informs
http://pubsonline.informs.org/journal/mnsc

MANAGEMENT SCIENCE
Vol. 67, No. 4, April 2021, pp. 2340–2364
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

Promotion Optimization for Multiple Items in Supermarkets

Maxime C. Cohen,[a] Jeremy J. Kalas,[b] Georgia Perakis[c]

[a] Desautels Faculty of Management, McGill University, Montreal, Quebec H3A 1G5, Canada; [b] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139; [c] Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Contact: maxime.cohen@mcgill.ca, http://orcid.org/0000-0002-2474-3875 (MCC); kalas.jeremy@gmail.com (JJK); georgiap@mit.edu, http://orcid.org/0000-0002-0888-9030 (GP)

Received: October 29, 2017
Revised: August 27, 2019; March 3, 2020
Accepted: March 12, 2020
Published Online in Articles in Advance: June 8, 2020

https://doi.org/10.1287/mnsc.2020.3641

Abstract. Promotions are a critical decision for supermarket managers, who must decide the price promotions for a large number of items. Retailers often use promotions to boost the sales of the different items by leveraging the cross-item effects. We formulate the promotion optimization problem for multiple items as a nonlinear integer program. Our formulation includes several business rules as constraints. Our demand models can be estimated from data and capture the postpromotion dip effect and cross-item effects (substitution and complementarity). Because demand functions are typically nonlinear, the exact formulation is intractable. To address this issue, we propose a general class of integer programming approximations. For demand models with additive cross-item effects, we prove that it is sufficient to account for unilateral and pairwise contributions and derive parametric bounds on the performance of the approximation. We also show that the unconstrained problem can be solved efficiently via a linear program when items are substitutable and the price set has two values. For more general cases, we develop efficient rounding schemes to obtain an integer solution. We conclude by testing our method on realistic instances and convey the potential practical impact for retailers.

History: Accepted by Yinyu Ye, optimization.
Funding: This research was funded by the National Science Foundation [Grants CMMI-1162034 and CMMI-1563343] and the Oracle Corporation [ERO Foundation Grant].

Keywords: promotions • dynamic pricing • retail analytics • integer optimization

1. Introduction

Retailers have access to several levers to increase sales and profits. One of these levers is price promotions. For example, in a supermarket, the retailer offers thousands of different items and must decide the price promotions for each item in each period. Setting the right promotions is critical because it can directly impact the retailer's profitability in an industry where profit margins are very small. It is common for retailers to have thousands of promotions simultaneously. Indeed, offering promotions can help retailers capture different segments of customers and stimulate traffic and demand in the store.

To this day, many supermarket managers are still deciding promotions manually by using a combination of experience and domain knowledge. The substantial amount of available data provides an opportunity to develop data-driven tools that can guide retailers in deciding on promotions at scale. Retailers need to set the price promotions for thousands of items while satisfying the relevant business rules. In this paper, we develop a model directly inspired by a collaboration with a large supermarket chain. Our model can help retailers automate the process of deciding on the promotions for a large number of items while capturing several key economic factors. We also believe that the models and algorithms developed in this paper are interesting from an optimization theory perspective.

In the paper by Cohen et al. (2017), the authors consider a simpler version of the problem where the retailer optimizes the promotions for a single item. In various settings, promoting a specific item can potentially affect the demand of several other items. For example, consider two similar substitutable items such as competing brands of cereals. When one item is on promotion, it can often reduce the sales of the other item, given that some customers may switch from one brand to another. Studying the brand-switching effect induced by retail promotions received great attention in the marketing community (see Van Heerde et al. (2003) and the references therein). When deciding on promotions, retailers need to account for the interplay between the different items. The main goal of this paper is to extend the work by Cohen et al. (2017) for a setting with multiple items, where the category manager needs to simultaneously decide on the price promotions of all the items in the category. The solution approach developed for a single item does not yield good results when applied to a setting with multiple items because it does not properly capture the relationships among the different items. As we will see, extending the solution approach from a single item to a setting with multiple items is not straightforward and requires different machinery. Interestingly, the method developed in this paper can be applied to more general nonlinear binary integer programming (IP) problems beyond the context of retail promotions.

In this paper, we focus on fast-moving consumer goods (FMCGs). Examples of FMCGs include dry goods (e.g., coffee, tea, sugar, and beans), toiletries and cleaning products, and packed beverages. FMCGs typically have a long shelf life and are sold quickly at a relatively low cost. Several sources provide evidence that a large portion of FMCG sales are made on promotions. For example, the group Information Resources, Inc. (IRI) found that in the United Kingdom, 54.6% of the products (by volume) were sold on promotion.[1] It is thus not surprising that FMGC manufacturers invest large budgets for promotions; according to Nielsen (2015), the annual promotion budget for FMGC manufacturers is around $1 trillion.

Planning retail promotions is a challenging process for at least four reasons. First, as discussed earlier, cross-item effects on demand may be significant and need to be taken into account. When an item is on promotion, it can also affect the demand of several other items (complements or substitutes). To our knowledge, previous work did not explicitly capture such effects. Second, it is important to carefully capture the business rules that restrict the different promotions. These rules are either set by the supermarket or negotiated with suppliers. For instance, the total number of promotions may be within a certain limit, and there are restrictions on promoting competing items at the same time (for more details, see Section 2.2). The third reason is related to the postpromotion dip effect on demand. Specifically, customers often strategically purchase large amounts of promoted products toward future consumption. This is especially true for nonperishable items. This stockpiling behavior decreases the demand for future time periods and hence makes the problem more challenging because it couples the different time periods. Finally, the large scale of the promotion optimization problem contributes to its difficulty. Indeed, it is common for supermarkets to hold thousands of stockkeeping units (SKUs), leading to a sizable number of decision variables.

In the retail sector, promotions can substantially affect sales and incentivize customers. In the back-to-school context, the International Council of Shopping Centers ran a survey in 2017 that found that promotions influence close to 90% of shoppers.[2] As discussed, our main goal is to develop a model and tool that can guide retailers when setting price promotions. We intend to propose a data-driven model that can leverage available data and automate the promotion planning process. In this paper, we tackle this problem by first formulating the promotion optimization problem for multiple items (which we call the *Multi-POP*). We then propose an efficient approximation solution approach and examine its performance—both analytically and computationally.

We study a class of demand functions that incorporates the postpromotion dip effect and cross-item effects. Our approximate solution approach will provide promotion prices along with analytical performance guarantees. Our Multi-POP formulation is a nonlinear integer program, and hence an approach based on directly optimizing the objective is not computationally tractable. We first convey that the method developed by Cohen et al. (2017) for a single item does not perform well in a setting with multiple items. To address this issue, we propose a more general class of IP approximation methods. We then show that for demand models with additive cross-item effects (i.e., promoting item j has an additive effect on the demand of item i), the problem can be efficiently solved by considering only unilateral and bilateral contributions, that is, accounting only for the effect of one or two simultaneous promotions (more details are presented in Section 3). We show that realistic-size instances can be solved efficiently: most cases are solved through a single linear program (LP), whereas some more involved cases are solved via an iterative rounding scheme based on solving several LPs. One strength of our method is the fact that it can be applied to a general demand function. We also establish a parametric bound on the performance guarantee relative to the optimal objective. Finally, we use our model and approximation solution to draw managerial insights that can help retailers improve their promotion decisions. Indeed, one of the goals of this research is to develop data-driven optimization models that can guide the promotion planning process for supermarket retailers.

1.1. Contributions

We next summarize our main contributions.

- We propose an IP approximation that accurately captures cross-item effects on demand. We introduce a nonlinear IP formulation for the Multi-POP that captures cross-item effects and several business rules. Our demand models include economic factors such as postpromotion dip and cross-item effects. Because the problem is not computationally tractable, we propose a class of IP approximations—referred to as $App(\kappa)$—based on approximating the objective by unilateral and higher-order contributions. We then focus on *App*(2), which approximates the objective by only accounting for unilateral (i.e., single promotions) and bilateral (i.e., pairs of promotions) contributions. We show that for demand models with additive cross-item effects, the *App*(2) approximation is exact. Consequently, one needs to account only for unilateral and bilateral contributions without computing an exponential number of coefficients.

- We show that the approximation solution can be computed efficiently. We observe that the constraint matrix of the approximated problem is not totally unimodular. Nevertheless, we show that for substitutable items and additive cross-item effects, the unconstrained problem admits an integral LP relaxation under two prices. Under these assumptions, we can thus obtain an optimal solution efficiently by solving a linear program. Ultimately, it allows us to solve large, realistic instances in short timeframes. We also show computationally that even more complicated instances can be solved in acceptable timeframes.
- We develop a performance bound for multiplicative demand functions with additive cross-item effects. We derive a parametric bound on the quality of the IP approximation relative to the optimal Multi-POP solution. We convey that for realistic retail instances, our bound yields a good performance guarantee. We also show that the approximation leads to the optimal solution for additively separable demand functions (in both past and cross-prices).
- We use our model to draw insights on promotion planning. We study the interplay between the post-promotion dip and cross-item effects. Understanding such insights can be useful for category managers who need to schedule the promotions of thousands of items. For example, we convey that when the degree of substitution increases, it becomes optimal to reduce the number of promotions. We also show that our model captures the loss-leader effect (i.e., the strategy of pricing a product below cost to stimulate the demand of other products), which is often observed in retail. Finally, we test the practicality of our model on realistic instances.

1.2. Literature Review

Our work is related to three streams of literature: nonlinear optimization, dynamic pricing, and promotions in the field of marketing.

1.2.1. Nonlinear Optimization. The promotion optimization problem is written as a nonlinear mixed-integer program (NMIP). Under most realistic demand models, the resulting objective function is nonconcave, and thus the problem is hard to solve. Hemmecke et al. (2010) present several structural assumptions that allow us to solve NMIPs in polynomial time. In many practical cases, these assumptions are not satisfied, so one needs to solve the NMIP by using techniques such as branch and bound and extended cutting-plane methods (Grossmann 2002). One of our approximation methods is based on solving an unconstrained binary quadratic program (UBQP). Rhys (1970) and Balinski (1970) have shown that the UBQP can be solved in polynomial time under certain conditions, which are similar to the conditions in our setting. There are a large number of follow-up studies on this topic. For example, in the context of project scheduling, Möhring et al. (2003) consider a similar formulation as ours and extend the reduction of a special case of ordinary precedence constraints and unit processing times to the minimum-cut problem.

In this paper, we exploit the discreteness of the Multi-POP to approximate the objective function. More precisely, we propose a general class of IP approximations. We show that approximating the objective by unilateral and bilateral deviations (i.e., contributions of one and two promotions) yields a good performance. This approximation is connected to the well-studied topic of quadratic programming (Frank and Wolfe 1956, Nocedal and Wright 2006). To our knowledge, the method presented in this paper differs from previous approaches in that it exploits the specific structure of the promotion optimization problem to derive structural insights.

1.2.2. Dynamic Pricing. The field of dynamic pricing was and still is extensively studied. For comprehensive reviews, see Talluri and Van Ryzin (2006) and Özer and Phillips (2012). The work by Cohen et al. (2017) studies the promotion optimization problem for a setting with a single item. The authors propose an efficient algorithm based on discretely linearizing the objective. They then show that their approximation yields a near-optimal solution in most practical instances, runs in milliseconds, and can easily be implemented by retailers. As we discuss later in this paper, the same methodology cannot be extended to the setting with multiple items without sacrificing performance. Instead, we introduce a more general method that explicitly accounts for the pairwise interactions of promoting two items simultaneously. Cohen and Perakis (2020) present an overview on promotion optimization in retail by discussing the models and insights developed in Cohen et al. (2017) and in the present paper (this book chapter is meant to be an overview and does not contain original research; it does not include the specifics of the algorithms, the formal analytical results and their proofs, and the computational experiments). Dynamic pricing is studied by a multitude of researchers in different contexts (Ahn et al. 2007, Levin et al. 2010, Su 2010, Cohen et al. 2018). Ahn et al. (2007) consider a setting where a fraction of the customers will strategically wait and purchase the product if the price falls below their valuation. Su (2010) investigates a model with several customer types (composed of shopping and holding costs and rates of consumption). Finally, Levin et al. (2010) consider a monopolist offering a perishable item to strategic customers. Note that most previous work considers stylized modeling assumptions and is often restricted to a single-item setting. In this paper, however, we consider a practical problem where the demand models are directly estimated from data, and the optimization formulation includes the promotion decisions of several interconnected items.

1.2.3. Promotions in Marketing. Retail promotions are at the core of the marketing discipline. For more details, see the book by Blattberg and Neslin (1990) and the references therein. In this stream of literature, researchers typically estimate dynamic demand models to study causal effects and sharpen our current understanding on the impact of promotions (Foekens et al. 1998, Cooper et al. 1999). Foekens et al. (1998) estimate parametric models calibrated with scanner data to infer the dynamic impact of promotions. As discussed earlier, for several categories of items, retail promotions may have a postpromotion dip effect. Specifically, when the item is promoted, consumers may stockpile by buying additional units toward future consumption. This effect results in a reduction in short-term future sales. A common way to model this effect is by assuming that the demand function depends both on the current price and on the prices from the previous periods (Macé and Neslin 2004, Ailawadi et al. 2007). The demand models used in our paper also consider that the demand depends explicitly on current and past prices as well as on prices of other items. Finally, Baardman et al. (2018) address the problem of optimally scheduling promotional vehicles (e.g., end-cap displays, in-store advertisements) for a retailer.

A number of studies empirically examine retail promotions (Van Heerde et al. 2003, Felgate and Fearne 2015). Most of these papers adopt a descriptive approach. In this paper, however, we are more interested in the prescriptive side of solving the promotion optimization problem to help retailers decide on future promotions.

1.2.4. Structure of the Paper. In Section 2, we describe the model and assumptions, as well as the formulation of the Multi-POP. In Section 3, we introduce and study a class of approximation methods for our problem. In Section 4, we use our model and solution approach to draw practical insights on promotion planning. Section 5 presents computational experiments and conveys the applicability of our solution approach. Finally, we report our conclusions in Section 6. Most proofs of the technical results are relegated to the appendix.

2. Problem Formulation

We consider a supermarket category such as ground coffee, cereals, and soft drinks with N items or SKUs. We seek to maximize the total profit throughout a selling season consisting of T periods (e.g., one quarter of 13 weeks). The price of item i at time t is denoted by p_t^i (the subscript index corresponds to the time, whereas the superscript index refers to the item). The (exogenous) unit cost is denoted by c_t^i.

2.1. Assumptions

We assume that the retailer carries a sufficient level of inventory to meet the demand of each of the N items during the selling season.[3] Although this assumption is not satisfied for all retail products (e.g., fashion items), it is typically satisfied for FMCGs. Unlike fashion items, which are often seasonal, FMCGs are available throughout the year. As mentioned earlier, FMCGs are easy to store and have a long shelf life. Retailers have an extensive experience in inventory and stocking decisions and often use modern forecasting tools to support ordering decisions (Cooper et al. 1999, Van Donselaar et al. 2006). In addition, grocery retailers are aware of the adverse effects of stock-outs (Campo et al. 2000, Corsten and Gruen 2004). We thus assume that FMCG retailers have enough inventory for the selling season. Cohen et al. (2017) examine retail data for FMCGs and confirm that the demand forecast accuracy is high and that a very small number of stock-outs occurred in a two-year period.

We assume that the demand is a time-dependent nonlinear function of the prices. More precisely, the demand of item i at time t is denoted by $d_t^i(\mathbf{p})$, where $\mathbf{p}$ is a vector of current and past prices (see more details in Equation (1)). The objective is to decide which items to promote and when to maximize the total profit over the selling season T.

Each item $i = 1, \ldots, N$ can take several prices: the regular price denoted q^{i0} and $K^i = |Q_i| - 1$ promotion prices denoted q^{ik}. The total number of price points for item i is called the *size of the price ladder* and is denoted $|Q_i|$. For example, if we have $K^i = 2$ promotion prices, then $q^{i2} < q^{i1} < q^{i0}$ and $|Q_i| = 3$.

Without loss of generality, we assume that the regular price $q^{i0} = q^0$ is the same across all items and all time periods (this assumption can be relaxed at the expense of a more cumbersome notation). Note that a typical real-world instance can be composed of 30–150 items, 13 periods, and several price points, yielding a large number of decision variables. We define the binary decision variable γ_t^{ik} to be equal to one if the price of item i at time t is set to q^{ik} and zero otherwise.

In several product categories, cross-item effects on demand can be significant. Specifically, a promotion for a particular item affects its own sales but also the sales of other items in the category. One can distinguish between two different types of cross-item effects: *substitutability* and *complementarity*. These two types of cross-item effects are well known in economics and operations management (Pindyck and Rubinfeld 2005). The standard example of substitutable items is competing brands such as Coke and Pepsi. In this case, it is clear that promoting a Coke product potentially increases Coke's sales but may also decrease Pepsi's sales. This follows from the fact that some customers may be indifferent between both items and are likely to switch from one brand to the other when one is on promotion. Mathematically, if items i and $j \neq i$ are substitutes, then $\partial d_t^i/\partial p_t^j \geq 0$ and $\partial d_t^j/\partial p_t^i \geq 0$. Two items i and j are complements if the consumption of i induces customers to purchase j, and vice versa, for example, shampoo and conditioner. In this case, it is clear that when the shampoo is on promotion, its own sales increase, but the sales of the conditioner can also increase because people typically purchase both items together. Mathematically, if items i and $j \neq i$ are complements, then $\partial d_t^i/\partial p_t^j \leq 0$ and $\partial d_t^j/\partial p_t^i \leq 0$. In what follows, we formulate the promotion optimization problem for multiple items while explicitly incorporating cross-item effects on demand. Before doing so, we impose the following assumption on the demand function.

Assumption 1. *The demand function depends explicitly on self past and current prices and on cross current prices.*

Assumption 1 implies that the demand does not explicitly depend on cross past prices. In other words, the demand of item i does not depend on the past prices of the other items in the category. This assumption is supported by the fact that most consumers may be loyal to a particular brand. As a result, they can stockpile large quantities of a specific item while it is on promotion. They are also aware of all the prices of the other items at time t, so they can potentially decide to switch and purchase a different item. However, consumers usually do not remember the past prices of other items in the category. We tested the validity of this assumption using data from a large retailer and observed that in the vast majority of product categories, the parameters related to the cross past prices were not statistically significant (the details of these tests are beyond the scope of this paper and hence omitted). The demand of item i at time t can be any nonlinear time-dependent function of the form

$$d_t^i(p_t^i, p_{t-1}^i, \ldots, p_{t-M^i}^i, \mathbf{p_t^{-i}}), \tag{1}$$

where M^i represents the memory parameter of item i, that is, the number of past prices that affect current demand, and $\mathbf{p_t^{-i}}$ denotes the vector of prices of all items but i at time t. Note that in practice M^i is estimated from data and can be different for different items. The function $d_t^i(\cdot)$ represents the expected demand. For FMCGs, the demand prediction accuracy is often high, so expected and actual demands are close to each other (in other words, we assume that demand is deterministic and can be accurately estimated from data).[4]

2.2. Business Rules

We identify two types of business rules: (1) *self-business rules*, that is, price constraints for each item separately, and (2) *cross-item business rules*, that is, rules that impose price constraints across several items. The self-business rules are the same as in Cohen et al. (2017), whereas the cross-item business rules are original from this paper.

2.2.1. Self-Business Rules.

1. *Discrete price ladder.* The price of each item is often restricted to lie in a finite set of admissible prices (e.g., prices must end with 9 cents). For simplicity, we assume that the elements of the price ladder are time independent, but our results still hold when this assumption is relaxed. Thus, the price of item i at time t can be written as $p_t^i = \sum_{k=0}^{K^i} q^{ik}\gamma_t^{ik}$, where the binary variable γ_t^{ik} is equal to one if the price of item i at time t is q^{ik} and zero otherwise.

2. *Maximal number of promotions per item.* The retailer may want to limit the promotion frequency of a specific item to preserve the store image and not train customers to be deal seekers. For example, it may be required to promote item i at most $L^i = 3$ times during the quarter. This requirement is captured by $\sum_{t=1}^{T} \sum_{k=1}^{K^i} \gamma_t^{ik} \leq L^i$.

3. *No-touch constraint.* A common requirement is to space out two successive promotions by a minimal number of separating periods, denoted by S^i. As before, this helps retailers preserve the store image and discourage customers from being deal seekers. In addition, this type of requirement may be directly dictated by the manufacturer, who sometimes

restricts the frequency of promotions. Such a requirement translates to $\sum_{\tau=t}^{t+S^i}\sum_{k=1}^{K^i}\gamma_\tau^{ik} \le 1\ \forall t$.

2.2.2. Cross-Item Business Rules.

1. *Total maximal number of promotions for the category.* The supermarket may want to restrict the number of promotions for the category during the next T periods (e.g., at most $L_T = 20$ promotions are allowed). This requirement translates to

$$\sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K^i}\gamma_t^{ki} \le L_T. \tag{2}$$

We notice that the constraint in (2) is relevant only if $L_T < \sum_{i=1}^N L^i$.

2. *Interitem constraints.* Business rules often impose price constraints across different items. For instance, for a given item, the smaller format should be priced at a lower price point than the larger format. Similarly, private labels should be cheaper than national brands. This type of interitem business rule can be written as linear price constraints: for example, $p_t^i \le p_t^j\ \forall t$ captures the fact that item i should be priced no higher than item j.

3. *Simultaneous promotions.* In some cases, category managers need to promote specific items at the same time. This requirement may come from a special promotional campaign or from a manufacturer trade fund. Specifically, the constraint $\gamma_t^{0i} = \gamma_t^{0j}\ \forall t$, captures the restriction that items i and j must be promoted at the same time (the binary variable γ_t^{0i} is one if there is no promotion for item i at time t).

4. *Maximal number of promotions per period.* Category managers often need to restrict the number of promotions in the category for each time period (e.g., a week). For example, at most, 10% of the items in the category may be promoted (i.e., $C^t = N/10$):

$$\sum_{i=1}^{N}\sum_{k=1}^{K^i}\gamma_t^{ki} \le C^t\ \forall t. \tag{3}$$

5. *Inter-no-touch constraints.* Spacing out promotions can also be required for a set of items. Once again, such constraints help to discourage customers from being deal seekers and preserve the store image. In this case, we need to separate the successive promotions for a given set of items by at least S^c separating periods. This requirement translates to $\sum_i\sum_{\tau=t}^{t+S^c}\sum_{k=1}^{K^i}\gamma_\tau^{ki} \le 1\ \forall t$, where the summation on i is taken over a specific set of items in the category. The special case with $S^c = 0$ corresponds to never holding simultaneously promotions to impose an exclusive deal—a common practice in retail.

2.3. Problem Formulation

We can now formulate the promotion optimization problem for multiple items (Multi-POP):

$$\begin{aligned}
&\max_{\gamma_t^{ik}} && \sum_{i=1}^{N}\sum_{t=1}^{T}(p_t^i - c_t^i)d_t^i(p_t^i, p_{t-1}^i, \dots, p_{t-M^i}^i, \mathbf{p_t^{-i}}), \\
&\text{subject to (s.t.)} && p_t^i = \sum_{k=0}^{K^i} q^{ik}\gamma_t^{ik}, && \forall i \\
& && \sum_{t=1}^{T}\sum_{k=1}^{K^i}\gamma_t^{ik} \le L^i, && \forall i \\
& && \sum_{\tau=t}^{t+S^i}\sum_{k=1}^{K^i}\gamma_\tau^{ik} \le 1, && \forall i,t \\
& && \sum_{k=0}^{K^i}\gamma_t^{ik} = 1, && \forall i,t \\
& && \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K^i}\gamma_t^{ki} \le L_T, \\
& && \sum_{i=1}^{N}\sum_{k=1}^{K^i}\gamma_t^{ki} \le C^t, && \forall t \\
& && \gamma_t^{ik} \in \{0,1\} && \forall i,t,k.
\end{aligned} \tag{Multi-POP}$$

As discussed, L^i, S^i, and K^i denote the limitation of promotions, no-touch period, and number of promotion prices in the price ladder for item i, respectively. Note that the objective is to maximize the total profit generated by the N items in the category over the selling season. In the formulation (Multi-POP), we have included the various self-business rules and two cross-item business rules (total maximal number of promotions and maximal number of promotions per period). Depending on the setting, one can also include alternative cross-item business rules as additional constraints. We highlight that even in the absence of cross-item business rules, the N items are linked via the cross-item effects on demand.

3. Solution Approach

In the Multi-POP formulation, one can observe two types of effects: (1) cross-time effects, which are driven by the effect of past prices on current demand for each item, and (2) cross-item effects, which arise from substitution and complementarity of the different items in the category. Because one needs to decide on promotions for all N items at all times, the problem is a large-scale nonlinear integer program. Our first attempt was to apply the linear IP approximation based on unilateral contributions of having a single promotion at a time, developed by Cohen et al. (2017). For the case of multiple items, it approximates the objective by the sum of unilateral promotions of each item separately. This approach fails to capture the cross-item effects and will lead to poor performance when the cross-item effects are significant. For instance, by ignoring substitution effects, such an approximation can lead to promoting all the items, whereas the optimal policy would be to not promote at all. To develop a better solution approach for Multi-POP, we need to find a way to incorporate the cross-item effects. To this end, we introduce the class of methods $App(\kappa)$ for any given $\kappa = 1, 2, \ldots, N$, where κ represents the degree of the approximation. More precisely,

- $App(1)$ approximates the Multi-POP objective function by the sum of the marginal contributions of a single promotion for each item and time period. In other words, $App(1)$ is based on the IP approximation method developed by Cohen et al. (2017) for the setting with a single item. As discussed earlier, however, this method typically yields a poor performance for a setting with multiple items, including in our various computational experiments.
- $App(2)$ approximates the Multi-POP objective function by the sum of the marginal contributions, as in $App(1)$, as well as the pairwise contributions, that is, having two items promoted simultaneously. We describe the details of $App(2)$ in the following paragraphs.
- $App(N)$ is an alternative IP approximation that includes the marginal contributions, the pairwise contributions, and so on, up to all possible combinations of having all N items promoted simultaneously.

One can also naturally consider any intermediate method for $2 < \kappa < N$. Note that there is a trade-off between simplicity and performance (in terms of the approximation method's accuracy). On the one hand, we have $App(1)$. This method only requires computation of the marginal contributions of having a single promotion at a time but can yield a poor performance because it does not capture the cross-item effects on demand. On the other hand, we have $App(N)$. This method is clearly more accurate than $App(1)$ because it captures all the cross-item effects on demand. Needless to say, this benefit comes at a cost. Indeed, $App(N)$ relies on computing the marginal contribution of promoting every combination of items at the same time. It thus requires computation of an exponential number of coefficients and solving an integer program that grows exponentially with the number of items. Note that, in general, $App(N)$ is not guaranteed to yield an optimal solution. For the special cases where either $T = 1$ or $M^i = 0\ \forall i$, $App(N)$ does yield an optimal solution because no time effects are induced from past promotions. In a general instance, however, $App(N)$ is not necessarily an exact method. We next present the $App(2)$ method in greater detail.

As mentioned earlier, $App(2)$ approximates the Multi-POP objective function by the sum of the marginal contributions (i.e., single promotion for each item and time period) and the pairwise contributions (i.e., two items promoted simultaneously). Accordingly, the approximated objective can be written as

$$MPOP(\mathbf{p^0}) + \max_{\gamma}\left\{\sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K^i} b_t^{ki}\gamma_t^{ki} + \sum_{i,j:i>j}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K^i}\sum_{\ell=1}^{K^j} b_t^{k\ell ij}\gamma_t^{k\ell ij}\right\}, \tag{4}$$

where b_t^{ki} and $b_t^{k\ell ij}$ are defined in (5) and (6), respectively. The binary decision variable $\gamma_t^{k\ell ij}$ is equal to one if the prices of items i and j at time t are set to q^{ik} and $q^{j\ell}$, respectively, and zero otherwise. Here $\mathbf{p^0} = (q^0, \ldots, q^0)$ denotes the NT-dimensional regular price vector and corresponds to the situation where the regular price is set for all items at all times. The term $MPOP(\mathbf{p^0})$ accounts for the total profit when no promotion is made. The middle term in (4) captures the marginal contributions of having a single promotion at a time. The marginal contribution of promoting item j at time t (using price q^{kj}) is captured by the NT-dimensional price vector $\mathbf{p_t^{kj}}$, which is given by

$$\left(\mathbf{p_t^{kj}}\right)_\tau = \begin{cases} q^{kj}, & \text{if } \tau = t \text{ and } i = j, \\ q^0, & \text{otherwise.} \end{cases}$$

The vector $\mathbf{p_t^{kj}}$ is such that the regular price q^0 is used for item j at all times but t, as well as for all items different from j at all times. Then the coefficient b_t^{kj} captures the marginal profit contribution of having a single promotion for item j at time t (using price q^{kj}) and is given by

$$b_t^{kj} = MPOP\left(\mathbf{p_t^{kj}}\right) - MPOP(\mathbf{p^0}). \tag{5}$$

The last term in (4) captures the pairwise contributions of having two items promoted at the same time. The marginal contribution of promoting both item j (using price q^{kj}) and item $u < j$ (using price $q^{\ell u}$) at time t is captured by the NT-dimensional price vector $\mathbf{p_t^{k\ell ju}}$, which is given by

$$\left(\mathbf{p_t^{k\ell ju}}\right)_\tau = \begin{cases} q^{kj}, & \text{if } \tau = t \text{ and } i = j, \\ q^{\ell u}, & \text{if } \tau = t \text{ and } i = u, \\ q^0, & \text{otherwise.} \end{cases}$$

The vector $\mathbf{p_t^{k\ell ju}}$ is such that the regular price q^0 is used for items j and u at all times but t, as well as for all items different from j and u at all times. Then the coefficient $b_t^{k\ell ju}$ captures the pairwise contribution on the total profit of having two simultaneous promotions for items j and u at time t and is given by

$$b_t^{k\ell ju} = MPOP\left(\mathbf{p_t^{k\ell ju}}\right) - MPOP\left(\mathbf{p_t^{kj}}\right) - MPOP(\mathbf{p_t^{\ell u}}) + MPOP(\mathbf{p^0}), \tag{6}$$

but note that we can also write $b_t^{k\ell ju} = MPOP(\mathbf{p_t^{k\ell ju}}) - MPOP(\mathbf{p^0}) - b_t^{kj} - b_t^{\ell u}$. Using this representation, $b_t^{k\ell ju}$ corresponds to the marginal contribution of having two simultaneous promotions for items j and u at time t relative to the case where the promotions are made separately.

To ensure that the $App(2)$ formulation is consistent, we need to add some additional constraints. Specifically, we want to ensure that when both items i and j are promoted, we include the pairwise contribution of these two items and both unilateral contributions. In other words, for each pair of items i and $j < i$, $\gamma_t^{ki} = \gamma_t^{\ell j} = 1$ if and only if $\gamma_t^{k\ell ij} = 1$ for each t and k, ℓ. To incorporate this type of condition, we add the four following constraints for each pair of items i, $j < i$, each t, and each promotion price q^{ki} and $q^{\ell j}$ to the $App(2)$ formulation:

$$\begin{aligned} &\gamma_t^{k\ell ij} \le \gamma_t^{ki}, \quad \gamma_t^{k\ell ij} \le \gamma_t^{\ell j}, \quad \gamma_t^{k\ell ij} \ge 0, \\ &\gamma_t^{k\ell ij} \ge \gamma_t^{ki} + \gamma_t^{\ell j} - 1. \end{aligned} \tag{7}$$

Note that by construction, $App(2)$ is exact for $N = 2$ in the sense that it accurately captures all the cross-item effects. For $N > 2$, $App(2)$ approximates the Multi-POP objective by the sum of unilateral and pairwise contributions and is not exact (unless the cross-item effects are present only for pairs of items). However, it performs significantly better than $App(1)$ because the latter does not account for cross-item effects at all. Finally, we can similarly formalize the definition of $App(\kappa)$, $\kappa = 3, \ldots, N$.

Ultimately, the $App(2)$ approximation consists of solving the problem in (4) while adding the additional constraints in (7). The decisions variables are the binary variables γ. We have one such variable for each of the N items, the T time periods, and the $K+1$ price points (i.e., $NT(K+1)$, assuming for simplicity that $K^i = K\ \forall i$). We also have one such variable for any pair of items $i > j$ at each time and price (i.e., $(N(N-1)/2)\,TK^2$). As discussed earlier, for the $App(N)$ method, the number of variables increases exponentially in N and K, so it may be impractical to go beyond $App(3)$ or $App(4)$. Nevertheless, as we show in Theorem 1, it is enough to consider $App(2)$ under some assumptions.

Theorem 1. *Assume that the cross-item effects for each item are additively separable; that is,*

$$\begin{aligned} &d_t^i(p_t^i, p_{t-1}^i, \cdots, p_{t-M^i}^i, \mathbf{p_t^{-i}}) \\ &\quad = h_t^i(p_t^i, p_{t-1}^i, \cdots, p_{t-M^i}^i) + \sum_{j \ne i} H_t^{ji}\left(p_t^j\right). \end{aligned} \tag{8}$$

Then we have $App(2) = App(3) = \ldots = App(N)$.

In Equation (8), the function $h_t^i(\cdot)$ represents the part of the demand that depends on self current and past prices, and the function $H_t^{ji}(\cdot)$ corresponds to the contribution of price p_t^j to the demand of item i at time t. Note that the result of Theorem 1 applies beyond the context of optimizing retail promotions. Indeed, the same result holds for a general binary nonlinear integer program, where the objective function can be expressed as a sum of functions, and each function depends on at most two decision variables. The assumption that the cross-item effects are additively separable allows us to show that $App(2)$ is equivalent to $App(N)$. As a result, it is sufficient to consider only unilateral and bilateral (pairwise) contributions to capture all the cross-item effects. This assumption is satisfied by several demand functions such as the log-linear model with linear cross-terms (see (12)) and the linear cross-elasticities model (see (13)). These models are often used in practice because they can be easily estimated from data. That said, one could consider alternative demand models that do not satisfy this assumption. In this case, $App(2)$ is not necessarily equal to $App(N)$. Nevertheless, we have observed computationally that the performance of $App(2)$ is often satisfactory because it provides a good approximation of the cross-item effects. As expected, the $App(2)$ solution coincides with the Multi-POP optimal solution under one of the following four conditions: (1) $T = 1$, (2) $M^i = 0,\ \forall i$, (3) $L^i = 1,\ \forall i$, or (4) $S^i \ge M^i,\ \forall i$. Indeed, under one of these conditions, the time effects (i.e., the impact of past prices on current demand) are not present, so we remain only with cross-item effects and thus obtain an optimal solution.

More generally, one can refine the $App(2)$ approximation to be optimal under the following condition.

Corollary 1. *If the function $h_t^i(p_t^i, p_{t-1}^i, \ldots, p_{t-M^i}^i)$ is additively separable for each item $i = 1, \ldots, N$, that is,*

$$\begin{aligned} &h_t^i(p_t^i, p_{t-1}^i, \ldots, p_{t-M^i}^i) \\ &\quad = f_t^i(p_t^i) + g_1^i(p_{t-1}^i) + \ldots + g_{M_i}^i\left(p_{t-M_i}^i\right), \end{aligned} \tag{9}$$

then the $App(2)$ solution is optimal.

In Equation (9), the function $f_t^i(\cdot)$ represents the part of the demand that depends on the current self-price p_t^i, whereas the function $g_u^i(\cdot)$ for $u = 1, \ldots, M_i$ captures the part of the demand that depends on p_{t-u}^i. Corollary 1 assumes that past and current self-prices are also additively separable. For example, this assumption is satisfied for a linear demand function. In this case, one can extend the definition of the $App(2)$ approximation to include all pairwise contributions of two simultaneous promotions: both for different items at the same period and for the same item at different periods (within M^i consecutive periods). In this case, all the pairwise interactions are accounted for, so the $App(2)$ method will lead to an optimal solution. We note that, ultimately, the key for $App(2)$ to be optimal is the additively separable property of the demand function.

We return to the more general demand function in (8), for which we can solve the Multi-POP by applying *App*(2) and accurately capture the cross-item effects. We are now interested in the following two questions:

1. As discussed earlier, to solve *App*(2), one needs to solve an integer program with a quadratic number of variables in N and K. Can we show that the LP relaxation of *App*(2) is integral so that we can solve *App*(2) efficiently as a linear program? Is the feasible region totally unimodular, as it is for *App*(1)?

2. Recall that for additively separable cross-item effects, we have $App(2) = App(N)$. However, *App*(2) does not always yield an optimal solution because an approximation error is induced by the effect of past prices for demand functions that are not additively separable as in (9). Can we derive a bound on the performance of the solution obtained from *App*(2) relative to the optimal solution?

The rest of this section is devoted to answering these two questions.

3.1. Integrality of *App*(2)

As discussed earlier, the *App*(2) formulation requires the addition of four consistency constraints for each time, pair of items, and pair of promotion prices. As expected, adding this set of constraints modifies the feasible region of the Multi-POP. As shown by Cohen et al. (2017), the single-item problem—which uses *App*(1)—has a tight LP relaxation because the constraint matrix is totally unimodular. We can show that in the absence of cross-item business rules, this property is preserved for the Multi-POP because the feasible region for each item is totally unimodular. As a result, *App*(1) leads to an integral formulation. Although this is not the case for *App*(2), we can show the following result.

Theorem 2. *Consider an additively separable demand as in* (8) *and* $K^i = 1$ *(i.e., the price ladder consists of the regular price and one promotion price). For substitutable items, the App*(2) *formulation is always integral in the absence of business rules.*

Proof. We prove the result of Theorem 2 by using the two following lemmas.

Lemma 1. *Consider an additively separable demand as in* (8) *with substitutable items. Then the cross-coefficients for App*(2), b_t^{ij}, $\forall i, j > i$, *and* $\forall t$ *are nonnegative.*

The proof of Lemma 1 can be found in the appendix. As we show in the proof of Lemma 1, under additively separable demand, the coefficient b_t^{ij} reduces to $b_t^{ij} = (q^0 - p_t^i)[H_t^{ji}(q^0) - H_t^{ji}(p_t^j)]$ and hence is easy to analyze and interpret.

We next consider the following optimization problem, called the *unconstrained binary quadratic program* (UBQP):

$$(UBQP) \max_x \sum_{i=1}^{N} b^i x_i + \sum_i \sum_{j>i} b^{ij} x_i x_j$$
$$\text{s.t.} \quad x_i \in \{0,1\}, \quad \forall i.$$

Lemma 2. *If all the cross-coefficients* b^{ij} *are nonnegative, one can solve the UBQP by a linear program.*

Lemma 2 follows from Rhys (1970), who has shown that the selection problem of shared fixed costs is integral because it can be viewed as a network flow formulation. More precisely, we can reformulate the UBQP as an integer program by defining a new variable for each pair of items $i, j > i$, denoted by x_{ij}, and adding the following consistency constraints:

$$x_{ij} \le x_i, \quad x_{ij} \le x_j, \quad x_{ij} \ge 0, \quad x_{ij} \ge x_i + x_j - 1.$$

Note that when $b^{ij} \ge 0$, the last constraint is redundant and can be omitted. In this case, the variable x_{ij} is equal to the product $x_i x_j$. Note also that the x_{ij} variables are continuous, whereas the x_i variables are binary. In addition, LP relaxation is not generally tight because the constraint matrix, which is composed of the consistency constraints, is not totally unimodular. Indeed, as shown by Padberg (1989), every vertex in the linear approximation to UBQP is $\{0, \frac{1}{2}, 1\}$-valued, so the constraint matrix is not totally unimodular. When all the cross-coefficients b^{ij} are nonnegative, however, the result of Lemma 2 follows directly from Rhys (1970).

We next conclude the proof of Theorem 2. By using the results of Lemmas 1 and 2, we can solve the UBQP as a linear program when the items are substitutable. Recall that the *App*(2) approximation solves the problem with (4) as an objective, which can be equivalently rewritten in the form of the UBQP. □

The result of Theorem 2 shows that there exists an integral solution to the LP relaxation. To arrive at an integral solution, we need to use the iterative rounding procedure given in the proof of Lemma 1. Theorem 2 admits the following geometric interpretation. As discussed earlier, the matrix of the feasible region of *App*(2) is not totally unimodular. Consequently, some of the extreme points can be fractional. By considering an additively separable demand with substitution effects, we ensure that the objective will induce the optimal solution to always lie on integer extreme points. Therefore, we can solve the LP relaxation and obtain an integer solution efficiently.

In Theorem 2, we assume that each item can have two prices: the regular price q^0 and the promotion price $q^{i1} < q^{i0}$. This case is common in practice because the promotion price is often negotiated upfront with the manufacturer via trade funds and other contractual agreements. Subsequently, the retailer needs to decide when to schedule the promotions of the different items. In the more general case where the retailer can choose among more than two promotion prices, the IP formulation is not guaranteed to be integral anymore. In such a case, as we will show in Section 5, one can still solve the integer program in low run times for realistic-size instances.

We highlight that in most supermarket categories, the items are either independent (i.e., no cross-item effects) or substitutable. For categories such as coffee, tea, and chocolate, we could not find any complementarity effects in the data we analyzed. Note also that even if some items are complements, we observed by extensive testing that *App*(2) often yields an optimal integer solution. Specifically, we considered a computational experiment based on 10,000 random instances with $N = 5$ and $K = 2$, in which the cross-price sensitivity values are randomly generated (more details on the instances are presented in Section 5). We then observed the following: (1) the *App*(2) solution was integral 99% of the time (in the absence of cross-item constraints) and (2) 91% of the time in the presence of cross-item constraints. More details on our computational tests are presented in Section 5. In addition, recall that in most applications, we only have substitution effects, so *App*(2) is always integral in the absence of business rules. Finally, we observed by extensive computational testing that even in the presence of business rules, *App*(2) yields an integral solution most of the time.

We next discuss the properties of the polytope from Lemma 2—called the *Boolean quadric polytope*—for the general case when cross-coefficients are not necessarily nonnegative. It is known from Padberg (1989) that every vertex of the LP relaxation of a Boolean quadric polytope is $\{0, \frac{1}{2}, 1\}$ valued. Using section 6 of Padberg (1989), there exist sufficient conditions for LP integrality, even when not all the cross-coefficients are nonnegative. Two such cases are the following: (1) when the set of items is acyclic and (2) when the items are pairwise complementary.

Corollary 2. *Consider an additively separable demand with $K^i = 1$ in the absence of business rules. Then we have*

a. *For a set of items (either substitute or complement) that corresponds to a forest in the graph, the App(2) formulation is always integral. In this case, the graph corresponding to the set of items is acyclic.*

b. *For a set of pairwise complementary items (i.e., $b^{ij} \le 0$) that can be transformed into a bipartite graph, the App(2) formulation is always integral.*

In Corollary 2, we use a graphical way of representing the N items so that items i and j are connected via an edge in the graph if $b^{ij} \neq 0$. In practice, the conditions of Corollary 2 are not easy to motivate in most retail settings, whereas the case where all the cross-coefficients are nonnegative is very common. Indeed, in most retail categories, the different items are either independent (i.e., $b^{ij} = 0$) or substitutable (i.e., $b^{ij} > 0$).

3.2. Rounding Schemes for *App*(2)

Because solving the LP relaxation of *App*(2) does not always lead to an integral solution, we next explore different schemes to round the fractional solution to make it integral. We first discuss a naive rounding scheme. We then introduce three rounding schemes based on iteratively resolving the LP relaxation. We next show analytically the convergence and integrality-preservation property of these rounding schemes. Finally, in Section 5.3, we compare computationally the performance of our rounding schemes relative to the *App*(2) solution by directly solving the integer program.

3.2.1. Naive Rounding Scheme. We first consider a naive scheme, obtained by rounding all the fractional variables in the LP solution to zero. Note that this rounding scheme guarantees that the new rounded solution is feasible. Because the solution is always $\{0, \frac{1}{2}, 1\}$ valued, all the fractional variables are necessarily equal to $\frac{1}{2}$. We next separate the analysis into two cases.

Case 1: There exists at least one fractional x_i^*. Because $x_{ij} \le x_i, \forall j$, then all the corresponding x_{ij}^* can only be zero or $\frac{1}{2}$. If $x_{ij}^* = 0$, then we do not need to round x_{ij}^*, and hence the constraint $x_{ij} \ge x_i + x_j - 1$ will still hold (the left side remains the same and the right side decreases). If $x_{ij}^* = \frac{1}{2}$, then it would be rounded down to zero. In this case, the new solution will be feasible if $0 \ge x_j - 1$, and this naturally holds because all x_j are less than or equal to one.

Case 2: There is no fractional x_i^*. In this case, if we have a fractional x_{ij}^*, it would be rounded to zero. Given the constraints $x_{ij} \le x_i$ and $x_{ij} \le x_j$, the corresponding x_i^* and x_j^* must be equal to one. In this case, however, the constraint $x_{ij} \ge x_i + x_j - 1$ cannot be satisfied by the original solution, so this case cannot exist.

Although the feasibility of the rounded solution is guaranteed, we will see from our computational tests that the preceding naive rounding scheme does not yield a good performance. In particular, the gap between the objective value and the optimal mixed-integer programming objective was found to be as large as 40% in our tested instances. This motivates us to propose more sophisticated rounding schemes that can still preserve feasibility while providing a better performance.

3.2.2. Greedy Iterative Rounding Scheme (Rounding Scheme 1). We next consider a rounding scheme that involves reoptimization. In this scheme, we inspect all the parameters b_i that correspond to a fractional x_i^* (if any). We then separate the procedure into three cases:

1. If there is no negative b_i, we increase the fractional x_i^* with the highest b_i to one;
2. If all b_i values are negative, we decrease the fractional x_i^* with the lowest b_i to zero; and
3. Otherwise, we compare the objective value of increasing the fractional x_i^* with the highest b_i to one relative to the objective value of decreasing the fractional x_i^* with the lowest b_i to zero while keeping all other x_i^* values unchanged. We then choose to add the constraint that yields the largest objective value (either $x_i = 1$ or $x_i = 0$).

We next resolve the LP relaxation problem with the additional constraint $x_i = 1$ or $x_i = 0$. We iteratively repeat the preceding procedure until all x_i values are integral. Once all x_i values are integral, we can obtain the values of x_{ij} using $x_{ij} = x_i \times x_j$. We next show that the new solution is feasible and that the preceding reoptimization procedure converges and satisfies the integrality-preservation property.

Proposition 1. *The integral x_i values from the original App(2) solution keep the same value in the new solution obtained from reoptimizing with the constraint $x_k = 1$ or $x_k = 0$, for some $k \neq i$.*

The proof of Proposition 1 can be found in the appendix. Using Proposition 1, the number of fractional x_i variables decreases in each iteration. Because we have a finite number of x_i variables, we conclude that our reoptimization rounding procedure converges in a finite number of iterations.

3.2.3. Exhaustive Greedy Iterative Rounding Scheme (Rounding Scheme 2). We next consider an alternative rounding scheme also based on reoptimization. In the preceding scheme, we only considered the coefficients b_i in a greedy iterative fashion. This procedure ignores the pairwise effects among different items and focuses on local objective improvements. To mitigate this issue, we propose the following rounding scheme. Instead of increasing or decreasing the fractional variable with the highest or lowest b_i, we exhaustively try all fractional variables in each iteration. Assume that we have solved the LP relaxation and obtained $0 < K \leq N$ fractional variables x_i. We then consider increasing each fractional x_i to one if its b_i is nonnegative and decreasing it to zero if its b_i is negative. For each fractional x_i, we resolve the optimization problem with the appropriate additional constraint. At this point, we have solved K optimization problems. We compare the K objective values and choose to add the constraint $x_k = 0$ or $x_k = 1$, which leads to the highest objective. We iterate the preceding procedure until all x_i variables are integral and compute the values of x_{ij} using $x_{ij} = x_i \times x_j$. We can see that this rounding scheme satisfies the convergence and integrality-preservation properties from Proposition 1.

3.2.4. Second-Order Greedy Iterative Rounding Scheme (Rounding Scheme 3). We finally consider one last reoptimization rounding scheme. In the previous rounding schemes, we did not account for *network effects* (i.e., the fact that changing the value of x_i may ultimately affect some connected variables x_j, $j \neq i$). We next propose a refined version of Rounding Scheme 1 that partially accounts for network effects. More precisely, instead of comparing the values of b_i for all fractional x_i^* variables in the original optimal solution, we now compare $b_i + \sum_j b_{ij}$, where j is such that the x_j^* variables are equal to one and connected to x_i^* in the original optimal solution. The rounding scheme proceeds as follows:

1. If there is no i with a negative $b_i + \sum_j b_{ij}$, we increase the fractional x_i^* with the highest $b_i + \sum_j b_{ij}$ to one;
2. If all $b_i + \sum_j b_{ij}$ values are negative, we decrease the fractional x_i^* with the lowest $b_i + \sum_j b_{ij}$ to zero; and
3. Otherwise, we compare the objective value of increasing the fractional x_i^* with the highest $b_i + \sum_j b_{ij}$ to one relative to decreasing the fractional x_i^* with the lowest $b_i + \sum_j b_{ij}$ to zero and add the constraint that leads to the largest objective value (either $x_i = 1$ or $x_i = 0$).

We next resolve the LP relaxation problem with the additional constraint $x_i = 1$ or $x_i = 0$. We iteratively repeat the preceding procedure until all x_i values are integral. Once all x_i values are integral, we can obtain the values of x_{ij} using $x_{ij} = x_i \times x_j$. We can see that this rounding scheme satisfies the convergence and integrality-preservation properties from Proposition 1.

We test computationally all four rounding schemes in Section 5.3.

3.3. Performance Bound

We next present a performance guarantee for the *App*(2) solution relative to the optimal Multi-POP solution. We consider a multiplicative demand with additive cross-item effects such as the log-linear model with additive linear cross-effects (see (12)). More precisely, the demand of item i at time t can be written as

$$d_t^i = f_t^i(p_t^i) \cdot g_1^i(p_{t-1}^i) \cdot g_2^i(p_{t-2}^i) \cdots g_{M^i}^i(p_{t-M^i}^i) + \sum_{j \neq i} H_t^{ji}\left(p_t^j\right), \tag{10}$$

where the first product of $M^i + 1$ functions represents the effect of item i's prices, and the second term corresponds to the additive cross-item effects induced by the other items $j \neq i$. For the demand model in (10), we develop the following parametric bound.

Corollary 3. *Let γ^{MPOP} be an optimal solution of Multi-POP and $\gamma^{App(2)}$ be an optimal solution of the App(2) approximation. Then, when all the items are substitutes, we have*

$$1 \leq \frac{MPOP(\gamma^{MPOP})}{MPOP(\gamma^{App(2)})} \leq \frac{1}{\underline{R}_M}, \tag{11}$$

where $\underline{R}_M = \min_{j=1,2,\ldots,N} \prod_{i=1}^{\tilde{L}^i-1} g^j_{i\cdot(S^i+1)}(q^{jK^j})$, with $\underline{R}_M = 1$ by convention, if $\tilde{L}^i = 1$ for all $i = 1,2,\ldots,N$.

Here $\tilde{L}^i$ denotes the effective maximal number of promotions for item i and is defined as

$$\tilde{L}^i = \min\{L^i, \tilde{N}^i\}, \qquad \text{where } \tilde{N}^i = \left\lfloor \frac{T-1}{S^i+1} \right\rfloor + 1.$$

We assume that $L^i \geq 1$ (the case of $L^i = 0$ is not interesting because no promotions are allowed for item i). Because $\tilde{N}^i \geq 1$, we also have $\tilde{L}^i \geq 1$.

The result presented in Corollary 3 can be shown in the same way as theorem 1 of Cohen et al. (2017), and hence we omit the proof for conciseness. In Cohen et al. (2017), the authors show that in a single-item setting, the *App*(1) approximation yields a provably good performance by deriving a parametric bound in closed form. In Corollary 3, we extend such a guarantee to the setting with multiple substitutable items. We make the following observations. First, in the setting with multiple items, we needs to consider the *App*(2) approximation instead of *App*(1). Second, because *App*(2) = *App*(*N*) under additively separable demand models, the only approximation is in terms of the dependence on past prices. Third, we show that the same bound as in the single-item setting holds, just that now the bound is driven by the item with the lowest-valued $g(\cdot)$ function.

Consequently, the good performance of the bound from the single-item setting is preserved. Indeed, the parametric bound $\underline{R}_M$ is characterized by the worst value of $\underline{R}_i$ for each item $i = 1,2,\ldots,N$. As discussed in Cohen et al. (2017) for the single-item setting, this bound yields a good performance guarantee for a wide range of practical instances. Consequently, it also performs well for the multiple-item setting. By considering several instances of the log-linear model with additive linear cross-effects (see (12)), we observe that the bound was always within 30% of optimal. We can also show the following comparative statics: the value of $1/\underline{R}_M$ improves (i.e., gets closer to one) when (1) L^i decreases, (2) S^i increases, and (3) q^{jK^j} increases.

More important, recall that the bound from (11) is only a theoretical performance guarantee, whereas in practice the ratio $(MPOP(\gamma^{MPOP}))/(MPOP(\gamma^{App(2)}))$ is often much closer to one. We can also show that the bound presented in Corollary 3 is (asymptotically) tight in a similar fashion as in Cohen et al. (2017). Finally, recall that for the additive demand model from (9), the *App*(2) method always leads to an optimal solution.

4. Insights

In this section, we present several managerial insights drawn from our promotion optimization problem. We first show that our model can capture the loss-leader effect, which is often observed in retail environments. We then consider a simple symmetric setting allowing us to investigate the impact of substitution and complementarity on the optimal promotion strategy.

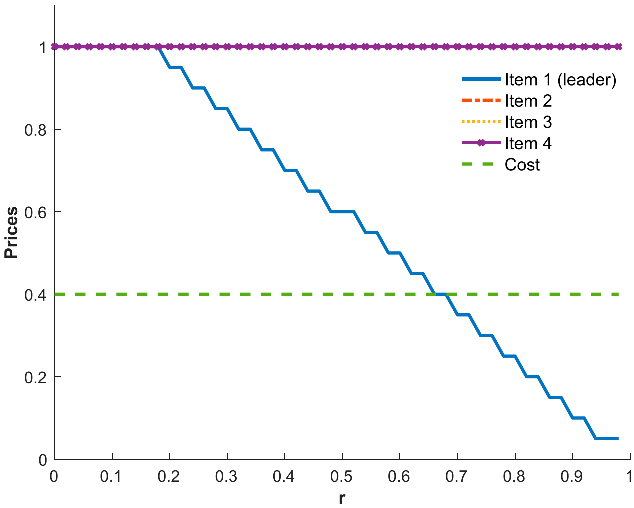
4.1. Loss Leader

The loss leader is a common phenomenon in which one item is priced below cost to extract significant profits from complementary items (Hess and Gerstner 1987). Examples include a printer and cartridges and a video console and games. We next present an illustrative example to convey that our promotion optimization model can capture the loss-leader effect. We consider an example of complementary items and show that the optimal promotion strategy leads to a loss leader. We consider a setting with $N = 4$ items and $T = 1$, where item 1 is the leader (e.g., the printer) and items 2, 3, and 4 are the complements (e.g., cartridges). We then consider a linear demand model $d(\mathbf{p}) = \bar{d} - \beta\tilde{M}\mathbf{p}$ with a price-sensitivity matrix given by

$$\tilde{M} = \begin{bmatrix} 1 & r/2 & r/2 & r/2 \\ r/2 & 1 & 0 & 0 \\ r/2 & 0 & 1 & 0 \\ r/2 & 0 & 0 & 1 \end{bmatrix}.$$

Here r represents the degree of complementarity between the leader and the complements. In addition, we assume that the complements are independent of each other. We assume that the cost of each item is $c = 0.4$, the regular price is $q^0 = 1$, and the promotion prices can be any price between zero and one with 0.02 increments. We then solve the unconstrained Multi-POP and plot the optimal solution in Figure 1. We can see that when $r = 0$ (i.e., independent items), it is optimal to have no promotion. When the degree of complementarity r increases, it becomes optimal to promote item 1 (the leader) but never the complements. When r is large enough (in this example $r > 0.66$), it becomes optimal to set the price of the leader below cost. Consequently, this illustrates that our model can capture the situation where it may be

Figure 1. (Color online) Illustration of the Loss-Leader Effect



optimal to sell an item below cost to extract positive profits from complementary items.

4.2. Cross-Item Effects

We next present several insights related to the impact of cross-item effects on promotion planning. Specifically, we consider solving the Multi-POP and examine the impact of cross-item effects on the optimal solution. For simplicity, we consider a setting with N identical items (i.e., the same demand function, cost, and price ladder) in the absence of business rules. In addition, we assume that the demand function does not depend on past prices (i.e., $M^i = 0\ \forall i$). These simplifying assumptions allow us to isolate and focus on the impact of the cross-item effects. The results are summarized in Table 1.[5]

We can see from Table 1 that for N substitutable items, it is either optimal to promote all the items at all times or to not promote at all. In particular, it depends on the magnitude of the substitution effect. Under strong substitution effects, promoting an item increases its own sales but at the same time decreases the sales of the other items. As a result, we are better off by not promoting at all. When the substitution effects are mild, the benefit of promoting one item overcomes the loss from cannibalization so that it becomes preferable to promote all the items (more details can be found in Section 5.5).

For N complements, the conclusions are reversed. Under strong cross-item effects, promoting one item yields larger sales from that particular item and at the same time increases the sales of other items. As a result, we are better off by promoting all the items at all times. This result is surprising because we would think that when two items are complements, it is enough to promote only one of them. However, recall that we assume a symmetric setting where any pair of items i and $j \neq i$ are complements. Then, promoting item i enhances the sales of both items i and j, and promoting item j has the same effect. In other words, by promoting both items, some additional buyers will be incentivized to purchase the complementary item. Under mild cross-item effects, promoting all the items is not optimal anymore. In this case, the benefit from the complementarity is not larger than the price decrease. It thus becomes optimal to promote only a subset of items (the size of the subset depends on the magnitude of the complementarity effect).

In practice, supermarkets solve the Multi-POP for large-scale instances that involve asymmetries, seasonality, postpromotion dip, substitutes, and complements, as well as several business rules. It is not easy to plan promotions while accounting for all these conflicting trade-offs. This suggests the need for an optimization tool, such as the one developed in this paper. Our model and approach can account for the different trade-offs and compute a near-optimal solution for the promotion planning problem. In addition, because our methods are solved in low run times, we can perform a sensitivity analysis that can test how the promotion planning is affected by changes in demand parameters or in business rules. This allows category managers to reach a better understanding on how several behavioral effects—such as the postpromotion dip effect and cross-item effects—impact promotion decisions.

5. Computational Experiments

In this section, we conduct computational tests to evaluate the performance of the $App(2)$ approximation method in terms of tightness and run time. We also examine the interplay between the postpromotion dip effect and cross-item effects.

5.1. Computational Setting

We consider a realistic demand model inspired by our collaboration with supermarket retailers. Specifically, we consider two types of demand models[6]:

Table 1. Summary of Multi-POP Insights on Cross-Item Effects

Scenario	Mild cross-item effects	Strong cross-item effects
N substitutes	Promote all items at all times	No promotions
N complements	Promote only a subset of items	Promote all items at all times

Table 2. Summary of the Range of Parameters Used in Section 5

Parameter	Interpretation	Log-log model in (12)	Linear model in (13)
a_t^i	Seasonality	$[500, 1{,}000]$	$[0, 60]$
b_0^i	Price sensitivity	$[2, 7]$	$[25, 75]$
b_k^i or α	Past prices effects	$[0, b_0^i]$	$[0.1, 1.5]$
M^i	Number of past prices	$\{0, 1, 2, 3\}$	$\{0, 1, 2, 3\}$
δ^{ji}	Cross-item effects	$[-a_i^t, a_i^t]$	$[-a_i^t, a_i^t]$
c_t^i	Cost of item i	$[0.65, 1]$	$[0.65, 1]$
Q	Price ladder	$[\max c_t^i, 1]$	$[\max c_t^i, 1]$
N	Number of items	$\{5, 10, \ldots, 150\}$	$\{5, 10, \ldots, 150\}$
T	Number of time periods	$\{1, 5, 10, \ldots, 35\}$	$\{1, 5, 10, \ldots, 35\}$

- *A log-log model with additive linear cross-item effects:*

$$d_t^i = a_t^i \exp\{-b_0^i \log(p_t^i)\} \prod_{k=1}^{M^i} (p_{t-k}^i)^{b_k^i} + \sum_{j=1}^{N} \delta^{ji} p_t^j. \qquad (12)$$

- *A linear model with additive linear cross-item effects:*

$$d_t^i = a_t^i - b_0^i p_t^i + \sum_{k=1}^{M^i} \alpha \left(\frac{1}{2}\right)^{k-1} p_{t-k}^i + \sum_{j=1}^{N} \delta^{ji} p_t^j. \qquad (13)$$

Note that both models have additively separable cross-item effects and are special cases of (8). As a result, the *App*(2) approximation is equivalent to *App*(N). For both models (12) and (13), we calibrated the different parameters using actual data from the coffee category of a large supermarket retailer. More precisely, we estimated the following parameters: a_t^i (corresponding to seasonality effects), b_0^i (price-sensitivity factor), b_k^i or α (which capture the effect of past prices on current demand), and δ^{ji} (cross-item effects). In addition, we estimated the value of M^i by removing the past prices that were not statistically significant in our estimation. To thoroughly test the robustness of our results, we further perturb the estimated demand models by randomly varying the estimated parameters. Ultimately, we consider a wide range of realistic parameter values, summarized in Table 2 (each parameter is randomly drawn from the range of values shown in the table). To avoid negative demand values, we focus on combinations of parameters that yield nonnegative demand values. Specifically, we randomly generate demand instances using the range of parameters in Table 2, and if the nonnegativity property is not satisfied, we drop the instance and generate a new one.

Note that the cross-item coefficients δ^{ji} are randomly drawn from the range $[0, a_i^t]$ when the items are substitutes and from $[-a_i^t, 0]$ when the items are complements. In our tests, we consider several discrete price ladders in the range $[0.65, 1]$ (e.g., when $|Q| = 2$, we use $\{0.65, 1\}$, and when $|Q| = 3$, we use $\{0.7, 0.85, 1\}$). Interestingly, we observed that all our qualitative insights hold for both demand models, under the wide range of parameter values we considered. This suggests that our computational insights are robust to the specific demand model under consideration.

Our computational environment consists of a 2015 Macbook Pro with 16 GB of random-access memory and a 2.2-GHz processor using the Gurobi 7.02 Python interface.

5.2. Testing the Tightness

We examine the impact of both cross-item effects and cross-item business rules on the tightness of the *App*(2) approximation. In this context, *tightness* refers to whether the mixed-integer program that characterizes the *App*(2) approximation has an integral LP relaxation. In Proposition 2, we showed that for an additively separable demand with substitutable items and a price ladder of size two, the *App*(2) approximation is tight in the absence of business rules. In Figures 2 and 3, we explore the tightness of *App*(2) in a setting with substitutes and complements, respectively, and vary the size of the price ladder between two and four. For each value of N, we randomly sample 1,000 log-log demand functions from the

Figure 2. (Color online) Proportion of Tight *App*(2) Instances When All Items Are Substitutes

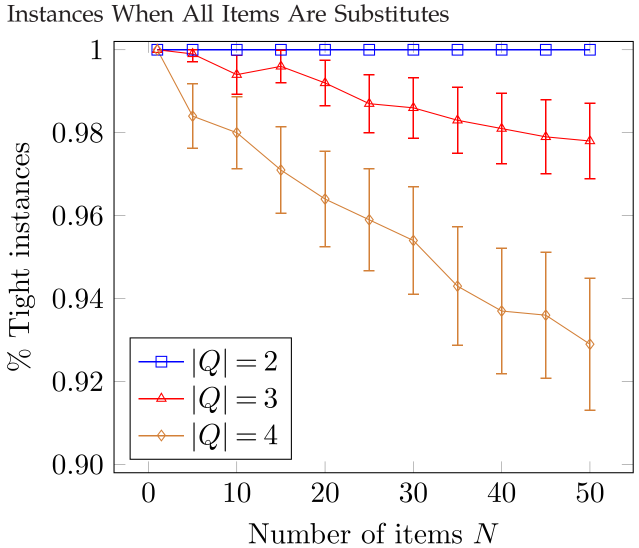


Figure 3. (Color online) Proportion of Tight *App*(2) Instances When All Items Are Complements

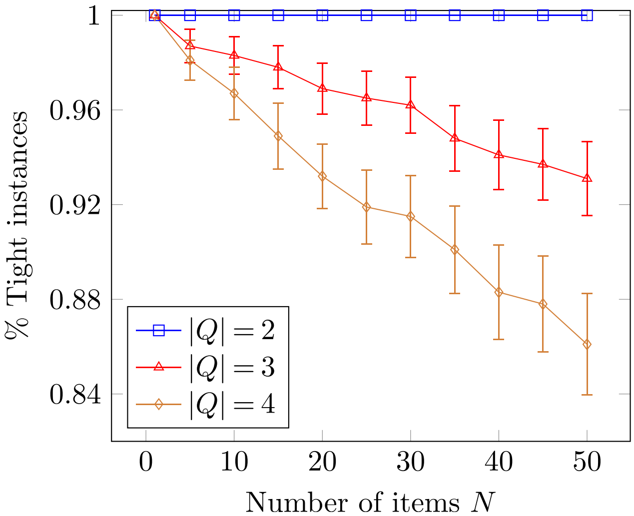
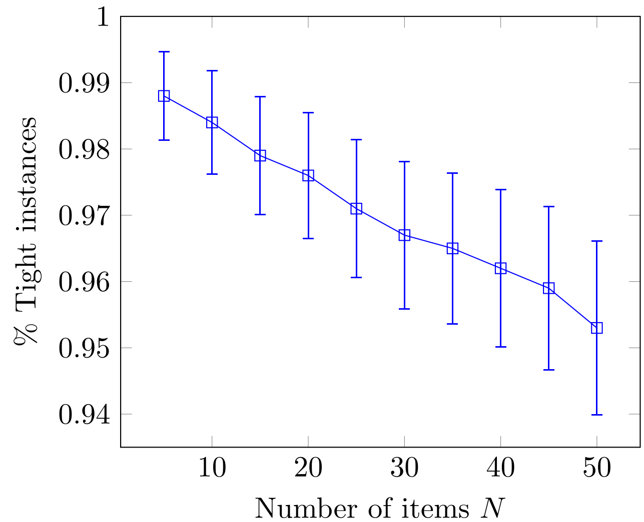


Figure 4. (Color online) Proportion of Tight *App*(2) Instances When Items Are Substitutes with at Most Five Other Items



parameter values in Table 2 and record the percentage of instances where there is no integrality gap.[7] For each value of N, we report the average value and the 95% confidence interval. As expected, we can see that under substitutable items, the *App*(2) solution is tight much more often relative to the setting with complements. In our tests, we observe that even with a price ladder with four prices, the setting with substitutes was tight 96.5% (for $N = 20$) and 93% (for $N = 50$) of the time, whereas the setting with complements was tight 93% (for $N = 20$) and 86% (for $N = 50$) of the time. Although the general *App*(2) approximation is not always tight, we will show in Section 5.4 that solving the mixed-integer program is often possible in low run times.

In a typical category of products, not all items have cross-item effects. It is common to observe that within a set of N related different items, only a small number of items are interacting. In Figures 4 and 5, we test the tightness of the *App*(2) approximation when the size of the price ladder is $|Q| = 4$, but we consider that each item can only have (nonzero) cross-item effects with at most five other items. For each data point, we generate 1,000 independent samples and report the percentage of tight instances. We can see that for both substitutes (Figure 4) and complements (Figure 5), the proportion of tight instances remains high. These tests convey that for product categories with sparse cross-item effects, the *App*(2) optimization model can be solved efficiently by running a linear program.

We next examine the impact of adding business rules to the formulation. We consider a setting with $N = 10$ items and a price ladder with $|Q| = 2$ prices. Based on 10,000 randomly generated instances, we observe that *App*(2) was always integral in the presence of self-business rules (i.e., constraints for each item separately). We then examine the impact of adding cross-item business rules on the tightness of the *App*(2) approximation. For each data point, we solve 1,000 randomly generated instances with a log-log demand function and additive mixed cross-item effects (i.e., either substitutes or complements) and record the percentage of tight instances. For simplicity, we focus on instances with $T = 1$. In Figure 6, we test the impact of imposing a limit on the total number of promotions L_T, as discussed in (2). We can see that the problem tends to be tight when L_T is either large or small. In Figure 7, we consider a setting with $N = 10$, $T = 10$, and $|Q| = 2$ and test the impact of

Figure 5. (Color online) Proportion of Tight *App*(2) Instances When Items Are Complements with at Most Five Other Items

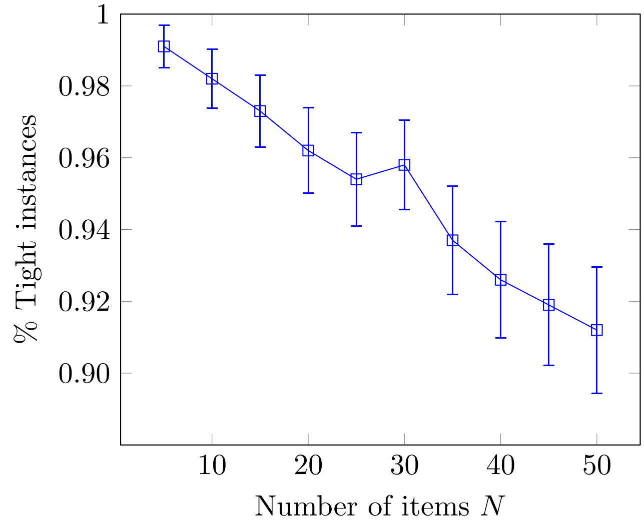
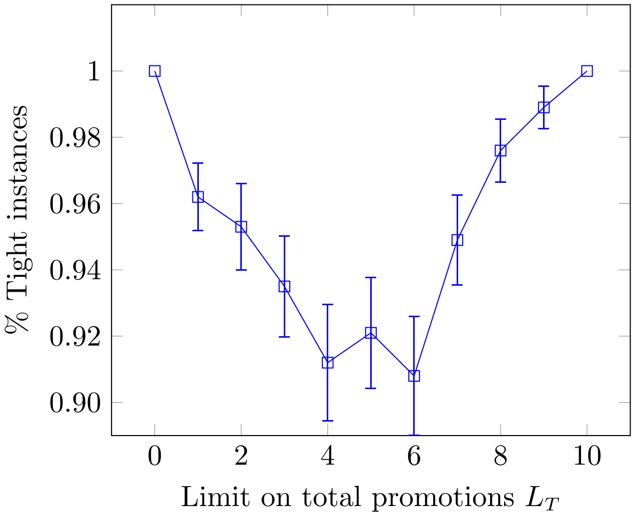


Figure 6. (Color online) Proportion of Tight *App*(2) Instances with a Constraint on the Number of Promotions



imposing a cross no-touch constraint in which successive promotions for any pair of items need to be spaced out by at least S^c periods. For each value of S^c, we generate and solve 1,000 *App*(2) instances and record the percentage of tight instances. Even in the presence of a cross no-touch constraint, the *App*(2) model is tight more than 90% of the time in our tested instances. For settings where the *App*(2) model is not guaranteed to be tight (e.g., a price ladder with more than two prices or in the presence of business rules), we propose to use one of the two following solutions: (1) implementing our proposed rounding schemes or (2) directly solving the mixed-integer program formulation instead of relying on the LP relaxation. In Section 5.3, we test our rounding schemes, whereas in Section 5.4, we show computationally that for

Figure 7. (Color online) Proportion of Tight *App*(2) Instances with a Cross No-Touch Constraint

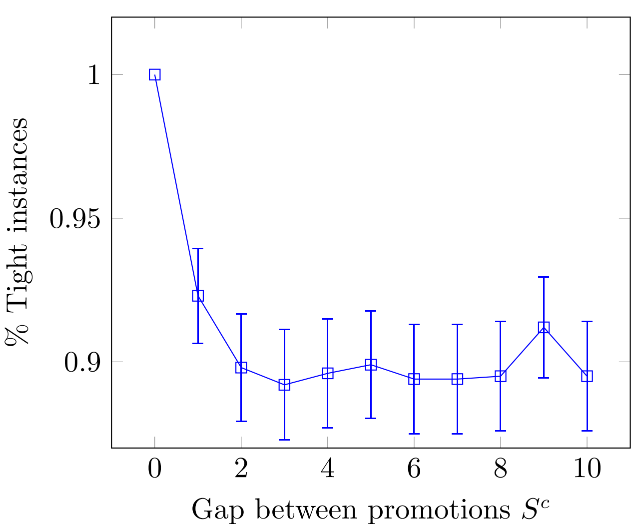


Table 3. Range of Parameters for Comparing the Rounding Schemes

Parameter	Range
a^i	[500,1,000]
b_0^i	[2,7]
δ^{ij}	$[-a^i, a^i]$
c^i	[0.45,0.55]
Q	$[\max c_i, 1]$
N	$\{5,6,\ldots,20\}$

many tests, the mixed-integer program solves within acceptable run times for retail applications.

5.3. Testing the Rounding Schemes

We next conduct computational tests to compare our different rounding schemes. We consider the following demand model for item *i*:

$$d^i = a^i \exp\{-b_0^i \log(p^i)\} + \sum_{j=1}^{N} \delta^{ij} p^j. \tag{14}$$

The range of parameter values is reported in Table 3. We randomly generate 5,000 independent instances for each value of the number of items N. To keep our tests general, the items in each instance can be substitutable, complementary, or not connected to the other items. For simplicity of exposition, we consider instances with small values of N. We will consider large-scale practical instances in Section 5.6.

We first plot the proportion of instances with the same solution as the optimal mixed-integer program solution in Figure 8. As we can see, the naive rounding scheme performs the worst, whereas our three proposed rounding schemes often lead to the same solution as the mixed-integer program.

We then plot the average and maximum (Figure 9) difference between the objective value and the optimal

Figure 8. (Color online) Proportion of Instances with the Same Solution as the Mixed-Integer Program

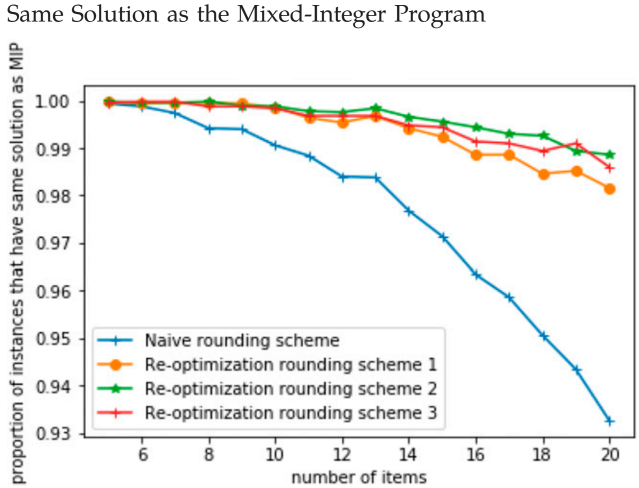
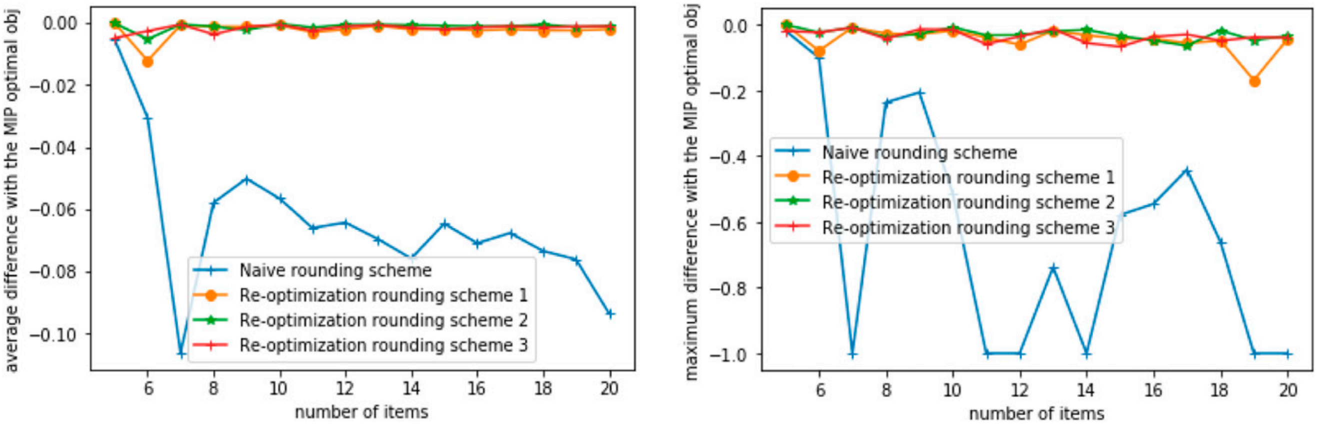


Figure 9. (Color online) Average and Maximum Difference between the Objective Value and the Optimal Objective



mixed-integer program objective for each rounding scheme. As expected, the naive rounding scheme can yield a low performance: an average loss of 10% and a maximum loss of 100% (such an outcome happens when all x_i values are fractional in the LP original solution). By contrast, our three proposed rounding schemes lead to a near-optimal solution: the average gap is lower than 2% in all our tested instances for all three rounding schemes. As expected, the performance of Rounding Scheme 2 is always slightly better than the performance of Rounding Scheme 1.

Finally, we report the average and maximum (Figure 10) number of reoptimizing iterations (i.e., the number of iterations needed to reach an integral solution) for our three proposed rounding schemes. For Rounding Schemes 1 and 3, each iteration corresponds to solving a single linear program, whereas for Rounding Scheme 2, it can correspond to solving several linear programs. We observe that the number of reoptimizing iterations remains small across all three rounding schemes. As expected, Rounding Scheme 2 usually requires a larger number of resolving iterations.

Given that Rounding Scheme 2 needs a larger number of iterations than Rounding Scheme 1, the slight performance improvement may not be enough to justify using Rounding Scheme 2. In summary, we conclude that the greedy rounding scheme (i.e., reoptimization Rounding Scheme 1) is a good candidate for solving our problem: (1) it yields the same solution as the mixed-integer program most of the time, (2) when it does not, the optimality gap is low, and (3) the required number of resolving iterations is small—thus making this scheme efficient and applicable to large instances, as illustrated in Section 5.6.

5.4. Testing the Run Times of the Mixed-Integer Program

We next investigate how the run time of *App*(2) is affected by the different input parameters. We use two different performance metrics: (1) the time to build the model, that is, computing the coefficients of all unilateral and bilateral contributions, called *building time*, and (2) the time to solve the model via the mixed-integer program, called *solving time*. In Figures 11 and 12, we report the model building and

Figure 10. (Color online) Average and Maximum Number of Reoptimizing Iterations

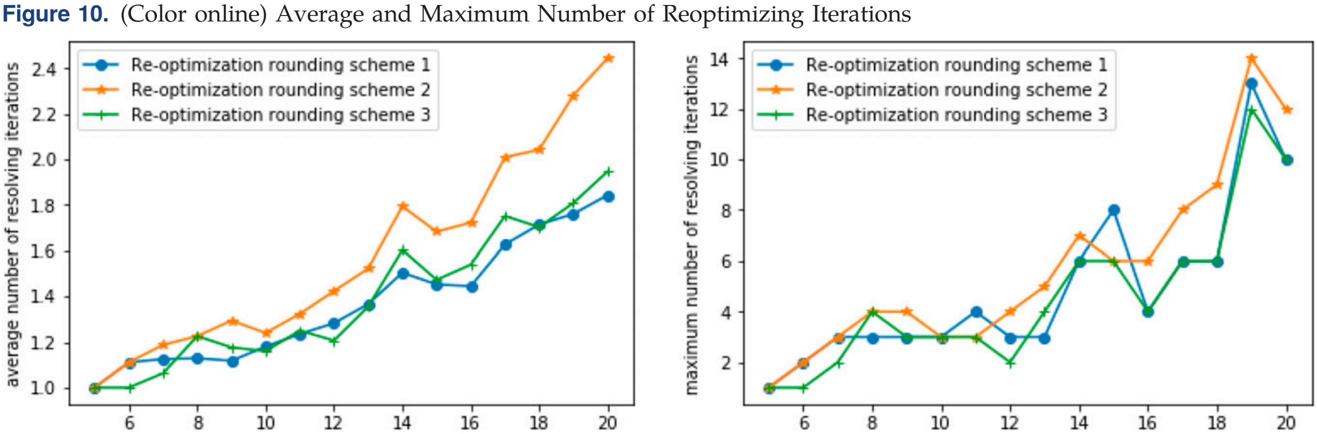


Figure 11. (Color online) Building Time of the *App*(2) Mixed-Integer Program Model

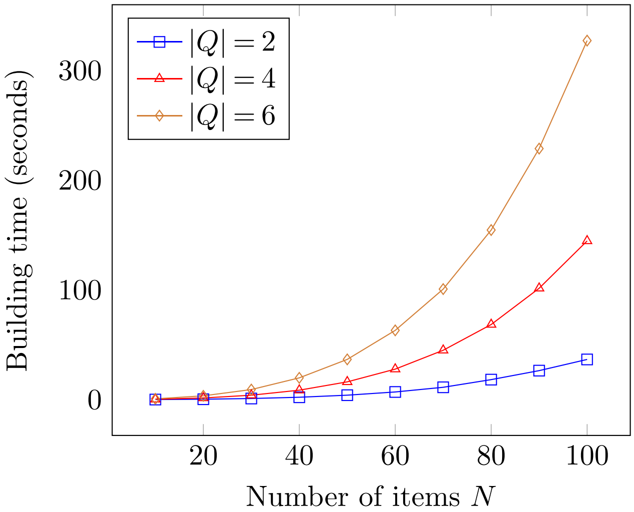
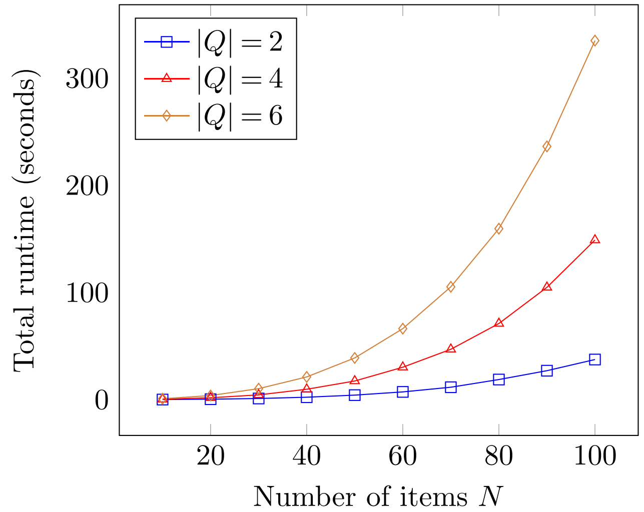


Figure 13. (Color online) Total Run Time of the *App*(2) Mixed-Integer Program Model



solving times, respectively. We consider a setting with $T = 1$, vary the size of the price ladder, and vary the number of items between 10 and 100. For each value of N, we randomly sample 100 independent log-log demand functions with additive cross-item effects. We then record the maximum run time (in seconds) observed over the 100 samples.

One can see that most of the total run time is spent building the model. Building the model consists of calculating all the unilateral and bilateral coefficients needed for the *App*(2) formulation (we have $O(N^2|Q|^2T)$ such coefficients).[8] Nevertheless, once the model is built, solving a single instance can be done very fast. In practice, this is a desirable feature because the addition or modification of business rules does not affect the model, so we do not need to build the model again when varying the specifics of the business rules. As a result, it can allow retailers to efficiently test several what-if scenarios with respect to changes in business rules.

Finally, in Figures 13 and 14, we examine the total run time to build and solve the mixed-integer program. In Figure 13, we consider a setting with $T = 1$ and randomly sample 100 log-log demand functions with additive cross-item effects while varying N and $|Q|$. In Figure 14, we use $|Q| = 2$ and vary N over a wider range of values. For each test, we record the maximum run time over the 100 samples. We can see that the *App*(2) method can be solved efficiently even for categories with more than 100 items. In addition, for settings with a price ladder of size two (i.e., facing decisions of whether to promote or not), we can solve

Figure 12. (Color online) Solving Time of the *App*(2) Mixed-Integer Program Model

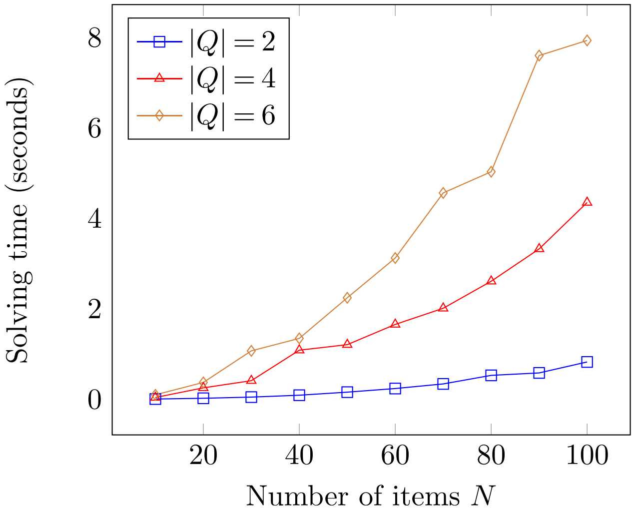
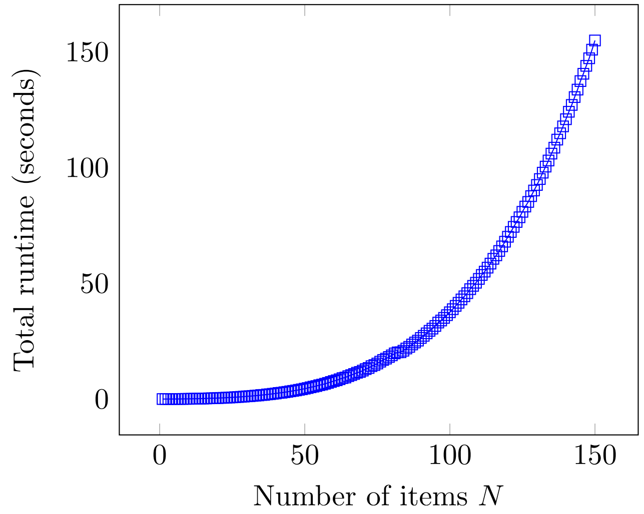


Figure 14. (Color online) Total Run Time of the *App*(2) Mixed-Integer Program Model with $T = 1$ and $|Q| = 2$



instances with a large number of items in a reasonable timeframe.

5.5. Interplay of Time and Cross-Item Effects

In addition to testing the performance of the *App*(2) approximation, we conduct computational experiments to observe general trends in the optimal promotion strategy in the presence of time effects (i.e., postpromotion dip) and cross-item effects (i.e., substitution and complementarity). In this section, we consider a realistic setting with $T = 35$, $N = 10$ symmetric items, and a price ladder with four price points (1, 0.9, 0.8, and 0.7). We use a linear demand model calibrated with data from the coffee category of a large retailer with $b_0^i = 50$, $\alpha = 15$, and vary the magnitude of the cross-item effects δ between −50 and 50. Our goal is to examine the interplay between α and δ, as well as their impact on the optimal promotion strategy. For each value of δ, we report the optimal number of promotions from the *App*(2) solution in Figure 15. We can see that for complementary items (i.e., when $\delta < 0$), the *App*(2) solution leads to offering more promotions, whereas the number of promotions decreases for substitutable items. This outcome is expected because when δ becomes large, promoting an item decreases the sales of other items via a cannibalization effect.

We next vary the time-effect parameter α for different values of δ. We consider the same setting as in Figure 15, but we now vary the effect of past prices on current demand. Specifically, we consider a linear demand model with a memory of three past prices ($M_i = 3$ for all i), where the effect of the past price p_{t-i} on current demand is equal to $\alpha(\frac{1}{2})^{i-1}$ for $i \in (1, 2, 3)$ and zero otherwise. In Figure 16, we report the total number of promotions from solving *App*(2) as a function of α for three different values of δ. We can see that the number of promotions increases as α decreases. This follows from the fact that when α is larger, promotions will reduce future demand and hence are less desirable. In addition, we can see that in a setting with complements (i.e., when $\delta < 0$), the optimal strategy will have a higher number of promotions.

Figure 15. (Color online) Total Number of Promotions as a Function of Δ When $T = 35$, $N = 10$, and $|Q| = 4$

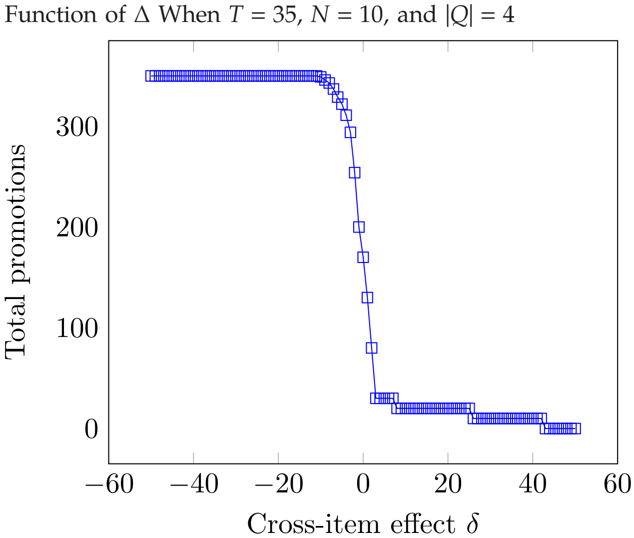
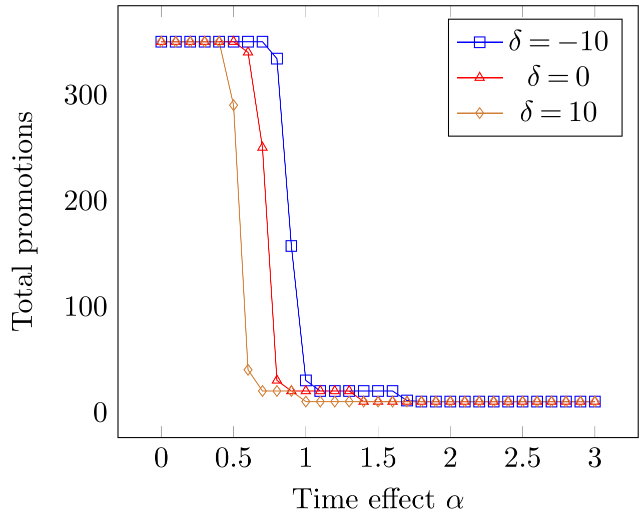


Figure 16. (Color online) Total Number of Promotions as a Function of α When $T = 35$, $N = 10$, and $|Q| = 4$



5.6. Computational Experiments for Larger Instances

In this section, we consider large-scale practical instances. In most retail categories (e.g., soft drinks, cereals, and yogurts), there can be 30–150 different items. We vary the number of items $N = 50, 100, 150, 200$, and 250. For each value of N, we consider several independent instances. As earlier, we consider that the demand model for item i is given by (14). The range of parameters and number of instances are reported in Tables 4 and 5 (for efficient computations, we decrease the number of independent instances as N increases). For simplicity, we consider two prices (i.e., $K^i = 1$) and a single time period (i.e., $T = 1$).

For each instance, we (1) solve the mixed-integer program, (2) solve the *App*(2) approximation solution

Table 4. Range of Parameters for Section 5.6

Parameter	Range
a^i	[500,1,000]
b_0^i	[2,7]
δ^{ij}	$[-a^i/5, a^i/5]$
c^i	[0.45,0.55]
Q	$[\max c_i, 1]$
N	{50,100, . . . ,250}

Table 5. Number of Instances as a Function of N

N	No. of instances
50	1,000
100	500
150	200
200	100
250	50

via LP relaxation, and (3) compute the proportion of tight instances. Given that we do not restrict the parameters in any way, we obtain a large proportion of instances that are not integral, especially when N is large. We can see from the left panel of Figure 17 that the percentage of tight instances decreases rapidly as N increases.

For nonintegral instances, we implement the greedy reoptimization rounding scheme (Rounding Scheme 1) and compare its solution to the optimal mixed-integer program solution in terms of both the objective value and the run time (see the right panel of Figure 17, where we report the maximum run times for different values of N). When the number of items is large, it can take up to several days to solve the mixed-integer program: it took 500,000 seconds = 5.787 days for $N = 250$. However, using Rounding Scheme 1, we can obtain a feasible integral solution in 1–20 minutes. Furthermore, the objective value was always within 95% of the mixed-integer program optimal objective in all our tested instances. Finally, for a realistic instance with $N = 100$ and $T = 10$, our method runs in 2–4 minutes.

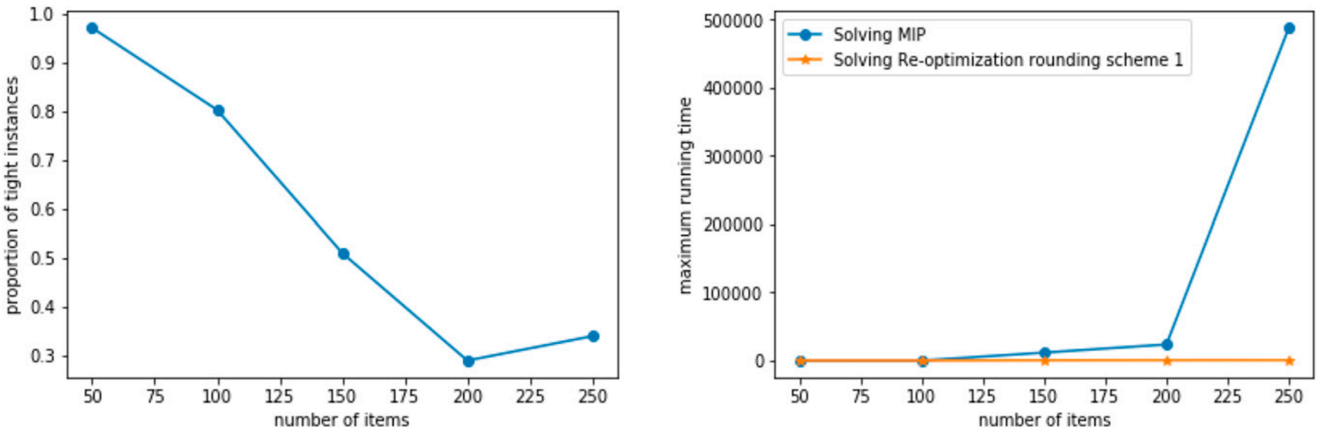
6. Conclusions

In many retail settings, promotions are a key instrument for driving sales and profits. In a typical supermarket, each category manager needs to decide on the promotion strategy for multiple items during the selling season. The large volume of available data allows retailers to improve demand forecasts. The next step is to use these accurate demand forecasting models to carefully decide future promotions to maximize profits. In this paper, we introduce and study a practical optimization formulation for deciding on the promotions of multiple items. Our model captures important business requirements and encompasses common demand models calibrated from data. Solving this problem is relevant to many retailers and can significantly enhance their bottom line. However, optimally solving this problem is challenging because it involves a large-scale nonlinear integer program. We propose a tractable approximation method for solving this problem and present analytical and computational results on the performance of the approximation solution.

We first show that a method based on linearizing the objective does not perform well because it was for a single item. We next consider a modified approach, called *App*(2), based on approximating the objective with unilateral and pairwise contributions. We show that when the demand has additively separable cross-item effects, *App*(2) accurately captures the cross-item effects without computing an exponential number of coefficients. We further prove that when the items are substitutable and the retailer decides between two prices, the *App*(2) formulation admits a tight LP relaxation in the absence of business rules. For cases where the LP relaxation is not tight, we propose three efficient rounding schemes. Armed with these results, we show that the *App*(2) method yields a solution with a parametric worst-case guarantee relative to the optimal solution.

We next use our model to study the interplay between the postpromotion dip effect and cross-item effects. We convey that when the degree of substitution increases, it becomes optimal to reduce the

Figure 17. (Color online) Proportion of Tight Instances and Maximum Run Time (in Seconds)



number of promotions. We also conduct several computational tests to show that the $App(2)$ method can be applied successfully in realistic instances. We first test the tightness of the LP relaxation when the sufficient conditions are relaxed. In most of our tested instances, solving the LP relaxation leads to an integer solution. We then show that solving the mixed-integer program is relatively fast and can be done in reasonable run times. Finally, we use our method to draw managerial insights on the interplay of the time and cross-item effects. More generally, the results presented in this research can guide retailers to test different promotion strategies and sharpen their understanding of promotion planning.

Acknowledgments

The authors thank the department editor (Yinyu Ye), the associate editor, and the anonymous referees for insightful comments that have helped improve this paper; Jingyi Wang, who helped us analyze and implement the rounding schemes; and Lennart Baardman, Tamar Cohen-Hillel, Arthur Flajolet, Kevin Jiao, and Kiran Panchamgam for insightful comments that helped improve this paper.

Appendix. Proofs

Proof of Theorem 1. We show that all the coefficients of order three and higher in the $App(N)$ formulation are equal to zero. It will then allow us to conclude that $App(2) = App(N)$. We first show that all the coefficients of order three (i.e., the sets with three items simultaneously on promotion) are zero. We then proceed by induction.

Recall that the coefficient for items i, j, and k at any time t is given by (we drop the time index for clarity)

$$\begin{aligned} b^{ijk} = {} & MPOP(i,j,k) - MPOP(i,j) - MPOP(i,k) \\ & - MPOP(j,k) + MPOP(i) + MPOP(j) \\ & + MPOP(k) - MPOP(0). \end{aligned} \tag{A.1}$$

Here $MPOP(i,j,k)$ denotes the total profit generated by the N items throughout the T periods when only items i, j, and k are on promotion at time t. Similarly, $MPOP(j)$ denotes the total profit when only item j is on promotion at time t. Note that all the terms in (A.1) at times different from t are zero. Note also that the coefficient in (A.1) affects only items i, j, and k and is zero otherwise. As a result, we remain only with three contributions (for items i, j, and k at time t). From symmetry in the indices, it is sufficient to consider item i and show that its contribution is zero. The contribution of (A.1) for item i at time t is

$$\begin{aligned} \left[b^{ijk}\right]_{i,t} = {} & \left(p_t^i - c_t^i\right)\Bigg[h_t^i\left(p_t^i, q^0, \ldots, q^0\right) + H_t^{ji}\left(p_t^j\right) \\ & + H_t^{ki}\left(p_t^k\right) + \sum_{\ell\neq i,\ell\neq j,\ell\neq k} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & - \left(p_t^i - c_t^i\right)\Bigg[h_t^i\left(p_t^i, q^0, \ldots, q^0\right) + H_t^{ji}\left(p_t^j\right) + \sum_{\ell\neq i,\ell\neq j} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & - \left(p_t^i - c_t^i\right)\Bigg[h_t^i\left(p_t^i, q^0, \ldots, q^0\right) + H_t^{ki}\left(p_t^k\right) + \sum_{\ell\neq i,\ell\neq k} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & - \left(q^0 - c_t^i\right)\Bigg[h_t^i\left(q^0, \ldots, q^0\right) + H_t^{ji}\left(p_t^j\right) \\ & + H_t^{ki}\left(p_t^k\right) + \sum_{\ell\neq i,\ell\neq j,\ell\neq k} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & + \left(p_t^i - c_t^i\right)\Bigg[h_t^i\left(p_t^i, q^0, \ldots, q^0\right) + \sum_{\ell\neq i} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & + \left(q^0 - c_t^i\right)\Bigg[h_t^i\left(q^0, \ldots, q^0\right) + H_t^{ji}\left(p_t^j\right) + \sum_{\ell\neq i,\ell\neq j} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & + \left(q^0 - c_t^i\right)\Bigg[h_t^i\left(q^0, \ldots, q^0\right) + H_t^{ki}\left(p_t^k\right) + \sum_{\ell\neq i,\ell\neq k} H_t^{\ell i}\left(q^0\right)\Bigg] \\ & - \left(q^0 - c_t^i\right)\Bigg[h_t^i\left(q^0, \ldots, q^0\right) + \sum_{\ell\neq i} H_t^{\ell i}\left(q^0\right)\Bigg] \\ = {} & \left(q^0 - c_t^i\right)[0] + \left(p_t^i - c_t^i\right)[0] = 0. \end{aligned}$$

The last step follows from canceling terms (the details are omitted for conciseness). Indeed, by using the additively separable assumption of the demand function, we find that the terms cancel each other out. Therefore, all the coefficients with three items simultaneously on promotion are equal to zero. We assume by induction that all the coefficients with $S = 4, 5, \ldots, K-1$ items are zero for some given $K - 1 < N$. We next show that the claim is true for K.

Note that we have several coefficients that include K items (one such coefficient for any subset of the N items with size K). For example, the coefficient for items $1, 2, \ldots, K$ at time t is given by

$$\begin{aligned} b^{12\ldots K} = {} & MPOP(1, 2, \ldots, K) - [\text{All the} K-1] \\ & + [\text{All the } K-2] - \ldots MPOP(0), \end{aligned}$$

where the term [All the $K-1$] refers to all the contributions of having a total of $K-1$ out of the K items on promotion.

Alternatively, we can express the coefficient $b^{12\dots K}$ as a function of the smaller-order coefficients as follows:

$$\begin{aligned}b^{12\dots K} &= MPOP(1,2,\dots,K) - MPOP(0)\\ &\quad -\sum[\text{All } bs \text{ with } K-1 \text{ items}]\\ &\quad -\sum[\text{All the } bs \text{ with } K-2 \text{ items}]\\ &\quad -\dots-\sum_{\text{pairs}} b^{ij} - \sum_{j=1}^{K} b^{j}.\end{aligned}$$

By using the induction hypothesis, we obtain:

$$b^{12\dots K} = MPOP(1,2,\dots,K) - MPOP(0)\sum_{\text{pairs}} b^{ij} - \sum_{j=1}^{K} b^{j}. \tag{A.2}$$

Here $MPOP(1,2,\dots,K)$ denotes the total profit generated by all N items during the T periods, when only items $1,2,\dots,K$ are on promotion at time t. We next look at the different types of contributions of the coefficient in (A.2). We note that all the terms at times different from t are zero. We also note that the coefficient in (A.2) affects only the items $1,2,\dots,K$ and is zero otherwise. From symmetry in the indices, it is sufficient to consider item i and show that its contribution is zero. For item i at time t, we have

$$\begin{aligned}[b^{12\dots K}]_{i,t} &= (p_t^i - c_t^i)\Bigg[h_t^i(p_t^i, q^0,\dots,q^0) + \sum_{\ell=1,2,\dots,K,\ell\neq i} H_t^{\ell i}(p_t^\ell)\\ &\quad + \sum_{\ell\neq 1,2,\dots,K,\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &\quad - (q^0 - c_t^i)\Bigg[h_t^i(q^0,\dots,q^0) + \sum_{\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &\quad - \sum_{\text{pairs } (i,n); i=1,2,\dots,K,n\neq i}\Bigg\{(p_t^i - c_t^i)\Bigg[h_t^i(p_t^i,q^0,\dots,q^0)\\ &\quad + H_t^{ni}(p_t^n)\\ &\quad + \sum_{j\neq i,j\neq n} H_t^{ji}(q^0) - h_t^i(p_t^i,q^0,\dots,q^0) - \sum_{j\neq i} H_t^{ji}(q^0)\Bigg]\\ &\quad - (q^0 - c_t^i)\Bigg[h_t^i(q^0,\dots,q^0) + H_t^{ni}(p_t^n)\\ &\quad + \sum_{j\neq i,j\neq n} H_t^{ji}(q^0) - h_t^i(q^0,\dots,q^0) + \sum_{j\neq i} H_t^{ji}(q^0)\Bigg]\Bigg\}\\ &\quad - (p_t^i - c_t^i)\Bigg[h_t^i(p_t^i,q^0,\dots,q^0) + \sum_{\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &\quad + (q^0 - c_t^i)\Bigg[h_t^i(q^0,\dots,q^0) + \sum_{\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &\quad - \sum_{j=1,\dots,K,j\neq i}(q^0 - c_t^i)\Big[H_t^{ji}(p_t^j) - H_t^{ji}(q^0)\Big]\\ &= (q^0 - c_t^i)[0] + (p_t^i - c_t^i)[0] = 0.\end{aligned}$$

As before, the last step follows from canceling terms. Note that all pairs of items where item i is not included have a contribution of zero. □

Proof of Corollary 1. Corollary 1 can be shown in a similar way as the proof of Theorem 1 by refining the definition of $App(2)$ to include all the pairwise contributions of two simultaneous promotions: both for different items at the same period and for the same item within M^i consecutive periods. In this case, when the function $h_t^i(p_t^i, p_{t-1}^i,\dots,p_{t-M^i}^i)$ is additively separable for each item, we find that the resulting objective function can be expressed as a sum of functions, and each function depends on at most two binary decision variables. Consequently, the same argument as in Theorem 1 applies. □

Proof of Lemma 1. Recall that the pairwise coefficient for items i and j at time t is given by (we drop the time index for clarity)

$$b^{ij} = MPOP(i,j) - MPOP(i) - MPOP(j) + MPOP(0). \tag{A.3}$$

As before, $MPOP(i,j)$ denotes the total profit generated by all N items during the T periods when only items i and j are on promotion at time t. We note that all the terms at times different from t are zero. We also note that the coefficient in (A.3) has three different types of contributions (for items i, j, and $k \neq i, j$ at time t). We next show that each one of the three contributions is nonnegative.

1. For item i at time t, we have

$$\begin{aligned}[b^{ij}]_{i,t} &= (p_t^i - c_t^i)\Bigg[h_t^i(p_t^i,q^0,\dots,q^0) + H_t^{ji}(p_t^j) + \sum_{\ell\neq i,\ell\neq j} H_t^{\ell i}(q^0)\Bigg]\\ &\quad - (p_t^i - c_t^i)\Bigg[h_t^i(p_t^i,q^0,\dots,q^0) + \sum_{\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &\quad - (q^0 - c_t^i)\Bigg[h_t^i(q^0,\dots,q^0) + H_t^{ji}(p_t^j) + \sum_{\ell\neq i,\ell\neq j} H_t^{\ell i}(q^0)\Bigg]\\ &\quad + (q^0 - c_t^i)\Bigg[h_t^i(q^0,\dots,q^0) + \sum_{\ell\neq i} H_t^{\ell i}(q^0)\Bigg]\\ &= (q^0 - c_t^i)\Big[H_t^{ji}(q^0) - H_t^{ji}(p_t^j)\Big] + (p_t^i - c_t^i)\Big[H_t^{ji}(p_t^j) - H_t^{ji}(q^0)\Big]\\ &= (q^0 - p_t^i)\Big[H_t^{ji}(q^0) - H_t^{ji}(p_t^j)\Big].\end{aligned}$$

We know that $q^0 \geq p_t^i$ (recall that q^0 is the regular price and p_t^i is a promotion price). Because items i and j are substitutes, the function $H_t^{ji}(\cdot)$ is nondecreasing, and thus, $H_t^{ji}(q^0) - H_t^{ji}(p_t^j) \geq 0$. Consequently, the preceding contribution is nonnegative.

2. For item j at time t, the exact same argument follows by symmetry.

3. For any item $k \neq i, j$ at time t, we have

$$\begin{aligned}[b^{ij}]_{k,t} &= (q^0 - c_k^t)\Bigg[h_t^k(q^0) + H_t^{ik}(p_t^i) + H_t^{jk}(p_t^j) + \sum_{\ell\neq i,\ell\neq j,\ell\neq k} H_t^{\ell k}(q^0)\Bigg]\\ &\quad - (q^0 - c_k^t)\Bigg[h_t^k(q^0) + H_t^{ik}(p_t^i) + \sum_{\ell\neq i,\ell\neq k} H_t^{\ell k}(q^0)\Bigg]\\ &\quad - (q^0 - c_k^t)\Bigg[h_t^k(q^0) + H_t^{jk}(p_t^j) + \sum_{\ell\neq j,\ell\neq k} H_t^{\ell k}(q^0)\Bigg]\\ &\quad + (q^0 - c_k^t)\Bigg[h_t^k(q^0) + \sum_{\ell\neq k} H_t^{\ell k}(q^0)\Bigg]\\ &= -(q^0 - c_k^t)\Big[H_t^{jk}(q^0) - H_t^{jk}(q^0)\Big] = 0. \quad \square\end{aligned}$$

Management Science, 2021, vol. 67, no. 4, pp. 2340–2364, © 2020 INFORMS

Proof of Proposition 1. If $x_k^* = \frac{1}{2}$ in the original solution, then there are two possibilities for any integral variable x_j that is connected with x_k (i.e., $b_{kj} \neq 0$): (1) $x_k = \frac{1}{2}, x_j = 1, x_{kj} = \frac{1}{2}$ or (2) $x_k = \frac{1}{2}, x_j = 0, x_{kj} = 0$. Without loss of generality, we assume that there is one x_j^* from each of these two cases and denote these two variables $x_{j_1}^*, x_{j_2}^*$, respectively.

We can see that there must be at least one fractional solution that is connected to x_k because otherwise we can increase the objective value by changing x_k to either one or zero while maintaining the other variables at the same values. Without loss of generality, we assume that there is one such fractional solution and denote it $x_{j_3}^*$.

We denote the original optimal solution as X and its objective value as $J(X)$. We assume by contradiction that the original integral $x_{j_i}^*$ does not remain the same in the new reoptimized solution. We denote the new optimal solution as X' and its objective value as $J'(X)$. We separate the analysis into two cases.

Case 1: $b_k \geq 0$. In this case, the new solution is such that $x_k^{'} = 1$. By assumption, there are two possibilities for $x_{j_1}^{'}$ and $x_{kj_1}^{'}$ in the new optimal solution: (a) $x_{j_1}^{'} = \frac{1}{2}$, $x_{kj_1}^{'} = \frac{1}{2}$, or (b) $x_{j_1}^{'} = 0$, $x_{kj_1}^{'} = 0$. Similarly, for $x_{j_2}^{'}$ and $x_{kj_2}^{'}$, the two possibilities are (a) $x_{j_2}^{'} = \frac{1}{2}$, $x_{kj_2}^{'} = \frac{1}{2}$, and (b) $x_{j_2}^{'} = 1$, $x_{kj_2}^{'} = \frac{1}{2}$. Because our proof does not rely on any connection between $x_{j_1}^{'}$ and $x_{j_2}^{'}$, we analyze these four scenarios separately and show that none of them is possible for X'.

1a. If $x_{j_1}^{'} = \frac{1}{2}$ and $x_{kj_1}^{'} = \frac{1}{2}$, then we have $J^{'}(X) = b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} b_{kj_1} + W_1$, where W_1 denotes the remaining part of the objective, that is, all the contributions from the variables that are not connected to k. With the additional constraint $x_k^{'} = 1$, we can obtain a feasible solution $\tilde{X}'$, with $x_{j_1}^{'} = 1, x_{kj_1}^{'} = 1$, so that $J^{'}(\tilde{X}^{'}) = b_k + b_{j_1} + b_{kj_1} W_0$, where W_0 is the remaining part of the objective of $\tilde{X}'$. By the optimality of $X^{'}$, we have

$$b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} b_{kj_1} + W_1 > b_k + b_{j_1} + b_{kj_1} + W_0.$$

By comparing X and $\tilde{X}'$, we can see that only x_k and x_{kj_1} are changing (otherwise, it would contradict the optimality of X). As a result, W_0 is also the remaining part of the original objective, and thus, $J(X) = \frac{1}{2} b_k + b_{j_1} + \frac{1}{2} b_{kj_1} + W_0$. Meanwhile, $\tilde{X}$ is also feasible for the original problem without the additional constraint, which is $x_k = \frac{1}{2}, x_{j_1} = \frac{1}{2}, x_{kj_1} = \frac{1}{2}$ (if $b_{kj_1} \geq 0$) and $x_{kj_1} = 0$ (if $b_{kj_1} < 0$), so $J(\tilde{X}) = \frac{1}{2} b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_1$. Because X is an optimal solution of the original problem, we have

$$\frac{1}{2} b_k + b_{j_1} + \frac{1}{2} b_{kj_1} + W_0 \geq \frac{1}{2} b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_1.$$

By summing up these two inequalities, we obtain

$$\begin{aligned}&\frac{3}{2} b_k + \frac{3}{2} b_{j_1} + b_{kj_1} + W_0 + W_1 \\ &\quad > \frac{3}{2} b_k + \frac{3}{2} b_{j_1} + b_{kj_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_0 + W_1,\end{aligned}$$

which is equivalent to $0 > \max\{b_{kj_1}, 0\}$, so we reach a contradiction.

1b. If $x_{j_2}^{'} = \frac{1}{2}$ and $x_{kj_2}^{'} = \frac{1}{2}$, then we have $J'(X) = b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} b_{kj_2} + W_1$, where W_1 denotes the remaining part of the objective. With the additional constraint $x_k' = 1$, we can obtain a feasible solution $\tilde{X}'$ with $x_{j_2}^{'} = 0, x_{kj_2}^{'} = 0$, so $J'(\tilde{X}) = b_k + W_0$, where W_0 is the remaining part of $\tilde{X}'$. By using the optimality of $X^{'}$, we can write

$$b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} b_{kj_2} + W_1 > b_k + W_0.$$

As in 1a, we can see that W_0 is also the remaining part of the original objective, so $J(X) = \frac{1}{2} b_k + W_0$. Meanwhile, $\tilde{X}$ is also feasible for the original problem without the additional constraint, which is $x_k = \frac{1}{2}, x_{j_2} = \frac{1}{2}$, $x_{kj_2} = \frac{1}{2}$ (if $b_{kj_2} \geq 0$) and $x_{kj_2} = 0$ (if $b_{kj_2} < 0$), so $J(\tilde{X}) = \frac{1}{2} b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} \max\{b_{kj_2}, 0\} + W_1$. Because X is an optimal solution of the original problem, we have

$$\frac{1}{2} b_k + W_0 \geq \frac{1}{2} b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} \max\{b_{kj_2}, 0\} + W_1.$$

By summing up these two inequalities, we obtain

$$\begin{aligned}&\frac{3}{2} b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} b_{kj_2} + W_0 + W_1 \\ &\quad > \frac{3}{2} b_k + \frac{1}{2} b_{j_2} + \frac{1}{2} \max\{b_{kj_2}, 0\} + W_0 + W_1,\end{aligned}$$

which is equivalent to $b_{kj_2} > \max\{b_{kj_2}, 0\}$, so we reach a contradiction.

Case 2: $b_k < 0$. In this case, the new solution is such that $x_k' = 0$. By assumption, there are two possibilities for $x_{j_1}^{'}$ and $x_{kj_1}^{'}$ in the new optimal solution: (a) $x_{j_1}^{'} = \frac{1}{2}$, $x_{kj_1}^{'} = 0$, or (b) $x_{j_1}^{'} = 0$, $x_{kj_1}^{'} = 0$. Similarly, for $x_{j_2}^{'}$ and $x_{kj_2}^{'}$, the two possibilities are (a) $x_{j_2}^{'} = \frac{1}{2}$, $x_{kj_2}^{'} = 0$, and (b) $x_{j_2}^{'} = 1$, $x_{kj_2}^{'} = 0$. As before, because our proof does not rely on any connection between $x_{j_1}^{'}$ and $x_{j_2}^{'}$, we analyze these four scenarios separately and show that none of them is possible for X'.

1a. If $x_{j_1}^{'} = \frac{1}{2}, x_{kj_1}^{'} = 0$, then we have $J^{'}(X) = \frac{1}{2} b_{j_1} + W_1$, where W_1 denotes the remaining part of the objective. With the additional constraint $x_k^{'} = 0$, we can obtain a feasible solution $\tilde{X}'$ with $x_{j_1}^{'} = 1, x_{kj_1}^{'} = 0$, so $J^{'}(\tilde{X}) = b_{j_1} + W_0$, where W_0 is the remaining part of the objective of $\tilde{X}'$. Because X is an optimal solution of the original problem, we have

$$\frac{1}{2} b_{j_1} + W_1 > b_{j_1} + W_0.$$

By comparing X and $\tilde{X}'$, we can see that only x_k and x_{kj_1} are changing (otherwise, it would contradict the optimality of X). Therefore, W_0 is also the remaining part of the original objective, so $J(X) = \frac{1}{2} b_k + b_{j_1} + \frac{1}{2} b_{kj_1} + W_0$. Meanwhile, $\tilde{X}$ is also feasible for the original problem without the additional constraint, which is $x_k = \frac{1}{2}$, $x_{j_1} = \frac{1}{2}$, $x_{kj_1} = \frac{1}{2}$ (if $b_{kj_1} \geq 0$), and $x_{kj_1} = 0$ (if $b_{kj_1} < 0$), so $J(\tilde{X}) = \frac{1}{2} b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_1$. Because X is an optimal solution of the original problem, we have

$$\frac{1}{2} b_k + b_{j_1} + \frac{1}{2} b_{kj_1} + W_0 \geq \frac{1}{2} b_k + \frac{1}{2} b_{j_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_1.$$

By summing up these two inequalities, we obtain

$$\begin{aligned}&\frac{1}{2} b_k + \frac{3}{2} b_{j_1} + \frac{1}{2} b_{kj_1} + W_0 + W_1 \\ &\quad > \frac{1}{2} b_k + \frac{3}{2} b_{j_1} + \frac{1}{2} \max\{b_{kj_1}, 0\} + W_0 + W_1,\end{aligned}$$

which is equivalent to $b_{kj_1} > \max\{b_{kj_1}, 0\}$, so we reach a contradiction.

1b. If $x'_{j_2} = \frac{1}{2}$ and $x'_{kj_2} = 0$, then we have $J'(X) = \frac{1}{2}b_{j_2} + W_1$, where W_1 denotes the remaining part of the objective. With the additional constraint $x'_k = 0$, we can obtain a feasible solution $\tilde{X}'$ with $x'_{j_2} = 0, x'_{kj_2} = 0$, so $J'(\tilde{X}) = W_0$, where W_0 is the remaining part of the objective of $\tilde{X}'$. From the optimality of X', we can write

$$\frac{1}{2}b_{j_2} + W_1 > W_0.$$

As in 1a, we can see that W_0 is also the remaining part of the original objective, so $J(X) = \frac{1}{2}b_k + W_0$. Meanwhile, $\tilde{X}$ is also feasible for the original problem without the additional constraint, which is $x_k = \frac{1}{2}$, $x_{j_2} = \frac{1}{2}$, $x_{kj_2} = \frac{1}{2}$ (if $b_{kj_2} \geq 0$), and $x_{kj_2} = 0$ (if $b_{kj_2} < 0$), so $J(\tilde{X}) = \frac{1}{2}b_k + \frac{1}{2}b_{j_2} + \frac{1}{2}\max\{b_{kj_2}, 0\} + W_1$. Because X is an optimal solution of the original problem, we obtain

$$\frac{1}{2}b_k + W_0 \geq \frac{1}{2}b_k + \frac{1}{2}b_{j_2} + \frac{1}{2}\max\{b_{kj_2}, 0\} + W_1.$$

By summing up these two inequalities, we obtain

$$\frac{1}{2}b_k + \frac{1}{2}b_{j_2} + W_0 + W_1 > \frac{1}{2}b_k + \frac{1}{2}b_{j_2} + \frac{1}{2}\max\{b_{kj_2}, 0\} + W_0 + W_1,$$

which is equivalent to $0 > \max\{b_{kj_2}, 0\}$, so we reach a contradiction.

We next discuss the case of any other x_i that is integral in the original solution and not connected to x_k but connected to x_j. If x_j does not change in the new solution, then x_i will also remain the same. If $x_j = \frac{1}{2}$ in the original solution but changes to one or zero in the new solution, the arguments from the preceding analysis apply, and thus, x_i will also remain the same. □

Endnotes

[1] https://www.iriworldwide.com/IRI/media/IRI-Clients/price_promo_special_report_2015_final_22-Oct-15_V2.pdf.

[2] https://retailleader.com/brick-and-mortar-makes-grade-back-school-shopping.

[3] The words *demand* and *sales* are used interchangeably.

[4] For example, in Cohen et al. (2017), the authors estimate such models for products in the coffee, chocolate, tea, and yogurt categories. They obtain an out-of-sample R^2 of between 0.759 and 0.964.

[5] Note that the magnitude of the cross-item effects is measured relative to the parameters of the demand functions such as seasonality intercept, price sensitivity, and the factors capturing the postpromotion dip effect. We provide more extensive tests in Section 5.5.

[6] The ordinary-least-squares (OLS) problem used to estimate the model in (13) is efficient. However, the model in (12) cannot be efficiently estimated using OLS. A common method is to first estimate the additive cross-item effects for each pair of items and then normalize their effects in the sales data to finally estimate the log-log part of the model using the normalized data.

[7] A similar behavior was observed for the linear demand model. This observation is true for the vast majority of the tests presented in this section.

[8] The building time of *App*(2) is dominated by calls to two key Gurobi's functions: add_var and add_constr. In its current form, the code generates the coefficients and updates the model serially. This procedure could potentially be improved by directly creating a model file and adding some parallelization. Optimizing our code to build the models to reduce the run time and memory use is beyond the scope of this paper.

References

Ahn Hs, Gümüş M, Kaminsky P (2007) Pricing and manufacturing decisions when demand is a function of prices in multiple periods. *Oper. Res.* 55(6):1039–1057.

Ailawadi KL, Gedenk K, Lutzky C, Neslin SA (2007) Decomposition of the sales impact of promotion-induced stockpiling. *J. Marketing Res.* 44(3):450–467.

Baardman L, Cohen MC, Panchamgam K, Perakis G, Segev D (2018) Scheduling promotion vehicles to boost profits. *Management Sci.* 65(1):50–70.

Balinski ML (1970) On a selection problem. *Management Sci.* 17(3): 230–231.

Blattberg RC, Neslin SA (1990) *Sales Promotion: Concepts, Methods, and Strategies* (Prentice Hall, Upper Saddle River, NJ).

Campo K, Gijsbrechts E, Nisol P (2000) Toward understanding consumer response to stock-outs. *J. Retailing* 76(2):219–242.

Cohen MC, Perakis G (2020) Optimizing promotions for multiple items in supermarkets.*Channel Strategies and Marketing Mix in a Connected World* (Springer, New York).

Cohen MC, Lobel R, Perakis G (2018) Dynamic pricing through data sampling. *Production Oper. Management* 27(6):1074–1088.

Cohen MC, Leung NHZ, Panchamgam K, Perakis G, Smith A (2017) The impact of linear optimization on promotion planning.*Oper. Res.* 65(2):446–468.

Cooper LG, Baron P, Levy W, Swisher M, Gogos P (1999) Promocast: A new forecasting method for promotion planning. *Marketing Sci.* 18(3):301–316.

Corsten D, Gruen T (2004) Stock-outs cause walkouts. *Harvard Bus. Rev.* 82(5):26–28.

Felgate M, Fearne A (2015) Analyzing the impact of supermarket promotions: A case study using Tesco clubcard data in the UK. Dato-on M., ed. *The Sustainable Global Marketplace: Proceedings of the Academy of Marketing Science.* (Springer, New York), 471–475.

Foekens EW, Leeflang SH, Wittink DR (1998) Varying parameter models to accommodate dynamic promotion effects. *J. Econometrics* 89(1):249–268.

Frank M, Wolfe P (1956) An algorithm for quadratic programming. *Naval Res. Logistics* 3(1–2):95–110.

Grossmann IE (2002) Review of nonlinear mixed-integer and disjunctive programming techniques. *Optimization Engrg.* 3(3): 227–252.

Hemmecke R, Köppe M, Lee J, Weismantel R (2010) Nonlinear integer programming. Jünger, M, Liebling, ThM, Naddef, D, Nemhauser, GL, Pulleyblank, WR, Reinelt, G, Rinaldi, G, Wolsey, LA, eds. *50 Years of Integer Programming 1958–2008* (Springer, Berlin), 561–618.

Hess JD, Gerstner E (1987) Loss leader pricing and rain check policy. *Marketing Sci.* 6(4):358–374.

Levin Y, McGill J, Nediak M (2010) Optimal dynamic pricing of perishable items by a monopolist facing strategic consumers. *Production Oper. Management* 19(1):40–60.

Macé S, Neslin SA (2004) The determinants of pre- and postpromotion dips in sales of frequently purchased goods. *J. Marketing Res.* 41(3):339–350.

Möhring RH, Schulz AS, Stork F, Uetz M (2003) Solving project scheduling problems by minimum cut computations. *Management Sci.* 49(3):330–350.

Nielsen (2015) The path to efficient trade promotions. Accessed April, 2020, http://www.nielsen.com/us/en/insights/reports/2015/the-path-to-efficient-trade-promotions.html.

Nocedal J, Wright SJ (2006) *Sequential Quadratic Programming* (Springer, New York).

Özer Ö, Phillips R (2012) *The Oxford Handbook of Pricing Management* (Oxford University Press, Oxford, UK).

Padberg M (1989) The Boolean quadric polytope: Some characteristics, facets and relatives. *Math. Programming* 45(1–3): 139–172.

Pindyck RS, Rubinfeld DL (2005) *Microeconomics* (Prentice-Hall, Upper Saddle River, NJ).

Rhys J (1970) A selection problem of shared fixed costs and network flows. *Management Sci.* 17(3):200–207.

Su X (2010) Intertemporal pricing and consumer stockpiling. *Oper. Res.* 58(4 pt. 2):1133–1147.

Talluri KT, Van Ryzin GJ (2006) *The Theory and Practice of Revenue Management*, vol. 68 (Springer Science & Business Media, New York).

Van Donselaar K, Van Woensel T, Broekmeulen R, Fransoo J (2006) Inventory control of perishables in supermarkets. *Internat. J. Production Econom.* 104(2):462–472.

Van Heerde HJ, Gupta S, Wittink DR (2003) Is 75% of the sales promotion bump due to brand switching? no, only 33% is. *J. Marketing Res.* 40(4):481–491.