



BCOL RESEARCH REPORT 15.06

Industrial Engineering & Operations Research
University of California, Berkeley, CA 94720-1777

A CONIC INTEGER PROGRAMMING APPROACH TO CONSTRAINED ASSORTMENT OPTIMIZATION UNDER THE MIXED MULTINOMIAL LOGIT MODEL

ALPER ŞEN, ALPER ATAMTÜRK AND PHILIP KAMINSKY

ABSTRACT. We consider the constrained assortment optimization problem under the mixed multinomial logit model. Even moderately sized instances of this problem are challenging to solve directly using standard mixed-integer linear optimization formulations. This has motivated recent research exploring customized optimization strategies and approximation techniques. In contrast, we develop a novel conic quadratic mixed-integer formulation. This new formulation, together with McCormick inequalities exploiting the capacity constraints, enables the solution of large instances using commercial optimization software.

Keywords: Assortment optimization, mixed multinomial logit, conic integer programming

October 2015; October 2016; May 2017

1. INTRODUCTION

Assortment planning, the selection of products that a firm offers to its customers, is a key problem faced by retailers, with direct impact on profitability, market share, and customer satisfaction. A growing stream of operations research literature focuses on assortment optimization problems, where the assortment is optimized to maximize revenue (see Kök et al., 2009, for a review). To solve this category of problems, customers' purchase behavior must be modeled in a way that captures the impact on the overall demand of product characteristics and customers' substitution between products. The most commonly used model for customer behavior in this setting is the the multinomial logit (MNL) model, which is based on a probabilistic model of individual customer utilities (see the pioneering work of van Ryzin and Mahajan (1999), and followup work by Cachon et al. (2005), Mahajan and van Ryzin (2001), Chong et al. (2001), Li (2007), Rusmevichientong et al. (2010),

A. Şen: Department of Industrial Engineering, Bilkent University Bilkent, Ankara, 06800, Turkey
alpersen@bilkent.edu.tr

A. Atamtürk: Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720-1777 USA atamturk@berkeley.edu

P. Kaminsky: Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720-1777 USA kaminsky@berkeley.edu.

Rusmevichientong and Topaloglu (2012), and Topaloglu (2013)). Despite its popularity, the MNL model has several key shortcomings: 1) It relies on the so-called Independence of Irrelevant Alternatives (IIA) assumption, so that a product’s market share relative to another product is constant regardless of the other products in the assortment; 2) the total market share of an assortment and the substitution rates within that assortment cannot be independently defined (Kök and Fisher, 2007). A partial remedy for these problems is possible under an extension of the MNL model, called the nested logit model. Recent work that studies assortment optimization under variants of the nested logit model includes Davis et al. (2014), Gallego and Topaloglu (2014) and Li et al. (2015).

In this paper, we consider assortment optimization under a generalization of the MNL model that does not have either of these limitations, the Mixed MNL model (MMNL). The MMNL model, introduced by Boyd and Mellman (1980) and Cardell and Dunbar (1980), has another important characteristic – as observed by McFadden and Train (2000): “any discrete choice model derived from random utility maximization... can be approximated as closely as one pleases by a MMNL model.” Assortment planning under the MMNL model, also known as the mixtures of MNL model (Feldman and Topaloglu, 2015), MNL with random choice parameters (Rusmevichientong et al., 2014), and latent-class MNL (Méndez-Díaz et al., 2014), has received considerable interest in the OR/MS community. The problem also arises as a subproblem in a new approach in revenue management called choice-based deterministic linear optimization that attempts to model customer choice behavior more realistically (Liu and van Ryzin, 2008).

We are particularly interested in assortment optimization under MMNL with constraints on the number of products in the assortment (so-called “capacity constraints”). While optimal assortments under MNL can be efficiently found (Rusmevichientong et al., 2010), this does not hold true for assortment optimization under MMNL, in either the capacitated or uncapacitated settings. Indeed, Bront et al. (2009) and Rusmevichientong et al. (2014) show that the assortment optimization problem under the mixed MNL model is NP-hard. Motivated by the computational complexity and the ineffectiveness of standard MILP formulations of the problem, Bront et al. (2009) propose a greedy heuristic. Méndez-Díaz et al. (2014) design and test a branch-and-cut algorithm that generates good but often not provably optimal solutions for both capacitated and uncapacitated versions. Rusmevichientong et al. (2014) identify special cases of the (uncapacitated) problem that are polynomially solvable, and characterize the performance of heuristics for other cases. Feldman and Topaloglu (2015) develop strong upper bounds on the optimal objective value.

In contrast, we show that by formulating this problem in a non-traditional manner, as a conic quadratic mixed-integer program, large instances of the capacitated version of the problem can be solved directly using commercial mathematical optimization software, obviating the need to develop customized heuristics or optimization software to solve the problem. The advantages of this approach are clear: commercial software is continually developed to take advantages of advances in optimization methods and hardware, it is supported by large software firms, and it allows the inclusion of new constraints without the need for reprogramming. We also show how to further strengthen the formulation with McCormick estimators derived through conditional bounds exploiting the capacity constraints.

2. BACKGROUND

In this section we present a short overview of the mixed multinomial logic model and conic integer optimization.

2.1. The Consumer Choice Model. First, recall the traditional MNL model. Let N be the set of products in the category indexed by j . Let S be the assortment – the subset of products offered by the retailer. Let ρ_j be the unit price for product j . The MNL model is based on the utility that a customer gets from consuming a product. For any product, this utility has two components $U_j = u_j + \epsilon_j$, where u_j is a deterministic component and ϵ_j is a random component which is assumed to be a Gumbel random variable with mean zero and variance $\mu^2\pi^2/6$. Given these, the probability that a customer purchases product j from a given assortment S is $p_j(S) = \nu_j/(\nu_0 + \sum_{k \in S} \nu_k)$, where $\nu_j = e^{(u_j - \rho_j)/\mu}$ and ν_0 corresponds to the no-purchase option.

As discussed above, we utilize the mixed MNL (MMNL) model. This model extends the MNL model by introducing a set M of customer classes. Let γ_i be the probability that the demand originates from customer class i . The demand in each customer class is governed by a separate MNL model. Let ν_{ij} be the customer preference for product j in class i and ν_{i0} be the no-purchase preference in class i . Let the unit revenue from product j in class i be ρ_{ij} . We can then write the expected revenue for a given assortment S as

$$\sum_{i \in M} \gamma_i \left[\frac{\sum_{j \in S} \rho_{ij} \nu_{ij}}{\nu_{i0} + \sum_{j \in S} \nu_{ij}} \right]. \tag{1}$$

Due to space or administrative restrictions, there can be various constraints on the depth of the assortment that can be carried. Let K be the set of resources that may constrain the assortment. Let β_{kj} denote the amount of resource k used by product j and let κ_k denote the amount of resource k available. The capacitated assortment optimization problem is therefore to select the assortment S in this setting.

2.2. Conic Integer Optimization. Conic optimization refers to optimization of a linear function over conic inequalities (Ben-Tal and Nemirovski, 2001). A conic quadratic constraint on $x \in \mathbf{R}^n$ is a constraint of the form

$$\|Ax - b\| \leq c'x - d.$$

Here $\|\cdot\|$ is the L2 norm, A is an $m \times n$ -matrix, b is an m -column vector, c is an n -column vector, and d is a scalar. We refer the reader to Lobo et al. (1998) and Alizadeh and Goldfarb (2003) for reviews of conic quadratic optimization and its applications.

Although there is an extensive body of literature on convex conic quadratic optimization, development of conic optimization with integer variables is quite recent (Çezik and Iyengar, 2005; Atamtürk and Narayanan, 2010, 2011; Atamtürk et al., 2013). With the growing availability of commercial solvers for these problems (for example, both CPLEX and Gurobi now include solvers for these models), conic quadratic integer models have recently been employed to address problems in portfolio optimization (Vielma et al., 2008), value-at-risk minimization (Atamtürk and Narayanan, 2008), machine scheduling (Aktürk et al., 2010), and supply chain network design (Atamtürk et al., 2012), airline rescheduling with speed control (Aktürk et al., 2014). However, to the best of our knowledge, this approach has not been previously used to solve assortment optimization problems.

Conic quadratic inequalities are often used to represent a rotated cone/hyperbolic inequality

$$x_1^2 \leq x_2 x_3, \quad (2)$$

on $x_1, x_2, x_3 \geq 0$. It is easily verified that hyperbolic inequality (2) can then be equivalently written as a conic quadratic inequality

$$\|(2x_1, x_2 - x_3)\| \leq x_2 + x_3. \quad (3)$$

In our conic reformulation of the assortment optimization problem, we make use of the rotated cone inequalities (2) in our models.

3. THE CAPACITATED ASSORTMENT OPTIMIZATION PROBLEM

In this section we first recall the traditional MILP formulation of the capacitated assortment optimization problem, and then present an alternative conic quadratic mixed 0-1 formulation of the problem, and strengthen the formulation using McCormick estimators based on conditional bounds.

3.1. The Traditional MILP Formulation. Given the MMNL demand model, define x_j to be 1 if product j is offered in the assortment and 0 otherwise. We can then state the capacitated assortment optimization problem as a nonlinear binary optimization:

$$\max \sum_{i \in M} \gamma_i \left[\frac{\sum_{j \in N} \rho_{ij} \nu_{ij} x_j}{\nu_{i0} + \sum_{j \in N} \nu_{ij} x_j} \right] \quad (4)$$

$$\text{(CAOP)} \quad \text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad \forall k \in K \quad (5)$$

$$x_j \in \{0, 1\}, \quad \forall j \in N. \quad (6)$$

Traditionally, (CAOP) is formulated as a Mixed Integer Linear Program (see, e.g. Bront et al., 2009; Méndez-Díaz et al., 2014). First, letting $y_i = 1/(\nu_{i0} + \sum_{j \in N} \nu_{ij} x_j)$, the problem can be posed as a bilinear mixed 0-1 optimization problem:

$$\max \sum_{i \in M} \sum_{j \in N} \gamma_i \rho_{ij} \nu_{ij} y_i x_j \quad (7)$$

$$\text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad \forall k \in K \quad (8)$$

$$\text{(CAOP')} \quad \nu_{i0} y_i + \sum_{j \in N} \nu_{ij} y_i x_j = 1, \quad \forall i \in M \quad (9)$$

$$y_i \geq 0, \quad \forall i \in M \quad (10)$$

$$x_j \in \{0, 1\}, \quad \forall j \in N. \quad (11)$$

The bilinear terms $y_i x_j$ in the formulation can be linearized using the standard “big- M ” approach: For any bilinear term yx , where y is continuous and non-negative and x is binary, define a new continuous variable $z = yx$ and add the following inequalities to the formulation: $y - z \leq U(1 - x)$, $0 \leq z \leq y$ and $z \leq Ux$, where U is a sufficiently *large upper bound* on y . Employing this technique, and selecting $1/\nu_{i0}$ for U , leads to the following mixed-integer

linear formulation:

$$\max \sum_{i \in M} \sum_{j \in N} \gamma_i \rho_{ij} \nu_{ij} z_{ij} \quad (12)$$

$$\text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad \forall k \in K \quad (13)$$

$$\nu_{i0} y_i + \sum_{j \in N} \nu_{ij} z_{ij} = 1, \quad \forall i \in M \quad (14)$$

$$\text{(MILP)} \quad \nu_{i0} (y_i - z_{ij}) \leq 1 - x_j, \quad \forall i \in M, \quad \forall j \in N \quad (15)$$

$$0 \leq z_{ij} \leq y_i, \quad \forall i \in M, \quad \forall j \in N, \quad (16)$$

$$\nu_{i0} z_{ij} \leq x_j, \quad \forall i \in M, \quad \forall j \in N \quad (17)$$

$$x_j \in \{0, 1\}, \quad \forall j \in N \quad (18)$$

$$z_{ij} \geq 0, \quad i \in M, \quad j \in N \quad (19)$$

$$y_i \geq 0, \quad i \in M. \quad (20)$$

As shown in Bront et al. (2009); Méndez-Díaz et al. (2014); Feldman and Topaloglu (2015), formulation (MILP) does not scale well. In particular, when the capacity constraints (13) are tight, solution times are prohibitive even for moderately sized instances.

3.2. The Conic Formulation. In order to give a conic reformulation, we first restate the objective as minimization. Letting $\bar{\rho}_i = \max_{j \in N} \rho_{ij}$, the objective (4) of (CAOP) can be written as

$$\max \sum_{i \in M} \gamma_i \bar{\rho}_i - \sum_{i \in M} \gamma_i \left[\frac{\nu_{i0} \bar{\rho}_i + \sum_{j \in N} \nu_{ij} (\bar{\rho}_i - \rho_{ij}) x_j}{\nu_{i0} + \sum_{j \in N} \nu_{ij} x_j} \right]. \quad (21)$$

As the first component in (21) is constant, we can pose the problem as minimizing the second component in (21). Also, since the objective coefficients are nonnegative, it suffices to use only lower bounds on y and z variables, leading to:

$$\min \sum_{i \in M} \gamma_i \nu_{i0} \bar{\rho}_i y_i + \sum_{i \in M} \sum_{j \in N} \gamma_i \nu_{ij} (\bar{\rho}_i - \rho_{ij}) z_{ij} \quad (22)$$

$$\text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad k \in K \quad (23)$$

$$\text{(CAOP'')} \quad z_{ij} \geq y_i x_j, \quad i \in M, \quad j \in N \quad (24)$$

$$y_i \geq \frac{1}{\nu_{i0} + \sum_{j \in N} \nu_{ij} x_j}, \quad i \in M \quad (25)$$

$$x_j \in \{0, 1\}, \quad j \in N \quad (26)$$

$$z_{ij} \geq 0, \quad i \in M, \quad j \in N \quad (27)$$

$$y_i \geq 0, \quad i \in M. \quad (28)$$

Observe that constraints (24)–(25) are satisfied at equality at an optimal solution. Now, defining

$$w_i = \nu_{i0} + \sum_{j \in N} \nu_{ij} x_j \quad (29)$$

and observing that $w \geq 0$, one can state constraints (25) in rotated cone form:

$$y_i w_i \geq 1. \quad (30)$$

As constraint (30) is satisfied at equality at an optimal solution, $w \geq 0$, and $x_j = x_j^2$ for a binary vector x , constraint (24) can also be stated in rotated cone form:

$$z_{ij}w_i \geq x_j^2 \quad (31)$$

Although redundant for the mixed-integer formulation, we also use the constraints

$$\nu_{i0}y_i + \sum_{j \in N} \nu_{ij}z_{ij} \geq 1, \quad \forall i \in M \quad (32)$$

to strengthen the continuous relaxation of the formulation.

The final conic quadratic mixed 0-1 program is, therefore:

$$\min \sum_{i \in M} \gamma_i \nu_{i0} \bar{\rho}_i y_i + \sum_{i \in M} \sum_{j \in N} \gamma_i \nu_{ij} (\bar{\rho}_i - \rho_{ij}) z_{ij} \quad (33)$$

$$\text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad k \in K \quad (34)$$

$$w_i = \nu_{i0} + \sum_{j \in N} \nu_{ij} x_j, \quad i \in M \quad (35)$$

$$\text{(CONIC)} \quad z_{ij} w_i \geq x_j^2, \quad i \in M \quad j \in N \quad (36)$$

$$y_i w_i \geq 1, \quad i \in M \quad (37)$$

$$\nu_{i0} y_i + \sum_{j \in N} \nu_{ij} z_{ij} \geq 1, \quad \forall i \in M \quad (38)$$

$$x_j \in \{0, 1\}, \quad j \in N \quad (39)$$

$$z_{ij} \geq 0, \quad i \in M, \quad j \in N \quad (40)$$

$$y_i \geq 0, \quad i \in M. \quad (41)$$

In contrast to the traditional formulation (MILP), the conic formulation does not require “big- M ” constants for linearization, which lead to weak LP relaxations especially for the tightly capacitated cases. On the other hand, for the conic formulation when capacity is low, small values of w_i tighten the constraints $z_{ij}w_i \geq x_j^2$, leading to stronger bounds. The next proposition provides a theoretical justification for adding inequalities (32) to the formulation. Preliminary computations also showed a significant strengthening of the conic formulation with the addition of inequalities (32).

Proposition 1. *Inequality (32) is a facet-defining for $\text{cl conv}\{(x, y, z) \in \{0, 1\}^N \times \mathbb{R}^M \times \mathbb{R}^{M \times N} : (24) - (28)\}$.*

Proof. Let $S = \{(x, y, z) \in \{0, 1\}^N \times \mathbb{R}^M \times \mathbb{R}^{M \times N} : (24) - (28)\}$. First, observe that even though constraints (25) are nonlinear, S is a union of polyhedra (one polyhedron for each assignment of the binary variables); hence, $\text{cl conv}(S)$ is a polyhedron. Let e_k be the k th unit vector, $\hat{y} = \sum_{k \in M} e_k / \nu_{k0}$ and $\hat{y}^i = \sum_{k \in M \setminus \{i\}} e_k / \nu_{k0}$. Consider the following $|N| + |M| + |M||N|$ points in S satisfying $\nu_{i0}y_i + \sum_{j \in N} \nu_{ij}z_{ij} = 1$: $(0, \hat{y}, 0)$; $(0, \hat{y} + \epsilon e_k, 0)$, $k \in M \setminus \{i\}$, $\epsilon > 0$; $(0, \hat{y}, \epsilon e_{kj})$, $k \in M \setminus \{i\}$, $j \in N$, $0 < \epsilon < 1$; $(e_j, \hat{y}^i + e_i / (\nu_{i0} + \nu_{ij}), e_{ij} / (\nu_{i0} + \nu_{ij}))$, $j \in N$; $(e_j, \hat{y}^i + (1 - \epsilon)e_i / (\nu_{i0} + \nu_{ij}), (1 + (\nu_{i0}\epsilon / \nu_{ij}))e_{ij} / (\nu_{i0} + \nu_{ij}))$, $j \in N$, $0 < \epsilon < 1$, where e_{ij} is the ij th unit vector. It is easily checked that these points are affinely independent.

3.3. McCormick Estimators. The capacitated assortment formulations can be further strengthened using McCormick estimators for the bilinear terms. To that end, we give simple upper and lower bounds on

$$y_i = \frac{1}{\nu_{i0} + \sum_{j \in N} \nu_{ij} x_j}, \quad i \in M. \quad (42)$$

The lower bounds make use of the capacity constraints (13). For $i \in M$, define the auxiliary problem

$$f_i = \max \sum_{j \in N} \nu_{ij} x_j \quad (43)$$

$$\text{(BND)} \quad \text{s.t.} \quad \sum_{j \in N} \beta_{kj} x_j \leq \kappa_k, \quad k \in K \quad (44)$$

$$x_j \in \{0, 1\}, \quad j \in N. \quad (45)$$

Proposition 2. *The following bounds on variables y_i , $i \in M$, are valid:*

$$y_i^\ell := \frac{1}{\nu_{i0} + f_i} \leq y_i \quad (46)$$

$$y_i^u := \frac{1}{\nu_{i0}} \geq y_i. \quad (47)$$

Proposition 2 provides global bounds on variables y . Next, we give conditional bounds. Let $f_{i|x_j=\xi}$ be the objective function value of (BND) when an additional constraint $x_j = \xi$, $j \in N$ is imposed.

Proposition 3. *For $j \in N$, the following conditional bounds on variables y_i , $i \in M$, are valid:*

$$x_j = 0 \Rightarrow y_{i|x_j=0}^\ell := \frac{1}{\nu_{i0} + f_{i|x_j=0}} \leq y_i \quad (48)$$

$$x_j = 1 \Rightarrow \begin{cases} y_{i|x_j=1}^\ell := \frac{1}{\nu_{i0} + f_{i|x_j=1}} \leq y_i \\ y_{i|x_j=1}^u := \frac{1}{\nu_{i0} + \nu_{ij}} \geq y_i. \end{cases} \quad (49)$$

(BND) is a binary multiple constraint knapsack problem, so it may be prohibitive to find the optimal f_i and $f_{i|x_j=\xi}$ except in special cases. However, note that to get a lower bound on y_i , an upper bound on the optimal value of (BND) is sufficient, and this can be found by solving an easier relaxation of the problem, e.g., the linear optimization relaxation.

For the special case of a single cardinality constraint, one can obtain exact closed form lower bounds on y .

Proposition 4. *For a single cardinality constraint of the form $\sum_{j \in N} x_j \leq \kappa$, the following global and conditional lower bounds on y_i , $i \in M$, are valid:*

$$y_i^\ell := \frac{1}{\nu_{i0} + \sum_{k=1}^{\kappa} \nu_{i[k]}} \quad (50)$$

$$y_{i|x_j=0}^\ell := \frac{1}{\nu_{i0} + \sum_{k=1}^{\kappa} \bar{\nu}_{i[k]}} \quad (51)$$

$$y_{i|x_j=1}^\ell := \frac{1}{\nu_{i0} + \nu_{ij} + \sum_{k=1}^{\kappa-1} \bar{\nu}_{i[k]}} \quad (52)$$

where $\nu_{i[k]}$ is defined as the k th largest of preferences ν_{im} , $m \in N$ and $\bar{\nu}_{i[k]}$ is defined as the k th largest of preferences ν_{im} , $m \in N \setminus \{j\}$.

Similar exact closed-form bounds can be developed when there are multiple non-overlapping cardinality constraints (i.e., the assortment can contain at most a fixed number of products from each product sub-group).

Using the global and conditional bounds on y_i , $i \in M$, above, one can write the following valid McCormick inequalities (McCormick, 1976) for each bilinear term $z_{ij} = y_i x_j$:

$$z_{ij} \leq y_{i|x_j=1}^u x_j, \quad i \in M, j \in N \quad (53)$$

$$(MC) \quad z_{ij} \geq y_{i|x_j=1}^l x_j, \quad i \in M, j \in N \quad (54)$$

$$z_{ij} \leq y_i - y_{i|x_j=0}^l (1 - x_j), \quad i \in M, j \in N \quad (55)$$

$$z_{ij} \geq y_i - y_i^u (1 - x_j), \quad i \in M, j \in N. \quad (56)$$

Note that the inequality (53) is also used in Méndez-Díaz et al. (2014) and that (56) is the same as (15) in model (MILP).

Based on the discussion thus far, four different formulations can be used to solve the capacitated assortment optimization problem under MMNL. The first one is (MILP), which can be strengthened by replacing constraints (15)-(17) with the stronger McCormick estimators (MC). We denote this strengthened formulation (MILP+MC). The third formulation is (CONIC), which can also be strengthened by adding McCormick inequalities (MC). This fourth formulation is denoted (CONIC+MC). Notice that one can convert (MILP) and (MILP+MC) to minimization problems by using the equivalent objective (33). This leads to the observation that (CONIC+MC) is a strengthening of (MILP+MC) with constraints (35), (36), and (37). Therefore, (CONIC+MC) is stronger than (MILP+MC), which is itself stronger than (MILP). The numerical experiments reported in the next section show the significance and the effect of differences in the strength of these formulations.

4. NUMERICAL STUDY

In order to test the effectiveness of the conic optimization approach and the McCormick inequalities, we perform a numerical study on four sets of problems. The optimization problems are solved with Gurobi 6.5.1 solver on a computer with an Intel Core i7-4510U 2.00 GHz (2.60 GHz Turbo) processor and 8 GB RAM operating on 64-bit Windows 10. We use the default settings of Gurobi except that we force the solver to use the linear outer-approximation approach when solving continuous relaxations of conic programs. The outer-approximation allows warm starts with the dual simplex method and speeds up solving node relaxations. The time limit is set to 600 seconds.

The first set of problems are created by randomly generating instances with $|N| = 200$ products and $|M| = 20$ customer classes. The product prices are the same across the customer classes ($\rho_{ij} = \rho_j$) and are drawn from a uniform[1,3] distribution. The preferences ρ_{ij} are drawn from a uniform[0,1] distribution. The parameter $\gamma_i = 1/20$ for all $i \in M$. The no purchase parameter $\nu_{i0} = \nu_0$ is either 5 or 10. The capacity constraint is in the form of a cardinality constraint. The maximum cardinality κ of the assortment is one of five possible values: $\{10, 20, 50, 100, 200\}$. For each of these $5 \times 2 = 10$ capacity and no-purchase probability combinations, we generate five instances, resulting in a total of 50 instances. All data files are publicly available at <http://ieor.berkeley.edu/~atamturk/data>.

We test the effectiveness of four formulations: (MILP), (MILP+MC), (CONIC) and (CONIC+MC). In addition, we compare these with the formulation of Méndez-Díaz et al. (2014), which strengthen (MILP) by replacing (17) with (53) and by introducing five classes of valid inequalities. Three of these are polynomial in the size of the model, while the rest are exponential. We run their formulation using the three classes of polynomial valid inequalities.

TABLE 1. Results for problems with 200 products and 20 classes.

ν_0	κ	assort bind	MILP		MILP+MC		Méndez-Díaz		CONIC		CONIC+MC	
			rgap	time/#	rgap	time/#	rgap	time/#	rgap	time/#	rgap	time/#
			egap	nodes	egap	nodes	egap	nodes	egap	nodes	egap	nodes
5	10	10.0	52.56%	–	12.33%	–	51.46%	–	3.20%	32.82/5	0.27%	8.72/5
		5	45.10%	3076	9.85%	6374	50.16%	0	0.00%	1449	0.00%	14
	20	20.0	33.38%	–	10.25%	–	33.37%	–	5.88%	122.74/4	0.36%	9.58/5
		5	32.07%	11626	8.34%	13819	33.36%	44.8	0.10%	2851	0.00%	23
	50	50.0	2.81%	481.16/2	0.94%	27.73/3	2.79%	–	17.14%	–	0.02%	2.38/5
		5	1.72%	87695	0.09%	27779	2.78%	102.6	2.94%	1566	0.00%	0
100	65.4	0.08%	4.26/5	0.03%	1.22/5	0.07%	366.16/5	23.66%	–	0.01%	1.82/5	
	0	0.00%	790	0.00%	0	0.00%	124	7.23%	768	0.00%	0	
200	65.4	0.08%	2.29/5	0.04%	1.06/5	0.07%	366.57/5	23.66%	–	0.01%	1.92/5	
	0	0.00%	343	0.00%	0	0.00%	117.6	13.12%	747	0.00%	0	
10	10	10.0	24.74%	–	7.20%	–	20.69%	–	1.93%	22.50/5	0.10%	6.47/5
		5	10.26%	47690	5.44%	6555	19.70%	0.2	0.00%	1054	0.00%	4
	20	20.0	38.66%	–	8.47%	–	38.65%	–	3.61%	86.77/5	0.16%	8.62/5
		5	31.57%	1613	7.20%	9498	38.61%	4.2	0.00%	1374	0.00%	7
	50	50.0	10.50%	–	2.92%	–	10.50%	–	10.31%	–	0.08%	7.37/5
		5	9.89%	25276	2.02%	31281	10.49%	48.6	1.30%	1454	0.00%	72
100	91.8	0.04%	3.46/5	0.01%	1.20/5	0.03%	306.05/5	18.40%	–	0.00%	1.77/5	
	1	0.00%	406	0.00%	0	0.00%	255.8	4.62%	766	0.00%	0	
200	92.0	0.04%	2.89/5	0.01%	0.93/5	0.03%	282.31/5	18.41%	–	0.00%	1.67/5	
	0	0.00%	462	0.00%	0	0.00%	82.4	5.86%	768	0.00%	0	
Average			16.29%	46.67/22	4.22%	4.58/23	15.76%	330.27/20	12.62%	63.23/19	0.10%	5.03/50
			13.06%	17898	3.30%	9531	15.51%	78.02	3.52%	1280	0.00%	12

Table 1 presents averages of root gap, end gap, solution time and the number of search nodes over five instances for each no purchase preference ν_0 , capacity level κ and formulation. The number of products in the assortment ($\sum_{j \in N} x_j^*$, averaged over five instances) and the number of instances where the capacity is binding in the optimal solution are given by assort and bind, respectively. The root gap is computed as $\text{rgap} = 100 \times (\text{zopt} - \text{zroot}) / |\text{zopt}|$, where zroot is the objective value of the continuous relaxation (before presolve and root cuts) and zopt is the value of the optimal integer solution. The end gap is computed as $\text{egap} = 100 \times (\text{zopt} - \text{zbb}) / |\text{zopt}|$, where zbb is the best lower bound at termination. If an instance is solved to optimality zbb equals zopt (within the default optimality gap 0.01). In the tables, time refers to the average solution time (in seconds) for the instances that are solved within the time limit and # refers to the number of instances solved within the time limit. The last row reports the averages for rgap , egap , time and nodes and the total number of instances solved.

As observed in previous studies, the traditional (MILP) formulation performs poorly, except when the capacity constraint is loose. The time limit is reached for 28 instances with

tight capacity constraints. The poor performance appears to be due to the weak relaxation, leading to excessive branching. The remaining gaps at termination are quite large for the unsolved instances. With the addition of McCormick inequalities (MC), root and end gaps improve substantially in all cases. The average root gap drops from 16.29% to 4.22%. However, this is still not enough to solve the capacitated cases. McCormick inequalities help to solve only one additional instance within the time limit.

For our data set, the polynomial inequalities of Méndez-Díaz et al. (2014) lead to a small reduction in root gaps compared to (MILP). Cutting plane algorithms implementing separation for the exponential classes of inequalities of Méndez-Díaz et al. (2014) may lead to a further reduction. Although we use a different data set, consistent with their numerical study, the Méndez-Díaz formulation is more effective for high capacity instances. Model (MILP+MC) is considerably stronger. The strength of (MILP+MC) over Méndez-Díaz is due to conditional McCormick inequalities (54) and (55) based on strong lower bounds on y .

In contrast to the linear formulations, most of the capacitated instances are solved easily with the conic formulation. This is due to small root gaps, leading to only limited enumeration. However, the performance of the conic formulation degrades for high capacity instances. Observe that MILP and CONIC formulations are not directly comparable. The CONIC formulation may be weaker than the MILP formulation for high capacity instances; whereas the MILP formulation tends to be weaker than the CONIC formulation for low capacity instances.

The results are dramatically better when the McCormick inequalities are added to the conic formulation. The average root gap drops to a mere 0.10% and *all* instances are solved to optimality, on average, in five seconds. On average, only 12 nodes are needed in the search tree. For some instances, the CONIC+MC is more than 100 times faster than the other approaches. This is due to the joint effect of the tightening of the formulation using conic constraints and McCormick inequalities as observed with very small root gaps for all instances. As noted in Section 3.3 CONIC+MC dominates MILP+MC.

In Table 2, we report the results of experiments for instances with 500 products and 50 classes. The preference values and prices are generated as before. Each class again has equal weight ($\gamma_i = 1/50$). The capacity κ is one of $\{20, 50, 100, 200, 500\}$, and the no purchase parameter ν_{i0} is either 10 or 20. Since our experiments do not indicate a significant improvement from employing the approach in Méndez-Díaz et al. (2014) over (MILP), we do not include it for the remaining experiments. We also note that five instances cannot be solved using any of the formulations within the time limit. For those instances, the optimal integer solutions are obtained separately using CONIC+MC formulation by extending the time limit. Therefore, root gap and end gap calculations are still with respect to the optimal integer solutions.

For the large instances, with the traditional (MILP) formulation the time limit is reached for 32 problem instances with tight capacity constraints. Although the addition of McCormick inequalities reduces the integrality gaps substantially, only two more instances can be solved within time limit. The root gaps for the conic formulation are much smaller for the capacitated cases; nevertheless, problems cannot be solved to optimality within the time limit for these large instances. Adding the McCormick inequalities to the conic formulation reduces the average root gap to 0.06% and allows the problems to be solved quickly. Many instances do not even require any branching, and 45 out of 50 instances are solved within

TABLE 2. Results for problems with 500 products and 50 classes.

ν_0	κ	assort bind	MILP		MILP+MC		CONIC		CONIC+MC		
			rgap egap	time/# nodes	rgap egap	time/# nodes	rgap egap	time/# nodes	rgap egap	time/# nodes	
10	20	20.0	58.05%	–	15.32%	–	2.28%	–	0.18%	282.58/4	
		5	57.54%	188	14.73%	114	0.37%	1239	0.02%	260	
	50	50.0	32.14%	–	11.14%	–	5.56%	–	0.11%	188.05/5	
		5	32.07%	1235	11.05%	546	2.61%	1261	0.00%	115	
	100	100.0	6.47%	–	2.37%	–	14.47%	–	0.03%	44.06/5	
		5	6.43%	2022	2.18%	3120	29.46%	1371	0.00%	6	
200	149.4	0.03%	30.49/5	0.01%	8.41/5	24.11%	–	0.00%	16.60/5		
	5	0.00%	650	0.00%	0	57.84%	417	0.00%	0		
500	149.4	0.03%	38.30/5	0.02%	13.04/5	24.11%	–	0.00%	18.30/5		
	5	0.00%	756	0.00%	10	55.57%	64	0.00%	0		
20	20	20.0	24.48%	–	9.57%	–	1.35%	–	0.04%	165.95/5	
		5	20.95%	1109	9.41%	127	0.10%	1539	0.00%	1	
	50	50.0	38.44%	–	10.42%	–	3.39%	–	0.14%	487.23/2	
		5	38.44%	840	10.37%	322	0.75%	1330	0.03%	421	
	100	100.0	15.32%	–	4.78%	–	8.54%	–	0.06%	232.29/4	
		5	15.30%	1557	4.72%	923	18.29%	1430	0.01%	276	
	200	197.8	0.07%	62.71/3	0.02%	40.56/5	18.90%	–	0.00%	16.77/5	
		3	0.06%	7039	0.00%	377	41.58%	173	0.00%	0	
	500	203.4	0.02%	15.84/5	0.01%	9.31/5	19.90%	–	0.00%	18.37/5	
		0	0.00%	423	0.00%	6	40.43%	47	0.00%	0	
	Average			17.51%	33.96/18	5.36%	17.83/20	12.26%	–	0.06%	119.43/45
				17.08%	1582	5.25%	555	24.70%	887	0.01%	108

the time limit. For the three instances that cannot be solved within the time limit, the end gap is only 0.04% on average.

A third set of problems is inspired by the work of Désir and Goyal (2014), who suggest a procedure to construct a family of hard benchmark instances to formally show that the MMNL assortment optimization problem is hard to approximate within any reasonable factor. Each MMNL instance is generated based on an undirected graph $G = (V, E)$. Each vertex in V corresponds to a product as well as a customer class ($V = M = N$). We denote by $C_i = \{j \mid (i, j) \in E\}$ the set of products that the customers in class i consider buying (this always includes product i and can be thought of as class i 's “consideration set”). Given this structure, we create a problem set with 100 products (and 100 classes). Each product has 10 neighbors in G so $|C_i| = 11$. These neighbors are selected randomly. However, this procedure may lead to unrealistic preference and price parameters; therefore, we use the following modification. We denote product i as class i 's favorite product and set $\nu_{ii} = 1$. For $i \neq j, (i, j) \in E$, ν_{ij} is drawn from a uniform $[0, 1]$ distribution. For $(i, j) \notin E$, $\nu_{ij} = 0$. The prices are randomly generated from a uniform $[1, 3]$ distribution. $\gamma_i, i \in M$ are drawn from a uniform $[0, 1]$ distribution. The capacity κ is one of $\{10, 20, 50, 100\}$, the no-purchase parameter ν_{i0} is either 1 or 2, and we again generate five instances for each parameter setting, leading to 40 instances. The results are reported in Table 3.

These instances are indeed harder than the previous sets. The root gaps for the (CONIC+MC) formulation are higher than those of the previous sets. Nevertheless, the relative effectiveness of the formulations is consistent with the earlier experiments. With the (CONIC+MC)

TABLE 3. Results for hard problems.

ν_0	κ	assort bind	MILP		MILP+MC		CONIC		CONIC+MC	
			rgap egap	time/# nodes	rgap egap	time/# nodes	rgap egap	time/# nodes	rgap egap	time/# nodes
1	10	10.0	32.60%	17.29/5	6.50%	9.30/5	8.28%	209.88/1	1.75%	4.22/5
		5	0.00%	2662	0.00%	1154	1.04%	15614	0.00%	84
	20	20.0	27.93%	–	10.99%	–	8.19%	349.10/2	1.71%	14.20/5
		5	4.07%	20345	2.04%	28002	0.98%	8680	0.00%	370
1	50	50.0	3.15%	89.21/5	0.80%	3.47/5	12.96%	–	0.12%	1.22/5
		5	0.00%	14380	0.00%	1032	2.02%	9343	0.00%	0
	100	64.2	1.25%	3.94/5	0.21%	0.49/5	13.57%	–	0.06%	0.50/5
		0	0.00%	6099	0.00%	23	1.20%	10950	0.00%	0
2	10	10.0	12.90%	7.93/5	2.92%	4.24/5	3.79%	124.74/5	0.61%	2.61/5
		5	0.00%	1565	0.00%	484	0.00%	7514	0.00%	38
	20	20.0	21.82%	298.66/1	6.26%	267.82/2	4.55%	129.21/4	0.69%	8.62/5
		5	0.75%	42703	0.65%	32411	0.04%	6233	0.00%	188
2	50	50.0	6.24%	482.84/3	1.30%	47.18/4	8.06%	–	0.20%	6.64/5
		5	0.26%	23303	0.08%	20401	0.81%	10649	0.00%	251
	100	79.8	0.39%	1.01/5	0.01%	0.18/5	8.01%	183.98/4	0.00%	0.39/5
		0	0.00%	445	0.00%	0	0.07%	6950	0.00%	0
Average			13.29%	80.83/29	3.63%	26.22/31	8.43%	174.03/16	0.64%	4.8/40
			0.64%	13938	0.35%	10438	0.77%	9492	0.00%	116

formulation, all instances are solved within the time limit with an average run time under five seconds.

In the final set of experiments, we compare the formulations on instances with generalized capacity constraints. The general capacity data set has 200 products and 20 classes. The preference values and prices are generated as in Table 1. The model has six capacity constraints. The first constraint is a general capacity constraint $\sum_{j \in S} \beta_{0j} \leq \kappa_0$, where β_{0j} is generated uniformly between 0 and 1. The other five constraints are subset cardinality constraints $|S \cap N_k| \leq \kappa_k$, $k = 1, \dots, 5$ where N_k , $k = 1, \dots, 5$ are disjoint sets with $|N_k| = 40$. In order to obtain the lower bounds for the conditional McCormick inequalities, we use the following approach: For both conditions ($x_j = 1$ and $x_j = 0$), we first solve the linear relaxation of (BND) with only the capacity constraint using the greedy algorithm. We then solve the same problem with only the non-overlapping subset cardinality constraints also using the greedy algorithm. We use the minimum of the two relaxation values to obtain the lower bounds. Separately considering the constraints allows us to utilize fast greedy algorithms instead of using the simplex or an interior point algorithm for each variable-value combination. The results are shown in Table 4, where space reports the amount of capacity used ($\sum_{j \in N} \beta_{0j} x_j^*$, averaged over five instances) and bind now reports the number of instances where *all* subset cardinality constraints are tight in the optimal solution.

The results in Table 4 are consistent with earlier experiments. The CONIC+MC formulation leads to tight relaxations under generalized capacity constraints as well. All 50 instances are solved in under 30 seconds on average, whereas with the second best formulation (MILP+MC), only 20 instances are solved. We note that the time to compute the conditional bounds is negligible as we utilize a greedy approach to solve the relaxations.

TABLE 4. Results for problems with generalized capacity constraints.

ν_0	κ_0, κ_k	space/assort bind	MILP		MILP+MC		CONIC		CONIC+MC	
			rgap	time/#	rgap	time/#	rgap	time/#	rgap	time/#
			egap	nodes	egap	nodes	egap	nodes	egap	nodes
10	5,2	4.48/10.0	24.80%	–	7.65%	–	1.95%	194.98/4	0.10%	5.97/5
		5	10.07%	53576	5.51%	6836	0.11%	9351	0.00%	4
	10,4	9.42/20.0	40.82%	–	9.32%	–	3.78%	–	0.25%	12.12/5
		5	38.73%	3414	7.99%	10972	1.48%	7870	0.00%	97
	25,10	24.11/50.0	13.16%	–	3.84%	–	10.02%	–	0.44%	114.38/5
		5	11.89%	49065	2.68%	36783	4.22%	3008	0.00%	1971
	50,20	45.19/87.6	0.12%	4.81/5	0.04%	1.47/5	17.00%	–	0.01%	2.56/5
		0	0.00%	1291	0.00%	27	5.50%	767	0.00%	26
	100,40	45.66/88.6	0.04%	1.17/5	0.02%	0.9/5	17.19%	–	0.00%	1.81/5
		0	0.00%	95	0.00%	0	4.47%	1406	0.00%	0
20	5,2	4.50/10.0	6.94%	–	2.61%	479.15/1	1.03%	48.86/5	0.02%	4.72/5
		5	1.01%	136413	0.63%	32437	0.00%	3120	0.00%	0
	10,4	9.36/20.0	21.43%	–	5.25%	–	2.15%	–	0.10%	10.75/5
		5	14.77%	21448	4.35%	9093	0.63%	15089	0.00%	55
	25,10	24.28/50.0	18.45%	–	3.87%	–	5.73%	–	0.24%	97.01/5
		5	17.73%	12587	3.10%	21435	2.07%	4784	0.00%	1559
	50,20	49.20/98.6	1.54%	206.11/2	0.32%	21.57/4	11.66%	–	0.06%	10.58/5
		2	0.79%	135867	0.04%	9802	3.37%	868	0.00%	122
	100,40	60.81/120.2	0.03%	0.99/5	0.01%	0.75/5	12.82%	–	0.00%	1.41/5
		0	0.00%	99	0.00%	0	5.20%	1321	0.00%	0
Average			12.73%	26.30/17	3.29%	29.05/20	8.33%	113.80/9	0.12%	26.13/50
			9.50%	41385	2.43%	12738	2.71%	4758	0.00%	383

5. CONCLUDING REMARKS

In this paper, we present a conic quadratic mixed-integer formulation of the capacitated assortment optimization problem under the mixed multinomial logit model, that is far more effective than traditional MILP formulations of this problem with tight capacity constraints. Additional performance improvements are gained by using McCormick estimators derived through conditional bounds exploiting the capacity constraints. The numerical results suggest that with the new formulations, commercially available software may be practically used to solve even relatively large assortment optimization problems to optimality. Given the promise of conic mixed-integer formulations for the MMNL problem, it is worthwhile to explore conic optimization formulations of assortment optimization problems based on other consumer choice models.

ACKNOWLEDGEMENTS

A. Şen was supported by a 2219 fellowship grant from The Scientific and Technological Research Council of Turkey (TÜBİTAK). He acknowledges with gratitude the financial support of TÜBİTAK and hospitality of University of California-Berkeley. A. Atamtürk was supported, in part, by grant FA9550-10-1-0168 from the Office of the Assistant Secretary of Defense for Research and Engineering. P. Kaminsky was supported, in part, by industry members of the I/UCRC Center for Excellence in Logistics and Distribution, and by National Science Foundation Grant No. 1067994.

REFERENCES

- Aktürk, M., Atamtürk, A., and Gürel, S. (2010). Match-up scheduling with manufacturing cost considerations. *Journal of Scheduling*, 13:95–110.
- Aktürk, M. S., Atamtürk, A., and Gürel, S. (2014). Aircraft rescheduling with cruise speed control. *Operations Research*, 62:829–845.
- Alizadeh, F. and Goldfarb, D. (2003). Second-order cone programming. *Mathematical Programming*, 95:3–51.
- Atamtürk, A., Berenguer, G., and Shen, Z.-J. M. (2012). A conic integer programming approach to stochastic joint location-inventory problems. *Operations Research*, 60:366–381.
- Atamtürk, A., Muller, L. F., and Pisinger, D. (2013). Separation and extension of cover inequalities for conic quadratic knapsack constraints with generalized upper bounds. *INFORMS Journal of Computing*, 25:420–431.
- Atamtürk, A. and Narayanan, V. (2008). Polymatroids and risk minimization in discrete optimization. *Operations Research Letters*, 36:618–622.
- Atamtürk, A. and Narayanan, V. (2010). Conic mixed-integer rounding cuts. *Mathematical Programming*, 122:1–20.
- Atamtürk, A. and Narayanan, V. (2011). Lifting for conic mixed-integer programming. *Mathematical Programming*, 126:351–363.
- Ben-Tal, A. and Nemirovski, A. (2001). *Lectures on modern convex optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia.
- Boyd, J. H. and Mellman, R. E. (1980). The effect of fuel economy standards on the us automotive market: an hedonic demand analysis. *Transportation Research Part A: General*, 14:367–378.
- Bront, J. J. M., Méndez-Díaz, I., and Vulcano, G. (2009). A column generation algorithm for choice-based network revenue management. *Operations Research*, 57:769–784.
- Cachon, G. P., Terwiesch, C., and Xu, Y. (2005). Retail assortment planning in the presence of consumer search. *Manufacturing & Service Operations Management*, 7:330–346.
- Cardell, N. S. and Dunbar, F. C. (1980). Measuring the societal impacts of automobile downsizing. *Transportation Research Part A: General*, 14:423–434.
- Çezik, M. T. and Iyengar, G. (2005). Cuts for mixed 0-1 conic programming. *Mathematical Programming*, 104:179–202.
- Chong, J.-K., Ho, T.-H., and Tang, C. S. (2001). A modeling framework for category assortment planning. *Manufacturing & Service Operations Management*, 3:191–210.
- Davis, J. M., Gallego, G., and Topaloglu, H. (2014). Assortment optimization under variants of the nested logit model. *Operations Research*, 62:250–273.
- Désir, A. and Goyal, V. (2014). Near-optimal algorithms for capacity constrained assortment optimization. *Available at SSRN 2543309*.
- Feldman, J. and Topaloglu, H. (2015). Bounding optimal expected revenues for assortment optimization under mixtures of multinomial logits. *Production and Operations Management*, 24:1598–1620.
- Gallego, G. and Topaloglu, H. (2014). Constrained assortment optimization for the nested logit model. *Management Science*, 60:2583–2601.
- Kök, A. G. and Fisher, M. L. (2007). Demand estimation and assortment optimization under substitution: Methodology and application. *Operations Research*, 55:1001–1021.

- Kök, A. G., Fisher, M. L., and Vaidyanathan, R. (2009). Assortment planning: Review of literature and industry practice. In Agrawal, N. and Smith, S. A., editors, *Retail Supply Chain Management*, volume 122 of *International Series in Operations Research & Management Science*, pages 99–153. Springer US.
- Li, G., Rusmevichientong, P., and Topaloglu, H. (2015). The d-level nested logit model: Assortment and price optimization problems. *Operations Research*, 63:325–342.
- Li, Z. (2007). A single-period assortment optimization model. *Production and Operations Management*, 16:369–380.
- Liu, Q. and van Ryzin, G. (2008). On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management*, 10:288–310.
- Lobo, M., Vandenberghe, L., Boyd, S., and Lebret, H. (1998). Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228.
- Mahajan, S. and van Ryzin, G. (2001). Stocking retail assortments under dynamic consumer substitution. *Operations Research*, 49:334–351.
- McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems. *Mathematical Programming*, 10:147–175.
- McFadden, D. and Train, K. (2000). Mixed MNL models for discrete response. *Journal of Applied Econometrics*, 15:447–470.
- Méndez-Díaz, I., Miranda-Bront, J. J., Vulcano, G., and Zabala, P. (2014). A branch-and-cut algorithm for the latent-class logit assortment problem. *Discrete Applied Mathematics*, 164:246–263.
- Rusmevichientong, P., Shen, Z.-J. M., and Shmoys, D. B. (2010). Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research*, 58:1666–1680.
- Rusmevichientong, P., Shmoys, D., Tong, C., and Topaloglu, H. (2014). Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management*, 23:2023–2039.
- Rusmevichientong, P. and Topaloglu, H. (2012). Robust assortment optimization in revenue management under the multinomial logit choice model. *Operations Research*, 60:865–882.
- Topaloglu, H. (2013). Joint stocking and product offer decisions under the multinomial logit model. *Production and Operations Management*, 22:1182–1199.
- van Ryzin, G. and Mahajan, S. (1999). On the relationship between inventory costs and variety benefits in retail assortments. *Management Science*, 45:1496–1509.
- Vielma, J., Ahmed, S., and Nemhauser, G. (2008). A lifted linear programming branch-and-bound algorithm for mixed integer conic quadratic programs. *INFORMS Journal on Computing*, 20:438–450.