \hat{\mathcal{P}}}{\text{minimize}} \quad \sum \pi_{\mathbf{p}} \cdot \left[\sum_{i=1}^I p_i \cdot (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) \right] \tag{EC.99a}$$

$$\text{subject to} \quad \mathbf{u} = \mathbf{F}\mathbf{z}, \tag{EC.99b}$$

$$\mathbf{A}\mathbf{z} \leq \mathbf{b}, \tag{EC.99c}$$

$$\mathbf{z} \in \{0, 1\}^n, \tag{EC.99d}$$

which is a mixed-integer linear program.

The dual separation problem is

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I p_i (\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j) \right]. \tag{EC.100}$$

By introducing the same binary variables as in the separation problem (24) (the linear demand separation problem for the convex \mathcal{U} setting), we obtain the following mixed-integer linear program:

$$\begin{aligned} \text{maximize}_{\mathbf{x}, \mathbf{y}} \quad & \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \alpha_i \cdot t \cdot x_{i,t} - \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} t^2 \cdot \beta_i \cdot x_{i,t} + \sum_{i=1}^I \sum_{j \neq i} \sum_{t_1 \in \mathcal{P}_i} \sum_{t_2 \in \mathcal{P}_j} \gamma_{i,j} \cdot t_1 \cdot t_2 \cdot y_{i,j,t_1,t_2} \right] \\ & \text{(EC.101a)} \end{aligned}$$

$$\text{subject to} \quad \sum_{t \in \mathcal{P}_i} x_{i,t} = 1, \quad \forall i \in [I], \quad \text{(EC.101b)}$$

$$\sum_{t_2 \in \mathcal{P}_j} y_{i,j,t_1,t_2} = x_{i,t_1}, \quad \forall i, j \in [I], j \neq i, t_2 \in \mathcal{P}_j, \quad \text{(EC.101c)}$$

$$\sum_{t_1 \in \mathcal{P}_i} y_{i,j,t_1,t_2} = x_{j,t_2}, \quad \forall i, j \in [I], j \neq i, t_1 \in \mathcal{P}_i, \quad \text{(EC.101d)}$$

$$x_{i,t} \in \{0, 1\}, \quad \forall i \in [I], t \in \mathcal{P}_i, \quad \text{(EC.101e)}$$

$$y_{i,j,t_1,t_2} \in \{0, 1\}, \quad \forall i, j \in [I], i \neq j, t_1 \in \mathcal{P}_i, t_2 \in \mathcal{P}_j. \quad \text{(EC.101f)}$$

Importantly, note that the size of this problem does not scale with the number of uncertainty realizations inside $\hat{\mathcal{U}}$; the form of this problem is equivalent to problem (24) where \mathbf{u} is replaced with $\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot \mathbf{u}$ (the “average” uncertain demand parameter). As we will see in the next couple of sections, the same will not be true for the semi-log and log-log demand models.

EC.7.2. Primal and dual subproblems for semi-log demand model

For the semi-log demand model, the primal separation problem is

$$\min_{\substack{\mathbf{u} \in \hat{\mathcal{U}} \\ \mathbf{p} \in \mathcal{P}}} \sum \pi_{\mathbf{p}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right]. \quad \text{(EC.102)}$$

Note that this objective function is convex in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$, because the weights $\pi_{\mathbf{p}}$ and p_i for a given $\mathbf{p} \in \mathcal{P}$ and $i \in [I]$ are nonnegative, and because the function $e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j}$ is convex in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Thus, the whole problem can be expressed as

$$\begin{aligned} \text{minimize}_{\mathbf{u}, \mathbf{z}} \quad & \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right] \\ & \text{(EC.103a)} \end{aligned}$$

$$\text{subject to} \quad \mathbf{u} = \mathbf{F}\mathbf{z}, \quad \text{(EC.103b)}$$

$$\mathbf{A}\mathbf{z} \leq \mathbf{b}, \quad \text{(EC.103c)}$$

$$\mathbf{z} \in \{0, 1\}^n, \quad \text{(EC.103d)}$$

which can be re-written as a mixed-integer exponential cone program.

The dual separation problem is

$$\max_{\substack{\mathbf{p} \in \mathcal{P} \\ \mathbf{u} \in \hat{\mathcal{U}}}} \sum \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right]. \quad \text{(EC.104)}$$

The objective function of this problem is in general not concave in \mathbf{p} . However, just as in Section 5.3, the related problem of optimizing the logarithm of this objective, which is

$$\max_{\mathbf{p} \in \mathcal{P}} \log \left[\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right] \right] \quad (\text{EC.105})$$

$$= \max_{\mathbf{p} \in \mathcal{P}} \log \left[\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I e^{\log \lambda_{\mathbf{u}} + \log p_i + \alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{i,j} p_j} \right] \quad (\text{EC.106})$$

can be reformulated as a mixed-integer exponential cone program using the same biconjugate-based technique in Section 5.3. In particular, when Assumption 1 holds, then problem (EC.106) is equivalent to

$$\begin{aligned}
\text{maximize}_{\mathbf{w}, \mathbf{x}, \boldsymbol{\mu}} \quad & \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \mu_{\mathbf{u},i} \cdot (\log \lambda_{\mathbf{u}} + \alpha_i) \\
& + \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} \log t w_{\mathbf{u},i,t} \\
& + \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} (-\beta_i) \cdot t \cdot w_{\mathbf{u},i,t} \\
& + \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \cdot \sum_{t \in \mathcal{P}_j} t \cdot w_{\mathbf{u},i,j,t} \\
& - \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \mu_{\mathbf{u},i} \log \mu_{\mathbf{u},i}
\end{aligned} \tag{EC.107a}$$

$$\text{subject to} \quad \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \mu_{\mathbf{u},i} = 1, \tag{EC.107b}$$

$$\sum_{t \in \mathcal{P}_j} w_{\mathbf{u},i,j,t} = \mu_{\mathbf{u},i}, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i, j \in [I], \tag{EC.107c}$$

$$\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I w_{\mathbf{u},i,j,t} = x_{j,t}, \quad \forall j \in [I], t \in \mathcal{P}_j, \tag{EC.107d}$$

$$\sum_{t \in \mathcal{P}_j} x_{j,t} = 1, \quad \forall j \in [I], \tag{EC.107e}$$

$$w_{\mathbf{u},i,j,t} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i, j \in [I], t \in \mathcal{P}_j, \tag{EC.107f}$$

$$\mu_{\mathbf{u},i} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i \in [I], \tag{EC.107g}$$

$$x_{j,t} \in \{0, 1\}, \quad \forall j \in [I], t \in \mathcal{P}_j, \tag{EC.107h}$$

where $x_{j,t}$ is a binary decision variable that is 1 if product j is offered at price $t \in \mathcal{P}_j$, and 0 otherwise; $\mu_{\mathbf{u},i}$ is a nonnegative decision variable introduced as part of the biconjugate-based reformulation; and $w_{\mathbf{u},i,j,t}$ is a decision variable that represents the linearization of $\mu_{\mathbf{u},i} \cdot x_{j,t}$ for all $\mathbf{u} \in \hat{\mathcal{U}}, i, j \in [I]$, and $t \in \mathcal{P}_j$.

As with problem (29), this problem can be expressed as a mixed-integer exponential cone program. One notable difference between formulation (EC.107) and formulation (29) from earlier is that the number of decision variables and constraints is larger because the decision variable $\mu_{\mathbf{u},i}$ is introduced for every combination of an uncertainty realization in $\hat{\mathcal{U}}$ and each product i ; thus, $\boldsymbol{\mu}$ represents a probability mass function over the set $\hat{\mathcal{U}} \times [I]$.

EC.7.3. Primal and dual subproblems for log-log demand model

For the log-log demand model, the primal separation problem is

$$\min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} \cdot \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j}. \quad (\text{EC.108})$$

Note that the objective function is convex in $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$; it is the nonnegative weighted combination of terms of the form $e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j}$, each of which are convex in $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Thus, the overall problem, which can be stated as

$$\underset{\mathbf{u}, \mathbf{z}}{\text{minimize}} \quad \sum_{\mathbf{p} \in \hat{\mathcal{P}}} \pi_{\mathbf{p}} \cdot \sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \quad (\text{EC.109a})$$

$$\text{subject to} \quad \mathbf{u} = \mathbf{F}\mathbf{z}, \quad (\text{EC.109b})$$

$$\mathbf{A}\mathbf{z} \leq \mathbf{b}, \quad (\text{EC.109c})$$

$$\mathbf{z} \in \{0, 1\}^n, \quad (\text{EC.109d})$$

is a mixed-integer convex program, and can be expressed as a mixed-integer exponential cone program.

The dual separation problem is

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{\mathbf{u} \in \mathcal{U}} \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \right]. \quad (\text{EC.110})$$

The objective function of this problem is in general not concave in \mathbf{p} . However, following the same method as in Section 5.4, we can reformulate the related problem of maximizing the logarithm, which is

$$\max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{\mathbf{u} \in \mathcal{U}} \lambda_{\mathbf{u}} \cdot \left[\sum_{i=1}^I p_i \cdot e^{\alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \right] \right) \quad (\text{EC.111})$$

$$= \max_{\mathbf{p} \in \mathcal{P}} \log \left(\sum_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I e^{\log \lambda_{\mathbf{u}} + \log p_i + \alpha_i - \beta_i \log p_i + \sum_{j \neq i} \gamma_{i,j} \log p_j} \right) \quad (\text{EC.112})$$

as a mixed-integer exponential cone program. Under Assumption 1, the resulting formulation is

$$\begin{aligned} \underset{\mathbf{x}, \mathbf{w}, \boldsymbol{\mu}}{\text{maximize}} \quad & \sum_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I \mu_{\mathbf{u},i} \cdot (\log \lambda_{\mathbf{u}} + \alpha_i) \\ & + \sum_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I \sum_{t \in \mathcal{P}_i} (1 - \beta_i) \log t \cdot w_{\mathbf{u},i,i,t} \\ & + \sum_{\mathbf{u} \in \mathcal{U}} \sum_{i=1}^I \sum_{j \neq i} \gamma_{i,j} \cdot \sum_{t \in \mathcal{P}_j} \log t \cdot w_{\mathbf{u},i,j,t} \end{aligned}$$

$$-\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \mu_{\mathbf{u},i} \log \mu_{\mathbf{u},i} \quad (\text{EC.113a})$$

$$\text{subject to } \sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I \mu_{\mathbf{u},i} = 1, \quad (\text{EC.113b})$$

$$\sum_{t \in \mathcal{P}_j} w_{\mathbf{u},i,j,t} = \mu_{\mathbf{u},i}, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i, j \in [I], \quad (\text{EC.113c})$$

$$\sum_{\mathbf{u} \in \hat{\mathcal{U}}} \sum_{i=1}^I w_{\mathbf{u},i,j,t} = x_{j,t}, \quad \forall j \in [I], t \in \mathcal{P}_j, \quad (\text{EC.113d})$$

$$\sum_{t \in \mathcal{P}_j} x_{j,t} = 1, \quad \forall j \in [I], \quad (\text{EC.113e})$$

$$w_{\mathbf{u},i,j,t} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i, j \in [I], t \in \mathcal{P}_j, \quad (\text{EC.113f})$$

$$\mu_{\mathbf{u},i} \geq 0, \quad \forall \mathbf{u} \in \hat{\mathcal{U}}, i \in [I], \quad (\text{EC.113g})$$

$$x_{j,t} \in \{0, 1\}, \quad \forall j \in [I], t \in \mathcal{P}_j, \quad (\text{EC.113h})$$

where the decision variables have the same meaning as those in formulation (EC.107).

EC.7.4. Complexity of randomization for finite \mathcal{P} , finite \mathcal{U}

In the case that \mathcal{P} and \mathcal{U} are both finite, it is possible to characterize the support size more directly than in the case where \mathcal{U} is convex, by using standard results in linear programming. In particular, we have the following result.

PROPOSITION EC.2. *Suppose that the uncertainty set \mathcal{U} is finite, and that $R(\mathbf{p}, \mathbf{u}) \geq 0$ for all $\mathbf{p} \in \mathcal{P}$ and $\mathbf{u} \in \mathcal{U}$. The RRPO problem $\max_{\boldsymbol{\pi} \in \Delta_{\mathcal{P}}} \min_{\mathbf{u} \in \mathcal{U}} \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$ has an optimal solution $\boldsymbol{\pi}^*$ with a support of size at most $|\mathcal{U}| + 1$.*

Proof: The RRPO problem can be equivalently written as

$$\underset{\mathbf{s}, t, \boldsymbol{\pi}}{\text{maximize}} \quad t \quad (\text{EC.114a})$$

$$\text{subject to} \quad t + s_{\mathbf{u}} = \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}), \quad \forall \mathbf{u} \in \mathcal{U}, \quad (\text{EC.114b})$$

$$\sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} = 1, \quad (\text{EC.114c})$$

$$\pi_{\mathbf{p}} \geq 0, \quad \forall \mathbf{p} \in \mathcal{P}, \quad (\text{EC.114d})$$

$$s_{\mathbf{u}} \geq 0, \quad \forall \mathbf{u} \in \mathcal{U}, \quad (\text{EC.114e})$$

$$t \geq 0, \quad (\text{EC.114f})$$

where $s_{\mathbf{u}}$ can be regarded as a slack variable for the constraint $t - \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u}) \leq 0$ (or equivalently, $t \leq \sum_{\mathbf{p} \in \mathcal{P}} \pi_{\mathbf{p}} R(\mathbf{p}, \mathbf{u})$). Observe that this is a linear program in standard form, with

finitely many equality constraints and finitely many variables. Note that the feasible region is non-empty, because given any $\boldsymbol{\pi} \in \Delta_{\mathcal{P}}$, we can select t and $\mathbf{s} = (s_{\mathbf{u}})_{\mathbf{u} \in \mathcal{U}}$ to obtain a feasible solution to the LP above; note that additionally the problem is bounded, because $\{R(\mathbf{p}, \mathbf{u}) \mid \mathbf{p} \in \mathcal{P}, \mathbf{u} \in \mathcal{U}\}$ is a finite set. Hence, it follows that there exists an optimal solution. Since the problem is in standard form, there is at least one extreme point (Corollary 2.2 of Bertsimas and Tsitsiklis 1997). By Theorem 2.7 of Bertsimas and Tsitsiklis (1997), there must exist an extreme point solution that is optimal. Since extreme points correspond to basic feasible solutions, this optimal solution must be such that at most $|\mathcal{U}| + 1$ coordinates of $(\mathbf{s}, t, \boldsymbol{\pi})$ are non-zero. For the resulting distribution $\boldsymbol{\pi}$, at most $|\mathcal{U}| + 1$ prices therefore have non-zero probability. \square

EC.8. Additional numerical results

EC.8.1. Experiments with discrete \mathcal{U} and log-log and semi-log demand models

In this set of experiments, we consider linear, log-log and semi-log demand models, where uncertainty is modeled through a discrete uncertainty set. We specifically consider a discrete budget uncertainty set \mathcal{U} here:

$$\mathcal{U} = \{\mathbf{u} = \mathbf{u}_0 - (\mathbf{u}_0 - \bar{\mathbf{u}}) \circ \boldsymbol{\xi} - (\mathbf{u}_0 - \underline{\mathbf{u}}) \circ \boldsymbol{\eta} \mid \mathbf{e}^\top \boldsymbol{\xi} + \mathbf{e}^\top \boldsymbol{\eta} \leq \Gamma, \boldsymbol{\xi} + \boldsymbol{\eta} \leq \mathbf{1}, \boldsymbol{\xi}, \boldsymbol{\eta} \in \{0, 1\}^{I+I^2}\}, \quad (\text{EC.115})$$

where $\underline{\mathbf{u}}$ and $\bar{\mathbf{u}}$ are respectively the component-wise lower and upper bounds of \mathbf{u} , \mathbf{u}_0 is the nominal value of the uncertain parameter vector $\mathbf{u} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$, \circ denotes the element-wise product of two vectors, Γ is the budget of uncertainty and $I + I + I(I - 1) = I + I^2$ is the total number of demand model parameters. Under the budget uncertainty set \mathcal{U} , up to Γ parameters can attain their lower bounds or upper bounds, whereas the remaining parameters can only attain their nominal values. We shall assume that the lower bound vector $\underline{\mathbf{u}}$ and upper bound vector $\bar{\mathbf{u}}$ are defined as $\underline{\mathbf{u}} = 0.7\mathbf{u}_0$ and $\bar{\mathbf{u}} = 1.3\mathbf{u}_0$, where \mathbf{u}_0 is the vector of nominal parameters.

For each of the three demand models (linear, semi-log and log-log), we vary the number of products I in $\{5, 10, 15\}$. For each value of I , we generate 24 random instances, where the values of $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ are independently randomly generated as follows:

1. *Linear demand.* Each $\alpha_i \sim \text{Uniform}(100, 200)$, $\beta_i \sim \text{Uniform}(5, 15)$, $\gamma_{i,j} \sim \text{Uniform}(-0.1, +0.1)$.
2. *Semi-log demand.* Each $\alpha_i \sim \text{Uniform}(8, 10)$, $\beta_i \sim \text{Uniform}(1.5, 2)$, $\gamma_{i,j} \sim \text{Uniform}(-0.5, +0.5)$.
3. *Log-log demand.* Each $\alpha_i \sim \text{Uniform}(10, 14)$, $\beta_i \sim \text{Uniform}(1.5, 2)$, $\gamma_{i,j} \sim \text{Uniform}(-0.8, +0.8)$.

We set the price set of each $i \in [I]$ as $\mathcal{P}_i = \{1, 2, 3, 4, 5\}$.

For each instance, we solve the nominal problem, the DRPO problem and the RRPO problem. For both RRPO and DRPO, we test a different collection of Γ values for the uncertainty set depending on the value of I .

To solve the RRPO problem for each instance, we execute the double column generation algorithm described in Section EC.7. For the linear demand model, we solve both primal and dual separation problems as mixed-integer programs in Gurobi.

For the log-log and semi-log demand models, we proceed differently. In our preliminary experimentation with the restricted dual problem, we observed that exactly solving the dual separation problem (EC.107) (for semi-log demand) or (EC.113) (for log-log demand) via Mosek takes quite a long time. Therefore, to reduce the computation time of RRPO with discrete \mathcal{U} , we instead use a random improvement heuristic to obtain the solution of dual separation problem. Specifically, we randomly select a price vector \mathbf{p}^0 as a starting point. We start with changing the price of product $i = 1$ and keeping the prices of all other products unchanged, to search for a price vector \mathbf{p}^1 that makes the objective value of the dual separation problem the largest. Then based on the current price vector \mathbf{p}^1 , we change the price of product $i = 2$ and keep the prices of all other products unchanged, to search for a better price vector \mathbf{p}^2 . We repeat this for all of the products, yielding the price vector \mathbf{p}^I . We repeat this procedure with 100 random starting points, and retain the best solution over these 100 repetitions. We note that in the heuristic form of the double column generation method, we continue to solve the primal separation problem (over $\mathbf{u} \in \mathcal{U}$) exactly using Mosek. The lower bound given by the restricted primal, $Z_{P,\hat{P}}$, therefore remains a valid lower bound on the true optimal RRPO objective, and will correspond to the worst-case expected revenue of a possibly suboptimal price distribution $\boldsymbol{\pi}$, at every iteration of the double column generation method. At termination, the objective value returned by the algorithm will therefore be a lower bound on the true optimal RRPO objective.

Although this approximate method cannot guarantee that the overall double column generation procedure converges to a provably optimal solution, our preliminary experimentation with small instances suggests that it obtains the exact solution of RRPO that one would obtain if the dual separation problem were solved to provable optimality. In Section EC.8.2 of the ecompanion, we undertake an in-depth comparison of this heuristic form of double column generation against the exact double column generation approach, where both separation problems are solved to provable optimality using Mosek. When one imposes a time limit on the exact approach, we find that the heuristic approach returns solutions of comparable quality (in cases where the exact approach terminates with the optimal solution), or better (in cases where the exact approach exhausts the time limit and does not return an optimal solution), and in general, the heuristic approach terminates in significantly less time than the exact approach.

With regard to the DRPO problem for each log-log and semi-log instance, we note that we do not have a solution algorithm or formulation to solve it exactly. Therefore, we again use the same random improvement heuristic to obtain an approximate solution of DRPO with these demand

models. We randomly pick a starting price vector, and change the price of one product at a time to improve the worst case objective value until we no longer get an improvement. We repeat this procedure 50 times and select the best resulting price vector from these 50 repetitions as the approximate solution of DRPO. We note that we use a smaller number of repetitions because each repetition involves solving worst-case problem over $\mathbf{u} \in \mathcal{U}$ repeatedly in order to evaluate the robust objective of each candidate price vector; this contributes to a large overall computation time for this approach. We denote this approximate solution by $\hat{\mathbf{p}}_{\text{DR}}$. With regard to the DRPO problem for linear demand, we observe that the objective function of DRPO is linear in the uncertain parameter vector \mathbf{u} , and that the description of the set polyhedron (EC.115) is integral (i.e., extreme points of this polyhedron naturally correspond to $\boldsymbol{\xi}, \boldsymbol{\eta} \in \{0, 1\}^{2I+I^2}$). Therefore, DRPO can be solved exactly by relaxing the requirement $\boldsymbol{\xi}, \boldsymbol{\eta} \in \{0, 1\}^{2I+I^2}$ in the uncertainty set (EC.115), and reformulating the worst-case objective using LP duality, leading to a mixed-integer linear program.

Lastly, for the nominal problem for each instance, we either formulate it as a mixed-integer linear program (in the linear demand case) or use the biconjugate technique to formulate it as a mixed-integer exponential cone program (in the log-log and semi-log demand cases).

We report the same metrics as in Section 7.1, with several modifications for the log-log and semi-log demand models. For those models, we use \hat{Z}_{DR} and $R(\hat{\mathbf{p}}_{\text{DR}}, \mathbf{u}_0)$ to denote the approximate objective value of the DRPO problem and revenue of the approximate DRPO solution $\hat{\mathbf{p}}_{\text{DR}}$ under the nominal parameter vector \mathbf{u}_0 . We also use \hat{Z}_{RR} to denote the approximate objective of the RRPO problem, and $\mathbb{E}[R(\hat{\mathbf{p}}_{\text{RR}}, \mathbf{u}_0)]$ denotes the expected revenue of the resulting randomized price vector $\hat{\mathbf{p}}_{\text{RR}}$ under the nominal demand model parameter vector \mathbf{u}_0 . Finally, the approximate relative improvement of RRPO over DRPO is defined as $\hat{\text{RI}} = (\hat{Z}_{\text{RR}} - \hat{Z}_{\text{DR}}) / \hat{Z}_{\text{DR}} \times 100\%$.

Table EC.3 shows the results for the linear demand model. Here, we interestingly find that the vast majority of instances are randomization-proof, i.e., the average RI is below 1%, if not exactly 0%. We note here that we tested other families of instances where $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ and \mathcal{P}_i are generated differently, but in virtually every case we found that the relative improvement of randomized over deterministic robust pricing was very small. These results, together with those for the convex \mathcal{U} case, suggest that randomized pricing is of limited benefit compared to deterministic pricing for the uncertain linear demand model case.

Tables EC.4 and EC.5 show how the results vary for different values of discrete uncertainty budget Γ for semi-log and log-log demand.² We can see that, in most of the cases we test, the

² We note here that for the log-log model, we encountered one instance ($I = 10, \Gamma = 44$) where \hat{Z}_{DR} was higher than \hat{Z}_{RR} ; in general, when solved to perfect optimality, one should see Z_{RR}^* should be higher than Z_{DR}^* . We have verified that the reason for this anomaly was a numerical error in the solution of the worst-case subproblem in Mosek within the DRPO random improvement heuristic. This instance is omitted in our calculation of \hat{Z}_{DR} , $\hat{\text{RI}}$ and $\mathbb{E}[R(\hat{\mathbf{p}}_{\text{DR}}, \mathbf{u}_0)]$, and the affected entries are indicated by * in Table EC.5.

randomized robust pricing strategy provides a substantial benefit over the deterministic robust price solution. The percentage improvement given by randomization ranges from 0% to as much as 488.59% for semi-log instances, and from 0% to 175.18% for log-log instances. Similar to the cases with convex \mathcal{U} , both \hat{Z}_{RR} and \hat{Z}_{DR} decrease as the uncertainty set becomes larger. While the RI metric generally decreases as Γ increases, in some instances it can be increasing in Γ at small values of Γ (this is visible in the average results metrics for $I = 10$ with semi-log demand). When Γ is large enough, the RI metric often becomes very small or even zero. This makes sense when interpreted through Corollary 2. Specifically, when nature is able to make a large number of demand model parameters take their worst values, it is likely that at the \mathbf{u}^* at which the optimal objective of DRPO is attained is such that the price vector for the nominal problem with \mathbf{u}^* coincides with the optimal price vector for DRPO. Thus, by Corollary 2, the problem will be randomization-proof.

With regard to the computation time, the computation time for both RRPO and DRPO increases with I . Interestingly, the computation time required by RRPO does not necessarily increase as the discrete uncertainty budget Γ increases; in some cases, when Γ is large, the RRPO solution degenerates to the DRPO solution, allowing the double column generation algorithm to terminate quickly. By comparing t_{RR} and t_{DR} , we can see that RRPO in general takes less time than DRPO. The computation time of the RRPO problem in semi-log instances is no more than approximately two minutes on average ($I = 15$, $\Gamma = 36$), while in log-log instances, solving RRPO requires no more than 1.5 minutes on average ($I = 15$, $\Gamma = 18$). Lastly, for linear demand, the computation time for RRPO is extremely small, requiring no more than a few seconds on average.

EC.8.2. Comparison of heuristic and exact solutions of RRPO in the finite \mathcal{P} , finite \mathcal{U} case

In this section, we perform a detailed comparison of the performance of the heuristic form of RRPO, where the dual separation problem is solved approximately using the random improvement heuristic, against the exact form of RRPO, where both the primal and dual separation problem are solved exactly. We consider the log-log and semi-log demand models, and we test the same instances described in Section EC.8.1.

For the exact approach, we incorporated the following termination logic. When the relative gap between the lower bound from the restricted primal and the upper bound from the restricted dual is within 10^{-6} . In certain cases, due to numerical precision issues arising from computing the log-log and semi-log objective functions and solving the primal and dual separation problems within Mosek, we would alternatively terminate when the set of price vectors $\hat{\mathcal{P}}$ no longer changes (i.e., when the dual separation problem is unable to produce a price vector that is not contained in $\hat{\mathcal{P}}$). Finally, we also imposed a time limit of 3600 seconds for the double column generation algorithm,

I	Γ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{RR}^*, \mathbf{u}_0)]$	t_{DR}	Z_{DR}^*	RI(%)	$R(\mathbf{p}_{DR}^*, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
5	3	8.76	1724.38	2458.17	0.43	1723.46	0.06	2462.66	0.32	2473.30	1719.29
5	6	0.27	1316.39	2382.10	0.04	1313.68	0.23	2383.35	—	—	1261.84
5	9	0.27	1159.33	2294.44	0.03	1157.98	0.12	2293.84	—	—	1039.24
5	12	0.19	1126.85	2259.05	0.03	1126.85	0.00	2259.05	—	—	987.29
5	18	0.20	1124.87	2256.23	0.03	1124.87	0.00	2256.23	—	—	984.10
5	24	0.17	1123.78	2256.23	0.03	1123.78	0.00	2256.23	—	—	982.18
10	5	0.53	3717.70	4990.89	0.11	3714.08	0.10	4997.68	0.09	5009.03	3711.50
10	7	0.59	3315.59	4972.42	0.11	3312.91	0.08	4973.49	—	—	3297.07
10	9	0.70	2982.02	4941.26	0.11	2979.68	0.08	4944.40	—	—	2942.80
10	14	1.35	2589.70	4782.85	0.12	2583.93	0.23	4791.83	—	—	2437.09
10	19	0.84	2372.37	4662.16	0.11	2371.21	0.05	4657.17	—	—	2133.38
10	26	0.56	2342.57	4636.61	0.10	2342.57	0.00	4636.61	—	—	2086.03
10	33	0.57	2339.18	4633.87	0.11	2339.18	0.00	4636.61	—	—	2081.46
10	44	0.57	2334.90	4627.78	0.10	2334.90	0.00	4627.78	—	—	2075.15
15	6	0.92	5920.62	7495.77	0.22	5918.51	0.04	7506.26	0.17	7513.95	5917.60
15	12	1.79	4706.04	7435.56	0.25	4701.16	0.11	7442.32	—	—	4668.21
15	18	3.16	4051.33	7250.57	0.27	4045.35	0.15	7248.00	—	—	3889.25
15	24	4.50	3722.40	7071.15	0.29	3713.86	0.23	7068.49	—	—	3428.59
15	36	1.12	3500.73	6889.94	0.21	3500.73	0.00	6889.47	—	—	3112.83

Table EC.3 Results for linear instances with discrete \mathcal{U} .

I	Γ	t_{RR}	\hat{Z}_{RR}	$\mathbb{E}[R(\hat{\mathbf{p}}_{RR}, \mathbf{u}_0)]$	t_{DR}	\hat{Z}_{DR}	\hat{RI} (%)	$R(\hat{\mathbf{p}}_{DR}, \mathbf{u}_0)$	t_N	Z_N^*	$Z_{N,WC}$
5	3	13.23	8.65×10^3	1.57×10^5	20.35	5.56×10^3	63.01	2.17×10^5	0.93	2.27×10^5	5.04×10^3
5	6	0.43	3.06×10^3	1.81×10^5	20.52	1.92×10^3	66.89	2.20×10^5	—	—	1.78×10^3
5	9	0.55	1.84×10^3	2.09×10^5	22.96	1.64×10^3	18.42	2.22×10^5	—	—	1.63×10^3
5	12	0.63	1.65×10^3	2.22×10^5	25.53	1.60×10^3	4.39	2.25×10^5	—	—	1.60×10^3
5	18	0.27	1.59×10^3	2.27×10^5	23.87	1.59×10^3	0.11	2.27×10^5	—	—	1.59×10^3
5	24	0.08	1.59×10^3	2.27×10^5	13.53	1.59×10^3	0.00	2.27×10^5	—	—	1.59×10^3
10	5	1.59	1.21×10^6	5.30×10^7	117.69	4.95×10^5	173.47	6.73×10^7	0.15	7.16×10^7	4.48×10^5
10	7	2.59	6.50×10^5	4.10×10^7	119.28	1.87×10^5	233.11	5.69×10^7	—	—	1.88×10^5
10	9	3.49	4.32×10^5	3.64×10^7	121.43	1.09×10^5	250.12	5.41×10^7	—	—	1.05×10^5
10	14	6.44	1.80×10^5	3.67×10^7	180.63	6.69×10^4	161.08	6.96×10^7	—	—	6.78×10^4
10	19	10.23	9.63×10^4	5.29×10^7	362.29	6.36×10^4	67.80	7.15×10^7	—	—	6.36×10^4
10	26	12.14	6.69×10^4	6.40×10^7	559.63	6.32×10^4	19.70	7.15×10^7	—	—	6.33×10^4
10	33	11.37	6.37×10^4	7.11×10^7	699.06	6.33×10^4	6.47	7.15×10^7	—	—	6.33×10^4
10	44	7.84	6.33×10^4	7.15×10^7	799.81	6.33×10^4	0.09	7.16×10^7	—	—	6.33×10^4
15	6	37.63	3.55×10^8	3.22×10^9	331.13	1.59×10^7	488.59	4.52×10^9	0.44	5.05×10^9	1.42×10^7
15	12	18.86	8.69×10^6	3.24×10^9	344.94	2.15×10^6	471.14	3.51×10^9	—	—	1.77×10^6
15	18	33.08	3.11×10^6	3.08×10^9	653.64	1.18×10^6	323.61	4.91×10^9	—	—	1.16×10^6
15	24	48.52	1.75×10^6	3.67×10^9	1525.41	1.12×10^6	165.75	5.03×10^9	—	—	1.11×10^6
15	36	101.38	1.21×10^6	4.82×10^9	3400.17	1.1×10^6	38.58	5.02×10^9	—	—	1.10×10^6

Table EC.4 Results for semi-log instances with discrete \mathcal{U} .

i.e., we exit from the loop in Algorithm 3 if the time limit exceeds 3600 seconds. We note here that the first two termination criteria were also implemented for the heuristic approach.

In Tables EC.6 and EC.7, we display the optimality gap G_E for the exact approach, the computation time t_E for the exact approach, the computation time t_H for the heuristic approach, and

I	Γ	t_{RR}	\hat{Z}_{RR}	$\mathbb{E}[R(\hat{\mathbf{p}}_{\text{RR}}, \mathbf{u}_0)]$	t_{DR}	\hat{Z}_{DR}	$\hat{\text{RI}}$ (%)	$R(\hat{\mathbf{p}}_{\text{DR}}, \mathbf{u}_0)$	t_{N}	Z_{N}^*	$Z_{\text{N,WC}}$
5	3	13.33	3.26×10^5	1.86×10^6	31.23	2.80×10^5	22.03	1.89×10^6	0.96	4.31×10^6	1.49×10^5
5	6	0.20	6.37×10^4	4.03×10^6	20.39	6.15×10^4	3.16	4.29×10^6	—	—	6.1×10^4
5	9	0.27	4.91×10^4	4.19×10^6	18.39	4.87×10^4	0.65	4.24×10^6	—	—	4.77×10^4
5	12	0.35	4.71×10^4	4.20×10^6	16.82	4.70×10^4	0.20	4.18×10^6	—	—	4.57×10^4
5	18	0.17	4.66×10^4	4.18×10^6	11.67	4.66×10^4	0.01	4.18×10^6	—	—	4.52×10^4
5	24	0.14	4.66×10^4	4.18×10^6	8.73	4.66×10^4	0.05	4.16×10^6	—	—	4.52×10^4
10	5	4.64	2.33×10^6	3.89×10^7	161.57	1.38×10^6	83.54	6.45×10^7	0.31	7.66×10^7	1.23×10^6
10	7	5.39	1.31×10^6	5.22×10^7	165.21	8.59×10^5	63.34	7.12×10^7	—	—	8.39×10^5
10	9	5.30	8.74×10^5	5.15×10^7	149.63	6.23×10^5	38.99	7.28×10^7	—	—	6.12×10^5
10	14	5.10	5.07×10^5	5.67×10^7	138.67	3.82×10^5	29.31	7.30×10^7	—	—	3.80×10^5
10	19	3.52	3.74×10^5	6.35×10^7	136.70	3.31×10^5	12.90	7.51×10^7	—	—	3.33×10^5
10	26	3.56	3.27×10^5	7.38×10^7	138.77	3.18×10^5	4.28	7.59×10^7	—	—	3.21×10^5
10	33	4.79	3.18×10^5	7.60×10^7	137.07	3.15×10^5	2.34	7.59×10^7	—	—	3.17×10^5
10	44	3.19	3.15×10^5	7.62×10^7	131.47	$3.19 \times 10^{5*}$	1.19*	$7.53 \times 10^{7*}$	—	—	3.14×10^5
15	6	35.68	1.71×10^7	4.33×10^8	552.64	7.75×10^6	175.18	7.83×10^8	1.20	8.38×10^8	7.20×10^6
15	12	62.70	5.55×10^6	4.93×10^8	582.20	2.94×10^6	102.06	7.78×10^8	—	—	2.76×10^6
15	18	76.60	2.92×10^6	4.55×10^8	561.96	1.92×10^6	65.73	8.24×10^8	—	—	1.88×10^6
15	24	54.30	2.03×10^6	6.79×10^8	572.08	1.71×10^6	32.76	8.19×10^8	—	—	1.71×10^6
15	36	48.34	1.69×10^6	8.11×10^8	600.30	1.63×10^6	8.07	8.32×10^8	—	—	1.63×10^6

Table EC.5 Results for log-log instances with discrete \mathcal{U} .

the relative improvement $RI_{E \rightarrow H}$ of the heuristic approach over the exact approach, for the log-log and semi-log demand models respectively. The optimality gap G_E is defined as

$$G_E = 100\% \times \frac{Z_{\text{RR},E,UB}^* - Z_{\text{RR},E,LB}^*}{Z_{\text{RR},E,UB}^*}, \quad (\text{EC.116})$$

where $Z_{\text{RR},E,UB}^*$ and $Z_{\text{RR},E,LB}^*$ are the final upper and lower bounds from the double column generation approach. The relative improvement $RI_{E \rightarrow H}$ is defined as

$$RI_{E \rightarrow H} = 100\% \times \frac{Z_{\text{RR},H,LB}^* - Z_{\text{RR},E,LB}^*}{Z_{\text{RR},E,LB}^*}, \quad (\text{EC.117})$$

where $Z_{\text{RR},H,LB}^*$ is the final lower bound from the heuristic approach. The relative improvement metric measures how much the price distribution from the heuristic double CG method improves on the price distribution from the exact double CG method. Note that this metric can be negative, in cases where the heuristic approach terminates with a suboptimal solution to the RRPO problem.

Before continuing, we note that in some instances, we found the gap G_E to be either slightly larger than the tolerance of 10^{-6} , or otherwise to be a very small negative number; both of these phenomena are due to the aforementioned numerical issues we encountered. We also note that for larger instances, the computation time sometimes significantly exceeds 3600 seconds. As noted before, the outer loop of the double CG algorithm is terminated when the run time exceeds 3600 seconds, and in some cases, the final iteration of the outer loop before this criterion is triggered can require a huge amount of time (due to solving both the restricted primal and restricted dual problems, each of which can involve multiple solves of the primal and dual separation problems).

For the log-log results shown in Table EC.6, we see that in general, the exact double CG algorithm terminates with a provably optimal solution, as the average optimality gap is very close to zero across all values of I and Γ . With regard to the heuristic double CG method, we can see the average RI metric is in general zero or no smaller than -0.36%. This implies that the heuristic double CG method returns solutions that are in general either the same or only very slightly suboptimal compared to the exact double CG solution. We additionally see that the average computation time of the heuristic double CG method is much smaller than that of the exact double CG method. For example, for $I = 15$, the average computation time of the exact double CG method can range anywhere from around 15 minutes to about 45 minutes, whereas for the heuristic double CG method it is no more than 1.5 minutes. These results indicate that for the log-log model, the heuristic method can obtain solutions of essentially the same quality as the exact method but in substantially less time.

For the semi-log results shown in Table EC.7, we see similar behavior as we do for the log-log models for $I \in \{5, 10\}$. For $I = 15$, we find that the exact double CG method in general does not terminate with a provably optimal solution (i.e., a gap of 0%) under the soft 3600 second time limit. The average optimality gap for the exact method ranges from about 14% to 40%, and the average computation time can be as high as 90 minutes. In contrast, the average computation time for the heuristic double CG method is no more than two minutes. In addition, for $\Gamma \geq 12$, the heuristic solution has an objective value for which the average improvement over the (suboptimal) exact double CG solution ranges from approximately 6% to 20%. This highlights that in settings where the exact double CG method requires a large amount of computation time and must be terminated prematurely, the heuristic method obtains higher quality solutions than the exact method.

EC.8.3. Effect of parameter bounds on relative improvement in discrete \mathcal{U} experiments

In this section, we explore the effect of the width of the budget uncertainty set in our experiments in Section EC.8.1. Recall that in those experiments, we assumed a budget uncertainty set, where $\bar{\mathbf{u}}$ and $\underline{\mathbf{u}}$ denote the vectors of upper and lower bounds, respectively, for the uncertain parameter vector \mathbf{u} , and these parameter vectors were defined as $\bar{\mathbf{u}} = 1.3\mathbf{u}_0$, $\underline{\mathbf{u}} = 0.7\mathbf{u}_0$. By setting the bounds in this way, we assume that each uncertain parameter can vary by +30% or -30% relative to its nominal value.

We now consider setting these bound vectors as

$$\bar{\mathbf{u}} = (1 + \delta)\mathbf{u}_0, \tag{EC.118}$$

$$\underline{\mathbf{u}} = (1 - \delta)\mathbf{u}_0, \tag{EC.119}$$

I	Γ	G_E	t_E	t_H	$\text{RI}_{E \rightarrow H}$
5	3	0.00	16.16	13.33	0.00
5	6	0.00	1.58	0.20	0.00
5	9	-0.34	2.21	0.27	-0.35
5	12	-0.06	2.45	0.35	-0.07
5	18	-0.03	1.88	0.17	-0.05
5	24	0.00	1.49	0.14	0.00
10	5	0.00	63.35	4.64	-0.12
10	7	0.01	77.43	5.39	0.00
10	9	0.01	80.18	5.30	0.00
10	14	0.01	83.99	5.10	-0.01
10	19	0.01	72.05	3.52	-0.03
10	26	0.01	74.75	3.56	0.00
10	33	0.01	70.30	4.79	0.00
10	44	0.01	68.23	3.19	0.00
15	6	0.01	952.93	35.68	-0.25
15	12	0.02	2356.89	62.70	-0.30
15	18	0.23	2739.09	76.60	-0.36
15	24	0.04	2033.25	54.30	-0.24
15	36	0.04	1351.31	48.34	-0.02

Table EC.6 Comparison of exact double CG and heuristic double CG methods for log-log demand model instances from Section EC.8.1.

I	Γ	G_E	t_E	t_H	$\text{RI}_{E \rightarrow H}$
5	3	0.00	18.30	13.23	0.00
5	6	0.00	7.57	0.43	0.00
5	9	0.00	8.52	0.55	0.00
5	12	0.00	7.64	0.63	0.00
5	18	0.00	4.17	0.27	0.00
5	24	0.00	1.91	0.08	0.00
10	5	0.00	151.81	1.59	-0.01
10	7	0.01	344.41	2.59	0.00
10	9	0.02	585.85	3.49	0.00
10	14	0.04	1279.63	6.44	-0.29
10	19	0.07	1845.95	10.23	0.02
10	26	0.04	1526.40	12.14	0.00
10	33	0.51	1084.73	11.37	0.07
10	44	0.02	686.13	7.84	0.00
15	6	0.19	1930.95	37.63	0.14
15	12	14.02	4163.44	18.86	5.82
15	18	38.81	5119.94	33.08	19.47
15	24	39.67	4808.48	48.52	19.86
15	36	37.17	4508.40	101.38	15.97

Table EC.7 Comparison of exact double CG and heuristic double CG methods for semi-log demand model instances from Section EC.8.1.

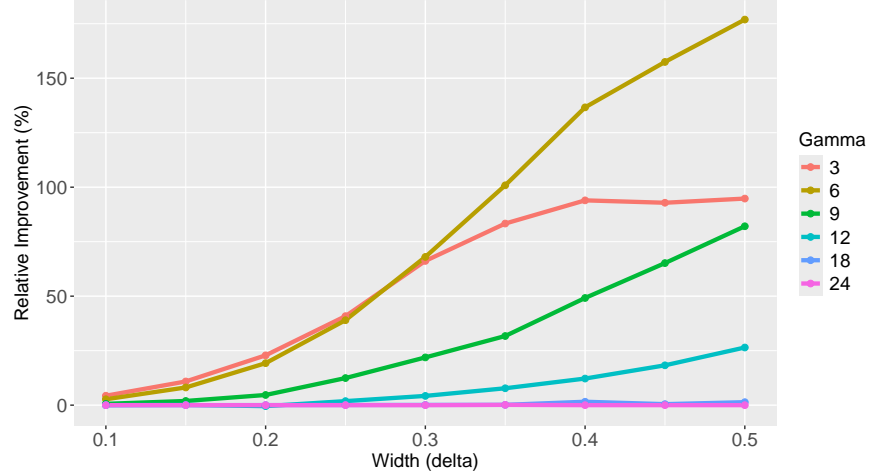


Figure EC.1 Plot of RI versus δ for semi-log demand models.

where $\delta \in (0, 1)$ is the *width parameter*.

We consider the same log-log and semi-log demand model instances from Section EC.8.1, and allow for δ to vary in the set $\{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5\}$. For simplicity, we restrict our focus to instances with $I = 5$, and allow the uncertainty set budget Γ to vary in the same way as before, i.e., $\Gamma \in \{3, 6, 9, 12, 18, 24\}$. For each instance, we solve the RRPO problem and DRPO problem using the same methods used in Section EC.8.1 (i.e., the heuristic double column generation approach and the random improvement heuristic, respectively). We calculate the average value of the RI metric across all 24 instances for each Γ and δ .

For the semi-log demand model, Figure EC.1 plots the RI metric against δ for different values of Γ , while Figure EC.2 plots the RI metric against Γ for different values of δ . In general, we can see that the relative improvement of the randomized robust price distribution over the deterministic robust price vector increases as the width parameter δ increases, suggesting that randomization becomes more beneficial when there is more uncertainty in each individual parameter. Interestingly, as observed in Section EC.8.1, the RI metric is initially increasing with Γ but decreases after a certain point, and this peak seems to be the same across all values of δ . Lastly, another interesting insight is that as Γ increases, the rate at which the RI increases with δ is lower, and for sufficiently high Γ values, the RI seems to stay very close to zero across all δ values.

For the log-log demand model, Figure EC.3 plots the RI metric against δ for different values of Γ , and Figure EC.4 plots the RI metric against Γ for different values of δ . Like in the semi-log case, the benefit of randomization over the deterministic robust solution does appear to increase as δ increases. The rate at which the RI increases becomes smaller when Γ is higher, and for the highest values of Γ ($\Gamma = 18$ or $\Gamma = 24$), changing δ (at least within the 0.1 to 0.5 range) does not seem to change the RI.

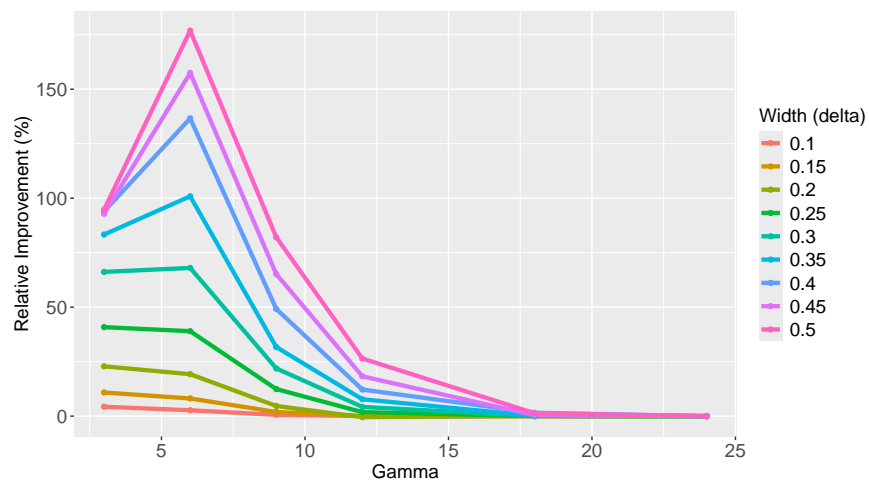


Figure EC.2 Plot of RI versus Γ for semi-log demand models.

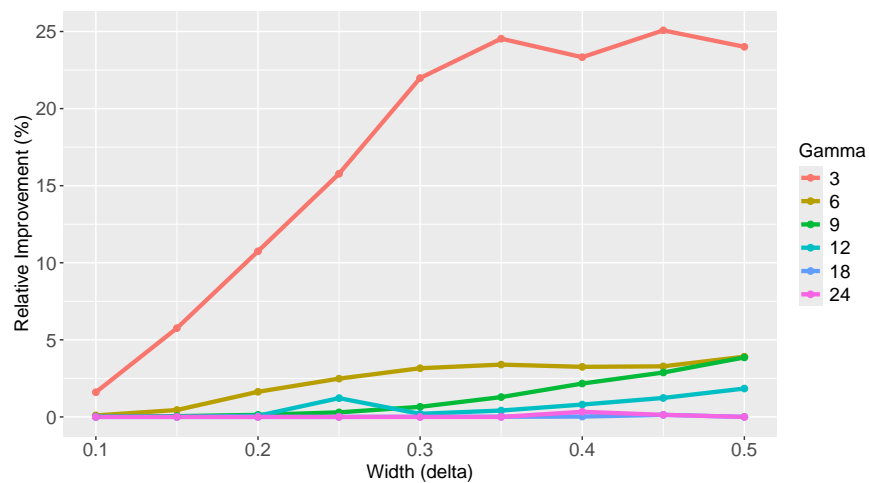


Figure EC.3 Plot of RI versus δ for log-log demand models.

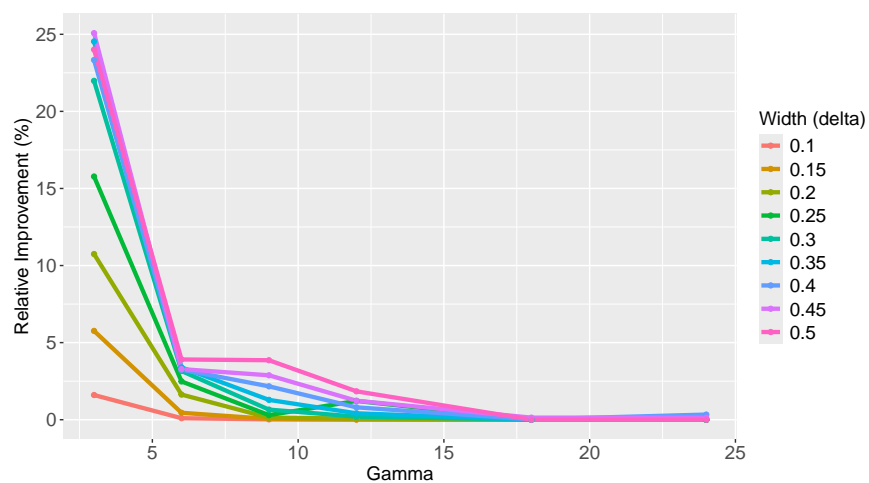


Figure EC.4 Plot of RI versus Γ for log-log demand models.

EC.8.4. Estimation results for orangeJuice data set

Tables EC.8 and EC.9 display the point estimates of α , β and γ for the semi-log and log-log demand models for the orangeJuice data set.

Parameters	Product										
	1	2	3	4	5	6	7	8	9	10	11
$\alpha_{\text{semi-log}}$	9.873	9.829	8.598	9.504	9.024	9.828	8.582	7.901	7.152	11.161	10.896
$\beta_{\text{semi-log}}$	1.0222	0.4581	1.2735	1.7888	1.3354	0.6507	1.6491	1.3945	2.0809	1.6290	0.0383
$\alpha_{\text{log-log}}$	10.140	10.956	8.266	8.421	9.045	10.613	7.832	7.127	6.563	11.326	11.198
$\beta_{\text{log-log}}$	2.7195	2.0410	3.3037	3.8855	2.9357	2.6101	3.6063	2.8209	3.9717	2.7942	0.1542

Table EC.8 Estimation results for α and β .

$\gamma_{\text{semi-log}}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$	$j = 9$	$j = 10$	$j = 11$
$i = 1$	–	0.0571	0.0813	0.0966	0.0193	-0.0232	0.1305	0.1904	0.1490	0.0582	0.0815
$i = 2$	0.1384	–	0.0041	0.0009	0.0204	0.0153	0.0090	0.1040	-0.0023	0.0491	0.0394
$i = 3$	0.3386	0.0916	–	0.1943	0.0702	-0.0062	0.0051	0.0950	-0.0310	0.0690	0.0950
$i = 4$	0.4313	0.0976	-0.1112	–	0.4089	0.3518	0.2085	-0.0777	-0.0352	0.0383	-0.2290
$i = 5$	0.1916	0.0490	0.3026	0.2966	–	-0.1538	0.1547	-0.0314	0.1034	0.3338	0.0370
$i = 6$	0.0211	0.0493	-0.0194	-0.0018	0.0888	–	0.0340	0.0472	-0.0167	0.0297	0.1119
$i = 7$	0.2007	0.0388	0.0706	0.0672	0.3233	0.0837	–	0.0377	0.2216	-0.0504	0.1405
$i = 8$	0.0117	0.0119	0.0932	0.0757	0.1023	-0.0160	0.1345	–	0.1372	0.2143	0.2699
$i = 9$	0.0955	0.0373	-0.0211	0.3651	0.4176	0.0358	0.2127	0.1462	–	0.2337	0.1627
$i = 10$	0.0412	-0.3941	0.0764	0.4867	0.4810	0.0109	-0.0814	-0.1047	0.0878	–	0.0274
$i = 11$	-0.0893	-0.1587	-0.1358	-0.0252	-0.0690	0.0079	-0.0574	-0.1117	-0.1271	0.0809	–
$\gamma_{\text{log-log}}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$	$j = 9$	$j = 10$	$j = 11$
$i = 1$	–	0.2196	0.1631	0.2129	0.0646	-0.0577	0.2576	0.3338	0.2494	0.0621	0.2939
$i = 2$	0.3474	–	0.0403	0.0004	0.0338	0.0492	0.0193	0.1879	-0.0042	0.0739	0.1257
$i = 3$	0.8673	0.5123	–	0.4400	0.1482	-0.0338	0.0527	0.1480	-0.1001	0.1267	0.3683
$i = 4$	1.1581	0.3822	-0.2283	–	0.8367	1.2659	0.4495	-0.1569	-0.0115	0.1003	-0.7321
$i = 5$	0.4624	0.2241	0.8344	0.6406	–	-0.6800	0.3223	0.0646	0.1426	0.5815	0.1782
$i = 6$	0.0462	0.2424	-0.0343	-0.0173	0.2086	–	0.0975	0.1187	-0.0364	0.0561	0.4159
$i = 7$	0.4644	0.2531	0.0971	0.1204	0.6997	0.2946	–	0.1728	0.4912	-0.0564	0.4497
$i = 8$	0.0652	0.1430	0.1980	0.1587	0.2705	-0.0988	0.3198	–	0.3034	0.2992	0.9436
$i = 9$	0.2971	-0.1190	-0.0216	0.7986	0.8825	0.3045	0.5869	0.1706	0.0	0.3171	0.4223
$i = 10$	0.1406	-1.7987	0.1061	1.0453	1.0852	0.0811	-0.1213	-0.1760	0.0454	–	-0.0371
$i = 11$	-0.2246	-0.7519	-0.3177	-0.0483	-0.1635	-0.0570	-0.1100	-0.1780	-0.2646	0.1341	–

Table EC.9 Estimation results for γ for orangeJuice data set.**EC.8.5. Performance results for orangeJuice data set**

Tables EC.10 and EC.11 below compare the performance of the nominal, deterministic robust and randomized robust pricing solutions under a discrete budget uncertainty set for the orangeJuice data set.

Γ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{\text{RR}}^*, \mathbf{u}_0)]$	t_{DR}	\hat{Z}_{DR}	RI(%)	$R(\hat{\mathbf{p}}_{\text{DR}}, \mathbf{u}_0)$	t_{N}	Z_{N}^*	$Z_{\text{N,WC}}$
5	14.64	162753.97	290939.28	225.36	102626.41	58.59	260321.17	0.81	590547.01	85304.36
10	6.29	70401.48	404458.22	208.42	47969.46	46.76	350396.25	–	–	38815.61
15	3.93	39567.50	349936.30	209.14	32757.43	20.79	334211.84	–	–	22798.44
20	16.70	31438.76	328664.19	197.37	25348.77	24.02	299970.20	–	–	15940.16

Table EC.10 Results for orangeJuice pricing problem with semi-log demand and discrete \mathcal{U} .

Γ	t_{RR}	Z_{RR}^*	$\mathbb{E}[R(\mathbf{p}_{\text{RR}}^*, \mathbf{u}_0)]$	t_{DR}	\hat{Z}_{DR}	RI(%)	$R(\hat{\mathbf{p}}_{\text{DR}}, \mathbf{u}_0)$	t_{N}	Z_{N}^*	$Z_{\text{N,WC}}$
5	12.85	272399.89	605265.00	306.63	174478.12	56.12	811254.69	0.87	1110000.00	117186.58
10	10.41	135761.31	750084.92	260.96	77297.42	75.63	896972.12	–	–	50458.15
15	15.59	72930.45	761785.89	193.13	44914.20	62.38	896972.12	–	–	27389.15
20	8.56	45153.74	770505.32	190.76	27675.07	63.16	409330.70	–	–	17502.32

Table EC.11 Results for orangeJuice pricing problem with log-log demand and discrete \mathcal{U} .