A Selective Newsvendor Approach to Order Management

Kevin Taaffe,[1] Edwin Romeijn,[2] Deepak Tirumalasetty[3]

[1] *Industrial Engineering, Clemson University, Clemson, South Carolina 29634*

[2] *Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109*

[3] *TransSolutions, LLC, Fort Worth, Texas 76155*

Received 26 April 2006; revised 20 July 2008; accepted 26 July 2008
DOI 10.1002/nav.20320
Published online 21 October 2008 in Wiley InterScience (www.interscience.wiley.com).

Abstract: Consider a supplier offering a product to several potential demand sources, each with a unique revenue, size, and probability that it will materialize. Given a long procurement lead time, the supplier must choose the orders to pursue and the total quantity to procure prior to the selling season. We model this as a selective newsvendor problem of maximizing profits where the total (random) demand is given by the set of pursued orders. Given that the dimensionality of a mixed-integer linear programming formulation of the problem increases exponentially with the number of potential orders, we develop both a tailored exact algorithm based on the L-shaped method for two-stage stochastic programming as well as a heuristic method. We also extend our solution approach to account for piecewise-linear cost and revenue functions as well as a multiperiod setting. Extensive experimentation indicates that our exact approach rapidly finds optimal solutions with three times as many orders as a state-of-the-art commercial solver. In addition, our heuristic approach provides average gaps of less than 1% for the largest problems that can be solved exactly. Observing that the gaps decrease as problem size grows, we expect the heuristic approach to work well for large problem instances. © 2008 Wiley Periodicals, Inc. Naval Research Logistics 55: 769–784, 2008

Keywords: selective newsvendor; demand selection; Bernoulli demands

1. INTRODUCTION

A well-known phrase in the service industry is that the "customer is always right." Many service-oriented businesses live and die by this motto, and they ensure that their customers receive quality service in hopes of building customer loyalty. Although this offers many advantages to consumers, it poses several logistical problems for producers and manufacturers of goods that have any significant amount of product or part lead time. Often, the manufacturer must make production and inventory decisions based on anticipated demand in future periods, and a proper understanding and treatment of the demand source information becomes essential.

The problem that we will study in this article is motivated by observations at a large manufacturer in the telecommunications industry. This manufacturer's sales teams attempt to secure orders for low-volume telecommunications infrastructure equipment. Unsecured orders are scheduled for a specific period of time into the future, with little knowledge about whether or not they will actually materialize. These orders are customized, but only at a certain (relatively late) point in the production process. Therefore the manufacturer can, in fact, begin procuring and assembling materials that are noncustomized, and then customize the product for individual customers once orders come in. In our setting, a customer is typically not an end-user and, moreover, is the dominant supply chain player who can therefore influence the manufacturer's production timeline. It is due to this imbalance in power that the manufacturer may allow customer orders/due dates within the procurement lead times for the product. Each customer has unique qualities, and some customers will invariably play a more dominant role in the industry. Therefore, the negotiated price of the product will be unique for each customer, and the salesforce allocation to each customer will also be unique.

In a more general context, we consider a firm that offers a product with uncertain demand. When the procurement lead time for this product is long, the firm needs to determine the procurement quantity (in a single-period setting) or sequence of quantities (in a multiperiod setting) in advance of the selling season. If the demand distribution is fixed and cannot be influenced, a standard newsvendor model can often be used to represent this problem and determine the optimal procurement quantity. We allow for demand flexibility

Correspondence to: K. Taaffe (taaffe@clemson.edu) by modeling the random demand as consisting of a set of potential demands (e.g., customer orders), each of which will materialize at a particular known level or not materialize at all. In addition, to maximize expected profit, the firm is able to not only choose the procurement quantity but also select the set of potential demands that it will actively pursue (where we assume that demands that are not actively pursued are essentially lost). Once the actual materialized demands are known, the firm must satisfy all pursued demands. In a single-period setting, underages are accounted for through an expediting or outsourcing cost; in a multiperiod setting we allow for backlogging at a cost. Overages in the single-period setting can be salvaged; in a multiperiod setting any overage can be held in inventory for use in subsequent periods.

Demand flexibility allows the firm to decide whether highly profitable, yet risky, orders are worth pursuing over less profitable, but possibly more predictable, orders. In contrast with the standard newsvendor problem, we allow the manufacturer to choose which orders to include in its "demand forecast," i.e., the manufacturer can shape the demand distribution for which to prepare by judiciously selecting which orders to actively pursue. Note that we allow the manufacturer to prepare for the orders that it wants to pursue by ordering materials and beginning a pre-customization phase of the production process (e.g., assembly). In many settings the manufacturer can customize its inventory within a short leadtime to meet the orders that materialize.

Inventory control and, more specifically, newsvendor problems have been actively researched. In addition to the summary by Porteus [13], Tsay et al. [20], and Cachon [2] provide reviews of research directions concerning supply chain contracts and competitive inventory management in single-period newsvendor settings. While product ordering decisions in many newsvendor problems typically assume a fixed demand distribution, papers that do address stochastic demand selection are Carr and Lovejoy [4] and Taaffe et al. [17, 18]. Carr and Lovejoy [4] examine a so-called *inverse newsvendor problem*, in which an optimal demand distribution is chosen based on some predefined order quantity or capacity set by a supplier. The demand distribution is selected from a set of feasible demand portfolios and based on an a priori ranking of several customer classes. Taaffe et al. [17] introduced the so-called *selective newsvendor problem*. With a set of independent markets, the problem is to simultaneously select the markets to supply as well as the appropriate order quantity to request from the supplier. Both Carr and Lovejoy [4] and Taaffe et al. [17, 18] assume that the demands of different customer classes or markets are independent and normally distributed. This is a realistic assumption if the demand of a customer class or market consists of demands of many independent customers. In contrast, in this paper we consider individual potential demands that cannot be modeled using normal random variables but, instead, are governed by Bernoulli distributions.

Our research is related to two classes of order management: (1) admission control and sequencing problems, and (2) multiproduct/multiperiod newsvendor problems. The former class of models focuses on developing rationing policies for distributing the product to a subset of customer demand (or, in effect, selecting only certain demands to fill). These models employ queueing theory and Markov Decision Processes in analyzing the admission policies (see, e.g., Carr and Duenyas [3], Gupta and Wang [6], Ha [7], and De Vе́ricourt et al. [5]). In the latter class of models, which is more closely related to ours, methods have been proposed for dealing with a product that is offered at several price (or demand) levels as well as across multiple periods. In Shumsky and Zhang [16], the firm must purchase its capacity for each demand level before the first period and cannot request any further replenishments. Demand flexibility is obtained by incorporating product substitution, or shifting product from one demand class to another. Other research has allowed additional quantities to be procured during the selling season. That is, as demand information is revealed, the manufacturer can make procurement decisions for the next period (see, e.g., Shen and Zhang [15], Monahan et al. [12], and Kouvelis and Gutierrez [9]).

In Section 2, we will present our single-period model, which we will call the selective newsvendor problem (SNP) with all-or-nothing orders. We develop an algorithm based on the L-shaped method (see, e.g., Van Slyke and Wets [21] and Laporte and Louveaux [10]) that is tailored to our problem. We close this section with a generalization of the cost and revenue structure by allowing the end-of-period shortage cost function to be piecewise-linear and convex and the end-of-period revenues from salvaging any overages to be piecewise-linear and concave. In Section 3, we formulate and analyze a multiperiod demand selection problem, generalizing the solution approach to the multiperiod case. In Section 4 we provide computational results for each of the three problem classes, and compare and contrast the solution times obtained for each problem with results obtained by a commercial solver.

2. SINGLE-PERIOD ALL-OR-NOTHING DEMAND SELECTION

2.1. Problem Formulation

Consider a set of n potential orders that a supplier can serve. Let D_i denote the random variable representing the magnitude of order i $(i = 1, \ldots, n)$ and assume that these order sizes (or demands) are statistically independent. As mentioned earlier, in this article we consider a situation where an order may either come in at a predefined level or not at all, i.e., demand i is governed by a Bernoulli distribution:

$$\Pr(D_i = x) = \begin{cases} 1 - p_i & \text{if } x = 0 \\ p_i & \text{if } x = d_i, \end{cases}$$

where p_i represents the probability that order i will materialize. The firm must decide, in advance of the selling season, both the orders it will pursue and the total quantity Q it will procure from its supplier to maximize its expected profit. Note that the model can allow for booked or firm orders by setting $p_i = 1$ for a specific order i.

Let the per unit procurement cost from this supplier be given by c. Furthermore, let r_i be the per unit revenue associated with order i. We can assume without loss of generality that $r_i > c$, otherwise we could immediately eliminate order i from consideration. We also include a fixed cost of S_i, which would account for the dollar value of the time and resources spent by the firm in trying to secure order i. This has been described as salesforce allocation costs in Section 1.

Given a set of selected orders, the firm must ultimately satisfy all realized ones. In situations when the customer can influence when an order should be delivered, the firm may be required to outsource production if the in-house quantity is not sufficient to meet demand by the contracted date. We assume a high-cost domestic supplier exists from which the firm can expedite units of the good (after observing demand) at a per unit cost of e, where $e > c$. Any unsold items can be salvaged for v per unit, where $c > v$. The customer has no obligation to the firm until the customer places a contracted order (i.e., there is no penalty cost for not placing an order.)

We model the fact that the firm can select the orders to pursue by introducing binary demand selection variables $y_i (i = 1, \ldots, n)$, where $y_i = 1$ corresponds to the selection of order i and $y_i = 0$ to the rejection of order i. The total expected profit can then be written as:

$$G(Q, y) = \sum_{i=1}^{n} (r_i d_i p_i - S_i) y_i - cQ + vE\left[\max\left(0, Q - \sum_{i=1}^{n} D_i y_i\right)\right] - eE\left[\max\left(0, \sum_{i=1}^{n} D_i y_i - Q\right)\right].$$

The *newsvendor problem with all-or-nothing demand* [AON] is now given by

$$\begin{aligned} \text{maximize} \quad & G(Q, y) \\ \text{subject to} \quad & Q \geq 0 \\ & y_i \in \{0, 1\} \quad i = 1, \ldots, n. \end{aligned}$$

We obtain a more explicit formulation of the profit function by defining a set $I_\omega \subseteq \{1, \ldots, n\}$ which contains those orders realized in demand scenario ω. Note that there are a total of $\Omega = 2^n$ potential scenarios. Furthermore, let

$$P_\omega = \prod_{i \in I_\omega} p_i \cdot \prod_{i \notin I_\omega} (1 - p_i)$$

denote the probability that demand scenario ω is realized. The expected profit function then reduces to

$$\begin{aligned} & G(Q, y) \\ & \quad = \sum_{i=1}^{n} (r_i d_i p_i - S_i) y_i - CQ \\ & \qquad + \sum_{\omega=1}^{\Omega} P_\omega \left[v \max\left(0, Q - \sum_{i \in I_\omega} d_i y_i\right) \right. \\ & \qquad \left. - e \max\left(0, \sum_{i \in I_\omega} d_i y_i - Q\right) \right] \\ & \quad = \sum_{i=1}^{n} (r_i d_i p_i - S_i) y_i - CQ + v \sum_{\omega=1}^{\Omega} P_\omega \left(Q - \sum_{i \in I_\omega} d_i y_i \right) \\ & \qquad - (e - v) \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i \in I_\omega} d_i y_i - Q\right) \\ & \quad = \sum_{i=1}^{n} (r_i d_i p_i - S_i) y_i - (c - v)Q - v \sum_{i=1}^{n} d_i p_i y_i \\ & \qquad - (e - v) \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i \in I_\omega} d_i y_i - Q\right) \\ & \quad = \sum_{i=1}^{n} ((r_i - v) d_i p_i - S_i) y_i - (c - v)Q \\ & \qquad - (e - v) \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i \in I_\omega} d_i y_i - Q\right). \end{aligned}$$

It is easy to see that, by introducing artificial variables u_ω representing the shortage in scenario $\omega (\omega = 1, \ldots, \Omega)$, [AON] can be formulated as a mixed-integer linear programming problem (MIP):

$$\text{maximize} \quad \sum_{i=1}^{n} ((r_i - v) d_i p_i - S_i) y_i - (c - v)Q - (e - v) \sum_{\omega=1}^{\Omega} P_\omega u_\omega$$

$$\begin{aligned} \text{subject to} \quad & u_\omega \geq \sum_{i \in I_\omega} d_i y_i - Q \quad \omega = 1, \dots, \Omega \qquad (1) \\ & u_\omega \geq 0 \quad \omega = 1, \dots, \Omega \\ & Q \geq 0 \\ & y_i \in \{0, 1\} \quad i = 1, \dots, n. \end{aligned}$$

Clearly, the number of variables and constraints in this problem grows linearly in the number of scenarios, and therefore exponentially in the number of orders. Although a 10-order problem has approximately 1000 decision variables and constraints, for a 20-order problem this increases to about 1,000,000. Even the construction of problems of this size becomes intractable. In the next section, we therefore develop an alternative, tailored solution approach that does not, in general, require all scenarios to be enumerated.

A potential solution approach would be to deal with this issue by using a cutting plane approach for constraints (1). In particular, we could define a master problem in which the constraints in (1) are imposed only for some subset of scenarios $\omega \in \mathcal{W} \subseteq \{1, \dots, \Omega\}$. Clearly, in an optimal solution of this master problem we would then have $u^*_\omega = 0$ for all $\omega \in \{l, \dots, \Omega\} \setminus \mathcal{W}$ so that the master problem is tractable for relatively small sets $\mathcal{W}$. If the optimal solution to the master problem satisfies *all* constraints (1), it is clearly optimal to (MIP). Otherwise, we would add one or more of the violated constraints and resolve the master problem. Now suppose, for ease of exposition, that the optimal solution to the full problem (MIP) is unique and equal to (Q^*, y^*). Then this procedure will need to generate *at least* all constraints for scenarios ω for which $\sum_{i \in I_\omega} d_i y^*_i > Q^*$. Since for the optimal order selection vector y^* the value Q^* is the $(e - c)/(e - v)$ quantile of the distribution of the aggregate demand in the selected orders, the number of constraints that this procedure can be expected to generate is on the order of

$$\left(\frac{e - c}{e - v} \right) \cdot 2^{n(y^*)},$$

where $n(y^*) = \sum_{i=1}^n y^*_i$ is the number of orders selected in the optimal solution. In cases where the number of attractive orders is substantial, this means that even such a cutting plane approach will quickly become intractable. In the next section, we therefore develop an alternative, tailored solution approach that, as we will show empirically, in general does not require the solution of a large-scale integer programming problem.

2.2. A Cutting Plane Algorithm

The MIP formulation of [AON] can be viewed as a two-stage stochastic programming problem with simple recourse, where the first stage decisions are the procurement quantity Q and the demand selection variables y, and the second stage decisions are the shortage variables u. In particular, [AON] exhibits simple recourse, a problem well-studied in stochastic programming. In this section, we develop a cutting plane algorithm for solving [AON]. Our method is based on the idea of the so-called L-shaped method (LSM) (see, e.g., Birge and Louveaux [1]), which applies Benders decomposition to a suitable reformulation of a linear two-stage stochastic programming problem with fixed recourse. However, we will employ the specific structure of [AON] to more directly derive our algorithm. Introducing a new decision variable θ, we can formulate [AON] as

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^n ((r_i - v) d_i p_i - S_i) y_i - (c - v) Q - (e - v) \theta \\ \text{subject to} \quad & \theta \geq \sum_{\omega=1}^{\Omega} P_\omega \max \left(0, \sum_{i \in I_\omega} d_i y_i - Q \right) \qquad (2) \\ & Q \geq 0 \\ & y_i \in \{0, 1\} \quad i = 1, \dots, n. \end{aligned}$$

It will be convenient to associate a binary vector $\xi^\omega \in \{0, 1\}^n$ with each scenario ω by letting $\xi^\omega_i = 1$ if $i \in I_\omega$ and $\xi^\omega_i = 0$ otherwise. We next replace constraint (2) by the following 2^Ω *linear* constraints, parameterized by a set of unique binary vectors $\delta \in \{0, 1\}^\Omega$ representing every possible choice in the maximum in the right-hand side of (2):

$$\theta \geq \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^n d_i y_i \xi^\omega_i - Q \right) \delta_\omega \quad \delta_\omega \in \{0, 1\}^\Omega. \qquad (3)$$

Because $\omega = 2^n$, the number of constraints in the resulting formulation of the problem is far too large for practical purposes. Therefore, our approach will be to include only a few of these constraints and add additional constraints as needed. In particular, after solving an approximation of [AON] in which only a subset of the constraints on θ are included, we determine whether there exists any omitted constraint on θ that is violated. If so, we add such a constraint to the formulation and re-solve the problem.

A crucial observation is that, for a particular solution (Q, y) to [AON], the constraint for which

$$\delta_\omega = \begin{cases} 1 & \text{if } \sum_{i \in I_\omega} d_i y_i > Q \\ 0 & \text{otherwise} \end{cases} \quad \omega = 1, \dots, \Omega \qquad (4)$$

is most restrictive in the sense that it achieves the maximum constraint violation among all constraints (3). (Note that setting $\delta_\omega = 1$ for some or all $\omega \in \Omega$ such that $\sum_{i \in I_\omega} d_i y_i = Q$ would yield a constraint, that achieves the same violation for (Q, y). However, because it is not possible to say which of these will yield the strongest constraint, we will in most cases use constraint 4 and refer to it as "the most restrictive constraint" despite the fact that it is not necessarily the only constraint that has this property.)

Therefore, suppose that $(\tilde{Q}, \tilde{y}, \tilde{\theta})$ is the optimal solution of the current approximation of [AON], and denote the indicator variable of the most restrictive constraint corresponding to $(\tilde{Q}, \tilde{y})$ by $\tilde{\delta}$. If the corresponding constraint is satisfied, the current solution is the optimal solution to [AON]. Otherwise, we may strengthen our approximation by including the constraint corresponding to $\tilde{\delta}$. Given that the total number of constraints is finite, it is guaranteed that this procedure will converge.

The issue that remains to be addressed is: given the large number of scenarios, can we efficiently construct constraint (3) for $\tilde{\delta}$ *without* enumerating all scenarios? To this end, let us reformulate the constraints of [AON] to highlight the coefficients of the decision variables:

$$\theta \geq \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^{n} d_i y_i \xi_i^\omega - Q \right) \delta_\omega$$
$$= \sum_{i=1}^{n} \left(\sum_{\omega=1}^{\Omega} P_\omega \xi_i^\omega \delta_\omega \right) d_i y_i - Q \left(\sum_{\omega=1}^{\Omega} P_\omega \delta_\omega \right) \qquad \delta \in \{0,1\}^\Omega.$$

Therefore, the problem is in fact to determine the coefficients $\zeta_0 \equiv \sum_{\omega=1}^{\Omega} P_\omega \tilde{\delta}_\omega$ (for Q) and $\zeta_i \equiv \sum_{\omega=1}^{\Omega} P_\omega \xi_i^\omega \tilde{\delta}_\omega$ (for $d_i y_i$, $i = 1, \ldots, n$). While in the worst case the computation of these coefficients may take $\mathcal{O}(2^n)$ time, we propose algorithms that can be expected to run much faster in general.

Note that the approximations of [AON] that only contain a subset of the constraints on θ are still MIPs that may be hard to solve. (Therefore, this approach is often referred to as the integer L-shaped method, ILSM.) As an alternative, we could apply the same approach to the LP-relaxation of [AON] and embed this algorithm in a branch-and-bound algorithm to solve the actual MIP. Interestingly, some researchers have found applications of the LSM where the former approach is much more efficient in practice than the second. Intuitively, this may be due to the fact that in the former approach, even if each approximate problem is a MIP, only one application of the ILSM is required. This is in contrast with the alternative, where numerous applications of the LSM may be required as part of a branch-and-bound scheme (see, e.g., Laporte and Louveaux [10], Laporte et al. [11], and Schaefer et al. [14]). In other words, the ILSM provides an immediate solution to [AON]. In our experiments we will compare the efficiency of solving [AON] using the ILSM versus solving its LP-relaxation using the LSM.

2.3. Coefficient Generation

It is easy to see that the performance of our algorithm for [AON] depends heavily on the computational effort that is required to find the coefficients of the cutting plane. We can compute the $n + 1$ coefficients of this constraint by traversing a binary tree whose leaves represent all potential demand scenarios. We will develop algorithms that attempt to compute the coefficients without explicitly having to enumerate all 2^n leaves of this tree. In particular, we will describe two algorithms, each one performing best for particular values of $\tilde{Q}$ and $\tilde{y}$.

2.3.1. Forward Algorithm

Let $(\tilde{Q}, \tilde{y}, \tilde{\theta})$ be the solution to the current approximation of [AON]. We then construct a binary tree that represents all scenarios as follows. At the root node we start with an empty scenario. Then, at a node at level k, we construct two child nodes that correspond to the scenarios where order k is realized or not, respectively. At a given node at depth k of this tree, we keep track of

1. the probability, say p, associated with all scenarios that are leaves of the corresponding subtree, i.e., the joint probability of the $k-1$ order realizations corresponding to the unique path from the root of the tree to the current node;
2. a lower bound on total demand, say Δ, associated with all scenarios that are leaves of the corresponding subtree, i.e., using a demand for order i of $d_i \tilde{y}_i (i = 1, \ldots, k)$, the total demand of the k order realizations along the unique path from the root of the tree to the current node.

We can now avoid searching the entire binary tree using the following observation. Suppose that, at some node, we have that $\Delta \geq \tilde{Q}$. We then know that for all scenarios that are leaves of the current subtree the total realized demand will be no less than $\tilde{Q}$ and we can prune the binary tree after updating the values of the constraint coefficients appropriately. In addition, note that $\Delta + \sum_{i=k+1}^{n} d_i \tilde{y}_i$ is an upper bound on the total demand associated with all scenarios that are leaves of the current subtree. We can thus also check whether this upper bound on realized demand exceeds $\tilde{Q}$; if not, we can prune the binary tree (without updating the values of the constraint coefficients) and exclude all scenarios that are leaves of the current subtree. The Forward Algorithm can be stated more formally as follows (where, at any stage of the algorithm, the vector ξ denotes the binary vector representing the current partial scenario):

```
Constraint Generation Algorithm—Forward
0. Initialize the $n+1$ constraint coefficients: $\zeta_i = 0$ ($i = 0, \ldots, n$). Set $k = 0$, $p = 1$, $\Delta = 0$, and $\xi_0 = 0$.
1. (Branching Step - Orders Realized) While $\Delta < \tilde{Q} \leq \Delta + \sum_{i=k+1}^{n} d_i \tilde{y}_i$:
   – set $k = k + 1$;
```

- add the next order: $\xi_k = 1$;
- update the node probability: $p = p \cdot p_k$; and
- update the total realized demand at the current node: $\Delta = \Delta + d_k \tilde{y}_k$.

2. If $\tilde{Q} \le \Delta$ the current scenario subtree *should be included* in the coefficients:

$$\zeta_i = \begin{cases} \zeta_i + p & \text{for } i = 0, \\ \zeta_i + p \cdot \xi_i & \text{for } i = 1, \dots, k, \\ \zeta_i + p \cdot p_i & \text{for } i = k+1, \dots, n. \end{cases}$$

Otherwise, $\tilde{Q} > \Delta + \sum_{i=k+1}^{n} d_i \tilde{y}_i$ and the subtree *should not be included* in the coefficients.

3. (Backtracking Step) While $\xi_k = 0$:
   - update the node probability: $p = p/(1 - p_k)$;
   - set $k = k - 1$.
4. (Change to Unrealized Order) If $k = 0$, stop...all constraint coefficients are computed. Otherwise:
   - remove order $k : \xi_k = 0$;
   - update the node probability: $p = p \cdot (1 - p_k)/p_k$; and
   - update the total realized demand at the current node: $\Delta = \Delta - d_k \tilde{y}_k$.

   Return to Step 1.

For the first violation in Step 1, no backtracking will occur in Step 3 (because all orders are included in the current scenario), and the order at current level k will be flipped to "not realized." We continue to traverse the tree of scenarios, pruning the tree by flipping orders from realized to unrealized, and backtracking when the current level order is unrealized. Note that, when we use the ILSM rather than the LSM, the vector $\tilde{y}$ is binary. In that case, we can simply remove all orders for which $\tilde{y}_i = 0$ from consideration and sort the selected orders in non-increasing order of d_i.

The algorithm, which is based on a scenario tree that starts with no orders realized, can be expected to work well if the value of $\tilde{Q}$ is relatively small or relatively large (i.e., when $\tilde{Q}$ is either close to 0 or close to the total potential demand in the current solution, $\sum_{i=1}^{n} d_i \tilde{y}_i$). In these cases, either one of the two bounds will be violated more quickly than for intermediate values of $\tilde{Q}$.

In the next section we will construct an algorithm that instead builds the scenario tree by starting with all of the orders realized. While similar in structure to the Forward Algorithm, that algorithm may provide reduced coefficient construction times for certain types of problems or for specific iterations in a problem, depending on the current solution information, $\tilde{Q}$ and $\sum_{i=1}^{n} d_i \tilde{y}_i$.

2.3.2. Backward Algorithm

The Backward Algorithm considers a binary tree that represents all scenarios as follows. At the root node we start with a scenario in which all orders are realized. Then, at a node at level k, we construct two child nodes that correspond to the scenarios where order k is removed from the root scenario or not, respectively. At a given node at depth k of this tree, we keep track of

1. the probability, say p, associated with all scenarios that are leaves of the corresponding subtree, i.e., the joint probability of the $k-1$ order realizations corresponding to the unique path from the root of the tree to the current node;
2. an upper bound on the total demand, say Δ', associated with all scenarios that are leaves of the corresponding subtree, i.e., using a demand for order i of $d_i \tilde{y}_i (i = 1, \dots, k-1)$, the total demand of the $k-1$ order realizations along the unique path from the root of the tree to the current node as well as all remaining demands.

It is clear that this scenario tree is equivalent to the one used in the previous section, and we can again avoid searching the entire binary tree. Suppose that, at some node, we have that $\Delta' < \tilde{Q}$. We then know that for all scenarios that are leaves of the current subtree the total realized demand will be less than $\tilde{Q}$ and we can prune the binary tree after updating the values of the constraint coefficients appropriately. In addition, $\Delta' - \sum_{i=k+1}^{n} d_i \tilde{y}_i$ is a lower bound on the total demand associated with all scenarios that are leaves of the current subtree. We can thus also check whether this lower bound on realized demand is still less than $\tilde{Q}$; if not, we can prune the binary tree (without updating the values of the constraint coefficients) and exclude all scenarios that are leaves of the current subtree. The Backward Algorithm can be stated more formally as follows:

Constraint Generation Algorithm—Backward

0. Initialize the $n+1$ constraint coefficients: $\zeta_0 = 1$ and $\zeta_i = p_i (i = 1, \dots, n)$. Set $k = 0$, $p = 1$, $\Delta' = \sum_{i=1}^{n} d_i \tilde{y}_i$, and $\xi_0 = 1$.
1. (Branching Step - Orders Not Realized) While $\Delta' - \sum_{i=k+1}^{n} d_i \tilde{y}_i < \tilde{Q} \le \Delta'$:
   - set $k = k + 1$;
   - remove the next order: $\xi_k = 0$;
   - update the node probability: $p = p \cdot (1 - p_k)$; and
   - update the total realized demand at the current node: $\Delta' = \Delta' - d_k \tilde{y}_k$.
2. If $\tilde{Q} > \Delta'$ the current scenario subtree *should be excluded* from the coefficients:

$$\zeta_i = \begin{cases} \zeta_i - p & \text{for } i = 0, \\ \zeta_i - p \cdot \xi_i & \text{for } i = 1, \dots, k, \\ \zeta_i - p \cdot p_i & \text{for } i = k+1, \dots, n. \end{cases}$$

Otherwise, $\tilde{Q} \leq \Delta' - \sum_{i=k+1}^{n} d_i \tilde{y}_i$ and the subtree *should not be excluded* from the coefficients.

3. (Backtracking Step) While $\xi_k = 1$:
   - update the node probability: $p = p/p^k$;
   - set $k = k - 1$.
4. (Change to Realized Order) If $k = 0$, stop...all constraint coefficients are computed. Other wise:
   - add order k: $\xi_k = 1$;
   - update the node probability: $p = p \cdot p_k/(1 - p_k)$; and
   - update the total realized demand at the current node: $\Delta' = \Delta' + d_k \tilde{y}_k$.

     Return to Step 1.

2.3.3. *Performance of the Constraint Generation Algorithms*

Even with all orders selected ($\tilde{y}_i = 1$ for all i), either algorithm will not require a complete enumeration of the 2^n nodes in the scenario tree, yet the algorithms are still not polynomial in n. Through significant experimentation, we did identify characteristics of the test instances that would affect constraint generation times. Most notably, the algorithms are affected by the current solution $(\tilde{y}, \tilde{Q})$, the critical fractile (ρ), and the likelihood (p_i) of each order i occurring.

Note that we can, in principle, apply the Forward and Backward Algorithms using *any* sorting of the orders. However, to be able to prune nodes as quickly as possible we will sort the n orders in *non-increasing* order of their potential demand. The following theorem shows that this provides the most efficient pruning and therefore the least number of nodes visited in the scenario tree.

THEOREM 2.1: Both constraint generation algorithms will visit the minimum number of nodes if the orders are sorted in non-increasing order of their potential demand, $d_i \tilde{y}_i$.

PROOF: A tree pruning will occur whenever either bound on $\tilde{Q}$ is violated (see Forward and Backward Algorithm descriptions). Consider Step 1 of either algorithm, which represents the descent into the scenario tree until we can prune. For the Forward (Backward) algorithm, each time this step is performed the lower (upper) bound on $\tilde{Q}$ is increased (decreased) by $\xi_k d_k \tilde{y}_k$. Likewise, whenever an order is flipped, we also have a change in bound value. For the Forward (Backward) algorithm, Step 4 will cause the upper (lower) bound on Q to decrease (increase) by $\xi_k d_k \tilde{y}_k$. Therefore, the most significant change in upper and lower bound values will be for the largest values of $\xi_k d_k \tilde{y}_k$. We will cause the condition in Step 1 to be violated most quickly, thereby minimizing the depth until we can prune the tree, by sorting the orders so that $d_1\tilde{y}_1 \geq d_2\tilde{y}_2 \geq \cdots \geq d_n\tilde{y}_n$. □

2.4. Extension to Piecewise-linear Cost Functions

In this section, we examine how [AON] can be generalized to allow for more general, in particular piecewise-linear convex, shortage and overage cost functions (where, for convenience, we will in this section refer to the salvage revenue functions as (negative) overage cost functions). In other words, as the shortage or overage increases, the corresponding marginal unit cost is nondecreasing, representing the fact that the unit salvage value may decrease as the quantity salvaged increases and, similarly, the unit expediting cost may increase as the quantity expedited increases. For the sake of brevity, we omit certain repetitive mathematical steps in arriving at the problem formulation and constraint generation functions. Complete details are provided in the Appendix.

Let the marginal shortage costs and salvage values be given by $e_j (j = 1, \ldots, J^s + 1)$ and $v_j (j = 1, \ldots, J^o + 1)$ where $v_{J^o+1} < \cdots < v_1 < c < e_1 < \cdots < e_{J^s+1}$. For convenience, we will also let $e_0 = v_0 = 0$. Finally, denote the corresponding breakpoints by $s_j (j = 1, \ldots, J^s)$ and $o_j (j = 1, \ldots, J^o)$, respectively, where $0 = s_0 < s_1 < \cdots < s_{J^s}$ and $0 = o_0 < o_1 < \cdots < o_{J^o}$. (Note that if $J^o = J^s = 0$ then we obtain [AON].) For convenience, we will let $J = 1 + J^o + J^s$.

After deriving the expected profit equation, we can expand the formulation used for [AON] to represent the MIP we call the *newsvendor problem with all-or-nothing demand and piecewise linear costs* [AON-PWL]. The dimensionality of this MIP increases linearly in the number of segments in the cost functions. It thus suffers from the same drawbacks as the MIP for [AON].

We can use a similar approach as was used for [AON] in generalizing the cutting plane algorithm for [AON-PWL]. Now, we introduce three (sets of) decision variables, each corresponding to one of the expected values in the objective function of [AON-PWL]: θ_1, $\theta_j^o (j = 1, \ldots, J^o)$, and $\theta_j^s (j = 1, \ldots, J^s)$. This results in one additional variable and constraint for each breakpoint in the overage or underage cost function. Using these decision variables, as well as replacing each single constraint by 2^Ω linear constraints, we reformulate [AON-PWL] as

$$\begin{aligned}\text{maximize} \quad & \sum_{i=1}^{n}((r_i - v_1)d_i p_i - S_i)y_i - (c - v_1)Q \\ & - (e_1 - v_1)\theta_1 - \sum_{j=1}^{J^o}(v_j - v_{j+1})\theta_j^o \\ & - \sum_{j=1}^{J^s}(e_{j+1} - e_j)\theta_j^s\end{aligned}$$

subject to

$$\theta^1 \geq \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^{n} d_i y_i \xi_i^\omega - Q \right) \delta_\omega^1$$

$$\delta^1 \in \{0,1\}^\Omega \quad (5)$$

$$\theta_j^o \geq \sum_{\omega=1}^{\Omega} P_\omega \left(Q - \sum_{i=1}^{n} d_i y_i \xi_i^\omega - o_j \right) \delta_{j\omega}^o$$

$$\delta_j^o \in \{0,1\}^\Omega; j = 1, \ldots, J^o \quad (6)$$

$$\theta_j^s \geq \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^{n} d_i y_i \xi_i^\omega - Q - s_j \right) \delta_{j\omega}^s$$

$$\delta_j^s \in \{0,1\}^\Omega; j = 1, \ldots, J^s \quad (7)$$

$$Q \geq 0$$

$$y_i \in \{0,1\} \quad i = 1, \ldots, n.$$

Analogous to [AON], we now identify the most restrictive constraints with respect to a given solution (Q, y) as those for which

$$\delta_\omega^1 = \begin{cases} 1 & \text{if } \sum_{i \in I_\omega} d_i y_i > Q \\ 0 & \text{otherwise} \end{cases} \quad \omega = 1, \ldots, \Omega \quad (8)$$

$$\bar{\delta}_{j\omega}^o = \begin{cases} 1 & \text{if } \sum_{i \in I_\omega} d_i y_i > Q - o_j \\ 0 & \text{otherwise} \end{cases} \quad \omega = 1, \ldots, \Omega;$$

$$j = 1, \ldots, J^o \quad (9)$$

$$\delta_{j\omega}^s = \begin{cases} 1 & \text{if } \sum_{i \in I_\omega} d_i y_i > Q + s_j \\ 0 & \text{otherwise} \end{cases} \quad \omega = 1, \ldots, \Omega;$$

$$j = 1, \ldots, J^s. \quad (10)$$

where the dummy indicator variables $\bar{\delta}_j^o$ are precisely the complements of the indicator variables δ_j^o in (6). It is easy to see that the coefficients in constraints (5)-(7) can be identified by the Forward and Backward Algorithms described in Section 2.3, with an appropriate modification of Q and post-processing for (6) and (7). We thus perform J binary tree searches to implicitly identify, in each iteration of the cutting plane method, all scenarios that determine a particular constraint.

3. MULTIPERIOD ALL-OR-NOTHING DEMAND SELECTION PROBLEMS

3.1. Introduction

We will next consider stochastic order selection problems in a dynamic environment, i.e., the timing of production and selection of individual uncertain orders are integrated to maximize expected profit. Each order within a multiperiod setting is considered independent. As introduced for the single-period model, customer orders are customized for a particular installation, and they are somewhat infrequent, which allows the assumption that each order can be treated individually. In addition, spreading an order over several periods is not desirable, and it is preferred to satisfy an entire order in the period that it is expected to materialize. We develop a basic multiperiod model that can serve as a proof of concept for extending the single-period news vendor-based models to a multiperiod setting. We will show how the cutting plane algorithm that was developed in earlier sections based on the L-shaped method can be extended to the multiperiod case, allowing us to solve problems to optimality for reasonable sets of orders and time periods where standard solvers fail to be able to do this.

3.2. Problem Formulation

We now consider a multiperiod nonstationary newsvendor problem where the planning periods are denoted by $t = 1, \ldots, T$. In period t, we have a set of n_t potential orders that a supplier can serve in period t. Let D_{it} denote the random variable representing the magnitude of order i in period t. As before, we assume that these order sizes are statistically independent Bernoulli random variables:

$$\Pr(D_{it} = x) = \begin{cases} 1 - p_{it} & \text{if } x = 0 \\ p_{it} & \text{if } x = d_i \end{cases}$$

$$i = 1, \ldots, n_t; t = 1, \ldots, T.$$

Items can be procured in each period t at a per unit cost of c_t. In general, an order-up-to policy is expected to be optimal for this problem (see Veinott [22]). Here, we consider an alternative policy that is not only easier to implement in practice but also lends itself much more readily to a study of the merits of demand flexibility. In particular, we assume that the supplier must decide, before the start of the planning horizon, a full sequence of order quantities and a set or order selection decisions. This then yields a two-stage stochastic programming model in which the decision variables $Q_t (t = 1, \ldots, T)$ and $y_{it} (i = 1, \ldots, n; t = 1, \ldots, T)$ are determined in the first stage and the sales and inventory decisions are made after a random scenario ω is realized. This policy can be viewed as a dynamic version of the stationary periodic review/fixed order quantity policy that is commonly used when demands are stochastic but stationary, while a dynamic order-up-to policy would generalize a stationary order-up-to policy for such a system. In each period, any surplus is charged a holding cost of v_t per unit while any shortage is charged a penalty cost of e_t per unit. In addition, the supplier must ultimately satisfy all realized demand but may sell any overage at a salvage value at the end of the planning horizon. To this end, we include in e_T the unit cost required to satisfy demand that is not satisfied by the end of the planning horizon and in v_T the salvage value for any remaining stock at the end of the planning horizon.

As in Section 2, we let r_{it} be the per unit revenue associated with order i in period t, while we also allow for a fixed cost of pursuing this order of $S_{it}(i = 1,\ldots,n_t, t = 1,\ldots,T)$. For convenience, we will denote the total number of potential orders by $N = \sum_{t=1}^{T} n_t$.

In an analogous fashion to the single-period model, we represent a particular realization of demands by a demand scenario in the form of a binary vector $\xi^\omega \in \{0,1\}^N$, where $\xi_{it}^\omega = 1$ represents that order i in period t will materialize. The probability that this scenario occurs will again be denoted by P_ω, and the number of scenarios is given by $\Omega = 2^N$. The total demand faced by the supplier in period t given an order selection vector y is then clearly given by $\sum_{i=1}^{n_t} d_{it} y_{it} \xi_{it}^\omega$. Finally, we denote the quantity held in inventory from period t to period $t+1$ in scenario ω by I_t^ω and the quantity backlogged from period t to period $t+1$ in scenario ω by B_t^ω. We can then formulate the *multiperiod selective newsvendor problem with all-or-nothing demands* [AON-MP] as a MIP problem as follows:

$$\text{maximize} \quad \sum_{t=1}^{T} \left(\sum_{i=1}^{n_t} (r_{it} d_{it} p_{it} - S_{it}) y_{it} - c_t Q_t - \sum_{\omega=1}^{\Omega} P_\omega \big(v_t I_t^\omega + e_t B_t^\omega \big) \right)$$

subject to

$$I_{t-1}^\omega + B_t^\omega + Q_t = \sum_{i=1}^{n_t} d_{it} y_{it} \xi_{it}^\omega + I_t^\omega + B_{t-1}^\omega \qquad t = 1,\ldots,T; \omega = 1,\ldots,\Omega \qquad (11)$$

$$I_t^\omega, B_t^\omega \geq 0 \quad t = 1,\ldots,T; \omega = 1,\ldots,\Omega \qquad (12)$$

$$Q_t \geq 0 \quad t = 1,\ldots,T$$

$$y_{it} \in \{0,1\} \quad i = 1,\ldots,n_t; t = 1,\ldots,T.$$

Note that $I_0^\omega = I_0$ and $B_0^\omega = B_0$ represent the initial inventory and backlogging levels which are of course independent of the scenario ω and are input parameters to the model rather than decision variables. It is easy to see that, without loss of generality, we can restrict ourselves to solutions in which $I_t^\omega \cdot B_t^\omega = 0$ for $t = 0,\ldots,T$ and $\omega = 1,\ldots,\Omega$, i.e., we cannot have positive inventory and backlogging amounts simultaneously. Also, if the salvage value exceeds the holding cost in period T, then $v_T < 0$ in order to correctly represent its value in the objective function. Finally, we can easily represent the case of allowing a single initial order (Q_1) to cover all subsequent demand periods by setting $Q_t = 0$ for $t = 2,\ldots,T$, which is simply a special case of [AON-MP].

It is important to observe that each scenario ω specifies a demand realization for all orders in all periods. Therefore, there exists significant redundancy in this problem formulation. In particular, consider two scenarios that coincide for all order realizations up to period t. Because the variables representing the order selection and order quantity decisions are set in the first stage and therefore *independent* of the scenario ω, the inventory and backlogging variables and flow balance constraints (11) up to and including period t are identical for these two scenarios. Explicitly imposing the (redundant) nonanticipativity constraints that would enforce this yields that [AON-MP] could be reformulated to eliminate such redundancies in constraint set (11). For notational convenience, however, we will not explicitly provide the reduced formulation. In either case, the dimensionality of the MIP increases very rapidly in the number of time periods and therefore quickly becomes intractable, as we saw for the analogous MIP formulations of [AON] and [AON-PWL]. We therefore focus again on developing a tailored cutting plane method for [AON-MP].

3.3. A Cutting Plane Algorithm for the Multiperiod Problem

For the purposes of extending our cutting plane algorithm to a multiperiod setting it is convenient to reformulate the [AON-MP] in a similar form as [AON] by eliminating the inventory and backlogging amounts, yielding a nonlinear formulation where the only decision variables are the procurement amounts $Q_t(t = 1,\ldots,T)$ and the order selection variables $y_{it}(i = 1,\ldots,n_t, t = 1,\ldots,T)$. In particular, using constraints (11) and (12) we can express the inventory and backorder variables as follows:

$$I_t^\omega = \max \left(0, I_0 - B_0 + \sum_{\tau=1}^{t} Q_\tau - \sum_{\tau=1}^{t} \sum_{i=1}^{n_t} d_{i\tau} y_{i\tau} \xi_{i\tau}^\omega \right) \quad t = 1,\ldots,T; \omega = 1,\ldots,\Omega$$

$$B_t^\omega = \max \left(0, B_0 - I_0 + \sum_{\tau=1}^{t} \sum_{i=1}^{n_t} d_{i\tau} y_{i\tau} \xi_{i\tau}^\omega - \sum_{\tau=1}^{t} Q_\tau \right) \quad t = 1,\ldots,T; \omega = 1,\ldots,\Omega.$$

Then, we can substitute into the objective function of [AON-MP] and reduce the expression to:

$$\sum_{t=1}^{T} \left(\sum_{i=1}^{n} (r_{it} d_{it} p_{it} - S_{it}) y_{it} - c_t Q_t \right) - \sum_{t=1}^{T} v_t \sum_{\omega=1}^{\Omega} P_\omega \max \left(0, I_0 - B_0 + \sum_{\tau=1}^{t} Q_\tau - \sum_{\tau=1}^{t} \sum_{i=1}^{n} d_{i\tau} y_{i\tau} \xi_{i\tau}^\omega \right) - \sum_{t=1}^{T} e_t \sum_{\omega=1}^{\Omega} P_\omega \max \times \left(0, B_0 - I_0 + \sum_{\tau=1}^{t} \sum_{i=1}^{n} d_{i\tau} y_{i\tau} \xi_{i\tau}^\omega - \sum_{\tau=1}^{t} Q_\tau \right)$$

$$
\begin{aligned}
&= \sum_{t=1}^{T}\left(\sum_{i=1}^{n}(r_{it}d_{it}p_{it} - S_{it})y_{it} - c_tQ_t\right.\\
&\qquad \left.- v_t\left(I_0 - B_0 + \sum_{\tau=1}^{t}Q_\tau - \sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}p_{i\tau}\right)\right)\\
&\qquad - \sum_{t=1}^{T}(e_t + v_t)\sum_{\omega=1}^{\Omega}P_\omega \max\left(0, B_0 - I_0\right.\\
&\qquad \left.+\sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}\xi_{i\tau}^{\omega} - \sum_{\tau=1}^{t}Q_\tau\right).
\end{aligned}
$$

This leads to the following formulation of [AON-MP]:

$$
\begin{aligned}
\text{maximize}\quad &\sum_{t=1}^{T}\left(\sum_{i=1}^{n}(r_{it}d_{it}p_{it} - S_{it})y_{it} - c_tQ_t\right.\\
&\qquad \left.- v_t\left(I_0 - B_0 + \sum_{\tau=1}^{t}Q_\tau - \sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}p_{i\tau}\right)\right)\\
&\qquad - \sum_{t=1}^{T}(e_t + v_t)\sum_{\omega=1}^{\Omega}P_\omega \max\left(0, B_0 - I_0\right.\\
&\qquad \left.+\sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}\xi_{i\tau}^{\omega} - \sum_{\tau=1}^{t}Q_\tau\right)\\
\text{subject to}\quad &\\
&Q_t \geq 0 \quad t = 1, \ldots, T\\
&y_{it} \in \{0,1\} \quad i = 1, \ldots, n_t; t = 1, \ldots, T.
\end{aligned}
$$

We next introduce variables θ_t corresponding to the expected product underage (or backlog) level in period t $(t = 1, \ldots, T)$. This leads to

$$
\begin{aligned}
\text{maximize} &\sum_{t=1}^{T}\sum_{i=1}^{n}(r_{it}d_{it}p_{it} - S_{it})y_{it} - c_tQ_t\\
&\quad - v_t\left(I_0 - B_0 + \sum_{\tau=1}^{t}Q_\tau - \sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}p_{i\tau}\right)\\
&\quad - (e_t + v_t)\theta_t\\
\text{subject to}&\\
&\theta_t \geq \sum_{\omega=1}^{\Omega}P_\omega \max\left(0, B_0 - I_0\right.\\
&\qquad \left.+\sum_{\tau=1}^{t}\sum_{i=1}^{n}d_{i\tau}y_{i\tau}\xi_{i\tau}^{\omega} - \sum_{\tau=1}^{t}Q_\tau\right) t = 1, \ldots, T\\
&Q_t \geq 0 \quad t = 1, \ldots, T\\
&y_{it} \in \{0,1\} \quad i = 1, \ldots, n; t = 1, \ldots, T.
\end{aligned}
$$

Comparing this formulation to the analogous formulation of [AON], it follows that we obtain one additional constraint for each period. The constraint corresponding to period t can be replaced by 2^{Ω} linear constraints, parameterized by binary vectors $\delta_t (t = 1, \ldots, T)$:

$$
\begin{aligned}
\theta_t &\geq \sum_{\omega=1}^{\Omega}P_\omega\left(B_0 - I_0 + \sum_{\tau=1}^{t}\sum_{i=1}^{n_t}d_{i\tau}y_{i\tau}\xi_{i\tau}^{\omega} - \sum_{\tau=1}^{t}Q_\tau\right)\delta_{\omega t}\\
&\qquad\qquad \delta_t \in \{0,1\}^{\Omega}\\
&= \sum_{\tau=1}^{t}\sum_{i=1}^{n_t}\left(\sum_{\omega=1}^{\Omega}P_\omega\xi_{i\tau}^{\omega}\delta_{\omega t}\right)d_{i\tau}y_{i\tau} - \left(\sum_{\omega=1}^{\Omega}P_\omega\delta_{\omega t}\right)\\
&\qquad \times\left(\sum_{\tau=1}^{t}Q_\tau - B_0 + I_0\right) \quad \delta_t \in \{0,1\}^{\Omega}. \quad (13)
\end{aligned}
$$

Similarly to the previous problems studied in this paper, the most restrictive constraints with respect to a given solution (Q, y) in (13) are given by

$$
\delta_{\omega t} = \begin{cases} 1 & \text{if } \sum_{\tau=1}^{t}\sum_{i=1}^{n_t}d_{i\tau}y_{i\tau}\xi_{i\tau}^{\omega} > \sum_{\tau=1}^{t}Q_\tau - B_0 + I_0\\ 0 & \text{otherwise}\end{cases}
$$
$$
\omega = 1, \ldots, \Omega; t = 1, \ldots, T.
$$

Thus, for each period t, we have to find $\sum_{\tau=1}^{t}n_\tau + 1$ such coefficients for Q_t and $d_{it}y_{it}$ in (13).

Using the same approach to constructing the coefficients as before, we can search the binary tree representing all potential demand scenarios. First, consider that a given scenario is no more than a realization of some subset of orders, for all of the periods in the planning horizon. However, orders that occur in any period after t cannot be included in the calculation of inventory or backlog quantities in period t. Therefore, we can construct the period t constraint using a tree that contains all selected orders up to and including period t, and we would perform a total of T such binary tree searches. In building the tree for generating the period t constraint, we can view the orders independently of the period in which they occur. From Theorem 2.1, we would rank the orders in nonincreasing order of $d_{i\tau}\tilde{y}_{i\tau} (\tau = 1, \ldots, t)$ to find the most restrictive period t constraint in minimum computation time. We can then map the order sizes (d_{it}) and selection decisions $(\tilde{y}_{it})$ onto one-dimensional arrays $(d_i'$ and $\tilde{y}_i'$ for $i = 1, \ldots, N)$ such that $d_1'\tilde{y}_1' \geq d_2'\tilde{y}_2' \geq \cdots \geq d_N'\tilde{y}_N'$. Then, we can use either the Forward or Backward Algorithm from Section 2.3 to determine the coefficients, with an appropriate adjustment of Q.

Alternatively, we could consider constructing a single tree search that determines the coefficients of all constraints in parallel by simultaneously considering all N potential orders. In this case, we could sort all N potential orders by nonincreasing value of $d_{it}\tilde{y}_{it}$ or, alternatively, sort all orders within

Table 1. Algorithm performance for [AON].

			Our algorithm				
	CPLEX		LP (LSM)		IP (ILSM)		
n	LP (sec)	IP (sec)	(sec)	No. of cuts	(sec)	No. of cuts	No. of orders selected
5	< 0.01	< 0.01	0.02	4	0.02	4	2
10	0.07	0.12	0.06	11	0.06	8	5
15	3.42	12.05	0.12	20	0.08	12	8
20	n/a	n/a	0.17	26	0.12	14	12
30	n/a	n/a	1.57	56	0.53	26	18
40	n/a	n/a	20.67	57	12.36	34	24
50	n/a	n/a	2951.00	65	2137.00	37	31

a given period t by nonincreasing value of $d_{ik}\tilde{y}_{ik}$ and consider the periods sequentially from $t = 1, \dots, T$. Perhaps surprisingly, as we will briefly discuss in Section 4.4, the approach where the constraint coefficients for the different periods are constructed independently consistently outperforms simultaneous constraint construction approach. We therefore omit the details of the algorithm for simultaneous constraint coefficient generation.

4. COMPUTATIONAL TESTS AND RESULTS

4.1. Results: Single-Period Model - [AON]

In this section, we will demonstrate the power of our algorithm as compared to solving the MIP formulation of [AON] using CPLEX Version 10 (from within OPL Studio). All tests were conducted on a machine with a 2.0 GHz Core 2 Duo processor and 2 GB of RAM. For the implementation of our cutting plane algorithm, we used CPLEX 10 with Concert Technology to solve all linear (in case of the LSM) or mixed-integer linear (in case of the ILSM) subproblems. We considered problem instances ranging in size from 5 to 50 potential orders. Unit revenue for the orders were drawn independently from the uniform distribution on [275, 325], denoted by $U[275, 325]$, and the production cost, expediting cost, and salvage values were set to be 200, 500, and 150, respectively. The fixed costs associated with each order, which will mainly respresent salesforce allocation, are drawn from U[2500, 7500]. The potential order sizes (or demands) were generated from a $U[100, 200]$ distribution, while the associated probabilities of realization were drawn from $U[0, 1]$. We generated 50 random problem instances for each problem size.

To identify the critical component of the solution process, we evaluated the performance of the ILSM applied to [AON] as well as the performance of the LSM applied to its linear relaxation. Table 1 presents the solution times required to find the optimal solution to the problem, along with the average number of cuts added by the cutting plane algorithm.

A straightforward application of CPLEX is able to solve both [AON] and its linear relaxation with up to 15 orders in reasonable time. However, because a direct solution using CPLEX requires complete enumeration of all possible scenarios, solving larger problems becomes intractable (e.g., for a 20-order problem there are over one million scenarios, and the input data file for the corresponding optimization problem requires 85 MB of disk space). In contrast, our cutting-plane algorithm can solve [AON] with up to 40 orders (having over 1 trillion scenarios) in about the same time as CPLEX requires to solve such problems with only 15 orders (with about 32,000 scenarios). It is interesting that, using our algorithm, [AON] itself is solved more rapidly than its linear relaxation. In addition, note that the number of cuts required to identify the optimal solution is quite small. This is important in maintaining reasonable solution times, since constraint generation time can be quite long. In fact, the average CPLEX solution time within our algorithm is practically negligible (less than one second for the 50-order problems). Thus, the solution times reported in Table 1 represent almost all constraint generation time.

The last column in the table shows the average number of selected orders in the optimal IP solution. This turns out to be a critical parameter in determining the size of problems that can be solved efficiently, since the performance of our algorithm is, for large instances, dominated by the time required to construct the constraint coefficients. In particular, finding the best cutting plane requires searching a binary tree with depth equal to the number of selected orders in the incumbent solution. We performed benchmarking tests to determine when to use the Forward or Backward Algorithms for constraint generation. Given a current solution $(\tilde{Q}, \tilde{y})$, the longest computation times occurred when $\tilde{Q}$ is at or near $\sum_{i=1}^{n} d_i\tilde{y}_i/2$. As $\tilde{Q}$ increased, the performance of the Forward Algorithm improved when compared to the Backward Algorithm. In fact, for $\tilde{Q} > \sum_{i=1}^{n} d_i\tilde{y}_i/2$, the Forward Algorithm outperforms the Backward Algorithm. In the Forward Algorithm, the coefficients are updated when we prune and *include* the current subtree. This occurs when the lower bound of $\Delta > \tilde{Q}$ is violated (see Section 2.3.1). We will prune more often due

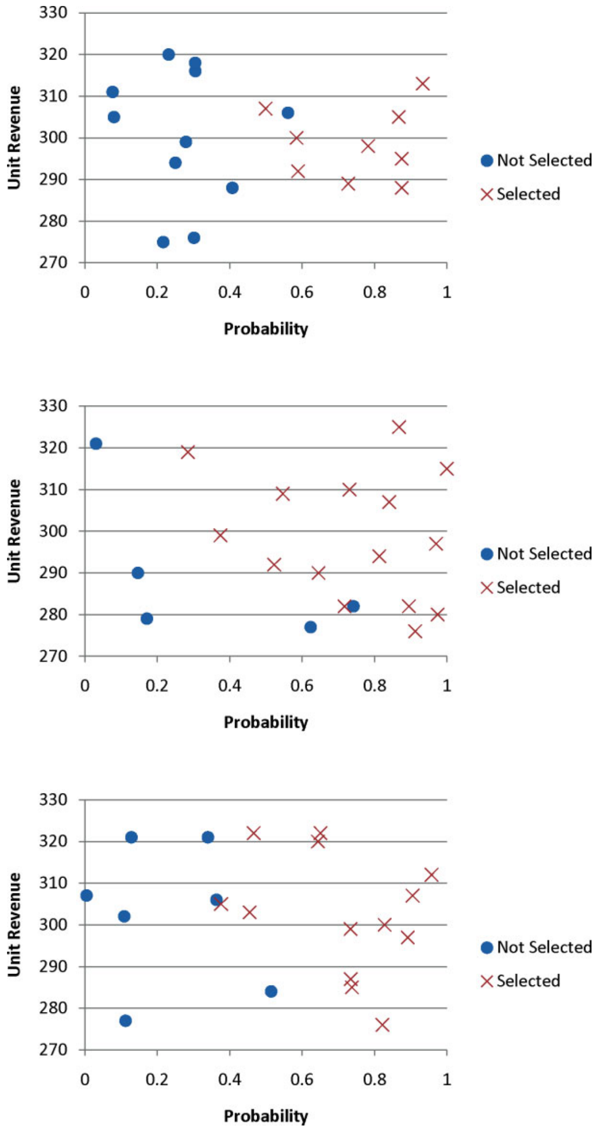

Figure 1. The impact of order characteristics on order selection. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

to this lower bound when $Q \le \sum_{i=1}^{n} d_i \tilde{y}_i/2$, which means the coefficients are updated more frequently across this range and the Backward Algorithm will outperform the Forward Algorithm. We will use this current solution information in selecting the appropriate algorithm to use.

Recall that we were interested in identifying order characteristics that affect the acceptance/rejection decision. Although unit revenue, fixed sales costs, demand size, and order likelihood all play a role, the results indicate that the probability that the order will materialize (or *order likelihood*) is the key determinant. Figure 1 presents an accept/reject classification of all orders for three unique test instances, where order likelihood is plotted against unit revenue for each order. In each of the three examples shown, there is a definite pattern for choosing orders. Notice, though, that the pattern changes across the examples, and in some cases orders with low unit revenue will still be selected if the order likelihood is high.

4.2. Comparison: Heuristic Approach vs. Optimal ILSM Approach

As discussed in Section 4.1, the ability to solve [AON] to optimality is limited by the number of orders selected during constraint generation, and as problem size grows, this limit will be reached quickly. To address this, we tested the ILSM approach against a heuristic approach that first (1) creates a rough forecast and then (2) builds to the forecast. The steps are described below.

Heuristic Algorithm to Find (Q, y)

1. Define the set of orders I such that $I = \{i \mid \frac{S_i}{p_i d_i} + c \le r_i\}$. Pursue all orders in I. Note that I includes all orders in which an approximation of the highest per-unit cost is less than the per-unit revenue.
2. Set Q by solving the standard newsvendor problem. We can find the exact value of Q based on the critical fractile value $(e - c)/(e - v)$ of the underlying demand distribution (see Taaffe et al. [18] and Taaffe and Romeijn [19] for further discussion). We can actually describe the cumulative demand distribution as a convolution of the distributions for the independent Bernoulli demands for each order (see, e.g., Kaplan and Barnett [8]).

Using the same set of 400 test instances as described in Section 4.1, we recorded the quality of the solutions achieved by the heuristic. Table 2 presents the number (and percent) of orders selected for both the ILSM and heuristic, as well as average and maximum optimality gaps for the heuristic approach only.

To find the expected profit that results from the heuristic solution for (Q, y), we can use one of two methods. For heuristic solutions with less than 30 orders, we can fix Q and

Table 2. Heuristic vs. ILSM for [AON].

	Orders selected				Optimality gap	
	ILSM		Heuristic		Average	Maximum
n	No	(%)	No	(%)	(%)	(%)
5	1.9	38	3.0	60	21.0	100.0
10	4.9	49	6.2	62	13.1	54.9
15	8.4	56	9.9	66	3.6	13.2
20	11.9	60	13.3	66	1.9	5.3
25	14.9	60	17.1	68	1.3	3.8
30	17.6	59	19.2	64	1.1	6.4
40	24.1	60	26.1	65	0.6	2.0
50	30.6	61	32.8	66	0.5	1.6

Table 3. Heuristic solution quality for test instances with small fixed costs.

	Optimality gap	
n	Average (%)	Maximum (%)
5	6.7	100.0
10	0.0	0.2
15	0.1	1.3
20	0.1	0.7
25	0.0	0.0
30	0.0	0.0
40	0.0	0.0
50	0.0	0.0

y and solve using the ILSM approach, and this will provide an expected profit as the objective function. This is very efficient and quick for problems of this size. For heuristic solutions with more than 30 orders, we estimate the expected profit with simulation.

While the heuristic performs poorly for small problems, it provides a good approximation for the optimal solution in larger problems (< 1% gap for 40- and 50-order problems). It is precisely for these problem sizes that the heuristic approach could be necessary. If over 30 orders are selected in the optimal solution, the exact ILSM approach becomes difficult to solve in a reasonable amount of time. The heuristic consistently selects between 60 and 70% of the orders, which is slightly more than the optimal ILSM approach.

Depending on the nature of the firm's business, it may not be necessary to allocate large fixed salesforce costs to help secure orders. Based on the test instances described in Section 4.1, on average, net revenue without setup is \$15,000 and the fixed cost is \$5000 (or 33% of this revenue). We conducted an experiment using similar test instances, except now the average setup cost was reduced to 10% of net revenue. Table 3 presents the solution quality of the heuristic approach using these new test instances.

From this analysis, we can say that the heuristic error depends to a large extent on the magnitude of the S_i values, and the heuristic does perform well under certain circumstances. Thus, not only can the heuristic be applied when the optimal ILSM approach reaches computational limitations, but the heuristic performs extremely well when fixed order costs are small. It should be noted that there may be other parameter settings in which the heuristic still has difficulty identifying high quality solutions, but the results from these tests are very encouraging.

4.3. Results: Single-Period Model with Piecewise Linear Costs - [AON-PWL]

We now describe computational tests using CPLEX and our algorithms to solve [AON-PWL]. For the most part we used the same order-specific data as in Section 4.1, but we now include several breakpoints in the underage cost and overage revenue functions. In particular, when the procurement quantity falls short of realized demand by 0/150/300 units, the firm incurs a unit expediting cost of 350/500/750, respectively. Similarly, when the procurement quantity exceeds realized demand by 0/150/300 units, the firm can obtain a unit salvage value for the product of 150/100/50, respectively. We again generated 10 random problem instances for each test case, and Table 4 summarizes the results.

As for [AON], we cannot obtain an optimal solution to [AON-PWL] via CPLEX when the number of orders is greater than 15. Using our cutting plane algorithm, we were able to solve problems with almost three times as many orders very efficiently. Note that Table 4 reports the total number of cuts generated by the algorithm. That is, because 5 cuts are generated per iteration, the number of iterations is in fact only 20% of the number of cuts generated.

4.4. Results: Multiperiod Model - [AON-MP]

Our final set of computational results summarizes the effectiveness of our solution approach to the multiperiod problem [AON-MP]. We used the following parameters to again generate 10 test instances for each problem size. The unit revenues and fixed order costs are generated as for [AON]. Furthermore, in each period, the unit procurement costs were generated from a $U[190,210]$ distribution whereas the holding cost and backorder costs were drawn from a $U[3,7]$ and $U[10,15]$ distribution, respectively. In the final period, the overage cost was set to 500 and the salvage value to 100. Table 5 presents the results obtained with both CPLEX and our cutting plane algorithm.

For [AON-MP], we again need to restrict ourselves to very small problem instances to obtain solutions via CPLEX. In fact, we were able to solve problems with a total of approximately $N = 12$ orders in all periods combined, as compared with instances of [AON] and [AON-PWL] with up to $n = 15$ orders. Problem sizes with 15 or more orders could

Table 4. Algorithm performance for [AON-PWL].

	CPLEX	Our algorithm		
		IP		
n	IP (sec)	(sec)	No. of cuts	No. of orders selected
5	0.02	0.08	25	3
10	1.96	0.27	50	5
15	1794.00	0.38	70	7
20	n/a	0.68	95	11
30	n/a	21.90	320	15
40	n/a	149.40	430	21

Table 5. Algorithm performance for [AON-MP].

			Our algorithm		
			IP		
		CPLEX			No. of orders
n	T	IP (sec)	(sec)	No. of cuts	selected
2	4	0.10	0.11	28	4
2	5	2.46	0.20	45	5
2	6	29.10	0.30	66	7
3	3	0.24	0.10	21	5
3	4	13.42	0.21	44	8
3	5	n/a	0.28	60	9
3	6	n/a	0.58	102	10
4	3	11.82	0.16	30	7
4	4	n/a	0.34	56	9
4	5	n/a	0.32	65	12
6	4	n/a	0.59	76	14
6	5	n/a	1.1	95	18
6	6	n/a	12.4	144	21
8	4	n/a	1.7	76	19
8	5	n/a	17.1	160	22
8	6	n/a	752.0	108	29
10	4	n/a	34.5	108	23
10	5	n/a	2694.0	150	30

not be solved due to insufficient memory. Our cutting plane algorithm, however, shows about the same performance on [AON-MP] as on [AON] and [AON-PWL] as a function of the total number of orders. We also point out that the actual number of iterations is No. of cuts/T, because our multiperiod algorithm adds T cuts in each iteration.

5. CONCLUDING REMARKS

In this article, we have introduced a new approach to order management problems in which each of a collection of potential orders will either materialize at a known level or not materialize at all. We presented several models, including a single period model with linear and nonlinear overage revenues and underage cost functions as well as a multiperiod model. For all models we developed a tailored cutting plane algorithm based on the L-shaped method for two-stage stochastic programming. Extensive numerical experiments show that our algorithm significantly outperforms the CPLEX MIP solver in the sense that we are able to solve much larger instances of the problem to optimality in a reasonable amount of time. In particular, we are able to efficiently solve problems that contain three times the number of potential orders than the maximum that can be handled by CPLEX. We also propose a heuristic approach that provides average gaps of less than 1% for the largest problems that can be solved exactly. Because our experiments show that the error gap decreases as the problem size grows, the heuristic approach can be expected to work well for large problem instances.

A natural extension of the research presented in this article would be to develop a heuristic approach to solving problems with more orders (for the single-period problems) and more orders and periods (for the multiperiod problem). We could also extend the algorithm to the case of multiple level orders, where any order comes in at one of several pre-defined levels or not at all.

A major focus of our future research will be to enhance the models by considering the effect that targeted pricing and advertising has on order sizes and their likelihood of occurrence. In particular, we plan to investigate how pricing and advertising can help shape the demand distribution that the supplier will face. We will in this context also consider the effect of limited capacities, which in a manufacturing setting will likely be fixed and cannot be influenced in the short term, since this may limit the supplier's ability to increase profits through demand shaping. Moreover, opportunities for marketing may be limited by the amount of funds available for this purpose. An additional consideration in a multiperiod setting is the possibility that an order could materialize in a different time period than the one originally prescribed.

APPENDIX

Extension to Piecewise-Linear Cost Functions

In this section, we will examine how [AON] can be generalized to allow for more general, in particular piecewise-linear convex, shortage and overage cost functions (where, for convenience, we will in this section refer to the salvage revenue functions as (negative) overage cost functions). In other words, as the shortage or overage increases, the corresponding marginal unit cost is non-decreasing, representing the fact that the unit salvage value may decrease as the quantity salvaged increases and, similarly, the unit expediting cost may increase as the quantity expedited increases.

Problem Formulation

Let the marginal shortage costs and salvage values be given by e_j $(j = 1, \dots, J^s + 1)$ and v_j $(j = 1, \dots, J^o + 1)$ where $v_{J^o+1} < \cdots < v_1 < c < e_1 < \cdots < e_{J^s+1}$. For convenience, we will also let $e_0 = v_0 = 0$. Finally, denote the corresponding breakpoints by s_j $(j = 1, \dots, J^s)$ and o_j $(j = 1, \dots, J^o)$, respectively, where $0 = s_0 < s_1 < \cdots < s_{J^s}$ and $0 = o_0 < o_1 < \cdots < o_{J^o}$. (Note that if $J^o = J^s = 0$ then we obtain [AON].) For convenience, we will let $J = 1 + J^o + J^s$.

The total expected profit can now be written as:

$$\begin{aligned} G(Q, y) &= \sum_{i=1}^{n} (r_i d_i p_i - S_i) y_i - cQ \\ &\quad + \sum_{\omega=1}^{\Omega} P_\omega \sum_{j=0}^{J^o} (v_{j+1} - v_j) \max\left(0, Q - \sum_{i=1}^{n} d_i y_i - o_j\right) \\ &\quad - \sum_{\omega=1}^{\Omega} P_\omega \sum_{j=0}^{J^s} (e_{j+1} - e_j) \max\left(0, \sum_{i=1}^{n} d_i y_i - Q - s_j\right) \\ &= \sum_{i=1}^{n} ((r_i - v_1) d_i p_i - S_i) y_i - (c - v_1) Q \end{aligned}$$

$$
\begin{aligned}
&- (e_1 - v_1)\sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i=1}^{n} d_i y_i - Q\right)\\
&- \sum_{i=1}^{J^o}(v_j - v_{j+1})\sum_{\omega=1}^{\Omega} P_\omega \max\left(0, Q - \sum_{i=1}^{n} d_i y_i - o_j\right)\\
&- \sum_{i=1}^{J^s}(e_{j+1} - e_j)\sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i=1}^{n} d_i y_i - Q - s_j\right).
\end{aligned}
$$

We refer to the resulting optimization problem as the *newsvendor problem with all-or-nothing demand and piecewise linear costs* [AON-PWL]. As for [AON], we can formulate [AON-PWL] as a MIP:

$$
\begin{aligned}
\text{maximize}\quad & \sum_{i=1}^{n}((r_i - v_1)d_i p_i - S_i)y_i - (c - v_1)Q - (e_1 - v_1)\sum_{\omega=1}^{\Omega} P_\omega u_\omega\\
& \quad - \sum_{j=1}^{J^o}(v_j - v_{j+1})\sum_{\omega=1}^{\Omega} P_\omega u^o_{j\omega} - \sum_{j=1}^{J^s}(e_{j+1} - e_j)\sum_{\omega=1}^{\Omega} P_\omega u^s_{j\omega}\\
\text{subject to}\quad & u_\omega \ge \sum_{i\in I_\omega} d_i y_i - Q \quad \omega = 1,\dots,\Omega\\
& u^o_{j\omega} \ge Q - \sum_{i\in I_\omega} d_i y_i - o_j \quad j = 1,\dots,J^o; \omega = 1,\dots,\Omega\\
& u^s_{j\omega} \ge \sum_{i\in I_\omega} d_i y_i - Q - s_j \quad j = 1,\dots,J^s; \omega = 1,\dots,\Omega\\
& u_\omega \ge 0 \quad \omega = 1,\dots,\Omega\\
& u^o_{j\omega} \ge 0 \quad j = 1,\dots,J^o; \omega = 1,\dots,\Omega\\
& u^s_{j\omega} \ge 0 \quad j = 1,\dots,J^s; \omega = 1,\dots,\Omega\\
& Q \ge 0\\
& y_i \in \{0,1\} \quad i = 1,\dots,n.
\end{aligned}
$$

The dimensionality of this MIP increases linearly in the number of segments in the cost functions. It thus suffers from the same drawbacks as the MIP for [AON]. In the remainder of this section, we will generalize the cutting plane algorithm to [AON-PWL].

A Cutting Plane Algorithm Under Piecewise-Linear Costs

We follow the approach used for [AON] and introduce three (sets of) decision variables, each corresponding to one of the expected values in the objective function of [AON-PWL]: θ_1, θ^o_j ($j = 1,\dots,J^o$), and θ^s_j ($j = 1,\dots,J^s$). Using these decision variables we reformulate [AON-PWL] as

$$
\begin{aligned}
\text{maximize}\quad & \sum_{i=1}^{n}((r_i - v_1)d_i p_i - S_i)y_i - (c - v_1)Q - (e_1 - v_1)\theta_1\\
& \quad - \sum_{j=1}^{J^o}(v_j - v_{j+1})\theta^o_j - \sum_{j=1}^{J^s}(e_{j+1} - e_j)\theta^s_j\\
\text{subject to}\quad & \theta_1 \ge \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i\in I_\omega} d_i y_i - Q\right)\\
& \theta^o_j \ge \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, Q - \sum_{i\in I_\omega} d_i y_i - o_j\right) \quad j = 1,\dots,J^o\\
& \theta^s_j \ge \sum_{\omega=1}^{\Omega} P_\omega \max\left(0, \sum_{i\in I_\omega} d_i y_i - Q - s_j\right) \quad j = 1,\dots,J^s\\
& Q \ge 0\\
& y_i \in \{0,1\} \quad i = 1,\dots,n.
\end{aligned}
$$

Comparing this formulation to the analogous formulation of [AON], we obtain one additional variable and constraint for each breakpoint in the overage or underage cost function. Each of these constraints can again be replaced by 2^Ω linear constraints, parameterized by binary vectors δ^1, δ^o_j, and δ^s_j:

$$
\begin{aligned}
\theta^1 &\ge \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^{n} d_i y_i \xi^\omega_i - Q\right)\delta^1_\omega \qquad \delta^1 \in \{0,1\}^\Omega\\
&= \sum_{i=1}^{n}\left(\sum_{\omega=1}^{\Omega} P_\omega \xi^\omega_i \delta^1_\omega\right) d_i y_i - \left(\sum_{\omega=1}^{\Omega} P_\omega \delta^1_\omega\right) Q \quad \delta^1 \in \{0,1\}^\Omega \qquad (14)\\
\theta^o_j &\ge \sum_{\omega=1}^{\Omega} P_\omega \left(Q - \sum_{i=1}^{n} d_i y_i \xi^\omega_i - o_j\right)\delta^o_{j\omega} \quad \delta^o_j \in \{0,1\}^\Omega; j = 1,\dots,J^o\\
&= \left(\sum_{\omega=1}^{\Omega} P_\omega \delta^o_{j\omega}\right)(Q - o_j) - \sum_{i=1}^{n}\left(\sum_{\omega=1}^{\Omega} P_\omega \xi^\omega_i \delta^o_{j\omega}\right) d_i y_i\\
&\qquad \delta^o_j \in \{0,1\}^\Omega; j = 1,\dots,J^o \qquad (15)\\
\theta^s_j &\ge \sum_{\omega=1}^{\Omega} P_\omega \left(\sum_{i=1}^{n} d_i y_i \xi^\omega_i - Q - s_j\right)\delta^s_{j\omega} \quad \delta^s_j \in \{0,1\}^\Omega; j = 1,\dots,J^s\\
&= \sum_{i=1}^{n}\left(\sum_{\omega=1}^{\Omega} P_\omega \xi^\omega_i \delta^s_{j\omega}\right) d_i y_i - \left(\sum_{\omega=1}^{\Omega} P_\omega \delta^s_{j\omega}\right)(Q + s_j)\\
&\qquad \delta^s_j \in \{0,1\}^\Omega; j = 1,\dots,J^s. \qquad (16)
\end{aligned}
$$

Analogous to [AON], we immediately see that the most restrictive constraints with respect to a given solution (Q, y) in (14)–(16) are those for which

$$
\begin{aligned}
\delta^1_\omega &= \begin{cases} 1 & \text{if } \sum_{i\in I_\omega} d_i y_i > Q\\ 0 & \text{otherwise}\end{cases} && \omega = 1,\dots,\Omega && (17)\\
\delta^o_{j\omega} &= \begin{cases} 1 & \text{if } \sum_{i\in I_\omega} d_i y_i < Q - o_j\\ 0 & \text{otherwise}\end{cases} && \omega = 1,\dots,\Omega; j = 1,\dots,J^o && (18)\\
\delta^s_{j\omega} &= \begin{cases} 1 & \text{if } \sum_{i\in I_\omega} d_i y_i > Q + s_j\\ 0 & \text{otherwise}\end{cases} && \omega = 1,\dots,\Omega; j = 1,\dots,J^s. && (19)
\end{aligned}
$$

From these it is easy to see that the coefficients in constraint (14) can be identified by the Forward and Backward Algorithms described in Section 2.3. Similarly, the coefficients in constraints (16) can be identified by the same algorithms with an appropriate modification of Q. Now note that we can rewrite constraints (15) as follows:

$$
\begin{aligned}
\theta^o_j \ge{} & \left(\sum_{\omega=1}^{\Omega} P_\omega - \sum_{\omega=1}^{\Omega} P_\omega\big(1 - \delta^o_{j\omega}\big)\right)(Q - o_j) - \sum_{i=1}^{n}\left(\sum_{\omega=1}^{\Omega} P_\omega \xi^\omega_i\right.\\
& \left. - \sum_{\omega=1}^{\Omega} P_\omega \xi^\omega_i \big(1 - \delta^o_{j\omega}\big)\right) d_i y_i \quad \delta^o_j \in \{0,1\}^\Omega; j = 1,\dots,J^o
\end{aligned}
$$

$$= \left(1 - \sum_{\omega=1}^{\Omega} P_\omega - \left(1 - \delta^o_{j\omega}\right)\right)(Q - o_j)$$
$$- \sum_{i=1}^{n}\left(p_i - \sum_{\omega=1}^{\Omega} P_\omega \xi_i^\omega\left(1 - \delta^o_{j\omega}\right)\right) d_i y_i \quad \delta^o_j \in \{0,1\}^\Omega; j = 1, \ldots, J^o \tag{20}$$

or, equivalently, as

$$\theta^o_j \geq \left(1 - \sum_{\omega=1}^{\Omega} P_\omega \bar{\delta}^o_{j\omega}\right)(Q - o_j) - \sum_{i=1}^{n}\left(p_i - \sum_{\omega=1}^{\Omega} P_\omega \xi_i^\omega \bar{\delta}^o_{j\omega}\right) d_i y_i$$
$$\bar{\delta}^o_j \in \{0,1\}^\Omega; j = 1, \ldots, J^o \tag{21}$$

where the dummy indicator variables $\bar{\delta}^o_j$ in (21) are precisely the complements of the indicator variables δ^o_j in (20). A most restrictive constraint is now the one given by

$$\bar{\delta}^o_{j\omega} = \begin{cases} 1 & \text{if } \sum_{i \in I_\omega} d_i y_i > Q - o_j \\ 0 & \text{otherwise} \end{cases} \quad \omega = 1, \ldots, \Omega; j = 1, \ldots, J^o$$

where we have used the observation following Eq. (4) to replace the weak inequality by a strict inequality. The coefficients in constraint (21) can thus also be identified by the Forward and Backward Algorithms with an appropriate modification of Q and post-processing. We thus perform J binary tree searches to implicitly identify, in each iteration of the cutting plane method, all scenarios that determine a particular constraint.

ACKNOWLEDGMENTS

The work of H. Edwin Romeijn was supported by the National Science Foundation under Grant No. DMI-0355533. The authors also thank the reviewers for their insightful comments and contributions to our work.

REFERENCES

[1] J. Birge and F. Louveaux, Introduction to stochastic programming, Springer Series in Operations Research, New York, New York, 1997.

[2] C. Cachon, "Competitive supply chain inventory management," Quantitative models for supply chain management, S. Tayur, R. Ganeshan, and M. Magazine (Editors), Kluwer, Boston, 1999.

[3] S. Carr and I. Duenyas, Optimal admission control and sequencing in a make-to-stock/make-to-order production system, Oper Res 48 (2000), 709–720.

[4] S. Carr and W. Lovejoy, The inverse newsvendor problem: Choosing an optimal demand portfolio for capacitated resources, Management Sci 46 (2000), 912–927.

[5] F. de Véricourt, F. Karaesmen, and Y. Dallery, Optimal stock allocation for a capacitated supply system, Management Sci 48 (2002), 1486–1501.

[6] D. Gupta and L. Wang, Capacity management for contract manufacturing, Oper Res 55 (2007), 367–377.

[7] A.Y. Ha, Optimal dynamic scheduling policy for a make-to-stock production system, Oper Res 45 (1997), 42–53.

[8] E.H. Kaplan and A. Barnett, A new approach to estimating the probability of winning the presidency, Oper Res 51 (2003), 32–40.

[9] P. Kouvelis and G. Gutierrez, The newsvendor problem in a global market: Optimal centralized and decentralized policies for a two-market stochastic inventory system, Management Sci 43 (1997), 571–585.

[10] G. Laporte and F.V. Louveaux, The integer L-shaped method for stochastic integer programs with complete recourse, Oper Res Lett 13 (1993), 133–142.

[11] G. Laporte, F.V. Louveaux, and L. Van Hamme, An integer l-shaped algorithm for the capacitated vehicle routing problem with stochastic demands, Oper Res 50 (2002), 415–423.

[12] G.E. Monahan, N.C. Petruzzi, and W. Zhao, The dynamic pricing problem from a newsvendor's perspective, Manufact Service Oper Management 6 (2004), 73–91.

[13] E.L. Porteus, "Stochastic inventory theory," Handbooks in operations research and management science, Volume 2, D. Heyman and M. Sobel (Editors), Elsevier, Amsterdam, The Netherlands, 1990, pp. 605–652.

[14] A.J. Schaefer, N. Kong, and S. Ahmed, Totally unimodular stochastic programs, 10th Inter Conf Stochastic Program, Tucson, Arizona, October 11–15, 2004.

[15] A. Sen and A.X. Zhang, The newsboy problem with multiple demand classes, IIE Trans 31 (1999), 431–444.

[16] R.A. Shumsky and F. Zhang, Dynamic capacity management with substitution, Oper Res (in press).

[17] K. Taaffe, J. Geunes, and H.E. Romeijn, Market entrance and product ordering decisions under demand uncertainty, in: proceedings of IIE research conference, Houston, Texas, 2004.

[18] K. Taaffe, J. Geunes, and H.E. Romeijn, Target market selection and market effort under uncertainty: The selective newsvendor problem, Eur J Oper Res 189 (2008), 987–1003.

[19] K. Taaffe and H.E. Romeijn, Selective newsvendor problems with all-or-nothing order requests, Proc IIE Res Conf, Atlanta, Georgia, 2005.

[20] A. Tsay, S. Nahmias, and N. Agrawal, "Modeling supply chain contracts: A review," Quantitative models for supply chain management, S. Tayur, R. Ganeshan, and M. Magazine (Editors), Kluwer, Boston, 1999.

[21] R. Van Slyke and R.J-B. Wets, L-shaped linear programs with application to optimal control and stochastic programming, SIAM J Appl Math 17 (1969), 638–663.

[22] A. Veinott, Optimal policy for a multi-product, dynamic, nonstationary inventory problem, Management Sci 12 (1965), 206–222.